CHAPTER 40

Configuring Network Security

This chapter contains network security information unique to Cisco IOS Software Release 12.2SX, which supplements the network security information and procedures in these publications:

- *Cisco IOS Security Configuration Guide*, Release 12.2, at this URL:

  http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122cgcr/fsecur_c/index.htm
- *Cisco IOS Security Command Reference*, Release 12.2, at this URL

  http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122cgcr/fsecur_r/index.htm

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

- The Cisco IOS Software Releases 12.2SX Command References at this URL:

  http://www.cisco.com/en/US/docs/ios/mcl/122sx_mcl.html
- The Release 12.2 publications at this URL:

  http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122cgcr/index.htm

This chapter consists of these sections:

Configuring MAC Address-Based Traffic Blocking

To block all traffic to or from a MAC address in a specified VLAN, perform this task:

Command	Purpose
`Router(config)# mac-address-table static mac_address vlan vlan_ID drop`	Blocks all traffic to or from the configured MAC address in the specified VLAN.
`Router(config)# no mac-address-table static mac_address vlan vlan_ID`	Clears MAC address-based blocking.

This example shows how to block all traffic to or from MAC address 0050.3e8d.6400 in VLAN 12:

```
Router# configure terminal
Router(config)# mac-address-table static 0050.3e8d.6400 vlan 12 drop
```

Configuring TCP Intercept

TCP intercept flows are processed in hardware.

For configuration procedures, see the *Cisco IOS Security Configuration Guide*, Release 12.2, "Traffic Filtering and Firewalls," "Configuring TCP Intercept (Preventing Denial-of-Service Attacks)," at this URL:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122cgcr/fsecur_c/ftrafwl/scfdenl.htm

Configuring Unicast Reverse Path Forwarding Check

These sections describe configuring Cisco IOS Unicast Reverse Path Forwarding check (Unicast RPF check):

Understanding PFC3 Unicast RPF Check Support

For a complete explanation of how Unicast RPF check works, see the *Cisco IOS Security Configuration Guide*, Release 12.2, "Other Security Features," "Configuring Unicast Reverse Path Forwarding" at this URL:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122cgcr/fsecur_c/fothersf/scfrpf.htm

The PFC3 provides hardware support for RPF check of traffic from multiple interfaces.

With strict-method Unicast RPF check, the PFC3 supports two parallel paths for all prefixes in the routing table, and up to four parallel paths for prefixes reached through any of four user-configurable RPF interface groups (each interface group can contain four interfaces).

With loose-method Unicast RPF check (also known as exist-only method), the PFC3 supports up to eight reverse-path interfaces (the Cisco IOS software is limited to eight reverse paths in the routing table).

There are four methods of performing Unicast RPF check in Cisco IOS:

- Strict Unicast RPF check
- Strict Unicast RPF check with allow-default
- Loose Unicast RPF check
- Loose Unicast RPF check with allow-default

You configure Unicast RPF check on a per-interface basis, but the PFC3 supports only one Unicast RPF method for all interfaces that have Unicast RPF check enabled. When you configure an interface to use a Unicast RPF method that is different from the currently configured method, all other interfaces in the system that have Unicast RPF check enabled use the new method.

Unicast RPF Check Guidelines and Restrictions

When configuring Unicast RPF check, follow these guidelines and restrictions:

- If you configure Unicast RPF check to filter with an ACL, the PFC determines whether or not traffic matches the ACL. The PFC sends the traffic denied by the RPF ACL to the route processor (RP) for the Unicast RPF check. Packets permitted by the ACL are forwarded in hardware without a Unicast RPF check (CSCdz35099).
- Because the packets in a denial-of-service attack typically match the deny ACE and are sent to the RP for the Unicast RPF check, they can overload the RP.
- The PFC provides hardware support for traffic that does not match the Unicast RPF check ACL, but that does match an input security ACL.
- The PFC does not provide hardware support for the Unicast RPF check for policy-based routing (PBR) traffic. (CSCea53554)

Configuring Unicast RPF Check

These sections describe how to configure Unicast RPF check:

- Configuring the Unicast RPF Check Mode, page 40-3
- Configuring the Multiple-Path Unicast RPF Check Mode on a PFC3, page 40-5
- Enabling Self-Pinging, page 40-6

Configuring the Unicast RPF Check Mode

There are two Unicast RPF check modes:

- Strict check mode, which verifies that the source IP address exists in the FIB table and verifies that the source IP address is reachable through the input port.
- Exist-only check mode, which only verifies that the source IP address exists in the FIB table.

Note The most recently configured mode is automatically applied to all ports configured for Unicast RPF check.

To configure Unicast RPF check mode, perform this task:

	Command	Purpose
Step 1	`Router(config)# interface {{vlan vlan_ID} | {type[1] slot/port} | {port-channel number}}`	Selects an interface to configure. **Note** Based on the input port, Unicast RPF check verifies the best return path before forwarding the packet on to the next destination.
Step 2	`Router(config-if)# ip verify unicast source reachable-via {rx | any} [allow-default] [list]`	Configures the Unicast RPF check mode.
	`Router(config-if)# no ip verify unicast`	Reverts to the default Unicast RPF check mode.
Step 3	`Router(config-if)# exit`	Exits interface configuration mode.
Step 4	`Router# show mls cef ip rpf`	Verifies the configuration.

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

When configuring the Unicast RPF check mode, note the following information:

- Use the **rx** keyword to enable strict check mode.
- Use the **any** keyword to enable exist-only check mode.
- Use the **allow-default** keyword to allow use of the default route for RPF verification.
- Use the *list* option to identify an access list.
  - If the access list denies network access, spoofed packets are dropped at the port.
  - If the access list permits network access, spoofed packets are forwarded to the destination address. Forwarded packets are counted in the interface statistics.
  - If the access list includes the logging action, information about the spoofed packets is sent to the log server.

Note When you enter the **ip verify unicast source reachable-via** command, the Unicast RPF check mode changes on all ports in the switch.

This example shows how to enable Unicast RPF exist-only check mode on Gigabit Ethernet port 4/1:

```
Router(config)# interface gigabitethernet 4/1
Router(config-if)# ip verify unicast source reachable-via any
Router(config-if)# end
Router#
```

This example shows how to enable Unicast RPF strict check mode on Gigabit Ethernet port 4/2:

```
Router(config)# interface gigabitethernet 4/2
Router(config-if)# ip verify unicast source reachable-via rx
Router(config-if)# end
Router#
```

This example shows how to verify the configuration:

```
Router# show running-config interface gigabitethernet 4/2
Building configuration...
Current configuration : 114 bytes
!
interface GigabitEthernet4/2
ip address 42.0.0.1 255.0.0.0
```

```
ip verify unicast reverse-path
no cdp enable
end
Router# show running-config interface gigabitethernet 4/1
Building configuration...
Current configuration : 114 bytes
!
interface GigabitEthernet4/1
ip address 41.0.0.1 255.0.0.0
ip verify unicast reverse-path (RPF mode on g4/1 also changed to strict-check RPF mode)
no cdp enable
end
Router#
```

Configuring the Multiple-Path Unicast RPF Check Mode on a PFC3

To configure the multiple-path Unicast RPF check mode on a PFC3, perform this task:

	Command	Purpose
Step 1	`Router(config)# mls ip cef rpf mpath {punt | pass | interface-group}`	Configures the multiple path RPF check mode on a PFC3.
	`Router(config)# no mls ip cef rpf mpath {punt | interface-group}`	Returns to the default (**mls ip cef rpf mpath punt**).
Step 2	`Router(config)# end`	Exits configuration mode.
Step 3	`Router# show mls cef ip rpf`	Verifies the configuration.

When configuring multiple path RPF check, note the following information:

- **punt** mode (default)—The PFC3 performs the Unicast RPF check in hardware for up to two interfaces per prefix. Packets arriving on any additional interfaces are redirected (punted) to the RP for Unicast RPF check in software.
- **pass** mode—The PFC3 performs the Unicast RPF check in hardware for single-path and two-path prefixes. Unicast RPF check is disabled for packets coming from multipath prefixes with three or more reverse-path interfaces (these packets always pass the Unicast RPF check).
- **interface-group** mode—The PFC3 performs the Unicast RPF check in hardware for single-path and two-path prefixes. The PFC3 also performs the Unicast RPF check for up to four additional interfaces per prefix through user-configured multipath Unicast RPF check interface groups. Unicast RPF check is disabled for packets coming from other multipath prefixes that have three or more reverse-path interfaces (these packets always pass the Unicast RPF check).

This example shows how to configure punt as the multiple path RPF check mode:

```
Router(config)# mls ip cef rpf mpath punt
```

Configuring Multiple-Path Interface Groups on a PFC3

To configure multiple-path Unicast RPF interface groups on a PFC3, perform this task:

	Command	Purpose
Step 1	`Router(config)# mls ip cef rpf interface-group [0 | 1 | 2 | 3] interface1 [interface2 [interface3 [interface4]]]`	Configures a multiple path RPF interface group on a PFC3.
Step 2	`Router(config)# mls ip cef rpf interface-group group_number`	Removes an interface group.
Step 3	`Router(config)# end`	Exits configuration mode.
Step 4	`Router# show mls cef ip rpf`	Verifies the configuration.

This example shows how to configure interface group 2:

```
Router(config)# mls ip cef rpf interface-group 2 fastethernet 3/3 fastethernet 3/4
fastethernet 3/5 fastethernet 3/6
```

Enabling Self-Pinging

With Unicast RPF check enabled, by default the switch cannot ping itself.

To enable self-pinging, perform this task:

	Command	Purpose
Step 1	`Router(config)# interface {{vlan vlan_ID} | {type[1] slot/port} | {port-channel number}}`	Selects the interface to configure.
Step 2	`Router(config-if)# ip verify unicast source reachable-via any allow-self-ping`	Enables the switch to ping itself or a secondary address.
	`Router(config-if)# no ip verify unicast source reachable-via any allow-self-ping`	Disables self-pinging.
Step 3	`Router(config-if)# exit`	Exits interface configuration mode.

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

This example shows how to enable self-pinging:

```
Router(config)# interface gigabitethernet 4/1
Router(config-if)# ip verify unicast source reachable-via any allow-self-ping
Router(config-if)# end
```