Default Control Plane Policy on Catalyst 6500/Sup2T and Catalyst 6880 Configuration Example

Document ID: 118806

Contributed by Mariusz Kazmierski, Cisco TAC Engineer.
Mar 04, 2015

Contents

Introduction
Prerequisites
    Requirements
    Components Used
Configure
Verify
Troubleshoot
Related Information

Introduction

This document describes in detail what types of traffic are matched against default class−maps, which are part of the default Catalyst 6500 Sup2T / Catalyst 6880 CoPP (Control Plane Policing) configuration that is automatically configured on the device. This is configured in order to protect its CPU from being overloaded.

Prerequisites

Requirements

There are no specific requirements for this document.

Components Used

This document is not restricted to specific software and hardware versions.

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, make sure that you understand the potential impact of any command.

Configure

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

If a specific class−map needs to be added / modified / removed from the current CoPP policy, then it must be done from the configuration mode in policy−map mode. See Catalyst 6500 Release 15.0SY Software Configuration Guide − Control Plane Policing (CoPP) for the exact syntax.

CoPP default exception classes have these descriptions:

Case	class−map name	Description
Maximum Transmission Unit (MTU) failure	class−copp−mtu−fail	Packet size exceeds the outgoing interface MTU size. If the Don't Fragment bit is not set, fragmentation is required. If the Don't Fragment bit is set, the Internet Control Message Protocol (ICMP) Destination Unreachable message indicates that "fragmentation needed and DF set" is supposed to be generated and sent back to the source. Reference: RFC−791, RFC−1191
Time To Live (TTL) failure	class−copp−ttl−fail	Packet TTL = 1 (for IPv4), Hop Limit = 0 or 1 (for IPv6) TTL = 0 (for IPv4) can be discarded in the hardware right away as the previous hop is supposed to destroy the packet when TTL is decremented to 0. Hop Limit = 0 (for IPv6) is different from TTL = 0 because it is stated in RFC−2460, section 8.2 that "Unlike IPv4, IPv6 nodes are not required to enforce maximum packet lifetime. That is the reason the IPv4 Time to Live field was renamed Hop Limit in IPv6". This means incoming IPv6 packet with Hop Limit = 0 is still valid, and the ICMP message should be sent back. Reference: RFC−791, RFC−2460
Options	class−copp−options	Packet with options (for IPv4), Hop−by−Hop Extension header (for IPv6). For example, Router Alert RFC−2113, Strict Source Route, and so on. Extension headers are not examined or processed by any node along a packet's delivery path, until the packet reaches the node (or each of the set of nodes in the case ofmulticast) identified in the Destination Address field of theIPv6 header. The only exception is the Hop−by−Hop Options header, which carries information that must be

		examined and processed by every node along a packet's delivery path, which includes the source and destination nodes. Hardware processing on option fields is not supported, that is software processing/switching is needed. Reference: RFC−791 / RFC−2460
Reverse Path Forwarding (RPF) failure (Unicast)	class−copp−ucast−rpf−fail	The packet failing RPF check is filtered. However, due to limited resources in the hardware, the RPF check cannot be done in hardware in certain cases (that is, more than 16 RPF interfaces linked to one IP). When that happens, the packet is sent to software for a complete RPF check. The first RPF failed data packet (addressed to a multicast group) is sent to software in order for the Protocol Independent Multicast (PIM)−assert process to start. Once the process is done, a designated router/forwarder is elected. If the next packet (same flow) does not come from the designated router, it triggers an RPF failure, and the hardware can drop it right away (in order to prevent a Denial of Service (DoS) attack).
RPF Failure (Multicast)	class−copp−mcast−rpf−fail	The first RPF failed data packet (addressed to a multicast group) is sent to the software in order for the PIM−assert process to start. Once the process is done, a designated router/forwarder is elected. If the next packet (same flow) does not come from the designated router, it triggers an RPF failure, and the hardware can drop it right away (in order to prevent a DoS attack). However, if the routing table is updated, a new designated router might need to be chosen (via PIM−assert), which means the RPF failed packet needs to reach the software (for PIM−assert to start again). In order to do that, a periodic leak to the software mechanism (per flow) for RPF−failed packet is available in the hardware. Note though, if there is a huge amount of flows then a periodic leak can be too much for the software to handle. The hardware CoPP is still required for multicast RPF failed packet. Reference: RFC−3704, RFC−2362
Hardware packet rewrite not supported	class−copp−unsupp−rewrite	While hardware can rewrite packets in various cases, some cases just cannot be done in the current hardware design. And for those, the

		hardware sends the packet to software.
ICMP no−route ICMP acl−drop ICMP redirect	class−copp−icmp−redirect−unreachable	Packets sent to software for the generation of ICMP messages. Such as ICMP redirect, ICMP destination unreachable (for example,. host unreachable or administratively prohibited). Reference: RFC−792 / RFC−2463
Cisco Express Forwarding (CEF) receive (destination IP is router's IP)	class−copp−receive	If the packet's destination IP is one of the router's IP addresses (will hit CEF receive adjacency), then the software is supposed to process the content.
CEF glean (destination IP belongs to one of router's network)	class−copp−glean	If the packet's destination IP belongs to one of the router's network, but it is not resolved (that is, no hit in the Forwarding Information Base (FIB) table), it will hit CEF glean adjacency, being sent to software where the resolution procedure will get started. For IPv4, the same flow continues to hit CEF glean until the address is resolved. For IPv6, a temporary FIB entry that matches the destination IP (and points to drop adjacency instead) gets installed during resolution. If it cannot be resolved in the specified duration, the FIB entry is removed (that is, the same flow starts to hit CEF glean again).
Packet destined to multicast IP 224.0.0.0/4	class−copp−mcast−ip−control	The control packet needs to be processed by the software.
Packet destined to multicast IP FF::/8	class−copp−mcast−ipv6−control	The control packet needs to be processed by the software.
Multicast packet which needs to be copied to software	class−copp−mcast−copy	In some cases, the multicast packet needs to be copied to software for a state update (the packet is still hardware bridged on the same VLAN). For instance, (*,G/m) hit for dense mode entry, dual−rpf SPT switchover.
Multicast packet getting a miss in FIB table	class−copp−mcast−punt	The destination IP (multicast IP) is a miss in the FIB table. The packet is punted to the software.
Directly connected source (IPv4)	class−copp−ip−connected	Multicast traffic from directly connected sources are sent to the software where a multicast state can be created (and installed in the hardware).
Directly connected source (IPv6)	class−copp−ipv6−connected	Multicast traffic from directly connected sources are sent to the software where a multicast state can be created (and installed in the hardware).
Broadcast packet	class−copp−broadcast	Broadcast packets (for example, IP/Non−IP with broadcast DMAC and IP unicast with Multicast DMAC) are leaked to the software.

Protocol unknown to (that is, unsupported by) in terms of hardware switching	class−copp−unknown−protocol	Non−IP protocol, such as Internetwork Packet Exchange (IPX) and so on, will not be hardware switched. They are sent to software and get forwarded there.
Multicast Data traffic coming in via routed port where PIM is disabled	class−copp−mcast−v4−data−on−routedPort	Multicast data traffic that comes in through a routed port (where PIM is disabled) is leaked to the software. However, it is not necessary to send them to software so they are dropped.
Multicast Data traffic coming in via routed port where PIM is disabled	class−copp−mcast−v6−data−on−routedPort	Multicast data traffic that comes in through a routed port (where PIM is disabled) is leaked to software. However, it is not necessary to send them to software so they are dropped.
Ingress ACL redirect to bridge the packet	class−copp−ucast−ingress−acl−bridged	The hardware has 8 ACL−related exceptions set by the software via an ACL redirect. This one relates to unicast packets bridged to the the CPU by the ACL for Ternary Content Addressable Memory (TCAM) related reasons.
Egress ACL redirect to bridge the packet	class−copp−ucast−egress−acl−bridged	The hardware has 8 ACL−related exceptions set by the software via an ACL redirect. This one relates to unicast packets bridged to the the CPU by the ACL for Ternary Content Addressable Memory (TCAM) related reasons.
Mcast ACL redirect to bridge packets to CPU	class−copp−mcast−acl−bridged	The hardware has 8 ACL−related exceptions set by the software via an ACL redirect. This one relates to multicast processing.
ACL bridge to CPU for Server Load Balancing processing	class−copp−slb	The hardware has 8 ACL−related exceptions set by the software via an ACL redirect. This one relates to a hardware redirect for a Server Load Balancing (SLB) decision.
ACL VACL log redirect	class−copp−vacl−log	The hardware has 8 ACL−related exceptions set by the software via an ACL redirect. This one relates to packet redirection by VLAN Access Control List (VACL) ACL to CPU for Cisco IOS® logging purposes.
DHCP snooping	class−copp−dhcp−snooping	DHCP snooped packets are redirected to the CPU for DHCP processing
MAC Policy Based Forwarding	class−copp−mac−pbf	Policy Based Forwarding is to be done in the CPU since the hardware is not capable to forward packets in this case.
IP−admission Network Admission Control	class−copp−ip−admission	In order to provide network access based on the host's antivirus credentials, there is posture validation via one of the these options: (1) The L2 interface will use LAN Port IP (LPIP), where Address Resolution Protocol (ARP) packets are redirected to the CPU, (2) The L3 interface uses Gateway IP (GWIP). After the

		validation, there is the authentication (*). For an L2 interface it is WebAuth, which performs HTTP packet interception and might also perform URL redirection (*). For the L3 interface, it is AuthProxy.
Dynamic ARP inspection	class−copp−arp−snooping	In order to prevent ARP poisoning (man−in−the−middle) attack, dynamic ARP inspection (also known as Dynamic ARP Inspection (DAI)) validates the ARP requests/responses by when it intercepts and then processes them in the CPU against one of the these: (1) user−configured ARP ACLs (for statically configured hosts), (2) MAC address to IP address bindings stored in trusted database (that is, DHCP bindings). Only valid ARP packets are used to update the local ARP cache or forwarded out. The validation process requires ARP packets CPU involvement, which means hardware CoPP is needed in order to prevent a DoS attack.
ACL redirect to CPU for WCCP	class−copp−wccp	Used in case the packet/flow needs to be redirected to the CPU for the Web Cache Communication Protocol (WCCP) forwarding decision.
ACL redirect to CPU for Service Insertion Architecture (SIA)	class−copp−service−insertion	Used in case the packet/flow needs to be redirected to the CPU for SIA decision.
IPv6 Network discovery	class−copp−nd	In order to redirect the IPv6 Network Discovery packet to the CPU to process further. Reference: RFC4861

Verify

Use this section in order to confirm that your configuration works properly.

In order to check if there was traffic observed in any of the configured CoPP class−maps, enter the ***show policy−map control−plane*** command.

Troubleshoot

There is currently no specific troubleshooting information available for this configuration.

Related Information

- ***Protecting Cisco Catalyst 6500 Series Switches Using Control Plane Policing, Hardware Rate Limiting, and Access−Control Lists***

- ***Catalyst 6500 Release 15.0SY Software Configuration Guide − Control Plane Policing (CoPP)***
- ***Technical Support & Documentation − Cisco Systems***

Updated: Mar 04, 2015 Document ID: 118806