



CSC OPEN CLASS

Cat6500 CPU troubleshooting & Protection Mechanism

Weidong Huang, Customer Support Engineer

weidhuan@cisco.com

December, 2014

Agenda

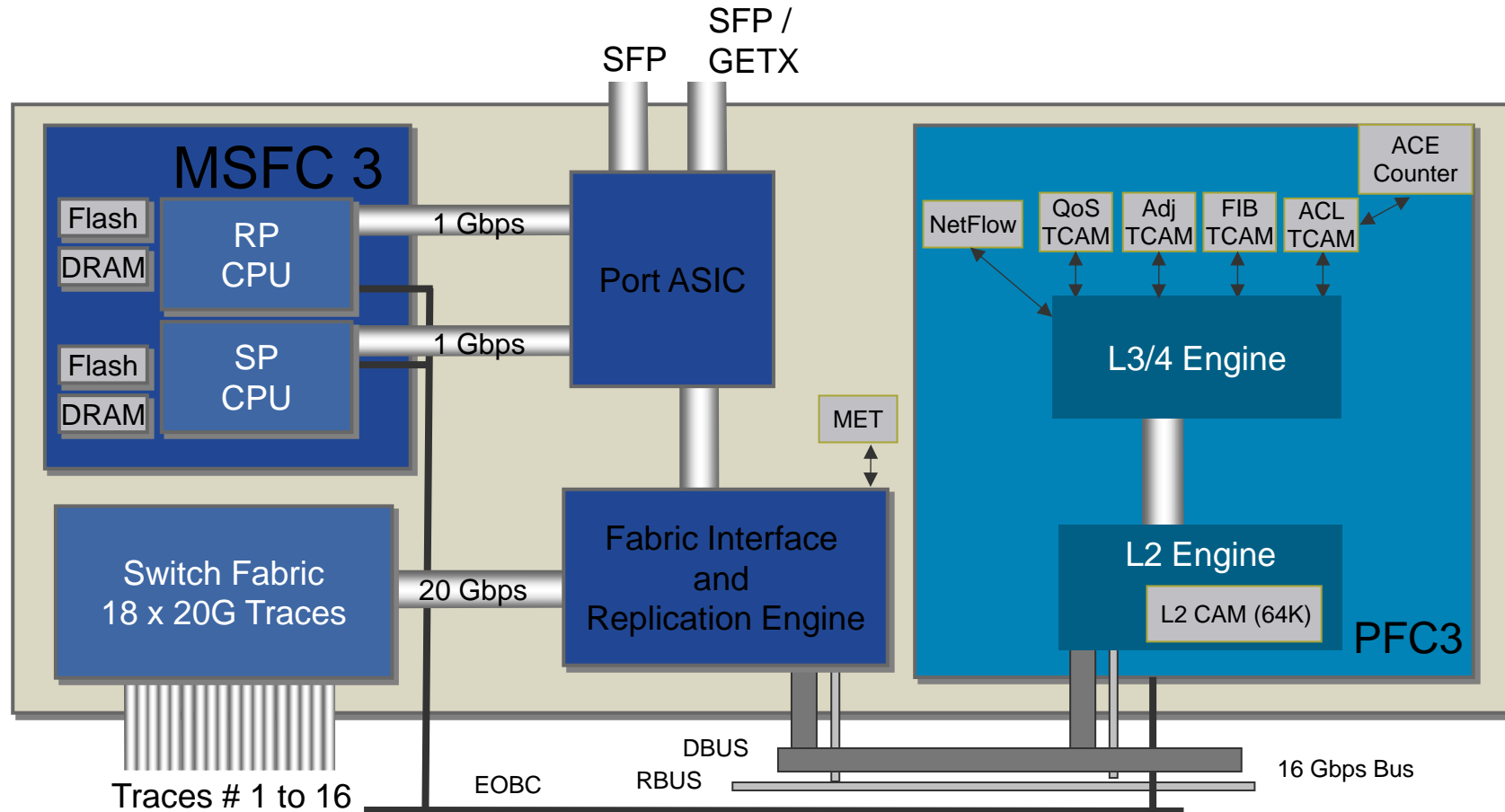
- Architecture
- Common Causes
- Commands and Tools
- Control-Plane Protection

Commonly Asked Questions

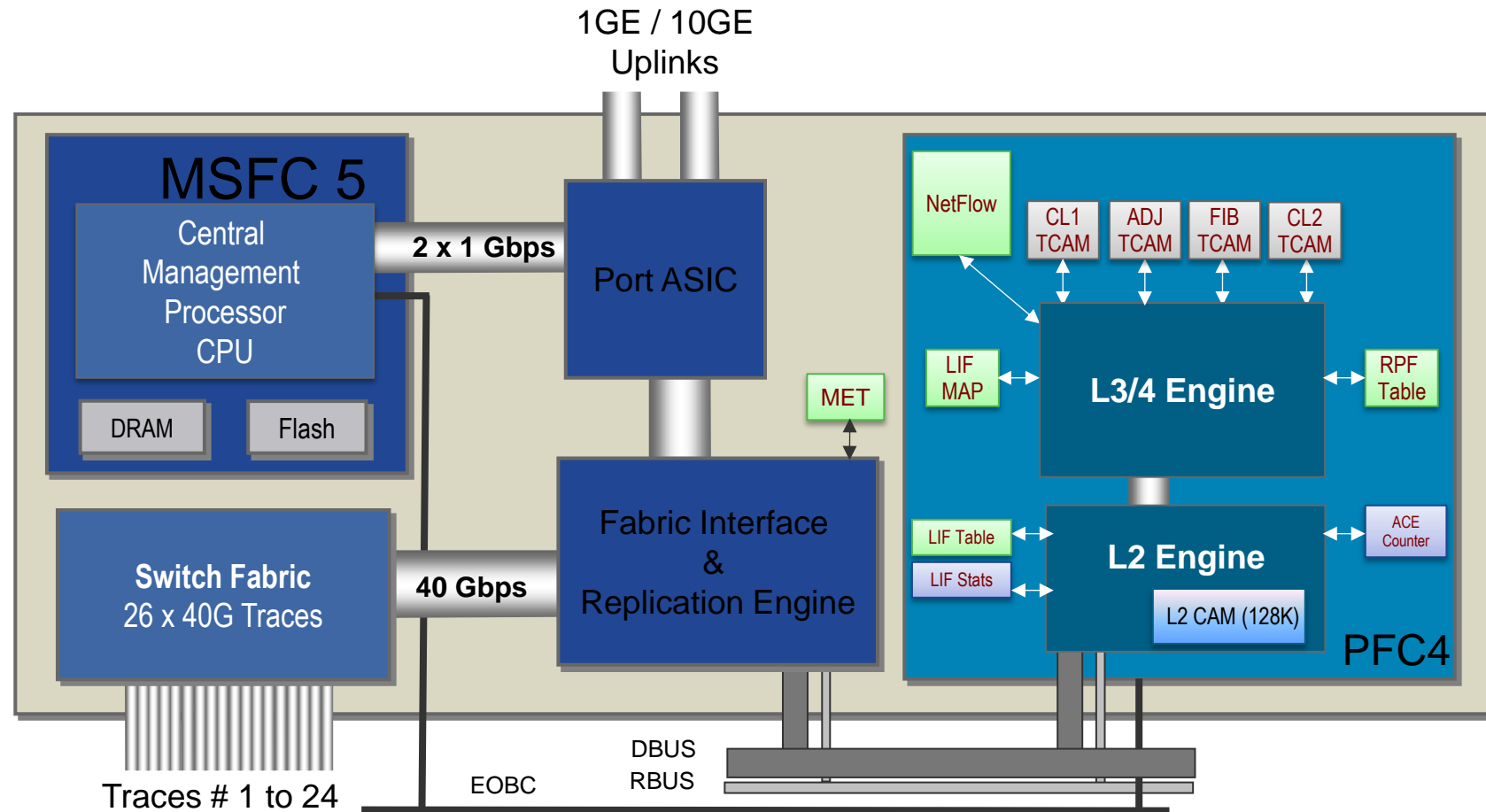
- Why should I be concerned about high CPU usage ?
- What are the usual symptoms of high CPU usage ?
- At what percentage level should I start troubleshooting ?

Architecture

Supervisor 720 Architecture



Supervisor 2T Architecture



Supervisor 720 MSFC3



The image shows a Cisco Supervisor 720 MSFC3 hardware unit with its cover removed, revealing the internal circuit board. The board is populated with several large integrated circuits, each with a black heat sink. A large black heat sink is positioned at the top left. The board is green and populated with various components, including capacitors and smaller chips. Two disk drives are visible at the bottom, labeled 'DISK 0' and 'DISK 1', each with an 'EJECT' button. The front panel is silver and black. The text labels are overlaid on the image, with dashed lines indicating the location of the components. The labels are: 'Switch Processor Bootflash 64MB' (top left), 'Route Processor Bootflash 64MB' (top right), 'Switch Processor' (middle left), 'Route Processor' (middle right), 'Switch Processor DRAM 512MB' (bottom left), and 'Route Processor DRAM 512MB' (bottom right). The board is populated with several large integrated circuits, each with a black heat sink. A large black heat sink is positioned at the top left. The board is green and populated with various components, including capacitors and smaller chips. Two disk drives are visible at the bottom, labeled 'DISK 0' and 'DISK 1', each with an 'EJECT' button. The front panel is silver and black. The text labels are overlaid on the image, with dashed lines indicating the location of the components. The labels are: 'Switch Processor Bootflash 64MB' (top left), 'Route Processor Bootflash 64MB' (top right), 'Switch Processor' (middle left), 'Route Processor' (middle right), 'Switch Processor DRAM 512MB' (bottom left), and 'Route Processor DRAM 512MB' (bottom right).

Switch Processor
Bootflash 64MB

Route Processor
Bootflash 64MB

Switch Processor

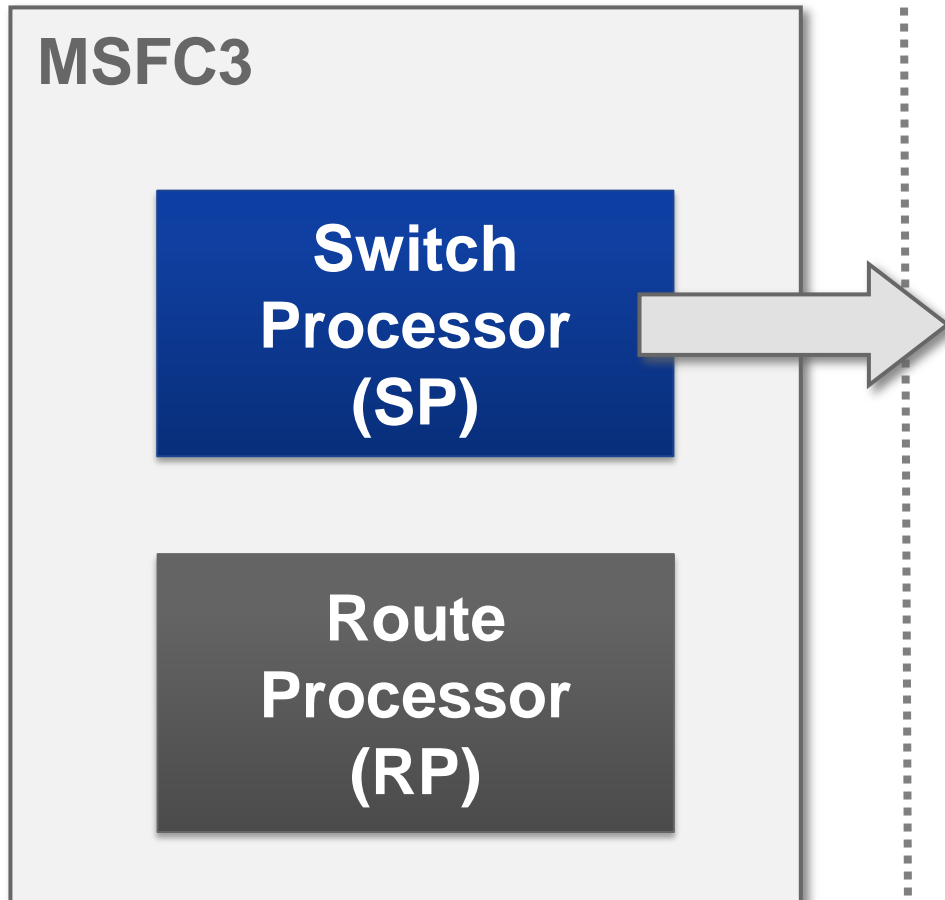
Route Processor

Switch Processor
DRAM 512MB

Route Processor
DRAM 512MB

MSFC3 - Switch Processor (SP)

Both the RP & SP CPU perform distinct functions...



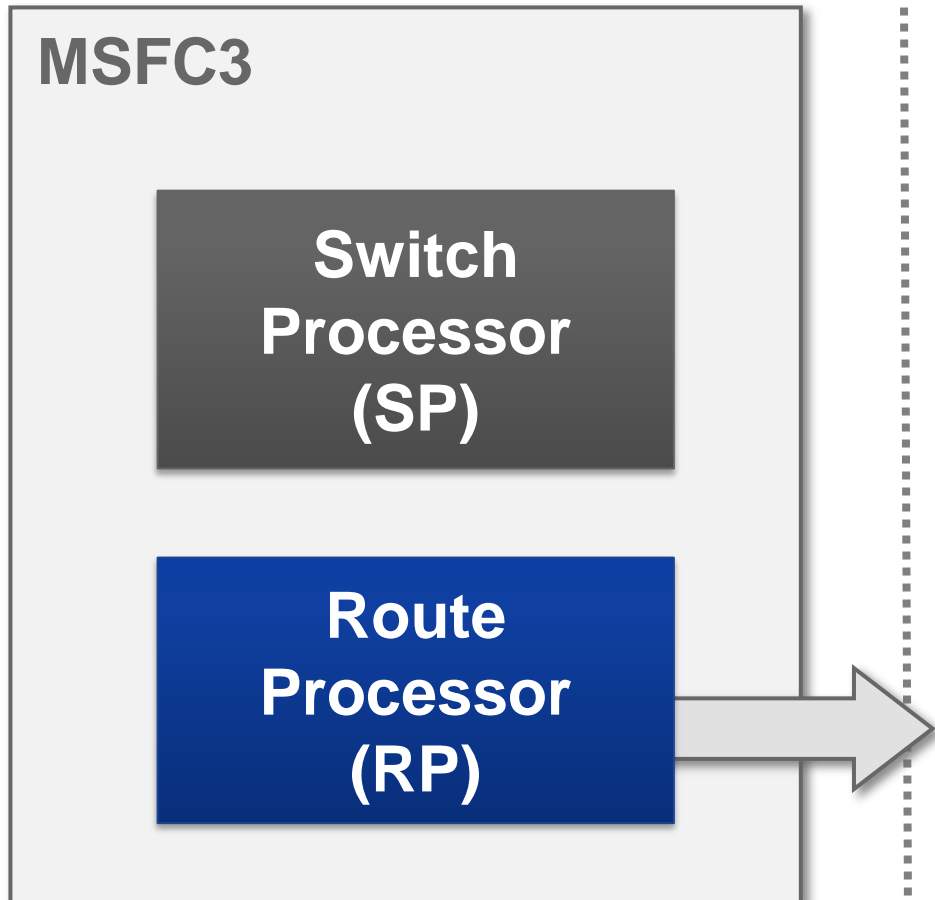
The “**Switch Processor**” CPU is physically located on the MSFC3

Has a dedicated CPU, DRAM and Flash

- The SP runs all Layer 2 functions & protocols such as VTP, Trunking, etc.
- Supports other Layer 2 features like STP, CDP, IGMP, SPAN, EtherChannel, etc.
- The SP owns the system at initial boot, before handing control over to the RP.
- Chassis & Power Management

MSFC3 - Route Processor (RP)

Both the RP & SP CPU perform distinct functions...



The “**Route Processor**” CPU is physically located on the MSFC3

Logically considered the “Router” component, connected via the Switch component

Has a dedicated CPU, DRAM and Flash

- The RP runs the Layer 3 functions & protocols such as OSPF, BGP, MPLS, PIM, etc.
- Supports other Layer 3 features like GRE, NAT, NDE, SNMP, SSH, etc.
- Manages the user interface (CLI) - Any Layer 2 commands are processed on the RP, then sent to the SP for execution

Supervisor 720: Flash or Disk



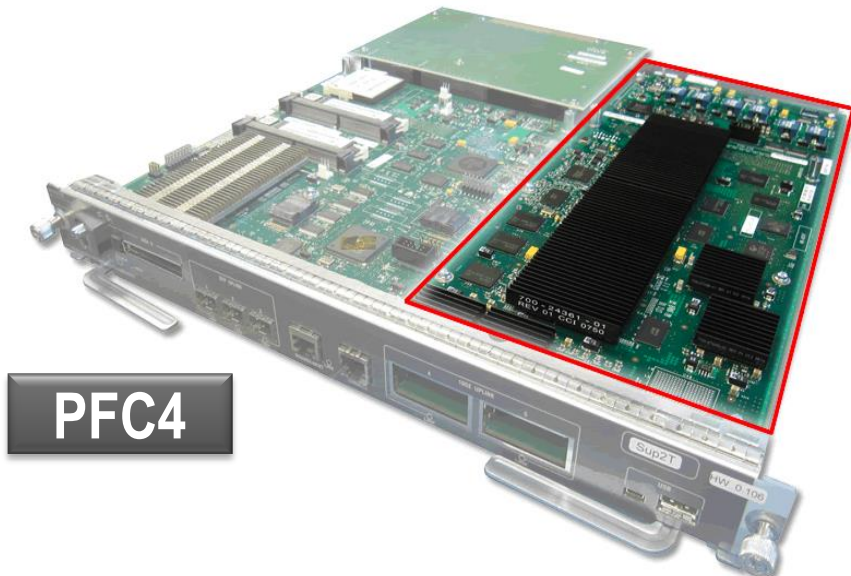
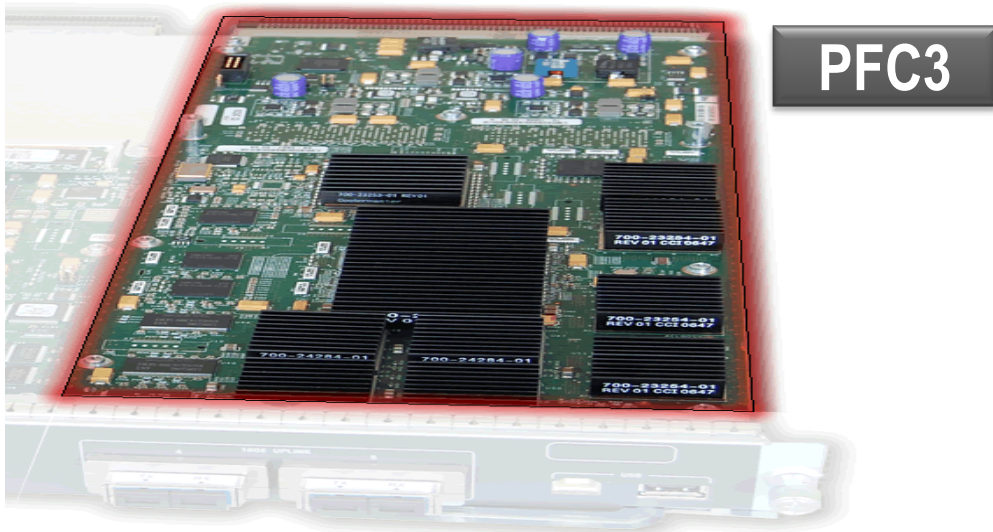
**MSFC3 SP and
RP Bootflash
Slots with
standard
Bootflash
installed...**



**MSFC3 SP
Bootflash Slot
with CF Bootflash
Adapter
installed...**

Catalyst 6500 Policy Feature Card

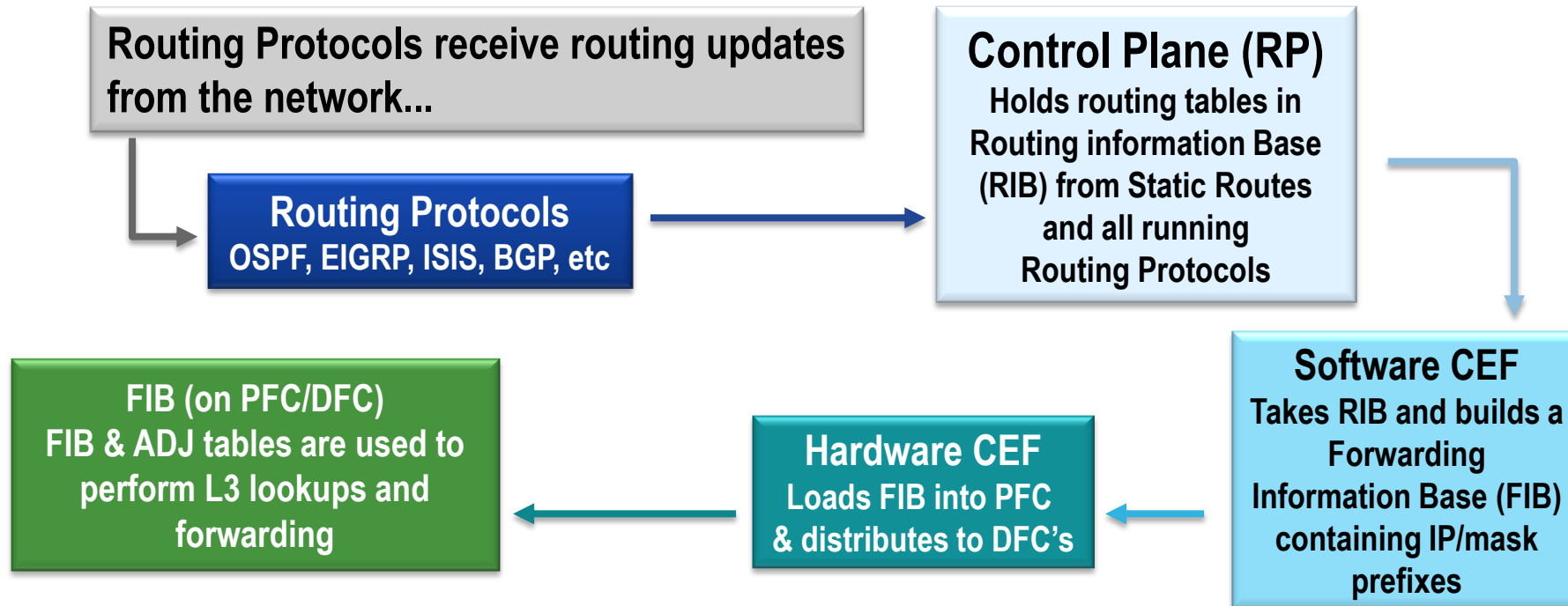
Overview of PFC3 and PFC4



- PFC Serves as Data Plane for 6500
- TCAM's used for high speed lookup into Forwarding (FIB), ACL (Security and QoS) and Netflow Tables
- PFC3 – 48Mpps Maximum Forwarding
- PFC4 – 60Mpps Maximum Forwarding
- Common features supported in hardware by PFC3 and PFC4 include: IPv4 - IPv6 - MPLS - Multicast - Policing - Classification - RACL - VACL - PACL - GRE - Tunneling - URPF - Control Plane Policing - and more
- Features introduced by the PFC4 include: Flexible NetFlow - ACL Dry Run - ACL Hitless Commit - Cisco TrustSec – VPLS - Egress NetFlow - IPv6 uRPF - Roles Based Access Control – 512K Multicast Routes – Improved EtherChannel Hash – and more

Catalyst 6500 IP Unicast Forwarding

PFC/DFC FIB Programming

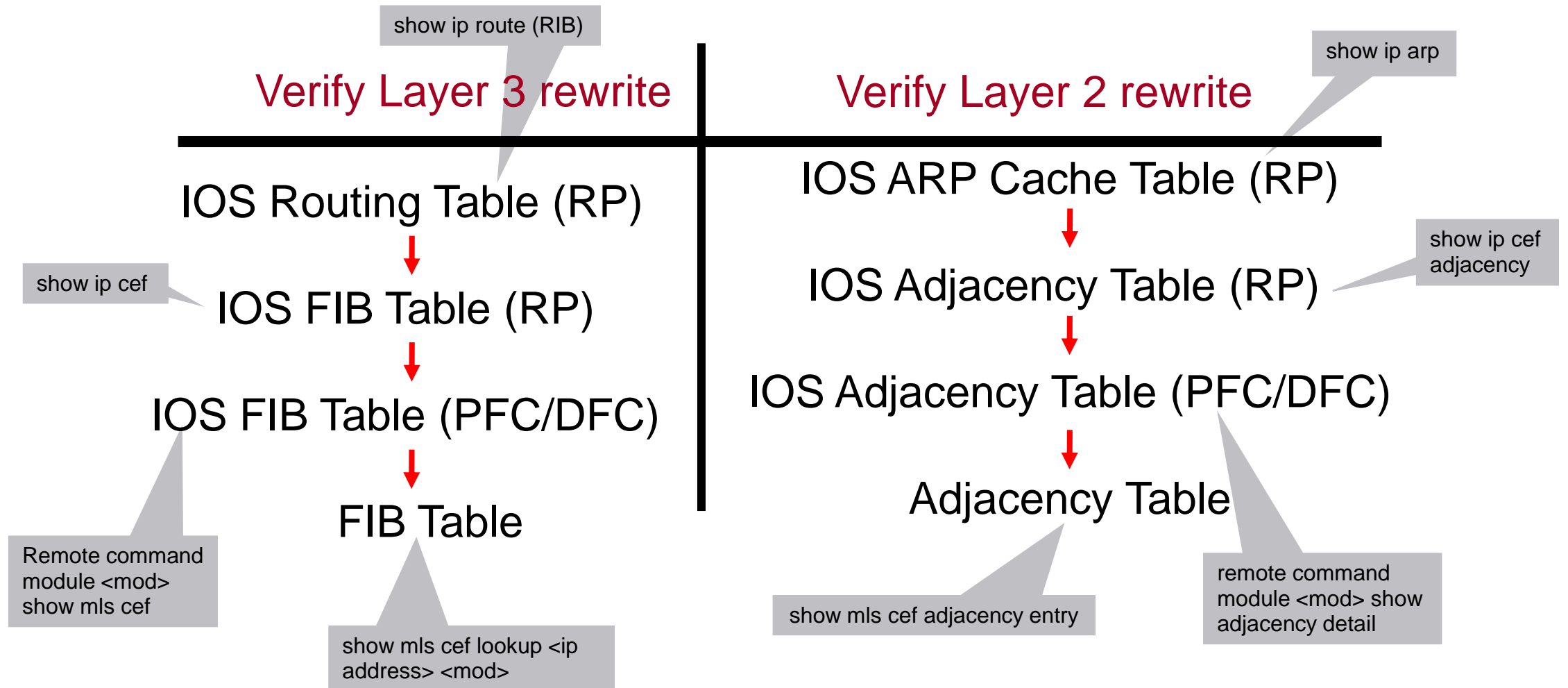


Hardware Based CEF Process

1. FIB lookup based on destination prefix (longest-match)
2. FIB "Hit" returns Adjacency pointer
3. Adjacency contains Rewrite (next-hop) information
4. ACL, QoS & NetFlow lookups occur in parallel, and effect final result

FIB / Adjacency Tables

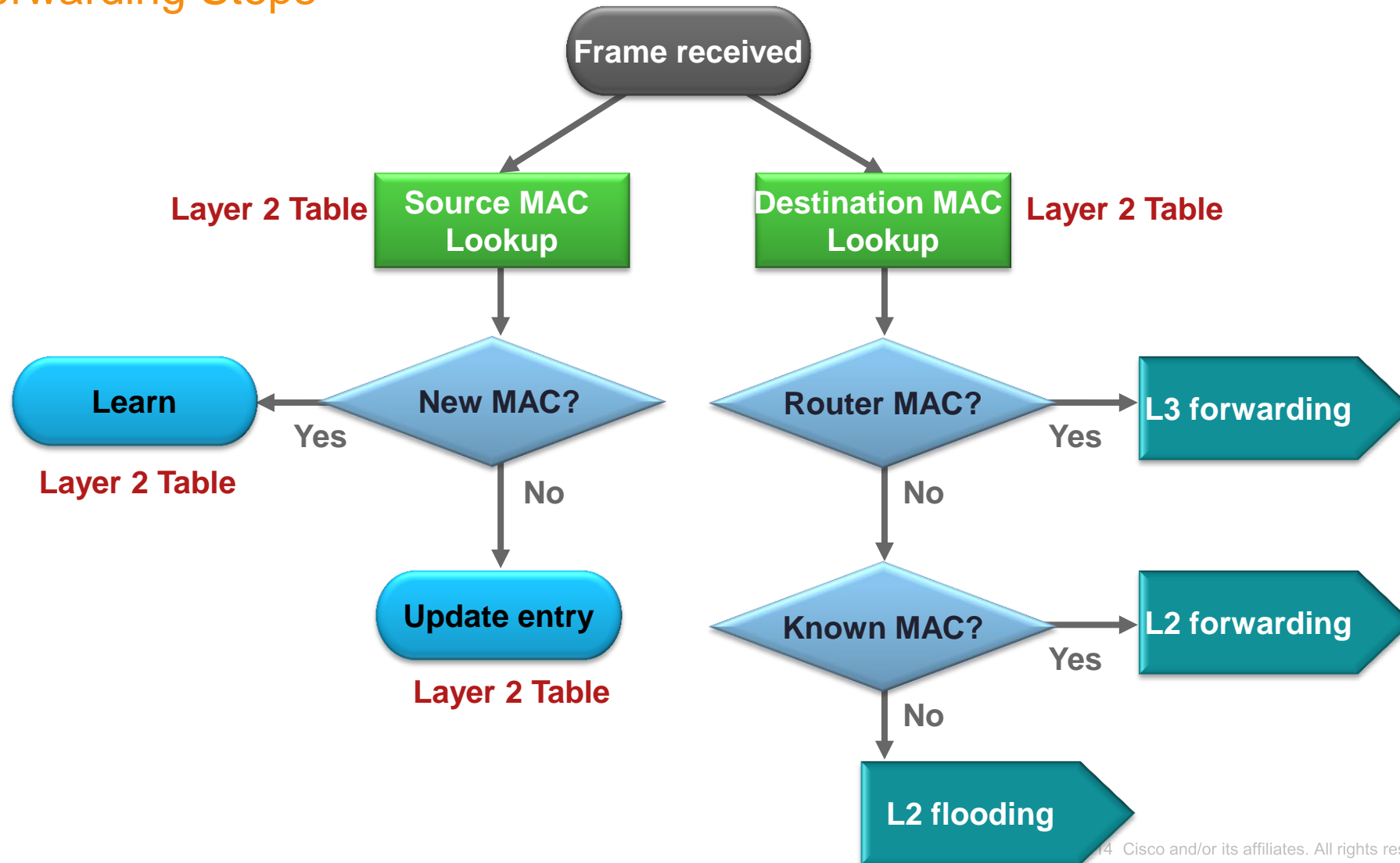
L3 FIB Table Programming Flow



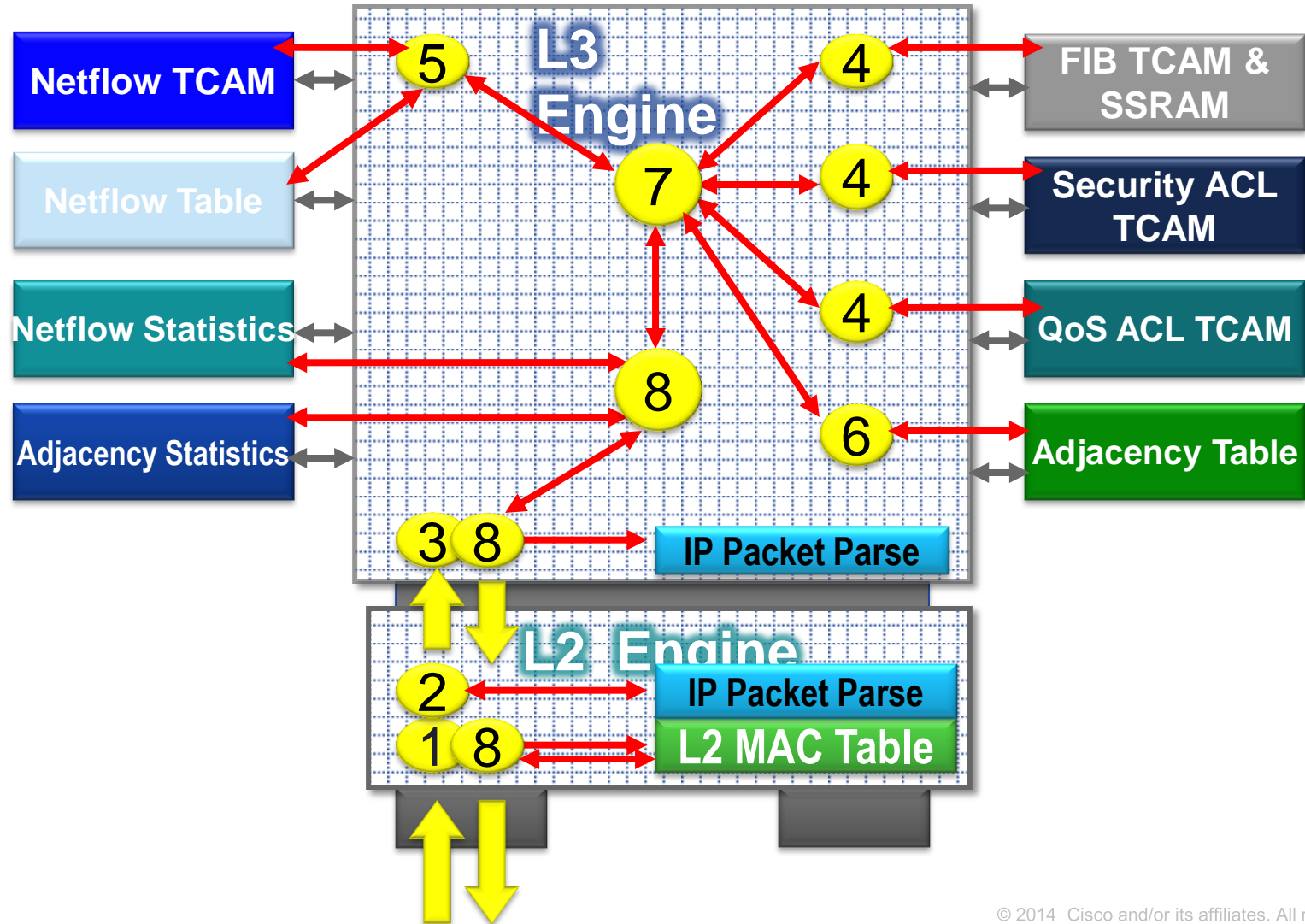
Replace "show mls" with "show platform hardware" for Sup2T engines.

Catalyst 6500 Internals

L2 Forwarding Steps




Catalyst 6500 PFC3/DFC3 Lookup Process



Common Causes

High CPU Utilization

Investigate CPU utilization via “show proc cpu” and find if the usage is due to process and/or interrupt



```
c6500# show process cpu
CPU utilization for five seconds: 99%/90%; one minute: 9%; five minutes: 8%
PID Runtime(ms) Invoked uSecs 5Sec 1Min 5Min TTY Process
<snip>
  2      720      88  8181  9.12%  1.11%  0.23%  18 Virtual Exec
<snip>
```

If CPU utilization is due to:

- Process → Causes: recurring events, control-plane process etc.
- Interrupts → Causes: inappropriate switching path, system running out of hardware resources etc.

High CPU Utilization

Due to processes ...

High CPU due to **ARP Input** Process:

- Caused by ARP flooding
- Static route configured with interface instead of next-hop IP address. This will generate ARP request for every packet that is not reachable via more specific routes.
`ip route 0.0.0.0 0.0.0.0 GigabitEthernet 2/5`

High CPU due to **BGP Scanner** Process:

- Walks the BGP table and confirms reachability of the next hops. It also checks conditional-advertisement to determine whether or not BGP should advertise condition prefixes, performs route dampening. It is normal to see this process spiking up for short duration, when the device carries huge internet routing table.
- Excessive BGP Control traffic received by CPU.

High CPU due to **SNMP Engine** Process:

- Due to aggressive polling of MIBs. “show snmp” provides SNMP input and output stats.

High CPU due to **IP Input** Process:

- Caused by traffic that needs to be process-switched or destined to the CPU.
- Most Common Reasons:
 - ❖ Broadcast storm
 - ❖ Traffic with IP-Options enabled
 - ❖ Traffic to which ICMP Redirect or Unreachable required e.g., TTL=1, ACL Deny etc.
 - ❖ Traffic that needs further CPU processing e.g., ACL Logging

High CPU due to **Exec/Virtual Exec** Process:

- Caused by sending when too many messages to console / VTY session(s)
- Usually caused due to packet debugging and sending logs to console / VTY session(s). Check “show debug” results and do “undebg all” if necessary.
- Are you running a “show tech” command and sending results to a console / VTY session ?

Interrupt Driven CPU Utilization

Common Causes:

1. TTL Expiration
2. ICMP Message Required
3. CEF/TCAM Errors
4. Netflow Features
5. ACL Logging
6. Unsupported PBR Configuration

Packets need to be special processed.

Lack of hardware resources.

Configuration issue.

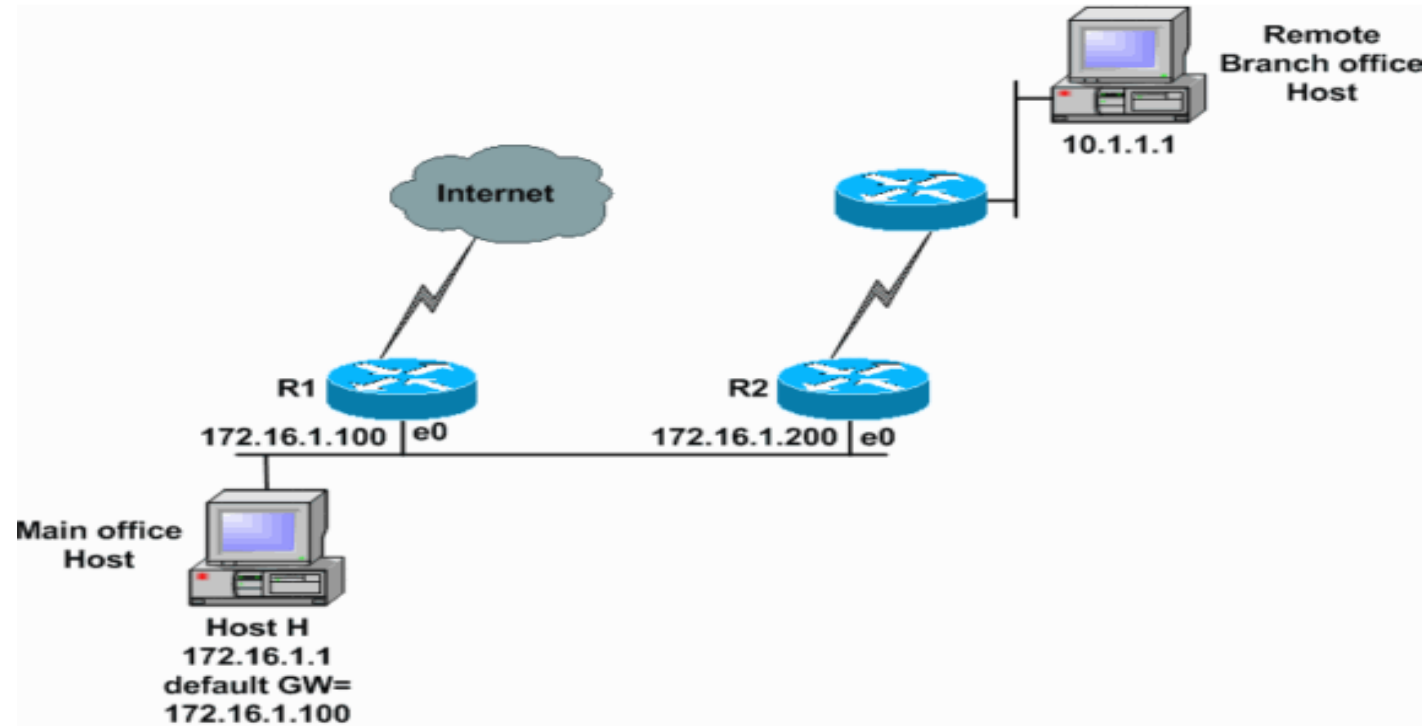
TTL Expiration

- Routed Packets with a TTL of 1 are punted to the RP to create an ICMP TTL exceeded message. Common reasons we may see TTL of 1:
 1. Routing Loop (easy to verify via traceroute)
 2. Bad host

```
c6500# show netdr captured-packets | i ttl
df 0, mf 0, fo 0, ttl 1, src 10.0.0.12, dst 14.1.104.1
df 0, mf 0, fo 0, ttl 1, src 10.0.0.12, dst 14.1.104.1
df 0, mf 0, fo 0, ttl 1, src 10.0.0.12, dst 14.1.104.1
df 0, mf 0, fo 0, ttl 1, src 10.0.0.12, dst 14.1.104.1
df 0, mf 0, fo 0, ttl 1, src 10.0.0.12, dst 14.1.104.1
df 0, mf 0, fo 0, ttl 1, src 10.0.0.12, dst 14.1.104.1
df 0, mf 0, fo 0, ttl 1, src 10.0.0.12, dst 14.1.104.1
df 0, mf 0, fo 0, ttl 1, src 10.0.0.12, dst 14.1.104.1
```

ICMP

1. ICMP Unreachable (acl-drop or no-route)
2. ICMP Redirects



CEF/TCAM errors

- Ensure that CEF is enabled on the ingress and egress interface

```
c6500#show ip int vlan 10
Vlan10 is up, line protocol is up
Internet address is 10.0.0.5/24
IP fast switching is enabled
IP fast switching on the same interface is disabled
IP Flow switching is disabled
IP CEF switching is enabled
IP Feature Fast switching turbo vector
```

- Ensure that the FIB TCAM has not hit an exception state. If it has, the customer should have received an error message in the log

```
%MLSCEF-SP-7-FIB_EXCEPTION: FIB TCAM exception, Some entries will be software switched
c6500#show mls cef exception status
Current IPv4 FIB exception state = FALSE
Current IPv6 FIB exception state = FALSE
Current MPLS FIB exception state = FALSE
```

- A FIB TCAM exception may occur if the number of routes exceeds the maximum allowed routes. Reducing the number of routes and cef entries can pull the FIB out of the exception status. However, new routes cannot be added once a FIB exception has been encountered. A reload of the router is required to fully clear the exception.

Common Causes – CEF/TCAM errors

- Large ACLs may not fit into TCAM. In this case, a punt is programmed in hardware such that the traffic can be checked in software. If the ACLs were unable to be merged, you should see the following messages in the log:

%FM-4-TCAM_ENTRY: Hardware TCAM entry capacity exceeded

%FMCORE-4-RACL_REDUCED: Interface <x> routed traffic will be software switched in <y> direction

- This may be due to “ACL Explosion” in which an ACL is expanded because the system is out of labels, masks, LOUs, etc... Start troubleshooting with “**show tcam counts**”. More information regarding ACL merging:
- http://www.cisco.com/en/US/products/hw/switches/ps708/products_white_paper09186a00800c9470.shtml

Netflow features

- Features that require netflow (NDE, WCCP, NAT, UBRL, SLB, Reflexive ACLs, etc...) may cause high CPU if the netflow table is exhausted or if there is a flow mask conflict.
- The following messages may be seen

```
%EARL_NETFLOW-SP-4-TCAM_THRLD:  
%FM-2-FLOWMASK_CONFLICT:  
%FM_EARL7-4-FEAT_FLOWMASK_REQ_CONFLICT:
```

- NetFlow-based features always need to see the first packet of a flow in software. Once the first packet of the flow reaches software, subsequent packets for the same flow are hardware-switched.
- If we can prove that each packet hitting the CPU is the first packet of a new flow, then the high CPU is expected. Note that this is not usually the case.
- <http://www.cisco.com/c/en/us/support/docs/switches/catalyst-6500-series-switches/70974-netflow-catalyst6500.html>

ACL Logging

- ACLs that contain ACEs with the log keyword will punt matching traffic to create syslogs. The log messages are not displayed for every packet but each matching packet will be software switched.

```
3d02h: %SEC-6-IPACCESSLOGP: list a1 denied udp 10.0.0.12(0) -> 172.18.108.26(0), 161513 packets
```

```
c6500# show ip access-list a1
```

```
Extended IP access list a1
```

```
10 deny ip host 10.0.0.12 host 172.18.108.26 log (288663 matches)
```

```
20 permit ip any any (103063 matches)
```

Unsupported PBR Config

- Policy Based Routing (PBR) on Sup720 requires each sequence number to have a **match** and a **set** statement. The Sup720 supports the “**match ip address**” and the following set statements:
 - set ip next-hop
 - set ip default next-hop
 - set interface null 0
- If either **match** or **set** statement is missing or an unsupported **set** or **match** statement is configured, then packets hitting the PBR sequence number will be software switched.
- Deny ACEs in PBR ACLs may be expanded in TCAM which can cause TCAM explosion. If the ACLs cannot be merged then a ‘punt’ will be programmed. We recommend trying to configure all PBR ACLs with only permit statements.

High CPU Utilization

Protocols and Services Processed in Software

Control Plane (L2/L3) Protocols	Control Plane Packet Forwarding
UDLD Protocol	IP Options
PAgP Protocol	Fragmentation
LACP Protocol	Select Tunnel Options
SNMP Protocol	ICMP Packets
Syslog Export	MTU failure
Netflow & Netflow Data Export	TTL=1 or TTL=0
Address Resolution Protocol (ARP)	Packets with Checksum error or error length
HSRP, VRRP, GLBP	RPF Check
Cisco Discovery Protocol (CDP)	Packets that require ARP resolution
VLAN Trunking Protocol	Non-IP (IPX, Appletalk)
Dynamic Trunking Protocol	ACL logging
Telnet, IP Sec, SSH	Broadcast traffic denied in RACL
BGP, OSPF, EIGRP, RIP, ISIS	Authentication Proxy
Web Cache Control Protocol	PBR traffic for certain "match" or "set" arguments

Commonly Asked Questions

- Why should I be concerned about high CPU usage ?

It is very important to protect the control-plane for network stability, as resources (CPU, Memory and buffer) are shared by control-plane and data-plane traffic (sent to CPU for further processing)

- What are the usual symptoms of high CPU usage ?
 - PING Lost
 - Slow response to Telnet / SSH
 - SNMP poll miss
 - Control-plane instability e.g., OSPF flap, EIGRP flap
 - Reduced switching / forwarding performance
- At what percentage level should I start troubleshooting ?

It depends on the nature and level of the traffic. It is very essential to find a baseline CPU usage during normal working conditions, and start troubleshooting when it goes above specific threshold.

Commands and Tools

Commands and tools

- *Monitor the CPU Utilization*
- What should be checked first?
- What traffic are punted?
- Take Action !

Monitoring the CPU

CPU Threshold Notification

- 6500(config)#process cpu threshold type total rising 90 interval 5

System will report logs once the configured threshold has been exceeded.

- Dec 16 14:41:16.926: %SYS-1-CPURISINGTHRESHOLD: Threshold: Total CPU Utilization(Total/Intr): 90%/0%, Top 3 processes(Pid/Util): 483/90%, 2/0%, 3/0%

- 6500# show proce cpu | inc 483

PID	Runtime(ms)	Invoked	uSecs	5Sec	1Min	5Min	TTY	Process
483	4652	34800	133	0.07%	0.00%	0.00%	1	Virtual Exec

Search the PID in “show process cpu” for the exact CPU process name.

Monitoring the CPU

SNMP Notification

- 6500(config)#process cpu threshold type total rising 90 interval 5

```
6500(config)# snmp-server enable traps cpu threshold
```

- [cpmCPUTotal5secRev](#) (.1.3.6.1.4.1.9.9.109.1.1.1.1.6): The overall CPU busy percentage in the last five-second period
- More information about using SNMP to collect CPU Utilization
<http://www.cisco.com/c/en/us/support/docs/ip/simple-network-management-protocol-snmp/15215-collect-cpu-util-snmp.html>

Monitoring the CPU

EEM Scripts to Monitor High CPU Utilization

event manager session cli username <Username>

give the username which has privilege to run the below commands

event manager applet High_CPU

event snmp oid **1.3.6.1.4.1.9.9.109.1.1.1.1.6.1** get-type exact entry-op ge entry-val 50 poll-interval 0.5

action 0.0 syslog msg "High CPU DETECTED. Please check nvram:high_cpu.txt"

action 0.1 cli command "enable"

action 0.2 cli command "show process cpu | append nvram:high_cpu.txt"

action 0.3 cli command xxx

action 0.4 cli command xxx

action 0.5 cli command xxx

.....

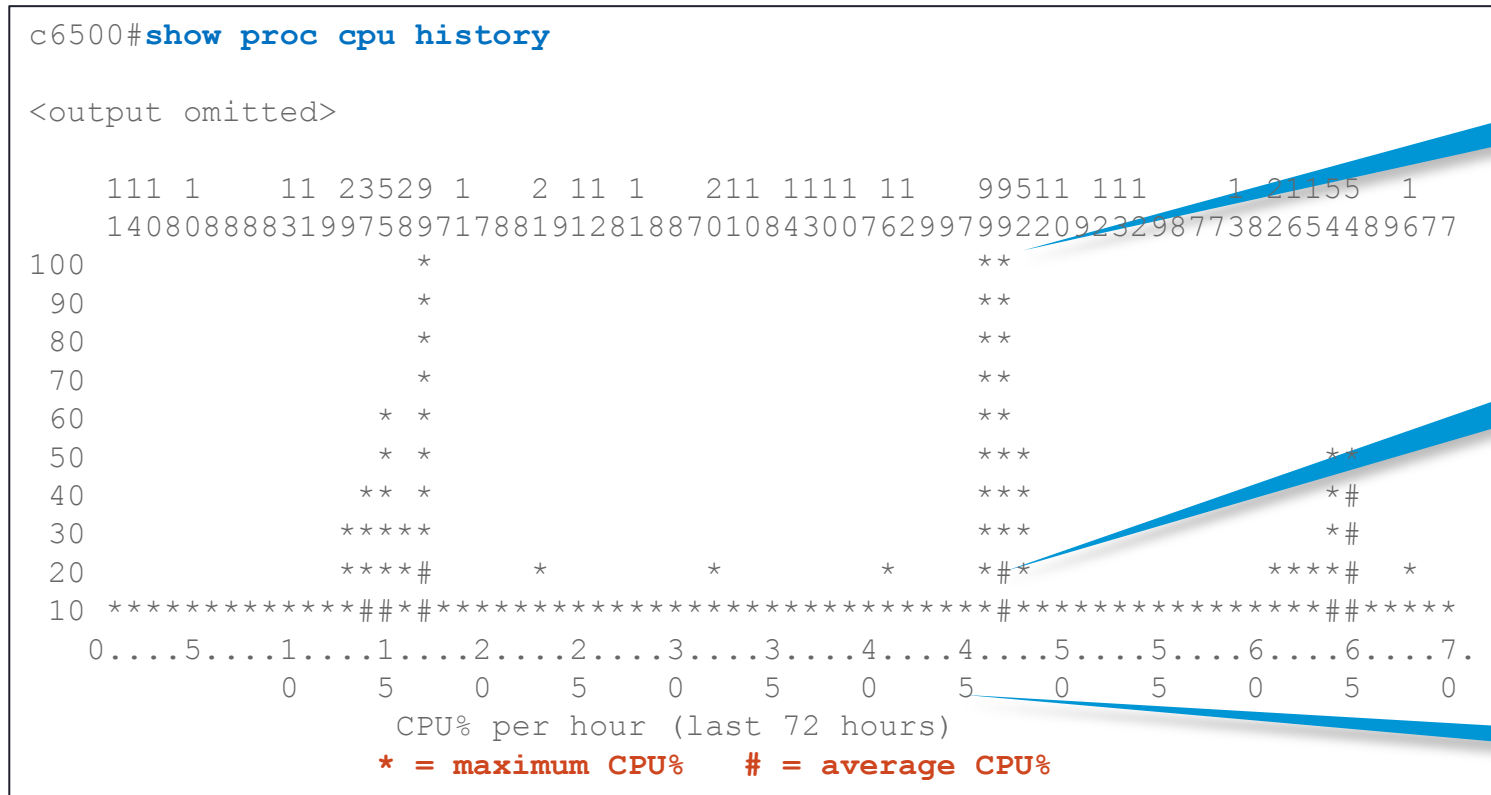
Commands and tools

- Monitor the CPU Utilization
- *What should be checked first?*
- What traffic are punted?
- Take Action !

When the Problem Happening

Show process CPU

- CPU History



* means peak utilization

means average utilization

Time past since this utilization

Usage of Hardware Capacity

```
Sup2T# show platform hardware capacity forwarding
```

L2 Forwarding Resources

MAC Table usage:	Module	Collisions	Total	Used	%Used
	1	0	131072	55	1%
	2	0	131072	55	1%
	5	0	131072	55	1%
	6	0	131072	53	1%

L3 Forwarding Resources

FIB TCAM usage:		Total	Used	%Used
72 bits (IPv4, MPLS, EoM)		1048576	68	1%
144 bits (IP mcast, IPv6)		524288	8	1%
288 bits (IPv6 mcast)		262144	1	1%
detail:	Protocol		Used	%Used
	IPv4		66	1%
	MPLS		1	1%
	EoM		1	1%
	IPv6		2	1%
	IPv4 mcast		6	1%
	IPv6 mcast		1	1%
Adjacency usage:		Total	Used	%Used
		1048576	32029	3%

Usage of Hardware Capacity

(Continued ...)

Forwarding engine load:

Module	pps	peak-pps	peak-time
1	2	1374	11:40:31 EST Fri Mar 23 2012
2	0	408	17:34:47 EST Thu Mar 22 2012
5	52	666	11:40:31 EST Fri Mar 23 2012
6	0	25	12:11:11 EST Fri Mar 23 2012

Sup2T# show platform hardware capacity ?

Note: Not all available options are shown here

```

acl          Show QoS/Security ACL capacity
cpu          Show CPU resources capacity
fabric       Show Fabric resources capacity
forwarding   Show forwarding engine capacity
monitor      Show SPAN resources capacity
multicast    Show L3 and LTL Multicast resources
netflow      Show Netflow capacity
pfc          Show PFC resources capacity
power        Show Power resources capacity
qos          Show QoS resources capacity
rate-limit   Show CPU Rate Limiters capacity
rewrite-engine Show rewrite-engine capacity
vlan         Show VLAN resources capacity
    
```

Usage of ACL TCAM

Usage of switching fabric

Usage of Netflow TCAM

Usage of QoS TCAM

Usage of hardware rate-limiters

FIB Status

%MLSCEF-SP-7-FIB_EXCEPTION: FIB TCAM exception, Some entries will be software switched

```
Sup720# show mls cef maximum-routes
```

```
FIB TCAM maximum routes :
=====
IPv4 + MPLS                - 192k (default)
IPv6 + IP Multicast        - 32k (default)
```

Do “**show mls cef exception status**” to see if the device had TCAM exception. If **TRUE**, **reload** is required to clear the exception state. In Sup2T, the command is “**show plat hardware cef exception status**”.

```
Sup720# show mls cef summary
```

```
Total routes:                194532
IPv4 unicast routes:          194527
IPv4 Multicast routes:        3
MPLS routes:                  0
  IPv6 unicast routes:        2
```

Total routes installed in the HW

Global PFC mode is the least-common-denominator of the PFC/DFC types present in the switch. For e.g., if the switch has Sup720/3BXL and a module with DFC3B, then global PFC mode will be PFC3B.

Max number of routes supported depends on the global PFC mode

```
Sup720# show platform hardware pfc mode
```

```
PFC operating mode : PFC3B
```

It requires reload to take effect.

Reconfigure maximum routes supported for specific protocol (IPv4, IPv6, MPLS etc.) as required.

```
Sup720(config)# mls cef maximum-routes ip 239
```

Commands and tools

- Monitor the CPU Utilization
- What should be checked first?
- *What traffic are punted?*
- Take Action !

What kind of Traffic Are Punted

Traffic to CPU statistics

```
Sup2T# show ip traffic
```

```
IP statistics:
```

```
Rcvd: 81676 total, 20945 local destination
```

```
0 format errors, 0 checksum errors, 41031 bad hop count
```

```
0 unknown protocol, 19609 not a gateway
```

```
0 security failures, 0 bad options, 120 with options
```

```
Frag: 0 reassembled, 0 timeouts, 0 couldn't reassemble
```

```
0 fragmented, 0 couldn't fragment
```

```
Bcast: 417 received, 0 sent
```

```
Mcast: 11423 received, 52655 sent
```

```
Sent: 61340 generated, 0 forwarded
```

```
Drop: 32 encapsulation failed, 0 unresolved, 0 no adjacency
```

```
ICMP statistics:
```

```
Rcvd: 0 format errors, 0 checksum errors, 17 redirects, 112 unreachable
```

```
812 echo, 812 echo reply, 0 mask requests, 0 mask replies, 0 quench
```

```
0 parameter, 0 timestamp, 0 info request, 0 other
```

```
ARP statistics:
```

```
Rcvd: 3518120 requests, 3636408 replies, 0 reverse, 0 other
```

```
<snip>
```

TTL < 2 traffic

Traffic with IP Options

Broadcast traffic

Unresolved ARP

ICMP sent / received for various reasons

ARP sent and received

Where the Traffic Come From

```
Sup720#show interface g 5/2 stats
```

```
GigabitEthernet5/2
```

Switching path	Pkts In	Chars In	Pkts Out	Chars Out
Processor	37	2638	11225	4284906
Route cache	178973881	10738432860	0	0
Distributed cache	132812893	8500025152	0	0
Total	311786811	19238460650	11225	42849062

Where the Traffic Come From

```
Sup720#show interface g 5/2 switching
```

```
GigabitEthernet5/2
```

```
    Throttle count          0
    Drops          RP          0          SP          0
    SPD Flushes      Fast      0          SSE          0
    SPD Aggress      Fast      0
    SPD Priority      Inputs    0          Drops          0
```

Protocol	Path	Pkts In	Chars In	Pkts Out	Chars Out
IP	Process	28733	11467979	8912	1209675
	Cache misses	0			
	Fast	0	0	2513	150780
	Auton/SSE	0	0	102493353	6559574592

```
<snip..>
```

Packet Capture Tools

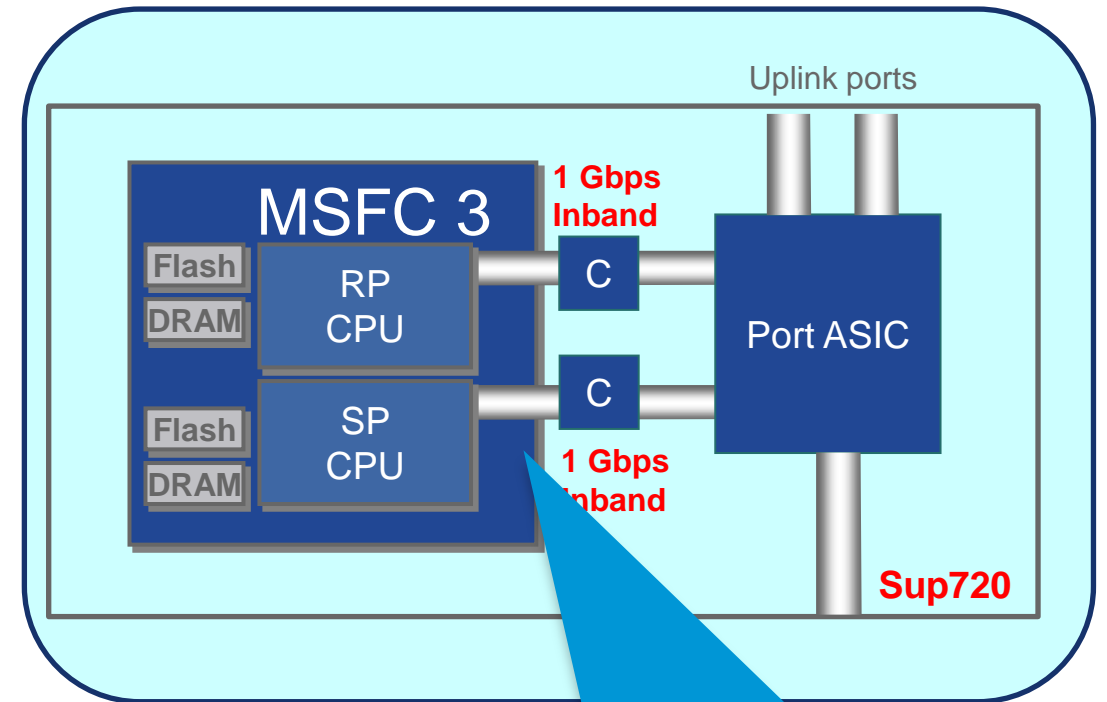
Inband SPAN

```
6500 (config)#monitor session 1 type local
6500 (config-mon-local)#source cpu ?
  rp  Route Processor (Active)
  sp  Switch Processor (Active)

6500 (config-mon-local)#source cpu rp ?
  both Monitor received and sent traffic from and
        to the cpu
  rx   Monitor traffic received from the cpu
  tx   Monitor traffic sent to the cpu
<cr>

6500 (config-mon-local)#destination interface Gig8/1
```

Inband SPAN is **NOT** supported in Sup2T engines.



Traffic sent / received on the inband channel(s) is replicated and sent to the SPAN destination port

Packet Capture Tools

Mini-Protocol Analyzer (MPA)

MPA will take one of the SPAN sessions available.

```
Sup2T(config)#monitor session 1 type capture
Sup2T(config-mon-capture)#source interface ten1/1 - 8 , ten2/1 - 8 both
```

Several **options** available to configure - Capture buffer-size, Capture filter, rate-limit packet capture etc.

```
Sup2T#sh monitor session capture
```

Session 1

```
-----
Type                : Capture Session
Source Ports        :
    Both             : Ten1/1-8, Ten2/1-8
Capture filters     : None
Egress SPAN Replication State:
Operational mode    : Centralized
Configured mode     : Centralized (default)
```

```
Sup2T# show monitor capture status
```

```
capture state : ON [running for 00:00:24.912]
capture mode  : Linear
Number of packets
    captured : 71
    dropped  : 0
    received : 71
```

```
Sup2T# monitor capture ?
```

```
buffer      Control Capture Buffers
circular    Capture buffer mode
clear       clear capture
dot1q       include dot1q info
export      Export to remote location
filter      software filter acl
length      Capture length
linear      Capture buffer mode
point       Control Capture Points
schedule    schedule capture at a specific time/date
start       start capture
stop        stop capture
```

Length of the packets captured. E.g., 68 Bytes

Start and Stop the capture, as needed

Packet Capture Tools

Mini-Protocol Analyzer

This command list all the captured data, with index

```
Sup2T#show monitor capture buffer
 1   len 60, 0180.c200.0000 001b.0da5.e286 0027 424203000002023C
 2   len 60, 001b.0da5.e286 001b.0da5.e286 9000 0000010000000000
 3   len 60, 0180.c200.0000 001b.0da5.e286 0027 424203000002023C
<snip>
```

```
Sup2T# show monitor capture buffer ?
<1-4294967295> start index
acl            filter output of captured Packets
brief         Brief output of captured Packets
detail        Detailed output of captured Packets
dump          Hex Dump of captured Packets
<snip>
```

Different options to display the data

Display specific packet using its index number

```
Sup2T# show monitor capture buffer 70 detail
70   Arrival time : 15:59:47.606 UTC Fri Mar 23 2012
     Packet Length : 92 , Capture Length : 68
     Ethernet II : 0100.5e00.0002 0000.0c07.acca 0800
     IP: s=192.168.10.2 , d=224.0.0.2, len 78
     UDP src=1985, dst=1985
```

Export the captured data to a **.cap file**
Sup2T# mon capture export buffer disk0:test.cap

NetDriver Debug

Does the CPU Inband Driver See the Packet?

```
6500#debug netdr capture ?
```

```
acl                (11) Capture packets matching an acl
and-filter         (3) Apply filters in an and function: all must match
continuous        (1) Capture packets continuously: cyclic overwrite
destination-ip-address (10) Capture all packets matching ip dst address
dstindex          (7) Capture all packets matching destination index
ethertype         (8) Capture all packets matching ethertype
interface         (4) Capture packets related to this interface
or-filter         (3) Apply filters in an or function: only one must match
rx               (2) Capture incoming packets only
source-ip-address (9) Capture all packets matching ip src address
srcindex         (6) Capture all packets matching source index
tx              (2) Capture outgoing packets only
vlan            (5) Capture packets matching this vlan number
<cr>
```

Debug Should **Not** Be Service-Impacting

NetDriver Debug

Does the CPU Inband Driver See the Packet?

```
6500# show netdr captured-packets
```

```
A total of 289 packets have been captured
```

```
The capture buffer wrapped 0 times
```

```
Total capture capacity: 4096 packets
```

```
----- dump of incoming inband packet -----
```

```
interface Vl1000, routine mistral_process_rx_packet_inlin
```

```
dbus info: src_vlan 0x3E8(1000), src_indx 0x45(69), len 0x40(64)
```

```
bpdu 0, index_dir 0, flood 1, dont_lrn 0, dest_indx 0x43E8(17384)
```

```
80000401 03E80400 00450000 40800000 E0000000 00000000 00000008 43E80000
```

```
mistral_hdr: req_token 0x0(0), src_index 0x45(69), rx_offset 0x76(118)
```

```
requeue 0, obl_pkt 0, vlan 0x3E8(1000)
```

```
destmac FF.FF.FF.FF.FF.FF, srcmac 00.A0.CC.21.94.C4, protocol 0806
```

```
layer 3 data: 00010800 06040001 00A0CC21 94C40500 01660000 00000000
```

```
05000102 00000000 00000000 00000000 00000000 000001FE
```

```
00000006 00000000 000003E8
```

```
...
```

```
DUT#undebg netdr
```

Make sure to turn it off afterwards

```
DUT#debug netdr clear-capture
```

Example of inbound packet on interface VLAN 1000

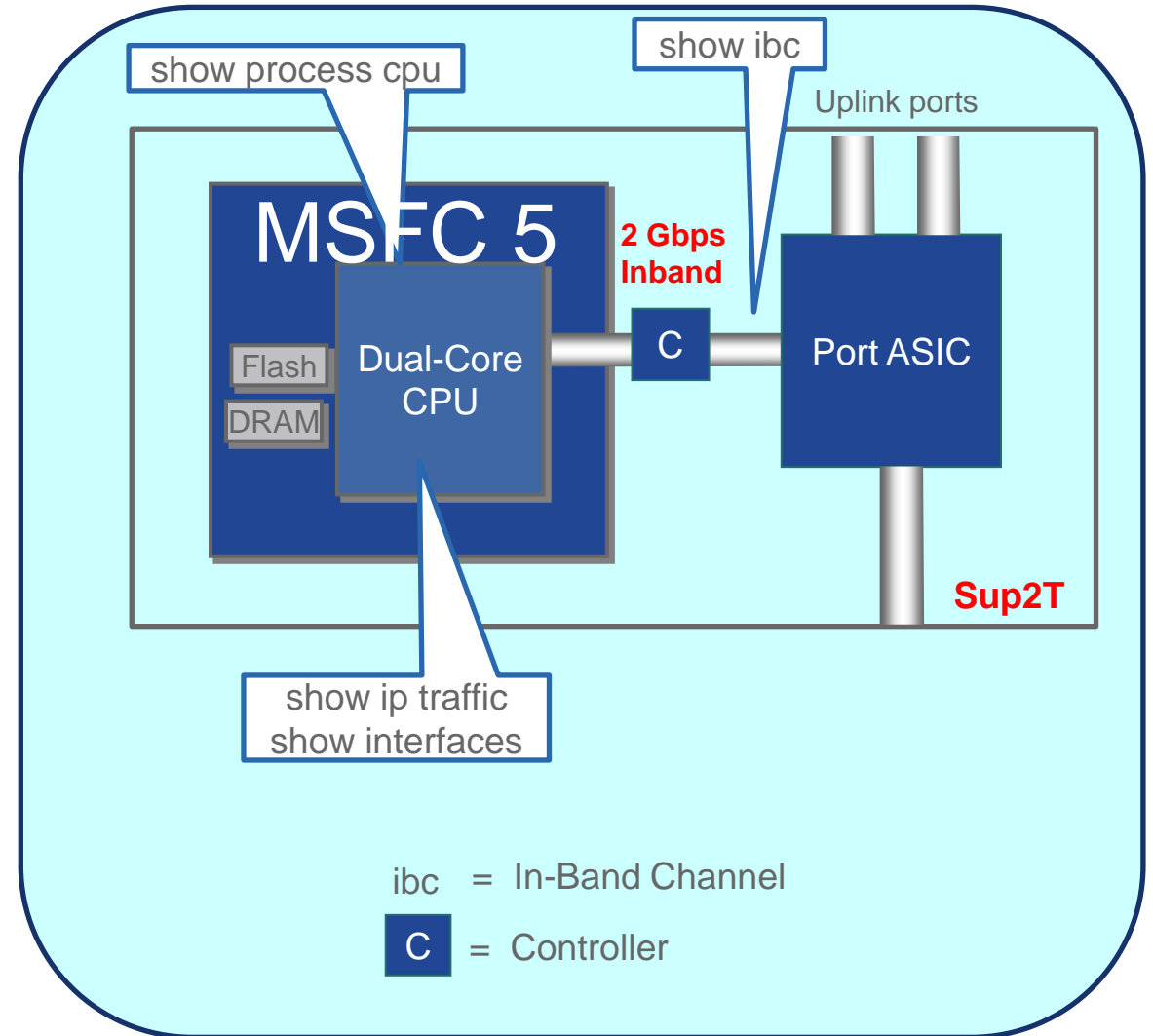
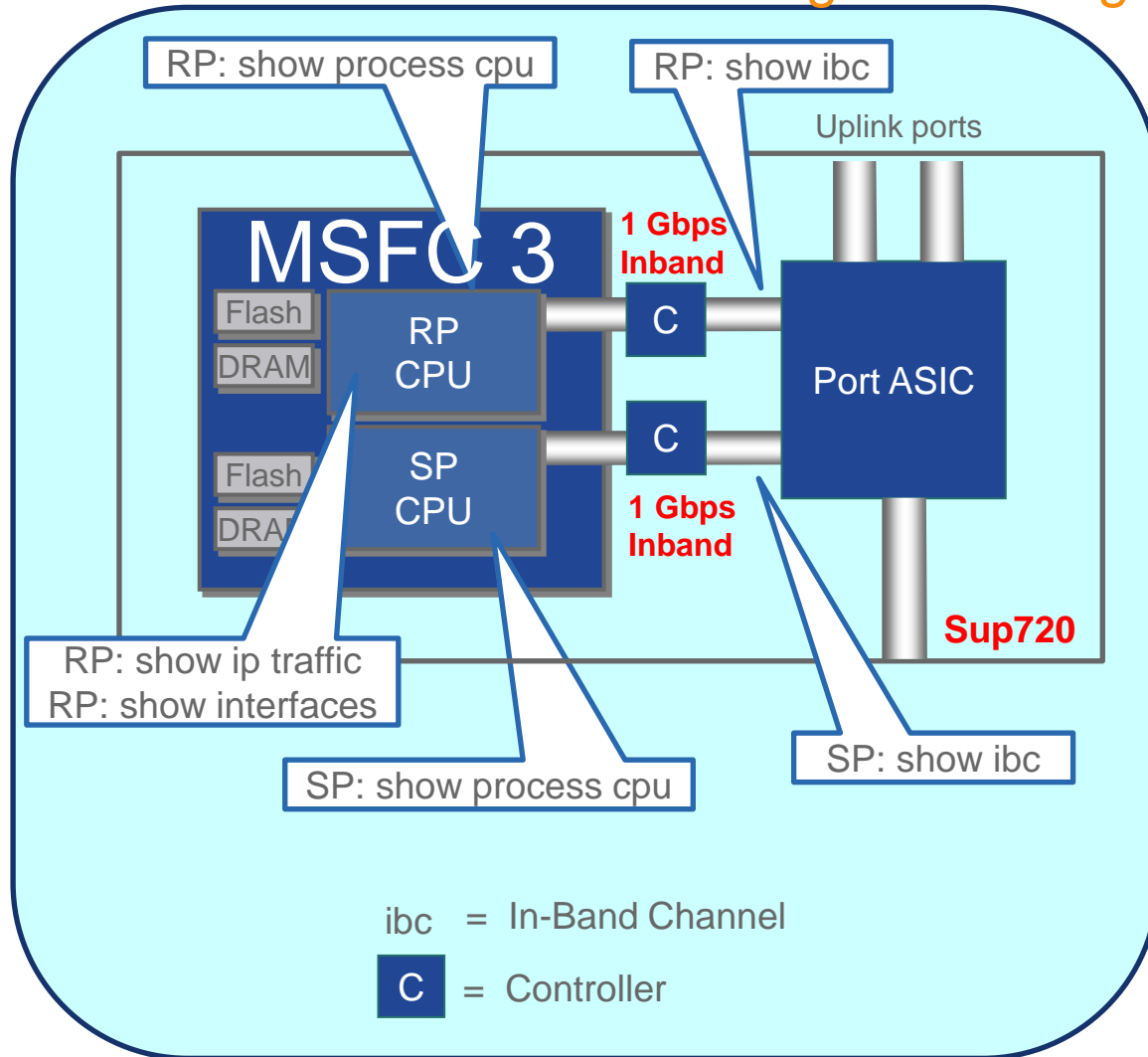
ARP packet

NetDR Parser Tool

- <http://www.cisco.com/c/en/us/support/web/tools-catalog.html>
- <http://netdr.54.227.241.219.xip.io/>

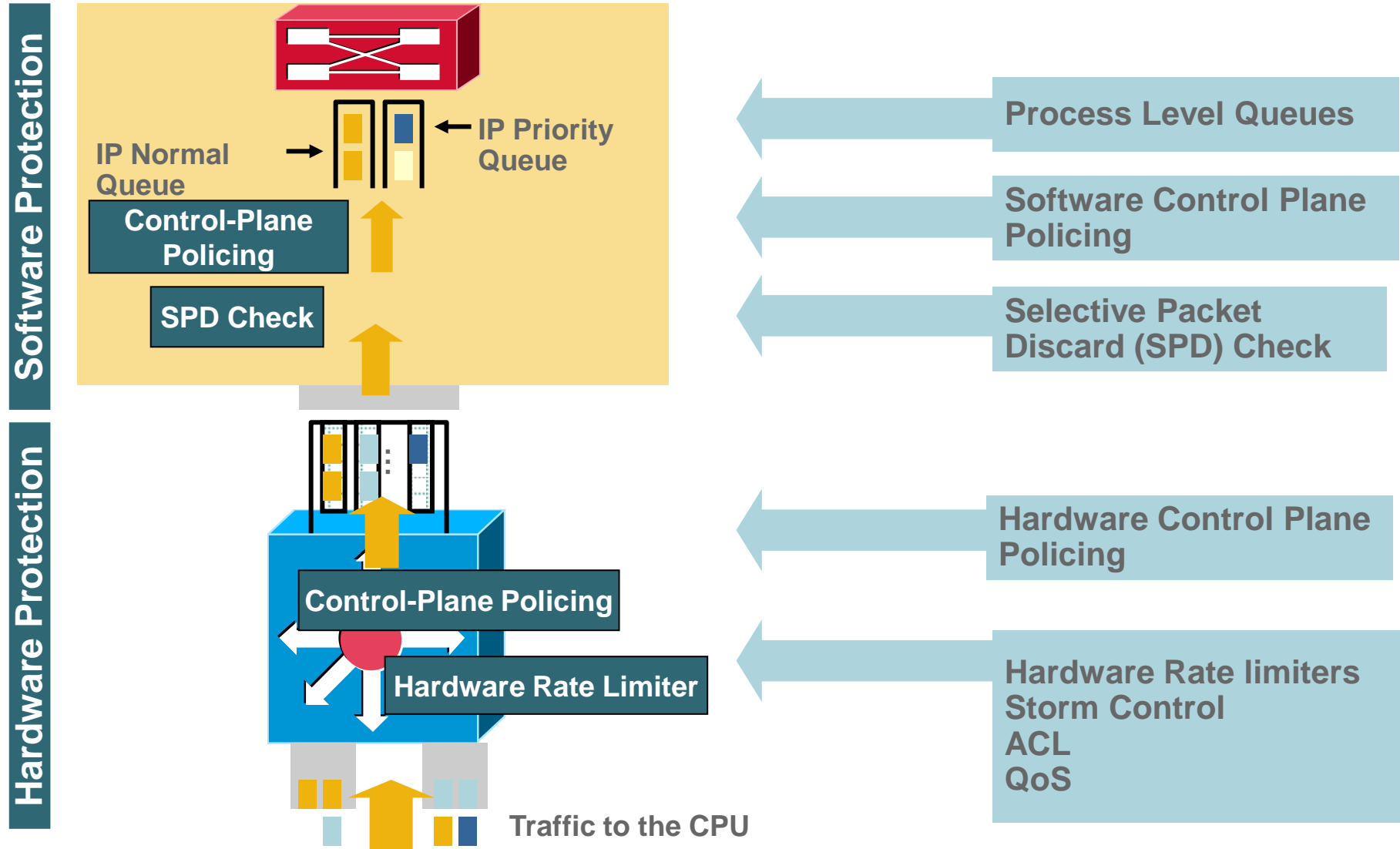
High CPU Utilization

Commands to troubleshooting CPU Usage



Control Plane Protection

Control Plane Protection



Control Plane Protection

PFC/DFC Hardware Rate Limiters Support

Unicast Rate Limiters	
CEF Receive	Traffic Destined to the Router
CEF Glean	ARP Packets
CEF No Route	Packets with Not Route in the FIB
ICMP Redirect	Packets that Require ICMP Redirects
IP Errors	Packet with IP Checksum or Length Errors
ICMP No Route	ICMP Unreachables for Unroutable Packets
ICMP ACL Drop	ICMP Unreachables for Admin Deny Packets
RPF Failure	Packets that Fail uRPF Check
L3 Security	CBAC, Auth-Proxy, and IPSEC Traffic
ACL Input	NAT, TCP Int, Reflexive ACLs, Log on ACLs
ACL Output	NAT, TCP Int, Reflexive ACLs, Log on ACLs
VACL Logging	CLI Notification of VACL Denied Packets
IP Options	Unicast Traffic with IP Options Set
Capture	Used with Optimized ACL Logging

Layer 2 Rate Limiters	
L2PT	L2PT Encapsulation/Decapsulation
PDU	Layer 2 PDUs

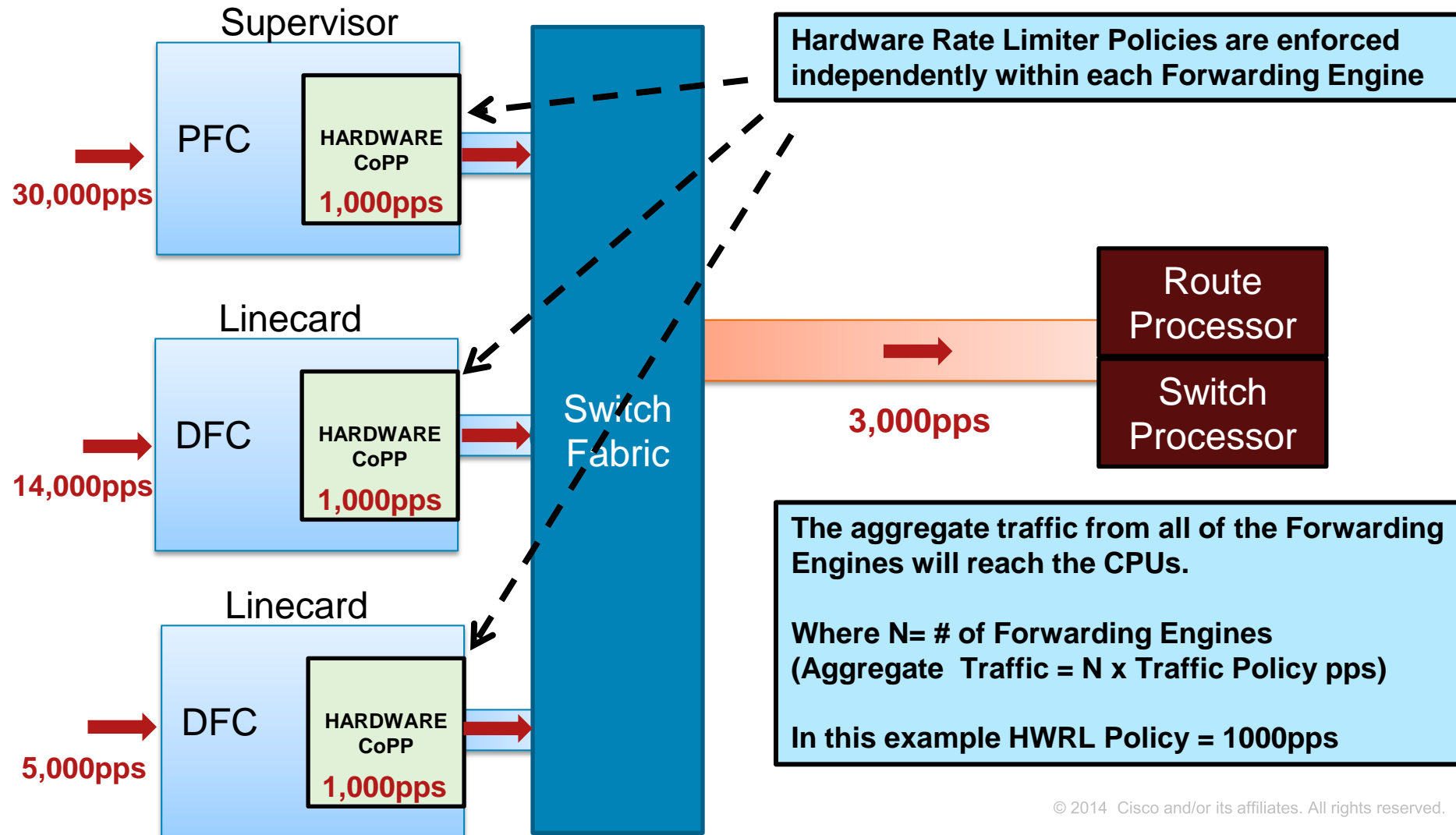
Multicast Rate Limiters	
Multicast FIB-Miss	Packets with No mroute in the FIB
IGMP	IGMP Packets (Actually Layer 2)
Partial Shortcut	Partial Shortcut Entries
Directly Connected	Local Multicast on Connected Interface
IP Options	Multicast Traffic with IP Options Set
V6 Directly Connect	Packets with No Mroute in the FIB
V6*, G M Bridge	IGMP Packets
V6*, G Bridge	Partial Shortcut Entries
V6 S, G Bridge	Partial Shortcut Entries
V6 Route Control	Partial Shortcut Entries
V6 Default Route	Multicast Traffic with IP Options Set
V6 Second Drop	Mulicast Traffic with IP Options Set

Shared Across the Ten Hardware Revocation Lists

General Rate Limiters	
MTU Failure	Packets Requiring Fragmentation
TTL Failure	Packets with TTL<=1

Control Plane Protection

Hardware Rate Limiters



Control Plane Protection

Hardware Rate Limiters

“show mls rate-limit”
for Sup720 engines.

- Rate-miters are implemented **in hardware**, to reduce flow of excess traffic to CPU

```
Sup2T# show platform rate-limit
State : ON - enabled but not sharing, ON/S - enabled and sharing
Share : NS - not sharing, G - group, S - static sharing, D - dynamic sharing
       : P/sec - Packets/sec, B/sec - Bytes/second, BP - Burst period (microsec)
Rate Limiter Type      State      P/sec  P/burst      B/sec  B/burst  BP      Share      Leak
-----
      CEF RECEIVE      OFF         -         -         -         -         -         -         -
      CEF GLEAN        ON        1000         -         -         -         1000000      NS      OFF
      IP ERRORS        OFF         -         -         -         -         -         -         -
UCAST IP OPTION        ON        1000         -         -         -         -         100 G: 0, S      ON
      ICMP ACL-DROP    ON      1000         -         -         -         100 G: 0, S    ON
      ICMP NO-ROUTE    ON         100         -         -         -         1000000      NS      OFF
      ICMP REDIRECT    OFF         -         -         -         -         -         -         -
      TTL FAILURE    OFF         -         -         -         -         -         -         -
```

Traffic hitting
ACL deny entry
are rate-limited

Traffic with
TTL=1 are **NOT**
rate-limited

```
Sup2T(config)#platform rate-limit ?
all          Rate Limiting for both Unicast and Multicast packets
layer2      layer2 protocol cases
multicast   Rate limiting for Multicast packets
unicast     Rate limiting for Unicast packets
```

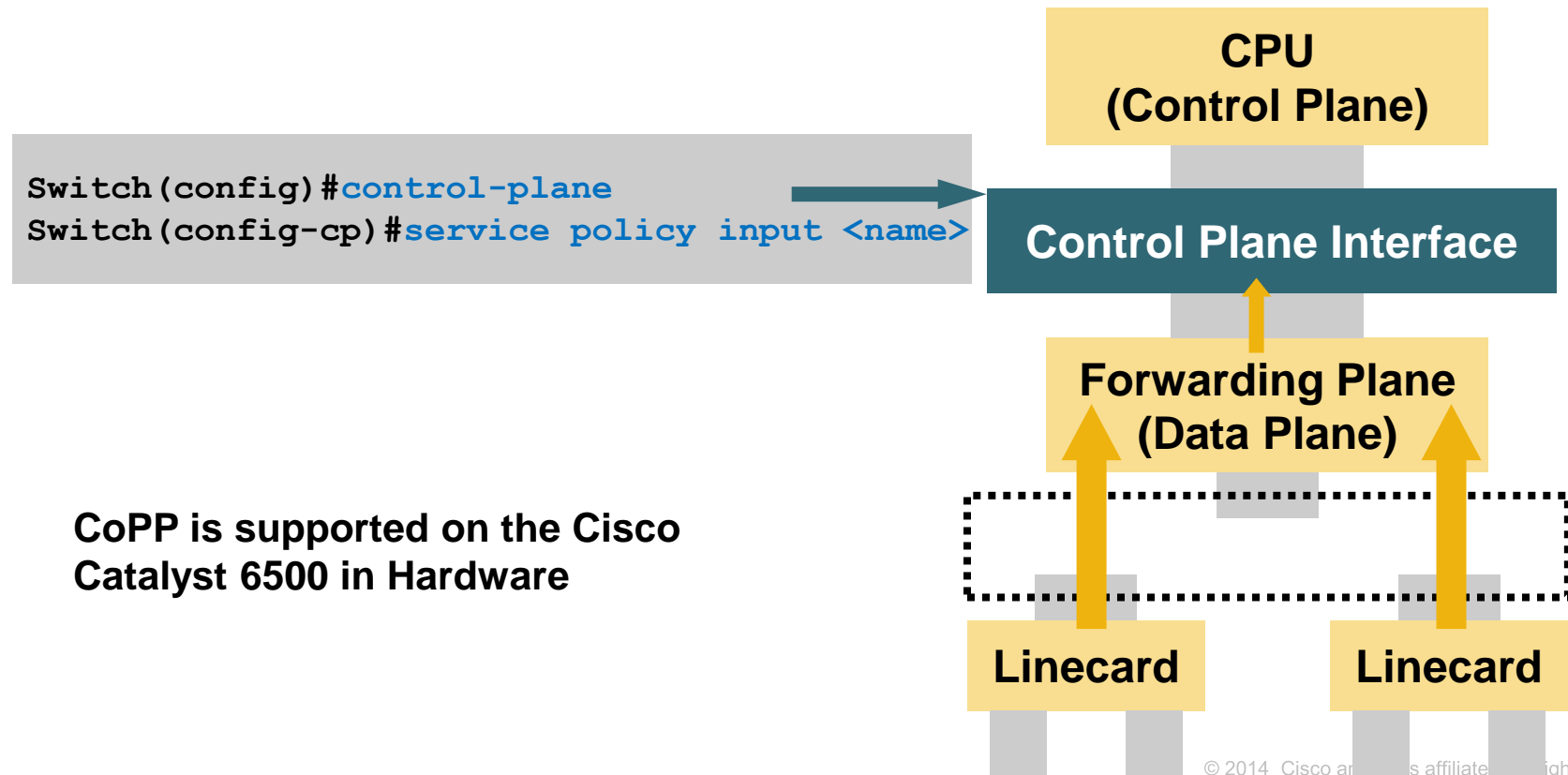
En/Disable and fine-tune
hardware rate-limiters.

Use “mls rate-limit” for
Sup720 engines.

Control Plane Protection

CoPP Support

- A new logical interface—the Control Plane Interface—has been introduced. A policer (service policy) can be applied on that interface thus limiting the **total** volume of traffic destined to the control plane. This mechanism is used to protect the operational integrity of the control plane



Control Plane Protection

CoPP Deployment—Step 1

- Step 1: Identify traffic of interest and classify it into multiple traffic classes:

BGP

IGP (EIGRP, OSPF, ISIS)

Management (telnet, TACACS, ssh, SNMP, NTP)

Reporting (SAA)

Monitoring (ICMP)

Critical applications
(HSRP, DHCP)

Undesirable

Default

```
ip access-list extended coppacl-bgp
permit tcp host 192.168.1.1 host 10.1.1.1 eq bgp
permit tcp host 192.168.1.1 eq bgp host 10.1.1.1
!
ip access-list extended coppacl-igp
permit ospf any host 224.0.0.5
permit ospf any host 224.0.0.6
permit ospf any any
!
ip access-list extended coppacl-management
permit tcp host 10.2.1.1 host 10.1.1.1 established
permit tcp 10.2.1.0 0.0.0.255 host 10.1.1.1 eq 22
permit tcp 10.86.183.0 0.0.0.255 any eq telnet
permit udp host 10.2.2.2 host 10.1.1.1 eq snmp
permit udp host 10.2.2.3 host 10.1.1.1 eq ntp
!
ip access-list extended coppacl-reporting
permit icmp host 10.2.2.4 host 10.1.1.1 echo
!
ip access-list extended coppacl-monitoring
permit icmp any any ttl-exceeded
permit icmp any any port-unreachable
permit icmp any any echo-reply
permit icmp any any echo
!
ip access-list extended coppacl-critical-app
permit ip any host 224.0.0.1
permit udp host 0.0.0.0 host 255.255.255.255 eq bootps
permit udp host 10.2.2.8 eq bootps any eq bootps
!
ip access-list extended coppacl-undesirable
permit udp any any eq 1434
```


Control Plane Protection

CoPP Deployment—Step 2

- Step 2: Associate the identified traffic with a class, and permit the traffic in each class

Must enable QoS globally, else CoPP will not be applied in hardware

HW CoPP classes are limited to one match per class-map

```
mls qos
```

```
class-map match-all copp-bgp
  match access-group name coppacl-bgp
class-map match-all copp-igp
  match access-group name coppacl-igp
class-map match-all copp-management
  match access-group name coppacl-management
class-map match-all copp-reporting
  match access-group name coppacl-reporting
class-map match-all copp-monitoring
  match access-group name coppacl-monitoring
class-map match-all copp-critical-app
  match access-group name coppacl-critical-app
class-map match-all copp-undesirable
  match access-group name coppacl-undesirable
```

```
policy-map copp-policy
  class copp-bgp
    police 3000000 conform-action transmit exceed-action drop
  class copp-igp
    police 3000000 conform-action transmit exceed-action drop
  class copp-management
    police 3000000 conform-action transmit exceed-action drop
  class copp-reporting
    police 3000000 conform-action transmit exceed-action drop
  class copp-monitoring
    police 3000000 conform-action transmit exceed-action drop
  class copp-critical-app
    police 3000000 conform-action transmit exceed-action drop
  class copp-undesirable
    police 3000000 conform-action transmit exceed-action drop
  class class-default
    police 3000000 conform-action transmit exceed-action drop
```

```
control-plane
  service-policy input copp-policy
```

Control Plane Protection

CoPP Deployment—Step 3

- Step 3: Adjust classification, and apply liberal CoPP policies for each class of traffic
- **show policy-map control-plane** displays dynamic information for monitoring control plane policy. Statistics include rate information and number of packets/ bytes confirmed or exceeding each traffic class
- CoPP rates on Sup720 are bps—pps is not possible. However, HWRL rates are in pps

```
Switch# show policy-map control-plane
Control Plane Interface
Service-policy input: copp-policy
<snip>
Hardware Counters:
class-map: copp-monitoring (match-all)
Match: access-group name coppacl-monitoring
police :
 30000000 bps 937000 limit 937000 extended limit
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
Earl in slot 7 :
112512 bytes
 5 minute offered rate 3056 bps
aggregate-forwarded 112512 bytes action: transmit
exceeded 0 bytes action: drop
aggregate-forward 90008 bps exceed 0 bps
Software Counters:
Class-map: copp-monitoring (match-all)
1036 packets, 128464 bytes
 5 minute offered rate 4000 bps, drop rate 0 bps
Match: access-group name coppacl-monitoring
police:
  cir 30000000 bps, bc 937500 bytes
conformed 1036 packets, 128464 bytes; action: transmit
exceeded 0 packets, 0 bytes; action: drop
conformed 4000 bps, exceed 0 bps
<snip>
```

Control Plane Protection

CoPP Deployment—Step 4

- Step 4: Fine tune the control plane policy
 - Narrow the ACL permit statements to only allow known authorized source addresses and depending on class defined, apply appropriate policy

Routing protocol traffic—**no rate limit or very conservative rate limit**

Management traffic—**conservative rate limit**

Reporting traffic—**conservative rate limit**

Monitoring traffic—**conservative rate limit**

Critical traffic—**conservative rate limit**

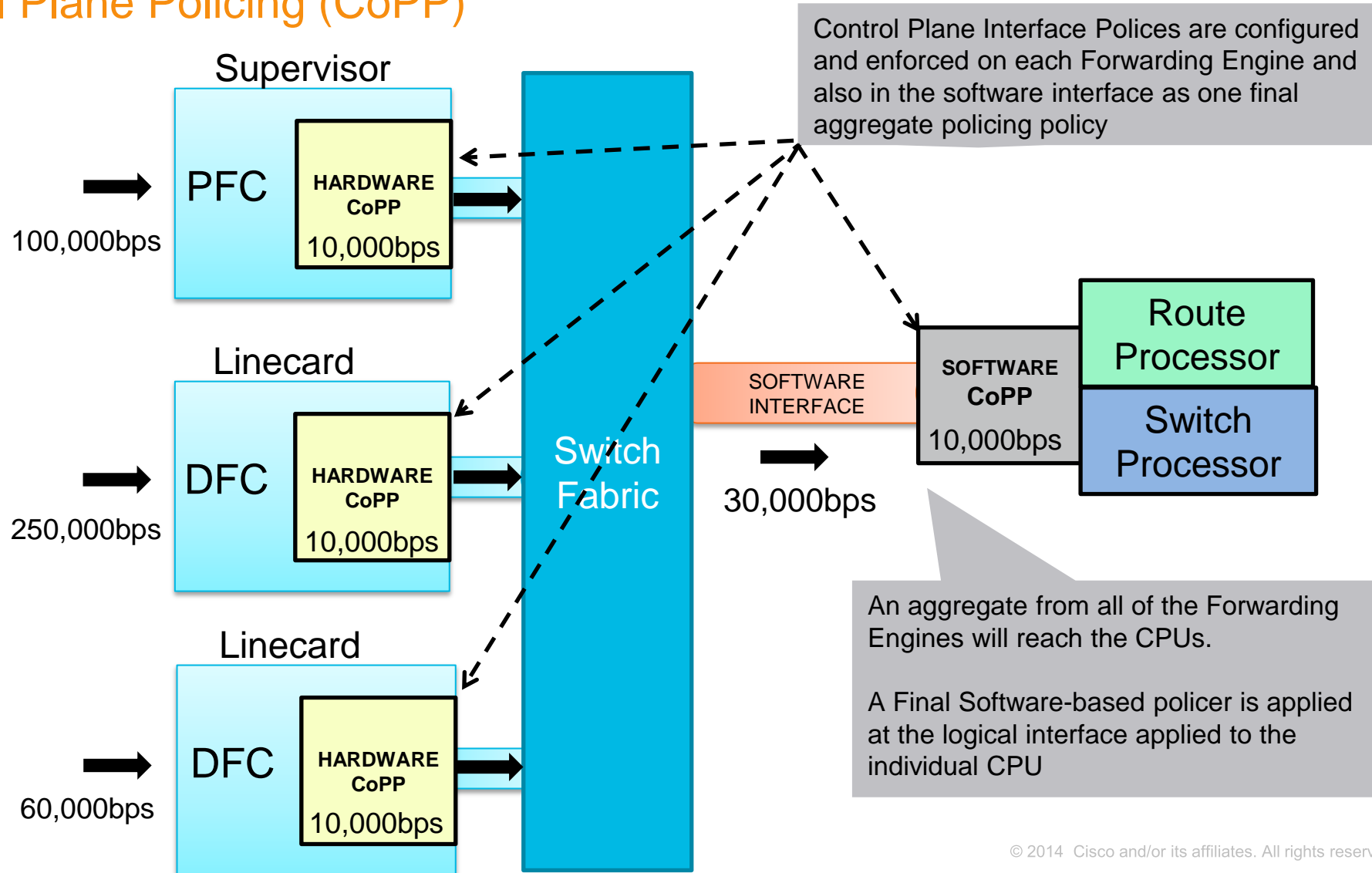
Default traffic—**low rate limit**

Undesirable traffic—**drop**

```
policy-map copp-policy
  class coppclass-bgp
    police 15000000 conform-action transmit exceed-action drop
  class coppclass-igp
    police 15000000 conform-action transmit exceed-action drop
  class coppclass-management
    police 2560000 conform-action transmit exceed-action drop
  class coppclass-reporting
    police 1000000 conform-action transmit exceed-action drop
  class coppclass-monitoring
    police 1000000 conform-action transmit exceed-action drop
  class coppclass-critical-app
    police 7500000 conform-action transmit exceed-action drop
  class coppclass-undesirable
    police 32000 conform-action transmit exceed-action drop
  class class-default
    police 1000000 conform-action transmit exceed-action drop
```

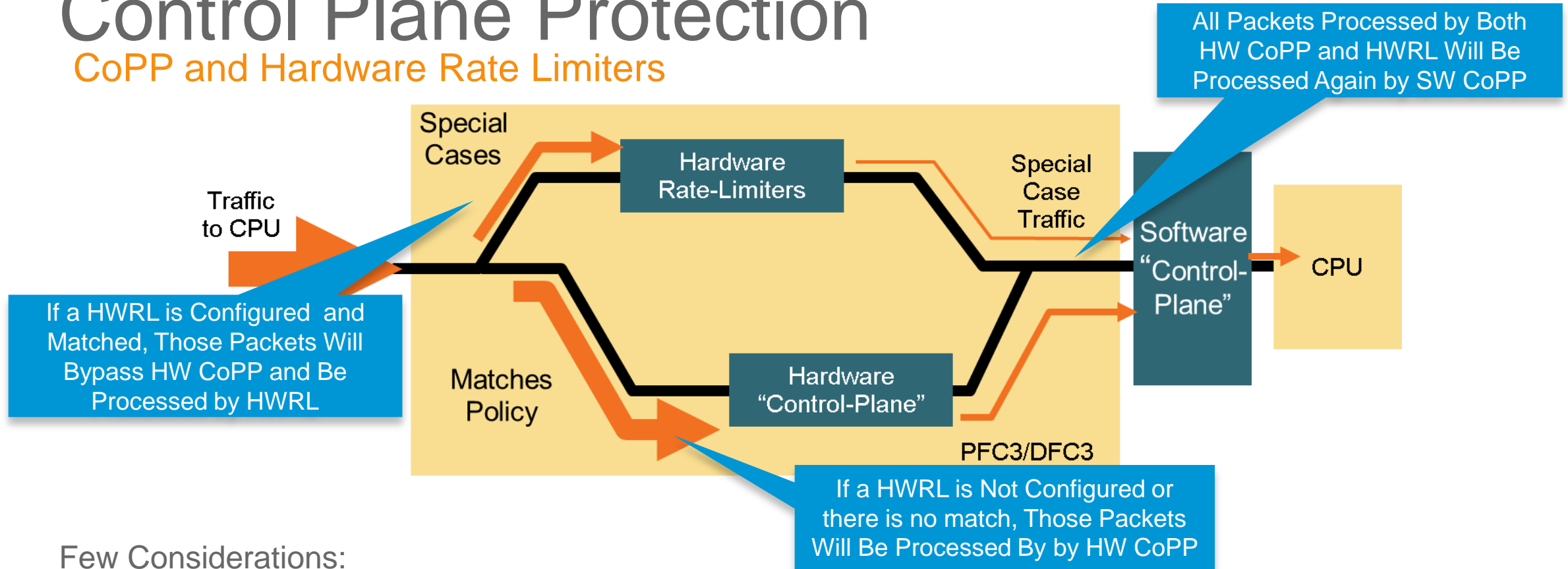
Control Plane Protection

Control Plane Policing (CoPP)



Control Plane Protection

CoPP and Hardware Rate Limiters



Few Considerations:

- When a packet matches both HW CoPP and HWRL, the packet undergoes HWRL policy and skips HW CoPP. In essence, HWRL **overrides** HW CoPP.
- Configure the CEF receive rate limiter with care

Given that the CEF receive rate-limiter matches all traffic destined to the Route Process ("good" frames and "bad" frames) and takes precedence over CoPP, it is best to only use CoPP instead.

Control Plane Protection

CoPP Deployment Considerations

Top deployment considerations:

- No HW CoPP processing unless “mls qos” is enabled: this enables also port-level QoS mechanisms
- HW CoPP will ignore a class that does not have a corresponding policing action
- HW CoPP decisions are per forwarding engines
 - SW CoPP for the aggregate traffic
- HW CoPP does not support ARP/ broadcast/ multicast traffic
 - Use multicast HWRL/Dynamic ARP Inspection or “mls qos protocol arp”/Storm Control in conjunction
 - Remember, software CoPP will still match multicast and broadcast traffic, so you MUST classify these packets in CoPP policies

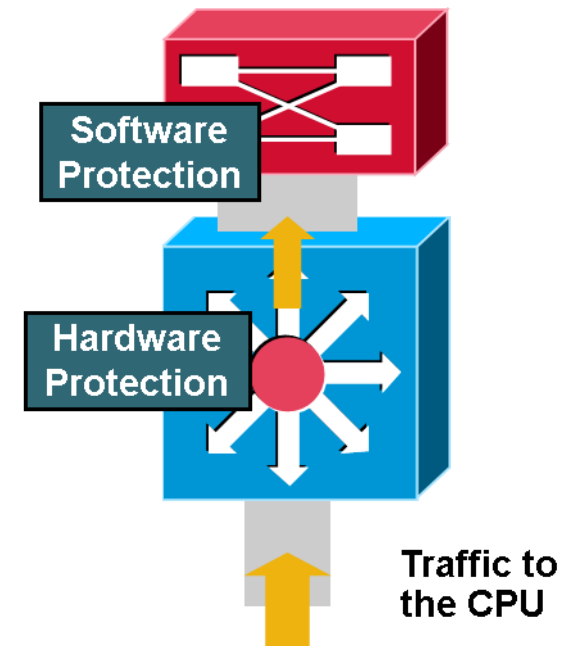
http://www.cisco.com/application/pdf/en/us/guest/netso/ns171/c649/ccmigration_09186a0080747eb2.pdf

http://www.cisco.com/en/US/products/hw/switches/ps708/products_white_paper0900aecd802ca5d6.shtml

Control Plane Protection

Summary

- The Catalyst 6500 supports hardware mechanisms for control plane protection
- Multiple levels of hardware and software protection are available to protect the CPU
- Whenever possible, CoPP should be used as the preferred method to mitigate attacks since it offers more granularity than other mechanisms



Selective Packet Discard

- SPD was designed to help ensure that network stability was not undermined during periods of high CPU-bound traffic.
- SPD was created to provide preferential treatment during congestion of this interface-to-process-switching queue. It provides extended buffering for control plane traffic, such as IGP, BGP, L2 keepalives, etc.

```
6509-S#show ip spd
```

```
Current mode: normal.
```

```
Queue min/max thresholds: 73/74, Headroom: 1000, Extended Headroom: 10
```

```
IP normal queue: 0, priority queue: 0.
```

```
SPD special drop mode: none
```



Based on IP Precedence

Selective Packet Discard

```
6509-S#show int g 5/2
```

Last clearing of "show interface" counters never

```
Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 17077812
```

```
Queueing strategy: fifo
```

```
Output queue: 0/40 (size/max)
```

```
5 minute input rate 11000 bits/sec, 13 packets/sec
```

```
5 minute output rate 14000 bits/sec, 20 packets/sec
```

```
6509-S(config)#int g 5/2
```

```
6509-S(config-if)#hold-queue 1000 in
```



Input hold-queue



Adjust hold-queue size

Selective Packet Discard

Parameter Name	Scope of Configuration	Scope of Operation	Default Value
hold-queue	Physical Interface	Physical Interface	75
Headroom	Global	Physical Interface	1000
Extended Headroom	Global	Physical Interface	10
min-threshold	Global	Physical Interface	(Smallest hold-queue on device) - 2
max-threshold	Global	Physical Interface	(Smallest hold-queue on device) - 1

```
6509-S(config)# ip spd queue min-threshold value
6509-S(config)# ip spd queue max-threshold value
6509-S(config)# spd headroom value
6509-S(config)# spd extended value
```

Selective Packet Discard

SPD State Check

Behavior Relative to Queue Length of a Single Interface	Selective Packet Discard State	Queuing Allowed by Packet Classification		
		hold-queue	Headroom	Extended Headroom
$\text{queue length} \leq \text{min-threshold}$	Normal	●	●	●
$\text{min-threshold} < \text{queue length} \leq \text{max-threshold}$	Random Drop	● *	●	●
$\text{max-threshold} < \text{queue length} \leq (\text{hold-queue} + \text{headroom})$	Full Drop		●	●
$(\text{hold-queue} + \text{headroom}) < \text{queue length} \leq (\text{hold-queue} + \text{headroom} + \text{extended headroom})$	Full Drop			●
$(\text{hold-queue} + \text{headroom} + \text{extended headroom}) < \text{queue length}$	Full Drop			

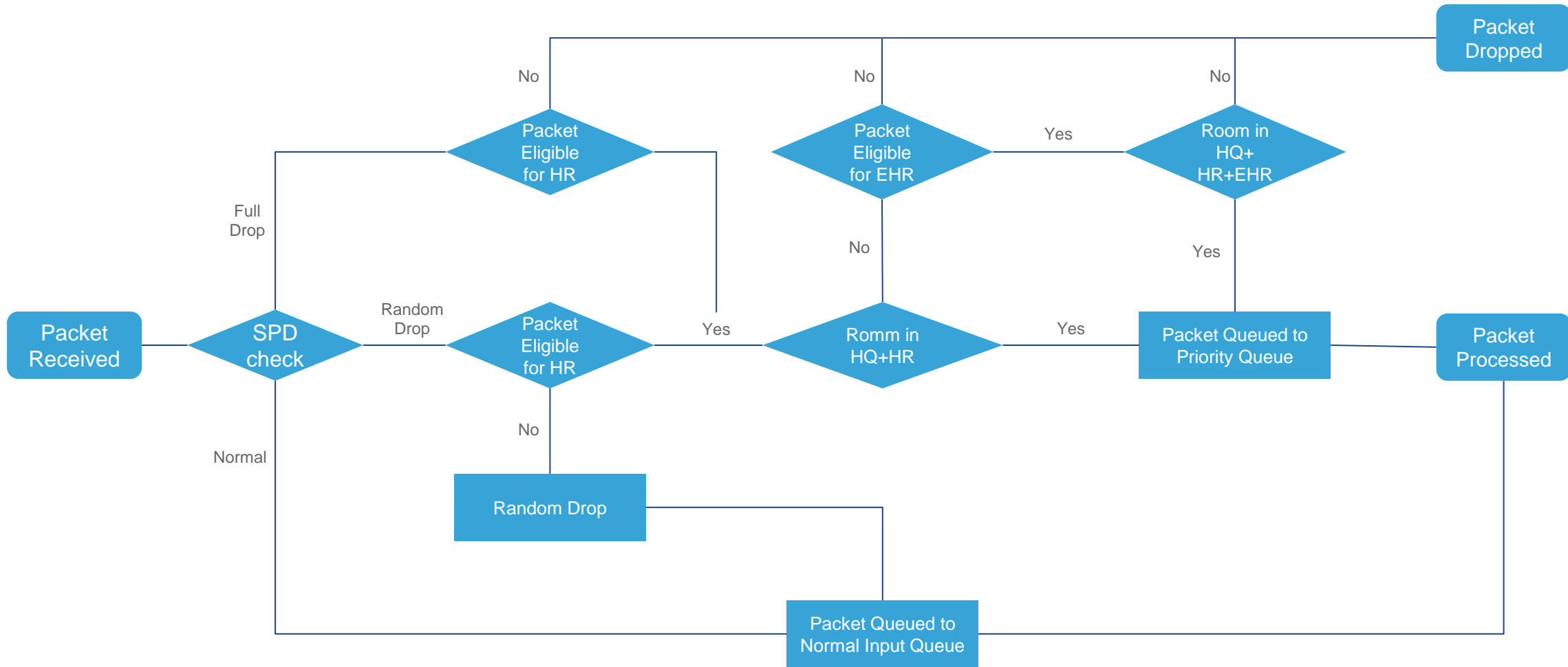
* Packets not eligible to be queued into headroom are subject to a random drop congestion-control mechanism.

```
6509-S#show ip spd
```

```
Current mode: normal.
```

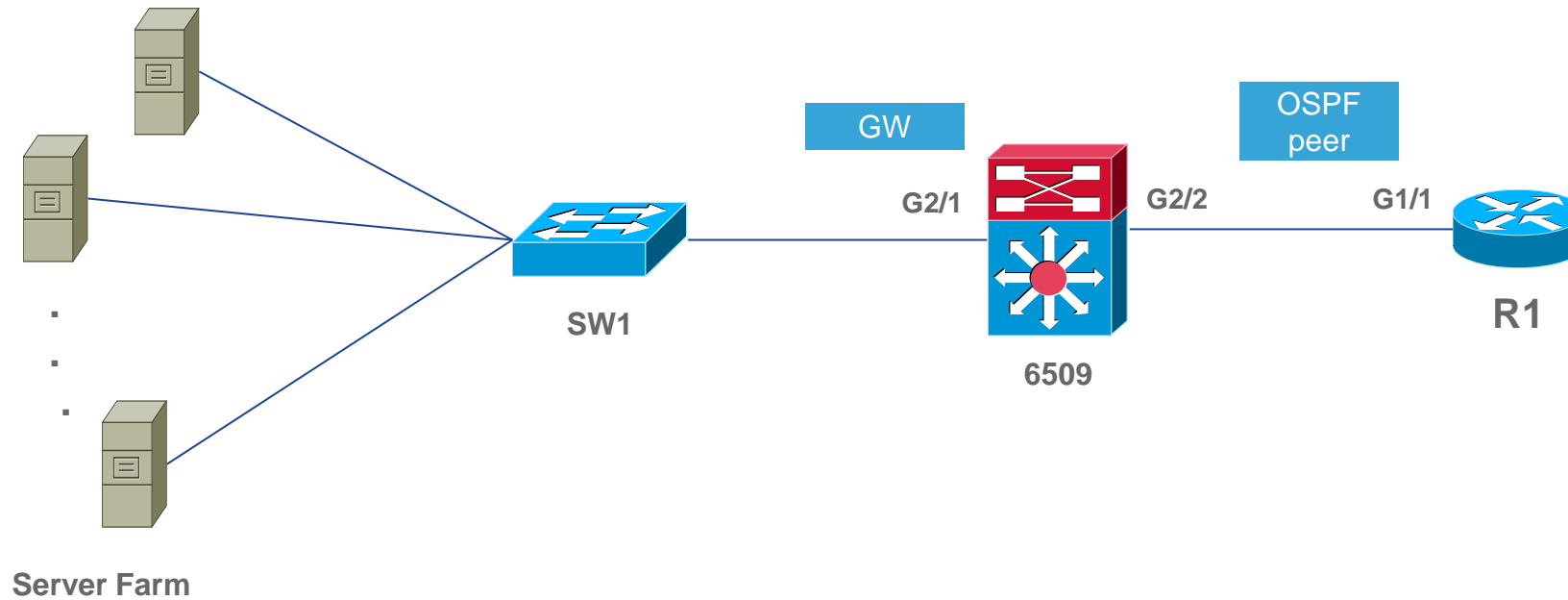
Selective Packet Discard

SPD Operation



SPD troubleshooting sharing

Topology



1. Servers need to monitor the availability of the gateway through several ping tests every second.
2. OSPF neighborhood with R1 keep stable.
3. Ping loss is found from R1 to 6509

SPD troubleshooting sharing

```
6509-S#show int g 2/2
```

```
Last clearing of "show interface" counters never
Input queue: 0/75/448/448 (size/max/drops/flushes); Total output drops: 0
Queueing strategy: fifo
Output queue: 0/40 (size/max)
5 minute input rate 23000 bits/sec, 25 packets/sec
5 minute output rate 18000 bits/sec, 24 packets/sec
```

```
6509-S#sh int g2/2 switching
```

```
GigabitEthernet2/2
  Throttle count          0
  Drops                   RP      448      SP          0
  SPD Flushes             Fast    448      SSE          0
  SPD Aggress             Fast          0
  SPD Priority            Inputs  0        Drops        0

  Protocol    Path    Pkts In  Chars In  Pkts Out  Chars Out
  IP          Process  4450    422443    6665     895558
  Cache misses
  Fast
  Auton/SSE
```

SPD drop is happening

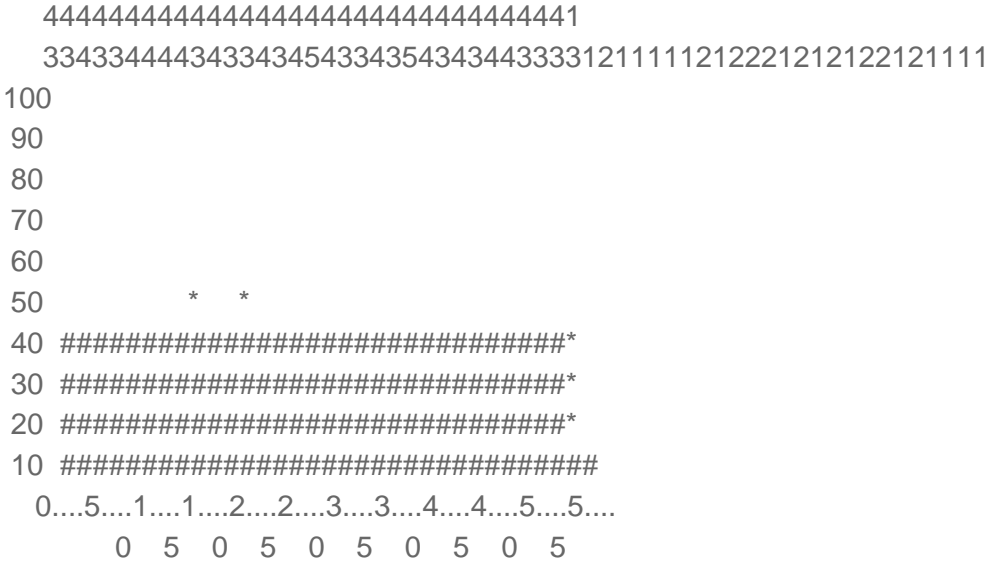
SPD troubleshooting sharing

6509-S#sh processes cpu sort

CPU utilization for five seconds: 43%/30%; one minute: 42%; five minutes: 42%

PID	Runtime(ms)	Invoked	uSecs	5Sec	1Min	5Min	TTY	Process
110	240928	483932	497	12.71%	12.15%	12.09%	0	IP Input
261	1552	3543	438	0.23%	0.01%	0.00%	0	Exec
155	1120	6225	179	0.07%	0.02%	0.00%	0	CEF process
4	6732	601	11201	0.00%	0.14%	0.15%	0	Check heaps
3	28	758	36	0.00%	0.00%	0.00%	0	OSPF Hello

6509-S#sh processes cpu history



CPU is not busy enough here, it should be able to reply our ping requests.

CPU% per minute (last 60 minutes)
 * = maximum CPU% # = average CPU%

SPD troubleshooting sharing

```
6509-s#sh ip spd
```

Current mode: **random drop**.

Queue min/max thresholds: 73/74, Headroom: 1000, Extended Headroom: 10

IP normal queue: **74**, priority queue: 11.

SPD special drop mode: none

```
6509-2#sh int g2/1 | in Input queue|rate
```

Input queue: **74/75/8271456/8271456** (size/max/drops/flushes); Total output drops: 0

Queueing strategy: fifo

30 second input rate **7619000** bits/sec, 14881 packets/sec

30 second output rate 1000 bits/sec, 1 packets/sec

We could see SPD state has been changes to random drop due to excessive packets are punted to CPU from G2/1

Workaround:

Increase the SPD min/max threshold and the input hold-queue of G2/1 together, to avoid the SPD goes into drop mode. **But this will increase the CPU utilization also.**

SPD troubleshooting sharing

Conclusion

- SPD take care interface-to-process-switching queue globally, not per interface
- Only increase hold-queue size will not bring the system out from SPD drop state
- If we want to protect the traffic which need to software processed, CPU resources would be sacrificed.

Troubleshooting High CPU Utilization

Summary

- CPU Architecture and Hardware Forwarding Path
- Some of the causes for high CPU utilization - due to processes and interrupts
- Various commands and tools available to troubleshoot High CPU condition
- Using Hardware Rate-Limiter, Control-Plane Policing (CoPP) to protect and monitor control-plane

Take Away Points

- It is very important to set baseline CPU usage and traffic level under normal working conditions, and find deviation when CPU usage spikes.
- It is critical to protect CPU, for a stable network.
- Closely monitor and justify usage of all the hardware resources.



Thank You !



CISCO

TOMORROW starts here.



Catalyst 6500 – High CPU Troubleshooting

Reference Materials

- Catalyst 6500/6000 Switch High CPU Utilization

<http://www.cisco.com/c/en/us/support/docs/switches/catalyst-6500-series-switches/63992-6k-high-cpu.html>

- Troubleshooting High CPU Utilization due to Processes

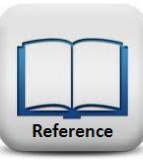
<http://www.cisco.com/c/en/us/support/docs/routers/7500-series-routers/41180-highcpu-processes.html>

- Troubleshooting High CPU Utilization in IP Input Process

<http://www.cisco.com/c/en/us/support/docs/routers/7500-series-routers/41160-highcpu-ip-input.html>

- Troubleshooting tools to analyze high CPU utilization issues on Catalyst 6500 Series switches

<https://supportforums.cisco.com/docs/DOC-22037>



Catalyst 6500 – Protecting Control Plane

Reference Materials

- Protecting Cisco Catalyst 6500 Series Switches Using Control Plane Policing, Hardware Rate Limiting, and Access-Control Lists
http://www.cisco.com/en/US/prod/collateral/switches/ps5718/ps708/white_paper_c11_553261.html
- Protecting the Cisco Catalyst 6500 Series Switches Against Denial-Of-Service Attacks
http://www.cisco.com/en/US/prod/collateral/switches/ps5718/ps708/prod_white_paper0900aecd802ca5d6.html
- Control Plane Policing Implementation Best Practices (general and platform specific)
http://www.cisco.com/web/about/security/intelligence/coppwp_gs.html
- Borderless Networks Security: Catalyst 6500 Control plane Protection Techniques for Maximum Uptime
http://www.cisco.com/en/US/prod/collateral/switches/ps5718/ps708/white_paper_c11-663623.html
- Cisco Catalyst 6500 Supervisor Engine 2T: NetFlow Enhancements
http://www.cisco.com/en/US/prod/collateral/switches/ps5718/ps708/white_paper_c11-652021.html

Understanding SPD

- <http://www.cisco.com/c/en/us/support/docs/routers/12000-series-routers/29920-spd.html?mdfid=268437990>
- Understanding Selective Packet Discard (SPD)
- <http://www.cisco.com/web/about/security/intelligence/spd.html>
- Understanding and Using Selective Packet Discard