

CISCO *Live!*

GO BEYOND

#CiscoLiveAPJC



Troubleshooting the Cisco Catalyst 9000 Series Switches

Jacky Lee
High Touch Engineer
BRKTRS-2811



#CiscoLiveAPJC

Cisco Webex App

Questions?

Use Cisco Webex App to chat with the speaker after the session

How

- 1 Find this session in the Cisco Live Mobile App
- 2 Click “Join the Discussion”
- 3 Install the Webex App or go directly to the Webex space
- 4 Enter messages/questions in the Webex space

Webex spaces will be moderated by the speaker until November 15, 2024.





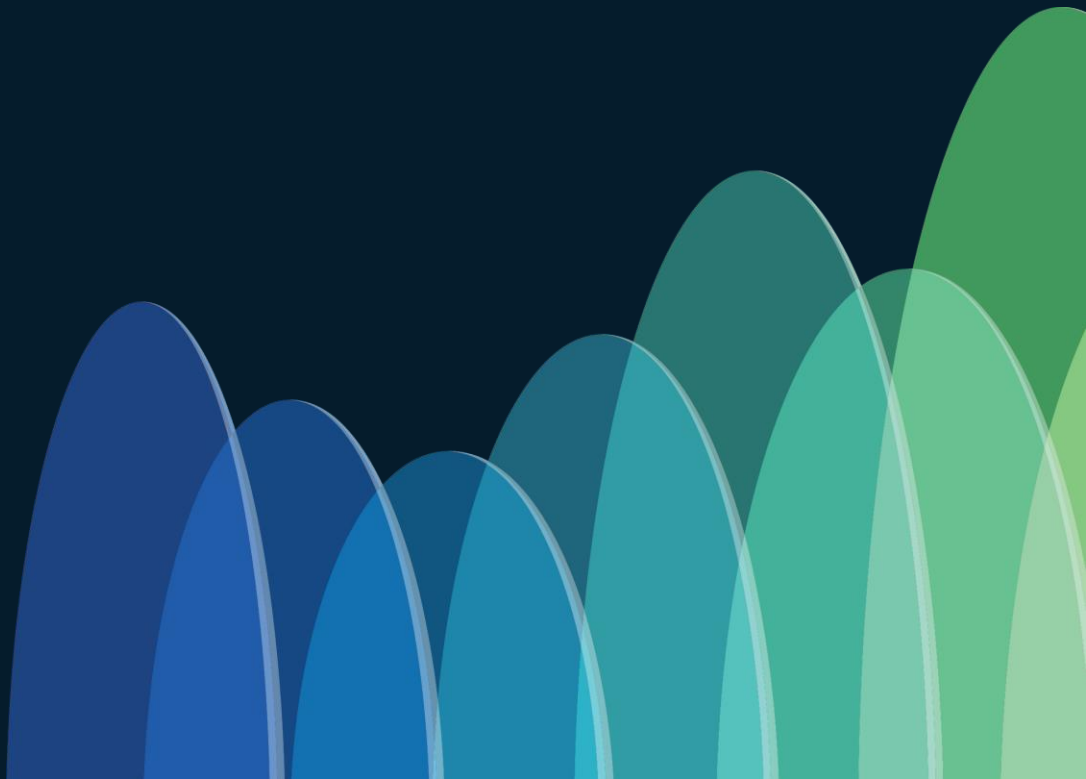
Agenda

- Introduction
- Platform Troubleshooting
- Packet Drops
- Forwarding Verification



PINK FLANNEL FLOWER

Introduction



Catalyst 9k family



	9200/9200L	9300/9300X	9400	9500/9500X	9600/9600X
Format	Stackable	Stackable	Chassis	Standalone	Chassis

All Catalyst 9000 series switches are based upon UADP and Silicon One ASICs and run IOS-XE.

Different feature sets and performance but same architecture

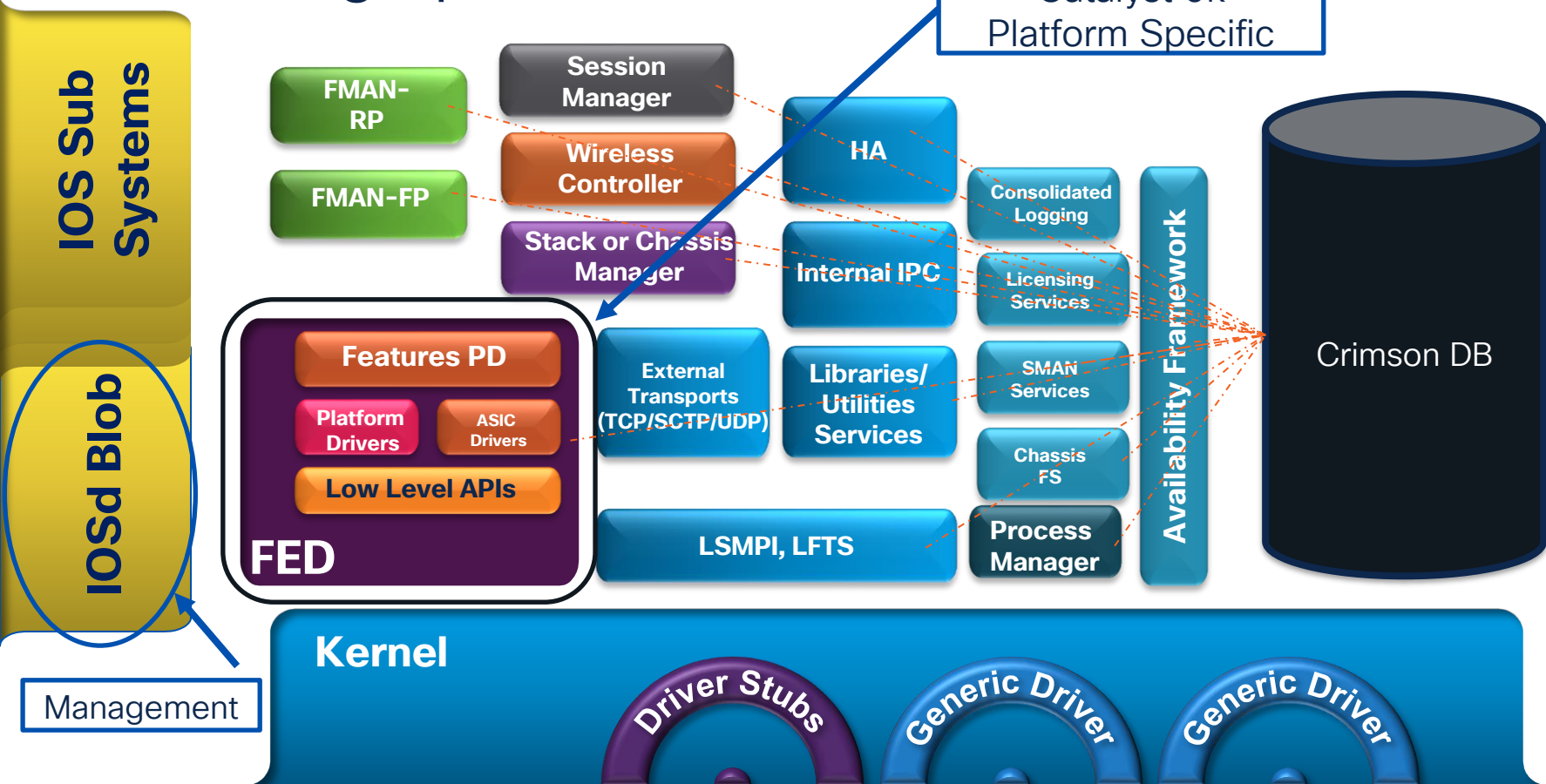


Catalyst IOS-XE Software release schedule

	Bengaluru 17.6	Cupertino 17.9	Dublin 17.12	17.13	17.14	17.15
Next planned release	17.6.7	17.9.6	17.12.4	na	na	na
End of Software Maintenance	March 2024	March 2025	March 2026	Dec 2024	March 2025	na
9200/9300/9300X 9400/9500/9600	Yes	Yes	Yes	Yes	Yes	Yes
9500X/9600X	No	Yes	Yes	Yes	Yes	Ye
Extended Maintenance Release	Yes	Yes	Yes	No	No	No

Recommended releases: <https://www.cisco.com/c/en/us/support/docs/switches/catalyst-9300-series-switches/214814-recommended-releases-for-catalyst-9200-9.html>

IOS-XE graphical overview



Platform Specific Show Commands

- Legacy IOS commands exist on the Catalyst platform for troubleshooting.
Ex: **show mac address-table**, **show ip route**, **show aaa**, **debug ip pim**.
- On IOS-XE many processes outside IOSd are non-platform specific
Ex: Forwarding manager (fman) , WebUI (nginx), Wireless (wncd)
- Debug/show commands similar on all IOS-XE Devices

```
Switch#show platform software ip switch active R0 cef  
ASR_1k#show platform software ip rp active cef  
Cat8kv#show platform resources
```

- Catalyst 9K platform specific commands:

show/set/debug platform hardware | software FED

```
9300#show platform software fed switch active ifm mappings  
9300#show platform software fed switch 5 ifm mappings  
9400#show platform software fed active ifm mappings
```

Debugging outside IOSd process

- To facilitate debugging/logging trace logs are available for all processes using always on tracing
- Tracing levels set with granularity (default notice). Always on tracing
- Common processes:
 - > **smd** - session manager
 - > **fed** - forwarding engine driver
 - > **wncd** - wireless process

```
Switch#set platform software trace smd switch active R0 dot1x-all verbose
Switch#show platform software trace level smd switch active R0 | inc dot1x
dot1x                                Notice
dot1x-all                          Verbose
dot1x-redun                         Notice
Switch#set platform software trace all notice ← set all back to default
```

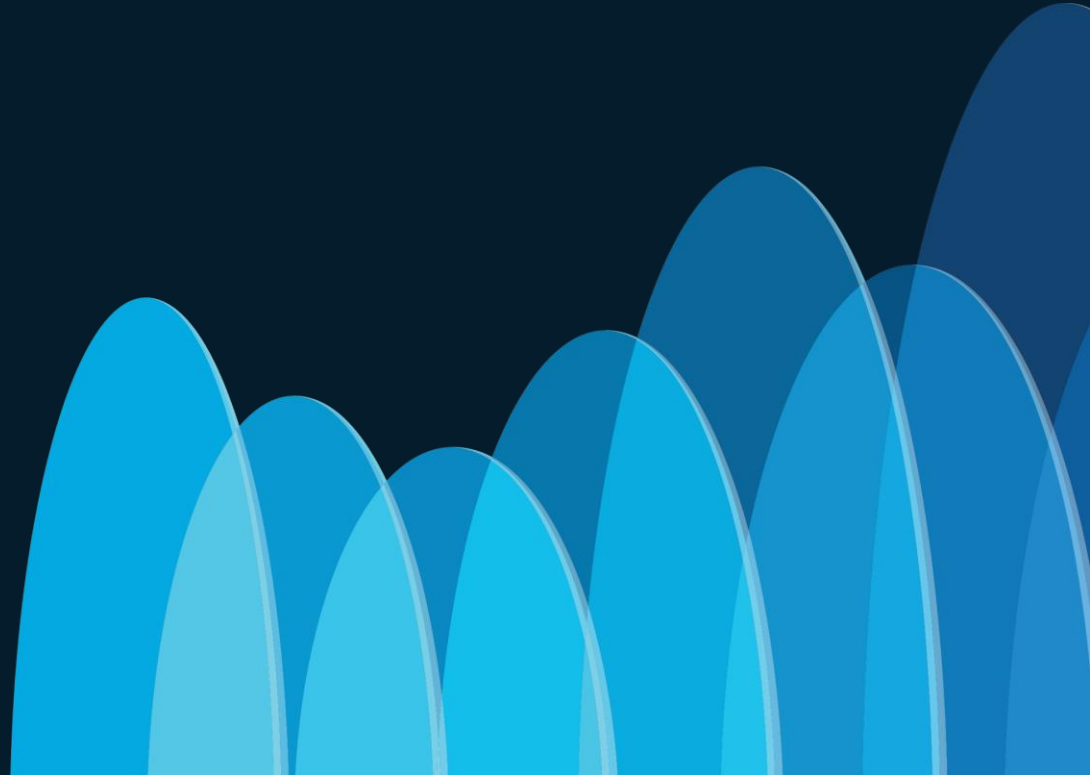
Displaying trace logs

- Tracelog files are stored in crashinfo:/logs in binary format.
- Analyzing traces: show logging process <process>
- Bundle creation of binary traces: “request platform software trace archive”

```
9300#sh logging process smd | inc RADIUS
2022/06/06 23:24:03.268912 {smd_R0-0}{1}: [radius] [24732]: (info): RADIUS: Send Accounting-Request to
10.48.91.222:1813 id 1813/184, len 850
2022/06/06 23:24:03.268937 {smd_R0-0}{1}: [radius] [24732]: (info): RADIUS: authenticator e5 d1 b7 4d 8b e9 d5
06 - 14 b9 8d b6 8c 29 93 94
2022/06/06 23:24:03.268945 {smd_R0-0}{1}: [radius] [24732]: (info): RADIUS: Vendor, Cisco [26] 211
2022/06/06 23:24:03.268954 {smd_R0-0}{1}: [radius] [24732]: (info): RADIUS: Cisco
AVpair [1] 205 "cts-pac-opaque="
2022/06/06 23:24:03.268960 {smd_R0-0}{1}: [radius] [24732]: (info): RADIUS: Vendor, Cisco [26] 36
2022/06/06 23:24:03.268966 {smd_R0-0}{1}: [radius] [24732]: (info): RADIUS: Cisco
AVpair [1] 30 "dc-profile-name=Cisco-Device"
```

Radius Debugs for dot1x / mab are in
Session Manager trace

Platform Troubleshooting



Looking at the Kernel CPU information

- Kernel CPU utilization can be seen per switch/supervisor module

```
9300_54#show proc cpu platform sorted location switch active R0
```

```
CPU utilization for five seconds: 12%, one minute: 9%, five minutes: 5%
```

```
Core 0: CPU utilization for five seconds: 14%, one minute: 10%, five minutes: 5%
```

```
Core 7: CPU utilization for five seconds: 14%, one minute: 10%, five minutes: 6%
```

Pid	PPid	5Sec	1Min	5Min	Status	Size	Name
14854	13498	51%	38%	22%	S	951436	linux_iosd-imag
15322	15160	42%	30%	15%	S	313236	fed main event
2960	2	8%	5%	2%	R	0	lsmpi-xmit
2961	2	5%	4%	2%	S	0	lsmpi-rx

CPU higher then expected

- linux_iosd-image process as seen on kernel is IOSd process
- Lsmpi processes are part of Linux Shared Memory Punt Interface
- Not all processes run on all switches.
- Changing the location will display cpu for standby or member switches

IOSd CPU utilization

ARP input is high

```
9300_54#show proc cpu sort
```

```
CPU utilization for five seconds: 25%/1%; one minute: 22%; five minutes: 11%
```

PID	Runtime(ms)	Invoked	uSecs	5Sec	1Min	5Min	TTY	Process
40	9474304	54714487	173	22.95%	16.12%	8.15%	0	ARP Input
24	95502	12431773	7	0.07%	0.00%	0.00%	0	IPC Mcast Pendin
41	152951	13280277	11	0.07%	0.00%	0.00%	0	ARP Background
53	102409	12729984	8	0.07%	0.00%	0.00%	0	Dynamic ARP Insp
83	4759850	48620416	97	0.07%	0.03%	0.02%	0	IOSD ipc task
121	5602021	50896336	110	0.07%	0.04%	0.05%	0	Crimson config p
136	5112131	792052381	6	0.07%	0.03%	0.02%	0	L2 LISP Punt Pro

- Inside IOSd process many functions and processes run
Examples: BGP, RIP, CEF, ARP, UDLD, CDP, SSH, SNMP, Telnet
- CPU not involved in data plane forwarding
- Control Plane Policing enabled by default to throttle traffic

CPU utilization in IOSd

- CPU history gives an overview of CPU history for last minute, hour and day

```
9300_54#show processor cpu history
```

```
          444      444      443      444      444      444      444      444      44
1189111 18711 1188911 18821111883111 784111188711 1987111189
100
90
50  **      **      **      **      **      **      ***      ***
40  *#*     *#*     *#*     *#*     *#*     *#*     *#*     *#*
30  ##*     ##*     ##*     ##*     ##*     ##*     ##*     ##*
20  ##*     ##*     ##*     ##*     ##*     ##*     ##*     ##*
10  ###     ###     ###     ###     ###     ###     ###     ###
0...5...1...1...2...2...3...3...4...4...5...5...6
      0    5    0    5    0    5    0    5    0    5    0
CPU% per minute (last 60 minutes)
* = maximum CPU%  # = average
```

CPU spiking regularly

- Pattern of Interval and length of CPU spikes can point to trigger.
- Average high CPU more cause for concern then short spikes

Catching CPU spikes in IOSd

- CPU spikes often occur when not actively watching the switch.
- IOS allows CPU monitoring:
`process cpu threshold type total rising <%> interval <s> falling <%> interval <s>`
- When threshold are exceeded syslog is generated
- Syslog can be used by Embedded Event Manager to automate collection of data during time of failure

```
%SYS-1-CPURISINGTHRESHOLD: Threshold: Total CPU Utilization(Total/Intr): 48%/15%, Top 3  
processses(Pid/Util): 40/32%, 94/0%, 105/0%
```

```
9300_54#show proc cpu | inc 40 | 94 | 105 | PID
```

Top 3 processes

PID	Runtime(ms)	Invoked	uSecs	5Sec	1Min	5Min	TTY	Process
40	19640520	69264320	283	22.48%	20.61%	10.14%	0	ARP Input
94	652410	6428606	101	0.07%	0.00%	0.00%	0	PuntInject Keepa
105	3477489	6127722	567	0.00%	0.02%	0.02%	0	Crimson flush tr

Determining Cause of Inband traffic

```
9300_54#show platform software fed switch active punt rates interfaces
```

Packets per second averaged over 10 seconds, 1 min and 5 mins

Interface Name	IF_ID	Recv 10s	Recv 1min	Recv 5min	Drop 10s	Drop 1min	Drop 5min
GigabitEthernet1/0/11	0x00000013	5999	1707	2074	0	0	0
Vlan192	0x00000036	5999	1707	2074	0	0	0

Active interfaces
sending to cpu

```
9300_54#
```

Per Queue Statistics

```
9300_54#show platform software fed switch active punt cpuq rates
```

Packets per second averaged over 10 seconds, 1 min and 5 mins

Q no	Queue Name	Rx 10s	Rx 1min	Rx 5min	Drop 10s	Drop 1min	Drop 5min
0	CPU_Q_DOT1X_AUTH	0	0	0	0	0	0
1	CPU_Q_L2_CONTROL	0	0	0	0	0	0
5	CPU_Q_FORUS_ADDR_RESOLUTION	6000	1874	2414	0	0	0

Queue
information

Drops typically seen on Port
Asic not in FED punt path

Using Embedded Packet Capture

- EPC provides packet capture on interface or control plane level
- Data capture in hardware, traffic copied to EPC process.
- Packet capture rate limited

```
9300_54#monitor capture CL interface GigabitEthernet 1/0/11 in
9300_54#monitor capture CL match any
9300_54#monitor capture CL start
9300_54#monitor capture CL stop
9300_54#monitor capture CL export location flash:cl.pcap
9300_54#sh monitor capture file flash:cl.pcap display-filter arp
Starting the packet display ..... Press Ctrl + Shift + 6 to exit
 352  57.9 00:00:ca:fe:ca:fe ff:ff:ff:ff:ff:ff ARP 60 Who has 192.168.5.1? Tell 192.168.2.8
 353  57.9 00:00:ca:fe:ca:fe ff:ff:ff:ff:ff:ff ARP 60 Who has 192.168.5.2? Tell 192.168.2.8
 354  57.9 00:00:ca:fe:ca:fe ff:ff:ff:ff:ff:ff ARP 60 Who has 192.168.5.3? Tell 192.168.2.8
 355  57.9 00:00:ca:fe:ca:fe ff:ff:ff:ff:ff:ff ARP 60 Who has 192.168.5.4? Tell 192.168.2.8
 356  57.9 00:00:ca:fe:ca:fe ff:ff:ff:ff:ff:ff ARP 60 Who has 192.168.5.5? Tell 192.168.2.8
```

Where and what to capture

Export capture to file

Displays capture from flash or buffer

Determine packets hitting a specific CPU queue

- FED allows limited packet captures done in punt and inject path
- Allows filter to be setup to collect specific data.
- Use filter “fed.queue==<queue>” for collecting specific queue
- Detail and brief packet display. Detail contains internal forwarding information

```
9300_54#debug platform software fed switch active punt packet-capture set-filter "fed.queue == 5"
9300_54#debug platform software fed switch active punt packet-capture [start|stop]
9300_54#show platform software fed switch active punt packet-capture brief
Punt packet capturing: disabled. Buffer wrapping: disabled
Total captured so far: 4096 packets. Capture capacity : 4096 packets
Capture filter : "fed.queue == 5"
----- Punt Packet Number: 3, Timestamp: 2023/05/31 13:14:45.888 -----
interface : physical: GigabitEthernet1/0/11[if-id: 0x00000013], pal: Vlan192 [if-id: 0x00000036]
metadata   : cause: 7 [ARP request or response], sub-cause: 1, q-no: 5, linktype: MCP_LINK_TYPE_IP
ether hdr  : dest mac: ffff.ffff.ffff, src mac: 0000.cafe.cafe
ether hdr  : ethertype: 0x0806 (ARP)
```

Metadata indicates
why it was punted

Control Plane Policing HW stats

- Control plane policing drops frames on port ASIC's to protect CPU
- Do not modify defaults without clear understanding of the impact

```
9300_1#show plat hardware fed switch active qos queue stats internal cpu policer
```

CPU Queue Statistics

QId	PlcIdx	Queue Name	Enabled	(default) Rate	(set) Rate	Queue Drop (Bytes)	Queue Drop (Frames)
4	2	Routing Control	Yes	5400	5400	0	0
5	14	Forus Address resolution	Yes	4000	65000	2212622868	345698644

CPU Queue Policer Statistics

Policer Index	Policer Accept Bytes	Policer Accept Frames	Policer Drop Bytes	Policer Drop Frames
12	0	0	0	0
13	9562218	122704	453808601	7079824

CPP Classes to queue map

PlcIdx	CPP Class	Queues
0	system-cpp-police-data	ICMP GEN/ BROADCAST/ ICMP Redirect/
14	system-cpp-police-forus	Forus Address resolution/ Forus traffic/

Control Plane
policing drops

Rate changed from
Default rate

Platform Memory

9600_SVL#sh processes memory platform sorted location switch 1 R0 brief
System memory: 15993924K total, 6935004K used, 9058920K free,
Lowest: 9025140K

Pid	Text	Data	Stack	Dynamic	RSS	Name
5373	256860	1488024	136	488	1488024	linux_iosd-imag
31175	203	885528	136	132984	885528	fed main event
26748	7236	295848	136	3808	295848	fman_rp
15847	8158	263112	136	4692	263112	fman_fp_image
27259	450	235088	136	4720	235088	dbm
28641	2583	228456	164	101856	228456	confd
30179	87	208392	136	7832	208392	pubd
25903	336	193308	136	388	193308	ndbmand
4272	790	177200	136	33732	177200	smand
18087	178	176024	136	13104	176024	sessmgrd

Location

IOSd

FED

Forwarding manager

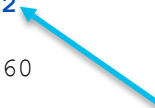
SMD

- Kernel memory utilization is available per switch
- linux_iosd-image process is IOSd
- Resident Set Size(RSS), memory occupied by each Process

Per process memory allocation

```
9300_54#show platform software memory smd switch active R0 brief
```

module	allocated	requested	allocs	frees
-----	-----	-----	-----	-----
Summary	10651060	10392132	461338	445155
OBJ_SMTEST_EVENTLOG	6528072	6496040	2002	0
AAA_CHUNK_ATTR_SUBLIST	1085842	1085762	5	0
OBJ_DC	625435	429323	12260	3
chunk	343897	342425	97	5
OBJ_RCLSRV_SEND_BUF_CH	263045	262981	4	
smd	213359	213295	4	
AAA_ATTR_LIST_HANDLE_I	133193	133145	3	
process	133172	133124	3	0
eventutil	120685	119757	2074	2016
OBJ_EPM_CACHE_HASH_ELE	97688	97128	44	9
AAA_CHUNK_ATTR_HEADER	78225	78145	5	0
OBJ_EAP_ALLOC_HDL	65552	65536	1	0
AAA_MLIST_ID	65552	65536	1	0



Event log, increases over time. Expected till log is full

- Detailed memory usage per process provides insight into memory usage per process.

IOSd Memory

```
9600_SVL#sh processes memory sorted holding
```

```
Processor Pool Total: 2959189564 Used: 687714160 Free: 2271475404
```

```
reserve P Pool Total: 102404 Used: 88 Free: 102316
```

```
lsmpi_io Pool Total: 6295128 Used: 6294296 Free: 832
```

Receive process,
should not be high

PID	TTY	Allocated	Freed	Holding	Getbufs	Retbufs	Process
0	0	488208064	77059624	381785696	0	0	*Init*
238	0	137113496	1017784	136266568	0	0	IP ARP Adjacency
82	0	50195904	2966104	27577736	0	0	IOSD ipc task
565	0	720962728	696333696	24623088	0	0	DHCPD Receive
4	0	42371936	17822928	22358600	0	0	RF Slave Main Th

- IOSd runs as a process , but does still provides some kernel features like memory management for all processes running inside IOSd
- Processor Pool: Pool for Processes on IOSd
- lsmpi_io : Linux Shared Memory Punt Interface memory, IO buffers

IOSd Memory Allocator

```
9600_SVL#sh processes memory 565  
Tracekey : 1#01327ced3d92c0f78c7959f154fa0114
```

```
Process ID: 565
```

```
Process Name: DHCPD Receive
```

```
Total Memory Held: 24623088 bytes
```

```
Processor memory Holding = 24623088 bytes
```

```
size = 10549344, count = 321, pc = :5644D0D8C000+934D623
```

```
size = 10450752, count = 318, pc = :5644D0D8C000+548EF85
```

```
size = 1478880, count = 45, pc = :5644D0D8C000+861423D
```

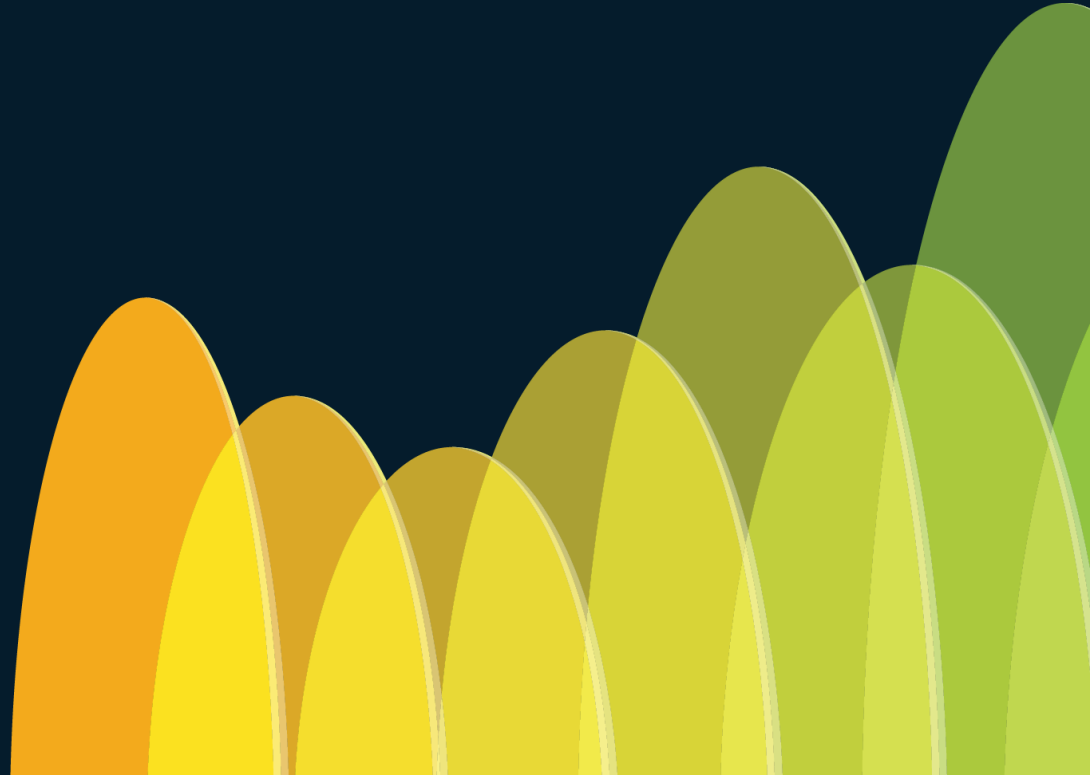
```
size = 1032040, count = 120, pc = :5644D0D8C000+8615EB3
```

```
size = 985920, count = 30, pc = :5644D0D8C000+8608243
```

More memory allocated,
didn't get freed-> bug

- Allocator PC indicator of what allocated memory
- Not all increased in memory are leaks, memory might get released again or functions might just need more memory
- Monitor over time and multiple boxes to determine possible pattern

Packet Drops



Checking interface status

Err-disabled => check
show interface status errdisable
or **show log** for cause

Vlan column shows routed(L3) ,
trunk or base vlan

Half Duplex 1Gb/s, possible
duplex mismatch?

```
9500_1#show interfaces status | ex notc
```

Port	Name	Status	Vlan	Duplex	Speed	Type
Twe1/0/7		connected	routed	a-full	a-1000	10/100/1000BaseTX SFP
Twe1/0/12		err-disabled	1	full	10G	SFP-10GBase-CU1M
Twe1/0/15		connected	routed	full	10G	SFP-10GBase-CU3M
Twe1/0/20		connected	4094	full	10G	SFP-10GBase-CU1M
Twe1/0/21		connected	routed	a-half	a-1000	10/100/1000BaseTX SFP
Twe1/0/22		disabled	1	auto	auto	10/100/1000BaseTX SFP
Hu1/0/26		connected	trunk	full	40G	QSFP 40G CU3M
Hu1/0/27		connected	4094	full	100G	QSFP 100G CU2M

Show interface status command gives a quick overview of interface status

Ethernet Statistics

Switch#**show controllers ethernet-controller gi 5/0/48**

Transmit		GigabitEthernet5/0/48 Receive	
1562496684	Total bytes	2968958225	Total bytes
5032561	Unicast frames	6004241	Unicast frames
700808558	Unicast bytes	1807110661	Unicast bytes
1269484	Multicast frames	2789759	Multicast frames
861688062	Multicast bytes	1161847500	Multicast bytes
1	Broadcast frames	1	Broadcast frames
0	Cos 0 Pause frames	0	Cos 0 Pause frames
1236978	Minimum size frames	871517	Minimum size frames
1892419	65 to 127 byte frames	2181611	65 to 127 byte frames
1941967	128 to 255 byte frames	2712229	128 to 255 byte frames
685594	256 to 511 byte frames	1260418	256 to 511 byte frames
20261	512 to 1023 byte frames	900135	512 to 1023 byte frames
524827	1024 to 1518 byte frames	868091	1024 to 1518 byte frames
0	8192 to 16383 byte frames	0	8192 to 16383 byte frames
0	16384 to 32767 byte frame	0	16384 to 32767 byte frame
0	> 32768 byte frames	0	> 32768 byte frames
0	Late collision frames	0	SymbolErr frames
0	Excess Defer frames	0	Collision fragments
0	Good (1 coll) frames	0	ValidUnderSize frames
0	Good (>1 coll) frames	0	InvalidOverSize frames
0	Deferred frames	0	ValidOverSize frames

LAST UPDATE 361 msecs AGO

Ethernet controller statistics give more detailed port statistics

Frame size distribution

Error statistics

Verifying link utilization

Input/output rates show average over 5 minutes (default). Traffic might be bursty in nature

```
Switch#show interfaces | inc line|rate
```

```
Vlan1 is up, line protocol is up , Autostate Enabled
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
GigabitEthernet0/0 is administratively down, line protocol is down
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
GigabitEthernet1/0/11 is up, line protocol is up (connected)
  30 seconds input rate 575000 bits/sec, 958 packets/sec
  30 seconds output rate 126975000 bits/sec, 10473 packets/sec
GigabitEthernet1/0/12 is down, line protocol is down (notconnect)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
GigabitEthernet1/0/13 is down, line protocol is down (notconnect)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
```

load-interval has a range of 30-600 seconds

```
Switch#show controllers utilization
```

Port	Receive Utilization	Transmit Utilization
Gi1/0/1	0	0
Gi1/0/11	0	92
.		
Gi1/0/24	0	0
Te1/1/1	0	0
Te1/1/4	9	0
Total Ports : 33		

Bandwidth in %
Current load

```
Total Ports Receive Bandwidth Percentage Utilization : 0
Total Ports Transmit Bandwidth Percentage Utilization : 0
Average Switch Percentage Utilization : 0
```

Tail Drops

```
9300_54#show interfaces gigabitEthernet 1/0/11 | inc output drops
  Input queue: 0/2000/0/0 (size/max/drops/flushes); Total output drops: 1277
9300_54#show controllers ethernet-controller gig 1/0/11 | inc Excess Def
      1277 Excess Defer frames                0 Collision fragments
SNMP:
SNMPv2-SMI::enterprises.9.2.2.1.1.27.8 = INTEGER: 1277
```

- Tail drops occur when no buffers are available
- Buffering occurs when traffic comes in faster than being transmitted
- Buffer allocation per class can be changed inside service-policy. <- Preferred
- Global multiplier to increase buffers up to 1200%
qos queue-softmax-multiplier <percentage>
Use with caution, only adjust when needed, drops are drops but not always a problem
- Ensure proper classification of traffic and apply correct policies, not all traffic is equal

QoS Hardware configuration

```
9300_45#show platform hard fed switch active qos queue config interface gi 1/0/11
Asic:0 Core:1 DATA Port:11 GPN:1 LinkSpeed:0x1
```

DTS			Hardmax			Softmax			PortSMin		Glb1SMin		PortStEnd	
0	1	5	200	12	3200	5	500	0	0	6	9600			
1	1	4	0	13	4800	5	750	2	300	6	9600			
Priority			Shaped/shared			weight		shaping_step		sharpedWeight				
0	0		Shared			50		0		0				
1	0		Shared			75		0		0				
Port			Port			Port		Port		Port				
Priority			Shaped/shared			weight		shaping_step						
2			Shaped			254		255						
Weight0			Max_Th0	Min_Th0		Weight1	Max_Th1	Min_Th1		Weight2	Max_Th2	Min_Th2		
0	0		2709	0	0	3028	0	0	0	3400	0	0		
1	0		3825	0	0	4275	0	0	0	4800	0	0		

Hardmax : Reserved
Softmax : Global pool

Queue mode:
shaped or shared
Queue limit for shaping
Step/weight * speed

Drop thresholds for
queue/threshold
In Buffers (256 byte)

- QoS configured using service policies on interfaces
- Applied service-policy translated into Hardware settings that match HW capabilities

QoS hardware statistics

```
9300_45#show platform hardware fed switch active qos queue stats interface gigabitEthernet 1/0/11
```

AQM Global counters

```
GlobalHardLimit: 7976 | GlobalHardBufCount: 0
GlobalSoftLimit: 11872 | GlobalSoftBufCount: 0
```

Highest number of buffers in use by interface since last issuing command

```
High Watermark Soft Buffers: 429 <--- clear on read
```

Asic:0 Core:1 DATA Port:10 Hardware Enqueue Counters

Q	Buffers (Count)	Enqueue-TH0 (Bytes)	Enqueue-TH1 (Bytes)	Enqueue-TH2 (Bytes)	Qpolicer (Bytes)
0	200	0	385820	46085690	0
1	0	0	0	0	0

Current buffer count per queue

Asic:0 Core:1 DATA Port:0 Hardware Drop Counters

Q	Drop-TH0 (Bytes)	Drop-TH1 (Bytes)	Drop-TH2 (Bytes)	SBufDrop (Bytes)	QebDrop (Bytes)	QpolicerDrop (Bytes)
0	0	0	412312	0	0	0
1	0	0	0	0	0	0

- At Asic level there are 8 Queues/3 Thresholds
- Enqueue/Drop Counters available per queue/per threshold
- Buffers (count) show currently assigned buffers to Queue (256 bytes)
- To enable high water mark counter monitoring.

set platform hardware fed switch active qos port-monitor interface <if>

Doppler ASIC packet forwarding drop counters

```
9300_45#show platform hardware fed switch active fwd-asic drops exceptions
****EXCEPTION STATS ASIC INSTANCE 0 (asic/core 0/0)****
=====
Asic/core | NAME | prev | current | delta
=====
0 0 NO_EXCEPTION 35364016 35364108 92
0 0 IPV4_CHECKSUM_ERROR 0 0 0
0 0 ROUTED_AND_IP_OPTIONS_EXCEPTION 2 2 0
0 0 CTS_FILTERED_EXCEPTION 0 0 0
0 0 AUTH_DRIVEN_DROP 0 0 0
0 0 PKT_DROP_COUNT 0 3732 3732
0 0 ALLOW_DOT1Q_EXCEPTION_COUNT 0 0 0
0 0 ALLOW_PRIORITY_TAGGED_EXCEPTION_COUNT 0 0 0
0 0 IGR_EXCEPTION_L5_ERROR 0 363 363
0 0 IP_UNICAST_TTL_REACHED_ZERO 0 0 0
0 0 MISC_FATAL_ERROR 0 0 0
```

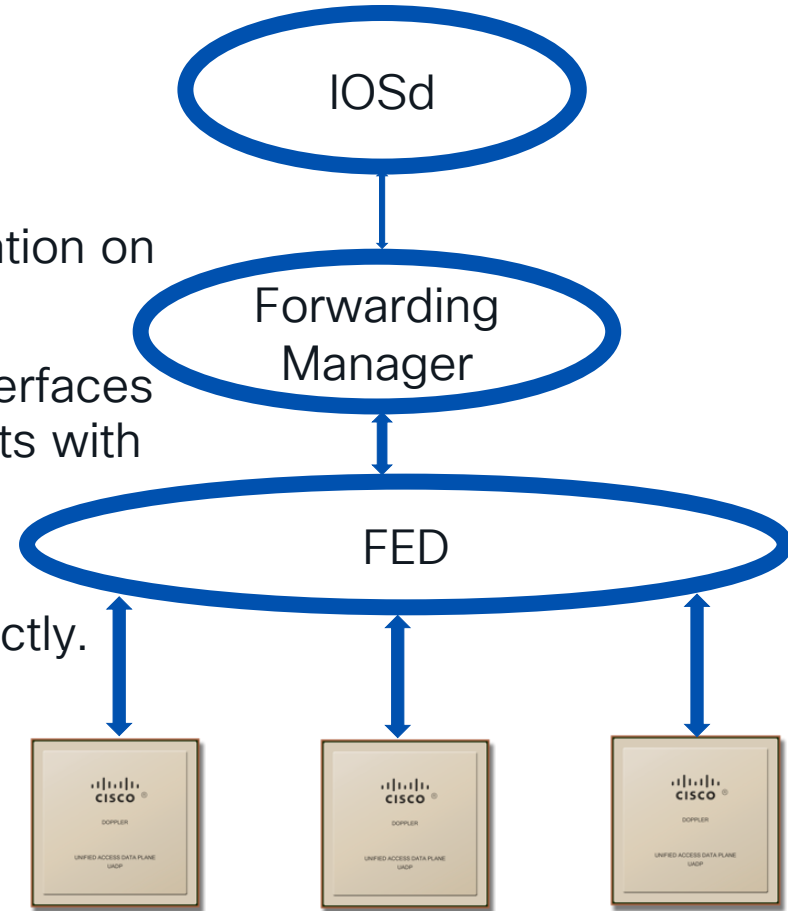
- Every packet passing through Port Asic gets parsed by the port-asic's on both receive (IGR) and transmit (EGR) side
- Exception drops are counted per Asic, not per port.

Forwarding Verification

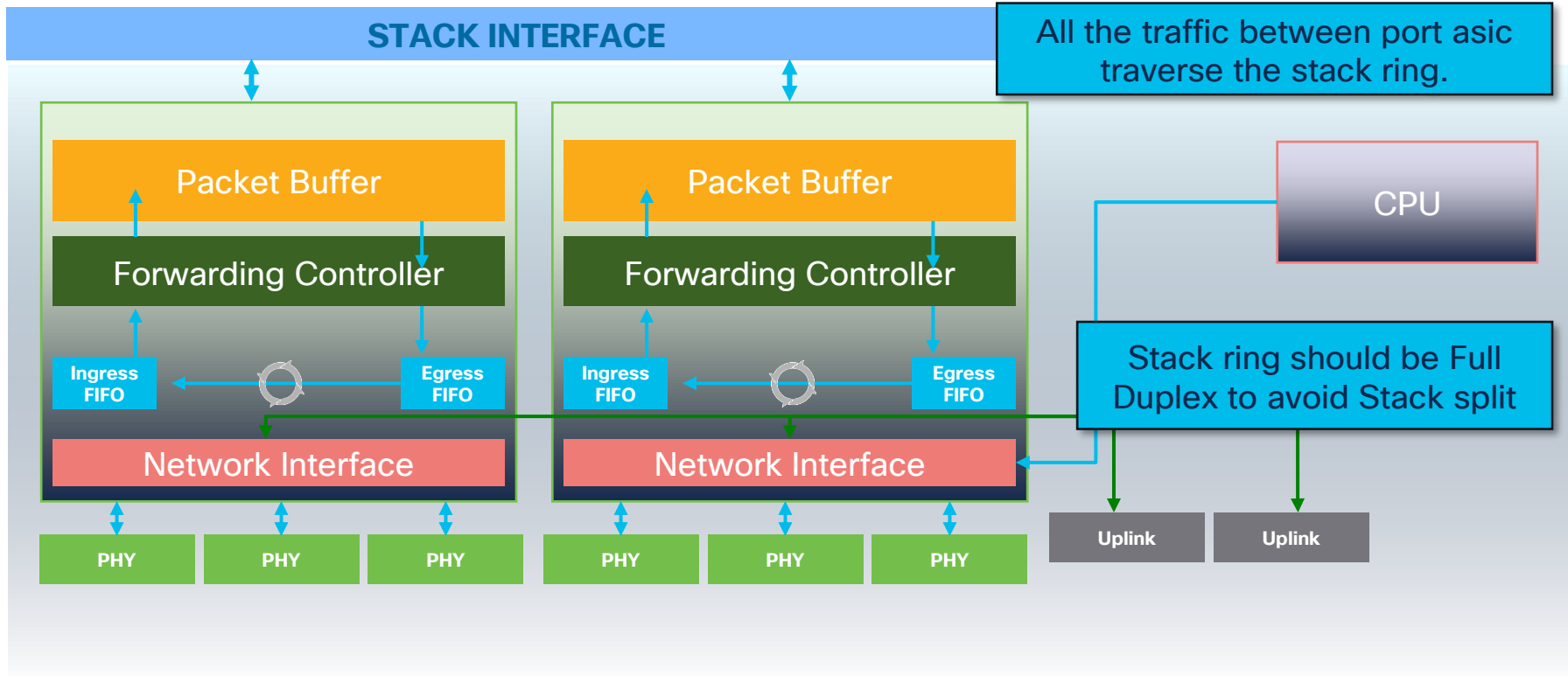


Troubleshooting Forwarding

- UADP/Silicon One responsible for dataplane forwarding.
- IOS-XE uses FMAN to forward related information on a platform-independent level.
- FED process is platform-dependant Layer, interfaces with Forwarding Manager process and interacts with UADP and Silicon One.
- Traffic between port asics uses ring or interconnect while Local traffic forwarded directly.
- Every Stack member/Supervisor module runs a FED process.



Catalyst 9200/9300 Stackable Switches



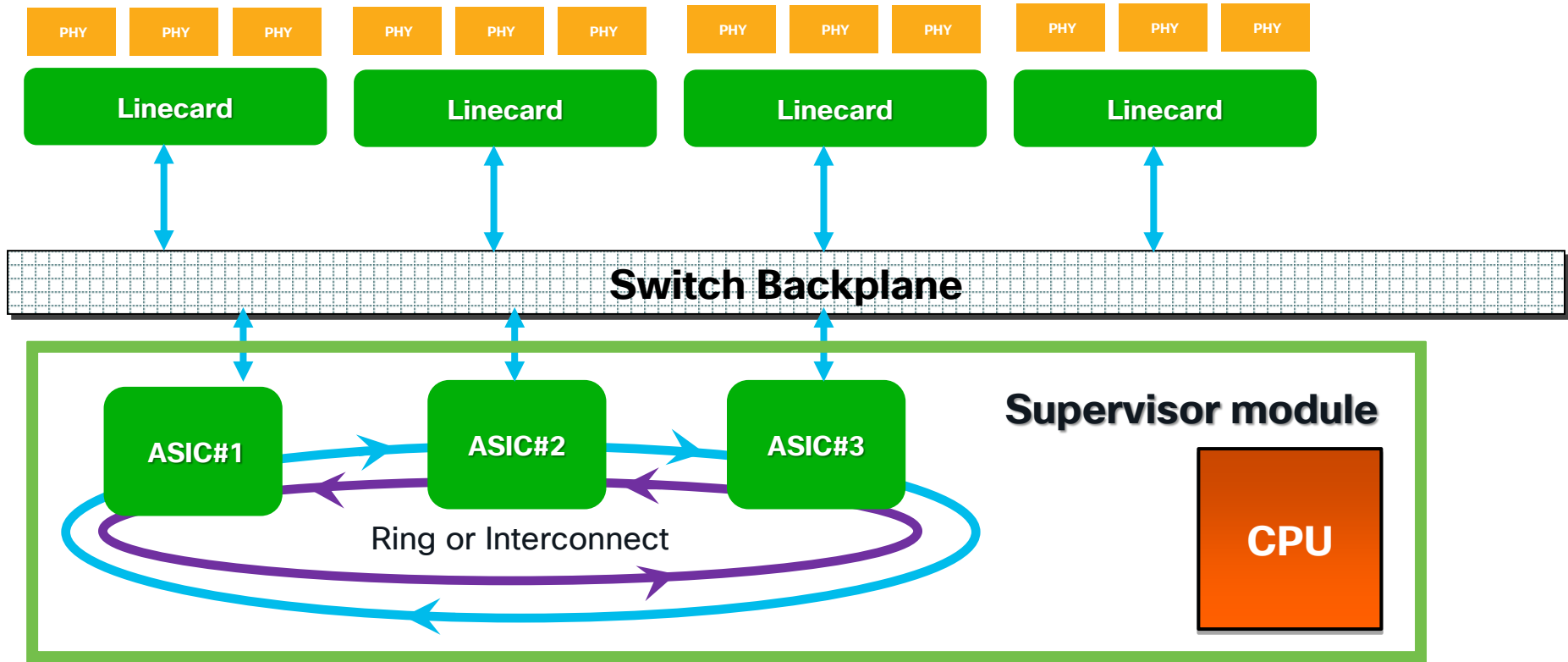
Switch Stacks

```
9300_10#sh switch stack-ports summary
```

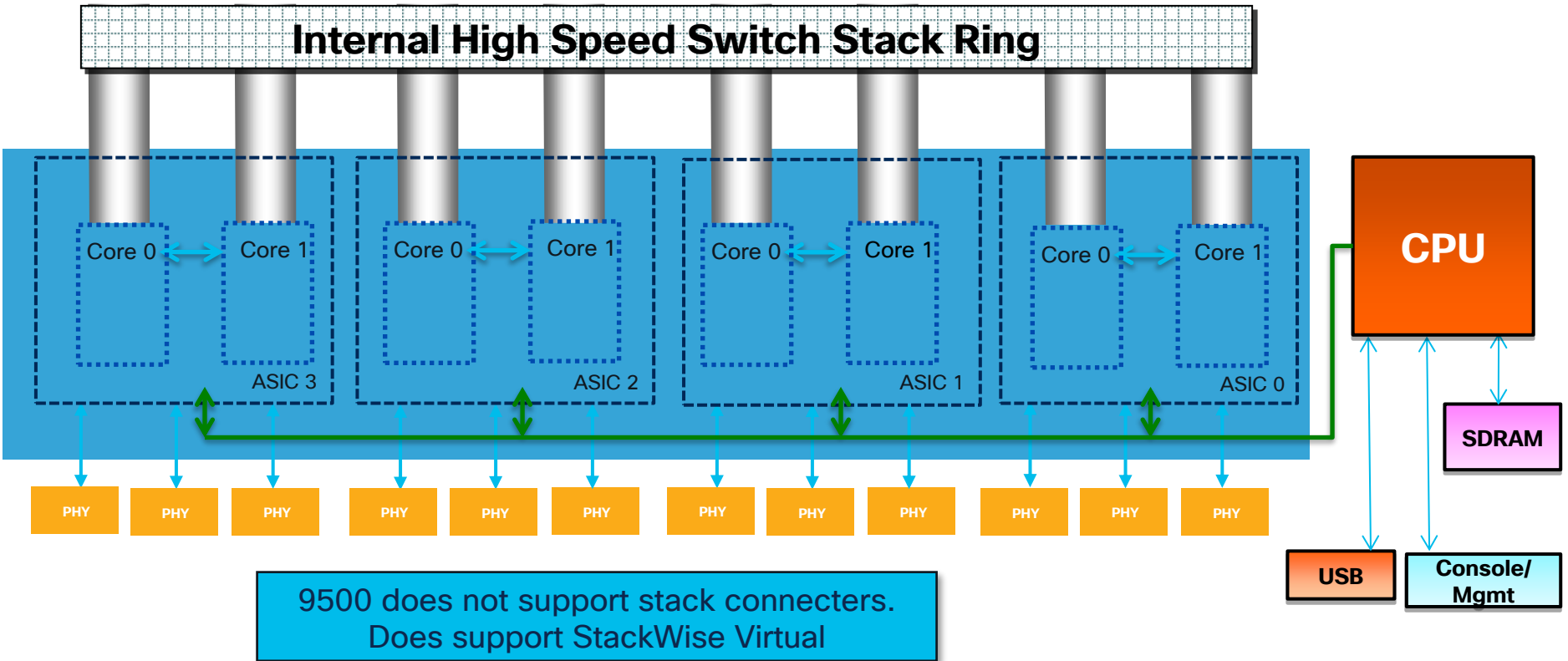
Sw#/Port#	Port Status	Neighbor/Port	Cable Length	Link OK	Link Active	Sync OK	#LinkOK	In Loopback
1/1	OK	4/2	100cm	Yes	Yes	Yes	1	No
1/2	OK	2/1	50cm	Yes	Yes	Yes	1	No
2/1	OK	1/2	50cm	Yes	Yes	Yes	2	No
2/2	OK	3/1	50cm	Yes	Yes	Yes	1	No
3/1	OK	2/2	50cm	Yes	Yes	Yes	1	No
3/2	OK	4/1	50cm	Yes	Yes	Yes	2	No
4/1	OK	3/2	50cm	Yes	Yes	Yes	1	No
4/2	OK	1/1	100cm	Yes	Yes	Yes	2	No

- Ensure all stack ports are connected and Link Ok
- Ring Wrapping halves the stack bandwidth and reduces redundancy.
- High Number of #linkOK might indicate bad connection
- Cable Length and Neighbor/Port should be detected
- Detailed information available with show switch stack-ports detail
- Collect show tech stack in case of stacking issues

Catalyst 9400/9600 Chassis based



Catalyst 9500 Fixed Switches



Interface Internal Mappings

Interface to ASIC mapping important to understand data flows

```
9300_1#show platform software fed switch active ifm mappings
```

Interface	IF_ID	Inst	Asic	Core	Port	SubPort	Mac	Cntx	LPN	GPN	Type	Active
GigabitEthernet1/0/1	0x8	1	0	1	0	0	26	6	1	1	NIF	Y
GigabitEthernet1/0/2	0x4c	1	0	1	1	0	6	7	2	2	NIF	Y
GigabitEthernet1/0/3	0x4d	1	0	1	2	0	28	8	3	3	NIF	Y

Internally used interface addressing:

- LPN : Local Port Number
- GPN : Global Port Number
- IF_ID : Interface Identification, used for many fed CLI
- Type : Type of interface, NIF = Network Interface
- Inst : Instance : ASIC + Core
- Port : Asic Ports
- Active : Is Interface Active

Troubleshoot between ports on same ASIC when possible

IFM Mappings logical interface

```
9600_SVL#show platform software fed switch active ifm interfaces vlan
Interface                               IF_ID                               State
```

```
-----
Vlan1                                  0x0005ffff                          READY
Vlan123                               0x00420012                          READY
Vlan900                               0x00420010                          READY
```

```
9600_SVL#show platform software fed switch active ifm interfaces svi
Interface                               IF_ID                               State
```

```
-----
Vlan123                               0x0000013b                          READY
```

```
9600_SVL#show platform software fed switch active ifm interfaces tunnel
Interface                               IF_ID                               State
```

```
-----
Tunnel1                               0x00000143                          READY
```

- An unique IF_ID is assigned to every logical and physical.
- Every type of interface has its unique IF_ID.
Ex: Layer 2 Vlan has a different IF_ID as the Layer 3 SVI

IF_ID more detail

• 9600_SVL#**show platform software fed switch active ifm if-id 0x138**

```
Interface IF_ID      : 0x00000000000000138
Interface Name       : FortyGigabitEthernet2/1/0/1
Interface Block Pointer : 0x7f33b89fd918
Interface Block State  : READY
Interface State       : Enabled
Interface Status      : ADD, UPD
Interface Type        : ETHER
Port Type             : ROUTE PORT
Port Location        : LOCAL
Slot                 : 15
Unit                 : 0
Slot Unit            : 1
SNMP IF Index        : 148
GPN                  : 577
Port Handle          : 0xb7000104
IPv4 MTU             : 9100
IPv6 MTU             : 0
IPv4 VRF ID         : 0x0
IPv6 VRF ID         : 0x0
Protocol flags       : 0x0007 [ ipv4 ipv6 pim_ipv4 ]
```

Specifying the IF-ID gives
verbose information

Information available varies
depending on Interface Type

One VRF ID per VRF, VRF ID 0 is
Global Routing Table

Layer 2 Forwarding. Verifying HW STP state

- Show spanning tree gives IOSd view of Spanning Tree
- Hardware forwarding states can be checked *per switch* on FED layer
- Outputs will show what interface are in forwarding state and if traffic is tagged or untagged
- Flood list indicates what Ports will receive flooded traffic on this switch

```
9300_54#sh platform hardware fed switch active vlan 192 egress
VLAN STP State in hardware
vlan id is:: 192
Interfaces in forwarding state: : Gi1/0/11(Untagged) , Te1/1/1(Untagged) , Te1/1/4(Tagged)

9300_54#sh platform hardware fed switch active vlan 192 ingress
VLAN STP State in hardware
vlan id is:: 192
Interfaces in forwarding state: : Gi1/0/11(Untagged) , Te1/1/1(Untagged) , Te1/1/4(Tagged)
flood list: : Gi1/0/11, Te1/1/1, Te1/1/4
```

Layer 2 Forwarding, IOSd mac address tables

```
9300_54#show mac address vlan 192
      Mac Address Table
-----
Vlan    Mac Address      Type        Ports
-----
  192    0000.cafe.cafe    DYNAMIC     Gi1/0/11
  192    0013.c3c1.0d89    DYNAMIC     Gi1/0/11
  192    0050.5693.5e70    DYNAMIC     Te1/1/4
  192    a0f8.4910.2dd3    STATIC      Vl192
  192    d0ec.35c9.d353    DYNAMIC     Te1/1/1
Total Mac Addresses for this criterion: 5
```

- **Show mac address** table contains a system wide mac table on IOSd
- Types can be static, dynamic, drop
- Mac Address of SVI interfaces also showing in mac address table

FED MATM Mac Address Table

```
9300_1#sh platform software fed switch 1 matm macTable vlan 192
```

VLAN	MAC	Type	Seq#	EC_Bi	Flags	machandle	siHandle	diHandle	*a_time	*e_time	ports
192	0000.cafe.cafe	0x1	71718	0	0	0x7f2348	0x7f2198	0x7f2478	300	112	GigabitEthernet1/0/11
192	d0ec.35c9.d353	0x1	71719	0	0	0x7f2328	0x7f2148	0x7f2588	300	14	TenGigabitEthernet1/1/1
192	0013.c3c1.0d89	0x1	71720	0	0	0x7f23d8	0x7f2378	0x7f4278	300	11	GigabitEthernet1/0/11
192	a0f8.4910.2dd3	0x8002	0	0	64	0x7f2308	0x7f21c8	0x5234	0	0	Vlan192
192	0050.5693.5e70	0x1	12772	0	0	0x7f23d8	0x7f2358	0x7f23e8	300	13	TenGigabitEthernet1/1/4

*a_time=aging_time(secs) *e_time=total_elapsed_time(secs)

Type:

MAT_DYNAMIC_ADDR	0x1	MAT_STATIC_ADDR	0x2	MAT_CPU_ADDR	0x4	MAT_DISCARD_ADDR	0x8
MAT_ALL_VLANS	0x10	MAT_NO_FORWARD	0x20	MAT_IPMULT_ADDR	0x40	MAT_RESYNC	0x80
MAT_DO_NOT_AGE	0x100	MAT_SECURE_ADDR	0x200	MAT_NO_PORT	0x400	MAT_DROP_ADDR	0x800
MAT_DUP_ADDR	0x1000	MAT_NULL_DESTINATION	0x2000	MAT_DOT1X_ADDR	0x4000	MAT_ROUTER_ADDR	0x8000
MAT_WIRELESS_ADDR	0x10000	MAT_SECURE_CFG_ADDR	0x20000	MAT_OPQ_DATA_PRESENT	0x40000	MAT_WIRED_TUNNEL_ADDR	0x80000
MAT_DLR_ADDR	0x100000	MAT_MRP_ADDR	0x200000	MAT_MSRR_ADDR	0x400000	MAT_LISP_LOCAL_ADDR	0x800000
MAT_LISP_REMOTE_ADDR	0x1000000	MAT_VPLS_ADDR	0x2000000				

- Every FED maintains its own Mac address table.
Switch learning the mac address owns it.
- Type Field indicates the type of mac address using a bitmap
- Sequence number of an entry changing would indicate relearning

Layer 3 ,Check The Routing Protocols

```
9300_54#sh ip route 10.39.102.0
Routing entry for 10.39.102.0/24
  Known via "eigrp 1", distance 90, metric 13656, precedence routine (0), type
internal
  Redistributing via eigrp 1
  Last update from 192.168.2.254 on Vlan192, 6w0d ago
  Routing Descriptor Blocks:
    * 192.168.2.254, from 192.168.2.254, 6w0d ago, via Vlan192
      Route metric is 13656, traffic share count is 1
      Total delay is 433 microseconds, minimum bandwidth is 938072 Kbit
      Reliability 246/255, minimum MTU 1500 bytes
      Loading 107/255, Hops 9

9300_54#sh ip arp 192.168.2.254
Protocol    Address                  Age (min)    Hardware Addr   Type   Interface
Internet    192.168.2.254            106          0050.5693.5e70  ARPA   Vlan192
```

- Troubleshoot from the top, first verify routing and ARP tables.
- Verify path is showing correct. If not, look at the Routing protocols

Cisco Express Forwarding, the FIB

```
9300_54#show ip cef 10.39.102.0/24 internal
10.39.102.0/24, epoch 3, RIB[I], refcnt 6, per-destination sharing
sources: RIB
  Vlan192(54): 192.168.2.254
    path list 7FF78ADE26F0, 20001 locks, per-destination, flags 0x4D [shble, hvsh, rif, hwcn]
      path 7FF78ADE3288, share 1/1, type attached nexthop, for IPv4
        nexthop 192.168.2.254 Vlan192, IP adj out of Vlan192, addr 192.168.2.254 7FF78624DBE8
    output chain:
      IP adj out of Vlan192, addr 192.168.2.254 7FF78624DBE8
```

Internal keyword
gives more detail

```
9300_54#sh adjacency 192.168.2.254 detail
Protocol Interface      Address
IP          Vlan192      192.168.2.254(10011)
                                     sourced in sev-epoch 0
                                     Encap length 14
                                     005056935E70A0F849102DD30800
                                     L2 destination address byte offset 0
                                     L2 destination address byte length 6
                                     Link-type after encap: ip
                                     ARP
```

Next hop
information

Adjacency -> rewrite info

- The CEF table contains the routing tables build from various sources like the RIB , LISP, etc.

Platform CEF tables (RP)

```
9300_54#sh platform software ip switch ac R0 cef prefix 10.39.102.0/24
```

Forwarding Table

Prefix/Len	Next Object	Index
-----	-----	-----
10.39.102.0/24	OBJ_ADJACENCY	0x16

Next Object points
to Adjacency

```
9300_54#sh platform software adjacency switch active R0 index 0x16
```

Number of adjacency objects: 5

Adjacency id: 0x16 (22)

Interface: Vlan192, IF index: 54, Link Type: MCP_LINK_IP

Encap: 0:50:56:93:5e:70:a0:f8:49:10:2d:d3:8:0

Encap Length: 14, Encap Type: MCP_ET_ARPA, MTU: 9100

Fixup: unknown

Fixup_Flags_2: unknown

Nexthop addr: 192.168.2.254

IP FRR MCP_ADJ_IPFRR_NONE 0

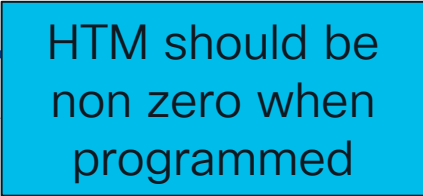
OM handle: 0x3480a3fb88

Adjacency shows
rewrite info

- FMAN forwarding tables should match CEF tables content
- Check Tables for both RP and FP to make sure both populated

FED Routing tables

```
9300_54#sh platform software fed switch active ip route 10.39.102.0/24
vrf      dest                                     htm      flag
---      ---                                     ---      ---
0        10.39.102.0/24                         0x0      0x1
FIB: prefix_hdl:0, mpls_ecr_prefix_hdl:0
===== OCE chain =====
ADJ:objid:22 {link_type:IP ifnum:0x36, adj:0x430025, si:0x7f23ace14  IPv4: 192.168.2.254 }
```



```
9300_54#sh platform software fed switch active ip adj 192.168.2.254
IPV4 Adj entries
dest          if_name      dst_mac          si_hdl          ri_hdl          pd_flags  adj_id
----          -
192.168.2.254  Vlan192      0050.5693.5e70  0x7f23ace14758 0x7f23acdd9a18 0x0      0x16
```

- HTM (Hash Table Manager), points to TCAM entry for route
- SI, Station Index, points to rewrite information and destination
- RI, Rewrite Index, how to rewrite the packet
- Lookups performed on both Ingress and Egress Supervisor/Switch

FED Routing tables

```
9300_54#sh platform software fed switch active ip route summary
```

```
Total number of v4 fib entries = 11348
```

```
Total number succeeded in hardware = 8186
```

```
Mask-Len 0 :- Total-count 1 hw-installed count 1
```

```
Mask-Len 4 :- Total-count 1 hw-installed count 1
```

```
Mask-Len 8 :- Total-count 2 hw-installed count 2
```

```
Mask-Len 24 :- Total-count 10002 hw-installed count 8171
```

```
Mask-Len 30 :- Total-count 1331 hw-installed count 0
```

```
Mask-Len 32 :- Total-count 11 hw-installed count 11
```

- Hardware installed count should match Total-Count
- Routes not installed into Hardware mostly due to resource issues
- TCAM installs routes in regions (masks)

FED Routing tables

```
9300_54#show platform hardware fed switch active fwd-asic resource tcam utilization
```

Codes: EM - Exact_Match, I - Input, O - Output, IO - Input & Output, NA - Not Applicable

CAM Utilization for ASIC [0]

Table	Subtype	Dir	Max	Used	%Used	V4	V6	MPLS	Other
Mac Address Table	EM	I	32768	97	0.30%	0	0	0	97
Mac Address Table	TCAM	I	1024	21	2.05%	0	0	0	21
L3 Multicast	EM	I	8192	0	0.00%	0	0	0	0
L3 Multicast	TCAM	I	512	9	1.76%	3	6	0	0
L2 Multicast	EM	I	8192	0	0.00%	0	0	0	0
L2 Multicast	TCAM	I	512	11	2.15%	3	8	0	0
IP Route Table	EM	I	24576	12	0.05%	11	0	1	0
IP Route Table	TCAM	I	8192	8190	99.98%	8177	10	2	1
QOS ACL	TCAM	IO	5120	181	3.54%	52	86	0	43
Security ACL	TCAM	IO	5120	129	2.52%	26	58	0	45
Netflow ACL	TCAM	I	256	42	16.41%	20	20	0	2

- TCAM Full situation can lead to performance issues
- Choose suitable **SDM template** to optimize TCAM allocation
- TCAM utilization can vary per ASIC
- Check syslog for error messages

Verifying forwarding through the switch

Run show platform hardware fed forward

```
9400_1#sh monitor capture file flash:capwap.pcap packet-number 11
Starting the packet display ..... Press Ctrl + Shift + 6 to exit
0.608962 172.20.26.6 -> 10.48.13.171 DTLSv1.2 123 Application Data
```

Capture and verify capture!

- Using the packet capture the show forward can be executed
- Execute on switch where the packet ingresses pcap needs to be in *local flash*

Execute show command

```
9400_1#sh plat hard fed active forward int gi6/0/19 pcap flash:capwap.pcap num 11 data
Show forward is running in the background. After completion, syslog will be generated.
```

- Once completed a syslog gets generated and results will be available
- Can only run ONE show forward at a time

Confirm execution completed

```
*Jun 4 23:04:43.075: %SHFWD-6-PACKET_TRACE_DONE: F0/0: fed: Packet Trace
Complete: Execute (show platform hardware fed switch <> forward last summary|detail)
```

Show platform hardware fed forward summary

```
9400_1#show platform hardware fed active forward last summary
```

```
Input Packet Details:
```

```
###[ Ethernet ]###
```

```
dst      = 00:00:0c:9f:fa:3d
```

```
src      = 54:8a:ba:7c:4e:d4
```

```
type     = 0x800
```

```
###[ IP ]###
```

```
version  = 4
```

```
ihl      = 5
```

```
tos      = 0xc0
```

```
len      = 109
```

```
ttl     = 64
```

```
proto    = udp
```

```
chksum   = 0x93d9
```

```
src      = 172.20.26.6
```

```
dst      = 10.48.13.171
```

```
###[ UDP ]###
```

```
sport    = 5272
```

```
dport    = 5246
```

```
len      = 89
```

```
chksum   = 0x62e
```

```
Ingress:
```

```
Port
```

```
: GigabitEthernet6/0/19
```

The packet
that is replayed

Sample has
TTL of 64

The ingress
interface

Show platform hardware fed forward summary

Decision:

Destination Index	:	21179	
Rewrite Index	:	27	
Dest Mod Index	:	0	[IGR_FIXED_DMI_NULL_VALUE]
CPU Map Index	:	0	[CMI_NULL]
Forwarding Mode	:	3	[Other or Tunnel]
Replication Bit Map	:		['coreData']
Winner	:		L3FWDIPV4 LOOKUP
Qos Label	:	49	
SGT	:	0	
DGTID	:	1	

Winner, how is
packet rewritten

Egress:

Possible Replication	:	
Port	:	TenGigabitEthernet3/0/2
Output Port Data	:	
Port	:	TenGigabitEthernet3/0/2
Global Port Number	:	302
Local Port Number	:	2
Asic Port Number	:	2
Asic Instance	:	5
Unique RI	:	0
Rewrite Type	:	0 [Unknown]
Mapped Rewrite Type	:	9 [L3_UNICAST_IPV4]

Destination(s)

Show platform hardware fed forward summary

```
Vlan                : 0
Mapped Vlan ID      : 0
Output Packet Details:
  Port              : TenGigabitEthernet3/0/2
###[ Ethernet ]###
  dst               = d0:ec:35:c9:d3:64
  src               = 00:ee:ab:9f:b1:4c
  type              = 0x800
###[ IP ]###
  version           = 4
  ihl               = 5
  tos               = 0xc0
  len               = 109
  id                = 51185
  flags             = DF
  frag              = 0
  ttl               = 63
  proto             = udp
  chksum            = 0x94d9
  src               = 172.20.26.6
  dst               = 10.48.13.171
```

The packet as it would go out on the port

L2 header as it would go out after routing

Sample packet now has TTL of 63

Complete Your Session Evaluations



Complete a minimum of 4 session surveys and the Overall Event Survey to be entered in a drawing to **win 1 of 5 full conference passes** to Cisco Live 2025.



Earn 100 points per survey completed and compete on the Cisco Live Challenge leaderboard.



Level up and earn **exclusive prizes!**



Complete your surveys in the **Cisco Live mobile app**.

Continue your education

- Visit the Cisco Stand for related demos
- Book your one-on-one Meet the Expert meeting
- Attend the interactive education with DevNet, Capture the Flag, and Walk-in Labs
- Visit the On-Demand Library for more sessions at www.CiscoLive.com/on-demand

Contact me at: jackkylee@cisco.com



Thank you

CISCO *Live!*

#CiscoLiveAPJC