Test post 2 25.8 upgrade

Introduction

Prerequisites

Requirements

Components Used

Configure

Network Diagram

Configurations

Verify

Troubleshoot

This is a high level troubleshooting guide to help engineers approach how to troubleshoot a traffic drop issues on ASR9000. It may not cover all scenarios but we have tried to generalise most common cases.

Jargon Buster

- **GDPlane :** Generic DataPlane
- **CEF :** Cisco Express Forwarding
- **RFD** Receive Frame Descriptor
- **PLU :** Prefix Lookup Unit
- **PHU :** PLU Hint Unit
- **TBM :** Tree Bit Map
- **BUM:** Broadcast/Unknown Unicast/L2 Multicast
- **LC -** Line Card
    - **Tomahawk-Based Line Cards**

The third generation of the ASR 9000 Series Ethernet line cards are often referred to as Tomahawk-based line cards. The term comes from the NPs that are used on these line cards.

- **Lightspeed-Based Line Cards**

The fourth generation of the ASR 9000 Series Ethernet line cards are often referred to as Lightspeed-based line cards. The term comes from the NPs that are used on these line cards. They are at times refered to as LSQ.

- **Lightspeed-Plus-Based Line Cards**

The fifth generation of the ASR 9000 Series Ethernet line cards are often referred to as Lightspeed-Plus-based line cards. The term comes from the NPs that are used on these line cards. They are at times refered to as LSP.

Understand ASR 9000 Series Line Card Types

- **VNI:** VxLAN Identified
  - **L2VNI:**Layer-2 Vxlan identifier
  - **L3VNI:** Layer-3 Vxlan identifier

- **Flood -** Typically unicast packet will go out only to one egress port. But if the switch doesn't know where to send the packet to (due to a MAC miss) then we will send the packet to all the member ports of the incoming vlan. Thats called a flood.
- **FGID -** Fabric Group Identifier
- **MGID -** Multicast Group Identifier

Introduction

This document list down various packet drop scenarios and the step by step method to proceed with debugging these cases.

R9K - Life of a Packet

Ingress Feature Ordering

Egress Feature Ordering

Modules involved in packet processing

"For us" traffic

A "for us" packet can be destined to LC or RSP depending on the application.

- ***For Punt to LC CPU***
  Packet From Wire → NP <-> Punt Switch <-> SPP (LC CPU) <-> Netio/Spio client <-> Application

- ***For Punt to RP CPU***
  Packet From Wire *→NP <-> LC FIA <-> Crossbar <-> RSP FIA <-> Punt/Dao/Cha FPGA <-> SPP (RSP CPU) <-> Netio/Spio client*

Transit traffic

- ***Packet From Wire*** *→Ingress NP <-> LC FIA <-> Crossbar <-> Egress NP →* ***Packet out to Wire***

Inject traffic

- ***From LC CPU***
  - **Egress inject**
    Application <-> Netio/Spio client <-> SPP (LC CPU) <-> Punt switch <-> Egress NP → Packet out to Wire

    - **Inject inward**
      *Application <-> SRPM <-> Punt switch <-> Ingress NP <-> LC FIA <-> Crossbar <-> Egress NP → **Packet out to Wire***

- ***From RP CPU***
    - **Egress inject**
      *Application <-> Netio/Spio client <-> SPP (RP CPU) <-> RSP FIA <-> Crossbar <-> LC FIA <-> Egress NP → **Packet out to Wire***

- ***LC CPU to RSP CPU***
  Netio/Spio client <-> SPP (LC CPU) <-> Punt switch <-> NP <-> LC Fia <-> Crossbar <-> RSP Fia <-> Punt/Dao/Cha FPGA <-> SPP (RSP CPU) <-> Netio/Spio client

- ***RSP CPU to LC CPU***
  Netio/Spio client <-> SPP (RSP CPU) <-> Punt/Dao/Cha FPGA <-> RSP Fia <-> Crossbar <-> LC Fia <-> NP <-> Punt switch <-> SPP (LC CPU) <-> Netio/Spio client

---

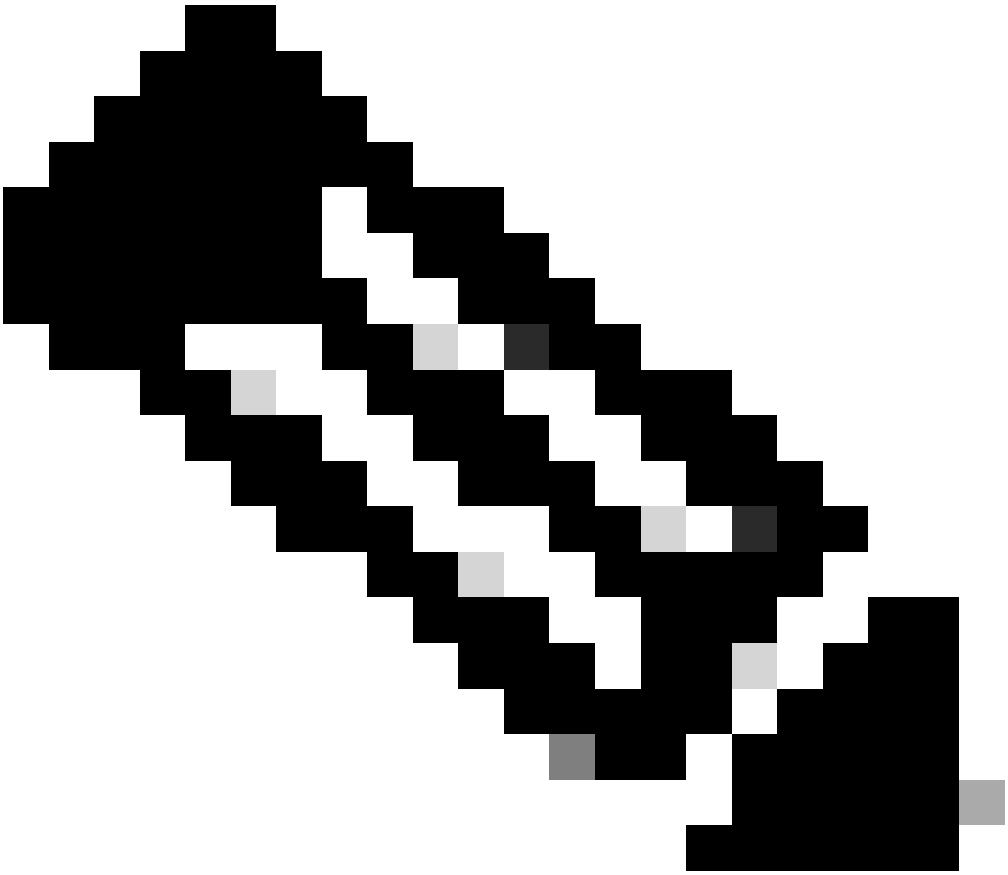
Identifying the problematic device :

1. ***Classify the traffic issue***
   Find out the type of issue. Identify if it is a complete drop, partial packet drop, silent drops or some other scenario.

2. ***Determine the traffic type***
   L2/L3/Unicast/BUM/Multicast

3. ***Identify a single victim flow***
   Wherever possible from the IXIA drill down and find one single flow which we will use to triage further

4. ***Trace the single flow and narrow down the Device (Suspect Device) which drops/starts duplication***



Note: Use the topology diagram and show interface counters to derive the actual path the packet is taking

---

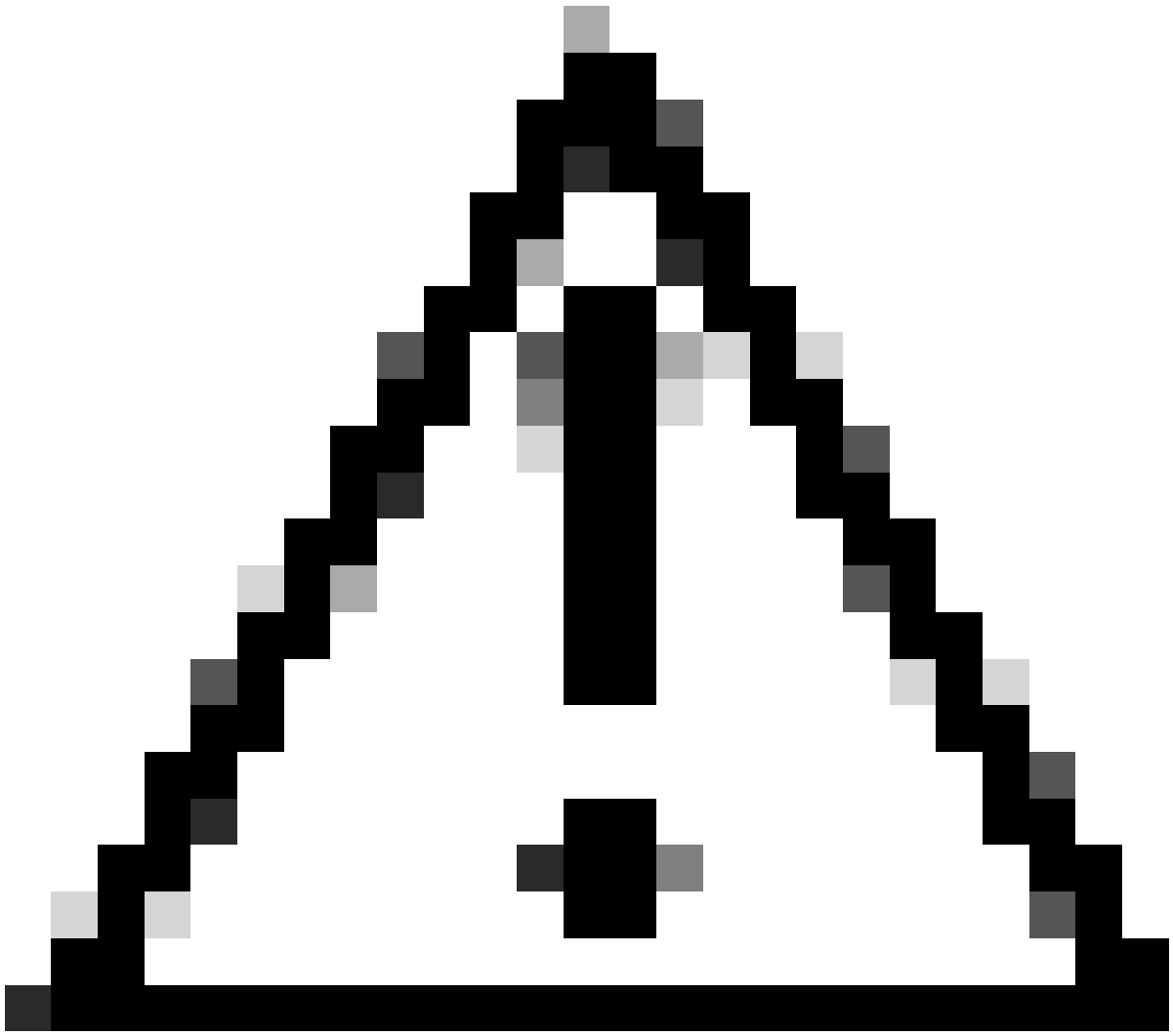
1. On IXIA stop all the traffic and create a new traffic stream "DEBUG" which has the single flow which we selected in step 2 and set the traffic rate to some higher rate (say 10K PPS if its a data traffic, for ARP/other sup bound control plane keep it at lower rates as it goes through rate limiters/copp)

2. On IXIA start only the new traffic item and starting from the ingress device use "sh int counters brief" to find on which device the problem starts(drop/duplication)

5. ***Get the below details of the flow and note it down for easier troubleshooting***
    1. Source MAC
    2. Destination MAC
    3. Source ip
    4. Destination IP
    5. Source Vlan
    6. Destination Vlan(if its routed traffic)
    7. VRF information of source and destination (if its L3 routed traffic)
    8. VNI information
        1. L2VNI if L2 traffic
        2. L3VNI if its L3 routed traffic

6. ***After narrowing down the flow, use below methods to locate the problematic router/device.***
    1. Traceroute/Ping/MPLS Ping/Ethernet Ping
    2. ACL - Check if traffic is really hitting the ingress ports of devices
    3. Interface counters
    4. Interface controller stats
    5. Label Switching stats
7. ***Verify there are no config related issues***
   Refer thesection of some of the common misconfigurations.

Start of actual debugging from the Suspect Device



Caution: Though Suspect device is dropping/flooding the traffic that doesn't mean thats the culprit. In some cases the device which sends traffic to Suspect device can be the culprit.

Place/Module of the traffic drop :

Interface level drops

- show interface <interface>

RP/0/RSP0/CPU0:YOG-CDCT-CN2-C9910# show interface Bundle-Ether602.3048 Thu May 11 13:16:40.091 WIB Bundle-

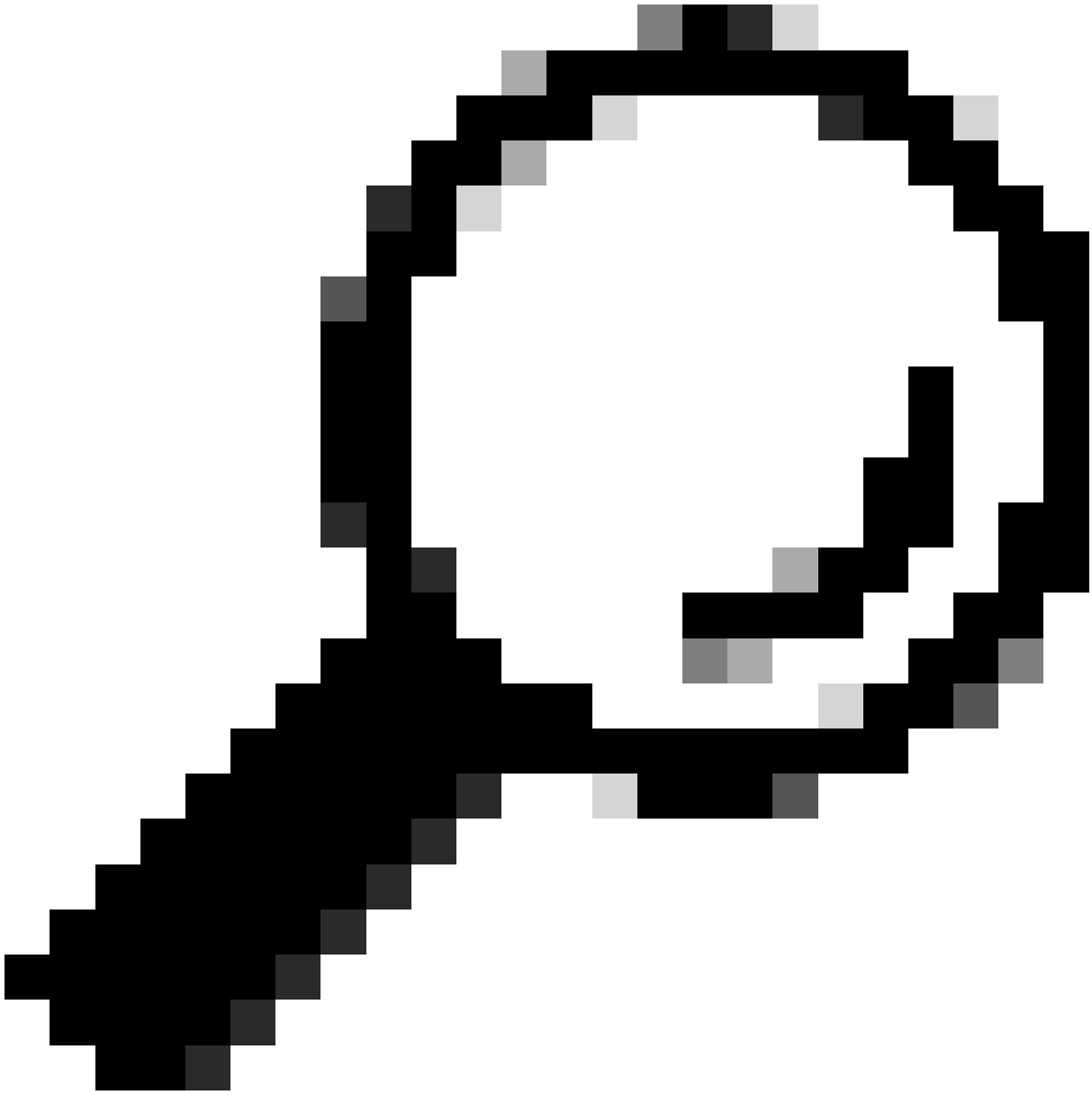
Ether602.3048 is up, line protocol is up Interface state transitions: 1 Dampening enabled: penalty 0, not suppressed half-life: 1 reuse: 750 suppress: 2000 max-suppress-time: 4 restart-penalty: 0 Hardware is VLAN sub-interface(s), address is ecce.13c9.d8c5 Description: ABIS_MCBSC_CDC4_NSN_VLAN3048 Internet address is 10.17.191.179/28 MTU 9216 bytes, BW 2000000 Kbit (Max: 2000000 Kbit) reliability 255/255, txload 0/255, rxload 0/255 Encapsulation 802.1Q Virtual LAN, VLAN Id 3048, loopback not set, Last link flapped 36w1d ARP type ARPA, ARP timeout 04:00:00 Last input 00:00:00, output never Last clearing of "show interface" counters 135y46w 30 second input rate 4000 bits/sec, 9 packets/sec 30 second output rate 0 bits/sec, 0 packets/sec 85212564 packets input, 5396765891 bytes, 2493252749 total input drops 0 drops for unrecognized upper-level protocol Received 20089331 broadcast packets, 39963824 multicast packets 70999919503 packets output, 7186711514645 bytes, 0 total output drops Output 309 broadcast packets, 57 multicast packets

- show controllers <interface> stats

```
RP/0/RSP0/CPU0#show controller hundredGigE0/3/0/40 stats Wed Oct 18 16:54:04.904 WEST
Statistics for interface HundredGigE0/3/0/40 (cached values): Ingress: Input total bytes =
16850796039449085 Input good bytes = 16850796039449085 Input total packets = 13769166661248
Input 802.1Q frames = 0 Input pause frames = 0 Input pkts 64 bytes = 257446881559 Input pkts
65-127 bytes = 1285971034813 Input pkts 128-255 bytes = 401173319511 Input pkts 256-511
bytes = 261817914140 Input pkts 512-1023 bytes = 323254550402 Input pkts 1024-1518 bytes =
6444289537421 Input pkts 1519-Max bytes = 4795213423402 Input good pkts = 13769166661248
Input unicast pkts = 13769157512691 Input multicast pkts = 9147481 Input broadcast pkts =
1076 Input drop overrun = 0 Input drop abort = 0  Input drop invalid VLAN = 0 Input drop
invalid DMAC = 0 Input drop invalid encap = 0 Input drop other = 0 Input error giant = 0
Input error runt = 0 Input error jabbers = 0 Input error fragments = 0 Input error CRC = 0
Input error collisions = 0 Input error symbol = 0 Input error other = 0 Input MIB giant =
4795213423402 Input MIB jabber = 0 Input MIB CRC = 0  Egress: Output total bytes =
3150799484820437 Output good bytes = 3150799484820437 Output total packets = 5987194620610
Output 802.1Q frames = 0 Output pause frames = 0 Output pkts 64 bytes = 59685948036 Output
pkts 65-127 bytes = 3272599163757 Output pkts 128-255 bytes = 477536708073 Output pkts 256-
511 bytes = 150420175953 Output pkts 512-1023 bytes = 191150155497 Output pkts 1024-1518
bytes = 999535758139 Output pkts 1519-Max bytes = 836266711152 Output good pkts =
5987194446122 Output unicast pkts = 5987180721273 Output multicast pkts = 13724849 Output
broadcast pkts = 0 Output drop underrun = 0 Output drop abort = 0 Output drop other = 0
Output error other = 174488
```

Early Fast drops

How to check EFD is dropping the packets ?



Tip: It is important to note that, tomahawk EFD drops are not shown in "show controller np counters <>" command output nor in "show drops" command output. A new enhancement request is opened to include EFD drops in "show drops" command. Refer

---

RP/0/RP0/CPU0:asr9k-1# sh controllers np fast-drop np0 location 0/0/CPU0 Fri Jan 27 12:17:57.333 PST Node: 0/0/CPU0: ---------- ---------- All fast drop counters for NP 0: TenGigE0/0/0/1/0-TenGigE0/0/0/1_9:[Priority1] 0 TenGigE0/0/0/1/0-TenGigE0/0/0/1_9:[Priority2] 0 TenGigE0/0/0/1/0-TenGigE0/0/0/1_9:[Priority3] 0 HundredGigE0/0/0/0-TenGigE0/0/0_1_9:[Priority1] 0 HundredGigE0/0/0/0-TenGigE0/0/0_1_9:[Priority2] 0 HundredGigE0/0/0/0-TenGigE0/0/0_1_9:[Priority3] 123532779 <=== Priority 3 packets dropped ---------- RP/0/RP0/CPU0:asr9k-1#

What data to collect?

"**np_perf**" facility on ASR9000 Tomahawk and Lightspeed line cards in troubleshooting NP fast drops

NP drops

1.Identify the relevant NP based on the ingress port information. The below command can be used to identify the NP

show controllers np portmap all location < >

```
RP/0/RSP0/CPU0:SRv6-R5# show controllers np portmap all location 0/1/CPU0 Wed May 17
05:30:40.389 EDT Node: 0/1/CPU0: ----------------------------------------------------
--------- Show Port Map for NP: 0, and RX Unicast Ports phy port num interface desc
uiMappedSourcePort 0 HundredGigE0_1_0_0 0 10 HundredGigE0_1_0_1 10 20 HundredGigE0_1_0_2
20 30 HundredGigE0_1_0_3 30 Show Port Map for NP: 1, and RX Unicast Ports phy port num
interface desc uiMappedSourcePort 0 HundredGigE0_1_0_4 0 10 TenGigE0_1_0_5_0 10 11
TenGigE0_1_0_5_1 11 12 TenGigE0_1_0_5_2 257 (bundle) 13 TenGigE0_1_0_5_3 13 20
HundredGigE0_1_0_6 20 30 TenGigE0_1_0_7_0 30 31 TenGigE0_1_0_7_1 31 32 TenGigE0_1_0_7_2
256 (bundle) 33 TenGigE0_1_0_7_3 33 RP/0/RSP0/CPU0:SRv6-R5#
```

2. Check the np counter statistics for the NP identified in Step (a).

show controller np counters <npnum>/all > location < >

RP/0/RSP0/CPU0:SRv6-R5# show controller np counters all location 0/3/CPU0 Wed Oct 18 16:54:46.557 WEST Node: 0/3/CPU0: ---------- Show global stats counters for NP0, revision v0 Last clearing of counters for this NP: 2543:27:1 Read 0 non-zero NP counters: Offset Counter FrameValue Rate (pps) ---------- ---------- ---------- 104 BFD discriminator zero packet 2 0 158 IPv4 PIM all routers detected 31878304 3 170 IPv6 LL hash lookup miss on egress 1 0 192 L2 MAC learning source MAC lookup miss 53 0 193 L2 MAC move on egress NP 55 0 194 L2 MAC notify delete 15 0 195 L2 MAC notify delete no entry 2 0 198 L2 MAC notify learn complete 2520170 0 200 L2 MAC notify received 21518947 3 201 L2 MAC notify reflection filtered 113082 0 202 L2 MAC notify refresh complete 18885133 3 205 L2 MAC notify update with bridge domain flush 366 0 206 L2 MAC notify update with port flush 30 0 208 L2 MAC update via reverse MAC notify skipped 2 0 220 L2 aging scan delete from BD key mismatch 108 0 221 L2 aging scan delete from XID invalid 3 0 223 L2 aging scan delete from entry aging out 2516745 0 227 L2 egress MAC modify 18885584 3 246 L2 ingress MAC bridge domain flush 2 0 250 L2 ingress MAC learn 53 0 251 L2 ingress MAC modify 113027 0 253 L2 ingress MAC move 2 0 256 L2 ingress MAC refresh update 113025 0 266 L2 on demand scan delete from BD key mismatch 3060 0 267 L2 on demand scan delete from XID invalid 49 0 292 MAPT - TBPG Event 1562667425 171 349 TBPG L2 mailbox events 1376253865 150 350 TBPG MAC scan events 22942615 3 351 TBPG stat events 88080247335 9620 361 VPLS egress MAC notify MAC lock retry 607 0 363 VPLS egress MAC notify entry lock retry 176 0 372 VPLS ingress entry lock not acquired 4758 0 373 VPLS ingress entry lock retry 1362180 0 400 DMAC mismatch MY_MAC or MCAST_MAC for L3 intf 1 0 420 GRE IPv4 decap qualification failed 34389 0 431 GRE IPv6 decap qualification failed 34 0 460 IP multicast route drop flag enabled 10985 0 463 IPv4 BFD SH packet TTL below min 2705 0 464 IPv4 BFD SH packet invalid size 2294 0 486 IPv4 multicast egress no route 1363073 0 488 IPv4 multicast fail RPF drop 222733 0 550 L2 VNI info no hash entry drop 7 0 562 L2 egress VLAN tag missing drop 97 0 576 L2 ingress LAG no match drop 172286 0 592 L2 ingress flood null FGID drop 62617 0 602 L2 on L3 ingress unknown protocol 4577940 0 617 LAC subscribers L2TP version mismatch drop 404 0 623 LSM dropped due to egress drop flag on label 3 0 635 MPLS over UDP decap is disabled 5 0 650 P2MP carries more than two labels 27398 0 652 P2MP with invalid v4 or v6 explicit null label 60273 0 671 Queue tail drops due to queue buffer limit 67132 0 728 VPWS ingress DXID no match drop 5 0 729 WRED curve probability drops 22229 0 734 CLNS multicast from fabric pre-route 1502161 0 738 IPv4 from fabric 984281239 206 739 IPv4 from fabric pre-route 4472288 0 740 IPv4 inward 18133144 2 742 IPv4 multicast from fabric pre-route 3293512 0 745 IPv6 from fabric 2412441 0 747 IPv6 link-local from fabric pre-route 281239 0 749 IPv6 multicast from fabric pre-route 529 0 753 Inject to fabric 406582755 151 754 Inject to port 199262152 106 755 MPLS from fabric 295962479 30 759 Pre-route punt request 101423 0 1410 Drop due to invalid table content 9 0 1418 IPv4 egress null route 1 0 1423 IPv4 invalid length in ingress 21 0 1443 MPLS MTU exceeded 3512168 0 1466 MPLS invalid payload when disposing all labels 85 0 1468 MPLS leaf with no control flags set 13281 0 1470 MPLS receive adjacency 1 0 1503 ARP 65599 0 1518 Bundle protocol 27441792 3 1524 Diags 152495 0 1572 IPv4 options 188 0 1587 ICMP generation needed 33 0 1599 TTL exceeded 173434294 21 1600 Punt policer: TTL exceeded 7506 0 1602 IPv4 fragmentation needed 213116 12 1605 IPv4 BFD 1288 0 1611 IFIB 466671481 157 1612 Punt policer: IFIB 413684 0 1632 IPv6 hop-by-hop 1285 0 1635 IPv6 TTL error 2230163 0 1695 Diags RSP active 152501 0 1698 Diags RSP standby 152559 0 1701 NetIO RP to LC CPU 78667597 8 1716 SyncE 9155455 1 1749 MPLS fragmentation needed 16 0 1752 MPLS TTL exceeded 194 0 1755 IPv4 adjacency null route 9760672 0 1756 Punt policer: IPv4 adjacency null route 1673465 0 1809 PTP ethernet 516895376 56 1899 DHCP broadcast 891 0 1995 IPv4 incomplete Rx adjacency 64643796 6 1996 Punt policer: IPv4 incomplete Rx adjacency

26014 0 1998 IPv4 incomplete Tx adjacency 9143978 1 1999 Punt policer: IPv4 incomplete Tx adjacency 13053 0 2013 IPv6 incomplete Tx adjacency 206 0 2022 MPLS incomplete Tx adjacency 339606 0 2028 Remote punt BFD 31 0 HW Received from Line 29024842290654 3235969 HW Transmit to Fabric 29024410231530 3235922 HW Received from Fabric 23530268012585 2664187 HW Transmit to Line 23530333637629 2664266 HW Host Inject Received 605997250 257 HW Host Punt Transmit 980248296 225 HW Local Loopback Received at iGTR 1081176162 309 HW Local Loopback Transmit by iGTR 1081176162 309 HW Local Loopback Received at Egress 1081176162 309 HW Transmit to TM from eGTR 23531332324079 2664493 HW Transmit to L2 23531313885879 2664491 HW Received from Service Loopback 18438204 2 HW Transmit to Service Loopback 18438204 2 HW Internal generated by PDMA 214940487672 23474

a . L3 specific counter

For Drops you shall check on the below wiki, to find the reason for it. See if it falls under L3 category.

If it's L3 category please get the all the L3 related outputs related to the flow.
L3 CEF Chain output and other show commands

show cef [ipv4 | ipv6 | mpls] hardware [ingress | egress] detail location <LC>

show mpls forwarding labels <LABEL> hardware egress detail location <LC>

show cef vrf <vrf> <IP> internal location <LC>

show cef vrf <vrf> <IP> hardware [ingres | egress] location <LC>

show cef mpls local-label <LABEL> EOS

show cef mpls local-label <LABEL> non-eos location <LC>

show mpls forwarding labels <LABEL> det hardware [ingres | egress] location <LC>

show commands related to Interface level outputs [Sub-interface, Bundles and its members..]

show uidb im database and its chain related to the interface.

show bundle <> related commands if bundle is invovled.

sh controllers pm vqi location <LC

PI Commands :

sh cef <IP> internal location <LC>

sh cef <IP> detail location <LC>

show cef unresolved loc <>

show cef adjacency loc <>

show cef [drops | exception] loc <>

show cef [misc | summary] loc <>

show cef [ipv4 | ipv6 | mpls] trace [error | eve | table] loc <>

show cef interface <> loc <>

show mpls forwarding [..] loc <>

b. L2 Specific counter

For Drops you shall check on the below wiki, to find the reason for it. See if it falls under L2 category.
LSP All Counter Details
If it's L2 category please get the all the L2 related outputs related to the flow.
L2VPN Chain output and other show commands

show l2vpn forwarding hardware ingress detail location <LC>

show l2vpn forwarding hardware egress detail location <LC>

show l2vpn forwarding bridge-domain <BD Group: BD Name> hardware ingress detail location <LC>

show l2vpn forwarding bridge-domain <BD Group: BD Name> hardware egress detail location <LC>

show l2vpn forwarding bridge-domain mac-address hardware ingress location <LC>

show l2vpn forwarding interface pw-ether <PW> hard detail location <LC>

show l2vpn xconnect interface pw-ether <PW> detail

show l2vpn forwarding main-port pwhe interface pw-ether600 hardware ingress detail location <LC>

show l2vpn mstp port msti 0

show l2vpn mstp port msti 1

show commands related to Interface level outputs [Sub-interface, Bundles and its members..]

show uidb im database and its chain related to the interface.

show bundle <> related commands if bundle is invovled.

sh controllers pm vqi location < >

show l2vpn forwarding bridge-domain mac-address internal private first 1000 location 0/RP0/CPU0

show evpn internal-label private location 0/RP0/CPU0

show evpn internal-label path-list private location 0/RP0/CPU0

show evpn internal-id private location 0/RP0/CPU0

show l2vpn forwarding bridge-domain mac-address internal private first 1000 location 0/RP1/CPU0

show evpn internal-label private location 0/RP1/CPU0

show evpn internal-label path-list private location 0/RP1/CPU0

3 .Do capture the monitor np counter on the drop counter.

Example: To monitor on the counter <DROP_COUNTER_NAME>, execute

monitor np counter <DROP_COUNTER_NAME> <np> location <LC>

RP/0/RP0/CPU0:agg03.rjo# monitor np counter PARSE_DROP_IPV4_CHECKSUM_ERROR np0 location 0/1/cpu0 Tue Oct 5 10:49:30.349 BRA Usage of NP monitor is recommended for cisco internal use only. Please use instead 'show controllers np capture' for troubleshooting packet drops in NP and 'monitor np interface' for per (sub)interface counter monitoring Warning: Every packet captured will be dropped! If you use the 'count' option to capture multiple protocol packets, this could disrupt protocol sessions (eg, OSPF session flap). So if capturing protocol packets, capture only 1 at a time. Warning: A mandatory NP reset will be done after monitor to clean up. This will cause ~150ms traffic outage. Links will stay Up. Proceed y/n [y] > y Monitor PARSE_DROP_IPV4_CHECKSUM_ERROR on NP0 ... (Ctrl-C to quit) Tue Oct 5 10:49:33 2021 -- NP0 packet From HundredGigE0_1_0_0: 86 byte packet 0000: b0 26 80 67 3c bd bc 16 65 5e 2c 04 08 00 45 00 0&.g<=<.e^,...E. 0010: 00 48 cb b9 40 00 38 11 53 7f b3 e8 5e 74 08 08 .HK9@.8.S.3h^t.. 0020: 08 08 b5 a6 00 35 00 34 e5 22 d0 f0 01 00 00 01 ..5&.5.4e"Pp.... 0030: 00 00 00 00 00 00 0e 6e 72 64 70 35 31 2d 61 70 ....nrdp51-ap 0040: 70 62 6f 6f 74 07 6e 65 74 66 6c 69 78 03 63 6f pboot.netflix.co 0050: 6d 00 00 01 00 01 m.....

monitor np counter <DROP_COUNTER_NAME> <np> detail location <LC>

RP/0/RP0/CPU0:PE23#monitor np counter 12 np3 detail location 0/0/CPU0 Mon Mar 11 18:20:49.180 IST Usage of NP monitor is recommended for cisco internal use only. Warning: Using monitor will cause brief traffic loss twice for setup and takedown. Each outage could exceed ? ms. After a packet is monitored, it will resume normal handling (i.e. forward, punt, drop, etc). Setup ... ready for traffic outage? [enter] Monitoring NP3 for NP counter 12 [Invalid stats pointer 12] ... (Ctrl-C to quit) Mon Mar 11 18:21:25 2024 -- NP3 packet 150 bytes ---------- ---------- NPU 03: Cluster 11: PPE 15: Thread 00 Ptrace 00 Received from : Fabric GPM Pkt Dump Contents: 00 04 08 0C 10 14 18 1C G 000: 70e42225 e554f86b d9a39247 88470057 80049000 0054004a 00000000 00000000 G 020: 00000000 9424118c 05000400 000000a6 c024400f 00000001 248c80c1 cd000004 G 040: 000186a0 000186a0 00000000 dad4994e 00200000 20c80318 000101ba 00060296 G 060: 0111bef8 64001700 7f000001 c0000ec8 03e83cff 064eddff 45c00034 00000000 G 080: 801318e5 044420b0 0000fc00 00000020 0000ff00 0000ff80 0000fd00 00000000 G 0a0: c040401d 82000201 080e0000 000e0000 eb190080 00000004 053942d2 00d20048 G 0c0: 003bbbff 0001f86b d9a3923f 810003e9 08004500 00640000 0000ff01 8bd21764 G 0e0: 00fe1764 00010800 b4ee1a25 0000abcd abcdabcd abcdabcd abcdabcd abcdabcd G 100: abcdabcd abcdabcd abcdabcd abcdabcd abcdabcd abcdabcd abcdabcd abcdabcd G 120: abcdabcd abcdabcd abcdabcd abcdabcd abcdabcd abcd49d6 79520000 00000000 DMEM Contents: 00 04 08 0C 10 14 18 1C 000: 40000256 009609c0 00000000 00000000 deadbeef deadbeef deadbeef deadbeef 020: f0000010 05000172 40000000 00000000 00000000 810003e9 00000000 0004fc49 040: 70000000 01004c1b 64000300 00000000 02000000 00000000 0000000e 000992a1 060: 00000000 00000000 00000000 01800000 00000000 f0000000 00000000 06d035fb 080: 000f0109 00711ef0 00000046 06d035fb 41800000 01000001 64000b00 00064ed1 0a0: 8200f86b d9a39247 1e01f25f 00009003 00000000 00000000 00000000 00000000 0c0: c0000000 00018c00 001e0000 00000000 000f0109 0077e248 00000076 00000000 0e0: 8200f86b d9a39247 1e01f25f 00009003 00000000 00000000 00000000 00000000 100: 83211001 00000000 0d94001b 0992a096 07aa60a2 1e000fa0 1c000001 00000000 120: deadbeef deadbeef deadbeef deadbeef deadbeef deadbeef deadbeef deadbeef 140: deadbeef deadbeef deadbeef deadbeef deadbeef deadbeef deadbeef deadbeef 160: deadbeef deadbeef deadbeef deadbeef deadbeef deadbeef deadbeef deadbeef 180: deadbeef deadbeef deadbeef deadbeef deadbeef deadbeef deadbeef deadbeef 1a0: deadbeef deadbeef deadbeef deadbeef deadbeef deadbeef deadbeef deadbeef 1c0: deadbeef deadbeef deadbeef deadbeef 28010100 00000000 00000000 0000004b 1e0: d0000000 00000000 00002000 0000004b 90004031 800019d3 00000000 00000000 200: 00000004 00000004 01000000 0a000000 b8008810 01010003 00000000 00000000 220: 00000000 00000000 00000000 010423de deadbeef deadbeef deadbeef deadbeef 240: deadbeef deadbeef deadbeef deadbeef deadbeef deadbeef deadbeef deadbeef 260: deadbeef deadbeef deadbeef deadbeef deadbeef deadbeef deadbeef deadbeef 280: deadbeef deadbeef deadbeef deadbeef deadbeef deadbeef deadbeef deadbeef 2a0: deadbeef deadbeef deadbeef deadbeef deadbeef deadbeef deadbeef deadbeef 2c0: 000f0108 004aea60 deadbeef deadbeef 001e0003 80000000 00000008 deadbeef 2e0: deadbeef deadbeef deadbeef deadbeef deadbeef deadbeef dead0096 00021000 300: 00008008 0992a004 04fc49ef deadbeef deadbeef deadbeef 000001ef deadbeef 320: 083961ef 03e83cff 061addff

deadbeef deadbeef deadbeef deadbeef deadbeef 340: deadbeef deadbeef deadbeef deadbeef c0000ec8 000011ef deadbeef 00000172 360: deadbeef 0d00beef 00000000 0000000e 00000000 000210c0 a2961b00 000210d2 380: 00000084 deadbeef 000210ec 03e83cff deadbeef deadbeef 00009000 0001f250 3a0: 800019d3 000210a0 deadbeef deadbeef 00000000 02000000 deadbeef 00000000 3c0: 00680d8f 01d20000 deadbeef 00ffbeef deadbe00 000000ef deadbeef deadbeef 3e0: 02000000 05800010 00040004 00000000 00020390 3195f8fe 04441100 044422c0 Register Contents: r00: 800980b9 00000001 0000000c 0000000c 004aea60 3c084204 00000032 18000004 r08: 04442200 044421d0 00000008 00000003 000210a0 00000004 00000001 00000001 r16: 80119bab 04442170 0000fc00 fffff000 000210a0 000210c0 000000ff 00ffffff r24: 80125faf 04442140 0000fc00 000210c0 000210a0 00000001 00000001 00500000 = = Above Register Contents are from windowbase = 3 = = = = Easy-to-read Register Contents (windowbase: 0) = = a00 000210a0 | a08 000210a0 | a16 004aea60 | a24 000210a0 a01 000210c0 | a09 00000001 | a17 3c084204 | a25 00000004 a02 000000ff | a10 00000001 | a18 00000032 | a26 00000001 a03 00ffffff | a11 00500000 | a19 18000004 | a27 00000001 a04 80125faf | a12 800980b9 | a20 04442200 | a28 80119bab a05 04442140 | a13 00000001 | a21 044421d0 | a29 04442170 a06 0000fc00 | a14 0000000c | a22 00000008 | a30 0000fc00 a07 000210c0 | a15 0000000c | a23 00000003 | a31 fffff000 Special Registers: sar = 0000001a window_start = 0000008a window_base = 00000003 epc = 00040250 exccause = 00000001 (SycallCause) ps = 00020330 sse_cop_errorcode = 00000000 h3ta = deb4cf17 h3tb = deadbf03 h3tc = deadbef7 h3tr = 2609d946 timestamp_high_1 = 00095a09 timestamp_low_1 = 19f84b99 cycle_count_high_1 = 0000008c cycle_count_low_1 = 1a732982 ppe_id = 6bc60d7e uidb_ext0 = 3195f8fe uidb_ext1 = 00020390 uidb_ext2 = 00000000 uidb_ext3 = 00000000 uidb_ext4 = 00000000 uidb_ext5 = 00000000 softerr_misc = 00000034 timestamp_high_2 = 00095a09 timestamp_low_2 = 19f8609a cycle_count_high_2 = 0000008c cycle_count_low_2 = 1a732ae8 css = 00000001 ci_count_1 = 000003a8 thread_stop = 00000000 ci_count_2 = 000003a8 memctrl = 00000000 softerr_dmem0 = 60411000 softerr_dmem1 = 67f15000 code_segment0 = 08000000 code_segment1 = 08000000 l1t_data_sbe_log = 00000000 l1t_mshr_sbe_log = 00000000 random_1 = 19382747 stall_control = 00000003 l1t_tag_parity_log = 6bc60d7e random_2 = 697de1cb depc = 00000000 timeout_control = 00000008 rtb0 = 00000000 rtb1 = 0000000f invalidate = 00000000 tlu_cmd_hdr = 6bc60d7e l1t_enables = 00000003 l1t_replacement_way = 00000007 excvaddr = 00060330 l1t_data_mbe_log = 00000000 l1t_mshr_mbe_log = 00000000 sse_cop_seedh = 356c9a7b sse_cop_tlu_cnt1_rw = 00000000 sse_cop_lock = 800034a7 sse_cop_tlu_perr = 00000000 sse_cop_tlu_cnte1_rw = 00000000 sse_cop_tps_tpd_parity_log = 00000000 sse_cop_tps_tpt_parity_log = 00000000 sse_cop_tmu_parity_log_rw = 00000000 sse_cop_tmu_sram_mbe_log_rw = 00000000 sse_cop_tmu_sram_sbe_log_rw = 00000000 sse_cop_tlu_cnte0_rw = 00000000 sse_cop_tlu_cntp_rw = 00000000 sse_cop_tlu_cnt0_rw = 00000000 rtt_index = 00000000 rtt_data0 = 00000000 rtt_data1 = 00000000 ppe_lock = 8000000c stack_limit = 04441400 stack_limit_enable = 00000001 sse_cop_dma_mem_access = 00000000 sse_cop_dma_mem_data = 000000a0 error_code = 00000000(No Error)

4.HW programming details - More details on the Forwarding HW structs programming can be collected using below. (Useful for NP team to debug further)

L3

show controller np struct R-LDI unsafe det all all location <LC>
show controller np struct NR-LDI unsafe det all all location <LC>

show controller np struct TE-NH-ADJ unsafe det all all location <LC>

show controller np struct RX-ADJ unsafe det all all location <LC>

show controller np struct TX-ADJ unsafe det all all location <LC>

show controller np struct NHINDEX unsafe det all all location <LC>

show controller np struct LAG unsafe det all all location <LC>

show controller np struct LAG-Info unsafe det all all location <LC>

L2

show controller np struct UIDB-EGR-EXT unsafe det all all location <LC>

show controller np struct UIDB-Ext unsafe det all all location <LC>

show controller np struct UIDB-ING-EXT unsafe det all all location <LC>

show controller np struct EGR-UIDB unsafe det all all location <LC>

show controller np struct XID unsafe det all all location <LC>

show controller np struct XID-EXT unsafe det all all location <LC>

show controller np struct BD unsafe det all all location <LC>

show controller np struct BD-EXT unsafe det all all location <LC>

show controller np struct BD-LEARN-COUNT unsafe det all all location <LC>

show controller np struct L2-BRGMEM unsafe det all all location <LC>

show controller np struct l2-fib unsafe det all all location <LC>
LSP: run ssh lc0_xr /pkg/bin/show_l2ufib <Collect in all active LCs>

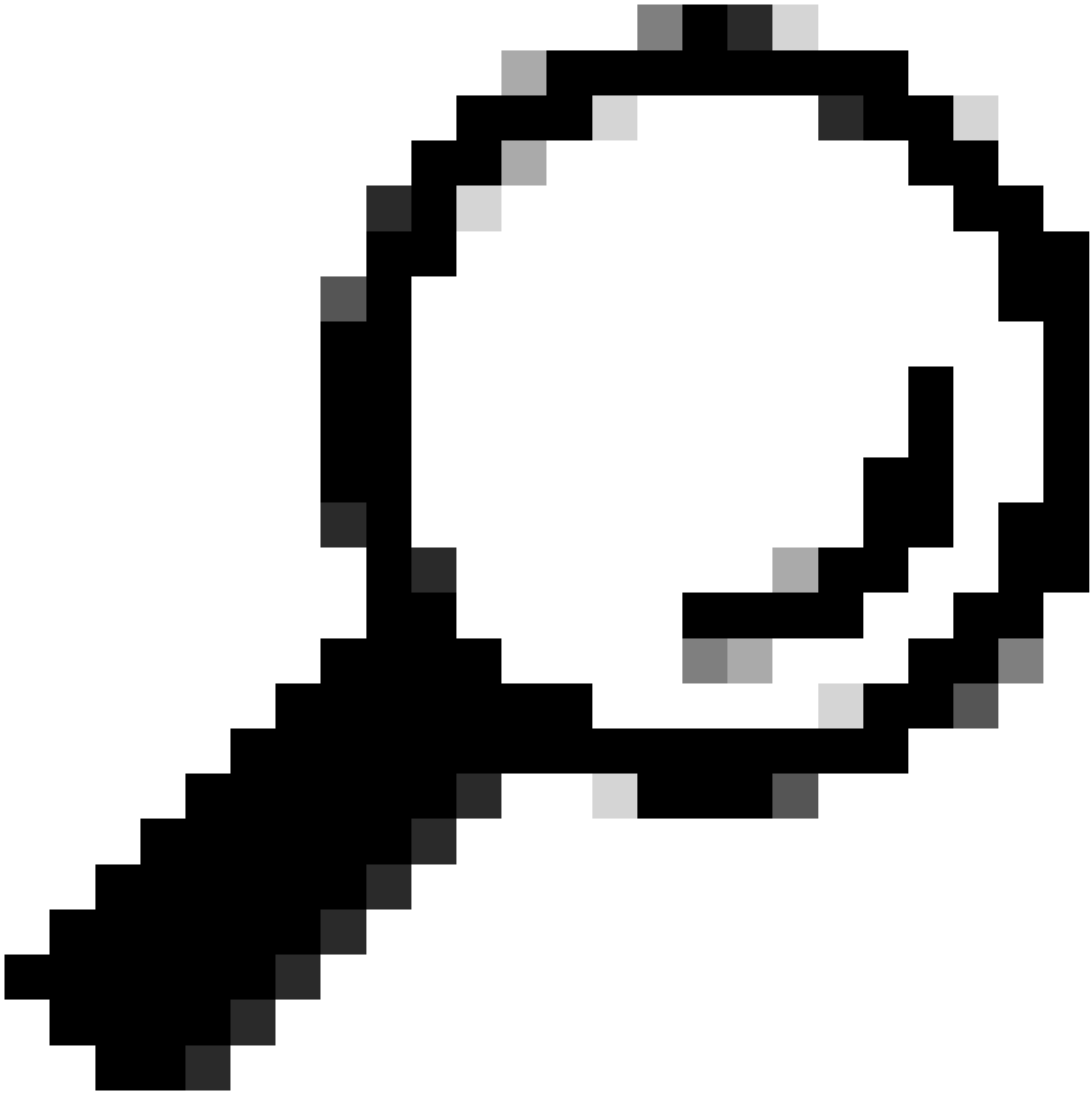
show controller np struct L2-MAILBOX unsafe det all all location <LC>

show controller np struct L2-MBX-HOST-TO-NP unsafe det all all location <LC>

show controller np struct L2-MBX-NP-TO-HOST unsafe det all all location <LC>

SPP

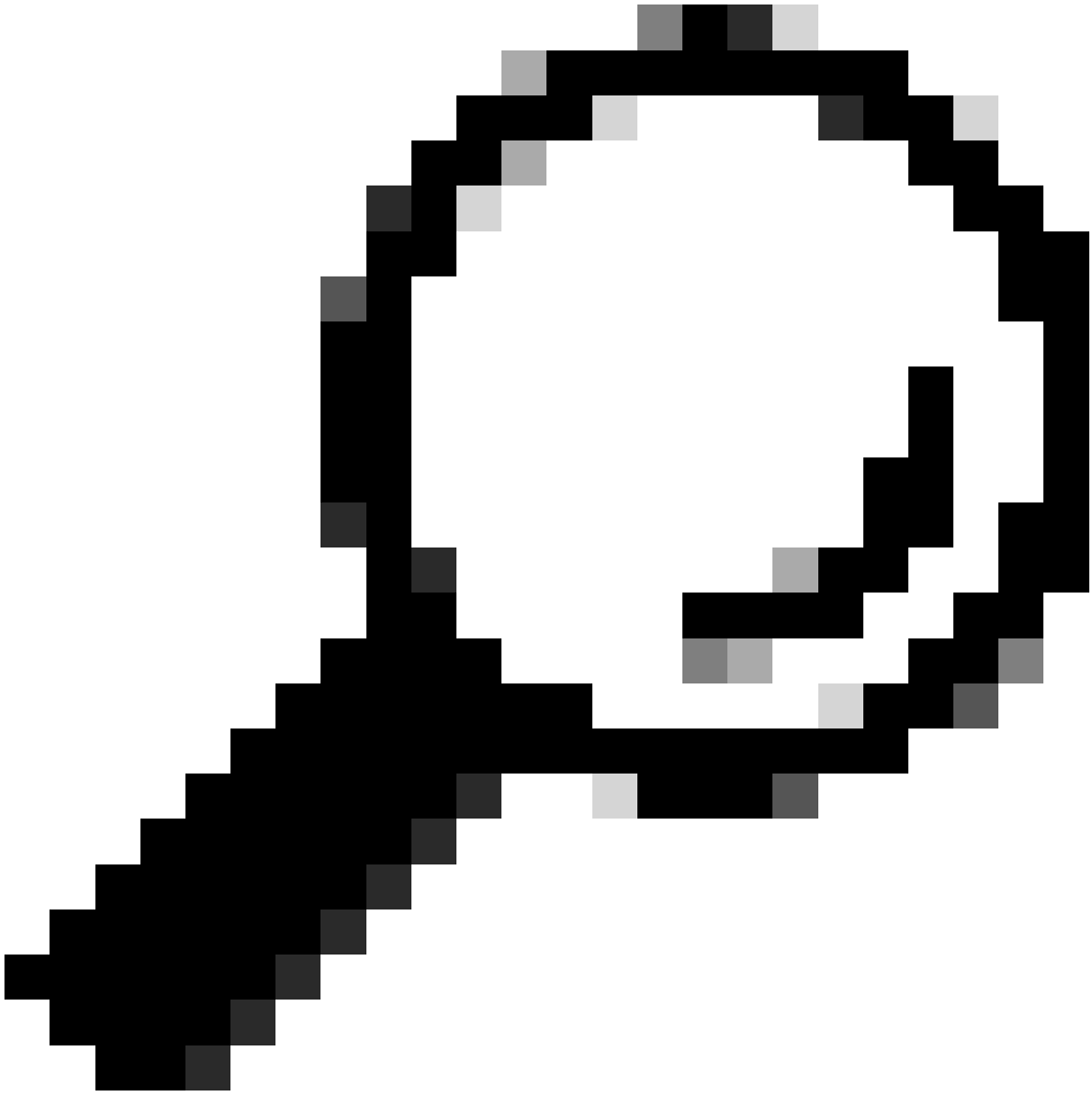
show spp sids stats location < >
show spp node-counters location < >



Tip: How to Capture/Filter packets at SPP? Refer

NETIO

show netio drops location < >



Tip: You can refer to

SRPM

ip maddr show eth-srpm (for checking multicast entries for the interface)

ip maddr show eth-srpm.1283 (for checking multicast entries for the interface)

ifconfig (check packet stats for the interfaces to see if RX and TX has been happening properly)

[xr-vm_node0_0_CPU0:~]$ip maddr show eth-srpm.1283 9: eth-srpm.1283 link 33:33:00:00:00:01 users 2 link 01:00:5e:00:00:01 users 2 link 33:33:ff:50:4e:52 users 2 link 01:56:47:50:4e:30 users 2 static link 33:33:00:01:00:03 users 2 inet 224.0.0.1 inet6 ff02::1:3 inet6 ff02::1:ff50:4e52 inet6 ff02::1 inet6 ff01::1 [xr-vm_node0_0_CPU0:~]$ifconfig eth-srpm.1283: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 9700 metric 1 inet6 fe80::544b:47ff:fe50:4e52 prefixlen 64 scopeid 0x20<link> ether 56:4b:47:50:4e:52 txqueuelen 1000 (Ethernet) RX packets 0 bytes 0 (0.0 B) RX errors 0 dropped 0 overruns 0 frame 0 TX packets 828560 bytes 138041161 (131.6 MiB) TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

LPTS

show lpts pifib hardware entry statistics loc <>
show lpts pifib hard police loc <>
show lpts pifib hardware static-police loc <>
show lpts pifib ha entry stats location <>

Fabric

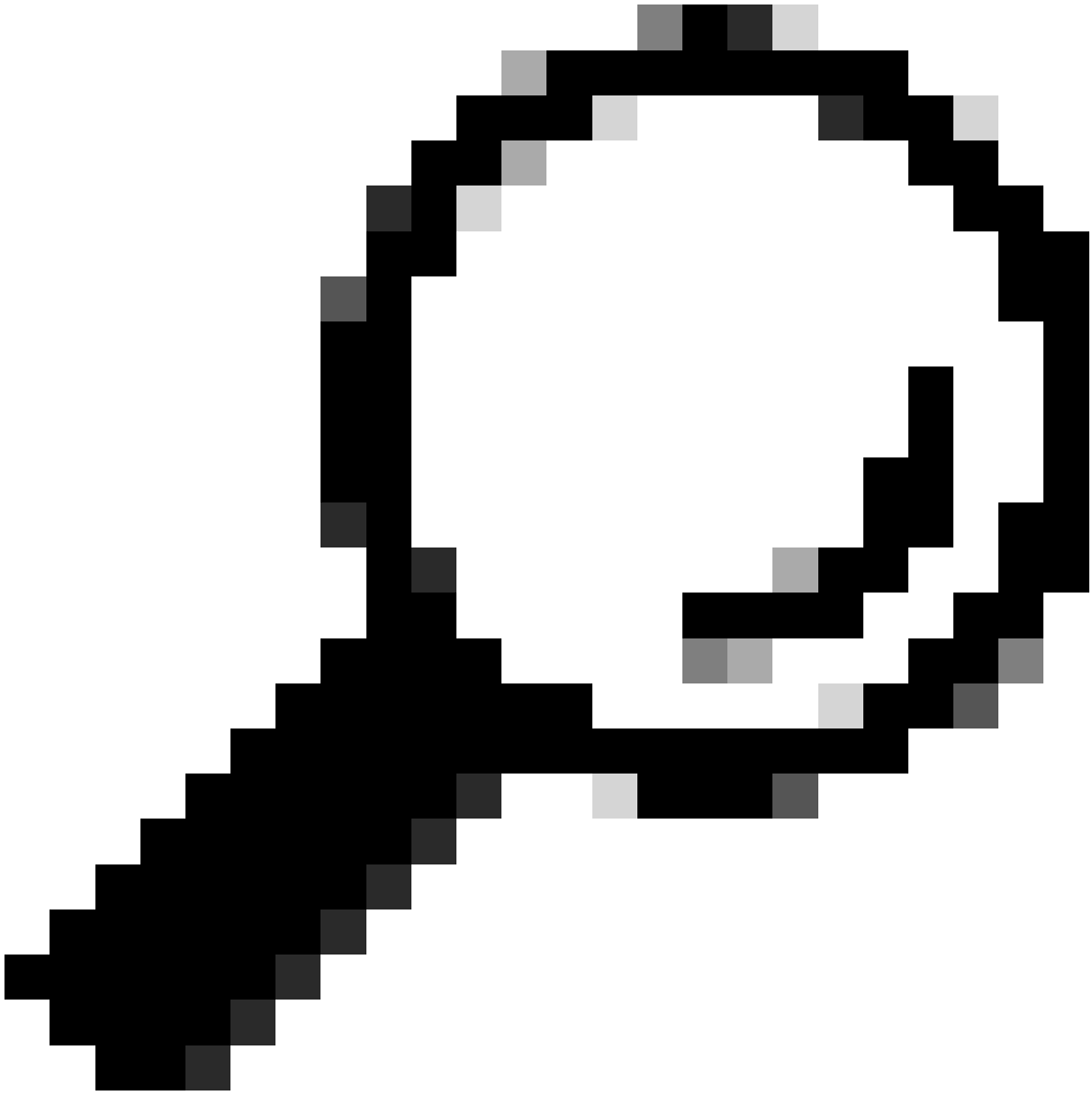
show controller fabric fia stats location <>
show controllers fabric fia drops ingress location <>
show controllers fabric fia drops egress location <>
show controller fabric fia status location <>
show controller pm vqi location <LC>
show controllers fabric vqi assignment location <>
show tech fabric
show_sm15_ltrace -s 2 fab_xbar | grep "sm15_pcie_read_fpoe"

Application

show ipv4 traffic brief location < >

Silent drops

We do not drop any packets without accounting, but several counters are incremented in the production setup, and the counter name may not be directly pointing to drop. In some cases, it's difficult to tell if the incoming packet is dropped or forwarded. So, using packet trace, we can watch the packet flow on the setup and validate whether or not the packet is being transmitted. the below is the sample packet trace output.



Tip: Embedded packet tracer, More info @ PRM Embedded Packet Trace. (https://xrdocs.io/asr9k//tutorials/xr-embedded-packet-tracer/)

CLI Commands Summary

Command Syntax	Description
clear packet-trace conditions all	Clears all buffered packet-trace conditions. Command is allowed only while packet tracing is inactive.
clear packet-trace counters all	Resets all packet-trace counters to zero.
packet-trace condition interface*interface*	Specify interfaces on which you expect to receive packets that you want to trace through the router.
packet-trace condition*n***offset***offset***value***value***mask***mask*	Specify set(s) of the Offset/Value/Mask that define the flow of interest.
packet-trace start	Start packet tracing.

Command Syntax	Description
packet-trace stop	Stop packet tracing.
show packet-trace description	See all counters registered with the packet tracer framework along with their descriptions.
show packet-trace status[**detail**]	See conditions buffered by the pkt_trace_master process running on the active RP and the packet tracer status (Active/Inactive). The detailed option of the command shows which processes are registered with the packet tracer framework on every card in the router. If packet tracer status is Active, output also shows which conditions were successfully programmed in data-path.
show packet-trace result	See the non-zero packet tracer counters.
show packet-trace result counter*name*[**source***source*] [**location***location*]	See the most recent 1023 increments of a specific packet-trace counter.

RP/0/RSP1/CPU0:ios#sh packet-trace results Tue Jan 24 19:28:56.151 UTC T: D - Drop counter; P - Pass counter Location | Source | Counter | T | Last-Attribute | Count ---------- ---------- ---------- - ---------- ---------- 0/0/CPU0 NP1 PACKET_MARKED P FortyGigE0_0_1_0 1522128420 0/0/CPU0 NP1 PACKET_ING_DROP D 2000908208 0/0/CPU0 NP1 PACKET_TO_FABRIC P 1522238547 0/0/CPU0 NP1 PACKET_TO_PUNT P 296246 0/0/CPU0 NP1 PACKET_INGR_TOP_LOOPBACK P 1000371630 0/0/CPU0 NP1 PACKET_INGR_TM_LOOPBACK P 1000375084 0/0/CPU0 spp-LIB ENTRY_COUNT P SPP PD Punt: stage1 299311 0/0/CPU0 NP1 PACKET_FROM_FABRIC P 1522238531 0/0/CPU0 NP1 PACKET_EGR_TOP_LOOPBACK P 1000371546 0/0/CPU0 NP1 PACKET_EGR_TM_LOOPBACK P 1000375020 0/0/CPU0 NP1 PACKET_TO_INTERFACE P FortyGigE0_0_1_0 1522241940

Triage flow :

First check the 'show drops, show drops all ongoing location all ' output multiple times to figure out the module/Code component for which the drops are being seen.

Once identified, component/module specific commands can be used to isolate the issue further.

show drops all ongoing location all → This gives real time drop info for NP/FIA/LPTS/CEF

```
RP/0/RP1/CPU0:R1# show drops all location 0/7/CPU0
Fri May 20 09:31:34.585 UTC
====================================
Checking for drops on 0/7/CPU0
====================================
show arp traffic:
[arp:ARP] IP Packet drop count for node 0/7/CPU0: 265
show cef drops:
[cef:0/7/CPU0] No route drops packets : 30536808
show spp node-counters:
```

```
[spp:pd_utility] Offload Drop: Interface Down: 1
[spp:port3/classify] Invalid: logged n dropped: 10
[spp:client/punt] client quota drop: 445639
show spp client detail:
[spp:ASR9K SPIO client stream ID 50, JID 253 (pid 7464)] Current:
0, Limit: 20000, Available: 0, Enqueued: 0, Drops: 445639
RP/0/RP1/CPU0:R1#
```

show drops

RP/0/RP1/CPU0:R1# show drops all location 0/7/CPU0 Fri May 20 09:31:34.585 UTC ========== Checking for drops on 0/7/CPU0 ========== show arp traffic: [arp:ARP] IP Packet drop count for node 0/7/CPU0: 265 show cef drops: [cef:0/7/CPU0] No route drops packets : 30536808 show spp node-counters: [spp:pd_utility] Offload Drop: Interface Down: 1 [spp:port3/classify] Invalid: logged n dropped: 10 [spp:client/punt] client quota drop: 445639 show spp client detail: [spp:ASR9K SPIO client stream ID 50, JID 253 (pid 7464)] Current: 0, Limit: 20000, Available: 0, Enqueued: 0, Drops: 445639 RP/0/RP1/CPU0:R1#

show pfm location all → Provides System related alarms like ASIC error , Punt_data_path_failed , etc

RP/0/RP0/CPU0:l51#show pfm location all Mon Apr 1 10:02:49.952 UTC node: node0_0_CPU0 ---------- CURRENT TIME: Apr 1 10:02:50 2024 PFM TOTAL: 0 EMERGENCY/ALERT(E/A): 0 CRITICAL(CR): 0 ERROR(ER): 0 ---------- ---------- Raised Time |S#|Fault Name |Sev|Proc_ID|Dev/Path Name |Handle ----------+--+----------+---+-------+----------+---------- Mon Jan 01 00:00:000|--|NONE |NO |0000000|NONE |0x00000000 node: node0_RP0_CPU0 ---------- CURRENT TIME: Apr 1 10:02:50 2024 PFM TOTAL: 0 EMERGENCY/ALERT(E/A): 0 CRITICAL(CR): 0 ERROR(ER): 0 ---------- ---------- Raised Time |S#|Fault Name |Sev|Proc_ID|Dev/Path Name |Handle ----------+--+----------+---+-------+----------+---------- Mon Jan 01 00:00:000|--|NONE |NO |0000000|NONE |0x00000000 RP/0/RP0/CPU0:l51#

"For us" packet drops

Check interface statistics

Check NP counters

Check SPP counters

Check netio counters

Check SRPM port statistics

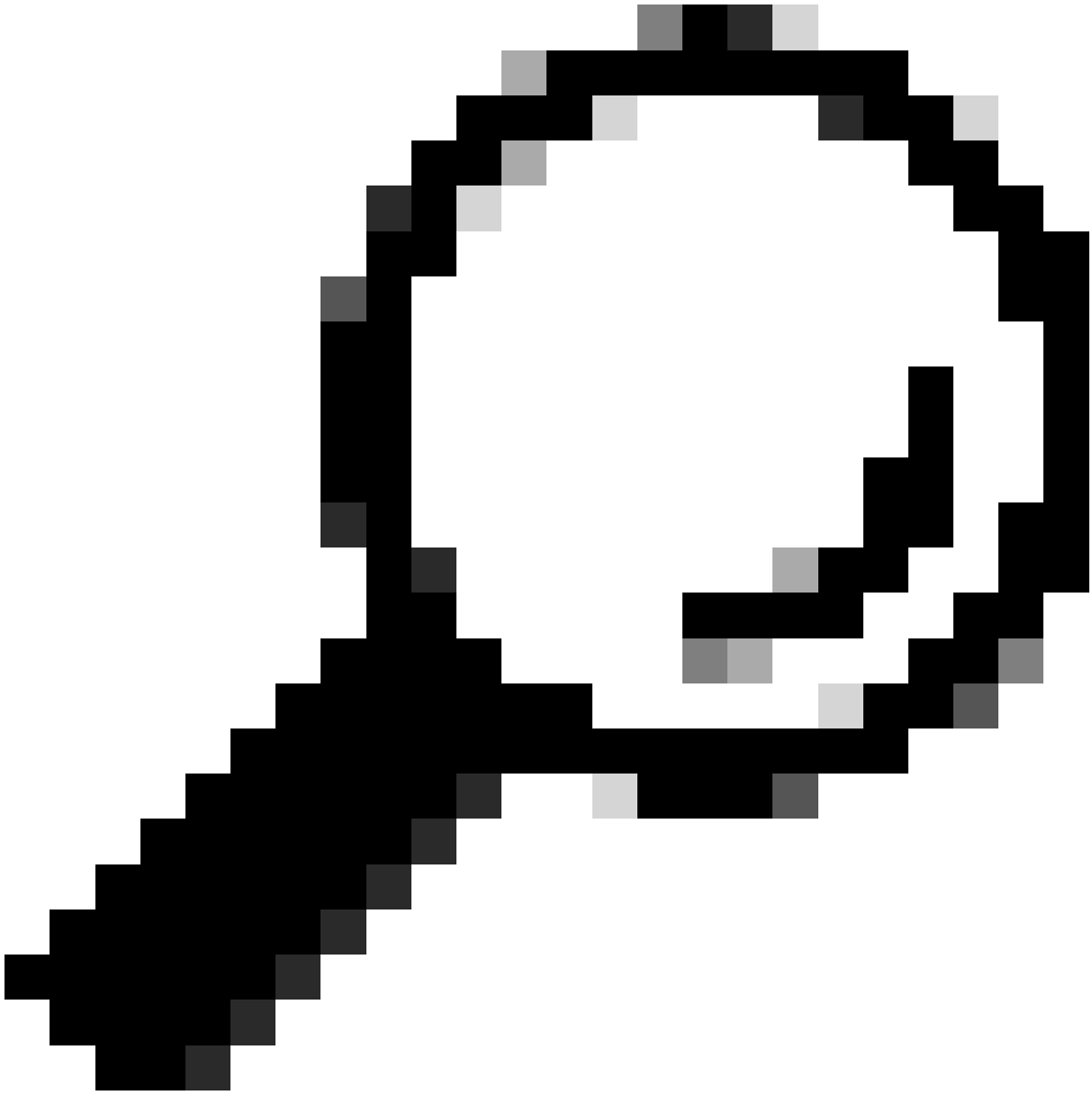
Check fabric statistics

Check Application level statistics

Transit packets drops

Check interface statistics
Check NP counters
Check fabric statistics

Injected traffic drops



Tip: Refer

Check Application level statistics
enable debug punt-inject l3/l2-packets <protocol> location <>
Check netio counters
Check SPP counters
Check SRPM port statistics
Check NP counters
Check interface statistic

Other Useful links :

Feedback