Troubleshoot ACI Intra-Fabric Forwarding - L3 Forwarding: Two Endpoints in different BDs

Contents

Introduction

This document describes steps to understand and troubleshoot an ACI L3 Forwarding scenario.

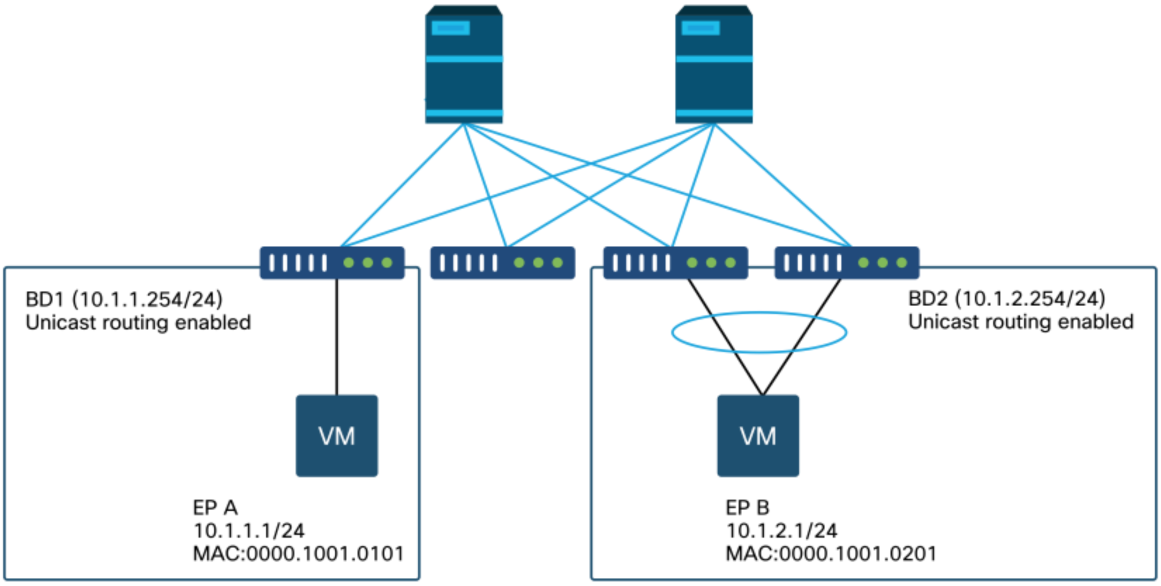
Background Information

The material from this document was extracted from the Troubleshooting Cisco Application Centric Infrastructure, Second Edition book, specifically the **Intra-Fabric fowarding - L3 forwarding: two endpoints in different BDs** chapter.

L3 forwarding: two endpoints in different BDs

This chapter explains a troubleshooting example where endpoints in different bridge domains can't talk to each other. This would be a flow routed by ACI fabric. Figure 1 illustrates the topology.

Endpoints in different bridge domains



High level troubleshooting workflow

The following are typical troubleshooting steps and verification commands:

First checks — validate programming

- BD pervasive gateway should be pushed to leaf nodes.
- Route to the destination BD subnet should be pushed to leaf nodes.
- ARP for the default gateway of the hosts should be resolved.

Second checks — validate learning and table entries via CLI on leaf nodes

- Check the source leaf and destination leaf nodes learn the endpoint and whether it learns the destination endpoint: Endpoint table — 'show endpoint'.TEP destination — 'show interface tunnel <x>'.Locating TEP destination in 'show ip route <TEP address> vrf overlay-1' command.
- Check spine nodes learns the endpoint: 'show coop internal info'.

Third checks — grab a packet and analyze the forwarding decisions

- With ELAM (ELAM Assistant or CLI) to validate the frame is there.
- Or with fTriage to track the flow.

Troubleshooting workflow for known endpoints

Check the pervasive gateway of the BD

In this example, the following source and destination endpoints will be used:

- EP A 10.1.1.1 under leaf1.
- EP B 10.1.2.1 under VPC pair leaf3 and leaf4.

Following pervasive gateways should be seen:

- 10.1.1.254/24 for BD1 gateway on leaf1.
- 10.1.2.254/24 for BD2 gateway on leaf3 and leaf4.

This can be checked using: 'show ip interface vrf <vrf name>' on the leaf nodes.

leaf1:

```
leaf1# show ip interface vrf Prod:VRF1
IP Interface Status for VRF "Prod:VRF1"
vlan7, Interface status: protocol-up/link-up/admin-up, iod: 106, mode: pervasive
 IP address: 10.1.1.254, IP subnet: 10.1.1.0/24
 IP broadcast address: 255.255.255.255
 IP primary address route-preference: 0, tag: 0
```

leaf3 and 4:

```
leaf3# show ip interface  vrf Prod:VRF1
IP Interface Status for VRF "Prod:VRF1"
vlan1, Interface status: protocol-up/link-up/admin-up, iod: 159, mode: pervasive
 IP address: 10.1.2.254, IP subnet: 10.1.2.0/24
 IP broadcast address: 255.255.255.255
 IP primary address route-preference: 0, tag: 0

leaf4# show ip interface  vrf Prod:VRF1
IP Interface Status for VRF "Prod:VRF1"
vlan132, Interface status: protocol-up/link-up/admin-up, iod: 159, mode: pervasive
 IP address: 10.1.2.254, IP subnet: 10.1.2.0/24
 IP broadcast address: 255.255.255.255
 IP primary address route-preference: 0, tag: 0
```

Note that leaf3 and leaf4 have the same pervasive gateway address, but different VLAN encapsulation for the SVI will likely be seen.

- leaf3 uses VLAN 1.
- leaf4 uses VLAN 132.

This is expected as VLAN 1 or VLAN 132 is local VLAN on the leaf.

If the pervasive gateway IP address is not pushed to the leaf, verify in APIC GUI that there are no faults that would prevent the VLAN from being deployed.

Checking routing table on the leaf

Leaf1 does not have any endpoint in subnet 10.1.2.0/24, however it must have the route to that subnet in order to reach it:

```
leaf1# show ip route 10.1.2.0/24 vrf Prod:VRF1
IP Route Table for VRF "Prod:VRF1"
'*' denotes best ucast next-hop
'**' denotes best mcast next-hop
```

```
'[x/y]' denotes [preference/metric]
'%<string>' in via output denotes VRF <string>

10.1.2.0/24, ubest/mbest: 1/0, attached, direct, pervasive
  *via 10.0.8.65%overlay-1, [1/0], 00:22:37, static, tag 4294967294
       recursive next hop: 10.0.8.65/32%overlay-1
```

Note that the route flagged with 'pervasive' and 'direct' have next-hop of 10.0.8.65. This is the anycast-v4 loopback address which exists on all spines.

```
leaf1# show isis  dteps  vrf overlay-1 | egrep 10.0.8.65
10.0.8.65          SPINE   N/A             PHYSICAL,PROXY-ACAST-V4
```

Similarly, leaf3 and leaf4 should have route for 10.1.1.0/24.

```
leaf3# show ip route 10.1.1.1 vrf Prod:VRF1
IP Route Table for VRF "Prod:VRF1"
'*' denotes best ucast next-hop
'**' denotes best mcast next-hop
'[x/y]' denotes [preference/metric]
'%<string>' in via output denotes VRF <string>

10.1.1.0/24, ubest/mbest: 1/0, attached, direct, pervasive
  *via 10.0.8.65%overlay-1, [1/0], 00:30:25, static, tag 4294967294
       recursive next hop: 10.0.8.65/32%overlay-1
```

If these routes are missing, it is likely because there is no contract between an EPG in BD1 and an EPG in BD2. If there is no local endpoint in BD1 under a leaf, the BD1 pervasive gateway doesn't get pushed to the leaf. If there is a local endpoint in an EPG that has a contract with another EPG in BD1, the BD1 subnet gets learned on the leaf.

ARP resolution for the default gateway IP

Since the leaf where a local endpoint resides should have a pervasive gateway, ARP requests for the pervasive gateway should always be resolved by the local leaf. This can be checked on the local leaf using the following command:

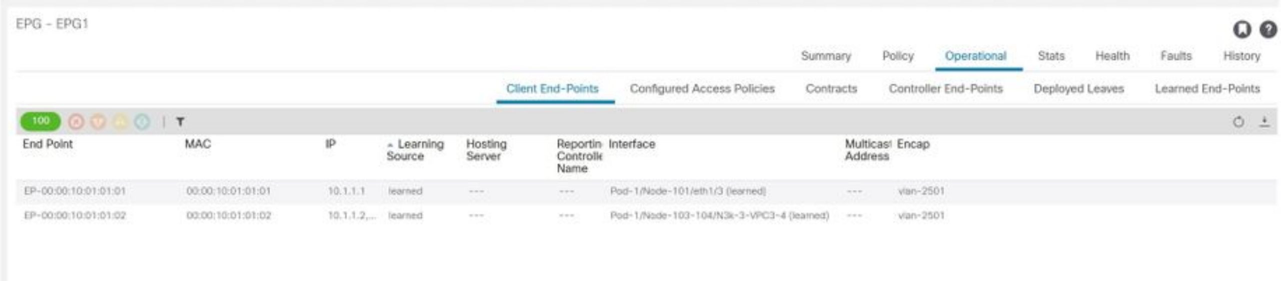
```
leaf1# show ip arp internal event-history event  | egrep 10.1.1.1
  [116] TID 26571:arp_handle_arp_request:6135: log_collect_arp_pkt; sip = 10.1.1.1; dip =
10.1.1.254;interface = Vlan7; phy_inteface = Ethernet1/3; flood = 0; Info = Sent ARP response.
   [116] TID 26571:arp_process_receive_packet_msg:8384: log_collect_arp_pkt; sip = 10.1.1.1; dip
= 10.1.1.254;interface = Vlan7; phy_interface = Ethernet1/3;Info = Received arp request
```

Ingress leaf source IP and MAC endpoint learning

In case of Layer 3 forwarding, ACI will perform Layer 3 source IP learning and destination IP lookup. Learned IP addresses are scoped to the VRF.

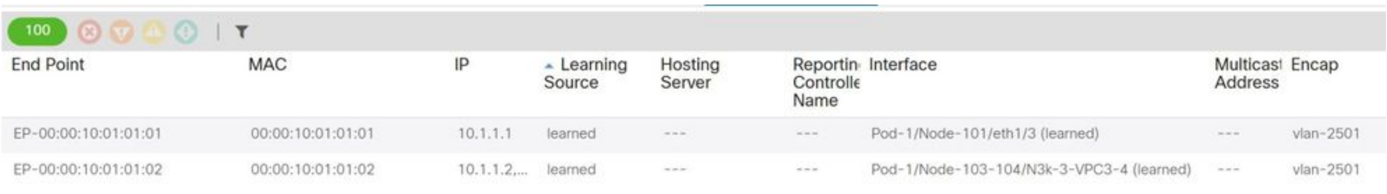
This can be checked on the GUI in an EPG's 'operational' tab. Note that here the IP and the MAC are both learned.

EPG Operational End-Points



End Point	MAC	IP	Learning Source	Hosting Server	Reportin Controlle Name	Interface	Multicast Address	Encap
EP-00:00:10:01:01:01	00:00:10:01:01:01	10.1.1.1	learned	---	---	Pod-1/Node-101/eth1/3 (learned)	---	vlan-2501
EP-00:00:10:01:01:02	00:00:10:01:01:02	10.1.1.2,...	learned	---	---	Pod-1/Node-103-104/N3k-3-VPC3-4 (learned)	---	vlan-2501

EPG Operational End-Points — detail



End Point	MAC	IP	Learning Source	Hosting Server	Reportin Controlle Name	Interface	Multicast Address	Encap
EP-00:00:10:01:01:01	00:00:10:01:01:01	10.1.1.1	learned	---	---	Pod-1/Node-101/eth1/3 (learned)	---	vlan-2501
EP-00:00:10:01:01:02	00:00:10:01:01:02	10.1.1.2,...	learned	---	---	Pod-1/Node-103-104/N3k-3-VPC3-4 (learned)	---	vlan-2501

Check local endpoint is learned on the local leaf. Here check on leaf1 that IP 10.1.1.1 is learned:

```
leaf1# show endpoint ip 10.1.1.1
Legend:
s - arp              H - vtep             V - vpc-attached     p - peer-aged
R - peer-attached-rl B - bounce           S - static           M - span
D - bounce-to-proxy  O - peer-attached    a - local-aged       m - svc-mgr
L - local            E - shared-service
+------------------------------------+---------------+-----------------+--------------+----------
---+
    VLAN/                           Encap           MAC Address       MAC Info/       Interface
    Domain                          VLAN            IP Address        IP Info
+------------------------------------+---------------+-----------------+--------------+----------
---+
46                                        vlan-2501    0000.1001.0101 L
eth1/3
Prod:VRF1                                 vlan-2501          10.1.1.1 L
                     eth1/3
```

As shown above, the endpoint content is:

- BD (internal VLAN for BD is 46) with VLAN encapsulation of the EPG (vlan-2501) and the MAC address learned on eth1/3
- VRF (Prod:VRF1) with the IP 10.1.1.1

This can be understood as equivalent to an ARP entry in a traditional network. ACI does not store ARP info in an ARP table for endpoints. Endpoints are only visible in the endpoint table.

The ARP table on a leaf is only used for L3Out next-hops.

```
leaf1# show ip arp vrf Prod:VRF1
Flags: * - Adjacencies learnt on non-active FHRP router
      + - Adjacencies synced via CFSoE
      # - Adjacencies Throttled for Glean
      D - Static Adjacencies attached to down interface    IP ARP Table for context Prod:VRF1
Total number of entries: 0
Address         Age       MAC Address     Interface
<NO ENTRY >
```

Ingress leaf destination IP lookup — known remote endpoint

Assuming the destination IP is known (known unicast), below is the 'show endpoint' output for destination IP 10.1.2.1. That is a remote learn since it does not reside on leaf1, specifically pointing to the tunnel interface where it is learned locally (tunnel 4).

Remote endpoints only contain either the IP or the MAC, never both in the same entry. MAC address and IP address in the same endpoint happens only when the endpoint is locally learned.

```
leaf1# show endpoint ip 10.1.2.1
Legend:
s - arp              H - vtep             V - vpc-attached     p - peer-aged
R - peer-attached-rl B - bounce           S - static           M - span
D - bounce-to-proxy  O - peer-attached    a - local-aged       m - svc-mgr
L - local            E - shared-service
+-----------------------------------+---------------+-----------------+--------------+----------
---+
    VLAN/                           Encap           MAC Address       MAC Info/       Interface
    Domain                          VLAN            IP Address        IP Info
+-----------------------------------+---------------+-----------------+--------------+----------
---+
Prod:VRF1                                                    10.1.2.1 p
tunnel4

leaf1# show interface tunnel 4
Tunnel4 is up
  MTU 9000 bytes, BW 0 Kbit
  Transport protocol is in VRF "overlay-1"
  Tunnel protocol/transport is ivxlan
  Tunnel source 10.0.88.95/32 (lo0)
  Tunnel destination 10.0.96.66
  Last clearing of "show interface" counters never
  Tx
  0 packets output, 1 minute output rate 0 packets/sec
  Rx
  0 packets input, 1 minute input rate 0 packets/sec
```

The destination TEP is the anycast TEP of the leaf3 and 4 VPC pair and is learned via uplinks to spine.

```
leaf1# show ip route 10.0.96.66 vrf overlay-1
IP Route Table for VRF "overlay-1"
'*' denotes best ucast next-hop
'**' denotes best mcast next-hop
'[x/y]' denotes [preference/metric]
'%<string>' in via output denotes VRF <string>

10.0.96.66/32, ubest/mbest: 4/0
  *via 10.0.88.65, eth1/49.10, [115/3], 02w06d, isis-isis_infra, isis-l1-int
  *via 10.0.128.64, eth1/51.8, [115/3], 02w06d, isis-isis_infra, isis-l1-int
  *via 10.0.88.64, eth1/52.126, [115/3], 02w06d, isis-isis_infra, isis-l1-int
  *via 10.0.88.94, eth1/50.128, [115/3], 02w06d, isis-isis_infra, isis-l1-int
```

Additional endpoint information for IP 10.1.2.1 can be collected using the 'show system internal epm endpoint ip <ip>' command.

```
leaf1# show system internal epm endpoint ip 10.1.2.1
MAC : 0000.0000.0000 ::: Num IPs : 1
IP# 0 : 10.1.2.1 ::: IP# 0 flags :  ::: l3-sw-hit: No
Vlan id : 0 ::: Vlan vnid : 0 ::: VRF name : Prod:VRF1
BD vnid : 0 ::: VRF vnid : 2097154
```

```
Phy If : 0 ::: Tunnel If : 0x18010004
Interface : Tunnel4
Flags : 0x80004420 ::: sclass : 32771 ::: Ref count : 3
EP Create Timestamp : 10/01/2019 13:53:16.228319
EP Update Timestamp : 10/01/2019 14:04:40.757229
EP Flags : peer-aged|IP|sclass|timer|
::::
```

In that output check:

- VRF VNID is populated — this is the VNID used to encapsulate the frame in VXLAN to the fabric.
- MAC address is 0000.0000.0000 as MAC address is never populated on a remote IP entry.
- BD VNID is unknown as for routed frames, the ingress leaf acts as the router and does a MAC rewrite. This means the remote leaf will not have visibility into the BD of the destination, only the VRF.

The frame will now be encapsulated in a VXLAN frame going to the remote TEP 10.0.96.66 with a VXLAN id of 2097154 which is the VNID of the VRF. It will be routed in the overlay-1 routing table (IS-IS route) and will reach the destination TEP. Here it will reach either leaf3 or leaf4 as 10.0.96.66 is the anycast TEP address of the leaf3 and leaf4 VPC pair.

Source IP learning on egress leaf

The outputs here are taken from leaf3 but would be similar on leaf4. When packets reach leaf3 (destination leaf and owner of the TEP), leaf will learn source IP of the packet in the VRF.

```
leaf3# show endpoint ip 10.1.1.1
Legend:
s - arp              H - vtep             V - vpc-attached     p - peer-aged
R - peer-attached-rl B - bounce           S - static           M - span
D - bounce-to-proxy  O - peer-attached    a - local-aged       m - svc-mgr
L - local            E - shared-service
+------------------------------------+---------------+------------------+---------------+----------
---+
    VLAN/                            Encap           MAC Address       MAC Info/       Interface
    Domain                           VLAN            IP Address        IP Info
+------------------------------------+---------------+------------------+---------------+----------
---+
Prod:VRF1                                                    10.1.1.1 p
tunnel26

leaf3# show interface tunnel 26
Tunnel26 is up
  MTU 9000 bytes, BW 0 Kbit
  Transport protocol is in VRF "overlay-1"
  Tunnel protocol/transport is ivxlan
  Tunnel source 10.0.88.91/32 (lo0)
  Tunnel destination 10.0.88.95
  Last clearing of "show interface" counters never
  Tx
  0 packets output, 1 minute output rate 0 packets/sec
  Rx
  0 packets input, 1 minute input rate 0 packets/sec
```

The destination TEP 10.0.88.95 is the TEP address of leaf1 and is learned via all uplinks to spine.

Destination IP lookup on egress leaf

The last step is for the egress leaf to lookup the destination IP. Look at the endpoint table for 10.1.2.1.

This gives the following information:

- The egress leaf knows the destination 10.1.2.1 (similar to a /32 host route in routing table) and the route is learned in correct VRF.
- The egress leaf knows the MAC 0000.1001.0201 (endpoint info).
- The egress leaf knows the traffic destined to 10.1.2.1 must be encapsulated in vlan-2502 and send out on port-channel 1 (po1).

```
leaf3# show endpoint ip 10.1.2.1
Legend:
s - arp              H - vtep             V - vpc-attached     p - peer-aged
R - peer-attached-rl B - bounce           S - static           M - span
D - bounce-to-proxy  O - peer-attached    a - local-aged       m - svc-mgr
L - local            E - shared-service
+-----------------------------------+---------------+-----------------+--------------+----------
---+
    VLAN/                           Encap           MAC Address       MAC Info/       Interface
    Domain                          VLAN            IP Address        IP Info
+-----------------------------------+---------------+-----------------+--------------+----------
---+
2                                         vlan-2502    0000.1001.0201 LpV
po1
Prod:VRF1                                 vlan-2502          10.1.2.1 LpV
po1
```

fTriage to follow the datapath

Use fTriage in the APIC to follow the datapath flow. Remember, fTriage relies on ELAM, so it needs real data flow. This allows confirmation of the full datapath, with confirmation that the packet exits the fabric on leaf3 port 1/16.

```
apic1# ftriage route -ii LEAF:101 -sip 10.1.1.1 -dip 10.1.2.1
fTriage Status: {"dbgFtriage": {"attributes": {"operState": "InProgress", "pid": "6888",
"apicId": "1", "id": "0"}}}
Starting ftriage
Log file name for the current run is: ftlog_2019-10-01-21-17-54-175.txt
2019-10-01 21:17:54,179 INFO     /controller/bin/ftriage route -ii LEAF:101 -sip 10.1.1.1 -dip
10.1.2.1
2019-10-01 21:18:18,149 INFO     ftriage:     main:1165 Invoking ftriage with default password
and default username: apic#fallback\\admin
2019-10-01 21:18:39,194 INFO     ftriage:     main:839  L3 packet Seen on bdsol-aci32-leaf1
Ingress: Eth1/3 Egress: Eth1/51 Vnid: 2097154
2019-10-01 21:18:39,413 INFO     ftriage:     main:242  ingress encap string vlan-2501
2019-10-01 21:18:39,419 INFO     ftriage:     main:271  Building ingress BD(s), Ctx
2019-10-01 21:18:41,240 INFO     ftriage:     main:294  Ingress BD(s) Prod:BD1
2019-10-01 21:18:41,240 INFO     ftriage:     main:301  Ingress Ctx: Prod:VRF1
2019-10-01 21:18:41,349 INFO     ftriage:   pktrec:490  bdsol-aci32-leaf1: Collecting transient
losses snapshot for LC module: 1
2019-10-01 21:19:05,747 INFO     ftriage:     main:933  SIP 10.1.1.1 DIP 10.1.2.1
2019-10-01 21:19:05,749 INFO     ftriage:  unicast:973  bdsol-aci32-leaf1: <- is ingress node
2019-10-01 21:19:08,459 INFO     ftriage:  unicast:1215 bdsol-aci32-leaf1: Dst EP is remote
2019-10-01 21:19:09,984 INFO     ftriage:     misc:657  bdsol-aci32-leaf1:
DMAC(00:22:BD:F8:19:FF) same as RMAC(00:22:BD:F8:19:FF)
2019-10-01 21:19:09,984 INFO     ftriage:     misc:659  bdsol-aci32-leaf1: L3 packet getting
routed/bounced in SUG
```

```
2019-10-01 21:19:10,248 INFO     ftriage:     misc:657  bdsol-aci32-leaf1: Dst IP is present in
SUG L3 tbl
2019-10-01 21:19:10,689 INFO     ftriage:     misc:657  bdsol-aci32-leaf1: RwDMAC
DIPo(10.0.96.66) is one of dst TEPs ['10.0.96.66']
2019-10-01 21:20:56,148 INFO     ftriage:     main:622  Found peer-node bdsol-aci32-spine3 and
IF: Eth2/1 in candidate list
2019-10-01 21:21:01,245 INFO     ftriage:     node:643  bdsol-aci32-spine3: Extracted Internal-
port GPD Info for lc: 2
2019-10-01 21:21:01,245 INFO     ftriage:     fcls:4414 bdsol-aci32-spine3: LC trigger ELAM with
IFS: Eth2/1 Asic :0 Slice: 0 Srcid: 32
2019-10-01 21:21:33,894 INFO     ftriage:     main:839  L3 packet Seen on bdsol-aci32-spine3
Ingress: Eth2/1 Egress: LC-2/0 FC-22/0 Port-1 Vnid: 2097154
2019-10-01 21:21:33,895 INFO     ftriage:   pktrec:490  bdsol-aci32-spine3: Collecting transient
losses snapshot for LC module: 2
2019-10-01 21:21:54,487 INFO     ftriage:      fib:332  bdsol-aci32-spine3: Transit in spine
2019-10-01 21:22:01,568 INFO     ftriage:  unicast:1252 bdsol-aci32-spine3: Enter
dbg_sub_nexthop with Transit inst: ig infra: False glbs.dipo: 10.0.96.66
2019-10-01 21:22:01,682 INFO     ftriage:  unicast:1417 bdsol-aci32-spine3: EP is known in COOP
(DIPo = 10.0.96.66)
2019-10-01 21:22:05,713 INFO     ftriage:  unicast:1458 bdsol-aci32-spine3: Infra route
10.0.96.66 present in RIB
2019-10-01 21:22:05,713 INFO     ftriage:     node:1331 bdsol-aci32-spine3: Mapped LC interface:
LC-2/0 FC-22/0 Port-1 to FC interface: FC-22/0 LC-2/0 Port-1
2019-10-01 21:22:10,799 INFO     ftriage:     node:460  bdsol-aci32-spine3: Extracted GPD Info
for fc: 22
2019-10-01 21:22:10,799 INFO     ftriage:     fcls:5748 bdsol-aci32-spine3: FC trigger ELAM with
IFS: FC-22/0 LC-2/0 Port-1 Asic :0 Slice: 2 Srcid: 24
2019-10-01 21:22:29,322 INFO     ftriage:  unicast:1774 L3 packet Seen on FC of node: bdsol-
aci32-spine3  with Ingress: FC-22/0 LC-2/0 Port-1  Egress: FC-22/0 LC-2/0 Port-1 Vnid: 2097154
2019-10-01 21:22:29,322 INFO     ftriage:   pktrec:487  bdsol-aci32-spine3: Collecting transient
losses snapshot for FC module: 22
2019-10-01 21:22:31,571 INFO     ftriage:     node:1339 bdsol-aci32-spine3: Mapped FC interface:
FC-22/0 LC-2/0 Port-1 to LC interface: LC-2/0 FC-22/0 Port-1
2019-10-01 21:22:31,572 INFO     ftriage:  unicast:1474 bdsol-aci32-spine3: Capturing Spine
Transit pkt-type L3 packet on egress LC on Node: bdsol-aci32-spine3 IFS: LC-2/0 FC-22/0 Port-1
2019-10-01 21:22:31,991 INFO     ftriage:     fcls:4414 bdsol-aci32-spine3: LC trigger ELAM with
IFS: LC-2/0 FC-22/0 Port-1 Asic :0 Slice: 1 Srcid: 0
2019-10-01 21:22:48,952 INFO     ftriage:  unicast:1510 bdsol-aci32-spine3: L3 packet Spine
egress  Transit pkt Seen on bdsol-aci32-spine3 Ingress: LC-2/0 FC-22/0 Port-1 Egress: Eth2/3
Vnid: 2097154
2019-10-01 21:22:48,952 INFO     ftriage:   pktrec:490  bdsol-aci32-spine3: Collecting transient
losses snapshot for LC module: 2
2019-10-01 21:23:50,748 INFO     ftriage:     main:622  Found peer-node bdsol-aci32-leaf3 and
IF: Eth1/51 in candidate list
2019-10-01 21:24:05,313 INFO     ftriage:     main:839  L3 packet Seen on bdsol-aci32-leaf3
Ingress: Eth1/51 Egress: Eth1/12 (Po1) Vnid: 11365
2019-10-01 21:24:05,427 INFO     ftriage:   pktrec:490  bdsol-aci32-leaf3: Collecting transient
losses snapshot for LC module: 1
2019-10-01 21:24:24,369 INFO     ftriage:     nxos:1404 bdsol-aci32-leaf3: nxos matching rule
id:4326 scope:34 filter:65534
2019-10-01 21:24:25,698 INFO     ftriage:     main:522  Computed egress encap string vlan-2502
2019-10-01 21:24:25,704 INFO     ftriage:     main:313  Building egress BD(s), Ctx
2019-10-01 21:24:27,510 INFO     ftriage:     main:331  Egress Ctx Prod:VRF1
2019-10-01 21:24:27,510 INFO     ftriage:     main:332  Egress BD(s): Prod:BD2
2019-10-01 21:24:30,536 INFO     ftriage:  unicast:1252 bdsol-aci32-leaf3: Enter dbg_sub_nexthop
with Local inst: eg infra: False glbs.dipo: 10.0.96.66
2019-10-01 21:24:30,537 INFO     ftriage:  unicast:1257 bdsol-aci32-leaf3: dbg_sub_nexthop
invokes dbg_sub_eg for vip
2019-10-01 21:24:30,537 INFO     ftriage:  unicast:1784 bdsol-aci32-leaf3: <- is egress node
2019-10-01 21:24:30,684 INFO     ftriage:  unicast:1833 bdsol-aci32-leaf3: Dst EP is local
2019-10-01 21:24:30,685 INFO     ftriage:     misc:657  bdsol-aci32-leaf3: EP if(Po1) same as
egr if(Po1)
2019-10-01 21:24:30,943 INFO     ftriage:     misc:657  bdsol-aci32-leaf3: Dst IP is present in
SUG L3 tbl
```

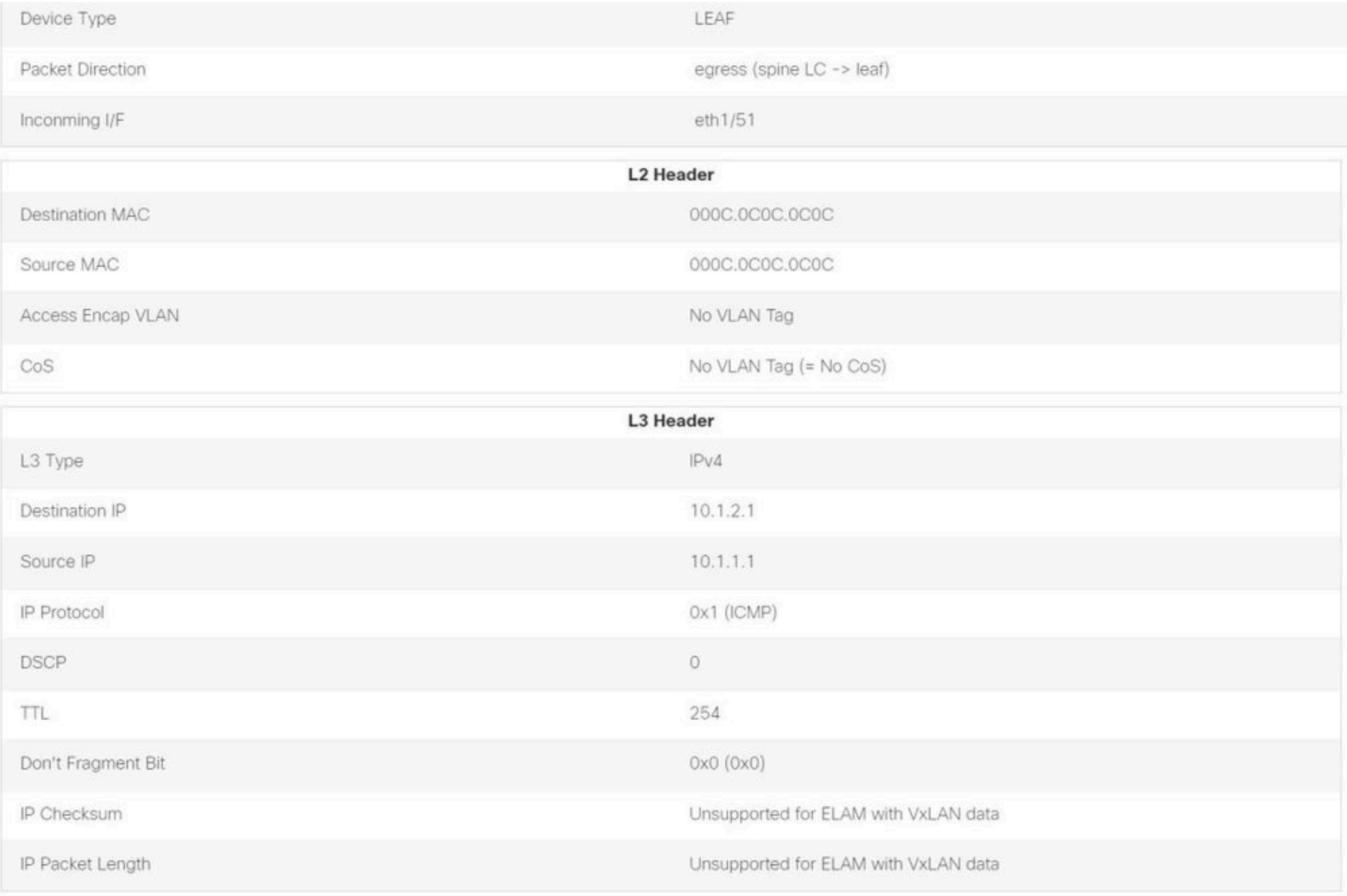
```
2019-10-01 21:24:31,242 INFO     ftriage:     misc:657  bdsol-aci32-leaf3: RW seg_id:11365 in
SUG same as EP segid:11365
2019-10-01 21:24:37,631 INFO     ftriage:     main:961  Packet is Exiting fabric with peer-
device: bdsol-aci32-n3k-3 and peer-port: Ethernet1/12
```

Packet capture on egress leaf using ELAM Assistant app

Below is the packet captured with the ELAM Assistant app on leaf3 coming from the spine. This shows that:

- The VNID from the outer Layer 4 information (VNID is 2097154).
- Outer L3 header source TEP and destination TEP.

ELAM Assistant — L3 flow egress leaf (part 1)

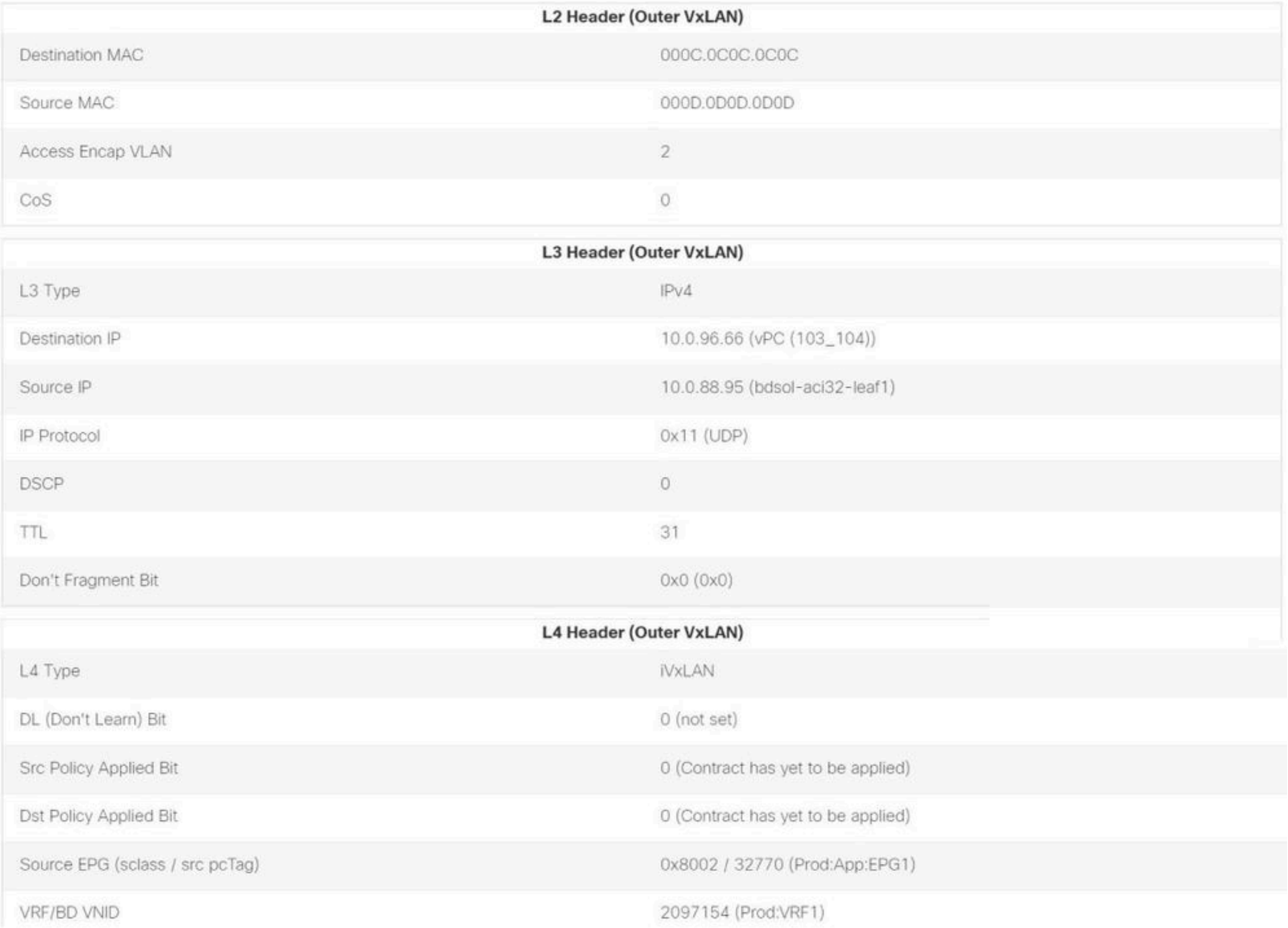


Device Type	LEAF
Packet Direction	egress (spine LC -> leaf)
Inconming I/F	eth1/51

L2 Header	
Destination MAC	000C.0C0C.0C0C
Source MAC	000C.0C0C.0C0C
Access Encap VLAN	No VLAN Tag
CoS	No VLAN Tag (= No CoS)

L3 Header	
L3 Type	IPv4
Destination IP	10.1.2.1
Source IP	10.1.1.1
IP Protocol	0x1 (ICMP)
DSCP	0
TTL	254
Don't Fragment Bit	0x0 (0x0)
IP Checksum	Unsupported for ELAM with VxLAN data
IP Packet Length	Unsupported for ELAM with VxLAN data

ELAM Assistant — L3 flow egress leaf (part 2)



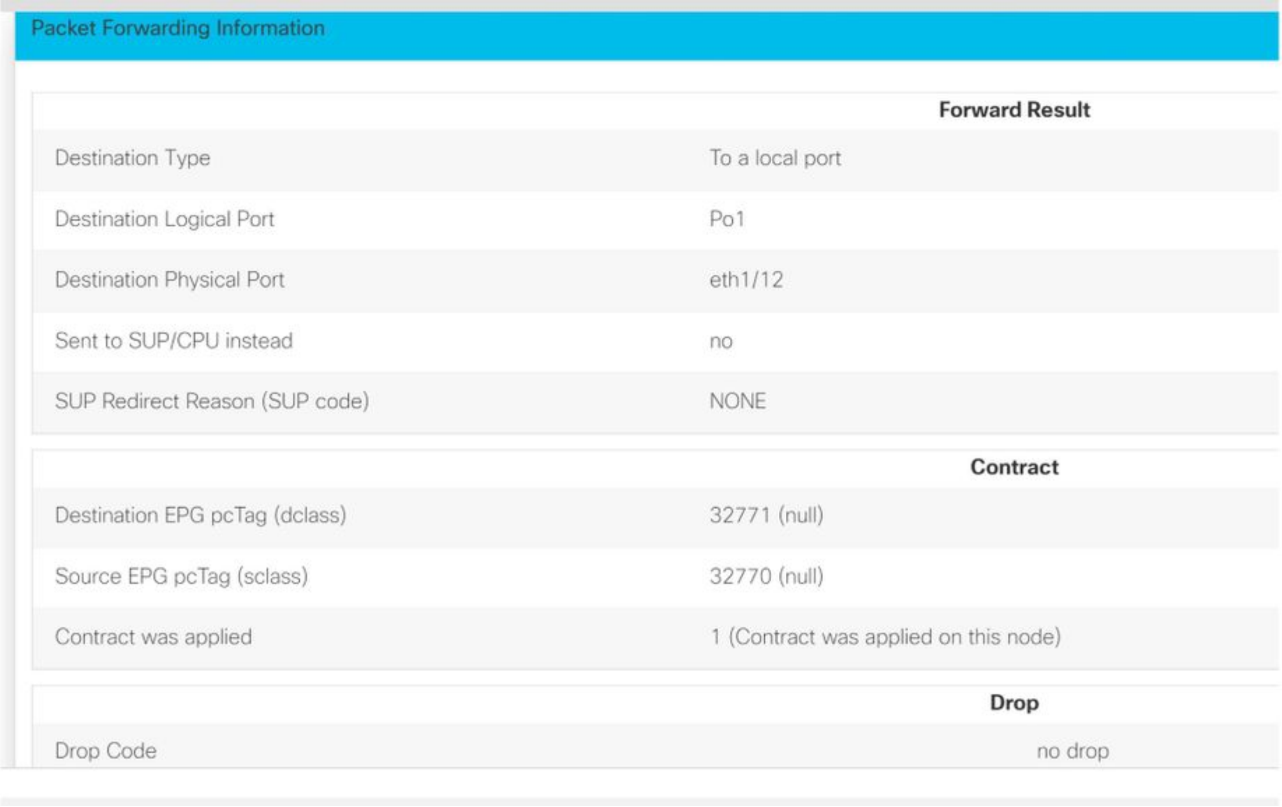
L2 Header (Outer VxLAN)	
Destination MAC	000C.0C0C.0C0C
Source MAC	000D.0D0D.0D0D
Access Encap VLAN	2
CoS	0

L3 Header (Outer VxLAN)	
L3 Type	IPv4
Destination IP	10.0.96.66 (vPC (103_104))
Source IP	10.0.88.95 (bdsol-aci32-leaf1)
IP Protocol	0x11 (UDP)
DSCP	0
TTL	31
Don't Fragment Bit	0x0 (0x0)

L4 Header (Outer VxLAN)	
L4 Type	iVxLAN
DL (Don't Learn) Bit	0 (not set)
Src Policy Applied Bit	0 (Contract has yet to be applied)
Dst Policy Applied Bit	0 (Contract has yet to be applied)
Source EPG (sclass / src pcTag)	0x8002 / 32770 (Prod:App:EPG1)
VRF/BD VNID	2097154 (Prod:VRF1)

The Packet Forwarding Information section proves it got out on port-channel 1

ELAM Assistant — L3 egress leaf — Packet Forwarding Information



Packet Forwarding Information

	Forward Result
Destination Type	To a local port
Destination Logical Port	Po1
Destination Physical Port	eth1/12
Sent to SUP/CPU instead	no
SUP Redirect Reason (SUP code)	NONE

	Contract
Destination EPG pcTag (dclass)	32771 (null)
Source EPG pcTag (sclass)	32770 (null)
Contract was applied	1 (Contract was applied on this node)

	Drop
Drop Code	no drop

Troubleshooting workflow for unknown endpoints

This section shows what differs when the ingress leaf does not know the destination IP.

Ingress leaf destination IP lookup

The first step is to check if there is an endpoint learn for the destination IP.

```
leaf1# show endpoint ip 10.1.2.1
Legend:
s - arp              H - vtep             V - vpc-attached     p - peer-aged
R - peer-attached-rl B - bounce           S - static           M - span
D - bounce-to-proxy  O - peer-attached    a - local-aged       m - svc-mgr
L - local            E - shared-service
+-----------------------------------+---------------+-----------------+--------------+----------
--+
    VLAN/                               Encap           MAC Address       MAC Info/       Interface
    Domain                              VLAN            IP Address        IP Info
+-----------------------------------+---------------+-----------------+--------------+----------
--+
<NO ENTRY>
```

There is nothing in endpoint table for the destination, so next step is to check the routing table looking for the longest prefix match route to the destination:

```
leaf1# show ip route 10.1.2.1 vrf Prod:VRF1
IP Route Table for VRF "Prod:VRF1"
'*' denotes best ucast next-hop
'**' denotes best mcast next-hop
'[x/y]' denotes [preference/metric]
'%<string>' in via output denotes VRF <string>

10.1.2.0/24, ubest/mbest: 1/0, attached, direct, pervasive
  *via 10.0.8.65%overlay-1, [1/0], 01:40:18, static, tag 4294967294
       recursive next hop: 10.0.8.65/32%overlay-1
```

Falling on the /24 BD subnet 10.1.2.0/24 means the leaf will encapsulate the frame in VXLAN with destination TEP 10.0.8.65 (anycast-v4 on spine). The frame will use a VXLAN id which is the VRF VNID.

COOP lookup on spine — destination IP is known

The packet will reach one of the spines that does COOP lookup in the IP database. The source must be verified and the destination IP needs to be learned correctly from the COOP database.

To find an IP in the COOP database, the key is VRF VNID (2097154 in this example)

From the output below, there is confirmation that the COOP database has the entry for the source IP from TEP 10.0.88.95 (leaf1) correctly.

```
spine1# show coop internal info ip-db key 2097154 10.1.1.1
IP address : 10.1.1.1
Vrf : 2097154
Flags : 0
EP bd vnid : 15302583
EP mac :  00:00:10:01:01:01
Publisher Id : 10.0.88.95
Record timestamp : 10 01 2019 14:16:50 522482647
Publish timestamp : 10 01 2019 14:16:50 532239332
Seq No: 0
Remote publish timestamp: 01 01 1970 00:00:00 0
URIB Tunnel Info
Num tunnels : 1
      Tunnel address : 10.0.88.95
      Tunnel ref count : 1
```

The output below shows that the COOP database has the entry for the destination IP from TEP 10.0.96.66 (Anycast TEP of the leaf3 and 4 VPC pair) correctly

```
spine1# show coop internal info ip-db key 2097154 10.1.2.1
IP address : 10.1.2.1
Vrf : 2097154
Flags : 0
EP bd vnid : 15957974
EP mac :  00:00:10:01:02:01
Publisher Id : 10.0.88.90
Record timestamp : 10 01 2019 14:52:52 558812544
Publish timestamp : 10 01 2019 14:52:52 559479076
Seq No: 0
Remote publish timestamp: 01 01 1970 00:00:00 0
URIB Tunnel Info
Num tunnels : 1
      Tunnel address : 10.0.96.66
      Tunnel ref count : 1
```

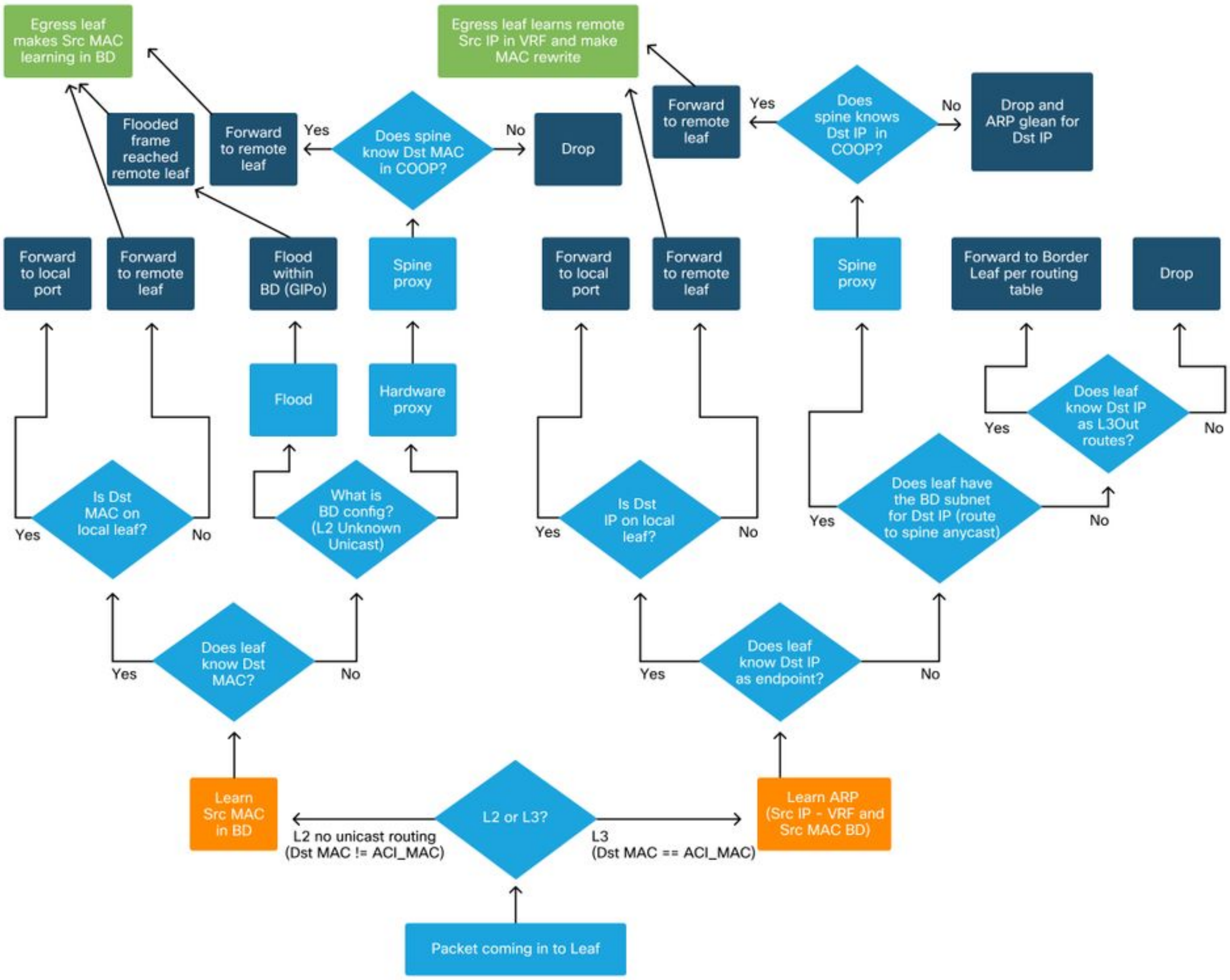
In the scenario here, COOP knows the destination IP so it will rewrite the destination IP of the outer IP header in the VXLAN packet to be 10.0.96.66 and then will send to leaf3 or leaf4 (depending on ECMP hashing). Note that the source IP of the VXLAN frame is not changed so it is still the leaf1 PTEP.

COOP lookup on spine - destination IP is unknown

In the case where the COOP entry for the destination IP is not populated (silent endpoint or aged out), the spine will generate an ARP glean to resolve it. For more information, refer to "Multi-Pod Forwarding" section.

ACI forwarding summary

The following drawing summarizes the ACI forwarding for Layer 2 and Layer 3 use case.


Packet coming in to Leaf
L2 or L3?
L2 no unicast routing
(Dst MAC != ACI_MAC)
L3
(Dst MAC == ACI_MAC)
Learn Src MAC in BD
Learn ARP (Src IP - VRF and Src MAC BD)
Does leaf know Dst MAC?
Yes
No
Is Dst MAC on local leaf?
Yes
No
What is BD config? (L2 Unknown Unicast)
Forward to local port
Forward to remote leaf
Flood
Hardware proxy
Flood within BD (GIPo)
Spine proxy
Does spine know Dst MAC in COOP?
Yes
No
Forward to remote leaf
Drop
Flooded frame reached remote leaf
Egress leaf makes Src MAC learning in BD
Does leaf know Dst IP as endpoint?
Yes
No
Is Dst IP on local leaf?
Yes
No
Forward to local port
Forward to remote leaf
Does leaf have the BD subnet for Dst IP (route to spine anycast)
Yes
No
Spine proxy
Does spine knows Dst IP in COOP?
Yes
No
Forward to remote leaf
Drop and ARP glean for Dst IP
Egress leaf learns remote Src IP in VRF and make MAC rewrite
Does leaf know Dst IP as L3Out routes?
Yes
No
Forward to Border Leaf per routing table
Drop