

The Cisco Live! logo features the word "CISCO" in a dark blue, sans-serif font, followed by "Live!" in a dark blue, cursive script font. The background of the entire image is a vibrant, multi-colored abstract pattern of overlapping, wavy lines and geometric shapes, transitioning from dark blue on the left to bright yellow and white in the center, and then to various shades of blue and green on the right.

CISCO *Live!*

Let's go

A Network Engineer's Blueprint for ACI Forwarding

Jessica Rueda, CCIE DC 65467, ACI Technical Leader, Customer Experience

Agenda

- What's Different About ACI Forwarding?
 - (iVXLAN, contracts, endpoint learning)
- Proxy Forwarding
- ACI Forwarding Tables
 - Endpoint tables, routing tables, hardware lookups
- Understanding the Configuration Options

Agenda

- Understanding the Tools
 - UI Tools
 - Elam
 - Ftriage
 - Span / ERSPAN
 - Flow Telemetry / NetFlow
- Debugging and Walking Through ACI Flows
 - (Routed, Bridged, BUM, Proxied)
- Troubleshooting Tips

Glossary of Acronyms

VxLAN packet acronyms

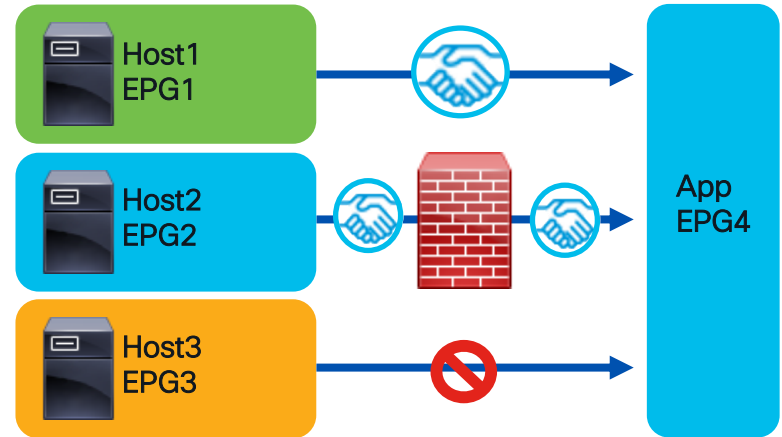
Acronyms	Definitions
ACI	Application Centric Infrastructure
APIC	Application Policy Infrastructure Controller
EP	Endpoint
EPG	Endpoint Group
BD	Bridge Domain
VRF	Virtual Routing and Forwarding
COOP	Council of Oracle Protocol
VxLAN	Virtual eXtensible LAN

Acronyms	Definitions
dXXXo	Outer Destination XXX (dIPo = Outer Destination IP)
sXXXo	Outer Source XXX (sIPo = Outer Source IP)
dXXXi	Inner Destination XXX (dIPi = Inner Destination IP)
sXXXi	Inner Source XXX (sIPi = Inner Source IP)
GIPO	Outer Multicast Group IP
VNID	Virtual Network Identifier

What's Different About ACI Forwarding?

What is “Application Centric”?

- Traditional networks use ACL’s to classify traffic
 - Usually based on L3 or L2 addresses
 - Makes security decisions (permit, deny, log, etc)
 - Makes forwarding decisions (policy based routing)
- ACI can classify traffic based on its EPG
- Traffic inherits the forwarding and security policy of the EPG



How is “Application Centric” Achieved?

Sources and Destinations Must be Classified into EPG’s

Endpoints

- Used by App EPG’s
- Represents the network identity of an end device
- Learned dynamically or configured statically

Policy-Prefixes

- Used by External EPG’s
- Classifies destination by longest prefix match
- Also used for shared-services
- Configured

PcTags

- The security ID of an EPG
- Used in contracts. Ex: Permit PcTag 1000 to PcTag 2000
- Sclass/dclass imply PcTag direction

Contracts

- Defines security and sometimes forwarding (pbr) policy between eggs
- Essentially an ACL between PcTags
- Consumer/Provider rather than src/dest

Vlan Types

※ PI-VLAN : Platform Independent VLAN

VLAN ID for external devices
(user configured value)

Internal ID on LEAF
(not shared across LEAFs)

For forwarding
(global value for entire fabric)

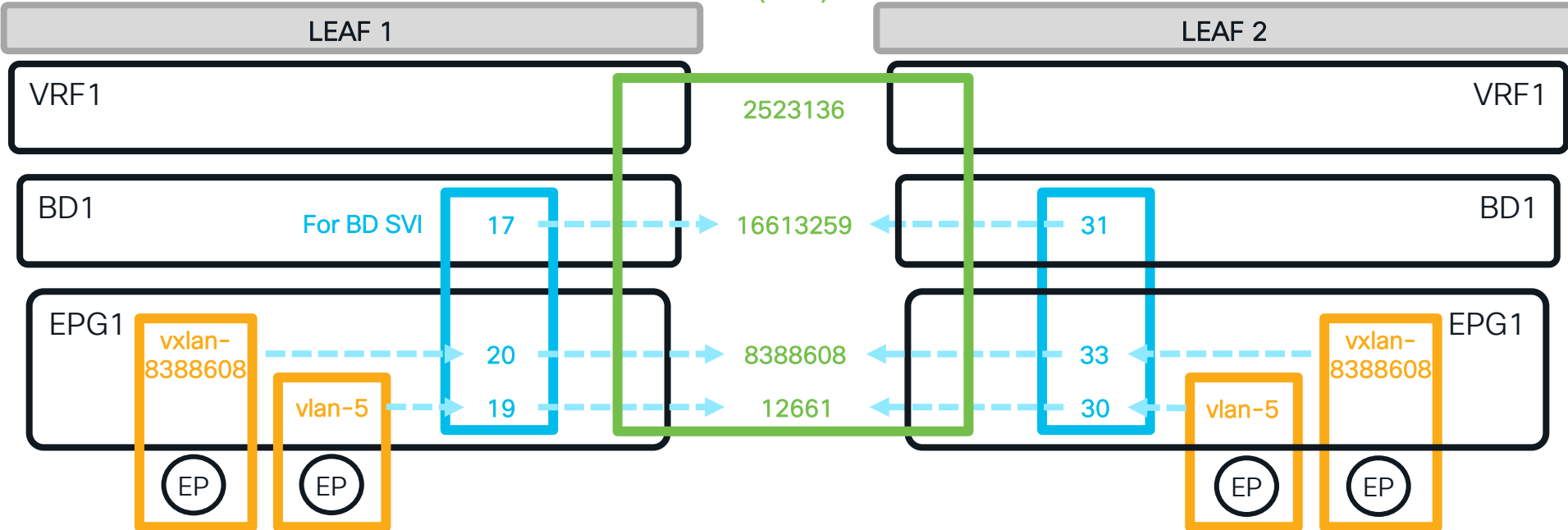
Access Encap VLAN

PI-VLAN

VxLAN ID
(VNID)

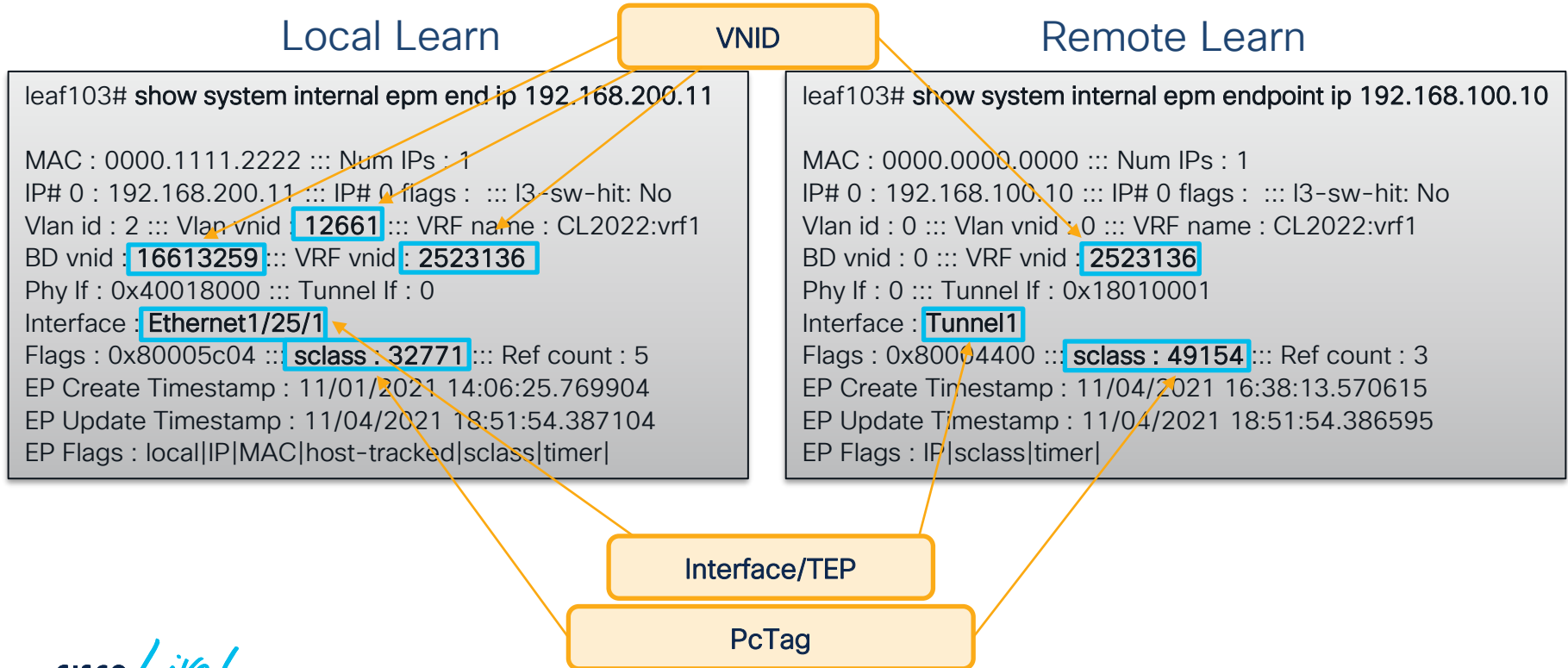
PI-VLAN

Access Encap VLAN



What is an Endpoint?

An Endpoint joins both forwarding and security policy



What is a TEP? (Tunnel Endpoint)

- IP addresses allocated for overlay communication
- VXLAN Traffic is sent to the TEP + VNID of destination

Most Common TEP Types

TEP Type	What is it?	What is it for?
Physical TEP (PTEP)	Unique Overlay IP Address for each individual Leaf/Spine	Non-vpc dataplane, I3out communication, apic-leaf comm, etc
VPC TEP (VTEP)	Unique Overlay IP Address for each VPC Pair	Traffic destined to endpoints that are connected behind VPC
Proxy TEP	Spine Anycast IP's used for proxy traffic	Leafs send to these TEPs when doing proxy forwarding

```
a-leaf101# show ip interface loopback0
IP Interface Status for VRF "overlay-1"
lo0, Interface status: protocol-up/link-up/admin-up, iod: 4, mode: ptep
```

What are Tunnels?

- Leafs/Spines Install Tunnel Interface to each known TEP
- Used for VXLAN Dataplane

How are Tunnels Learned?

Dataplane Learns →

```
leaf# moquery -c tunnelIf -f 'tunnel.If.id=="tunnel1"'  
  
id           : tunnel1  
dest        : 10.0.72.67  
idRequestorDn : sys/*/db-dtep/dtep-[10.0.72.67]
```

Through BGP
(I3out routes) →

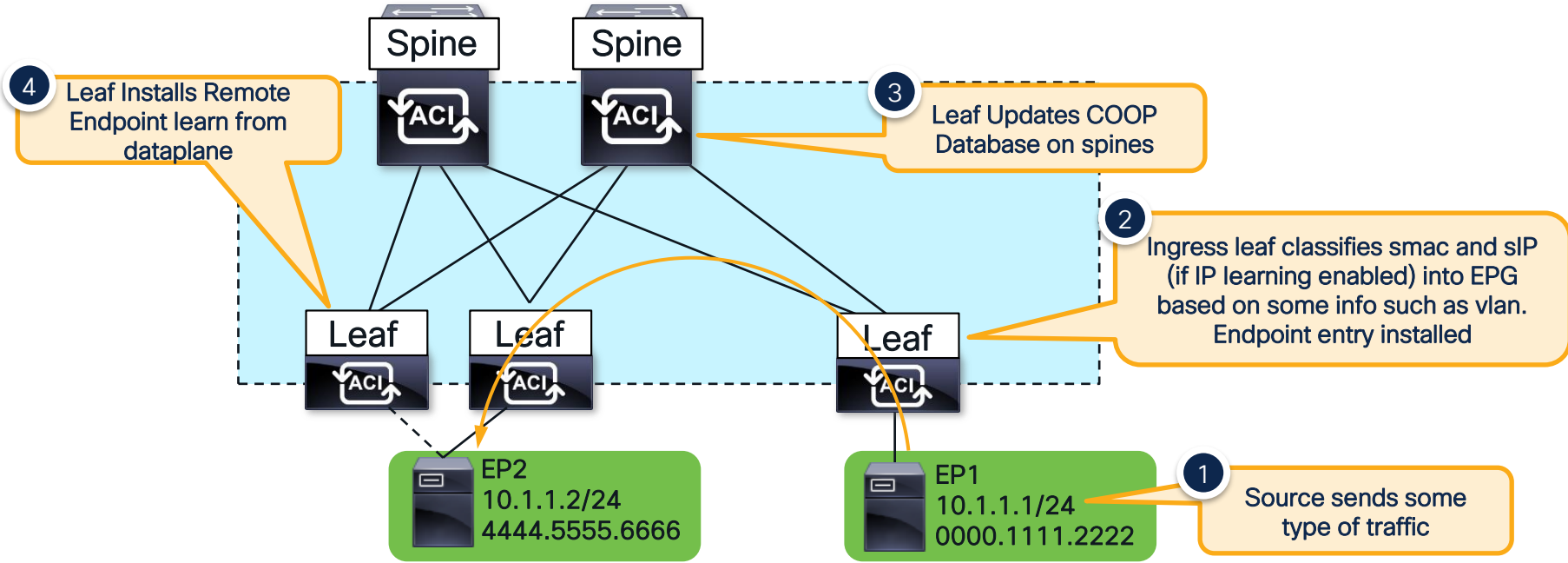
```
leaf# moquery -c tunnelIf -f 'tunnel.If.id=="tunnel2"'  
  
id           : tunnel1  
dest        : 10.0.72.64  
idRequestorDn : sys/bgp/*/db-dtep/dtep-[10.0.72.64]
```

Local POD ISIS
Database →

```
leaf# moquery -c tunnelIf -f 'tunnel.If.id=="tunnel3"'  
  
# tunnel.If  
id           : tunnel1  
dest        : 10.0.152.64  
idRequestorDn : sys/isis/*/lvl-l1/db-dtep/dtep-[10.0.152.64]
```

How is an Endpoint Learned?

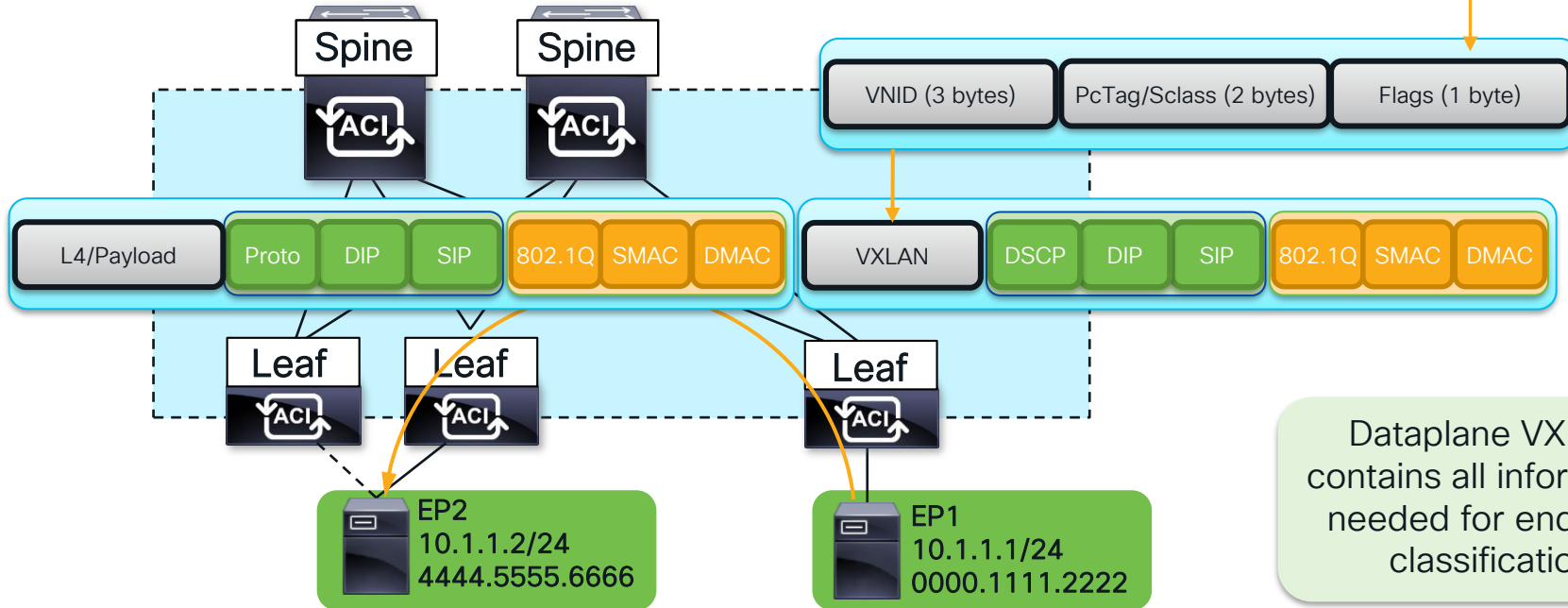
How does the Egress leaf classify traffic into the correct EPG?



Overlay iVXLAN

ACI uses VXLAN with some additional bits

Bit pos 4 – Source Policy Applied
Bit pos 5 – Destination Policy Applied
Bit pos 7 – Don't learn



Dataplane VXLAN contains all information needed for endpoint classification

How is Traffic Classified with no EP Learn?

In most of these cases, the pcTag is based on a policy-prefix lookup

There will be no endpoint learn in several cases

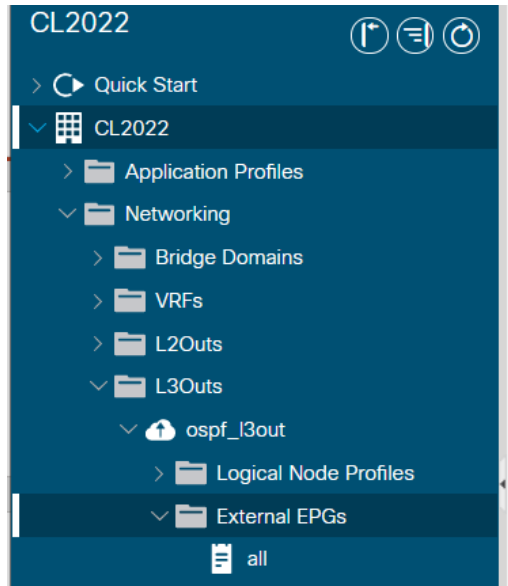
- Source/dest is behind an I3out
- Source/dest is in another vrf
- Endpoint learning is disabled by some option

If ingress leaf doesn't apply policy, egress leaf should (indicated via policy-applied bits in ivxlan header)

How is Traffic Classified with no EP Learn?

Destination Behind L3out

```
leaf101# vsh_lc -c "show forwarding route 10.99.99.100 platform vrf CL2022:vrf1"  
!  
Policy Prefix 10.99.99.0/24  
!  
vrf: 16(0x10), routed_if: 0x0 epc_class: 32772(0x8004)
```



External EPGs

External EPGs		
Name	Description	pcTag
all	10.99.99.0/24 Network	32772

Classification based on longest l3out policy prefix

How is Traffic Classified with no EP Learn?

Destination is unknown and is proxied

```
leaf101# show ip route 192.168.200.20 vrf CL2022:vrf1
192.168.200.0/24, ubest/mbest: 1/0, attached, direct, pervasive
  *via 10.0.176.66%overlay-1, [1/0], 4d05h, static, tag 4294967294
  recursive next hop: 10.0.176.66/32%overlay-1
```

“Pervasive” indicates this is a BD or EPG subnet (fvSubnet). Send to spine proxy-addr

```
leaf101# vsh_lc -c "show forwarding route 192.168.200.20 platform vrf CL2022:vrf1"
!
Policy Prefix 0.0.0.0/0
!
Vrf: 16(0x10), routed_if: 0x0 epc_class: 1(0x1)
```

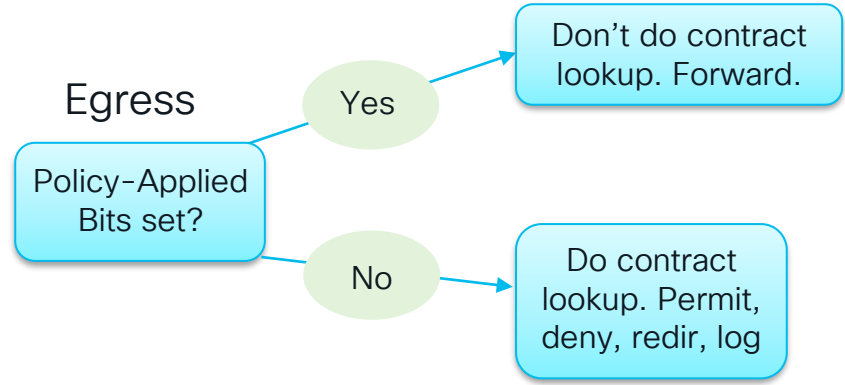
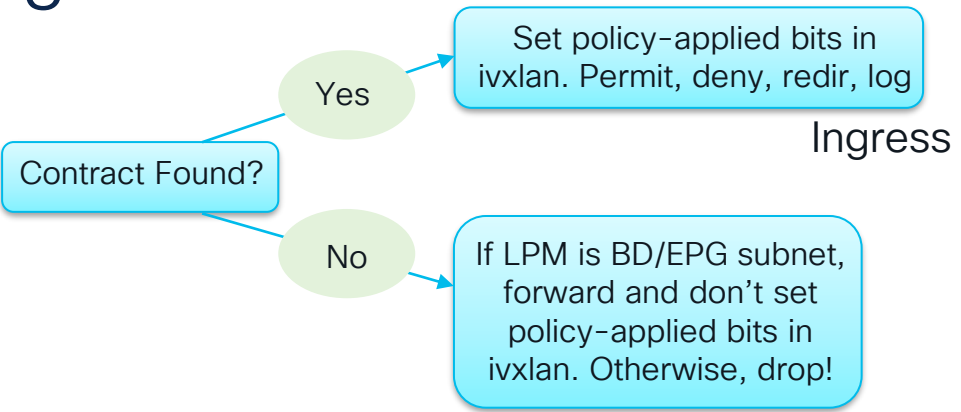
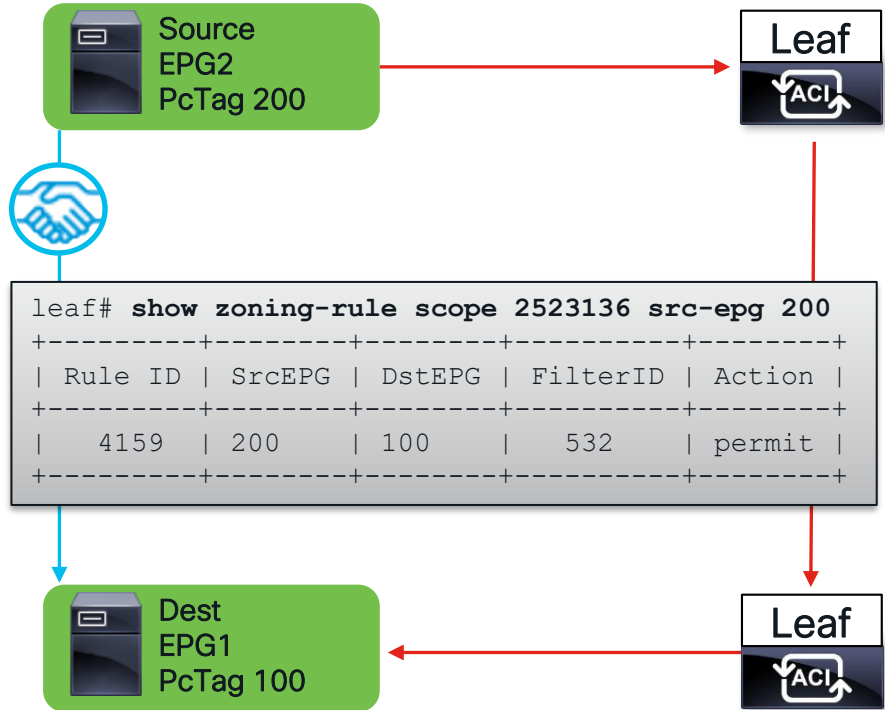
-pcTag of 1 indicates the fabric owns the subnet, don't apply policy
-policy applied flags not set in ivxlan header

Don't apply policy, Forward to proxy Anycast!

```
leaf101# show isis dtep vrf overlay-1 | egrep "Type|PROXY"
DTEP-Address    Role  Encapsulation  Type
10.0.176.66     SPINE N/A             PHYSICAL,PROXY-ACAST-V4
10.0.176.65     SPINE N/A             PHYSICAL,PROXY-ACAST-MAC
10.0.176.64     SPINE N/A             PHYSICAL,PROXY-ACAST-V6
```

Contracts and Forwarding

Check hidden slide for impact of "Policy Control Enforcement Direction" setting



What About Flooded Traffic?

The following traffic may be flooded:

- Broadcast
- Multicast
- Unknown Unicast
- Control Plane maintenance (EP announce, fabric ARP, etc)

The screenshot shows the Cisco DNA Center interface for 'Networking - Bridge Domains'. A table lists three bridge domains (bd1, bd2, bd3) with their respective segments, VRFs, and multicast addresses. An orange box highlights the 'Multicast Address' column, and a blue box labeled 'GiPo' has an arrow pointing to the address '225.0.159.112' for bd3.

Name	Segment	VRF	Multicast Address
bd1	15859679	vrf1	225.0.2.128
bd2	16613259	vrf1	225.0.8.48
bd3	16187328	vrf2	225.0.159.112

How does ACI flood?

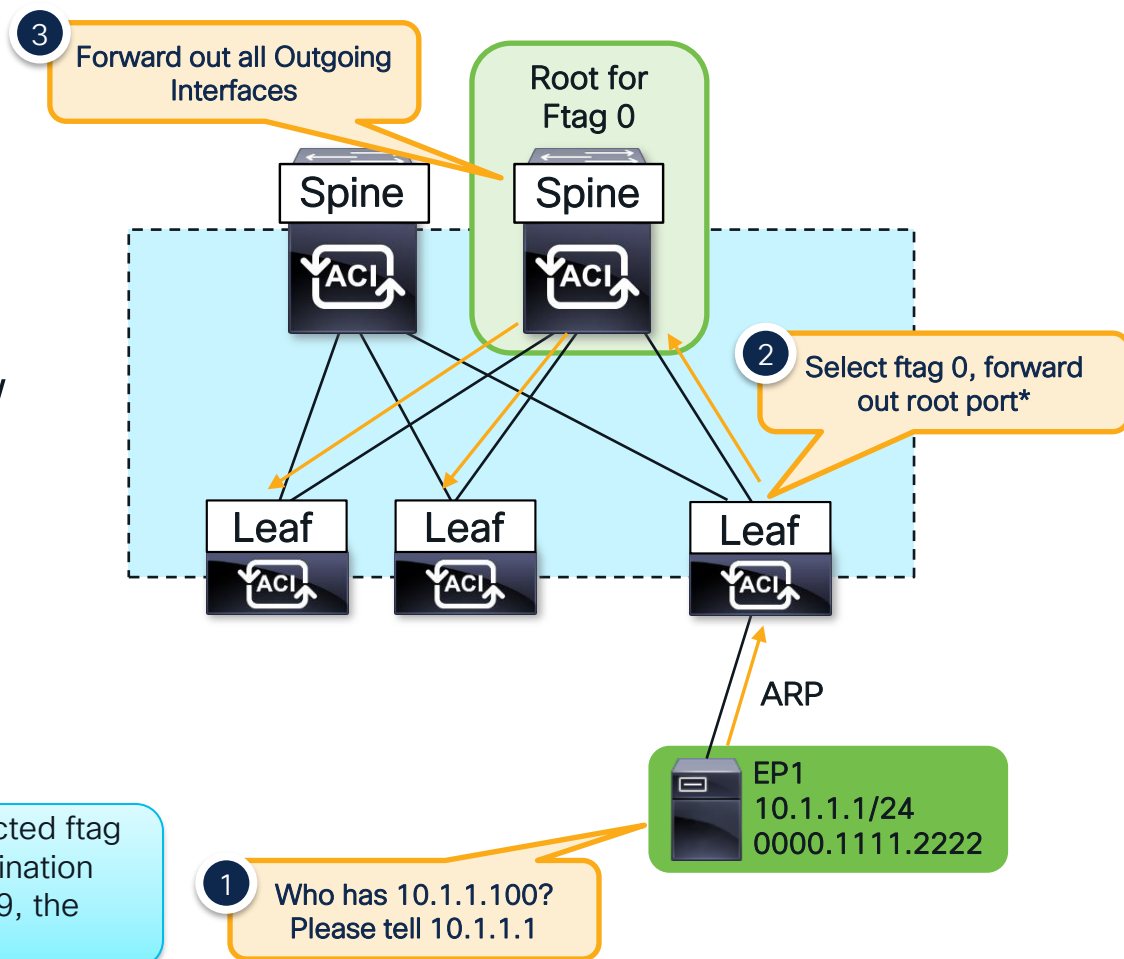
- The GiPo is an overlay multicast address allocated to a BD or VRF
- Flooded traffic is sent to the BD GiPo (I2 flood) or VRF GiPo (I3 flood)
- Flooding is done on a loop-free tree called an FTAG
- Security policy NOT applied

GiPo

What are FTAGs?

- FTAGs are loop-free trees within the overlay used by flooded traffic
- FTAGs are picked per flow from values 0x0 – 0xc
- One spine is root for each tree
- Outgoing interfaces calculated by ISIS

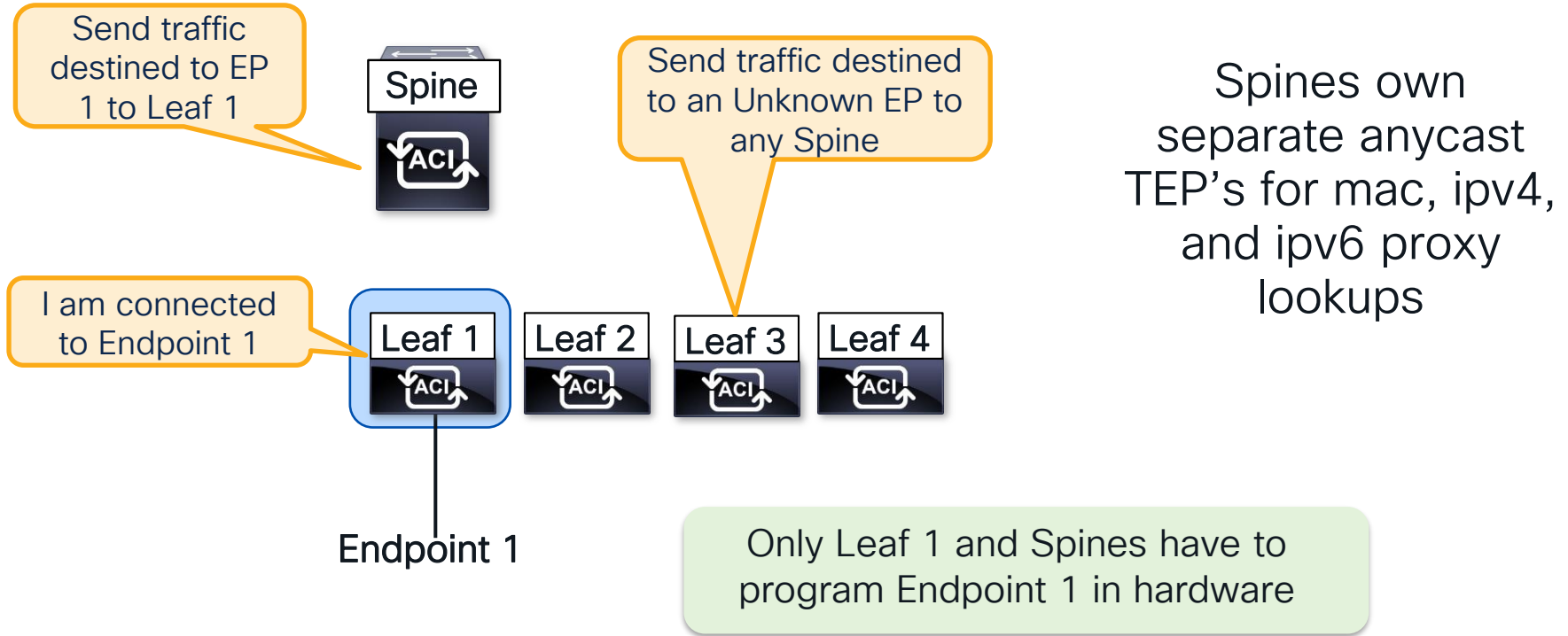
*Note, the ingress leaf communicates the selected ftag to the rest of the fabric by adding it to the destination gipo. If the gipo is 225.0.0.0 and the ftag is 0x9, the destination address would be 225.0.0.9



Proxy Forwarding

What is Proxy Forwarding?

Why? Scaling out Endpoint Learning



How to check the Spine-Proxy TEP

```
leaf1# show ip route vrf CL2022:vrf1

192.168.0.0/24, ubest/mbest: 1/0, attached, direct, pervasive
  *via 10.0.16.64%overlay-1, [1/0], 00:21:39, static
```

BD Subnet (Pervasive Route)

next-hop should be
SPINE-PROXY

```
leaf1# show isis dsteps vrf overlay-1 | grep PROXY
10.0.16.65          SPINE    N/A          PHYSICAL, PROXY-ACAST-MAC
10.0.16.64          SPINE    N/A          PHYSICAL, PROXY-ACAST-V4
10.0.16.67          SPINE    N/A          PHYSICAL, PROXY-ACAST-V6
```

next-hop of Pervasive Route
is IPv4 Spine Proxy TEP

Three types of Spine Proxy TEP

- Proxy-Acast-MAC
 - ✓ Spine-Proxy for L2 traffic (L2 Unknown Unicast mode “Hardware Proxy”)
- Proxy-Acast-V4
 - ✓ Spine-Proxy for IPv4 traffic (includes ARP Request with ARP Flooding mode “OFF”)
- Proxy-Acast-V6
 - ✓ Spine-Proxy for IPv6 traffic

What is COOP?

COOP is the proxy-database of ACI

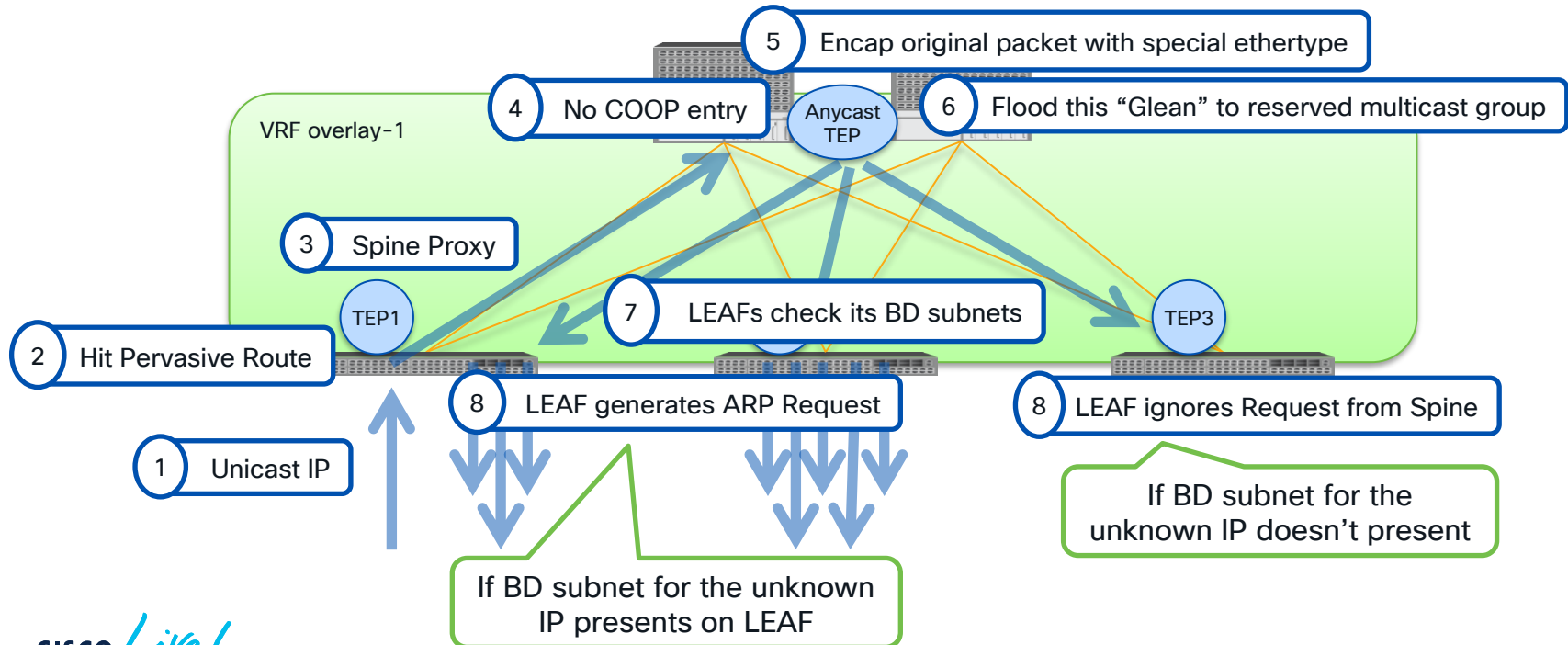
- Council of Oracles Protocol – A TCP protocol for citizens (Leafs) to publish records to oracles (Spines)
- Used for announcing endpoints, fabric owned IP's, multicast information, and more
- Synced across Pods/Sites with BGP EVPN
- Each Endpoint Record contains all information to forward (VNID, leaf TEP, mac, etc)
- COOP records pushed into hardware on spines
- For modular spines, scale is achieved by pushing each EP onto only two Fabric Modules

What if the Endpoint isn't in COOP? (ARP Glean)

What if Spine's COOP DB doesn't know the destination when proxy'ed?

✗ L2 Traffic : Drop

✓ L3 Traffic : ARP Glean



How ACI Builds Forwarding Tables

Building Adjacency Tables

ACI combines ARP and MAC Tables into the Endpoint Table

Legacy Behavior

- ARP/ND tables map Layer 3 to Layer 2
- ARP/ND tables are updated by control-plane messages
- MAC Address Table used for switching decisions
- Mac Address Table updated by dataplane

ACI Behavior

- Endpoint table contains endpoints, which are Layer 2 addresses OR Layer 3 addresses OR a combination of Layer 2 and Layer 3 addresses
- By default, both Layer 2 and Layer 3 information is updated by dataplane
- Used for security and forwarding policy

Building Endpoint Tables

Endpoints can be programmed via software process or by hardware dataplane learns (HAL)

Resource

Table Info

Commands to Verify

Supervisor

EPM - Endpoint Manager
Sup process for managing
endpoints.

```
show system internal epm endpoint mac <addr>  
show system internal epm endpoint ip <addr>
```

Line Card

EPMC - Endpoint Manager Client
Line card process that sits
between hardware layer (HAL)
and EPM

```
vsh_lc -c "show system internal epmc endpoint mac  
<addr>"  
vsh_lc -c "show system internal epmc endpoint ip <addr>"
```

ASIC

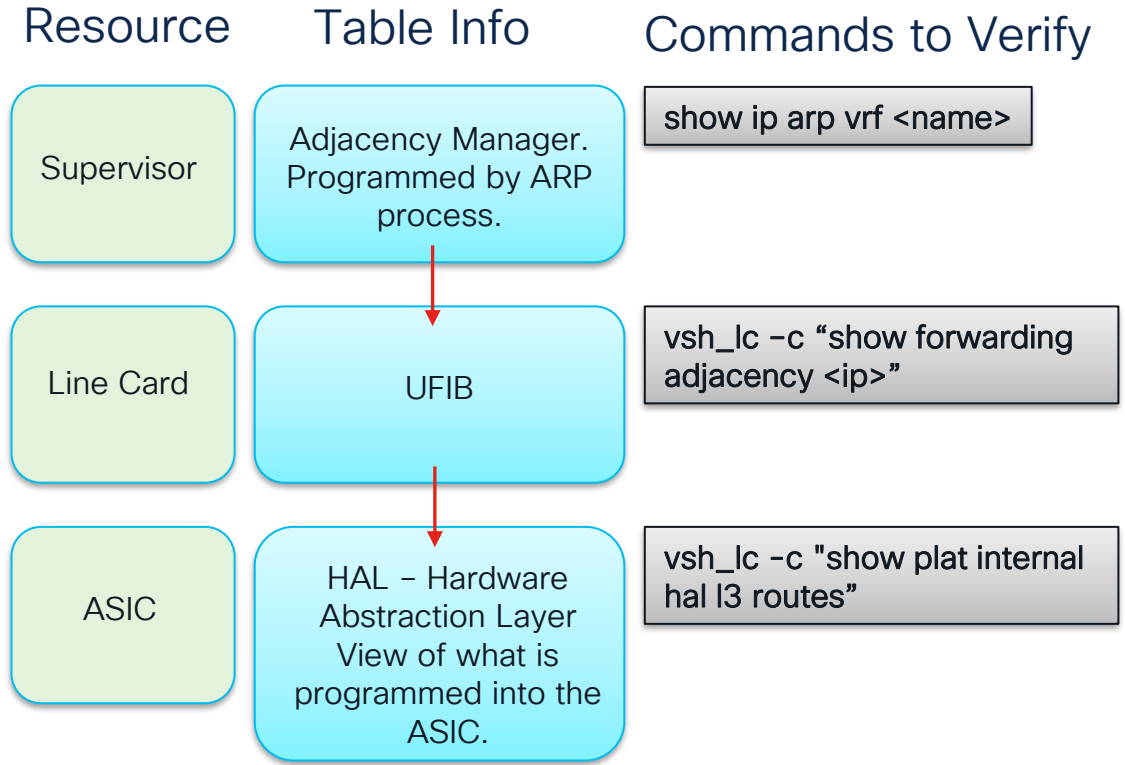
HAL - Hardware Abstraction Layer
View of what is programmed into
the ASIC.

```
vsh_lc -c "show plat internal hal ep I2 mac <addr>"  
vsh_lc -c "show plat internal hal ep I3 ip <ip/pfx len>"  
!  
!L3 Endpoints are put into HW Routing Table  
vsh_lc -c "show plat internal hal I3 routes | grep EP"
```

What about ARP?

ARP Tables are still used in ACI for...

- L3outs
- Overlay adjacencies
 - VXLAN Endpoints (AVE, K8s, Openstack, etc)
 - APIC / Fabric node adjacencies



Building Routing Tables

Resource

Table Info

Commands to Verify

Supervisor

URIB / MRIB – the unicast and multicast routing tables.
Programmed by route protocol

```
show ip route x.x.x.x/y vrf <name>  
show ip mroute x.x.x.x/y vrf <name>
```

Line Card

UFIB / MFIB – the unicast and multicast forwarding tables on the Line Card

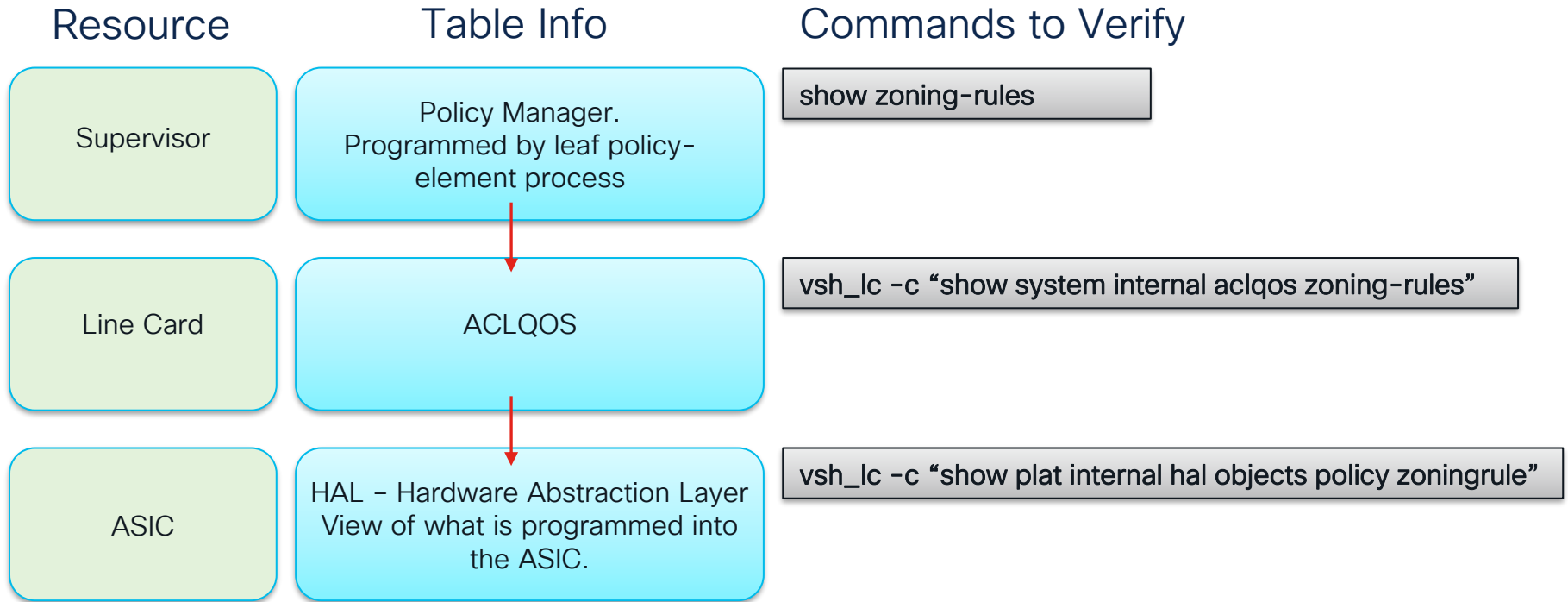
```
vsh_lc -c "show forwarding route <ip/pfx len> vrf <name>"  
vsh_lc -c "show forwarding multicast route vrf <name>"
```

ASIC

HAL – Hardware Abstraction Layer
View of what is programmed into the ASIC.

```
vsh_lc -c "show platform internal hal I3 routes vrf <name>"  
vsh_lc -c "show platform internal hal I3 mcast routes vrf <name>"  
vsh_lc -c "show plat internal hal I3 routes vrf <name>" | grep MC
```

Programming Contracts

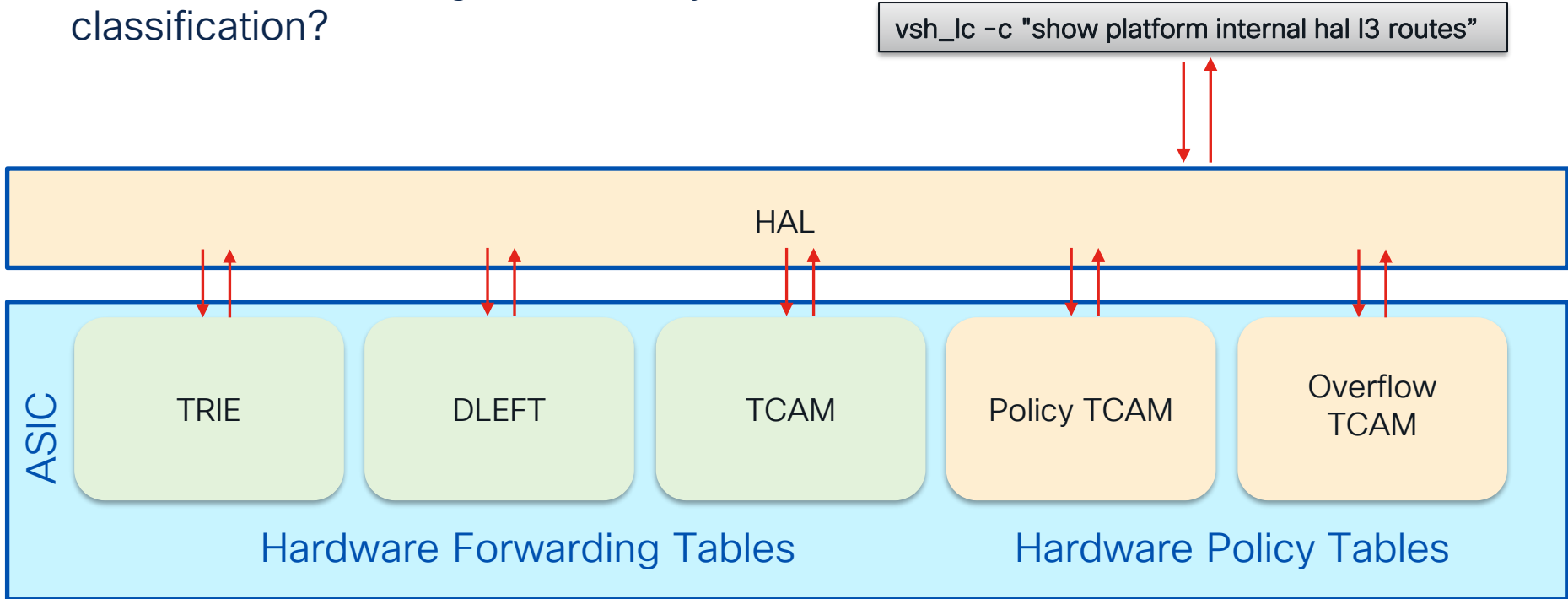


HAL – Hardware Abstraction Layer

Applicable to EX and Later Hardware

Wouldn't it be great if there was a single point to validate forwarding and security classification?

```
vsh_lc -c "show platform internal hal l3 routes"
```



HAL – Hardware Abstraction Layer

Applicable to EX and Later Hardware

L3 Lookup of Hardware Tables

```
module-1# show plat internal hal l3 routes vrf CL2022:vrfl
```

VRf	Prefix/Len	RT	Type	LID	CLSS	Flags
4626	192.168.100.10/ 32	EP	TRIE	!!	c002	le,bne,sne, dl
4626	10.99.99.0/ 24	UC	TCAM	!!	8004	sc,spi,dpi
4626	192.168.255.0/ 24	UC	TCAM	!!	24	sc,spi,dpi, dr
4626	192.168.200.11/ 32	EP	TRIE	!!	8003	sc, le,sne

Much more info available in full output!

Consolidated view of routes for Endpoints, Shared Services, and External routes

PcTag from destination EPG...used for contract lookup

Understanding the Configuration Options

VRF Level Forwarding Options

Feature

What Does it Do?

Policy Control Enforcement Preference

If disabled, policy is never applied between EPGs. If enabled, contracts are enforced.

IP Dataplane Learning

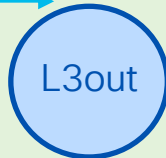
If Disabled, ACI uses legacy behavior for learning endpoints. Layer 3 endpoints are learned by ARP/GARP/ND and Layer 2 endpoints are learned by dataplane.

Policy Control Enforcement Direction

If set to Ingress, contract enforcement for I3out flows is done on service leaf. Egress enables enforcement on Border Leaf (requires remote learning to be enabled)

Ingress Enforcement

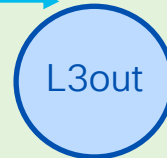
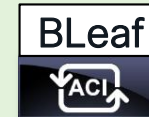
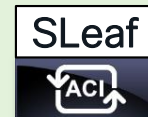
Ingress leaf sets policy applied bits



Egress leaf does not set policy applied bits

Egress Enforcement

Ingress leaf does not set policy applied bits



Egress leaf sets policy applied bits

Bridge-Domain Level Forwarding Options

Feature	What Does it Do?
L3 Unknown Multicast Flooding	For non-link-local L3 multicast traffic in a PIM-disabled BD, should a leaf with no snooping entries flood in BD (flood) or wait for joins (OMF)?
Multidestination Flooding	For L2 mcast and broadcast, flood, drop, or flood within epg encap? If flooding with EPG encap, proxy-arp is required for cross-epg L2 communication
L2 Unknown Unicast	If destination mac is unicast and unknown, flood or proxy to spines?



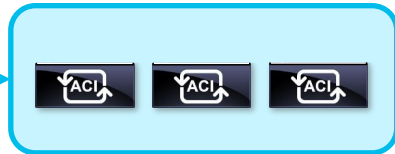
Proxied, L2 Unknown Unicast is dropped if the Destination MAC isn't known in COOP

Bridge-Domain Level Forwarding Options

Feature	What Does it Do?
Limit IP Learning to Subnet	Only learn IP's if they are within the configured BD subnet for local learns.
Unicast Routing	Enable IP learning as well as unicast routing (if a BD subnet is configured)
IP Data-plane Learning	Configured underneath the BD subnet. When disabled, IP/IPv6 learning is done via ARP/ND
ARP Flooding	When disabled, ARP is unicast routed based on the Target IP (if known)



Who has
192.168.100.11?



```
leaf# show endpoint ip 192.168.100.11
leaf# show ip route 192.168.100.11 vrf CL2022:vrf1

192.168.100.0/24, ubest/mbest: 1/0, direct, pervasive
*via 10.0.176.66%overlay-1, [1/0], 01w00d, static
recursive next hop: 10.0.176.66/32%overlay-1
```

Proxy!

EPG Level Forwarding Options

Feature	What Does it Do?
Flood in Encapsulation	Feature is enabled for just the EPG (rather than all EPG's in the BD). Requires proxy ARP for L2 traffic between encaps.
L4-L7 Virtual IP's	Designed for Direct Server Return flows. This disables dataplane learning per IP. IP is learned by ARP/ND.
Disable DP Learning Per-IP/Prefix	Disables dataplane learning. More specific than VRF-level option. In most cases should be used for DSR too.

New in 5.2, can also be configured on BD

Global Forwarding Options

Feature	What Does it Do?
Enforce Subnet Check	Don't learn an IP (both local and remote) if it is not within a configured BD subnet in the VRF.
Disable Remote EP Learning on BL's	Remote IP learning is disabled for Unicast flows on a leaf in a specific VRF if an I3out exists in the same VRF

```
graph TD; A[Disable Remote EP Learning on BL's] --> B[Multicast sources are still learned]; A --> C[Also implicitly disabled when intersite I3out is configured];
```

Understanding the Tools

Start with High-level Tools

Use Endpoint Tracker for Building a Topology

EP Tracker

End Point Search

EP Locally Learned on pod 2, nodes 401-402

172.16.31.100

Search

Learned At	Tenant	Application	EPG	IP
2/401-2/402, vPC: vpc-esxi-10.2.10.19 (learned,vmm)	CiscoLive	Database	DB	172.16.31.100

End Point Search

No EP Learn, is this an L3out?

10.255.255.100

Search

Learned At	Tenant	IP
No items have been found.		

Start with High-level Tools

Use Atomic Counters to Check for Overlay Drops and Latency (PTP)

Add EP to EP Policy



Name:

Description:

Administrative State: Disabled Enabled

Features: Atomic Counter
 Latency Statistics

Source Type: EP IP

Source IP:

Application Profile EPG/ESG Client Endpoint Internet Protocol

Destination IP:

Application Profile EPG/ESG Client Endpoint Internet Protocol

Filters:

Name	Protocol	Source port	Destination port	Description
ip	Unspecified	Unspecified	Unspecified	

Start with High-level Tools

Use Atomic Counters to Check for Overlay Drops and Latency (PTP)

CiscoLive

- Policies
- Protocol
- Troubleshooting
 - SPAN
 - Traceroute
 - Atomic Counter and Laten...
 - EP to EP
 - CL-AC
 - EP to EPG

EP to EP CL-AC

EP-to-EP Atomic Counter - CL-AC

Source	Destination	Last Collection (30 seconds) Pkt			
		Transmit	Admitted	Dropped	Excess
uni/tn-CiscoLive/ap-Databas...	uni/tn-CiscoLive/ap-APP/epg...	29	29	0	0

104 Microseconds of delay in overlay

No overlay drops!

- SPAN
- Traceroute
- Atomic Counter and Laten...
 - EP to EP
 - CL-AC
 - EP to EPG

EP-to-EP Latency Average - CL-AC

Last 30 Seconds Collection 04/25/2022 16:06:05			Cumulative (04/25/2022 15:04:45 - 04/25/2022 16:06:05)		
Average(μs)	Standard Deviation(μs)	Packet Count	Average(μs)	Max(μs)	Packet Count
104.8575	0.0000	29	104.8575	104.8575	3768

Start with High-level Tools

Use Tenant Visibility tools to check for Contract Drops

The screenshot shows the CiscoLive interface for a tenant. The navigation menu on the left includes Quick Start, CiscoLive, Application Profiles, Networking, Contracts, Policies, Services, and Security. The main content area is titled 'Tenant - CiscoLive' and has tabs for Summary, Dashboard, Policy, Operational, Stats, Health, Faults, and History. Under the 'Operational' tab, there are sub-tabs for Endpoints, Flows, Packets, Policy Tags, and Resource IDs. The 'Packets' sub-tab is active, and the 'L3 Drop' filter is selected. A table displays a list of denied packets. A callout box with an orange border and arrow points to the first row of the table, containing the text 'This flow is being contract dropped'.

Timestamp	VRF	Src IP	Dest IP	Protocol	Src Port	Dest Port	Node
2022-04-25T17:19:44.070+00:00	CustA	172.16.31.100	10.255.255.100	icmp	unspecified	unspecified	node-402
2022-04-25T17:19:39.430+00:00	CustA	172.16.31.100	10.255.255.100	icmp	unspecified	unspecified	node-402
2022-04-25T17:18:53.350+00:00	CustA	172.16.31.100	10.255.255.100	icmp	unspecified	unspecified	node-402
2022-04-25T17:11:12.545+00:00	CustA	172.16.31.100	10.255.255.100	icmp	unspecified	unspecified	node-402
2022-04-25T17:18:52.870+00:00	CustA	172.16.31.100	10.255.255.100	icmp	unspecified	unspecified	node-402
2022-04-25T17:18:52.326+00:00	CustA	172.16.31.100	10.255.255.100	icmp	unspecified	unspecified	node-402

```
apic4# show aclog deny l3 pkt tenant common vrf CORE
srcIp dstIp protocol srcPort dstPort node srcIntf vrfEncap
-----
<EMPTY>
```

Contract Parser

The script checks zoning rules, filters, statistics against EPG names

```
Leaf# contract_parser.py --help
--nz, --nonzero          display only entries with non-zero hits
--incremented            display only entries that have incremented since last checked
--node NODES [NODES ...]
                        display entries specific to one or more leaf nodes
--contract CONTRACT [CONTRACT ...]
                        display only rules that match a specific contract. The
                        name of the contract is in the form
                        uni/tn-<tenant>/brc-<contract>
--vrf VRF [VRF ...]     display entries for a specific vrf. The integer vnid
                        of the vrf can be provided or the vrf name in the form
                        <tenant>:<vrf>
--epg EPG [EPG ...]    display entires for specific EPG. The integer pctag or
                        DN name can be provided. Note the dn is a partial dn
                        in the form
                        tn-<tenant>/ap-<applicationProfile>/epg-<epg>
```

ACI FTAG VIEWER

Check the FTAG topology in an ACI fabric

```
apic1# ./aci_ftag_viewer.py --pod 2 --ftag 0
```

```
#####  
# Pod 2 FTAG 0  
# Root spine-303  
# active nodes: 3, inactive nodes: 0  
#####  
spine-303  
+- 1/1 ----- 1/49 leaf-401  
+- 1/2 ----- 1/49 leaf-402  
+- 1/3 ..... (EXT) Ethernet1/9 n9504
```

No errors on FTAG 0!

```
Pod 2 FTAG 0: all nodes reachable on tree
```

Start with High-level Tools

Port Counters are as Useful as Ever

```
leaf1# show interface eth1/8
Ethernet1/8 is up
admin state is up, Dedicated Interface
Last link flapped 03:07:41
RX
 3527922 unicast packets !omitted
 4041582 input packets 609518993 bytes
 12 jumbo packets 0 storm suppression bytes
 0 runts 0 giants 0 CRC 0 Stomped CRC 0 no buffer
 0 input error 0 short frame 0 overrun !omitted
 0 watchdog 0 bad etype drop 0 bad proto drop !omitted
 0 input with dribble 0 input discard
 0 input buffer drop 0 input total drop
TX
 32262479565 unicast packets !omitted
 32395063346 output packets 49034781261
 32249687943 jumbo packets
 0 output error 0 collision 0 deferred
 0 lost carrier 0 no carrier 0 babble 0 output discard
 0 output buffer drops 0 output total drops
```

Frames received with bad FCS

Indicates a previously stomped frame was received

What is a Stomp?

- When a frame is received with a bad FCS and/or is malformed

AND

- The frame is cut-through switched

The switch will invert the new CRC to tell the first store-and-forward device to drop it

Frame transmitted with stomped CRC

Buffer drops, sign of congestion

0 output error

0 output buffer drops

Start with High-level Tools

Using moquery to check port counters fabric-wide

#Check Fabric-wide for FCS Errors

```
moquery -c rmonDot3Stats -f 'rmon.Dot3Stats.fCSErrors>="1"' | egrep "dn|fCSErrors"
```

#Check Fabric-wide for total CRC Stomp + FCS Errors

```
moquery -c rmonEtherStats -f 'rmon.EtherStats.cRCAlignErrors>="1"' | egrep "dn|cRCAlignErrors"
```

#Check Fabric-wide for Output Buffer Drops

```
moquery -c rmonEgrCounters -f 'rmon.EgrCounters.bufferdropPkts>="1"' | egrep "dn|bufferdropPkts"
```

#Check Fabric-wide Output Errors

```
moquery -c rmonIfOut -f 'rmon.IfOut.errors>="1"' | egrep "dn|errors"
```


ELAM – Embedded Logic Analyzer Module

- It is a tripwire in hardware
- The first frame to match a specified condition ‘trips’ it
- Report is created with vast amount of data regarding asic decisions

Dst - TCP 10.0.0.1:3000

Dst - TCP 10.0.0.1:3001

Dst - TCP 10.0.0.1:3002



```
vsh_lc
debug platform internal tah elam asic 0
trigger reset
trigger init in-select 6 out-select 1
set outer ipv4 dst_ip 10.0.0.1
set outer 14 dst-port 3001
start
```

Frame was not
dropped in lookups!

```
module-1(DBG-elam-insel6) # stat
ELAM STATUS
=====
Asic 0 Slice 0 Status Armed
Asic 0 Slice 1 Status Triggered

module-1(DBG-elam-insel6) # ereport | grep "drop reason"
RW drop reason           : no drop
LU drop reason           : no drop
```

Matching frame was
caught!

What ASIC should be set in the ELAM?

```
vsh_lc  
debug platform internal <asic> elam asic 0
```

Model	Role	Asic for Elam
N9K-C*C	Fixed Spine	roc
N9K-C*GX	Fixed Spine	app
N9K-C*-EX	Leaf	tah
N9K-C*-FX/FXP/FX2	Leaf	roc
N9K-C*-GX	Leaf	app
N9K-C*-GX2	Leaf	cho
N9K-X97*-EX	Spine LC	tah
N9K-X97*-FX	Spine LC	roc
N9K-X97*-GX	Spine LC	app
N9K-C95*-FM-E	Spine FM	tah
N9K-C950*-FM-E2	Spine FM	roc
N9K-C95*-FM-G	Spine FM	app

Steps to Using Elam on Gen2+ Leaf or Fixed Spine

Elams are run from the line card shell

Refer to "What ASIC should be set in the ELAM" slide

Leaves and fixed spines are single ASIC switches. Always use ASIC 0

vsh_lc

```
debug platform internal tah elam ASIC 0
```

trigger reset

```
trigger init in-select 6 out-select 0
```

```
set outer ipv4 dst_ip 10.0.0.1
```

```
set outer 14 dst-port 3001
```

```
start
```

Failing to reset the trigger can cause past elam configurations to take effect. Always reset the trigger!

Use 0 or 1

```
module-1 (DBG-elam) # trigger init in-select ?  
!omitted  
14 Outer(12(vntag)|13|14)-inner(12|13|14)-ieth  
6 Outer12-outer13-outer14  
7 Inner12-inner13-inner14  
!omitted
```

Determines which headers conditions can be matched in. Use 14 or 7 when matching vxlan encapsulated headers.

Steps to Using Elam on Gen2+ Leaf or Fixed Spine

Use "set outer" or "set inner" depending on in-select and if matching outer or inner headers in vxlan packet

Which headers to match conditions for?

```
vsh_lc
debug platform internal tah elam asic 0
trigger reset
trigger init in-select 6 out-select 0
set outer ipv4 dst_ip 10.0.0.1
set outer 14 dst-port 3001
start
```

What to match in the header?

Finally enable the elam!

When running `stat` if `Triggered` is seen, this means a matching packet was received

Reading an Elam

ereport available since 4.2

At a high-level...

```
module-1 (DBG-elam-inse16) # ereport
!omitted
-----
Outer L3 Header
-----
L3 Type           : IPv4
IP Version        : 4
DSCP              : 0
IP Packet Length  : 84 ( = IP header(28 bytes) + IP payload )
Don't Fragment Bit : set
TTL              : 64
IP Protocol Number : ICMP
Destination IP    : 192.168.200.11
Source IP         : 192.168.100.10
!omitted
Contract Result
Contract Drop     : no
Contract Logging  : no
Contract Applied  : yes
Contract Hit      : yes
```

- ereport provides a simple, human-readable report output
- ereport requires ≥ 5.2 code for modular spines
- Groups data into outer/inner, headers, and lookup results

What if Elam Shows a Drop?

ereport available since 4.2

ereport

Lookup Drop

LU drop reason : SECURITY_GROUP_DENY

Common Drop Reasons

Drop Code	What Does it Mean?	What to Do?
ACL_DROP	For traffic destined to the CPU on an FX switch it is expected and cosmetic. Also seen when traffic was received from a fabric port and the leaf has a remote EP learn with no bounce flag.	Ignore if its an FX switch and destined to local switch IP/process. Otherwise, check for incorrect EP learn.
DCI_*_XLATE_MISS	For multisite / remote-leaf, there was no matching vnid or ptag translation found.	Check contracts between local and remote resources.
INFRA_ENCAP_SRC_TEP_MISS	No route and/or tunnel found back to the outer source IP	Check for a tunnel pointing back to the outer source IP. Also, check for a route in overlay.
SECURITY_GROUP_DENY	Frame was contract dropped	Make sure a contract is configured to allow the flow.
SRC_VLAN_MBR	Received vlan not programmed on ingress port.	Check if the frame was correct tagged/untagged. Make sure no invalid-path faults exist for the epg.
UC_PC_CFG_TABLE_DROP	No route was found for the destination.	Check the routing table for the destination.
VLAN_XLATE_MISS	Received vlan doesn't exist on the switch.	Check if the frame is tagged with correct vlan. Check for invalid-path faults on the epg.

Steps to Using Elam on Gen2+ Modular Spine

Challenges of Modular Spines

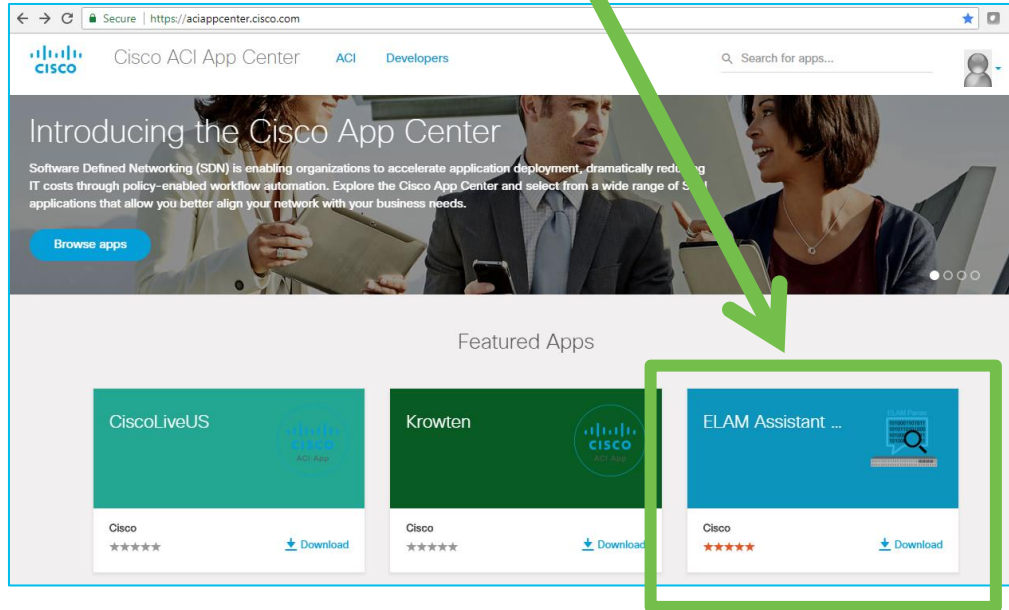
- Line cards (and potentially FM's) have multiple asics
- Elam must specify asic number
- Ingress/Egress ports may be internal LC – FM connections
- ereport only available in 5.2 and later

Fortunately, spine elams aren't needed as commonly as leaf elams!

Shouldn't ELAM be More Simple?

Elam Assistant in DCAppCenter

<https://dcappcenter.cisco.com>



ELAM (Embedded Logic Analyzer Module)

- Perform an ASIC level packet capture

ELAM Assistant

- You can perform ELAM like a TAC engineer!
- With a nicely formatted result report

Detail Explanations:

- <https://dcappcenter.cisco.com/elam-assistant.html>
- How to use video, pictures
 - A download link for ELAM Assistant

ELAM Assistant in ACI AppCenter (example)

1. Perform an Elam

The screenshot shows the ACI AppCenter interface for the ELAM Assistant. The top navigation bar includes System, Tenants, Fabric, Virtual Networking, Admin, Operations, **Apps**, and Integrations. Below this, there are sub-navigations for Installed Apps, Faults, and Downloads. The main content area is titled 'ELAM Assistant' and 'Capture a packet with ELAM (Embedded Logic Analyzer Module)'. On the left, a sidebar lists various nodes for capture, including 'node-101 (site2-pod1-leaf1)', 'node-102 (site2-pod1-leaf2)', 'node-203 (site2-pod1-spine3)', 'node-303 (site2-pod2-spine3)', 'node-401 (site2-pod2-leaf1)', 'node-402 (site2-pod2-leaf2)', and 'Unsupported Nodes'. The main area displays 'ELAM Parameters' with a table of configurations. The table has columns for Status, Node, Direction, Source I/F, Parameters, and VxLAN (outer) header. Three rows are shown, with the first row having a 'Set' button and the others having 'Report Ready' buttons. Below the table are buttons for 'Set ELAM(s)' and 'Check Trigger'. At the bottom, there is a section for 'ELAM Report Parse Result (report name:)' with tabs for Express, Detail, and Raw, and a 'Select a report.' prompt. Two callout boxes are present: one on the left pointing to the 'Report Ready' status with the text 'Triggered!! and Report is Ready', and one on the right pointing to the 'Set Parameters' button with the text 'Set Parameters'.

Status	Node	Direction	Source I/F	Parameters	VxLAN (outer) header
Set	node-401	from downlink	any	dst ip 10.255.255.100	
Report Ready	node-402	from downlink	any	dst ip 10.255.255.100	
Report Ready	node-303	from LEAF/IPN	any	dst ip 10.255.255.100	

Triggered!!
and
Report is Ready

Set Parameters

ELAM Assistant in ACI AppCenter (example)

2. Read a Report

Click to see report

Report shows up here

Scroll Down

ELAM Parameters

Name your capture

Status	Node	Direction	Source I/F	Parameter
Set	node-401	from downlink	any	
Report Ready	node-402	from downlink	any	
Report Ready	node-303	from LEAF/IPN	any	

▶ Set ELAM(s)

ELAM Report Parse Result (report name: node-402_slot1...)

Express Detail Raw

Captured Packet Information

Basic Info	
Device Type	Leaf
Packet Direction	ingress (from)
Incoming I/F	eth1/4

L2 Header	
Destination MAC	0022.BDF8.19FF
Source MAC	0050.569A.65DB
Access Encap VLAN	844

Packet Forwarding Information

Forward Result	
Destination Type	To another ACI node (LEAF, AVS/AVE etc.)
Destination TEP	10.1.240.33 (MAC Spine-Proxy)
Destination Physical Port	eth1/49

Contract	
Destination EPG pcTag (dclass)	0x4002 / 16386 (L3OUT CiscoLive:L3out-CUST:EEPG2)
Source EPG pcTag (sclass)	0x8005 / 32773 (CiscoLive:Database:DB)
Contract was applied	1 (Contract was applied on this node)

Drop	
Drop Code	no drop

FTRIAGE – Automating Elams

Orchestrate End-to-End
ELAMs from the APIC!

```
apic1# ftriage route -ii LEAF:101,102 -dip 10.99.99.100 -sip 192.168.100.10
20:19:54 INFO main:1295 L3 packet Seen on leaf102 Ingress: Eth1/34 (Po5) Egress: Eth1/54 Vnid: 2523136
20:19:55 INFO main:1364 leaf102: Packet's egress outer [SIP:10.0.176.67, DIP:10.0.64.70]
20:19:55 INFO main:1371 leaf102: Outgoing packet's Vnid: 2523136
20:19:56 INFO main:353 Computed ingress encap string vlan-3501
20:20:03 INFO main:464 Ingress BD(s) CL2022:bd1
20:20:03 INFO main:476 Ingress Ctx: CL2022:vrfl Vnid: 2523136
!
20:21:46 INFO main:1295 L3 packet Seen on spine1005 Ingress: Eth1/1 Egress: Eth1/3 Vnid: 2523136
20:22:38 INFO fib:737 spine1005: Transit in spine
20:23:32 INFO main:1295 L3 packet Seen on leaf103 Ingress: Eth1/29 Egress: Eth1/27/4 Vnid: NULL
!
20:24:02 INFO fib:219 leaf103: L3 out interface Ethernet1/27/4
20:24:10 INFO main:781 Computed egress encap string vlan-1055
20:24:17 INFO main:1796 Packet is Exiting fabric with peer-device: N3K-1 and peer-port: Ethernet1/31
```

SPAN / ERSPAN

Don't neglect old friends!

- Both local span and erspan supported
- ERSPAN requires an I3 endpoint learned anywhere in the fabric
- Still the best tool for checking –
 - Packet contents
 - Frame format
 - Retransmissions
 - ...and anything else that can be seen in a pcap

Other Tools Requiring External Resources

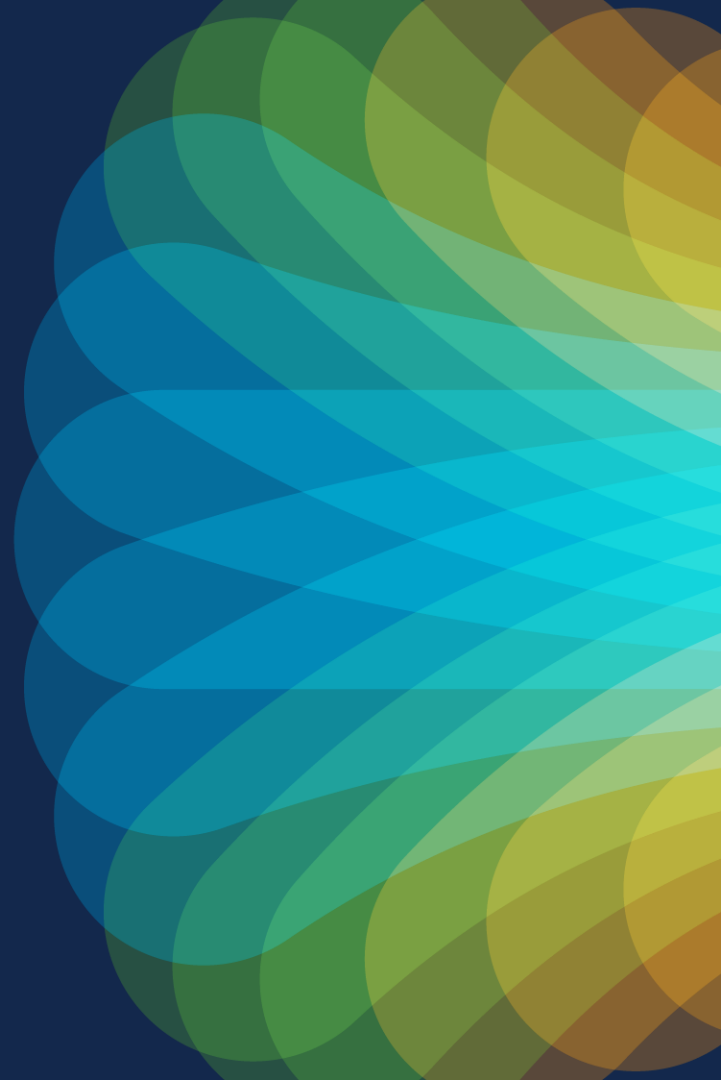
Netflow

- Captures flow information based on specified criteria
- Useful for troubleshooting packet loss and latency

Flow Telemetry

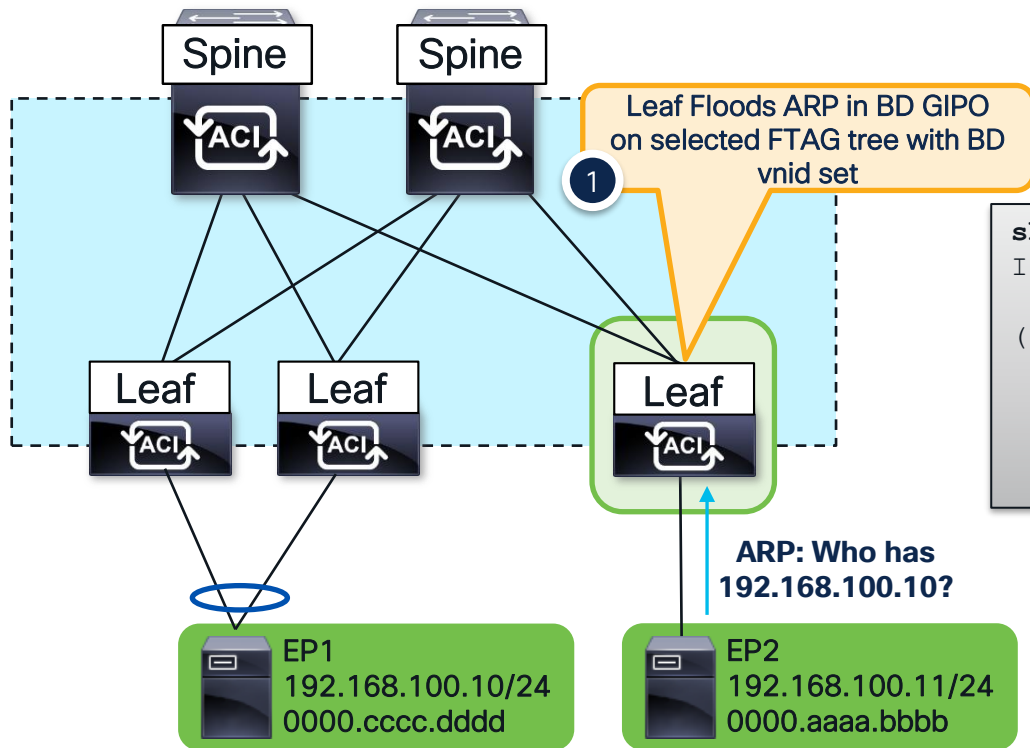
- Hardware directly streams flow data to Nexus Dashboard Insights
- Useful for troubleshooting packet loss and latency
- Latency measurements leverage PTP for additional accuracy
- NDI can perform additional flow analytics

Debugging ACI BUM Flows



ARP - Ingress Leaf

Bridge Domain Settings:
Unicast Routing Disable
ARP Flooding Enabled



Check GIPO Route

```
show ip mroute 225.0.2.128 vrf overlay-1
IP Multicast Routing Table for VRF "overlay-1"

(*, 225.0.2.128/32), uptime: 22w2d, isis
Incoming interface: Null, RPF nbr: 0.0.0.0
Outgoing interface list: (count: 2)
  Ethernet1/29.9, uptime: 8w2d
  Ethernet1/30.10, uptime: 22w2d
```

ARP – How to Find the GiPo

From the GUI...

The screenshot shows the Cisco APIC GUI with the 'Tenants' tab selected. The left sidebar shows a tree view with 'CL2022' expanded to 'Networking' and then 'Bridge Domains'. The main content area displays 'Networking - Bridge Domains' with a table of configurations.

Name	Segment	VRF	Multicast Address
bd1	14811121	vrf1	225.0.2.128
bd2	16613259	vrf1	225.0.8.48
bd3	16187328	vrf2	225.0.159.112

From the APIC CLI...

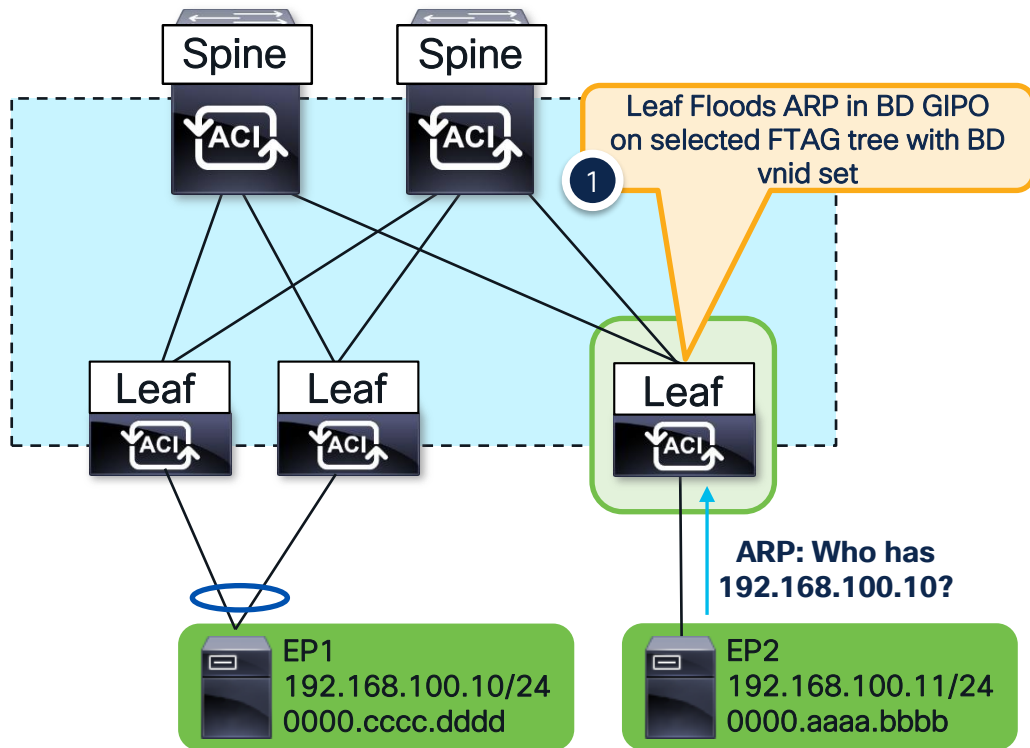
```
moquery -c fvBD -f 'fv.BD.dn*"tn-CL2022/BD-bd1"'  
  
# fv.BD  
arpFlood           : yes  
bcastP             : 225.0.2.128  
dn                 : uni/tn-CL2022/BD-bd1
```

From the Switch CLI...

```
moquery -c l2BD -f 'l2.BD.name=="CL2022:bd1"' -x rsp-subtree=full rsp-subtree-class=fmcastGrp  
# fmcast.Grp  
addr               : 225.0.2.128  
dn                 : sys/ctx-[vxlan-2523136]/bd-[vxlan-14811121]/fmgrp-[225.0.2.128]  
rn                 : fmgrp-[225.0.2.128]
```


ARP - Ingress Leaf

Bridge Domain Settings:
Unicast Routing Disable
ARP Flooding Enabled



ELAM the ARP request!

```
vsh_lc
debug plat internal app elam asic 0
trigger reset
trigger init in-select 6 out-select 0
set outer arp source-ip 192.168.100.11
set outer arp target-ip 192.168.100.10
start
!
stat
ELAM STATUS
=====
Asic 0 Slice 0 Status Armed
Asic 0 Slice 1 Status Armed
Asic 0 Slice 2 Status Triggered
Asic 0 Slice 3 Status Armed
```

ARP - Ingress Leaf Elam Results (ereport)

Bridge Domain Settings:
Unicast Routing Disable
ARP Flooding Enabled

Outer L2 Header

Access Encap VLAN : 3502 (0xDAE)

Make sure this matches
what is expected

Outer L3 Header

ARP Opcode : Request(0x1)
ARP Sender IP : 192.168.100.11
ARP Target IP : 192.168.100.10

Contract Result

Contract Drop : no
Contract Applied : no

FINAL FORWARDING LOOKUP

Bits set in Final Forwarding Block: : IFABRIC_IG MC TENANT MYTEP **BRIDGE** MISS **FLOOD**

Frame is flooded in the Bridge Domain!

Lookup Drop

LU drop reason : **no drop**

Not Dropped in lookups!

ARP – How to Find the FTAG

No other way than Elam...

```
module-1(DBG-elam-insel6)# ereport | grep "nopad.ftag"  
wol_lu2ba_sb_info.mc_info.mc_info_nopad.ftag: 0x8
```

Selected ftag is 0x8

- Leaf forwards to root port and OIF's for ftag 8
- Since GIPO is 225.0.2.128, Dest multicast address is 225.0.2.136 (gipo + ftag)
- Check ftag topology with **show isis internal mcast routes ftag**

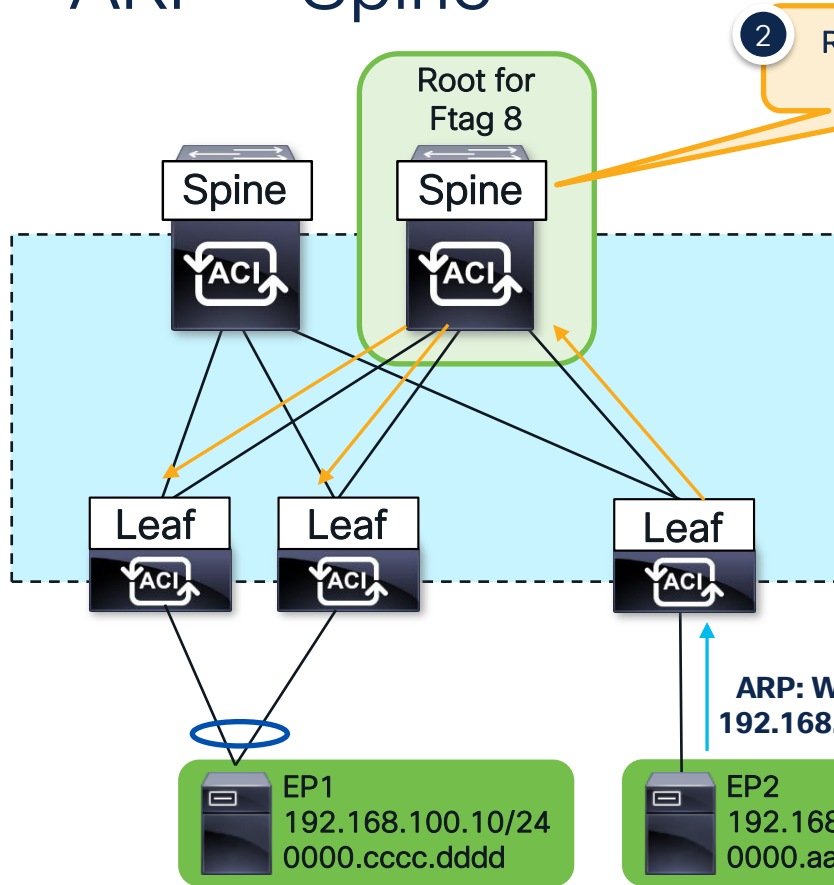
```
leaf103# show isis internal mcast routes ftag  
IS-IS process: isis_infra  
VRF : default  
FTAG Routes  
=====
```

FTAG ID:	8	[Enabled]	Cost:(1/	6/	0)
----------	---	-----------	--------	----	----	----

```
-----  
Root port: Ethernet1/29.9  
OIF List:
```

Leaf appends ftag to gipo and forwards out Eth1/29 to spine

ARP - Spine



2 Root spine for ftag 8 forwards out OIFs

Bridge Domain Settings:
Unicast Routing Disable
ARP Flooding Enabled

This spine is the root!

```
spine1005# show isis internal mcast routes ftag
IS-IS process: isis_infra
VRF : default
FTAG Routes
=====
FTAG ID: 8 [Root] [Enabled] Cost:( 0/ 0/ 0)
-----
Root port: -
OIF List:
Ethernet1/1.20
Ethernet1/2.21
Ethernet1/3.19
```

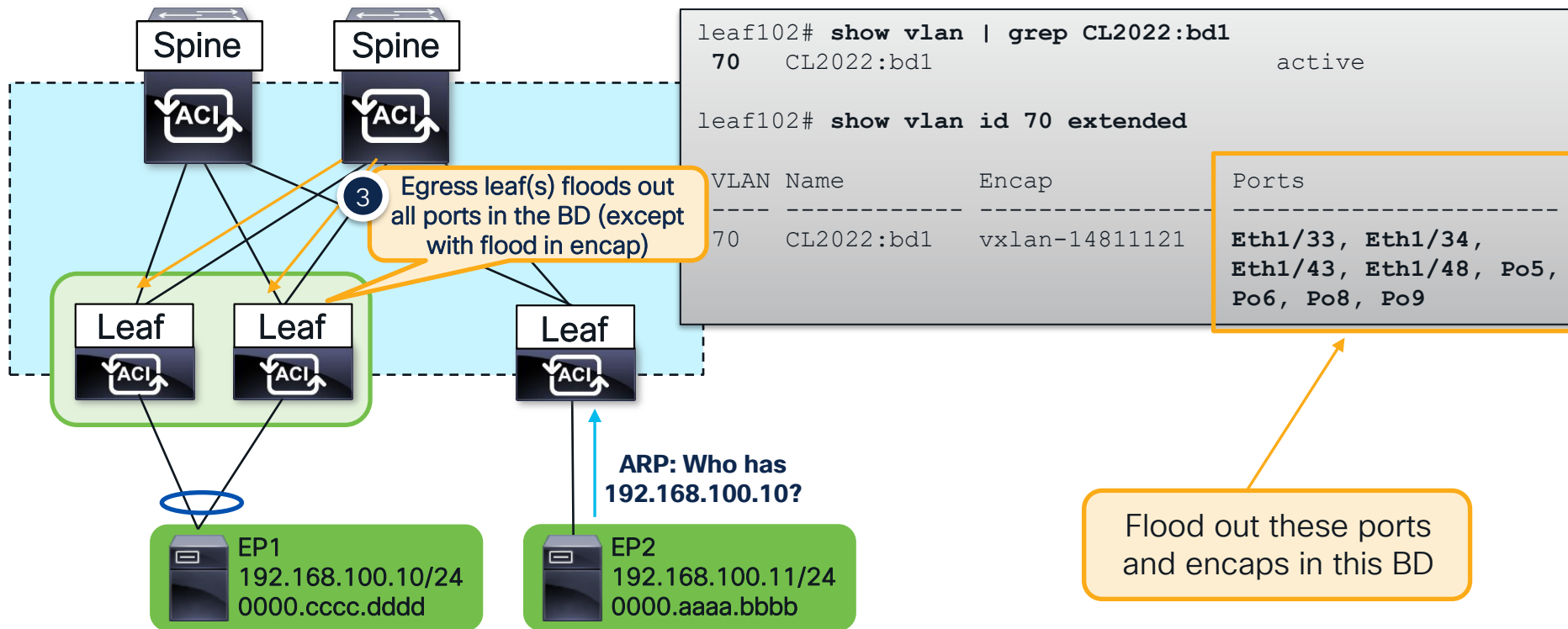
ARP: Who has 192.168.100.10?

EP2
192.168.100.11/24
0000.aaaa.bbbb

Spine forwards out OIFs

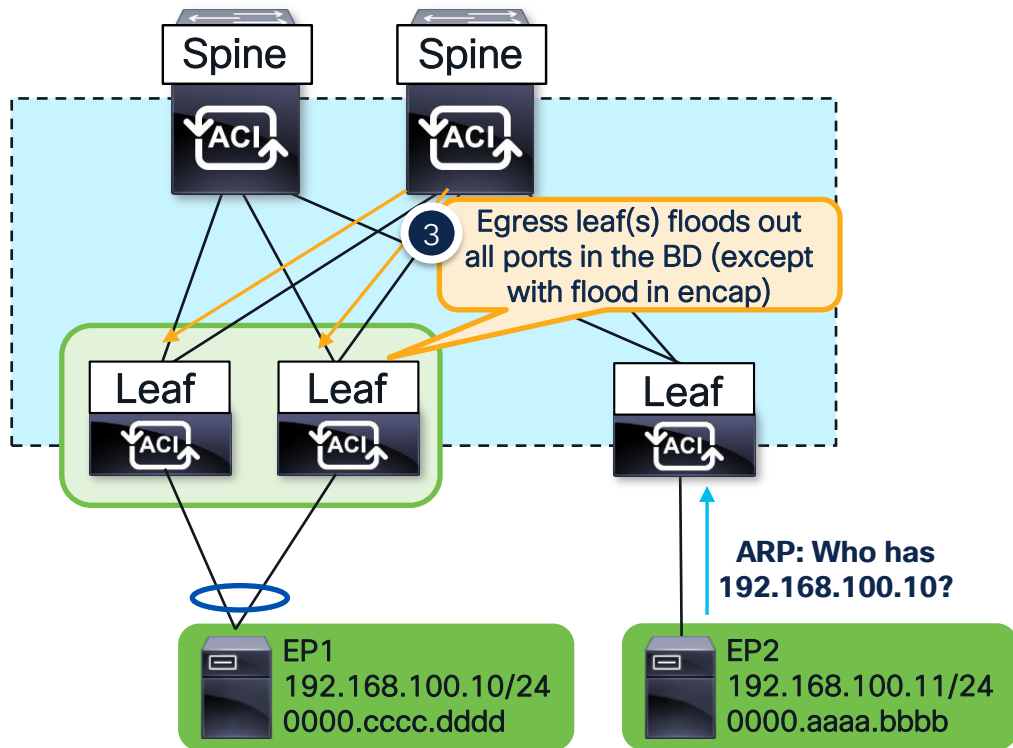
ARP - Egress Leaf

Bridge Domain Settings:
Unicast Routing Disable
ARP Flooding Enabled



ARP - Egress Leaf

Bridge Domain Settings:
Unicast Routing Disable
ARP Flooding Enabled



ELAM the ARP request!

```
vsh_lc
debug plat internal tah elam asic 0
trigger reset
trigger init in-select 14 out-select 1
set inner arp source-ip 192.168.100.11
set inner arp target-ip 192.168.100.10
set inner 12 dst_mac ffff.ffff.ffff
start
```

```
stat
ELAM STATUS
=====
Asic 0 Slice 0 Status Triggered
Asic 0 Slice 1 Status Armed
```

ARP - Egress Leaf Elam Results (ereport)

Bridge Domain Settings:
Unicast Routing Disable
ARP Flooding Enabled

Outer L3 Header

Destination IP : 225.0.2.136

Destination is GIPO
(225.0.2.128) + FTAG (0x8)

Inner L3 Header

ARP Sender IP : 192.168.100.11

ARP Target IP : 192.168.100.10

Outer L4 Header

VRF or BD VNID : 14811121(0xE1FFF1)

Contract Result

Contract Drop : no

Frame is flooded in the Bridge Domain!

FINAL FORWARDING LOOKUP

Bits set in Final Forwarding Block: : IFABRIC_EG MC INFRA ENCAP MYTEP **BRIDGE MISS FLOOD**

Lookup Drop

Not Dropped in lookups!

LU drop reason : no drop

ARP – Egress Leaf Port is VPC

Bridge Domain Settings:
Unicast Routing Disable
ARP Flooding Enabled

- Both VPC members receive a flooded copy
- One VPC member is the Designated Forwarder (DF) for the flow
- DF is hashed per flow
- Only DF floods out VPC interfaces

Non-DF Leaf

```
module-1(DBG-elam-insell14)# ereport | grep df | grep vpc  
sug_lub_latch_results_vec.lub4_1.vpc_df: 0x0  
sug_fpx_lookup_vec.lkup.dciptvec.pt.vpc_df: 0x0  
sug_fpc_lookup_vec.fplu_vec.lkup.dciptvec.pt.vpc_df: 0x0  
sug_fpc_lookup_vec.fplu_vec.lkup.dciptvec.pt.vpc_df: 0x0
```

DF Leaf

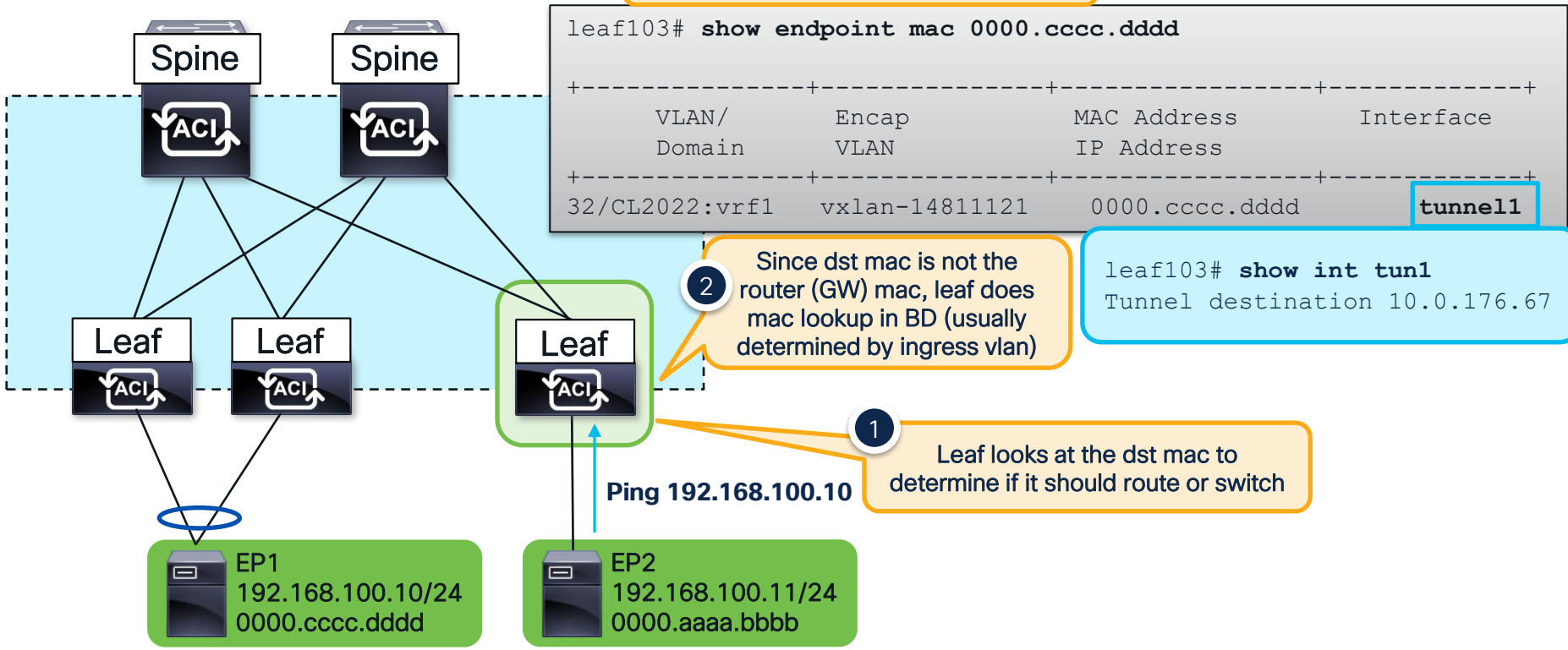
```
module-1(DBG-elam-insell14)# ereport | grep df | grep vpc  
sug_lub_latch_results_vec.lub4_1.vpc_df: 0x1  
sug_fpx_lookup_vec.lkup.dciptvec.pt.vpc_df: 0x1  
sug_fpc_lookup_vec.fplu_vec.lkup.dciptvec.pt.vpc_df: 0x1  
sug_fpc_lookup_vec.fplu_vec.lkup.dciptvec.pt.vpc_df: 0x1
```


Debugging ACI Bridged Flows

Known Unicast – Ingress Leaf

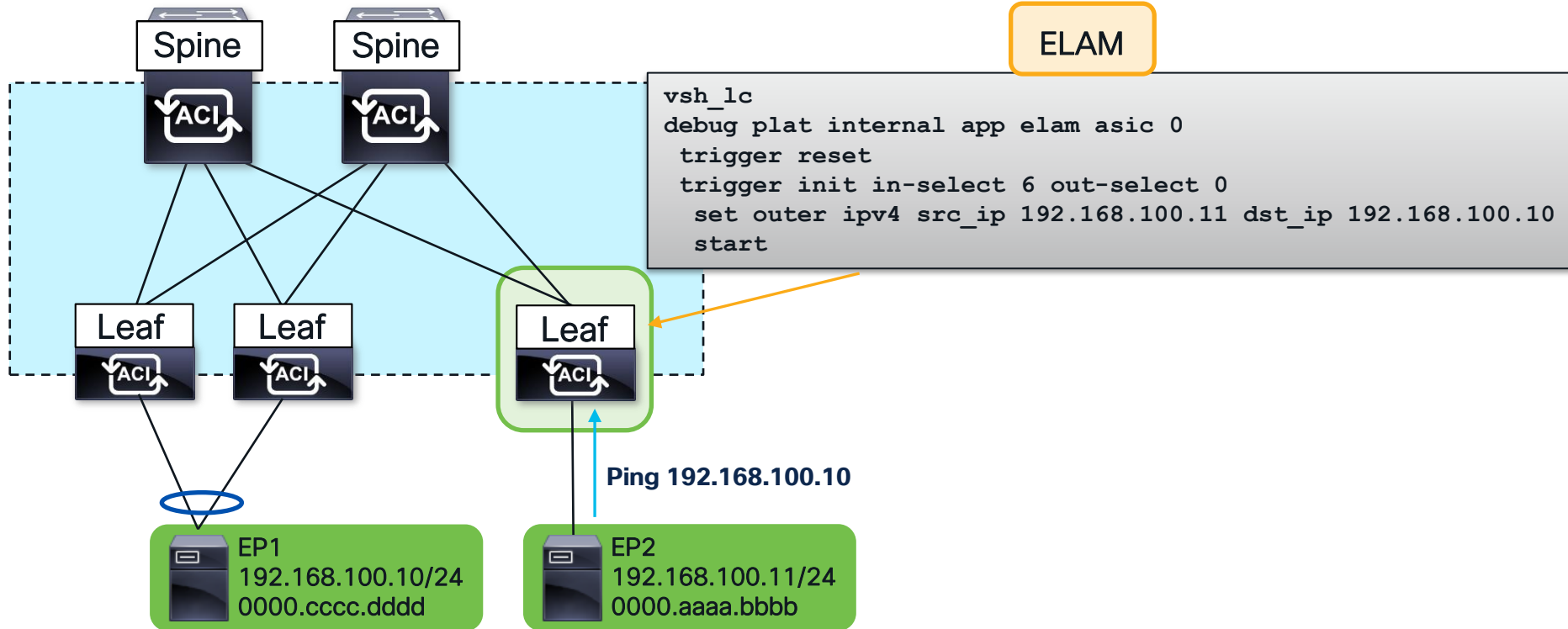
Bridge Domain Settings:
Unicast Routing Disable
Unknown Unicast Flood

Lookup dst mac in ingress BD



Known Unicast – Ingress Leaf

Bridge Domain Settings:
Unicast Routing Disable
Unknown Unicast Flood



Known Unicast – Ingress Leaf

Forwarding Verifications

Bridge Domain Settings:
Unicast Routing Disable
Unknown Unicast Flood

Outer L2 Header

```
-----  
Destination MAC   : 0000.cccc.dddd  
Source MAC        : 0000.aaaa.bbbb  
Access Encap VLAN : 3502 ( 0xDAE )
```

Dest mac that is looked up within BD

Make sure this is the expected vlan

Outer L3 Header

```
-----  
IP Protocol Number : ICMP  
Destination IP      : 192.168.100.10  
Source IP           : 192.168.100.11
```

Dest is tunnel

Other Forwarding Information

```
-----  
Encap Index is valid : yes  
Encap Index          : 34 ( 0x22 )
```

```
show plat internal hal tunnel rtep apd
```

```
=====
```

ifId	IP	RwEncapIdx
18010001	10.0.176.67	22

```
=====
```

Forward to this overlay TEP

FINAL FORWARDING LOOKUP

```
-----  
Bits set in Final Forwarding Block: IFABRIC_IG UC TENANT MYTEP BRIDGE HIT
```

Lookup Drop

```
-----  
LU drop reason       : no drop
```

Not Dropped in lookups!

Unicast + Bridge (L2 lookup) +
Destination Known

Known Unicast – Ingress Leaf

Forwarding Verifications

Bridge Domain Settings:
Unicast Routing Disable
Unknown Unicast Flood

```
ereport | grep "ovector "  
ovector : 152 ( 0x98 )
```

```
show platform internal hal 12 port gpd
```

```
=====
```

IfId	Ifname	As	AP	Sl	Sp	Ss	Ovec
1a01c000	Eth1/29	0	59	2	18	18	98

```
=====
```

Traffic is forwarded out Eth1/29!

Known Unicast – Ingress Leaf

Contract Verification

Bridge Domain Settings:
Unicast Routing Disable
Unknown Unicast Flood

```
Contract Lookup Key
-----
IP Protocol           : ICMP( 0x1 )
L4 Src Port          : 2048( 0x800 )
L4 Dst Port          : 35914( 0x8C4A )
sclass (src pCtag)   : 49154( 0xC002 )
dclass (dst pCtag)   : 49154( 0xC002 )
src pCtag is from local table : yes
Unknown Unicast / Flood Packet : no

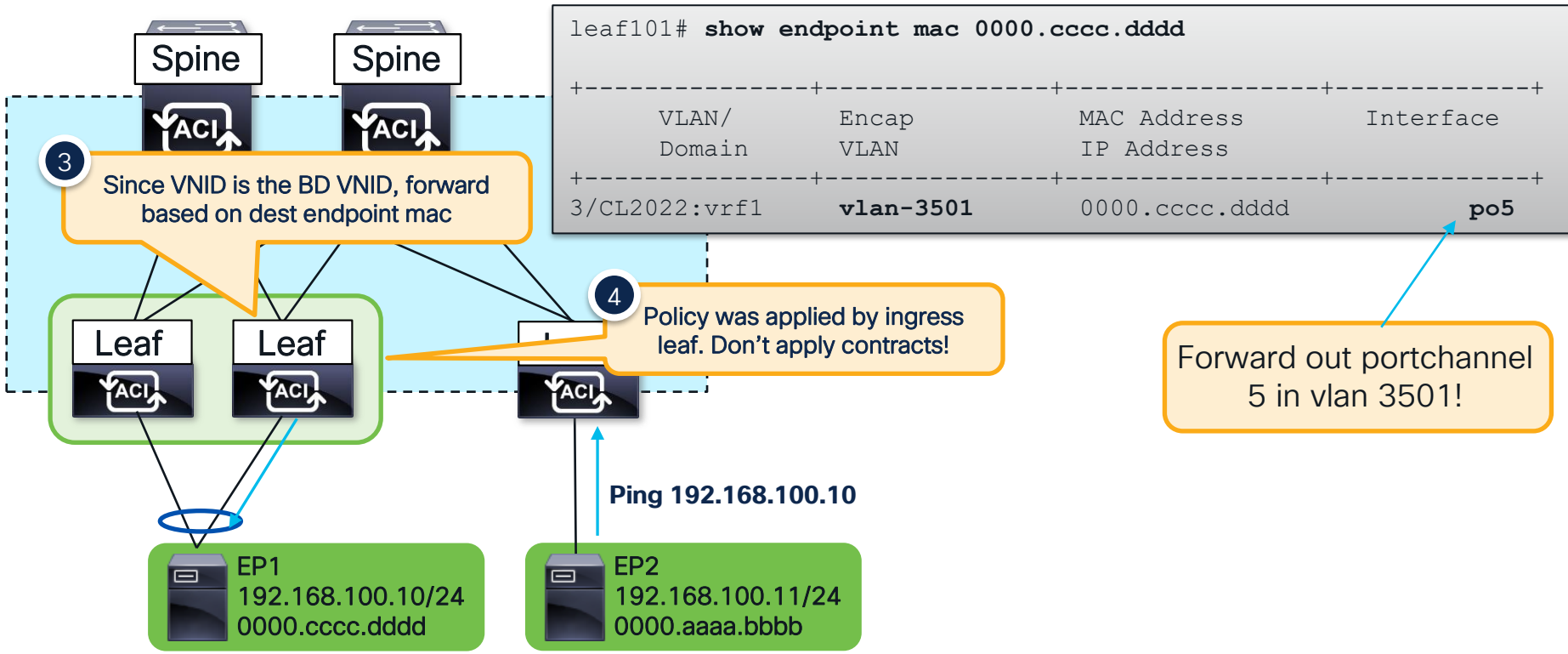
Contract Result
-----
Contract Drop        : no
Contract Applied     : yes
Contract Hit         : yes
Contract Aclqos Stats Index : 131025
( show sys int aclqos zoning-rules | grep -B 9 "Idx: 131025" )
```

Source and Dest EPG is the same. Implicitly permit!
(unless isolation enabled)

Contract Applied and
no Drop!

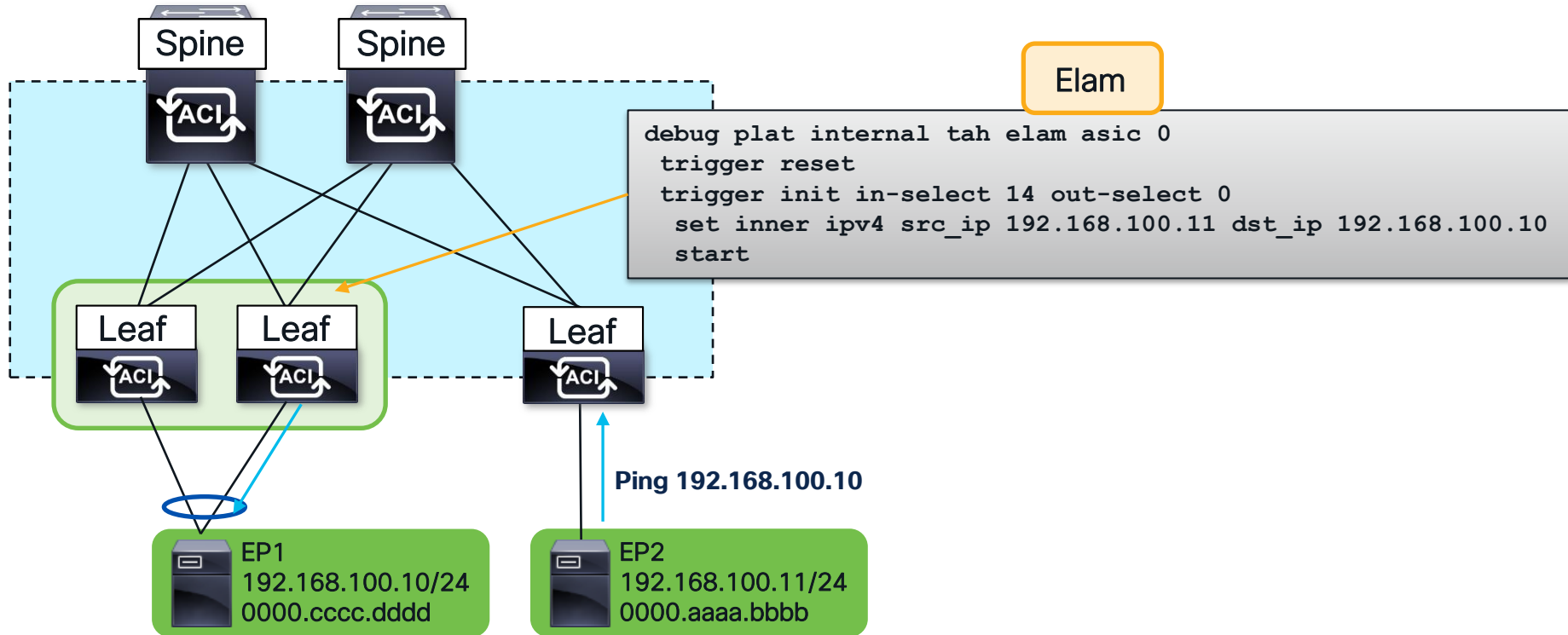
Known Unicast – Egress Leaf

Bridge Domain Settings:
Unicast Routing Disable
Unknown Unicast Flood



Known Unicast – Egress Leaf

Bridge Domain Settings:
Unicast Routing Disable
Unknown Unicast Flood



Known Unicast – Egress Leaf

Bridge Domain Settings:
Unicast Routing Disable
Unknown Unicast Flood

```
Inner L2 Header
-----
Inner Destination MAC : 0000.cccc
Inner L3 Header
-----
Destination IP      : 192.168.100
Outer L4 Header
-----
L4 Type             : iVxL4
Src Policy Applied Bit : 1
Dst Policy Applied Bit : 1
VRF or BD VNID      : 14811121 ( 0xE1FFF1 )
Sideband Information
-----
ovector             : 146 ( 0x92 )
FINAL FORWARDING LOOKUP
-----
Bits set in Final Forwarding Block: IFABRIC_EG UC INFRA ENCAP MYTEP BRIDGE HIT
Lookup Drop
-----
LU drop reason      : no drop
```

Contracts have already been applied. No need to check.

Mac lookup done in bridge domain with this VNID

```
show platform internal hal 12 port gpd
```

IfId	Ifname	As AP	SI	Sp	Ss	Ovec	
1a021000	Eth1/34	0	32	1	9	12	92

Forward out Eth1/34!

Unicast + Bridge (L2 lookup) + Destination Known

Debugging ACI Routed Flows

Known Unicast – Ingress Leaf

Bridge Domain Settings:
Unicast Routing Enabled

Lookup dst IP in ingress VRF

```
leaf103# show endpoint ip 192.168.100.10
+-----+-----+-----+
| VLAN/ | MAC Address | Interface |
| Domain | IP Address  |           |
+-----+-----+-----+
| CL2022:vrf1 | 192.168.100.10 | tunnel1 |
+-----+-----+-----+
```

```
leaf103# show int tun1
Tunnel destination 10.0.176.67
```

2 Since dst mac is the router (GW) mac, leaf does IP lookup in VRF of source IP

1 Leaf looks at the dst mac to determine if it should route or switch

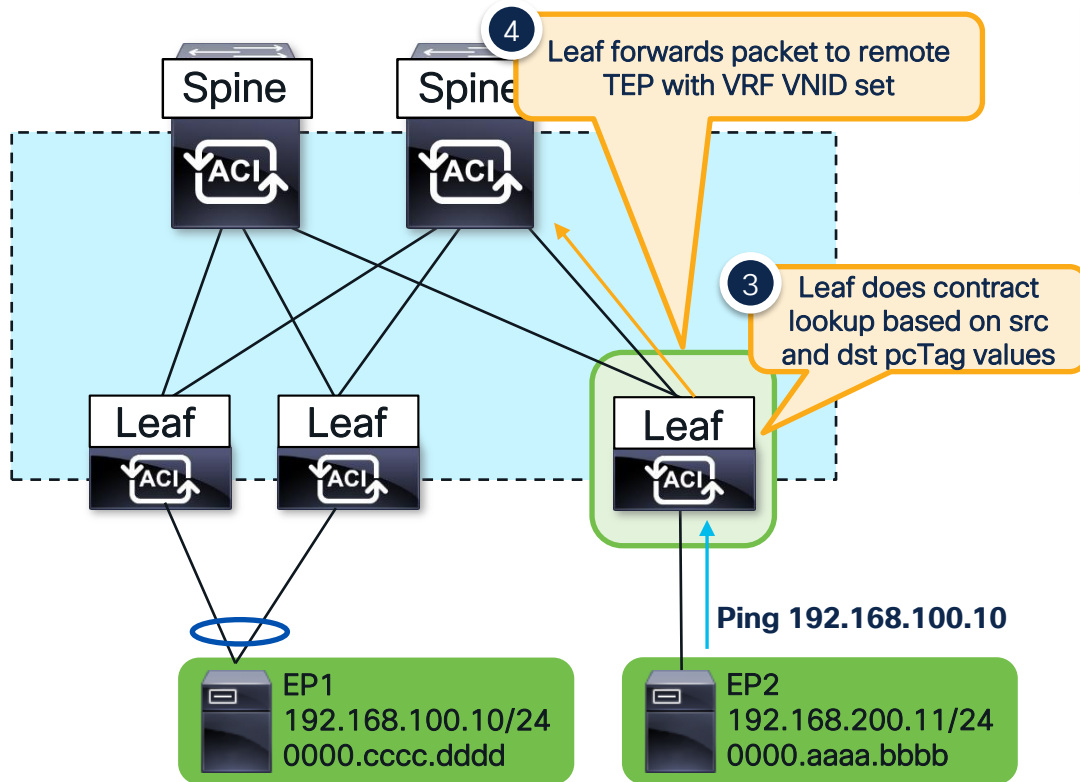
Ping 192.168.100.10

EP1
192.168.100.10/24
0000.cccc.dddd

EP2
192.168.200.11/24
0000.aaaa.bbbb

Known Unicast – Ingress Leaf

Bridge Domain Settings:
Unicast Routing Enabled



Get Sclass

```
103# show sys internal epm endpoint ip
192.168.200.11
!omitted
BD vnid : 16613259 ::: VRF vnid : 2523136
sclass : 32771
```

Get Dclass

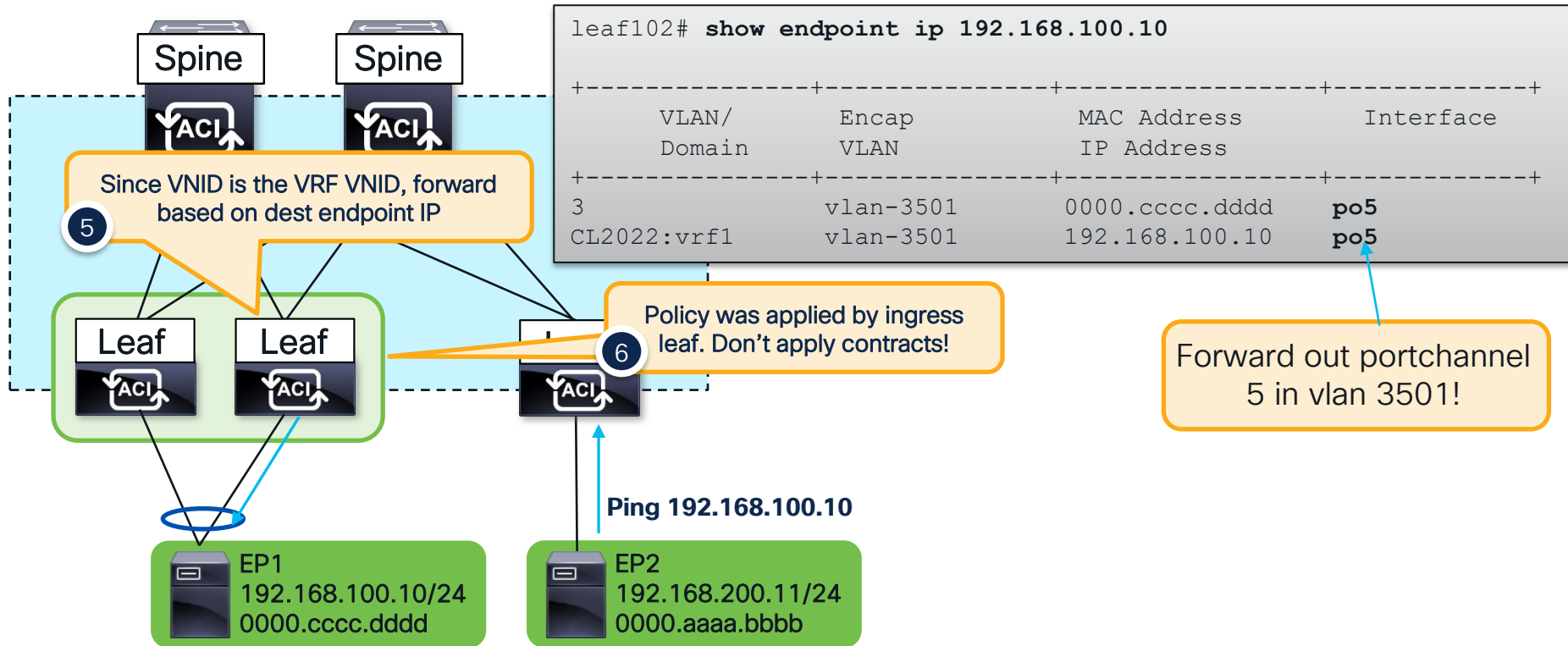
```
103# show sys internal epm endpoint ip
192.168.100.10
!omitted
BD vnid : 0 ::: VRF vnid : 2523136
sclass : 49154
```

Check Contract

```
103# show zoning-rule src-epg 32771
dst-epg 49154 scope 2523136
+-----+-----+-----+
| RuleID | Name | Action |
+-----+-----+-----+
| 4209 | CL2022:allow-all | permit |
+-----+-----+-----+
```

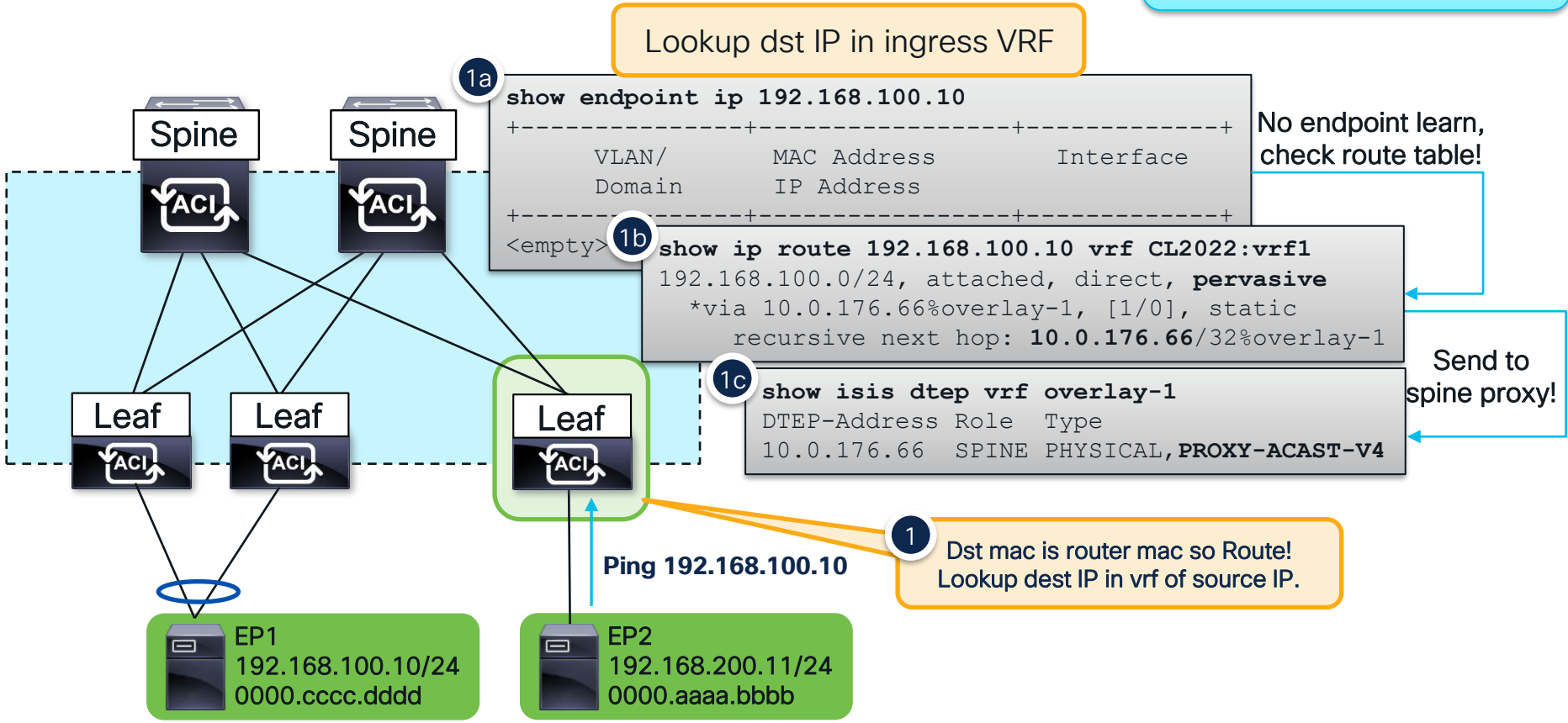
Known Unicast – Egress Leaf

Bridge Domain Settings:
Unicast Routing Enabled



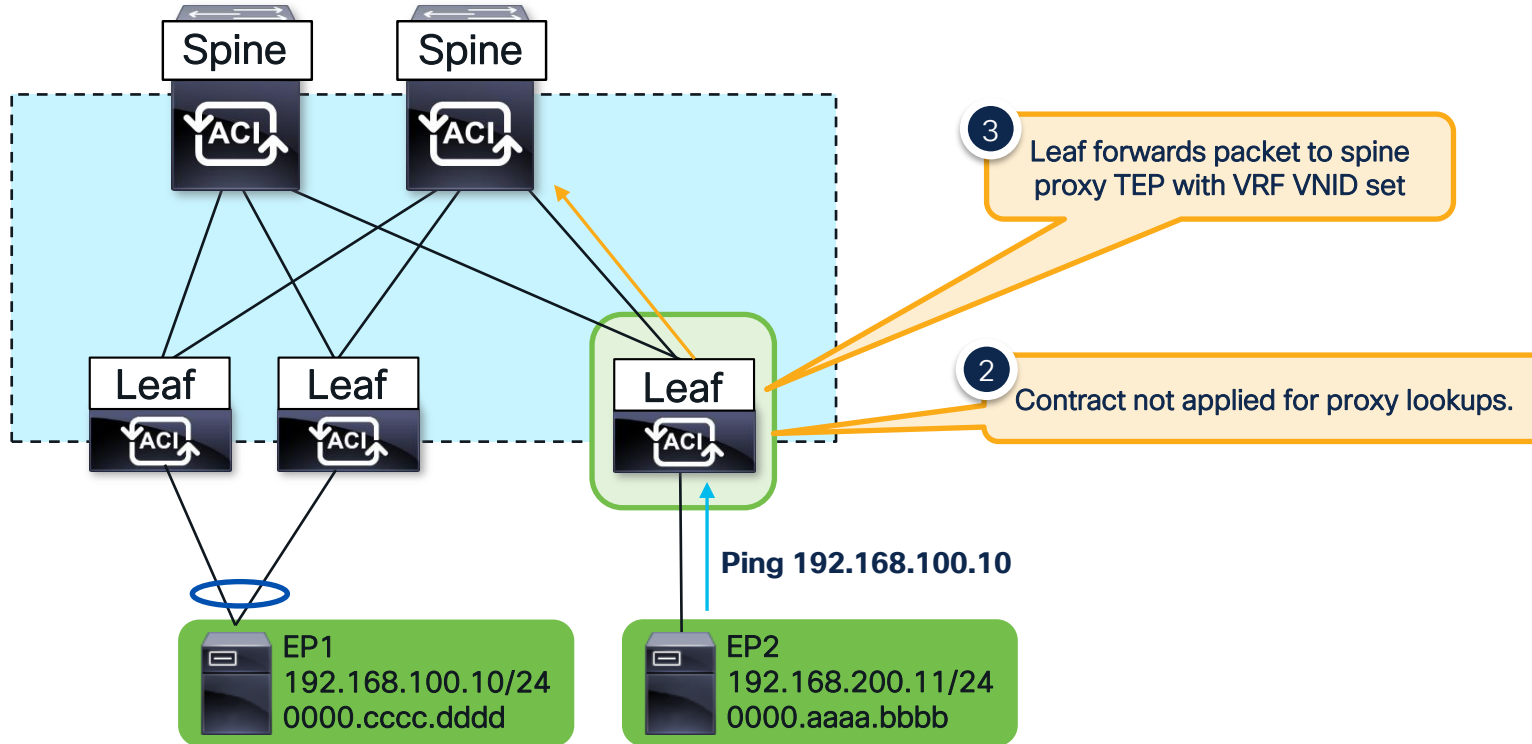
Proxied Unicast – Ingress Leaf

Bridge Domain Settings:
Unicast Routing Enabled



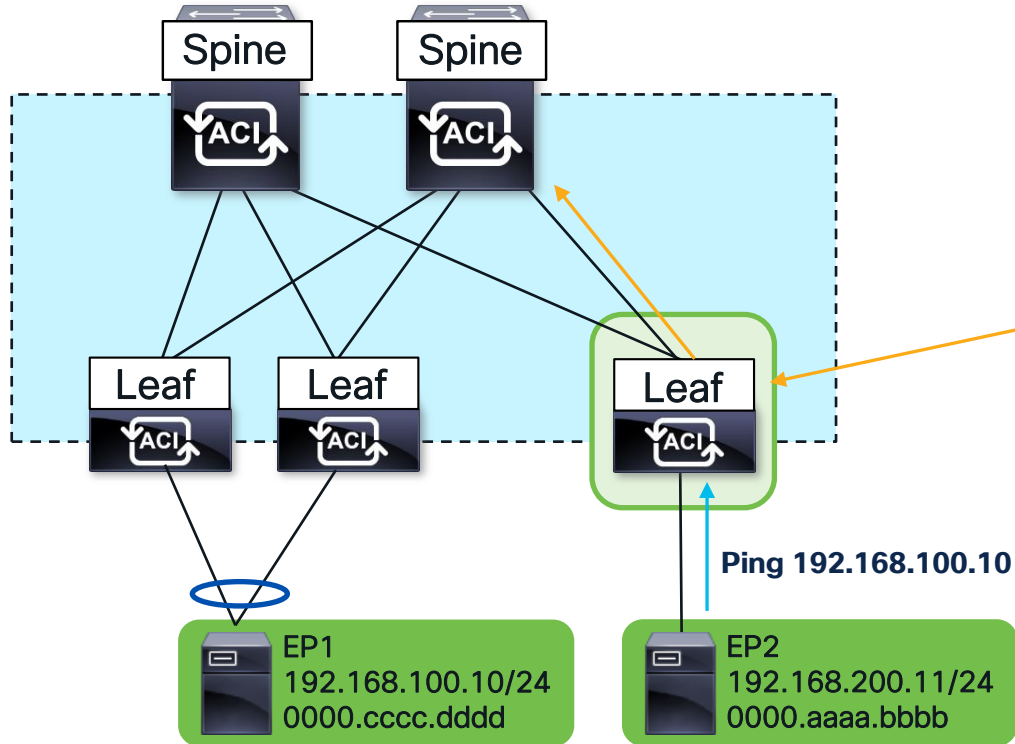
Proxied Unicast – Ingress Leaf

Bridge Domain Settings:
Unicast Routing Enabled



Proxied Unicast – Ingress Leaf

Bridge Domain Settings:
Unicast Routing Enabled



ELAM

```
vsh_lc
debug plat internal app elam asic 0
trigger reset
trigger init in-select 6 out-select 0
set outer ipv4 src_ip 192.168.200.11
set outer ipv4 dst_ip 192.168.100.10
start
stat
ELAM STATUS
=====
Asic 0 Slice 0 Status Triggered
Asic 0 Slice 1 Status Armed
```


Proxied Unicast – Ingress Leaf

Bridge Domain Settings:
Unicast Routing Enabled

Forwarding Verifications

Outer L2 Header

Destination MAC : 0022.BDF8.19FF
Access Encap VLAN : 3769 (0xEB9)

ACI Router Mac. Route this packet!

Make sure this is the expected vlan

Outer L3 Header

Destination IP : 192.168.100.10
Source IP : 192.168.200.11

Dest is tunnel

Other Forwarding Information

Encap Index is valid : **yes**
Encap Index : 1 (0x1)

```
show plat internal hal tunnel rtep apd
=====
ifId      IP           RwEncapIdx
=====
18010007 10.0.176.66 1
```

Forward to this overlay TEP

FINAL FORWARDING LOOKUP

Bits set in Final Forwarding Block: IFABRIC_IG **UC** TENANT MYTEP **ROUTE HIT**

Lookup Drop

LU drop reason : **no drop**

Not Dropped in lookups!

Unicast + Route (L3 lookup) +
L3 Route Found

Proxied Unicast – Ingress Leaf

Forwarding Verifications

Bridge Domain Settings:
Unicast Routing Enabled

```
ereport | grep "ovector "  
ovector : 152 ( 0x98 )
```

```
show platform internal hal 12 port gpd
```

```
=====
```

IfId	Ifname	As AP	Sl	Sp	Ss	Ovec
1a01c000	Eth1/29	0	59	2	18 18	98

```
=====
```

Traffic is forwarded out Eth1/29!

Proxied Unicast – Ingress Leaf

Contract Verification

Bridge Domain Settings:
Unicast Routing Enabled

```
Contract Lookup Key
-----
IP Protocol           : ICMP( 0x1 )
L4 Src Port          : 2048( 0x800 )
L4 Dst Port          : 31219( 0x79F3 )
sclass (src pCtag)   : 32771( 0x8003 )
dclass (dst pCtag)   : 1( 0x1 )
src pCtag is from local table : yes
Unknown Unicast / Flood Packet : no
```

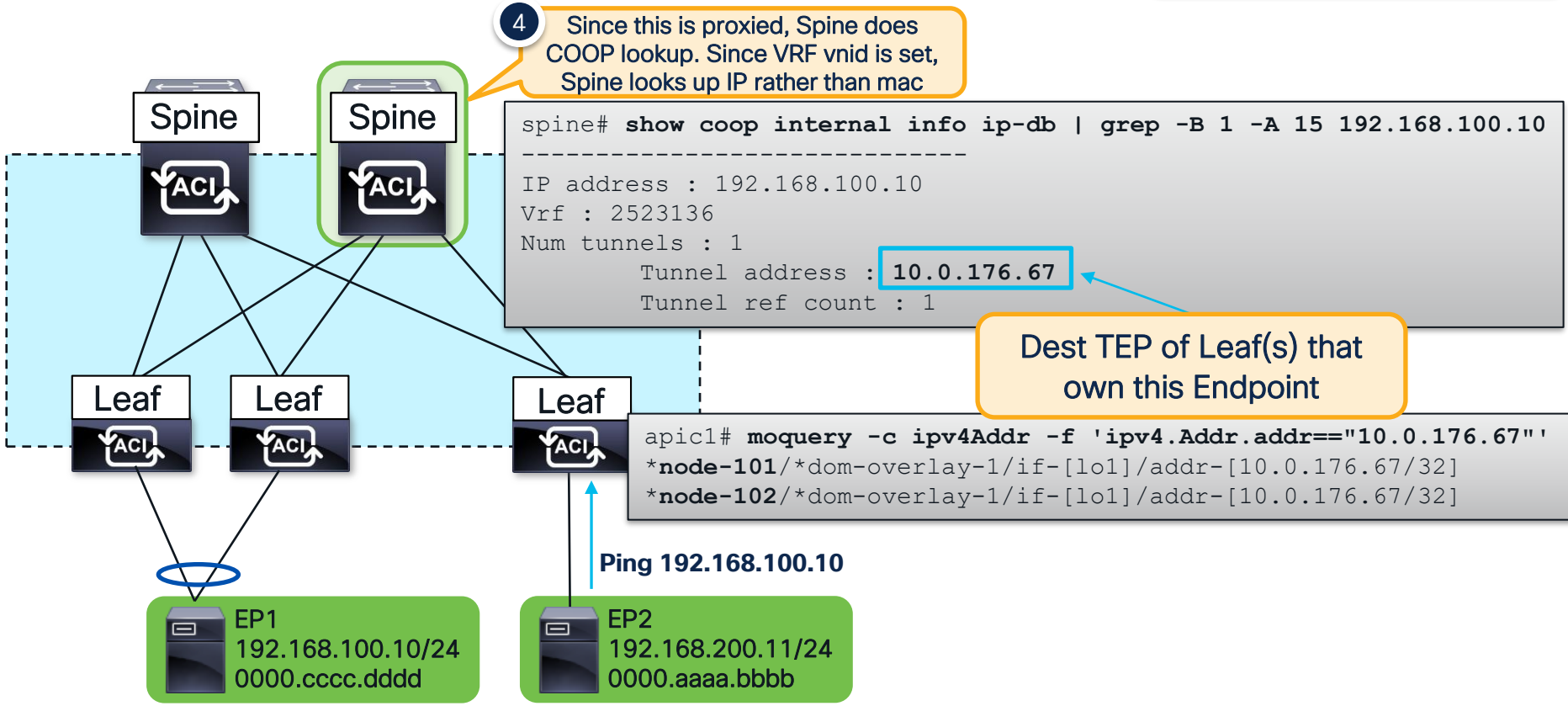
Dest EPG is 1 for fabric owned subnets

```
Contract Result
-----
Contract Drop        : no
Contract Applied     : no
Contract Hit         : yes
Contract Aclqos Stats Index : 131025
```

Contract not applied since this is proxied!

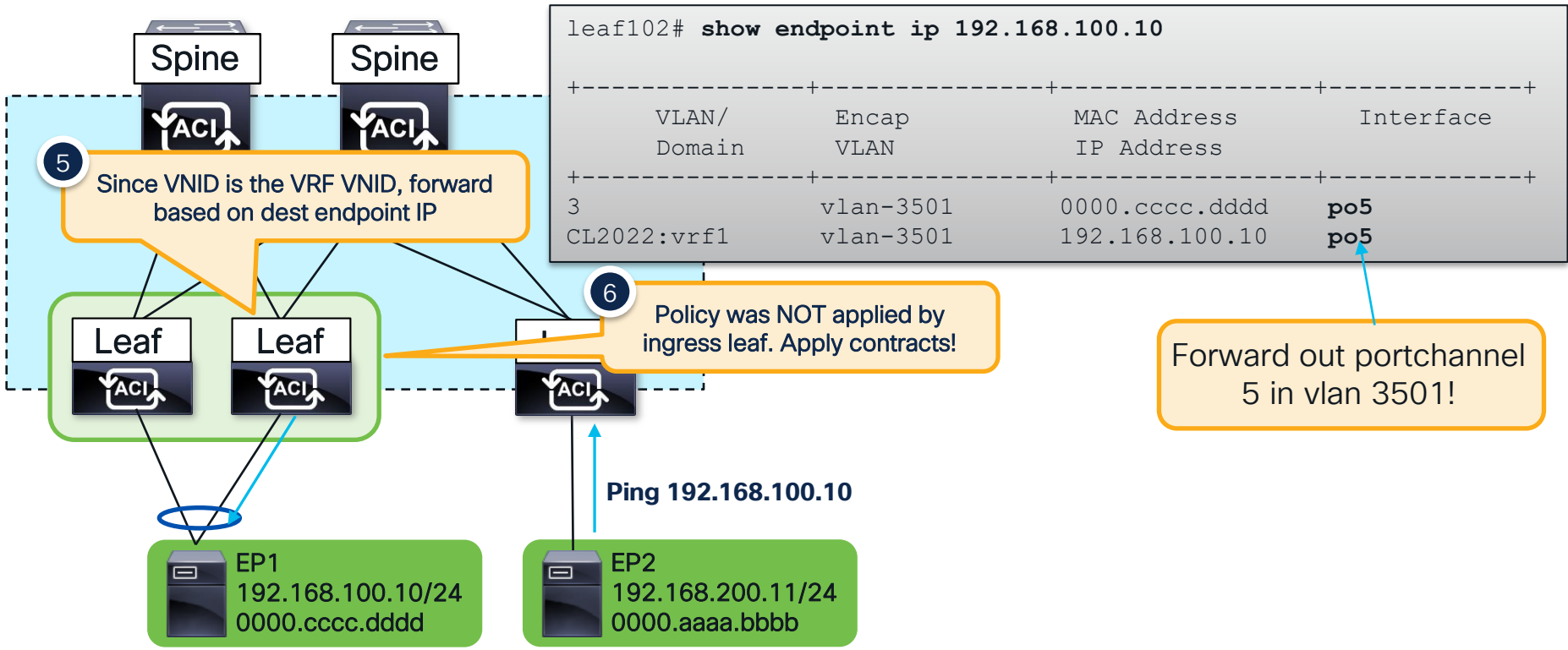
Proxied Unicast – Spine

Bridge Domain Settings:
Unicast Routing Enabled



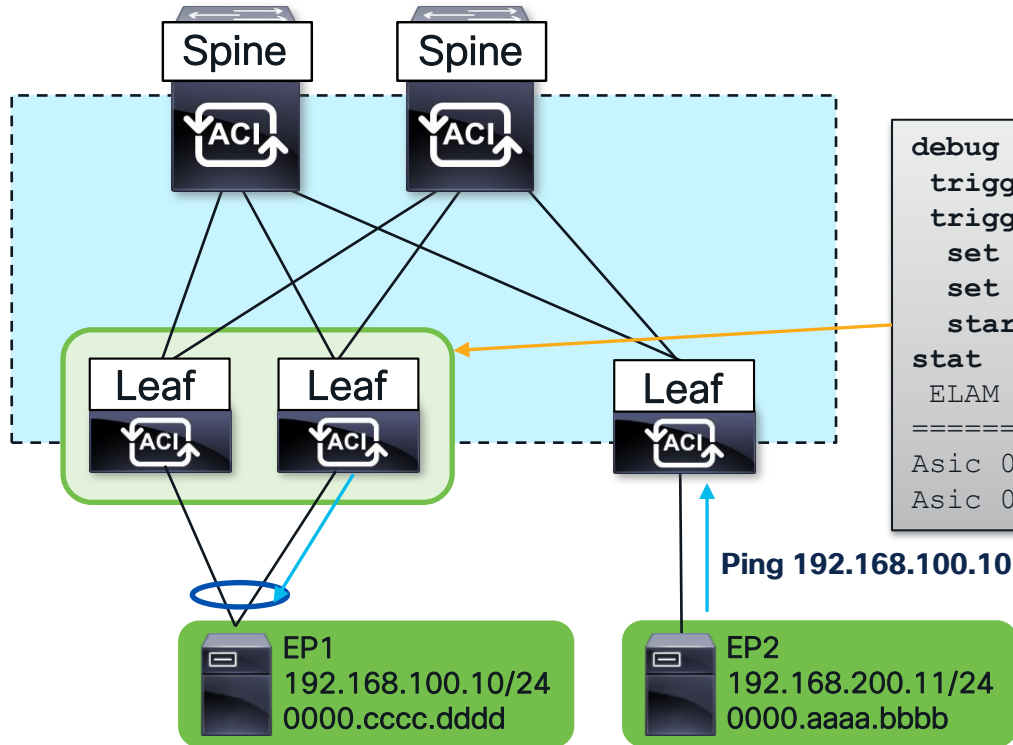
Proxied Unicast – Egress Leaf

Bridge Domain Settings:
Unicast Routing Enabled



Proxied Unicast – Egress Leaf

Bridge Domain Settings:
Unicast Routing Enabled



Elam

```
debug plat internal tah elam asic 0
trigger reset
trigger init in-select 14 out-select 0
set inner ipv4 src_ip 192.168.200.11
set inner ipv4 dst_ip 192.168.100.10
start
stat
ELAM STATUS
=====
Asic 0 Slice 0 Status Triggered
Asic 0 Slice 1 Status Armed
```

Proxied Unicast – Egress Leaf

Contract Verification

Bridge Domain Settings:
Unicast Routing Enabled

```
Contract Lookup Key
-----
IP Protocol           : ICMP( 0x1 )
L4 Src Port          : 2048( 0x800 )
L4 Dst Port          : 33226( 0x81CA )
sclass (src pCtag)   : 32771( 0x8003 )
dclass (dst pCtag)   : 49154( 0xC002 )
src pCtag is from local table : no
Unknown Unicast / Flood Packet : no

Contract Result
-----
Contract Drop        : no
Contract Applied     : yes
Contract Hit         : yes
Contract Aclqos Stats Index : 131025
```

Source and Dest EPG used
for contract lookup.

Contract Applied and
no Drop!

But how do I know which
contract this is actually hitting?

Proxied Unicast – Egress Leaf

Bridge Domain Settings:
Unicast Routing Enabled

Contract Verification

```
Contract Result
-----
Contract Drop           : no
Contract Applied       : yes
Contract Hit           : yes
Contract Aclqos Stats Index : 81836
```

Hardware Index of matching contract

Run this from vsh_lc

Zoning-rule ID

```
show sys int aclqos zoning-rules | grep -B 9 "Idx: 81836"
=====
Rule ID: 4234 Scope 16 Src EPG: 32771 Dst EPG: 49154 Filter
532
=====

=== SDK Info ===
Result/Stats Idx: 81836
```

Run this from normal shell

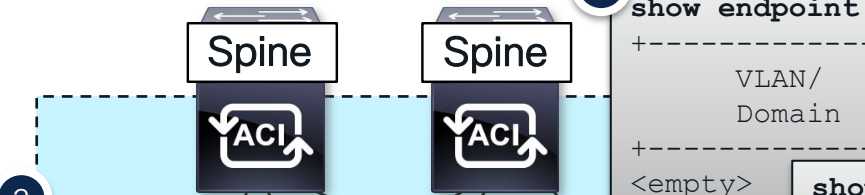
```
show zoning-rule rule-id 4234
+-----+-----+-----+-----+-----+-----+-----+
| Rule ID | SrcEPG | DstEPG | FilterID | Scope | Name | Action |
+-----+-----+-----+-----+-----+-----+-----+
| 4163 | 32771 | 49154 | 532 | 2523136 | CL2022:allow-all | permit |
+-----+-----+-----+-----+-----+-----+-----+
```

Traffic hit this contract!

L3Out Destination – Ingress Leaf

Bridge Domain Settings:
Unicast Routing Enabled

Lookup dst IP in ingress VRF



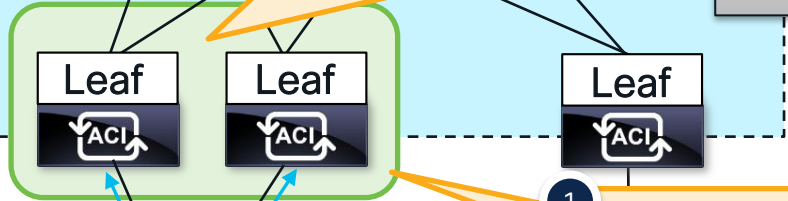
2 Since dst mac is the router (GW) mac, leaf does IP lookup in VRF of source IP

```
2a show endpoint ip 10.99.99.100
-----+-----+-----+
VLAN/   MAC Address   Interface
Domain  IP Address
-----+-----+-----+
<empty>
```

No endpoint learn,
check route table!

```
2b show ip route 10.99.99.100 vrf CL2022:vrf1
10.99.99.0/24, ubest/mbest: 1/0
 *via 10.0.64.70%overlay-1, [200/20], bgp-65100
  recursive next hop: 10.0.64.70/32%overlay-1
```

Send to BL
PTEP!



```
2c acidiag fvnread | grep 10.0.64.70
Name      IP Address      Role
-----+-----+-----+
leaf103   10.0.64.70/32  leaf
```

1 Leaf looks at the dst mac to determine if it should route or switch

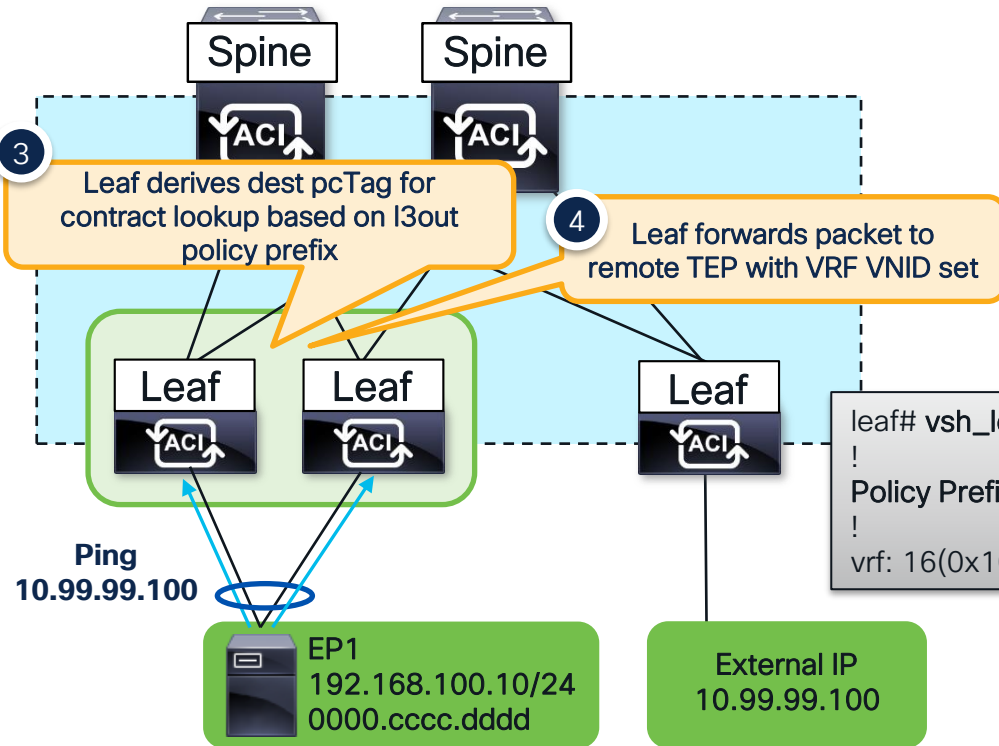
Ping
10.99.99.100

EP1
192.168.100.10/24
0000.cccc.dddd

External IP
10.99.99.100

L3Out Destination – Ingress Leaf

Bridge Domain Settings:
Unicast Routing Enabled



External EPGs

External EPGs		
Name	Description	pcTag
all	10.99.99.0/24 Network	32772

```
leaf# vsh_lc -c "show forwarding route 10.99.99.100 platf vrf CL2022:vrf1"
!
Policy Prefix 10.99.99.0/24
!
vrf: 16(0x10), routed_if: 0x0 epc_class: 32772(0x8004)
```

L3Out Destination – Egress Leaf

Bridge Domain Settings:
Unicast Routing Enabled

Lookup dst IP in received VRF

5 Since received VNID is the VRF VNID, forward based on dest endpoint IP

```
5a show endpoint ip 10.99.99.100
-----+-----+-----+
VLAN/   MAC Address   Interface
Domain  IP Address
-----+-----+-----+
<empty>
```

No endpoint learn, check route table!

```
5b show ip route 10.99.99.100 vrf CL2022:vrf1
10.99.99.0/24, ubest/mbest: 1/0
*via 10.55.0.100, vlan25, [110/20], ospf, type-2
```

```
5c show ip arp 10.55.0.100 vrf CL2022:vrf1
Address      MAC Address   Interface
10.55.0.100  0005.73ff.593c  vlan25
```

```
5d show mac address addr 0005.73ff.593c vl 25
-----+-----+-----+
VLAN      MAC Address   Ports
-----+-----+-----+
* 25      0005.73ff.593c  eth1/27/4
```

Forward based on ARP and MAC Adjacencies

6 Policy was applied by ingress leaf. No need to apply contracts

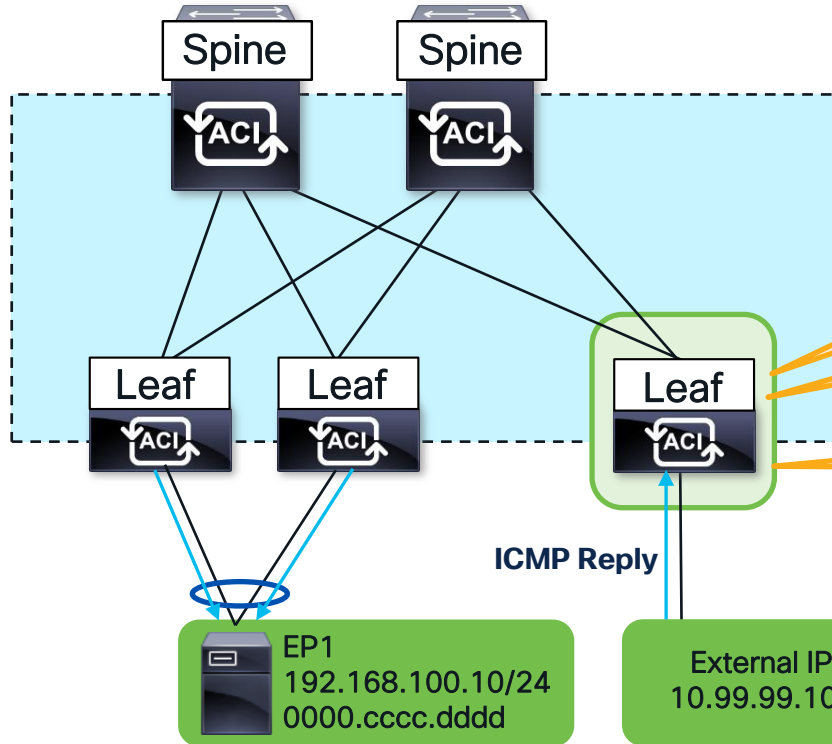
Ping 10.99.99.100

EP1
192.168.100.10/24
0000.cccc.dddd

External IP
10.99.99.100

L3Out Source – Ingress Border Leaf

Bridge Domain Settings:
Unicast Routing Enabled



2b

If dest IP is not learned endpoint and subnet is BD subnet, proxy!

2a

Forward based on longest prefix-match within source VRF. EP learns are always longest.

1

If VRF is in ingress mode, BL doesn't apply policy

Refer back to the Routed Known Unicast and Proxied Unicast for more verifications

Troubleshooting Tips

Troubleshooting TIP

Check Endpoint Table
before Routing Table

When Troubleshooting Layer 3 Flows Always...

1) Check if there is an Endpoint Learn

```
show endpoint ip <addr>  
show system internal epm endpoint ip <addr>
```

If not then...

2) Check if there is a BD (pervasive) static route

If not then...

3) Check if there is an External Route

```
show ip route x.x.x.x/y vrf <name>
```

Single point to validate forwarding & security
vsh_lc -c "show platform internal hal I3 routes"

Troubleshooting TIP

If the Ingress leaf can resolve dclass it will apply the contract.

Internal EPG classification

1) Check if there is an Endpoint Learn

```
show system internal epm endpoint ip <addr>
```

If not then...

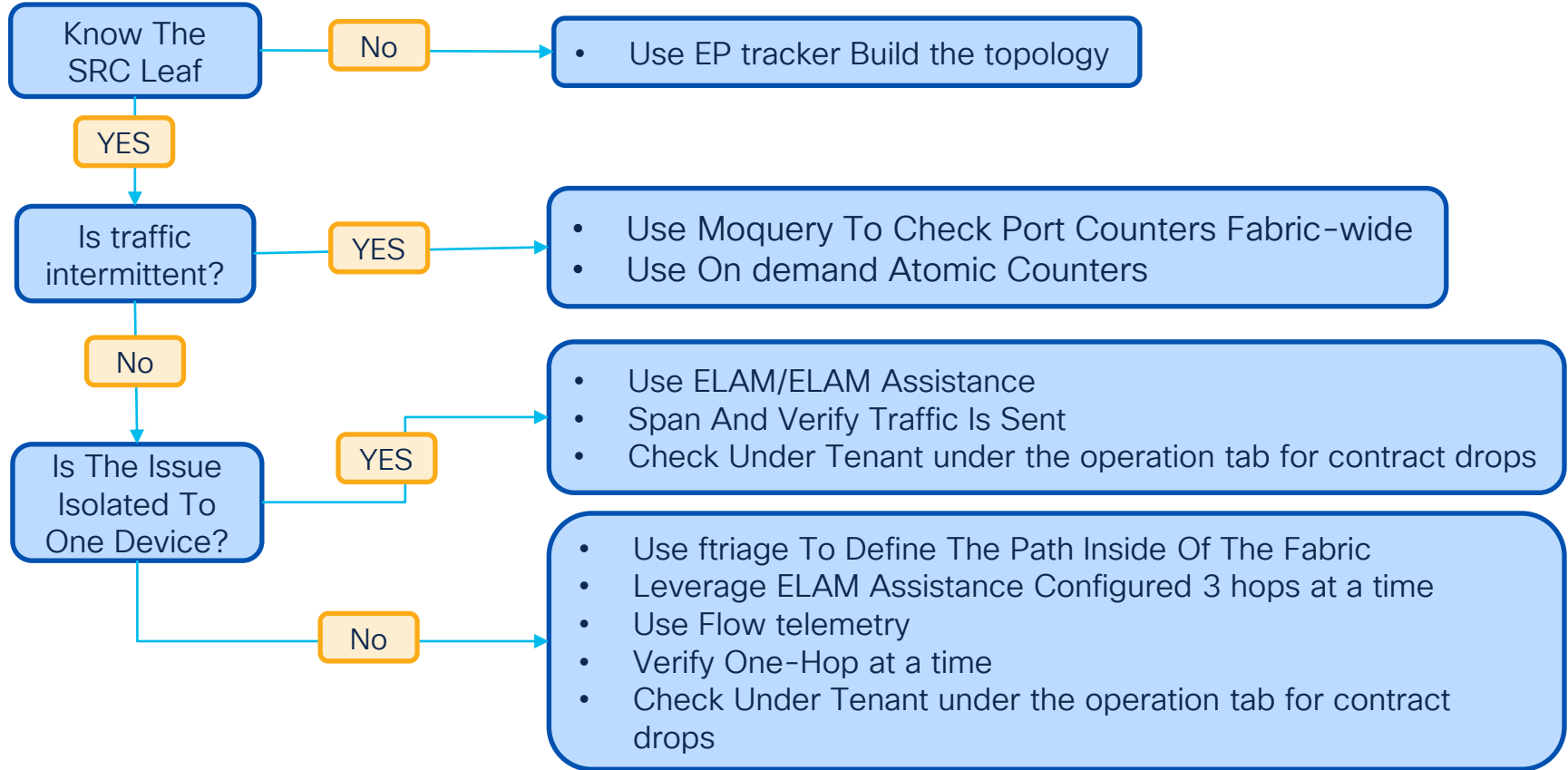
2) Verify the longest prefix match

```
leaf101# vsh_lc -c "show forwarding route  
10.99.99.100 platform vrf CL2022:vrf1"
```

```
leaf101# show zoning-prefixes | grep 10.99.99.
```

Single point to validate forwarding & security
vsh_lc -c "show platform internal hal I3 routes"

Tools Decision Map





The bridge to possible

Thank you

CISCO *Live!*

The Cisco Live! logo features the word "CISCO" in a bold, black, sans-serif font, followed by "Live!" in a black, cursive script font. The background of the entire image is a vibrant, multi-colored abstract pattern of overlapping, wavy bands in shades of red, orange, yellow, green, and blue, creating a sense of motion and energy.

CISCO *Live!*

Let's go