EX硬件：ACI数据包转发深入探讨。

目录

简介

本文档介绍使用以应用为中心的基础设施(ACI)中基于“EX”的ACI交换机的不同转发方案。它将展示如何验证硬件是否已正确编程，以及我们如何将数据包转发到适当终端组(EPG)中的正确目标终端(EP)。

先决条件

要求

本文档没有任何特定的要求。

使用的组件

本文档中的信息基于下列硬件和软件版本：

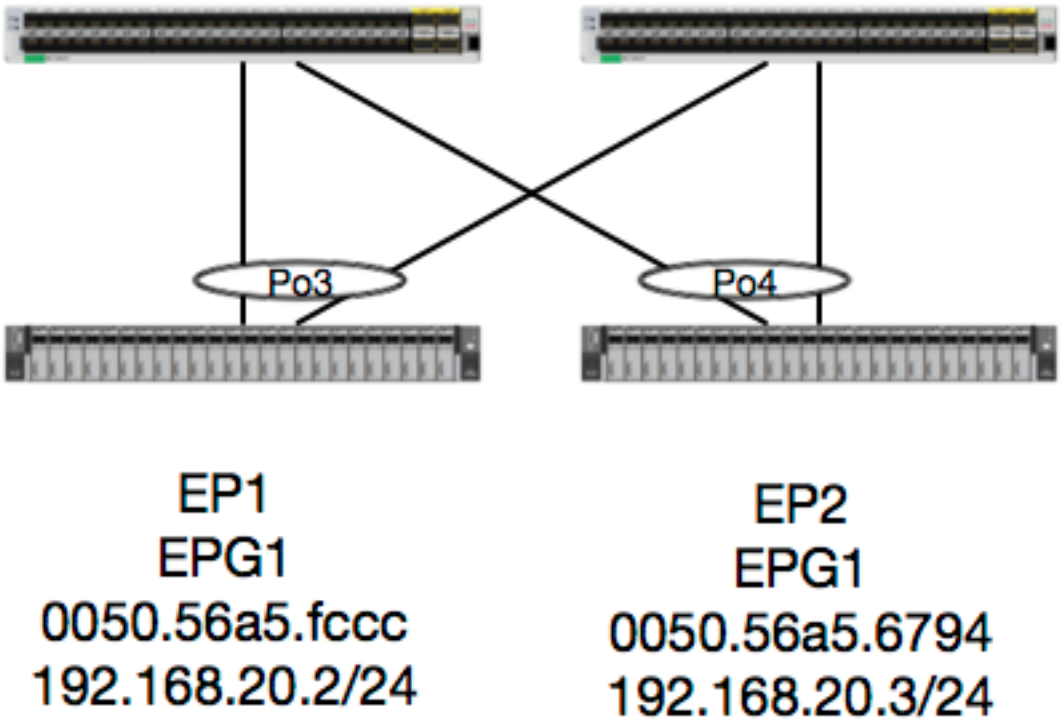
- ACI交换矩阵，由使用EX硬件的两台主干交换机和两台枝叶交换机组成
- ESXi主机，带有两个上行链路，可连接到每台枝叶交换机
- Nexus 5000设备充当路由器。
- 用于初始设置的应用策略基础设施控制器(APIC)

本文档中的信息都是基于特定实验室环境中的设备编写的。本文档中使用的所有设备最初均采用原始（默认）配置。如果您使用的是真实网络，请确保您已经了解所有命令的潜在影响。

场景

同一EPG/相同枝叶 — 交换帧中的2个EP

拓扑



根据此拓扑，从EP1到EP2的流是L2流，应在源流量进入的任何枝叶上本地交换。 要检查第2层(L2)的流，首先需要确定交换机是否以及在何处接收帧的mac地址表：

```
leaf4# show mac address-table | grep fccc
* 30       0050.56a5.fccc    dynamic      -       F    F    po3
leaf4# show mac address-table | grep 6794
* 30       0050.56a5.6794    dynamic      -       F    F    po4
```

为了查看封装vlan，我们还可以检查EP数据库：

```
leaf4# show endpoint mac 0050.56a5.fccc
Legend:
 O - peer-attached    H - vtep            a - locally-aged     S - static
 V - vpc-attached     p - peer-aged       L - local            M - span
 s - static-arp       B - bounce
```

```
+----------------------------------+--------------+-----------------+-------------+----------
---+
      VLAN/                           Encap           MAC Address       MAC Info/
Interface
      Domain                          VLAN            IP Address        IP Info
+----------------------------------+--------------+-----------------+-------------+----------
---+
30                                       vlan-2268    0050.56a5.fccc LV
po3
Joey-Tenant:Joey-Internal                vlan-2268      192.168.20.2 LV
po3


calo2-leaf4# show endpoint mac 0050.56a5.6794
Legend:
 O - peer-attached    H - vtep             a - locally-aged     S - static
 V - vpc-attached     p - peer-aged        L - local            M - span
 s - static-arp       B - bounce
+----------------------------------+--------------+-----------------+-------------+----------
---+
      VLAN/                           Encap           MAC Address       MAC Info/
Interface
      Domain                          VLAN            IP Address        IP Info
+----------------------------------+--------------+-----------------+-------------+----------
---+
30                                       vlan-2268    0050.56a5.6794 LV
po4
Joey-Tenant:Joey-Internal                vlan-2268      192.168.20.3 LV
po4
```

我们知道FD_VLAN 30匹配，但我们始终可以在软件中验证映射：

```
leaf4# show vlan extended | grep 2268
 30   enet  CE         vlan-2268
```

当然，我们可以检查硬件，确保VLAN 30作为前面板封装映射到VLAN 2268。

```
leaf4# vsh_lc
module-1# show system internal eltmc info vlan 30


            vlan_id:             30   :::      hw_vlan_id:             22
          vlan_type:        FD_VLAN   :::         bd_vlan:             28
   access_encap_type:         802.1q   :::    access_encap:           2268
   fabric_encap_type:          VXLAN   :::    fabric_encap:          11960
              sclass:          32778   :::           scope:             11
            untagged:              0
     acess_encap_hex:          0x8dc   :::  fabric_enc_hex:         0x2eb8
     pd_vlan_ft_mask:            0x8
    fd_learn_disable:              0
        qos_class_id:              0   :::      qos_pap_id:              0
          qq_met_ptr:             25   :::      ipmc_index:              0
   ingressBdAclLabel:              0   ::: ingBdAclLblMask:              0
    egressBdAclLabel:              0   ::: egrBdAclLblMask:              0
         qos_map_idx:              0   :::     qos_map_pri:              0
        qos_map_dscp:              0   :::      qos_map_tc:              0
        vlan_ft_mask:          0xe30
           hw_bd_idx:              0   :::      hw_epg_idx:          11267
         intf_count:               2   ::: glbl_scp_if_cnt:              2

<SNIPPED>
```

在软件学习的情况下，我们还可以验证硬件编程了这些EP的L2信息。 在新硬件中，硬件抽象层

(HAL)是硬件的软件状态。 HAL的工作是接收软件编程请求并将其推送到硬件。

为了查看有关终端的L2硬件信息，我们可以查看HAL中的L2表，查看给定MAC地址：

```
leaf4# vsh_lc
module-1# show platform internal hal ep l2 mac 0050.56a5.fccc
LEGEND:
-------
BDId:              BD Id                                                 BD Name:        BD
Name
T:                 EP Type (Pl: Physical Vl: Virtual Xr: Remote           EP Mac:         Mac
L2 IfId:           L2 Interface                                          L2 IfName:      L2
IfName
FDId:              FD Id                                                 FD Name:        FD
Name
S Class:           S Class                                               Age Intvl:      Age
Interval
P A:               Packet Action (F: Forward, T: Trap to CPU,
                                  L: Log & Forward, D: Drop, N: None)
S T:               Static Ep                                             S E:
Secure EP
L D:               Learn Disable                                         B N D:          Bind
Notify Disable
E N D:             Epg Notify Disable                                    B E:
Bounce Enable
I D L:             IVxlan Dont Learn                                     SPI:
Source Policy Incomplete
DPI:               Dest Policy Incomplete                                SPA:
Source Policy Applied
DPA:               Dest Policy Applied                                   DSS:            Dest
Shared Service
IL:                Is Local                                              VUB:            Vnid
Use Bd
SO:                SA Only

 L2 EP Count: 1

================================================================================================
======================
                                                                                          B E
I S D S D D   V
     BD          EP               L2       L2              FD        S     Age   P S S L N N
B D P P P P S I U S
BdId Name      T  Mac              IfId     Ifname      FDId Name     Class Intvl A T E D D D
E L I I A A S L B O
================================================================================================
======================
1c   BD-28     Pl 00:50:56:a5:fc:cc 16000002 Po3         1e   FD-30     800a  29f   F 0 0 0 1 0
0 0 0 0 0 0 0 1 0 0


module-1# show platform internal hal ep l2 mac 0050.56a5.6794
================================================================================================
======================
                                                                                          B E
I S D S D D   V
     BD          EP               L2       L2              FD        S     Age   P S S L N N
B D P P P P S I U S
BdId Name      T  Mac              IfId     Ifname      FDId Name     Class Intvl A T E D D D
E L I I A A S L B O
================================================================================================
======================
```

```
1c   BD-28     Pl 00:50:56:a5:67:94 16000003 Po4          1e   FD-30     800a  29f   F 0 0 0 1 0
0 0 0 0 0 0 0 1 0 0
```

既然我们已经映射了硬件，我们来执行ELAM并查看数据包的去向。

ELAM

```
leaf4# vsh_lc
module-1# debug platform internal tah elam asic 0
module-1(DBG-TAH-elam)# trigger reset
module-1(DBG-TAH-elam)# trigger init in-select 6 out-select 0
module-1(DBG-TAH-elam-insel6)# set outer l2 src_mac 0050.56a5.fccc dst_mac 0050.56a5.6794
module-1(DBG-TAH-elam-insel6)# start
module-1(DBG-TAH-elam-insel6)# stat
ELAM STATUS
===========
Asic 0 Slice 0 Status Armed
Asic 0 Slice 1 Status Triggered

module-1(DBG-TAH-elam-insel6)# report | grep ovec
  sug_elam_out_sidebnd_no_spare_vec.ovector_idx: 0x9E
```

很好，因此Leaf4在Asic 0切片1上收到该帧。在新硬件上使用ELAM时，有一个新字段在故障排除时非常重要：ovector_idx。 此索引是帧/数据包应从中转发出去的物理端口索引。 一旦您拥有ovector_idx，我们可以使用此命令查找它映射到的端口：

```
module-1(DBG-TAH-elam-insel6)# show platform internal hal l2 port gpd
Legend:
-------
IfId:       Interface Id                                IfName:      Interface Name
I P:        Is PC Mbr                                   IfId:           Interface Id
Uc PC Cfg:      UcPcCfg Idx                             Uc PC MbrId:    Uc Pc Mbr Id
As:             Asic                                    AP:             Asic Port
Sl:             Slice                                   Sp:             Slice Port
Ss:             Slice SrcId                             Ovec:           Ovector (slice |
srcid)
L S:            Local Slot                              Reprogram:
L3:             Is L3
    P:    PifTable                                      Xla Idx:        Xlate Idx
    RP:   Rw PifTable                                   Ovx Idx:        OXlate Idx
    IP:   If Profile Table                              N L3:           Num. of L3 Ifs
    RS:   Rw SrcId Table                                NI L3:          Num. of Infra L3 Ifs
    DP:   DPort Table                                   Vif Tid:        Vif Tid
    SP:   SrcPortState Table                            RwV Tid:        RwVif Tid
    RSP:  RwSrcPortstate Table                          Ing Lbl:        Ingress Acl Label
    UC:   UCPcCfg                                       Egr Lbl:        Egress Acl Label
    UM:   UCPcMbr                                       Reprogram:
PROF ID:        Lport Profile Id
    VS:   VifStateTable                                 HI:             LportProfile Hw
Install
    RV:   Rw VifTable
Num. of Sandboxes: 1

Sandbox_ID: 0, BMP: 0x0
 Port Count: 8

================================================================================================
=========================================
                        Uc   Uc                          |       Reprogram       |
| Rep |
                     I PC   Pc                       L |   R I R D    R  U U X | L Xla Ovx N
```

```
NI Vif    RwV    Ing  Egr  | V R | PROF H
IfId     Ifname      P Cfg  MbrID As AP Sl Sp Ss Ovec S | P P P S P Sp Sp C M L | 3 Idx Idx L3
L3 Tid    Tid    Lbl  Lbl  | S V | ID   I
=============================================================================================
==========================================
1a004000 Eth1/5      1 0    1d    0  d  0  c  18 18   1   0 0 0 0 0 0  0  0 0 0   0 0   0   0  0
-      -      800  0      0 1   0    0
1a005000 Eth1/6      1 0    b     0  e  0  d  1a 1a   1   0 0 0 0 0 0  0  0 0 0   0 0   0   0  0
-      -      800  0      0 1   0    0
1a006000 Eth1/7      0 26   5     0  f  0  e  1c 1c   1   0 0 0 0 0 0  0  0 0 0   0 0   0   0  0
D-256  -      800  0      0 1   e    0
1a007000 Eth1/8      0 2e   7     0  10 0  f  1e 1e   1   0 0 0 0 0 0  0  0 0 0   0 0   0   0  0
D-84   -      800  0      0 1   30   0
1a01e000 Eth1/31     1 0    2d    0  37 1  e  1c 9c   1   0 0 0 0 0 0  0  0 0 0   0 0   0   0  0
-      -      0    0      0 1   0    0
1a01f000 Eth1/32     1 0    3d    0  38 1  f  1e 9e   1   0 0 0 0 0 0  0  0 0 0   0 0   0   0  0
-      -      0    0      0 1   0    0
1a030000 Eth1/49     0 2    1     0  49 1  20 38 b8   1   0 0 0 0 0 0  0  0 0 0   1 8   6   2  2
D-24d  -      400  0      0 0   1    0
1a031000 Eth1/50     0 3    3     0  29 1  0  0  80   1   0 0 0 0 0 0  0  0 0 0   1 9   7   2  2
D-350  -      400  0      0 0   1    0
```

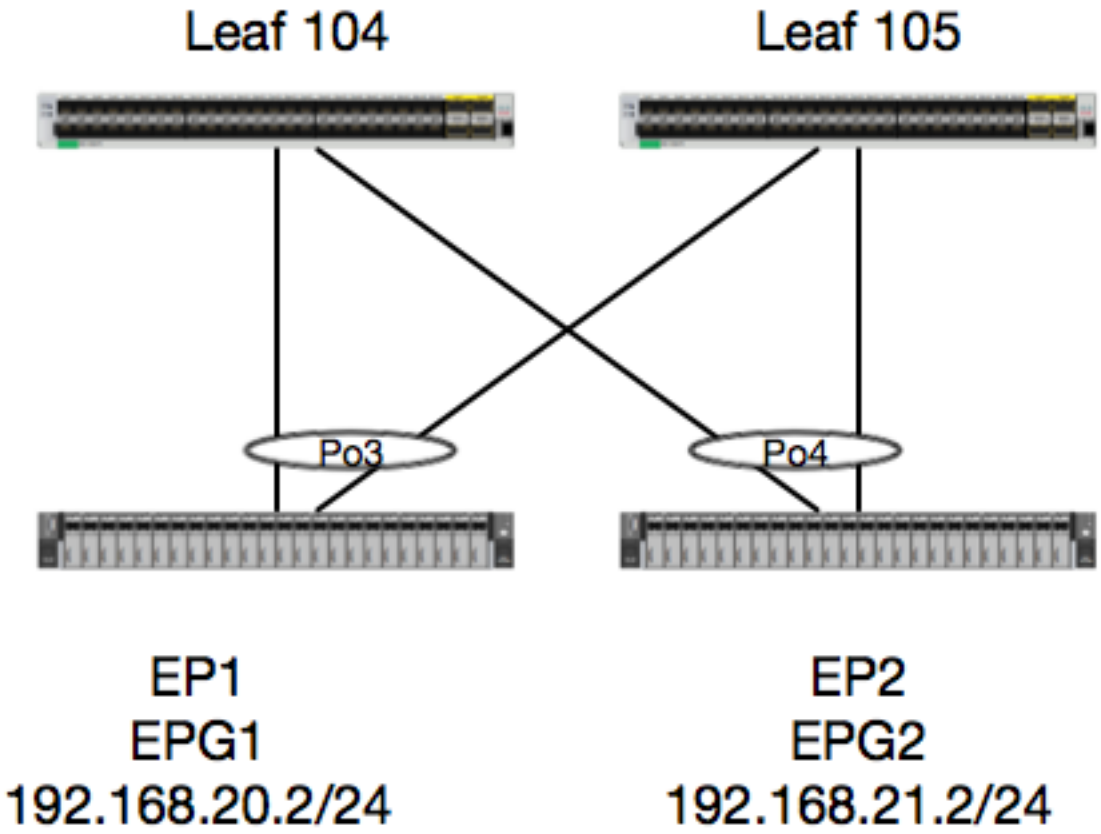
交换机认为应将数据包从接口Ethernet 1/32转发出去。我们是否在PO4上学习了该MAC地址？

```
leaf4# show port-channel summary
Flags:  D - Down        P - Up in port-channel (members)
        I - Individual  H - Hot-standby (LACP only)
        s - Suspended   r - Module-removed
        S - Switched    R - Routed
        U - Up (port-channel)
        M - Not in use. Min-links not met
        F - Configuration failed
------------------------------------------------------------------------------
Group Port-       Type     Protocol  Member Ports
      Channel
------------------------------------------------------------------------------
1     Po1(SU)     Eth      LACP      Eth1/5(P)
2     Po2(SU)     Eth      LACP      Eth1/6(P)
3     Po3(SU)     Eth      LACP      Eth1/31(P)
4     Po4(SU)     Eth      LACP      Eth1/32(P)
```

是，因此数据包将从接口1/32转发到目的主机。

不同EPG/相同枝叶中的2个EP — 路由数据包

拓扑



在本示例中，我们将跟踪数据包从EP1到EP2的数据包流，这些数据包存在于同一个vPC枝叶对中。 两个EP位于使用不同BD的不同EPG中。

首先要检查EP数据库，看看我们是否学过EP:

```
leaf4# show endpoint ip 192.168.20.2
Legend:
 O - peer-attached    H - vtep             a - locally-aged     S - static
 V - vpc-attached     p - peer-aged        L - local            M - span
 s - static-arp       B - bounce
+-----------------------------------+--------------+----------------+-------------+----------
---+
      VLAN/                          Encap          MAC Address       MAC Info/
Interface
      Domain                         VLAN           IP Address        IP Info
+-----------------------------------+--------------+----------------+-------------+----------
---+
30                                       vlan-2268    0050.56a5.fccc LV
po3
Joey-Tenant:Joey-Internal                vlan-2268      192.168.20.2 LV
po3


calo2-leaf4# show endpoint ip 192.168.21.2
Legend:
 O - peer-attached    H - vtep             a - locally-aged     S - static
 V - vpc-attached     p - peer-aged        L - local            M - span
 s - static-arp       B - bounce
+-----------------------------------+--------------+----------------+-------------+----------
---+
      VLAN/                          Encap          MAC Address       MAC Info/
Interface
```

```
      Domain                            VLAN            IP Address        IP Info
+-----------------------------------+--------------+-----------------+-------------+----------
---+
8                                       vlan-2200     0050.56a5.0c11 LV
po4
Joey-Tenant:Joey-Internal               vlan-2200       192.168.21.2 LV
po4
```

由于我们已经了解了EP并了解了IP信息，因此我们应该能够在硬件中查看EP学习信息：

```
leaf4# vsh_lc
module-1# show platform internal hal ep l3 all
LEGEND:
-------
VrfName:           Vrf Name                                                T:              Type
(Pl: Physical, Vl: Virtual, Xr: Remote)
EP IP:             Endpoint IP
S Class:           S Class                                                 Age Intvl:      Age
Interval
S T:               Static Ep                                               S E:
Secure EP
L D:               Learn Disable                                           B N D:          Bind
Notify Disable
E N D:             Epg Notify Disable                                      B E:
Bounce Enable
I D L:             IVxlan Dont Learn                                       SPI:
Source Policy Incomplete
DPI:               Dest Policy Incomplete                                  SPA:
Source Policy Applied
DPA:               Dest Policy Applied                                     DSS:            Dest
Shared Service
IL:                Is Local                                                VUB:            Vnid
Use Bd
SO:                SA Only                                                 EP NH L3IfName: EP
Next Hop L3 If Name
NHT:               Next Hop Type (L2: L2 Entry L3: L3 Next Hop)            BD Name:        L2 NH
BD Name
EP Mac:            EP Mac                                                  L3 IfName:      L3 NH
If Name
L2 IfName:         L2 If Name                                              FD Name:        L2
Entry FD Name
IP:                L3 NH IP

 L3 EP Count: 12

================================================================================================
================================================================================
                                                                  B E   I S D S D D   V   EP-NH
N  |
Vrf              EP                              S     Age   S S L N N B D P P P P S I U S L3
H  | BD        EP                L3           L2          FD
Name          T  IP                             Class Intvl T E D D D E L I I A A S L B O
IfName       T   | Name      Mac                IfName       Ifname      Name      IP
================================================================================================
================================================================================
common*rewall Pl 10.6.112.1                       1     0     1 0 0 0 0 0 1 1 0 0 0 0 1 0 0 -
L3   -         00:00:00:00:00:00 -           -           -         0.0.0.0
common*rewall Pl 10.6.114.1                       1     0     1 0 0 0 0 0 1 1 0 0 0 0 1 0 0 -
L3   -         00:00:00:00:00:00 -           -           -         0.0.0.0
common*rewall Pl 10.6.114.129                     1     0     1 0 0 0 0 0 1 1 0 0 0 0 1 0 0 -
L3   -         00:00:00:00:00:00 -           -           -         0.0.0.0
common*efault Pl 100.100.101.1                    1     0     1 0 0 0 0 0 1 1 0 0 0 0 1 0 0 -
```

```
L3   -         00:00:00:00:00:00 -           -           -         0.0.0.0
Joey-T*ternal Pl 192.168.1.1                   1     0     1 0 0 0 0 0 1 1 0 0 0 0 1 0 0 -
L3   -         00:00:00:00:00:00 -           -           -         0.0.0.0
Joey-T*ternal Xr 192.168.1.100                 8013  128   0 0 0 1 0 0 0 0 0 0 0 0 0 1 0 -
L3   -         00:0c:0c:0c:0c:0c Tunnel2     Tunnel2     -         0.0.0.0
Joey-T*ernal2 Pl 192.168.3.1                   1     0     1 0 0 0 0 0 1 1 0 0 0 0 1 0 0 -
L3   -         00:00:00:00:00:00 -           -           -         0.0.0.0
Joey-T*ternal Pl 192.168.20.1                  1     0     1 0 0 0 0 0 1 1 0 0 0 0 1 0 0 -
L3   -         00:00:00:00:00:00 -           -           -         0.0.0.0
Joey-T*ternal Pl 192.168.20.2                  800a  0     0 0 0 0 0 0 0 0 0 0 0 1 0 0 -
L2   BD-28     00:50:56:a5:fc:cc -           Po3         FD-30     -
Joey-T*ternal Pl 192.168.21.1                  1     0     1 0 0 0 0 0 1 1 0 0 0 0 1 0 0 -
L3   -         00:00:00:00:00:00 -           -           -         0.0.0.0
Joey-T*ternal Pl 192.168.21.2                  800c  0     0 0 0 0 0 0 0 0 0 0 0 1 0 0 -
L2   BD-7      00:50:56:a5:0c:11 -           Po4         FD-8      -
Joey-T*ternal Pl 2001:0:0:100::1               1     0     1 0 0 0 0 0 1 1 0 0 0 0 1 0 0 -
L3   -         00:00:00:00:00:00 -           -           -         0.0.0.0
```

HAL第3层（第3层）表非常有用，因为它为我们提供了第3层学习的EP的VLAN/端口信息。 我们知道Po4的目的地存在，因此应将数据包从Po4的任何端口转发出去。

让我们运行拉丁美洲医学院，看看我们得到什么！

ELAM

```
leaf4# vsh_lc
module-1# debug platform internal tah elam asic 0 module-1(DBG-TAH-elam)# trigger init in-select
6 out-select 0
module-1(DBG-TAH-elam-insel6)# set outer ipv4 src_ip 192.168.20.2 dst_ip 192.168.21.2
module-1(DBG-TAH-elam-insel6)# start
module-1(DBG-TAH-elam-insel6)# stat
ELAM STATUS
===========
Asic 0 Slice 0 Status Armed
Asic 0 Slice 1 Status Armed

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

module-1(DBG-TAH-elam-insel6)# report | grep ovec
  sug_elam_out_sidebnd_no_spare_vec.ovector_idx: 0x9E
```

很好，我们触发了数据包，发现“ovector_idx”为0x9E。ovector索引是应该将数据包转发出去的传出物理接口索引。 让我们看看哪个端口具有该索引：

```
module-1(DBG-TAH-elam-insel6)# show platform internal hal l2 port gpd
Legend:
-------
IfId:       Interface Id                                IfName:      Interface Name
I P:        Is PC Mbr                                   IfId:           Interface Id
Uc PC Cfg:      UcPcCfg Idx                             Uc PC MbrId:    Uc Pc Mbr Id
As:             Asic                                    AP:             Asic Port
Sl:             Slice                                   Sp:             Slice Port
Ss:             Slice SrcId                             Ovec:           Ovector (slice |
srcid)
L S:            Local Slot                              Reprogram:
```

```
L3:             Is L3
    P:    PifTable                                          Xla Idx:        Xlate Idx
    RP:   Rw PifTable                                       Ovx Idx:        OXlate Idx
    IP:   If Profile Table                                  N L3:           Num. of L3 Ifs
    RS:   Rw SrcId Table                                    NI L3:          Num. of Infra L3 Ifs
    DP:   DPort Table                                       Vif Tid:        Vif Tid
    SP:   SrcPortState Table                                RwV Tid:        RwVif Tid
    RSP:  RwSrcPortstate Table                              Ing Lbl:        Ingress Acl Label
    UC:   UCPcCfg                                           Egr Lbl:        Egress Acl Label
    UM:   UCPcMbr                                           Reprogram:
PROF ID:        Lport Profile Id
    VS:   VifStateTable                                     HI:             LportProfile Hw
Install
    RV:   Rw VifTable
Num. of Sandboxes: 1

Sandbox_ID: 0, BMP: 0x0
 Port Count: 8

================================================================================================
==========================================
                        Uc   Uc                          |       Reprogram       |
| Rep |
                     I PC   Pc                       L |   R I R D    R  U U X | L Xla Ovx N
NI Vif    RwV    Ing  Egr  | V R | PROF H
IfId     Ifname      P Cfg  MbrID As AP Sl Sp Ss Ovec S | P P P S P Sp Sp C M L | 3 Idx Idx L3
L3 Tid    Tid    Lbl  Lbl  | S V | ID   I
================================================================================================
==========================================
1a004000 Eth1/5      1 0    1d    0  d  0  c  18 18   1   0 0 0 0 0 0  0  0 0 0   0 0   0   0  0
-      -      800  0      0 1   0    0
1a005000 Eth1/6      1 0    b     0  e  0  d  1a 1a   1   0 0 0 0 0 0  0  0 0 0   0 0   0   0  0
-      -      800  0      0 1   0    0
1a006000 Eth1/7      0 26   5     0  f  0  e  1c 1c   1   0 0 0 0 0 0  0  0 0 0   0 0   0   0  0
D-256  -      800  0      0 1   c    0
1a007000 Eth1/8      0 2f   7     0  10 0  f  1e 1e   1   0 0 0 0 0 0  0  0 0 0   0 0   0   0  0
D-199  -      800  0      0 1   2e   0
1a01e000 Eth1/31     1 0    2d    0  37 1  e  1c 9c   1   0 0 0 0 0 0  0  0 0 0   0 0   0   0  0
-      -      0    0      0 1   0    0
1a01f000 Eth1/32     1 0    3d    0  38 1  f  1e 9e   1   0 0 0 0 0 0  0  0 0 0   0 0   0   0  0
-      -      0    0      0 1   0    0
1a030000 Eth1/49     0 2    1     0  49 1  20 38 b8   1   0 0 0 0 0 0  0  0 0 0   1 6   4   2  2
D-24d  -      400  0      0 0   1    0
1a031000 Eth1/50     0 3    3     0  29 1  0  0  80   1   0 0 0 0 0 0  0  0 0 0   1 5   3   2  2
D-350  -      400  0      0 0   1    0
```

看来我们应该把它从端口1/32发出去，对吗？

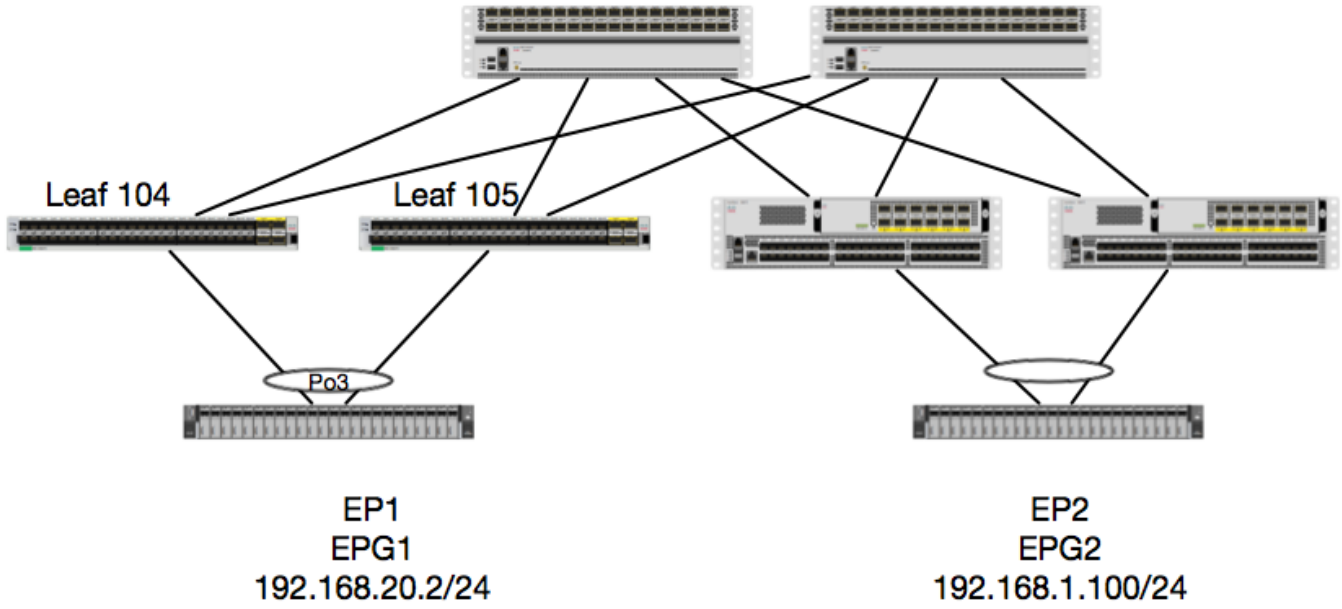
```
leaf4# show port-channel summary
Flags:  D - Down        P - Up in port-channel (members)
        I - Individual  H - Hot-standby (LACP only)
        s - Suspended   r - Module-removed
        S - Switched    R - Routed
        U - Up (port-channel)
        M - Not in use. Min-links not met
        F - Configuration failed
--------------------------------------------------------------------------------
Group Port-       Type     Protocol  Member Ports
      Channel
--------------------------------------------------------------------------------
1     Po1(SU)     Eth      LACP      Eth1/5(P)
2     Po2(SU)     Eth      LACP      Eth1/6(P)
3     Po3(SU)     Eth      LACP      Eth1/31(P)
```

```
4      Po4(SU)     Eth       LACP       Eth1/32(P)
```

是的，这是正确的。

不同EPG/不同枝叶中的2个EP — 路由数据包

拓扑



在本示例中，我们将跟踪从EP1到EP2的数据包流，其中EP1存在于EX vPC对中，而EP2存在于远程第1代vPC枝叶对中。 两个EP位于使用不同BD的不同EPG中。

我们再来看看EP从哪里学到：

```
leaf4# show endpoint ip 192.168.20.2
Legend:
 O - peer-attached    H - vtep             a - locally-aged     S - static
 V - vpc-attached     p - peer-aged        L - local            M - span
 s - static-arp       B - bounce
+-----------------------------------+--------------+----------------+-------------+----------
---+
      VLAN/                          Encap          MAC Address       MAC Info/
Interface
      Domain                         VLAN           IP Address        IP Info
+-----------------------------------+--------------+----------------+-------------+----------
---+
30                                      vlan-2268    0050.56a5.fccc LV
po3
Joey-Tenant:Joey-Internal               vlan-2268      192.168.20.2 LV
po3


calo2-leaf4# show endpoint ip 192.168.1.100
Legend:
 O - peer-attached    H - vtep             a - locally-aged     S - static
 V - vpc-attached     p - peer-aged        L - local            M - span
 s - static-arp       B - bounce
+-----------------------------------+--------------+----------------+-------------+----------
---+
```

```
      VLAN/                           Encap           MAC Address       MAC Info/
Interface
      Domain                          VLAN            IP Address        IP Info
+---------------------------------+---------------+-----------------+--------------+----------
---+
Joey-Tenant:Joey-Internal                              192.168.1.100
tunnel2
```

现在，让我们验证硬件已编程：

```
leaf4# vsh_lc
module-1# show platform internal hal ep l3 all
LEGEND:
-------
VrfName:           Vrf Name                                                T:              Type
(Pl: Physical, Vl: Virtual, Xr: Remote)
EP IP:             Endpoint IP
S Class:           S Class                                                 Age Intvl:      Age
Interval
S T:               Static Ep                                               S E:
Secure EP
L D:               Learn Disable                                           B N D:          Bind
Notify Disable
E N D:             Epg Notify Disable                                      B E:
Bounce Enable
I D L:             IVxlan Dont Learn                                       SPI:
Source Policy Incomplete
DPI:               Dest Policy Incomplete                                  SPA:
Source Policy Applied
DPA:               Dest Policy Applied                                     DSS:            Dest
Shared Service
IL:                Is Local                                                VUB:            Vnid
Use Bd
SO:                SA Only                                                 EP NH L3IfName: EP
Next Hop L3 If Name
NHT:               Next Hop Type (L2: L2 Entry L3: L3 Next Hop)            BD Name:        L2 NH
BD Name
EP Mac:            EP Mac                                                  L3 IfName:      L3 NH
If Name
L2 IfName:         L2 If Name                                              FD Name:        L2
Entry FD Name
IP:                L3 NH IP

 L3 EP Count: 12

================================================================================================
================================================================================
                                                                  B E   I S D S D D   V   EP-NH
N  |
Vrf              EP                                 S     Age   S S L N N B D P P P P S I U S L3
H  | BD        EP                L3           L2          FD
Name          T  IP                                 Class Intvl T E D D D E L I I A A S L B O
IfName       T  | Name      Mac                IfName       Ifname      Name      IP
================================================================================================
================================================================================
common*rewall Pl 10.6.112.1                        1     0     1 0 0 0 0 0 1 1 0 0 0 0 1 0 0 -
L3   -         00:00:00:00:00:00 -           -           -         0.0.0.0
common*rewall Pl 10.6.114.1                        1     0     1 0 0 0 0 0 1 1 0 0 0 0 1 0 0 -
L3   -         00:00:00:00:00:00 -           -           -         0.0.0.0
common*rewall Pl 10.6.114.129                      1     0     1 0 0 0 0 0 1 1 0 0 0 0 1 0 0 -
L3   -         00:00:00:00:00:00 -           -           -         0.0.0.0
common*efault Pl 100.100.101.1                     1     0     1 0 0 0 0 0 1 1 0 0 0 0 1 0 0 -
L3   -         00:00:00:00:00:00 -           -           -         0.0.0.0
```

```
Joey-T*ternal Pl 192.168.1.1                    1     0     1 0 0 0 0 0 1 1 0 0 0 0 1 0 0 -
L3   -         00:00:00:00:00:00 -            -           -         0.0.0.0
Joey-T*ternal Xr 192.168.1.100                  8013  128   0 0 0 1 0 0 0 0 0 0 0 0 0 1 0 -
L3   -         00:0c:0c:0c:0c:0c Tunnel2      Tunnel2     -         0.0.0.0
Joey-T*ernal2 Pl 192.168.3.1                    1     0     1 0 0 0 0 0 1 1 0 0 0 0 1 0 0 -
L3   -         00:00:00:00:00:00 -            -           -         0.0.0.0
Joey-T*ternal Pl 192.168.20.1                   1     0     1 0 0 0 0 0 1 1 0 0 0 0 1 0 0 -
L3   -         00:00:00:00:00:00 -            -           -         0.0.0.0
Joey-T*ternal Pl 192.168.20.2                   800a  0     0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 -
L2   BD-28     00:50:56:a5:fc:cc -            Po3         FD-30     -
Joey-T*ternal Pl 192.168.21.1                   1     0     1 0 0 0 0 0 1 1 0 0 0 0 1 0 0 -
L3   -         00:00:00:00:00:00 -            -           -         0.0.0.0
Joey-T*ternal Pl 192.168.21.2                   800c  0     0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 -
L2   BD-7      00:50:56:a5:0c:11 -            Po4         FD-8      -
Joey-T*ternal Pl 2001:0:0:100::1                1     0     1 0 0 0 0 0 1 1 0 0 0 0 1 0 0 -
L3   -         00:00:00:00:00:00 -            -           -         0.0.0.0
```

硬件认为隧道2上存在EP。隧道2的目的地是什么？

```
module-1# show system internal eltmc info interface tunnel2
           IfInfo:
          interface:        Tunnel2   :::         ifindex:      402718722
                iod:             66   :::           state:             up
                Mod:              0   :::            Port:              0
        Tunnel Index:             0   :::   Tunnel Dst ip:     0xc0a87843
        Tunnel Encap:         ivxlan   ::: Tunnel VPC Peer:              0
   Tunnel Dst ip str: 192.168.120.67   :::      Tunnel ept:            0x1

         [SDK Info]:
         tunnl_name:
             vrf_id:              2   :::        if_index:     0x18010002
         hwencapidx:              0   :::       encaptype:              1
          mac_proxy:              0   :::        v4_proxy:              0
           v6_proxy:              0   :::    ip_addr_type:              0
       ipv4_address:     0xc0a87843

[SDB INFO]:
                iod:             66
         pc_if_index:              0
        fab_if_index:              0
               sv_if:              0
            src_idx:              0
           int_vlan:              0
         encap_vlan:              0
     mod_port_status:     0x41620003
          v6_tbl_id:     0x80000002
          v4_tbl_id:            0x2
         router_mac:00.00.00.00.00.00
         unnumbered:              0
           trunk_id:              0
         tunnel_mod:              0
        tunnel_port:              0
            tep_ip:     0xc0a87843
         ip_if_mode:              0
         sdk_vrf_id:              2
                mtu:           9366   :::        ipmtu_id:              0
      is_fex_fabric:              0
```

由于vPC上存在目的地，因此该目的IP应为远程枝叶的vPC虚拟IP。 让我们检查一下远程枝叶，看：

```
leaf1# show system internal epm vpc
```

```
Local TEP IP                 : 192.168.160.95
Peer TEP IP                  : 192.168.160.93
vPC configured               : Yes
vPC VIP                      : 192.168.120.67
MCT link status              : Up
Local vPC version bitmap     : 0x7
Peer vPC version bitmap      : 0x7
Negotiated vPC version       : 3
Peer advertisement received  : Yes
Tunnel to vPC peer           : Up
```

非常完美，因此它从远程vPC对学习了目的EP。 让我们看看ELAM看到了什么，并检验我们是否正确转发数据包：

ELAM

```
module-1# debug platform internal tah elam asic 0
module-1(DBG-TAH-elam)# trigger init in-select 6 out-select 0
module-1(DBG-TAH-elam-insel6)# set outer ipv4 src_ip 192.168.20.2 dst_ip 192.168.1.100
module-1(DBG-TAH-elam-insel6)# start
module-1(DBG-TAH-elam-insel6)# stat
ELAM STATUS
===========
Asic 0 Slice 0 Status Armed
Asic 0 Slice 1 Status Triggered
```

现在，对于EX硬件上的远程目标，有2个ELAM值在排除数据包流故障时非常重要。 ovector_idx与之前类似，encap_idx:

```
module-1(DBG-TAH-elam-insel6)# report | grep ovec
  sug_elam_out_sidebnd_no_spare_vec.ovector_idx: 0xB8
module-1(DBG-TAH-elam-insel6)# report | grep encap
  sug_lurw_vec.encap_l2_idx: 0x0
  sug_lurw_vec.encap_pcid: 0x0
  sug_lurw_vec.encap_idx: 0x6
  sug_lurw_vec.encap_vld: 0x1
```

在EX硬件上，我们确实能够驱动数据包应转发出的目标端口。 以前，我们通常只检查encap idx，并验证目的idx是正确的隧道。 此处我们可以验证哪些端口映射到8B:

```
module-1(DBG-TAH-elam-insel6)# show platform internal hal l2 port gpd
Legend:
-------
IfId:       Interface Id                                IfName:      Interface Name
I P:        Is PC Mbr                                   IfId:           Interface Id
Uc PC Cfg:      UcPcCfg Idx                             Uc PC MbrId:    Uc Pc Mbr Id
As:             Asic                                    AP:             Asic Port
Sl:             Slice                                   Sp:             Slice Port
Ss:             Slice SrcId                             Ovec:           Ovector (slice |
srcid)
L S:            Local Slot                              Reprogram:
L3:             Is L3
    P:    PifTable                                      Xla Idx:        Xlate Idx
    RP:   Rw PifTable                                   Ovx Idx:        OXlate Idx
    IP:   If Profile Table                              N L3:           Num. of L3 Ifs
    RS:   Rw SrcId Table                                NI L3:          Num. of Infra L3 Ifs
    DP:   DPort Table                                   Vif Tid:        Vif Tid
```

```
    SP:   SrcPortState Table                                    RwV Tid:        RwVif Tid
    RSP:  RwSrcPortstate Table                                  Ing Lbl:        Ingress Acl Label
    UC:   UCPcCfg                                               Egr Lbl:        Egress Acl Label
    UM:   UCPcMbr                                               Reprogram:
PROF ID:        Lport Profile Id
    VS:   VifStateTable                                         HI:             LportProfile Hw
Install
    RV:   Rw VifTable
Num. of Sandboxes: 1

Sandbox_ID: 0, BMP: 0x0
 Port Count: 8

================================================================================================================
=========================================
                       Uc   Uc                          |       Reprogram       |
| Rep |
                     I PC   Pc                       L |   R I R D    R  U U X | L Xla Ovx N
NI Vif    RwV    Ing  Egr  | V R | PROF H
IfId     Ifname      P Cfg  MbrID As AP Sl Sp Ss Ovec S | P P P S P Sp Sp C M L | 3 Idx Idx L3
L3 Tid    Tid    Lbl  Lbl  | S V | ID   I
================================================================================================================
=========================================
1a004000 Eth1/5      1 0    1d    0  d  0  c  18 18   1   0 0 0 0 0 0  0  0 0 0   0 0   0   0  0
-      -      800  0      0 1   0    0
1a005000 Eth1/6      1 0    b     0  e  0  d  1a 1a   1   0 0 0 0 0 0  0  0 0 0   0 0   0   0  0
-      -      800  0      0 1   0    0
1a006000 Eth1/7      0 26   5     0  f  0  e  1c 1c   1   0 0 0 0 0 0  0  0 0 0   0 0   0   0  0
D-256  -      800  0      0 1   c    0
1a007000 Eth1/8      0 2f   7     0  10 0  f  1e 1e   1   0 0 0 0 0 0  0  0 0 0   0 0   0   0  0
D-199  -      800  0      0 1   2e   0
1a01e000 Eth1/31     1 0    2d    0  37 1  e  1c 9c   1   0 0 0 0 0 0  0  0 0 0   0 0   0   0  0
-      -      0    0      0 1   0    0
1a01f000 Eth1/32     1 0    3d    0  38 1  f  1e 9e   1   0 0 0 0 0 0  0  0 0 0   0 0   0   0  0
-      -      0    0      0 1   0    0
1a030000 Eth1/49     0 2    1     0  49 1  20 38 b8   1   0 0 0 0 0 0  0  0 0 0   1 6   4   2  2
D-24d  -      400  0      0 0   1    0
1a031000 Eth1/50     0 3    3     0  29 1  0  0  80   1   0 0 0 0 0 0  0  0 0 0   1 5   3   2  2
D-350  -      400  0      0 0   1    0
```

Switch认为应将其转发到接口Eth1/49上的主干。但如何验证封装是否正确？

我们首先需要查看有关隧道的硬件信息。 我们可以通过运行以下HAL命令来实现此目的：

```
module-1(DBG-TAH-elam-insel6)# show platform internal hal tunnel rtep pi
Non-Sandbox Mode
LEGEND:
-------
Tun Ifid:  Tunnel Ifid                                      IfName:     Tunnel If Name
Lid:       Logical Id                                       ET:        Encap Type V:
Vxlan I: IVxlan N: NVGRE
VrfId:     Vrf Id                                           Vrf Name: Vrf Name
IP:        Tunnel's IP
Hw Enc:    Hw Encap Idx                                     IVP:        Is VPC Peer
IL:        Is Local                                         P4:         Proxy for v4
P6:        Proxy for V6                                     PM:         Proxy for Mac
II:        Is Ingress Only                                  IC:         Is Copy Service
C OBd:     Copy Service Outer Bd                            U D:        Use DF
NBT:       Next Base Type E: ECMP N: Next-Hop               NB Id:      Next Base Id
NH cnt:    Next Hop Count                                   VrfId:      Vrf Id
Vrf Name:  Vrf Name                                         IP:         IP Address
Mac:       Mac                                              L3 IfId:    L3 IfId
L3IfName:  L3 If Name                                       L2 IfId:    L2 IfId
```

```
L2IfName:  L2 If Name

Num. of Sandboxes: 1

Sandbox_ID: 0, BMP: 0x0
 Remote Tep Count: 15

================================================================================================================
================================================================================================================
=======
                                                                    I                   N N
|

                      E             Vrf                         Hw   V I P P P I I C   U B B
NH  |        Vrf                                        L3       L3           L2       L2

IfId     Ifname       T Lid  VrfId  Name         IP            Enc  P L 4 6 M I C OBd D T Id
Cnt | VrfId  Name         IP             Mac               IfId     IfName       IfId     IfName

================================================================================================================
================================================================================================================
=======
18010002 Tunnel2      I 3005 2      overlay-1    192.168.120.670    0 0 0 0 0 0 0 1   0 E 2
2     2      overlay-1    0.0.0.0       0d:0d:0d:0d:0d:00 1a030001 Eth1/49.1    1a030000 Eth1/4
9

2      overlay-1    0.0.0.0       0d:0d:0d:0d:0d:00 1a031002 Eth1/50.2    1a031000 Eth1/5
0
```

此输出为我们提供了一些值，我们关心：

IfId — 分配给隧道的接口ID

IP — 目的地的IP。 这应该与ELTMC匹配。

L3 IfId — 交换机可用于转发到适当目的地的第3层接口。

一旦我们知道IfId，就可以验证我们在elam中获得的encap是否与隧道目标匹配：

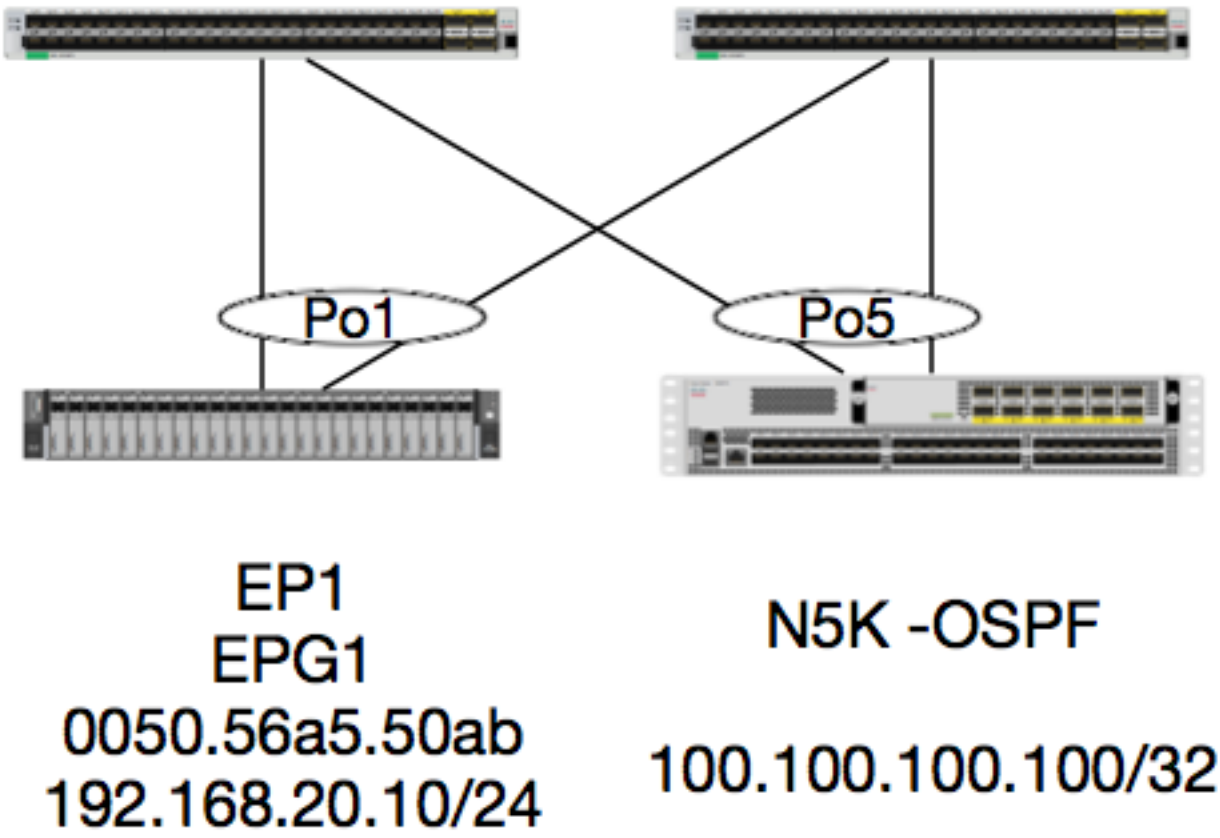
```
module-1(DBG-TAH-elam-insel9)# show platform internal hal tunnel rtep apd
Non-Sandbox Mode
LEGEND:
-------
ifId:        Interface Id                                IP:           IP address
HwVrfId:     Hardware Vrf Id                             SrcTepIdx:    Source Tep Index
BDXlate:     Egress BDXlate                              DstInfoIdx:   Destination info index
RwEncapIdx:  Rw Encap Index                              ECMPIdx:      ECMP Index
Num:         Number of hops                              ECMPMbrIdx:   ECMP member Index
L2 Index:    L2 Index                                    RwDmacIdx:    Rw Dmax Index

Num. of Sandboxes: 1

Sandbox_ID: 0, BMP: 0x0
 Remote Tep Count: 15

================================================================================================================
==============================
ifId     IP             HwVrfId BDXlate SrcTepIdx DstInfoIdx RwEncapIdx ECMPIdx  ECMPMbrIdx Num
L2Index  RwDmacIdx
================================================================================================================
==============================
```

```
18010002 192.168.120.67 2       1       3a9a      3005       6          0        0          2
1a030000 0         <---- RwEncapIdx is 6!  Same as the "encap_idx" in the ELAM Report.

1a031000 1
```

此隧道的RwEncapIdx（重写封顶索引）为6，即在elam中显示的。

1 EP —> L3输出 — 路由流

拓扑



在本例中，我们将跟踪数据包的数据包流，该数据包从EP1向运行OSPF的N5K上的环回发送ICMP。N5K通过L3Out连接在同一对EX交换机上。

由于我们在本文档开头对本地EP编程进行了验证，因此我们假设EP在硬件中已正确学习，并继续进行路由验证。

首先，我们检查OSPF状态和路由表：

```
leaf6# show ip ospf neighbors vrf jr:sb
 OSPF Process ID default VRF jr:sb
 Total number of neighbors: 2
 Neighbor ID     Pri State            Up Time  Address         Interface
 27.27.27.1        1 FULL/BDR         00:22:39 10.10.27.1      Vlan28   <---- Leaf5
 27.27.27.3        1 FULL/DROTHER     00:22:37 10.10.27.3      Vlan28   <----  N5K

leaf6# show ip route vrf jr:sb 100.100.100.100
IP Route Table for VRF "jr:sb"
'*' denotes best ucast next-hop
'**' denotes best mcast next-hop
```

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

100.100.100.100/32, ubest/mbest: 1/0
    *via 10.10.27.3, vlan28, [110/5], 00:16:58, ospf-default, intra
```

因此，我们知道路由表显示下一跳为10.10.27.3的5K。开始不错，但如何验证硬件是什么？

我们首先检查硬件中的邻接表，确保ARP解析为10.10.27.3，并且使用正确的接口对其进行编程：

```
leaf6# vsh_lc
module-1# show forwarding adjacency

IPv4 adjacency information, adjacency count 20

next-hop        rewrite info   interface       phy i/f
--------------  -------------- --------------- ---------------
10.10.27.1      0022.bdf8.19ff Vlan28          Tunnel3
10.10.27.3      8c60.4f02.88fc Vlan28          port-channel5
```

MAC地址与5K上的地址匹配：

```
ACI-5548-B# show interface vlan 3117
Vlan3117 is up, line protocol is up
  Hardware is EtherSVI, address is  8c60.4f02.88fc
  Internet Address is 10.10.27.3/29
  MTU 1500 bytes, BW 1000000 Kbit, DLY 10 usec
```

在EX平台上，有一个分配给VRF的“hw_vrf_idx”。当我们验证硬件编程时，将引用此索引。 让我们来查一下索引：

```
module-1# show system internal eltmc info vrf jr:sb
VRF-TABLE: jr:sb
            vrf_type:         tenant   :::      context_id:              6
       overlay_index:              0   :::            vnid:        2129921
               scope:              5   :::          sclass:          16386
         v4_table_id:            0x5   :::     v6_table_id:     0x80000005
          intf_count:              5   :::   intrn_vlan_id:              0
            VRF Intf:         Vlan11   ::: src_plcy_incomp:              0
            vnid_hex:       0x208001   :::  ingress_policy:            0x1
       vrf_intf_list: Vlan28,Vlan16,Vlan9,Vlan11,loopback2,
          hw_vrf_idx:           4612   ::: nb_egr_outer_bd:              0
     sb_egr_outer_bd:              0
        vrf_bd_list: 28,16,11,9,
     sb_egr_outer_bd:              0   :::      sdk_vrf_id:              5

          [SDK Info]:
            vrf_name:          jr:sb
              vrf_id:              5   :::      hw_vrf_idx:           4612
            vrf_vnid:        2129921   :::        is_infra:              0
      tornbinfrahwbd:              0   :::  torsbinfrahwbd:              0
   ingressBdAclLabel:              0   ::: ingBdAclLblMask:              0
    egressBdAclLabel:              0   ::: egrBdAclLblMask:              0
            sg_label:              5   :::          sclass:          16386
       sp_incomplete:              1   :::      sclassprio:              3

[SDB INFO]:
v4 table
            vrf type:              1
              vrf id:              5
                vnid:        2129921
```

```
 internal infra vlan:              0
 external router mac:00:22:bd:f8:19:ff
v6 table
            vrf type:              1
              vrf id:              5
                vnid:        2129921
 internal infra vlan:              0
 external router mac:00:22:bd:f8:19:ff
::::
```

在检测到邻接关系后，HAL应编程路由。 我们可以使用以下命令检查此项：

```
module-1# show platform internal hal l3 routes | head

-------------------------------------------------------------------------------------------------
-------------------------------------------------------------------------------------------------
LEGEND:
|
-------------------------------------------------------------------------------------------------
-------------------------------------------------------------------------------------------------
LID: Logical ID           RID: Route ID               PID: Physical ID       NB-ID:Next-Base ID
HIT IDX: Next-Hop HitIndex   CLP : Class Priority       TBI: Trie Base Index             |
SC : Sup-Copy             SSR: Src Sup-Redirect       DSR: Dst Sup-Redirect  TDD  :TTL Disable
NB: NextBaseType             SDC : Src Direct Connect   TRO: Trie Offset                 |
SPI: Src Policy Inc       DPI: Dst Policy Inc         DR : Default Route     LE   :Learn Enable
[E:Ecmp/A:Adj]              ILL : Is Link Local        ISS: Is Shared Services          |
RT : Route Type           FWD: Forwarding             HR : Host Routes       EP   :Ext Prefixes
DLR: Default Lpm Route       CLSS: Class Id             RDEL: Route in Deletion          |
BNE: Bind Notify Enable   SNE: Sclass Notify Enable   BE : Bounce Enable     IDL  :Ivxlan
DoNotLearn  DL : Dest Local              SA  : Src Only             AI : Age Interval
|
SF : Static Flag          SH : Src Hit                DH: Dest Hit
|
module-1# show platform internal hal l3 routes

-------------------------------------------------------------------------------------------------
-------------------------------------------------------------------------------------------------
LEGEND:
|
-------------------------------------------------------------------------------------------------
-------------------------------------------------------------------------------------------------
LID: Logical ID           RID: Route ID               PID: Physical ID       NB-ID:Next-Base ID
HIT IDX: Next-Hop HitIndex   CLP : Class Priority       TBI: Trie Base Index             |
SC : Sup-Copy             SSR: Src Sup-Redirect       DSR: Dst Sup-Redirect  TDD  :TTL Disable
NB: NextBaseType             SDC : Src Direct Connect   TRO: Trie Offset                 |
SPI: Src Policy Inc       DPI: Dst Policy Inc         DR : Default Route     LE   :Learn Enable
[E:Ecmp/A:Adj]              ILL : Is Link Local        ISS: Is Shared Services          |
RT : Route Type           FWD: Forwarding             HR : Host Routes       EP   :Ext Prefixes
DLR: Default Lpm Route       CLSS: Class Id             RDEL: Route in Deletion          |
BNE: Bind Notify Enable   SNE: Sclass Notify Enable   BE : Bounce Enable     IDL  :Ivxlan
DoNotLearn  DL : Dest Local              SA  : Src Only             AI : Age Interval
|
SF : Static Flag          SH : Src Hit                DH: Dest Hit
|
-------------------------------------------------------------------------------------------------
-------------------------------------------------------------------------------------------------
|     |                                          |   |     |          | LID |<-----------------
- Trie ------------>|<Dleft Trie>|                                                              |
| VRF |                Prefix/Len                | RT| RID |    LID   | Type| PID | FPID/| HIT
|N|  NB-ID  |  NB Hw | PID | FPID/|   TBI   |TRO|Ifindex|CLSS|CLP| AI |SH|DH| Flags            |
|-----|-----------------------------------------|---|-----|----------|-----|     | TID  | IDX
|B|         |  Idx   |     |  TID |---------|---|-------|----|---|----|--|--|------------------|
|-----|-----------------------------------------|---|-----|----------|-----|<-----------------
```

```
- DLEFT ------------>|-----|------|---------|---|-------|----|---|----|--|--|------------------|
|     |                                          |   |     |          |     | PID | FPID/| HIT
|N|  NB-ID  |  NB Hw |     |      |         |   |       |    |   |    |  |  |                  |
|     |                                          |   |     |          |     |     |  TID | IDX
|B|         |  Idx   |     |      |         |   |       |    |   |    |  |  |                  |
|     |                                          |   |     |          |     |<-----------------
- TCAM ------------->|     |      |         |   |       |    |   |    |  |  |                  |
|     |                                          |   |     |          |     | PID | TCAM | HIT
|N|  NB-ID  |  NB Hw |     |      |         |   |       |    |   |    |  |  |                  |
|     |                                          |   |     |          |     |     |  ID  | IDX
|B|         |  Idx   |     |      |         |   |       |    |   |    |  |  |                  |
-------------------------------------------------------------------------------------------------
-------------------------------------------------------------------------------------------------
|Sandbox_ID: 0  Asic Bitmap: 0x0
-------------------------------------------------------------------------------------------------
-------------------------------------------------------------------------------------------------
```

```
module-1# show platform internal hal l3 routes | egrep 100.100.100.100
| 4612|                        100.100.100.100/ 32| UC|   e4|      4a04| TRIE|   10| 5/  0|
6010|A|     7567|    802e| 186a| 1/  2|       10|  0|      0|   f|  3|   0| 0| 0|spi,dpi
```

此输出提供了有关下一跳路由的信息。 4612是jr:sb VRF的hw_vrf_idx。为了验证下一跳，TCAM中的“NB硬件IDX”将用于下一表：

```
module-1# show platform internal hal l3 nexthops
Non-Sandbox Mode
LEGEND:
-------
NHOP ID       : Nhop Identifier (Hex)            CONS          : H/W S/W info Consistency
TYPE          : Nexthop Type                     ACTN          : Nexthop Action
Vrf           : L3 Vrf of the Nhop               L3 INTF       : L3 interface index (Hex)
L2 INTF       : L2 interface index (Hex)         BDID Or RwVRF : Bridge Domain Id Or Rewrite
Vrfid (Hex)
INFR          : ACI Infra valid                  PVRF          : Preserve VRF
LRN           : Learn Enabled                    VRFR          : VRF Rewrite
PID           : Physical ID                      FPID          : FP of this nexthop
TLID          : Tile Id within FP                HIT IDX       : Location of this Nhop (Hex)

Mac Entry:
TYP         : Type                      INTF        : Interface related Info (Hex)
LRN         : Learn Info                DL          : Destination Local
MLD         : Unused                    VNB         : Vnid use BD
DFL         : Default Entry             VLD         : MacKey Valid
FT          : FID Type                  FV          : FID Valid
FID         : FID value (Hex)           Mac         : L2 MAC Address
L2 Ifabric Info:
CLSS        : Source Class                          CLP         : Source Class Priority
EPG         : EndPoint Group                        BNE         : Bind Notification Enabled
SNE         : Source Address Notification Enabled CNE         : Source class Notification
Enabled
DL          : iVxlan DL                             SPI         : Source Policy Incomplete
DPI         : Dest Policy Incomplete
IP Address  : IP address

Sandbox_ID: 0  Asic Bitmap: 0x0
Summary info for 31 L3 Nexthop objects
         C T A                      BDID  I P   V         T    |--------------------Mac Entry-
-----------------|----L2 Ifabric Info----|
 NHOP    O Y C         L3       L2    Or   N V L R         L HIT|T        L   M V D V|----------
----Mac Key-------|     C      B S C   S D|
  ID     N P T        INTF     INTF  RwVRF F R R F      FP I IDX|Y  INTF  R D L N F L|F F  FID
|     L      N N N D P P|
 (Hex)   S E N Vrf    (H)       (H)   (H)  R F N R PID  ID D (H)|P  (H)   N L D B L D|T V  (H)
```

```
Mac          |CLSS P EPG  E E E L I I| IP Address
--------+-+-+-+----+--------+--------+----+-+-+-+-+----+--+-+----+-+------+-+-+-+-+-+-+-+-+-----
-+-----------------+----+-+----+-+-+-+-+-+-+---------------------------

module-1# show platform internal hal l3 nexthops | grep 802e
    7567 N I F    5  901001c 16000004   1c 0 0 0 0   2e  9 0 802e 0     22 0 0 0 0 0 1 1 1
1214 8c:60:4f:02:88:fc    0 0 2c0d 0 0 0 0 0 0 10.10.27.3
```

此处，我们将“NB硬件IDX”映射到“HIT IDX”。这将显示与下一跳MAC/IP对应的条目。这相当于在第1代ACI枝叶交换机上查看Broadcom中的“l3 defip show”和“l3 egress show”。

如我们所见，该表具有正确的信息：

L2 INTF:0x16000004 —>端口通道5的ifIndex

点击IDX: 从Nb Hw Idx在Hall3路由中驱动的索引

MAC:8c:60:4f:02:88:fc —> 5000上下一跳SVI的MAC

EPG: L3 EPG的类

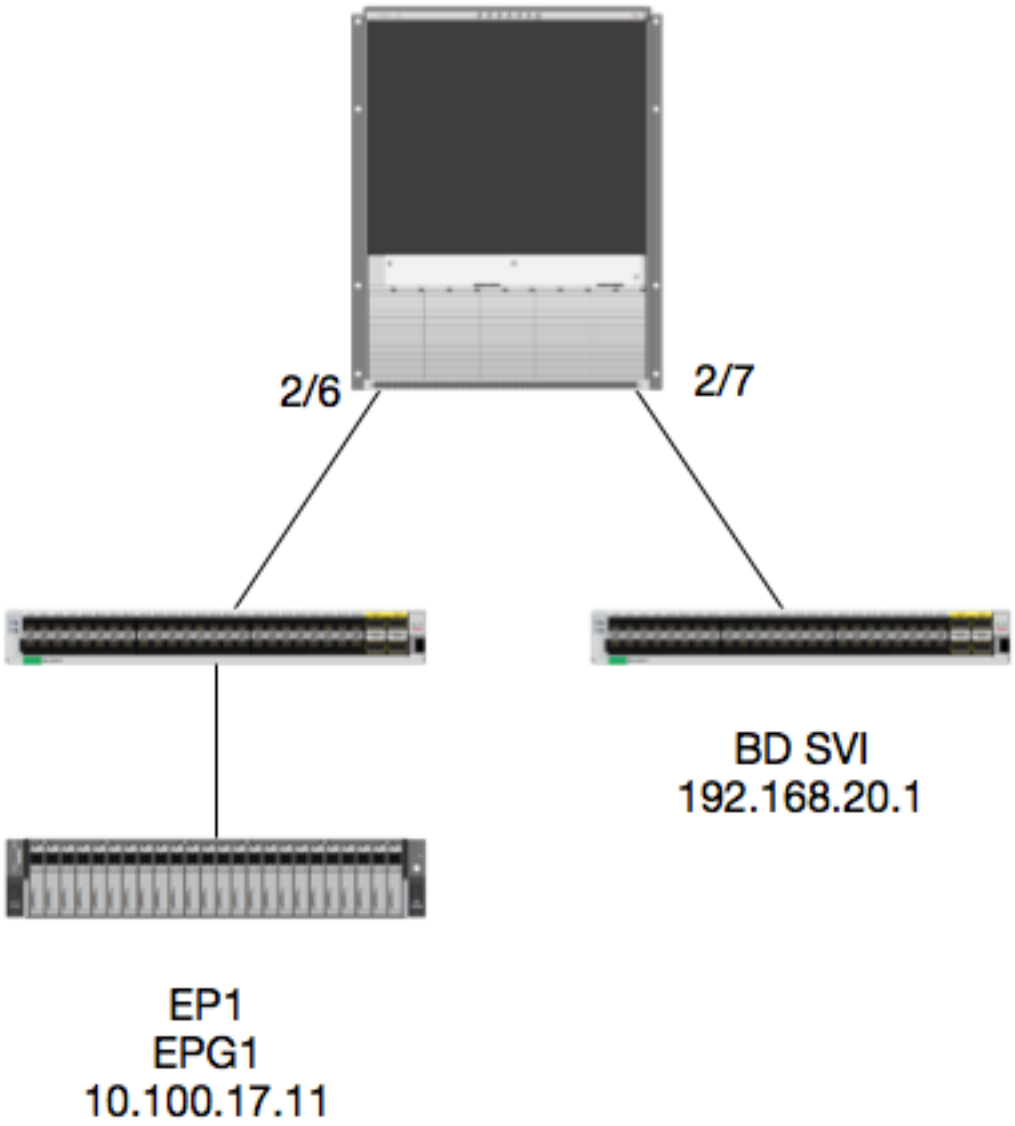
IP Address:10.10.27.3 —> 5000上SVI的下一跳IP

ELAM

```
leaf6# pwd
/var/sysmgr/tmp_logs

leaf6# cat elam_report.txt | grep ip.da
    sug_pr_lu_vec_l3v.ip.da: 0x00000000000000000064646464
leaf6# cat elam_report.txt | grep ip.sa
    sug_pr_lu_vec_l3v.ip.sa: 0x0000000000000000C0A8140A

leaf6# cat elam_report.txt | grep adj
    sug_lurw_vec.dst_addr.adj: 0x8C604F0288FC
      sug_lurw_vec.dst_addr.adj.padfield: 0x04F0288FC
      sug_lurw_vec.dst_addr.adj.idx: 0x2318
  sug_lurw_vec.adj_vld: 0x0

leaf6# cat elam_report.txt | grep macdarslt.hit_idx
        sug_fpc_lookup_vec.fplu_vec.rslt.macdarslt.hit_idx: 0x802E
```

1 EP —>远程EP或SVI — 主干验证

拓扑



逻辑

在本示例中，我们将跟踪从EP1发往远程BD交换虚拟接口(SVI)的数据包的数据包流。本示例的目的是验证主干转发以确保数据包发送到正确的枝叶。 假设数据包已发送到入口枝叶上的主干代理。

在主干上，我们首先验证目的IP的Council of Oracles Protocol(COOP)，因为数据包被发送到主干代理以进行查找：

```
calo1-spine1# show coop internal info ip-db | grep -A 10 192.168.20.1
IP address : 192.168.20.1
Vrf : 2129921
Flags : 0
EP vrf vnid : 2129921
EP IP :  192.168.20.1
Publisher Id : 10.0.224.88
Record timestamp : 11 04 2016 16:41:16 422062712
Publish timestamp : 11 04 2016 16:41:16 424633605
Seq No: 0
Remote publish timestamp: 01 01 1970 00:00:00 0
URIB Tunnel Info
Num tunnels : 1
Tunnel address : 10.0.224.88    <---- REMOTE LEAF
```

```
Tunnel ref count : 1
```

让我们检验TEP地址为哪个枝叶：

```
spine1# acidiag fnvread | grep 10.0.224.88
     105        1          calo1-leaf5      FDO20160TPS     10.0.224.88/32    leaf
active   0
```

由于我们知道数据包进入模块2的主干端口6，因此我们可以连接到模块2并查看端口布局。

```
spine1# vsh
Cisco iNX-OS Debug Shell
This shell should only be used for internal commands and exists
for legacy reasons. User should use ibash infrastructure as this
will be deprecated.
calo1-spine1# attach module 2
Attaching to module 2 ...
To exit type 'exit', to abort type '$.'
No directory, logging in with HOME=/
Bad terminal type: "xterm-256color". Will assume vt100.
Cisco iNX-OS Debug Shell
This shell should only be used for internal commands and exists
for legacy reasons. User should use ibash infrastructure as this
will be deprecated.
Loading parse tree (LC). Please be patient...
module-2#

module-2# show platform internal hal l2 port gpd
Legend:
-------
IfId:       Interface Id                                    IfName:      Interface Name
I P:        Is PC Mbr                                       IfId:           Interface Id
Uc PC Cfg:      UcPcCfg Idx                                 Uc PC MbrId:    Uc Pc Mbr Id
As:             Asic                                        AP:             Asic Port
Sl:             Slice                                       Sp:             Slice Port
Ss:             Slice SrcId                                 Ovec:           Ovector (slice |
srcid)
L S:            Local Slot                                  Reprogram:
L3:             Is L3
    P:    PifTable                                          Xla Idx:        Xlate Idx
    RP:   Rw PifTable                                       Ovx Idx:        OXlate Idx
    IP:   If Profile Table                                  N L3:           Num. of L3 Ifs
    RS:   Rw SrcId Table                                    NI L3:          Num. of Infra L3 Ifs
    DP:   DPort Table                                       Vif Tid:        Vif Tid
    SP:   SrcPortState Table                                RwV Tid:        RwVif Tid
    RSP:  RwSrcPortstate Table                              Ing Lbl:        Ingress Acl Label
    UC:   UCPcCfg                                           Egr Lbl:        Egress Acl Label
    UM:   UCPcMbr                                           Reprogram:
PROF ID:        Lport Profile Id
    VS:   VifStateTable                                     HI:             LportProfile Hw
Install
    RV:   Rw VifTable
Num. of Sandboxes: 1

Sandbox_ID: 0, BMP: 0x0
 Port Count: 7

================================================================================================
==========================================
                        Uc   Uc                          |       Reprogram       |
| Rep |
                     I PC   Pc                       L |   R I R D    R  U U X | L Xla Ovx N
```

```
NI Vif    RwV    Ing  Egr  | V R | PROF H
IfId     Ifname      P Cfg  MbrID As AP Sl Sp Ss Ovec S | P P P S P Sp Sp C M L | 3 Idx Idx L3
L3 Tid    Tid    Lbl  Lbl  | S V | ID   I
=============================================================================================
==========================================
1f5      SpInBndMgmt 0 9de  1a    0  0  0  0  0  0    0   0 0 0 0 0 0  0  0 0 0   0 0   0   0  0
D-2d4  D-3e1  0    0      0 0   1    0
1a080000 Eth2/1      0 9a   1c    0  11 0  10 20 20   1   0 0 0 0 0 0  0  0 0 0   1 b   b   1  1
D-f3   D-61   100  0      0 0   1    0
1a081000 Eth2/2      0 9b   22    0  d  0  c  18 18   1   0 0 0 0 0 0  0  0 0 0   1 c   c   1  1
D-1ee  D-30b  100  0      0 0   1    0
1a084000 Eth2/5      0 9e   1e    0  3d 1  14 28 a8   1   0 0 0 0 0 0  0  0 0 0   1 1   1   1  1
D-19a  D-2ee  100  0      0 0   1    0
1a085000 Eth2/6      0 9f   24    0  39 1  10 20 a0   1   0 0 0 0 0 0  0  0 0 0   1 e   e   1  1
D-87   D-184  100  0      0 0   1    0
1a086000 Eth2/7      0 a0   26    0  35 1  c  18 98   1   0 0 0 0 0 0  0  0 0 0   1 d   d   1  1
D-1d0  D-357  100  0      0 0   1    0
1a088000 Eth2/9      0 a2   20    1  d  0  c  18 18   1   0 0 0 0 0 0  0  0 0 0   0 0   0   0  0
D-3ea  D-1a9  100  0      0 0   1    0
```

Ethernet 2/6是连接到枝叶6的接口，位于ASIC 0 SLICE 1上

现在，我们知道在哪个ASIC上运行ELAM。 ASIC 0.

```
module-2# debug platform internal tah elam asic 0
module-2(DBG-TAH-elam)# trigger reset
module-2(DBG-TAH-elam)# trigger init in-select 13 out-select 0
module-2(DBG-TAH-elam-insel13)# set inner ipv4 src_ip 10.100.17.11 dst_ip 192.168.20.1
module-2(DBG-TAH-elam-insel13)# start
stat
module-2(DBG-TAH-elam-insel13)# stat
ELAM STATUS
===========
Asic 0 Slice 0 Status Armed
Asic 0 Slice 1 Status Armed

module-2(DBG-TAH-elam-insel13)# stat
ELAM STATUS
===========
Asic 0 Slice 0 Status Triggered   <---- Packet triggered from FM
Asic 0 Slice 1 Status Triggered   <---- Packet triggered from Front Panel
```

从ELAM看，我们可以找到矢量指数：

```
Front Panel ELAM drove sug_elam_out_sidebnd_no_spare_vec.ovector_idx: 0xB8
```

现在，如何将0xb8映射到端口？ 由于我们知道应将数据包发送到交换矩阵模块(FM)进行查找，因此我们可以查看内部端口映射以查找目的FM:

```
module-2# show platform internal hal l2 internal-port pi
Num. of Sandboxes: 1
Legend:
-------
IfId:      Interface Id               IfName:     Interface Name
As:        Asic                       AP:         Asic Port
Sl:        Slice                      SP:         Slice Port
Ss:        Slice SrcId                Ovec:       Ovector
UcPcCfgId: Uc Pc CfgId                Lb Mbrid:   LB MbrId

Sandbox_ID: 0, BMP: 0x0
 Internal Port Count: 32
```

```
=====================================================================
                                                        UcPc  Lb
IfId     IfName                      As AP Sl SP Ss Ovec CfgId MbrId
=====================================================================
7d       -                           0  21 0  20 38 38   0     4
7e       -                           0  29 1  0  0  80   0     8
7f       -                           1  21 0  20 38 38   0     c
80       -                           1  29 1  0  0  80   0     10
81       -                           2  21 0  20 38 38   0     14
82       -                           2  29 1  0  0  80   0     18
83       -                           3  21 0  20 38 38   0     1c
84       -                           3  29 1  0  0  80   0     20
95       -                           0  19 0  18 30 30   0     3
96       -                           0  49 1  20 38 b8   0     7
97       -                           1  19 0  18 30 30   0     b
98       -                           1  49 1  20 38 b8   0     f
99       -                           2  19 0  18 30 30   0     13
9a       -                           2  49 1  20 38 b8   0     17
9b       -                           3  19 0  18 30 30   0     1b
9c       -                           3  49 1  20 38 b8   0     1f
ad       -                           0  25 0  24 40 40   0     1
ae       -                           0  41 1  18 30 b0   0     6
af       -                           1  25 0  24 40 40   0     9
b0       -                           1  41 1  18 30 b0   0     e
b1       -                           2  25 0  24 40 40   0     11
b2       -                           2  41 1  18 30 b0   0     16
b3       -                           3  25 0  24 40 40   0     19
b4       -                           3  41 1  18 30 b0   0     1e
dd       -                           0  15 0  14 28 28   0     2
de       -                           0  4d 1  24 40 c0   0     5
df       -                           1  15 0  14 28 28   0     a
e0       -                           1  4d 1  24 40 c0   0     d
e1       -                           2  15 0  14 28 28   0     12
e2       -                           2  4d 1  24 40 c0   0     15
e3       -                           3  15 0  14 28 28   0     1a
e4       -                           3  4d 1  24 40 c0   0     1d
```

使用ASIC0/Ovec B8，我们可以获得MbrId 0x7，切片无关紧要。

此MbrId是USD上的接口，映射到FM上的接口。 请记住，此MbrId为十六进制，必须转换为十进制。

通过查看USD接口并检查端口7，我们可以找出哪个FM:

```
module-2# show platform internal usd port info | grep -A 3 "Int 7"(if the interface has multiple
digits, will be "Int##" with no space)

Port  73.0 (Int 7) : Admin UP Link UP Remote slot22.asic0
                slice:1 slice port:32 lcl srcid:56 gbl srcid:184
                asic mrl:0xd07c010, mac mrl:0x12c84010, mac:16, chan:0
                speed 106G serdes: 0x328 0x329 0x32a 0x32b
```

“slot”基于0,FM编号基于1，因此我们需要在此处列出的编号中添加1。 这意味着应将数据包发送到FM 23。

综合IP

就像在Alpine一样，有一个合成IP用作外部IP地址，用于确定COOP查找的哈希值。 为了找到这一点，您需要运行以下命令并为内部DST IP运行grep:

```
module-2(DBG-TAH-elam-insel7)# show forwarding route synthetic vrf all | grep 192.168.20.1
SYNTH-88        1.203.211.185/32    0x208001          192.168.20.1
```

这显示1.203.211.185是我们的合成IP。基于此，我们还可以将FM电缆上的“外部DST IP”设置为此。 我们应该在FM上触发：

交换矩阵模块ELAM

```
module-23(DBG-TAH-elam-insel7)# trigger reset
module-23(DBG-TAH-elam)# trigger init in-select 13 out-select 0
module-23(DBG-TAH-elam-insel13)# set outer ipv4 dst_ip 1.203.211.185   <----- DST IP IS THE
SYNTHETIC IP
module-23(DBG-TAH-elam-insel13)# set inner ipv4 src_ip 10.100.17.11 dst_ip 192.168.20.1
module-23(DBG-TAH-elam-insel13)# start
stat
module-23(DBG-TAH-elam-insel13)# stat
ELAM STATUS
===========
Asic 0 Slice 0 Status Armed
Asic 0 Slice 1 Status Armed
Asic 0 Slice 2 Status Armed
Asic 0 Slice 3 Status Armed
Asic 0 Slice 4 Status Armed
Asic 0 Slice 5 Status Armed

module-23(DBG-TAH-elam-insel13)# stat
ELAM STATUS
===========
Asic 0 Slice 0 Status Armed
Asic 0 Slice 1 Status Armed
Asic 0 Slice 2 Status Triggered    <---- Triggered on SLICE 2
Asic 0 Slice 3 Status Armed
Asic 0 Slice 4 Status Armed
Asic 0 Slice 5 Status Armed
```

显然，请转储完整报告，但是，让我们看看我们触发的此数据包的ovector_idx:

lac_elam_out sidend_no_spare_vec.ovector_idx:**0x20** < — 在以下命令中使用的矢量索引

如何确定哪个接口具有该矢量？ 在FM上，运行以下命令：

****由于Bug CSCvf42796 ，将所有FM命令附加为“| no-more”。 否则，某些表条目可能不会显示在最终输出中。**

```
module-23(DBG-TAH-elam-insel13)# show platform internal hal l2 port gpd | no-more
Legend:
-------
IfId:       Interface Id                                IfName:      Interface Name
I P:        Is PC Mbr                                   IfId:           Interface Id
Uc PC Cfg:      UcPcCfg Idx                             Uc PC MbrId:    Uc Pc Mbr Id
As:             Asic                                    AP:             Asic Port
Sl:             Slice                                   Sp:             Slice Port
Ss:             Slice SrcId                             Ovec:           Ovector (slice |
srcid)
L S:            Local Slot                              Reprogram:
L3:             Is L3
    P:    PifTable                                      Xla Idx:        Xlate Idx
    RP:   Rw PifTable                                   Ovx Idx:        OXlate Idx
```

```
    IP:   If Profile Table                                  N L3:           Num. of L3 Ifs
    RS:   Rw SrcId Table                                    NI L3:          Num. of Infra L3 Ifs
    DP:   DPort Table                                       Vif Tid:        Vif Tid
    SP:   SrcPortState Table                                RwV Tid:        RwVif Tid
    RSP:  RwSrcPortstate Table                              Ing Lbl:        Ingress Acl Label
    UC:   UCPcCfg                                           Egr Lbl:        Egress Acl Label
    UM:   UCPcMbr                                           Reprogram:
PROF ID:        Lport Profile Id
    VS:   VifStateTable                                     HI:             LportProfile Hw
Install
    RV:   Rw VifTable
Num. of Sandboxes: 1

Sandbox_ID: 1, BMP: 0x1
 Port Count: 8

================================================================================================================
=========================================
                       Uc   Uc                          |       Reprogram       |
| Rep |
                     I PC   Pc                        L |   R I R D    R  U U X | L Xla Ovx N
NI Vif    RwV    Ing  Egr  | V R | PROF H
IfId     Ifname      P Cfg  MbrID As AP Sl Sp Ss Ovec S | P P P S P Sp Sp C M L | 3 Idx Idx L3
L3 Tid    Tid    Lbl  Lbl  | S V | ID   I
================================================================================================================
=========================================
ae       fc0-lc1:0-0 1 0    3     0 11 0 10 20 20 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 - - 0 0 0 0 0
0
af       fc0-lc1:0-1 1 0    4     0  3d 2  c  18 98   1   0 0 0 0 0 0  0  0 0 0   0 0   0   0  0
-      -      0    0      0 0   0    0
b0       fc0-lc1:1-0 1 0    13    0  d  0  c  18 18   1   0 0 0 0 0 0  0  0 0 0   0 0   0   0  0
-      -      0    0      0 0   0    0
b1       fc0-lc1:1-1 1 0    14    0  39 2  8  10 90   1   0 0 0 0 0 0  0  0 0 0   0 0   0   0  0
-      -      0    0      0 0   0    0
b2       fc0-lc1:2-0 1 0    23    0  5d 3  14 28 e8   1   0 0 0 0 0 0  0  0 0 0   0 0   0   0  0
-      -      0    0      0 0   0    0
b3       fc0-lc1:2-1 1 0    24    0  21 1  8  10 50   1   0 0 0 0 0 0  0  0 0 0   0 0   0   0  0
-      -      0    0      0 0   0    0
b4       fc0-lc1:3-0 1 0    33    0  51 3  8  10 d0   1   0 0 0 0 0 0  0  0 0 0   0 0   0   0  0
-      -      0    0      0 0   0    0
```

该矢量映射到ASIC 0/SLICE 0上的LC1（插槽2中的线卡，因为它基于0）。从最初在LC上运行的ELAM中，我们触发到此片：

```
module-2# debug platform internal tah elam asic 0
module-2(DBG-TAH-elam)# trigger reset
module-2(DBG-TAH-elam)# trigger init in-select 13 out-select 0
module-2(DBG-TAH-elam-insel13)# set inner ipv4 src_ip 10.100.17.11 dst_ip 192.168.20.1
module-2(DBG-TAH-elam-insel13)# start
stat
module-2(DBG-TAH-elam-insel13)# stat
ELAM STATUS
===========
Asic 0 Slice 0 Status Armed
Asic 0 Slice 1 Status Armed

module-2(DBG-TAH-elam-insel13)# stat
ELAM STATUS
===========
Asic 0 Slice 0 Status Triggered   <---- Packet triggered from FM
Asic 0 Slice 1 Status Triggered   <---- Packet triggered from Front Panel
```

此ELAM上的矢量是sug_elam_out_sidend_no_spare_vec.ovector_idx:0x98(从“hal l2 port gpd”中可

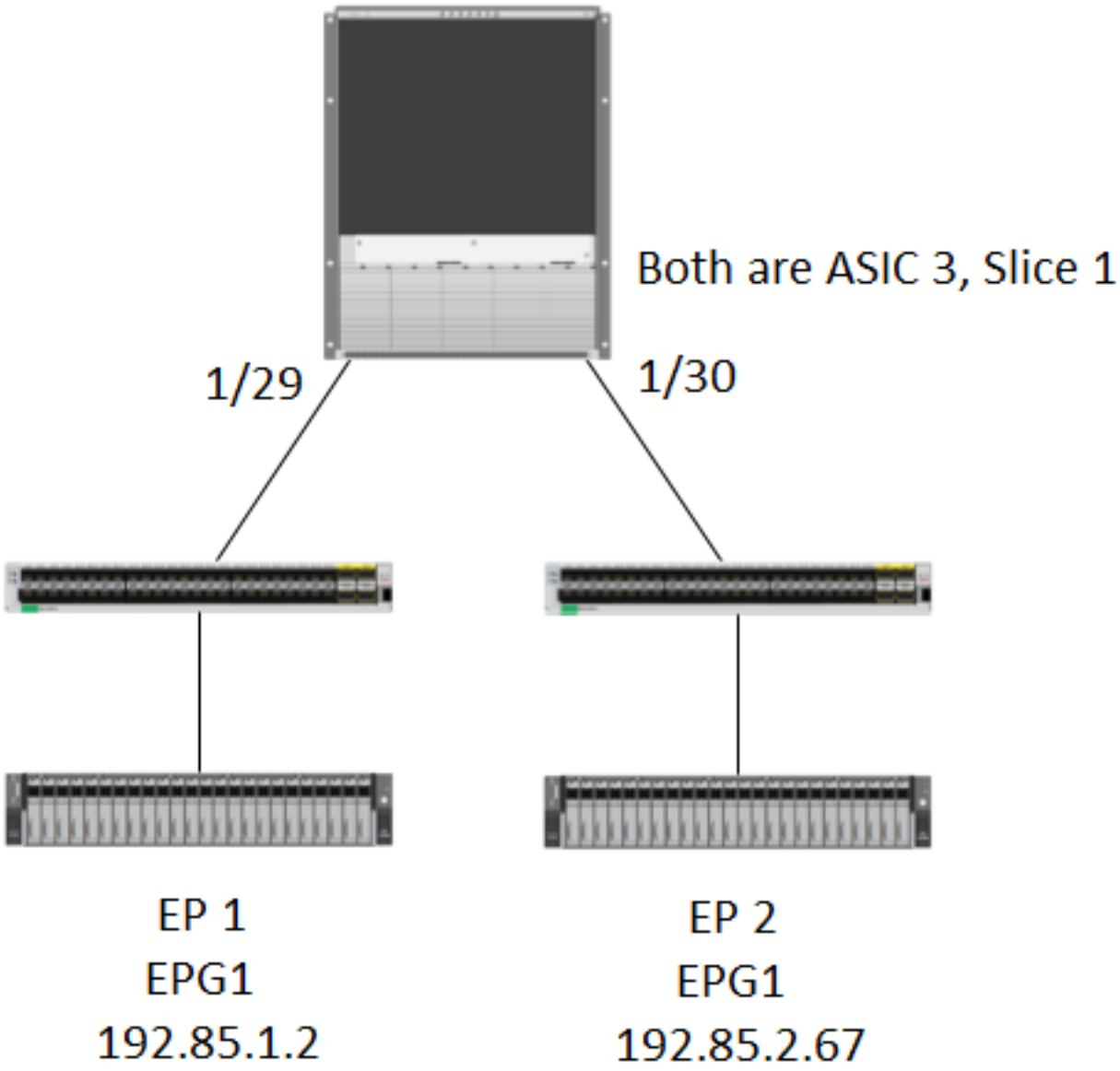
知)映射到LC上的正确接口：

```
================================================================================================================
=========================================
                      Uc   Uc                            |       Reprogram       |
| Rep |
                    I PC   Pc                          L |   R I R D    R  U U X | L Xla Ovx N
NI Vif    RwV    Ing  Egr  | V R | PROF H
IfId     Ifname      P Cfg  MbrID As AP Sl Sp Ss Ovec S | P P P S P Sp Sp C M L | 3 Idx Idx L3
L3 Tid    Tid    Lbl  Lbl  | S V | ID   I
================================================================================================================
=========================================
1f5      SpInBndMgmt 0 9de  1a    0  0  0  0  0  0    0   0 0 0 0 0 0  0  0 0 0   0 0   0   0  0
D-2d4  D-3e1  0    0      0 0   1    0
1a080000 Eth2/1      0 9a   1c    0  11 0  10 20 20   1   0 0 0 0 0 0  0  0 0 0   1 b   b   1  1
D-f3   D-61   100  0      0 0   1    0
1a081000 Eth2/2      0 9b   22    0  d  0  c  18 18   1   0 0 0 0 0 0  0  0 0 0   1 c   c   1  1
D-1ee  D-30b  100  0      0 0   1    0
1a084000 Eth2/5      0 9e   1e    0  3d 1  14 28 a8   1   0 0 0 0 0 0  0  0 0 0   1 1   1   1  1
D-19a  D-2ee  100  0      0 0   1    0
1a085000 Eth2/6      0 9f   24    0  39 1  10 20 a0   1   0 0 0 0 0 0  0  0 0 0   1 e   e   1  1
D-87   D-184  100  0      0 0   1    0
1a086000 Eth2/7      0 a0   26    0  35 1  c 18 98    1   0 0 0 0 0 0  0  0 0 0   1 d   d   1  1  D-
1d0  D-357  100  0      0 0   1    0
1a088000 Eth2/9      0 a2   20    1  d  0  c  18 18   1   0 0 0 0 0 0  0  0 0 0   0 0   0   0  0
D-3ea  D-1a9  100  0      0 0   1    0
```

以太网2/7是连接到枝叶5的接口。

额外场景：获取不在“hal internal-port pi”输出中的Ovector

拓扑



逻辑

在某些情况下，我们会在“**show platform internal hal l2 internal-port pi”表中捕获没有Ovector的**数据包。 在下面的场景中，我们实际上正在捕获从FM返回的数据包，因此我们需要查看另一个表以查看数据包选择的前面板端口。

请注意，上述拓扑是一个完全不同的环境，在该环境中可以获知中转流量（无代理路由）。 模块是N9K-X9732C-EX。

```
@module-1# debug platform internal tah elam asic 3
@module-1(DBG-elam)# trigger reset
@module-1(DBG-elam)# trigg init in-select 13 out-select 0
@module-1(DBG-elam-insel13)# set inner ipv4 src_ip 192.85.1.2 dst_ip 192.85.2.67
@module-1(DBG-elam-insel13)# star
@module-1(DBG-elam-insel13)# stat
ELAM STATUS
===========
Asic 3 Slice 0 Status Armed
Asic 3 Slice 1 Status Triggered
```

```
@module-1(DBG-elam-insel13)# report | grep ovector
 sug_elam_out_sidebnd_no_spare_vec.ovector_idx: 0xA0 <<<<<<<<<<<<<<<< now we look for this in
the "hal internal-port pi" command


@module-1# show platform internal hal l2 internal-port pi
No sandboxes exist
Num. of Sandboxes: 1
Legend:
-------
IfId:      Interface Id               IfName:     Interface Name
As:        Asic                       AP:         Asic Port
Sl:        Slice                      SP:         Slice Port
Ss:        Slice SrcId                Ovec:       Ovector
UcPcCfgId: Uc Pc CfgId                Lb Mbrid:   LB MbrId

Sandbox_ID: 0, BMP: 0x0
 Internal Port Count: 24

=====================================================================
                                                      UcPc  Lb
IfId     IfName                       As AP Sl SP Ss Ovec CfgId MbrId
=====================================================================
7d       -                            0  21 0  20 38 38   0     4
7e       -                            0  29 1  0  0  80   0     8
7f       -                            1  21 0  20 38 38   0     c
80       -                            1  29 1  0  0  80   0     10
81       -                            2  21 0  20 38 38   0     14
82       -                            2  29 1  0  0  80   0     18
83       -                            3  21 0  20 38 38   0     1c
84       -                            3  29 1  0  0  80   0     20
ad       -                            0  25 0  24 40 40   0     1
ae       -                            0  41 1  18 30 b0   0     6
af       -                            1  25 0  24 40 40   0     9
b0       -                            1  41 1  18 30 b0   0     e
b1       -                            2  25 0  24 40 40   0     11
b2       -                            2  41 1  18 30 b0   0     16
b3       -                            3  25 0  24 40 40   0     19
b4       -                            3  41 1  18 30 b0   0     1e
dd       -                            0  15 0  14 28 28   0     2
de       -                            0  4d 1  24 40 c0   0     5
df       -                            1  15 0  14 28 28   0     a
e0       -                            1  4d 1  24 40 c0   0     d
e1       -                            2  15 0  14 28 28   0     12
e2       -                            2  4d 1  24 40 c0   0     15
e3       -                            3  15 0  14 28 28   0     1a
e4       -                            3  4d 1  24 40 c0   0     1d  <<<<<<<  we cant find an
entry that matches 0xA0


@module-1# show platform internal hal l2 port gpd
Legend:
-------
<snip>

Sandbox_ID: 0, BMP: 0x0
 Port Count: 6

=============================================================================================================
=========================================
                      Uc   Uc                          |       Reprogram
|                                        | Rep |
                    I PC   Pc                       L |   R I R D    R  U U X | L Xla Ovx N  NI
Vif    RwV    Ing  Egr  | V R | PROF H
```

```
IfId     Ifname      P Cfg  MbrID As AP Sl Sp Ss Ovec S | P P P S P Sp Sp C M L | 3 Idx Idx L3
L3 Tid    Tid    Lbl  Lbl  | S V | ID   I
=============================================================================================
=========================================
1f5      SpInBndMgmt 0 9de  1a    0  0  0  0  0  0    0   0 0 0 0 0 0  0  0 0 0   0 0   0   0
0  D-2d4  D-3e1  0    0      0 0   1    0
1a000000 Eth1/1      0 1b   1c    0  11 0  10 20 20   1   0 0 0 0 0 0  0  0 0 0   1 1   1   1
1  D-13b  D-33b  500  0      1 0   3    0
1a01c000 Eth1/29     0 37   1e    3  3d 1  14 28 a8   1   0 0 0 0 0 0  0  0 0 0   1 8   8   1
1  D-3f2  D-7a   100  0      0 0   2    0
1a01d000 Eth1/30     0 38   20    3  39 1  10 20 a0   1   0 0 0 0 0 0  0  0 0 0   1 5   5   1
1  D-36e  D-362  100  0      0 0   2    0
1a01e000 Eth1/31     0 39   22    3  35 1  c  18 98   1   0 0 0 0 0 0  0  0 0 0   1 9   9   1
1  D-273  D-8    100  0      0 0   2    0
1a01f000 Eth1/32     0 3a   24    3  31 1  8  10 90   1   0 0 0 0 0 0  0  0 0 0   1 a   a   1
1  D-154  D-5d   100  0      0 0   2    0
```

1/30是连接到枝叶102的phys接口，通过拓扑、ASIC 3、片1进行验证