确定通过ACI交换矩阵的数据包流

目录

[简介](#)
[先决条件](#)
[要求](#)
[使用的组件](#)
[确定ACI交换矩阵数据包流](#)
[单BD/单EPG，在同一枝叶上带两个终端](#)
[单BD/单EPG，在不同枝叶上带两个终端](#)
[单个BD/两个EPG，在同一枝叶上的每个EPG中有一个终端](#)
[两个BD/两个EPG，每个EPG中有一个终端位于同一枝叶（路由数据包）上](#)

简介

本文档介绍如何在各种情况下确定通过以应用为中心的基础设施(ACI)交换矩阵的数据包流。

> **注意**：本文档中描述的所有情况都涉及运行ACI交换矩阵，以便跟踪硬件中的数据包流。

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：

- 由两台主干交换机和两台枝叶交换机组成的ACI交换矩阵
- ESXi主机，带有两个上行链路，可连接到每台枝叶交换机
- 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。

确定ACI交换矩阵数据包流

本节介绍可能使用ACI交换矩阵的各种情况以及如何确定数据包流。

单BD/单EPG，在同一枝叶上带两个终端

本节介绍如何验证同一枝叶交换机上同一终端组(EPG)/网桥域(BD)内两个终端的硬件编程和数据包流。如果虚拟机(VM)在同一主机上运行，由于它们位于同一EPG中，则流量会隔离到主机上的虚拟交换机(VS)，并且流量永远不必离开主机。如果VM在不同主机上运行，则以下信息适用。

您应首先检验的是是否获取了枝叶交换机上源和目标IP地址的介质访问控制(MAC)地址信息。以下是本示例中使用的MAC和IP地址信息：

- 源MAC地址:**0050.5695.17b7**
- 源 IP 地址：**192.168.3.2**
- 目的MAC地址:**0050.5695.248f**
- 目的 IP 地址：**192.168.3.3**

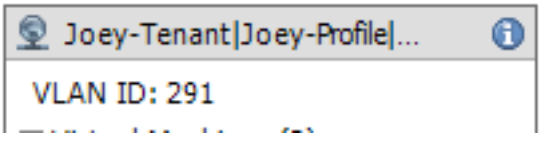
输入**show mac address-table**命令以验证以下信息：

```
leaf2# show mac address-table
Legend:
* - primary entry, G - Gateway MAC, (R) - Routed MAC, O - Overlay MAC
age - seconds since last seen,+ - primary entry using vPC Peer-Link,
(T) - True, (F) - False
VLAN MAC Address Type age Secure NTFY Ports/SWID.SSID.LID
---------+-----------------+--------+---------+------+----+------------------
16 0050.5695.248f dynamic - F F tunnel4
* 19       0050.5695.17b7    dynamic      -       F    F    eth1/31
* 19       0050.5695.248f    dynamic      -       F    F    eth1/31
```

如图所示，系统获取同一VLAN中两个终端的MAC地址。此VLAN是独立于平台(PI)的VLAN，对每台交换机都具有本地意义。要验证这是否是正确的PI VLAN，请连接到**vsh_lc**，然后在CLI中输入以下命令：

```
module-1# show system internal eltmc info vlan brief
VLAN-Info
VlanId HW_VlanId Type Access_enc Access_enc Fabric_enc Fabric_enc BDVlan
Type Type
================================================================================
9 11 BD_VLAN Unknown 0 VXLAN 16613250 9
10 12 BD_VLAN Unknown 0 VXLAN 15990734 10
13 13 FD_VLAN 802.1q 299 VXLAN 8507 10
16 14 BD_VLAN Unknown 0 VXLAN 16449431 16
17 15 FD_VLAN 802.1q 285 VXLAN 8493 16
18 16 BD_VLAN Unknown 0 VXLAN 15761386 18
19       17         FD_VLAN    802.1q       291     VXLAN      8499      18
```

HW_VlanId是Broadcom使用的VLAN。**VlanId**是PI VLAN，它映射到从VLAN池派生的**Access_enc** VLAN 291，并且是传播到分布式虚拟交换机(DVS)端口组的VLAN:



由于此流量在同一BD和同一VLAN中，因此应在Broadcom ASIC上本地交换流量。要验证Broadcom在硬件中是否有正确的条目，请连接到Broadcom外壳并查看第2层(L2)表：

```
leaf2# bcm-shell-hw
unit is 0
Available Unit Numbers: 0
bcm-shell.0> l2 show
mac=00:22:bd:f8:19:ff vlan=19 GPORT=0x7f modid=2 port=127 Static
mac=00:50:56:95:68:c4 vlan=25 GPORT=0x5f modid=0 port=95/xe94 Hit
mac=00:22:bd:f8:19:ff vlan=16 GPORT=0x7f modid=2 port=127 Static
mac=00:22:bd:f8:19:ff vlan=29 GPORT=0x7f modid=2 port=127 Static
mac=00:22:bd:f8:19:ff vlan=32 GPORT=0x7f modid=2 port=127 Static
mac=00:22:bd:f8:19:ff vlan=26 GPORT=0x7f modid=2 port=127 Static
mac=00:50:56:95:24:8f vlan=17 GPORT=0x1f modid=0 port=31/xe30 Hit
mac=00:22:bd:f8:19:ff vlan=18 GPORT=0x7f modid=2 port=127 Static
mac=00:22:bd:f8:19:ff vlan=21 GPORT=0x7f modid=2 port=127 Static
mac=00:22:bd:f8:19:ff vlan=34 GPORT=0x7f modid=2 port=127 Static
mac=00:50:56:95:26:5e vlan=25 GPORT=0x5f modid=0 port=95/xe94 Hit
mac=00:50:56:95:c3:6f vlan=24 GPORT=0x5f modid=0 port=95/xe94 Hit
mac=00:50:56:95:5c:4d vlan=28 GPORT=0x1e modid=0 port=30/xe29 Hit
mac=00:22:bd:f8:19:ff vlan=12 GPORT=0x7f modid=2 port=127 Static Hit
mac=00:22:bd:f8:19:ff vlan=11 GPORT=0x7f modid=2 port=127 Static
mac=00:50:56:95:17:b7 vlan=17 GPORT=0x1f modid=0 port=31/xe30 Hit
mac=00:50:56:95:4e:d3 vlan=30 GPORT=0x1e modid=0 port=30/xe29 Hit
mac=00:22:bd:f8:19:ff vlan=14 GPORT=0x7f modid=2 port=127 Static
```

输出显示Broadcom ASIC编程正确，流量应在VLAN 17中本地交换。

单BD/单EPG，在不同枝叶上带两个终端

本节介绍如何验证同一EPG/BD中不同枝叶交换机上两个终端的硬件编程和数据包流。

您首先应该检验的是，是否已获知枝叶交换机上源IP地址和目的IP地址的MAC地址信息。以下是本示例中使用的MAC和IP地址信息：

- 源MAC地址:0050.5695.17b7
- 源 IP 地址：192.168.3.2
- 目的MAC地址:0050.5695.bd89
- 目的IP地址:192.168.3.11

在两台枝叶交换机的CLI中输入**show mac address-table命令，以验证以下信息：**

```
leaf2# show mac address-table
Legend:
* - primary entry, G - Gateway MAC, (R) - Routed MAC, O - Overlay MAC
age - seconds since last seen,+ - primary entry using vPC Peer-Link,
(T) - True, (F) - False
VLAN MAC Address Type age Secure NTFY Ports/SWID.SSID.LID
---------+-----------------+--------+---------+------+----+------------------
* 19       0050.5695.17b7    dynamic      -       F    F    eth1/31
* 19 0050.5695.248f dynamic - F F eth1/31

leaf_1# show mac address-table
Legend:
* - primary entry, G - Gateway MAC, (R) - Routed MAC, O - Overlay MAC
age - seconds since last seen,+ - primary entry using vPC Peer-Link,
(T) - True, (F) - False
VLAN MAC Address Type age Secure NTFY Ports/SWID.SSID.LID
---------+-----------------+--------+---------+------+----+------------------
27 0050.5695.248f dynamic - F F tunnel7
27 0050.5695.17b7 dynamic - F F tunnel7
```

```
* 28       0050.5695.bd89    dynamic      -       F    F    eth1/25
```

如输出所示，源IP地址在第二个枝叶交换机(leaf2)上获知，而目的IP地址在第一个枝叶交换机(leaf_1)上获知。 由于这些流量位于不同的枝叶交换机上，因此必须将流量发送到第二台枝叶交换机上的NorthStar ASIC，以便流量可以上游发送到主干交换机。要遵循NorthStar逻辑，请连接到线卡vsh。

输入以下命令可查看本地条目列表：

```
leaf2# vsh_lc
module-1# show platform internal ns forwarding lst-l2
error opening file
: No such file or directory

==============================================================================
TABLE INSTANCE : 0
==============================================================================
Legend:
POS: Entry Position O: Overlay Instance
V: Valid Bit MD/PT: Mod/Port
PT: Pointer Type(A=Adj, E=ECMP, D=DstEncap N=Invalid)
PTR: ECMP/Adj/DstEncap/MET pointer
ML: MET Last
ST: Static PTH: Num Paths
BN: Bounce CP: Copy To CPU
PA: Policy Applied PI: Policy Incomplete
DL: Dst Local SP: Spine Proxy
------------------------------------------------------------------------------
MO SRC P M S B C P P D S
POS O VNID Address V DE MD/PT CLSS T PTR L T PTH N P A I L P
------------------------------------------------------------------------------
111 0 fd7f82 00:22:bd:f8:19:ff 1 0 00/00 1 A 0 0 1 1 0 0 0 1 0 0
131 0 f1ffde 00:22:bd:f8:19:ff 1 0 00/00 1 A 0 0 1 1 0 0 0 1 0 0
169 0 f37fd3 00:50:56:95:26:5e 1 0 00/24 4002 A 0 0 0 1 0 0 0 1 0 0
331 0 f37fd2 00:50:56:95:5c:4d 1 0 00/2e 8003 A 0 0 0 1 0 0 0 1 0 0
719 0 f3ffce 00:22:bd:f8:19:ff 1 0 00/00 1 A 0 0 1 1 0 0 0 1 0 0
945 0 f7ffae 00:22:bd:f8:19:ff 1 0 00/00 1 A 0 0 1 1 0 0 0 1 0 0
1390 0 fa7f9a 00:22:bd:f8:19:ff 1 0 00/00 1 A 0 0 1 1 0 0 0 1 0 0
1454 0 efffee 00:22:bd:f8:19:ff 1 0 00/00 1 A 0 0 1 1 0 0 0 1 0 0
1690 0 f37fd3 00:22:bd:f8:19:ff 1 0 00/00 1 A 0 0 1 1 0 0 0 1 0 0
1720 0 f37fd3 00:50:56:95:c3:6f 1 0 00/24 c002 A 0 0 0 1 0 0 0 1 0 0
1902 0 f1ffde 00:50:56:95:4e:d3 1 0 00/2e 8006 A 0 0 0 1 0 0 0 1 0 0
2176 0 f07fea 00:50:56:95:17:b7 1  0 00/0f 8004 A    0 0 0   1 0 0 0 0 0 0
2819 0 faff97 00:22:bd:f8:19:ff 1 0 00/00 1 A 0 0 1 1 0 0 0 1 0 0
3297 0 f07fea 00:22:bd:f8:19:ff 1 0 00/00 1 A 0 0 1 1 0 0 0 1 0 0

==============================================================================
TABLE INSTANCE : 1
==============================================================================
Legend:
POS: Entry Position O: Overlay Instance
V: Valid Bit MD/PT: Mod/Port
PT: Pointer Type(A=Adj, E=ECMP, D=DstEncap N=Invalid)
PTR: ECMP/Adj/DstEncap/MET pointer
ML: MET Last
ST: Static PTH: Num Paths
BN: Bounce CP: Copy To CPU
PA: Policy Applied PI: Policy Incomplete
DL: Dst Local SP: Spine Proxy
------------------------------------------------------------------------------
MO SRC P M S B C P P D S
POS O VNID Address V DE MD/PT CLSS T PTR L T PTH N P A I L P
```

```
--------------------------------------------------------------------------------
169 0 f37fd3 00:50:56:95:26:5e 1 0 00/24 4002 A e 0 0 1 0 0 0 0 1 0
331 0 f37fd2 00:50:56:95:5c:4d 1 0 00/2e 8003 A 9 0 0 1 0 0 0 0 1 0
1720 0 f37fd3 00:50:56:95:c3:6f 1 0 00/24 c002 A c 0 0 1 0 0 0 0 1 0
1902 0 f1ffde 00:50:56:95:4e:d3 1 0 00/2e 8006 A f 0 0 1 0 0 0 0 1 0
2176 0 f07fea 00:50:56:95:17:b7 1  0 00/0f 8004 A    d 0 0   1 0 0 0 0 1 0
3507 0 fa7f9a 00:50:56:95:3e:ee 1 0 00/2e c005 A 10 0 0 1 0 0 0 0 1 0
3777 0 f37fd3 00:50:56:95:68:c4 1 1 04/04 4002 A 11 0 0 1 1 0 0 0 0 0
3921 0 f07fea 00:50:56:95:24:8f 1 0 00/0f 8004 A d 0 0 1 0 0 0 0 1 0
```

输入以下命令以查看目标条目列表（查找目标MAC地址）：

```
module-1# show platform internal ns forwarding gst-l2
error opening file
: No such file or directory

================================================================================
TABLE INSTANCE : 0
================================================================================
Legend:
POS: Entry Position O: Overlay Instance
V: Valid Bit MD/PT: Mod/Port
PT: Pointer Type(A=Adj, E=ECMP, D=DstEncap N=Invalid)
PTR: ECMP/Adj/DstEncap/MET pointer
ML: MET Last
ST: Static PTH: Num Paths
BN: Bounce CP: Copy To CPU
PA: Policy Applied PI: Policy Incomplete
DL: Dst Local SP: Spine Proxy
--------------------------------------------------------------------------------
MO SRC P M S B C P P D S
POS O VNID Address V DE MD/PT CLSS T PTR L T PTH N P A I L P
--------------------------------------------------------------------------------
2139 0 ff7f72 00:50:56:95:7b:16 1 0 00/00 8006 A d 0 0 1 0 0 0 0 1 0
2195 0 faff97 00:50:56:95:5d:6e 1 0 00/00 8005 A f 0 0 1 0 0 0 0 1 0
3379 0 f07fea 00:50:56:95:bd:89 1 1  00/00 8004 A 10  0 0 1   0 0 0 0 0 0
4143 0 f07fea 00:50:56:95:17:b7 1 0 00/00 8004 A a 0 0 1 0 0 0 0 1 0
4677 0 f07feb 00:50:56:95:68:c4 1 0 00/00 4002 A e 0 0 1 0 0 0 0 1 0
5704 0 f07fea 00:50:56:95:24:8f 1 0 00/00 8004 A a 0 0 1 0 0 0 0 1 0
6191 0 f7ffaf 00:50:56:95:00:33 1 0 00/00 4007 A c 0 0 1 0 0 0 0 1 0
```

请注意这些输出中的指针(PTR)字段，即邻接指针。此值用于下一命令以查找目的封装VLAN。这是一个十六进制值，您必须将其转换为十进制值（十进制0 x 10表示16）。

在CLI中输入以下命令，16作为邻接指针：

```
module-1# show platform internal ns forwarding adj 16
error opening file
: No such file or directory

================================================================================
TABLE INSTANCE : 0
================================================================================
Legend
TD: TTL Dec Disable UP: USE PCID
DM: Dst Mac Rewrite SM: Src Mac Rewrite
RM IDX: Router Mac IDX SR: Seg-ID Rewrite
--------------------------------------------------------------------------------
ENCP T U USE D S RM S SRC
POS SEG-ID PTR D P PCI M DST-MAC M IDX R SEG-ID CLSS
--------------------------------------------------------------------------------
 16      0 2ffa 0 0    0 1 00:0c:0c:0c:0c:0c 0   0 0      0    0
```

请注意此输**出中**的ENCP PTR值，该值用于查找目标隧道终端(TEP)地址：

```
module-1# show platform internal ns forwarding encap 0x2ffa
error opening file
: No such file or directory

================================================================================
TABLE INSTANCE : 0
================================================================================
Legend
MD: Mode (LUX & RWX) LB: Loopback
LE: Loopback ECMP LB-PT: Loopback Port
ML: MET Last TD: TTL Dec Disable
DV: Dst Valid DT-PT: Dest Port
DT-NP: Dest Port Not-PC ET: Encap Type
OP: Override PIF Pinning HR: Higig DstMod RW
HG-MD: Higig DstMode KV: Keep VNTAG
--------------------------------------------------------------------------------
M PORT L L LB MET M T D DT DT E TST O H HG K M E
POS D FTAG B E PT PTR L D V PT NP T IDX P R MD V D T Dst MAC DIP
--------------------------------------------------------------------------------
12282 0 c00 0 1 0 0 0 0 0 0 0 3 7 0 0 0 0 0 3 00:00:00:00:00:00 192.168.56.93
```

在这种情况下，帧通过本地TEP的源IP地址和列出的TEP的目的IP地址封装在iVXLAN中。根据ELTMC输出，该BD的VXLAN ID为**15761386**，因此这是放入VXLAN数据包的ID。当流量到达另一端时，它被解封，并且由于目的MAC地址是本地的，因此它会从Broadcom的**l2 show命**令中从端口转发。

单个BD/两个EPG，在同一枝叶上的每个EPG中有一个终端

本节介绍如何验证不同EPG中但具有相同BD的两个终端的硬件编程和数据包流。流量流向同一枝叶交换机。这也称为物理本地到物理本地（PL到PL）桥接数据包。 它是桥*接的*，因为允许两个封装VLAN之间通信，而无需第3层(L3)接口执行路由。

您应首先检验的是枝叶交换机上源IP地址和目标IP地址的MAC地址信息是否在预期接口上获取(本例中为**1/48**)。 以下是本示例中使用的MAC和IP地址信息：

- 源MAC地址:**0050.5695.908b**
- 源 IP 地址：**192.168.1.50**
- 目的MAC地址:**0050.5695.bd89**
- 目的IP地址:**192.168.1.51**

在CLI中**输入show mac address-table命令以验证以下信息：**

```
leaf1# show mac address-table | grep 908b
* 34       0050.5695.908b    dynamic      -       F    F    eth1/48
leaf1# show mac address-table | grep bd89
* 38       0050.5695.bd89    dynamic      -       F    F    eth1/48
```

然后，您应进入Broadcom(BCM)外壳，并验证BCM是否获取了正确的MAC地址信息：

```
bcm-shell.0> l2 show
mac=00:50:56:95:bd:89 vlan=55 GPORT=0x30 modid=0 port=48/xe47
mac=00:50:56:95:90:8b vlan=54 GPORT=0x30 modid=0 port=48/xe47 Hit
```

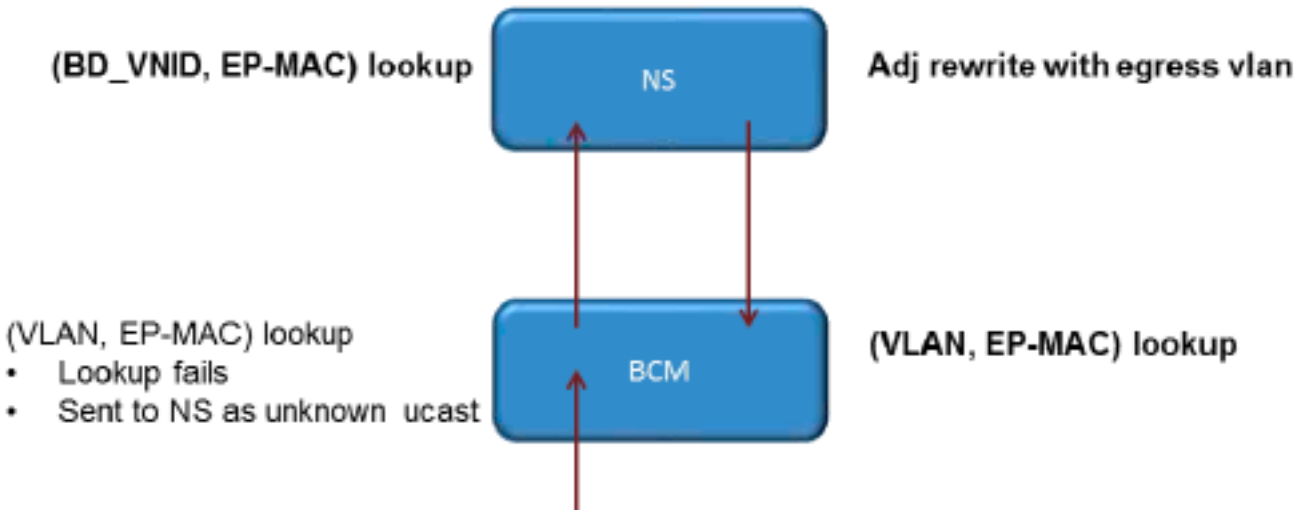
输出显示BCM已获知MAC地址信息；但是，MAC地址位于不同的VLAN中。这是预期的，因为流量

来自具有不同封装VLAN（不同EPG）的主机。

输入ELTMC，以验证BCM外壳中显示的**HW_VlanID**与两个封装VLAN的BD VLAN对应：

```
module-1# show system internal eltmc info vlan brief
VLAN-Info
VlanId HW_VlanId Type Access_enc Access_enc Fabric_enc Fabric_enc BDVlan
Type Type
================================================================================
13 15 BD_CTRL_VLAN 802.1q 4093 VXLAN 16777209 0
14 16 BD_VLAN Unknown 0 VXLAN 15957970 14
15 17 BD_VLAN Unknown 0 VXLAN 16613250 15
16 18 FD_VLAN 802.1q 301 VXLAN 8509 15
17 19 BD_VLAN Unknown 0 VXLAN 16220082 17
18 46 BD_VLAN Unknown 0 VXLAN 14745592 18
19 50 BD_VLAN Unknown 0 VXLAN 16646015 19
20 51 FD_VLAN 802.1q 502 VXLAN 8794 19
21 23 BD_VLAN Unknown 0 VXLAN 16121792 21
22 24 FD_VLAN 802.1q 538 VXLAN 8830 21
23 25 BD_VLAN Unknown 0 VXLAN 15826915 23
24 28 FD_VLAN 802.1q 537 VXLAN 8829 23
25 26 BD_VLAN Unknown 0 VXLAN 16351138 25
26 29 FD_VLAN 802.1q 500 VXLAN 8792 25
27 27 BD_VLAN Unknown 0 VXLAN 16678779 27
28 30 FD_VLAN 802.1q 534 VXLAN 8826 27
29 52 BD_VLAN Unknown 0 VXLAN 15859681 29
31 47 FD_VLAN 802.1q 602 VXLAN 9194 18
32 31 FD_VLAN 802.1q 292 VXLAN 8500 55
33 20 BD_VLAN Unknown 0 VXLAN 15761386 33
34       54         FD_VLAN    802.1q       299     VXLAN      8507      54
35 33 BD_VLAN Unknown 0 VXLAN 16449431 35
38       55         FD_VLAN    802.1q       300     VXLAN      8508      54
39 53 FD_VLAN 802.1q 501 VXLAN 8793 29
```

在此ELTMC输出中，您可以看到每个条目的**HW_VlanId**映射到流量进入交换机时标记的**Access_enc**（检查VMware端口组以验证其是否虚拟化），以及**VlanId**是出现在MAC地址表。在本例中，这是桥接连接，因为BD VLAN相同（两者都在VLAN 54上）。 下图显示BCM与NorthStar的交互：



NorthStar调整数据包，使用目的IP地址的**HW_VlanId**重写出口帧。这样，BCM在该VLAN中具有本地命中，并通过端口**1/48发送该帧**。

两个BD/两个EPG，每个EPG中有一个终端位于同一枝叶（路由数据包）上

本节介绍如何验证使用不同BD的不同EPG中两个终端的硬件编程和数据包流。流量流向同一枝叶交换机，但必须路由。这也称为PL到PL路由数据包。

您应首先检验的是，在中，枝叶交换机上源IP地址和目标IP地址的MAC地址信息是否都是通过预期接口获取的(本例中为1/48)。 以下是本示例中使用的MAC和IP地址信息：

- 源MAC地址:0050.5695.908b
- 源 IP 地址：192.168.1.50
- 默认网关:192.168.1.1
- 目的MAC地址:0050.5695.bd89
- 目的IP地址:192.168.3.51
- 默认网关:192.168.3.1

虽然您可以查看MAC地址表以验证L2信息，但L3路由流量解决方案的一个重要部分是终端管理器(EPM)。 EPM是跟踪特定设备上所有终端的过程。

验证EPM是否了解第一台枝叶交换机(Leaf1)上的两个终端:

```
leaf1# show endpoint ip 192.168.1.50
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
+---------------+---------------+-----------------+-------------+-------------+
56                      vlan-299    0050.5695.908b L                     eth1/48
Joey-Tenant:Joey-Internal        vlan-299      192.168.1.50 L
```

源IP地址在Ethernet 1/48上获知，并且它是此交换机的本地地址。

```
leaf1# show endpoint ip 192.168.3.51
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
+---------------+---------------+-----------------+-------------+-------------+
44                      vlan-291    0050.5695.bd89 L                     eth1/48
Joey-Tenant:Joey-Internal vlan-291 192.168.3.51 L
```

如图所示，目的IP地址是在Ethernet 1/48上获知的，并且是此交换机的本地地址。

要获取有关这些终端的更详细信息，请连接到线卡(LC):

```
leaf1# vsh_lc
module-1# show system internal epmc endpoint ip 192.168.1.50

MAC : 0050.5695.908b ::: Num IPs : 1
IP# 0 : 192.168.1.50 ::: IP# 0 flags :
Vlan id : 56 ::: Vlan vnid : 8507 ::: BD vnid : 15990734
```

```
VRF vnid : 2523136 ::: phy if : 0x1a02f000 ::: tunnel if : 0
Interface : Ethernet1/48
VTEP tunnel if : N/A ::: Flags : 0x80004c04
Ref count : 5 ::: sclass : 0x2ab5
Timestamp : 02/01/1970 00:43:53.129731
last mv timestamp 12/31/1969 19:00:00.000000 ::: ep move count : 0
previous if : 0 ::: loop detection count : 0
EP Flags : local,IP,MAC,class-set,timer,
Aging:Timer-type : Host-tracker timeout ::: Timeout-left : 423 ::: Hit-bit :
Yes ::: Timer-reset count : 406

PD handles:
Bcm l2 hit-bit : Yes
[L2]: Asic : NS ::: ADJ : 0x14 ::: LST SA : 0x83a ::: LST DA : 0x83a :::
GST ING : 0xedb ::: BCM : Yes
[L3-0]: Asic : NS ::: ADJ : 0x14 ::: LST SA : 0xe56 ::: LST DA : 0xe56 :::
GST ING : 0x12ae ::: BCM : Yes
::::
```

注意VRF vnid和BD vnid值。

```
module-1# show system internal epmc endpoint ip 192.168.3.51

MAC : 0050.5695.bd89 ::: Num IPs : 1
IP# 0 : 192.168.3.51 ::: IP# 0 flags :
Vlan id : 44 ::: Vlan vnid : 8499 ::: BD vnid : 15761386
VRF vnid : 2523136 ::: phy if : 0x1a02f000 ::: tunnel if : 0
Interface : Ethernet1/48
VTEP tunnel if : N/A ::: Flags : 0x80004c04
Ref count : 5 ::: sclass : 0x8004
Timestamp : 02/01/1970 00:43:53.130524
last mv timestamp 12/31/1969 19:00:00.000000 ::: ep move count : 0
previous if : 0 ::: loop detection count : 0
EP Flags : local,IP,MAC,class-set,timer,
Aging:Timer-type : Host-tracker timeout ::: Timeout-left : 532 ::: Hit-bit :
Yes ::: Timer-reset count : 1

PD handles:
Bcm l2 hit-bit : Yes
[L2]: Asic : NS ::: ADJ : 0x15 ::: LST SA : 0x28e ::: LST DA : 0x28e :::
GST ING : 0xd33 ::: BCM : Yes
[L3-0]: Asic : NS ::: ADJ : 0x15 ::: LST SA : 0x497b ::: LST DA : 0x497b :::
GST ING : 0x1e98 ::: BCM : Yes
::::
```

此输出中的VRF vnid值是相同的，因为两个路由都是路由表（相同情景）中相同虚拟路由和转发(VRF)的一部分。 BD vnid值不同，因为两个终端位于不同的BD中。

正如您查看NorthStar表以验证L2级MAC地址的硬件编程一样，您也可以执行同样的操作以验证L3表：

```
module-1# show platform internal ns forwarding lst-l3
error opening file
: No such file or directory

==============================================================================
TABLE INSTANCE : 0
==============================================================================
Legend:
POS: Entry Position O: Overlay Instance
V: Valid Bit MD/PT: Mod/Port
```

```
PT: Pointer Type(A=Adj, E=ECMP, D=DstEncap N=Invalid)
PTR: ECMP/Adj/DstEncap/MET pointer
ML: MET Last
ST: Static PTH: Num Paths
BN: Bounce CP: Copy To CPU
PA: Policy Applied PI: Policy Incomplete
DL: Dst Local SP: Spine Proxy
--------------------------------------------------------------------------------
MO SRC P M S B C P P D S
POS O VNID Address V DE MD/PT CLSS T PTR L T PTH N P A I L P
--------------------------------------------------------------------------------
2881 0 268000 192.168.1.1       1  0 00/00    1 A    0 0 1   1 0 0 0 1 0 0
3003 0 208001 80.80.80.10 1 0 00/14 800d A 0 0 0 1 0 0 0 1 0 0
3051 0 208001 30.30.30.30 1 0 00/14 c009 A 0 0 0 1 0 0 0 0 0 0
3328 0 268000 192.168.2.1 1 0 00/00 1 A 0 0 1 1 0 0 0 1 0 0
3670 0 268000 192.168.1.50      1  0 00/09 2ab5 A    0 0 0   1 0 0 0 0 0 0
3721 0 2b8001 50.50.50.1 1 0 00/00 1 A 0 0 1 1 0 0 0 1 0 0
3903 0 268000 192.168.3.1       1  0 00/00    1 A    0 0 1   1 0 0 0 1 0 0
18811 0 268000 192.168.3.51 1 0 00/09 8004 A 0 0 0 1 0 0 0 0 0 0
```

下图说明了通过ASIC的流：