

Determinar o fluxo de pacotes através de uma estrutura da ACI

Contents

[Introduction](#)

[Prerequisites](#)

[Requirements](#)

[Componentes Utilizados](#)

[Determine o fluxo de pacote de estrutura da ACI](#)

[BD/EPG único com dois endpoints no mesmo folheto](#)

[BD/EPG único com dois endpoints em folhetos diferentes](#)

[BD/dois EPGs únicos com um endpoint em cada EPG na mesma folha](#)

[Dois BDs/dois EPGs com um endpoint em cada EPG na mesma folha \(pacote roteado\)](#)

Introduction

Este documento descreve como determinar o fluxo de pacotes através de uma estrutura da Application Centric Infrastructure (ACI) em várias situações.

Note: Todas as situações descritas neste documento envolvem uma estrutura de ACI operacional para que o fluxo de pacotes no hardware possa ser rastreado.

Prerequisites

Requirements

Não existem requisitos específicos para este documento.

Componentes Utilizados

As informações neste documento são baseadas nas seguintes versões de hardware e software:

- Uma estrutura da ACI que consiste em dois switches spine e dois switches leaf
- Um host ESXi com dois uplinks que vão para cada um dos switches leaf
- Um Application Policy Infrastructure Controller (APIC) usado para a configuração inicial

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is

live, make sure that you understand the potential impact of any command.

Determine o fluxo de pacote de estrutura da ACI

Esta seção descreve as várias situações em que uma estrutura da ACI pode ser usada e como determinar o fluxo do pacote.

BD/EPG único com dois endpoints no mesmo folheto

Esta seção descreve como verificar a programação de hardware e o fluxo de pacotes para dois endpoints no mesmo grupo de endpoints (EPG)/domínio de bridge (BD) no mesmo switch leaf. Se as máquinas virtuais (VMs) são executadas no mesmo host, já que estão no mesmo EPG, o tráfego é isolado para o switch virtual (VS) no host e o tráfego nunca precisa sair do host. Se as VMs são executadas em hosts diferentes, as informações a seguir se aplicam.

A primeira coisa que você deve verificar é se as informações de endereço de Controle de Acesso ao Meio (MAC - Media Access Control) para os endereços IP origem e destino no switch Leaf são aprendidas. Estas são as informações de endereço MAC e IP usadas neste exemplo:

- Endereço MAC origem: **0050.5695.17b7**
- Endereço IP origem: **192.168.3.2**
- Endereço MAC destino: **0050.5695.248f**
- Endereço IP de destino: **192.168.3.3**

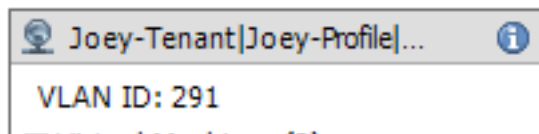
Insira o comando **show mac address-table** para verificar essas informações:

```
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
```

Como mostrado, o sistema aprende os endereços MAC para ambos os endpoints na mesma VLAN. Essa VLAN é a VLAN independente de plataforma (PI) e é localmente significativa para cada switch. Para verificar se esta é a VLAN PI correta, conecte-se à **vsh_lc** e insira este comando na CLI:

```
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
```

O **HW_VlanId** é a VLAN usada pelo Broadcom. O **VlanId** é a VLAN PI, que mapeia para a **Access_enc** VLAN 291 derivada do pool de VLANs e é a VLAN propagada para o grupo de portas do Switch virtual distribuído (DVS - Distributed Virtual Switch):



Como esse fluxo de tráfego está no mesmo BD e na mesma VLAN, o tráfego deve ser comutado localmente no ASIC Broadcom. Para verificar se a Broadcom tem as entradas corretas no hardware, conecte-se à shell Broadcom e visualize a tabela da Camada 2 (L2):

```
leaf2# bcm-shell-hw
unit is 0
Available Unit Numbers: 0
bcm-shell.0> 12 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
```

A saída mostra que a programação de ASIC Broadcom está correta e que o tráfego deve comutar localmente na VLAN 17.

BD/EPG único com dois endpoints em folhetos diferentes

Esta seção descreve como verificar a programação de hardware e o fluxo de pacotes para dois endpoints dentro do mesmo EPG/BD, mas em diferentes switches leaf.

A primeira coisa que você deve verificar é se as informações de endereço MAC para os endereços IP origem e destino nos switches leaf são aprendidas. Estas são as informações de endereço MAC e IP usadas neste exemplo:

- Endereço MAC origem: **0050.5695.17b7**
- Endereço IP origem: **192.168.3.2**
- Endereço MAC destino: **0050.5695.bd89**
- Endereço IP destino: **192.168.3.11**

Insira o comando **show mac address-table** na CLI de ambos os switches Leaf para verificar essas informações:


```

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 flffde 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 flffde 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

```

Insira este comando para ver uma lista das entradas de destino (procure o endereço MAC de destino):

```

module-1# show platform internal ns forwarding gst-12
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
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

```

Anote o campo **Ponteiro (PTR)** nessas saídas, que é o ponteiro de adjacência. Esse valor é usado no próximo comando para encontrar a VLAN encapsulada de destino. Esse é um valor HEX que você deve converter em um valor decimal (0 x 10 em decimal é 16).

Insira este comando na CLI, com **16** como o ponteiro de adjacência:

```
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
```

Anote o valor do **PTR ENCP** nesta saída, que é usado para encontrar o endereço de destino do Ponto de Extremidade do Túnel (TEP):

```
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 3 00:00:00:00:00:00 192.168.56.93
```

Nesse caso, o quadro é encapsulado em iVXLAN através do endereço IP de origem do TEP local e do endereço IP de destino do TEP listado. Com base na saída ELTMC, o ID da VXLAN para esse BD é **15761386**, portanto, esse é o ID que é colocado no pacote VXLAN. Quando o tráfego chega ao outro lado, é desencapsulado e, como o endereço MAC de destino é local, é encaminhado para fora da porta no comando **I2 show** da Broadcom.

BD/dois EPGs únicos com um endpoint em cada EPG na mesma folha

Esta seção descreve como verificar a programação de hardware e o fluxo de pacotes para dois endpoints em EPGs diferentes, mas com o mesmo BD. O tráfego flui para o mesmo switch

Leaf. Isso também é conhecido como pacote de conexão física local a físico (PL-to-PL). Ele é *Bridged* porque a comunicação é permitida entre duas VLANs encapsuladas sem a necessidade de uma interface de Camada 3 (L3) para executar o roteamento.

A primeira coisa que você deve verificar é se as informações de endereço MAC para os endereços IP origem e destino nos switches Leaf são aprendidas na interface esperada (1/48 nesse caso). Estas são as informações de endereço MAC e IP usadas neste exemplo:

- Endereço MAC origem: **0050.5695.908b**
- Endereço IP origem: **192.168.1.50**
- Endereço MAC destino: **0050.5695.bd89**
- Endereço IP destino: **192.168.1.51**

Insira o comando **show mac address-table** na CLI para verificar essas informações:

```
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
```

Em seguida, você deve entrar no shell Broadcom (BCM) e verificar se o BCM aprende as informações corretas do endereço MAC:

```
bcm-shell.0> 12 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
```

A saída mostra que o BCM aprendeu as informações de endereço MAC; no entanto, os endereços MAC estão em VLANs diferentes. Isso é esperado, à medida que o tráfego entra do host com diferentes VLANs encapsuladas (EPGs diferentes).

Entre no ELTMC para verificar o **HW_VlanID** exibido no shell BCM contra a VLAN BD para as duas VLANs encapsuladas:

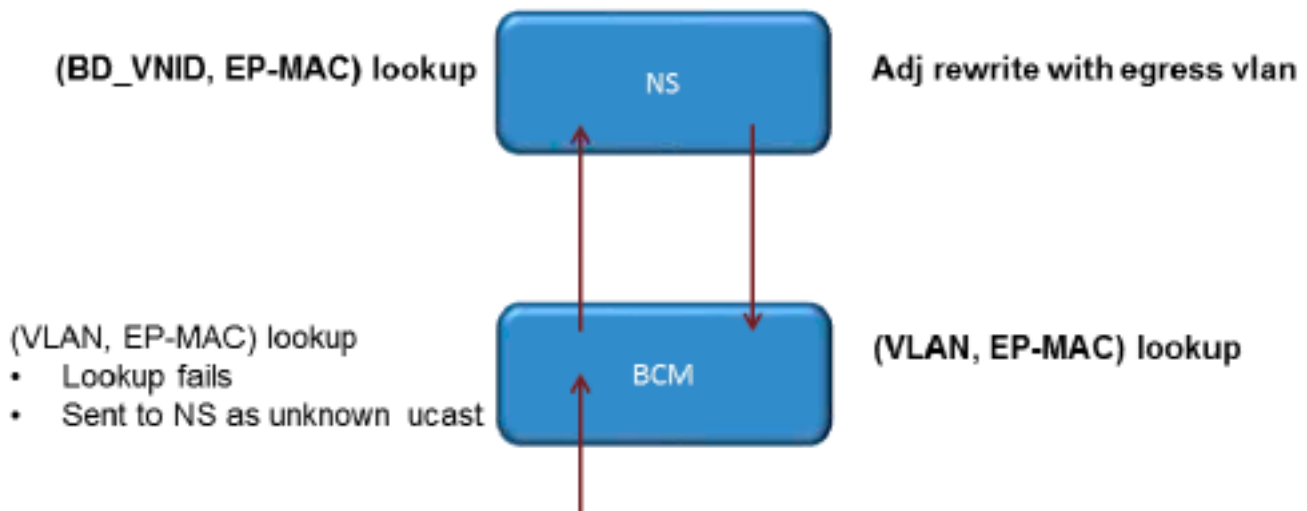
```
module-1# show system internal eltmc info vlan brief
VLAN-Info
VlanId HW_VlanId Type Access_enc Access_enc Fabric_enc Fabric_enc BDVlan
Type Type
=====
13 15 BD_CTRL_VLAN 802.1q 4093 VXLAN 16777209 0
14 16 BD_VLAN Unknown 0 VXLAN 15957970 14
15 17 BD_VLAN Unknown 0 VXLAN 16613250 15
16 18 FD_VLAN 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

```

Nesta saída do ELTMC, você pode ver que o **HW_VlanId** para cada entrada é mapeado para o **Access_enc** com o qual o tráfego é marcado quando entra no switch (verifique os grupos de portas do VMware para verificar se é virtualizado) e se o **VlanId** é a VLAN IP que aparece na tabela de endereços MAC. Essa é uma conexão com bridge nesse caso porque a VLAN BD é a mesma (ambas estão na VLAN 54). Este diagrama mostra a interação entre BCM e NorthStar:



O NorthStar ajusta o pacote e regrava o quadro de saída com o **HW_VlanId** do endereço IP de destino. Dessa forma, o BCM tem um acerto local nessa VLAN e envia o quadro pela porta **1/48**.

Dois BDs/dois EPGs com um endpoint em cada EPG na mesma folha (pacote roteado)

Esta seção descreve como verificar a programação de hardware e o fluxo de pacotes para dois endpoints em EPGs diferentes que usam BDs diferentes. O tráfego flui para o mesmo switch Leaf, mas deve ser roteado. Isso também é conhecido como um pacote *roteado* PL-to-PL.

A primeira coisa que você deve verificar é se as informações de endereço MAC para os endereços IP origem e destino no switch Leaf aprenderam na interface esperada (**1/48** nesse caso). Estas são as informações de endereço MAC e IP usadas neste exemplo:

- Endereço MAC origem: **0050.5695.908b**
- Endereço IP origem: **192.168.1.50**
- Gateway padrão: **192.168.1.1**
- Endereço MAC destino: **0050.5695.bd89**
- Endereço IP destino: **192.168.3.51**
- Gateway padrão: **192.168.3.1**

Embora você possa visualizar a tabela de endereços MAC para verificar as informações de L2, uma parte importante da solução para o tráfego roteado de L3 é o Endpoint Manager (EPM). O EPM é o processo que rastreia todos os endpoints em um dispositivo específico.

Verifique se o EPM tem conhecimento dos dois endpoints no primeiro switch Leaf (**Leaf1**):

```
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
O endereço IP origem é aprendido na Ethernet 1/48, e é local para esse switch.
```

```
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
Como mostrado, o endereço IP destino é aprendido na Ethernet 1/48 e é local para esse switch.
```

Para obter informações mais detalhadas sobre esses endpoints, conecte-se à Placa de linha (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
::::
```

Anote os valores VRF vnid e BD vnid.

```
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 vniid : 8499 ::: BD vniid : 15761386
VRF vniid : 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
::::
```

O valor **VRF vniid** nesta saída é o mesmo porque ambas as rotas fazem parte do mesmo Virtual Routing and Forwarding (VRF) na tabela de roteamento (mesmo contexto). O valor **de vniid BD** é diferente, já que os dois endpoints estão em BDs diferentes.

Assim como você visualizou as tabelas do NorthStar para verificar a programação de hardware para os endereços MAC em um nível L2, você pode fazer o mesmo para verificar a tabela L3:

```
module-1# show platform internal ns forwarding lst-13
```

```
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
```

Este diagrama ilustra o fluxo através dos ASICs:

