排除ACI交换矩阵内转发故障 — MultiPod转发

目录

简介

本文档介绍了解ACI多Pod转发场景并对其进行故障排除的步骤。

背景信息

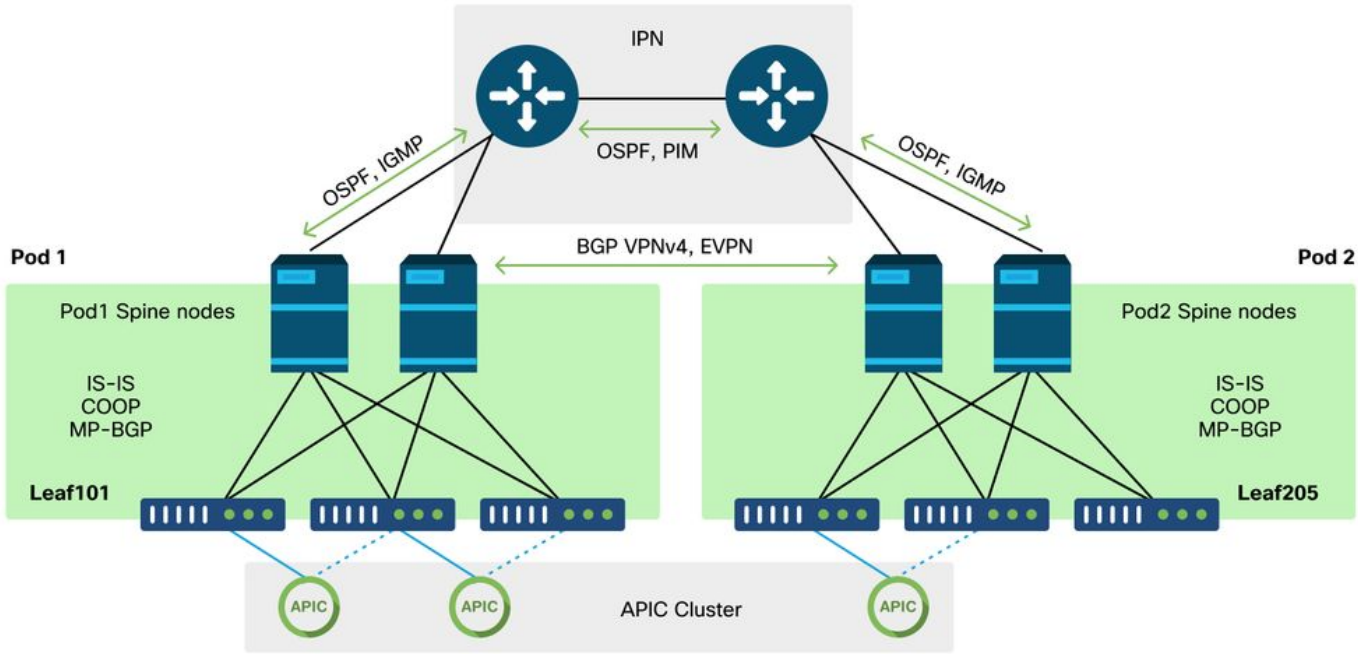
本文档中的内容摘自 思科以应用为中心的基础设施故障排除（第二版） 书，特别是 **交换矩阵内转发 — 多Pod转发** 第章。

多Pod转发概述

本章将介绍如何对多Pod环境中各Pod之间的连接不正常的情况进行故障排除

在查看具体的故障排除示例之前，务必花些时间从较高的层次了解多Pod组件。

多Pod组件



与传统ACI交换矩阵类似，多Pod交换矩阵仍被视为单个ACI交换矩阵，并依赖单个APIC集群进行管理。

在每个单独的Pod中，ACI在重叠中利用与传统交换矩阵相同的协议。其中包括IS-IS，用于交换TEP信息以及组播传出接口(OIF)选择、COOP（用于全局终端存储库）和BGP VPNv4（用于通过交换矩阵分发外部路由器）。

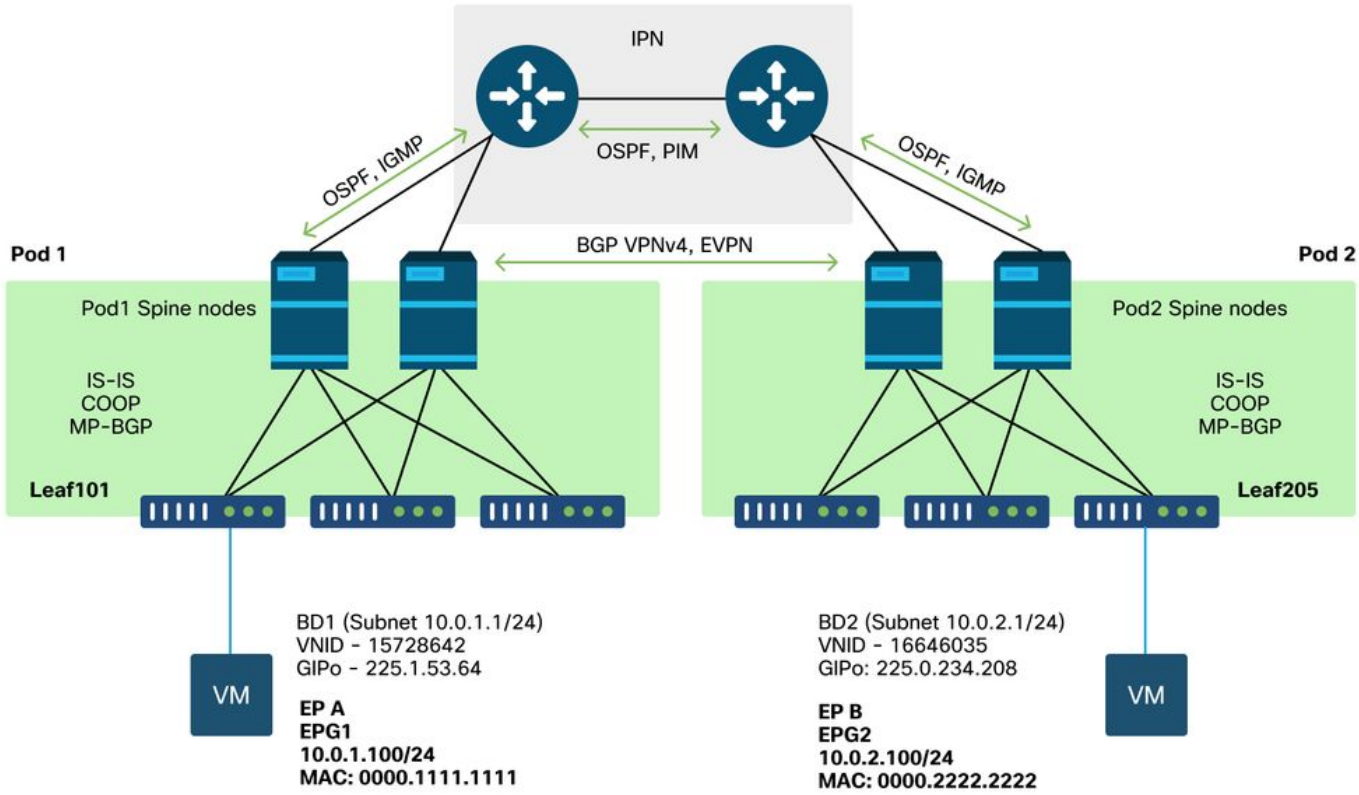
多Pod构建在这些组件上，因为它必须将每个Pod连接在一起。

- 要交换有关远程Pod中TEP的路由信息，OSPF用于通过IPN通告汇总TEP池。
- 为了交换从一个Pod获知的外部路由，BGP VPNv4 address-family在脊柱节点之间扩展。每个Pod都成为独立的路由反射器集群。
- 要跨Pod同步终端以及存储在COOP中的其他信息，BGP EVPN地址系列在脊柱节点之间扩展。
- 最后，为了处理Pod中广播、未知单播和组播(BUM)流量的泛洪，每个Pod中的主干节点充当

IGMP主机，IPN路由器通过双向PIM交换组播路由信息。
大部分多Pod故障排除场景和工作流程类似于单Pod ACI交换矩阵。此多Pod部分将主要介绍单Pod与多Pod转发之间的区别。

多Pod拓扑示例

与故障排除任何场景一样，从了解预期状态开始很重要。本章示例参考此拓扑。



多Pod转发故障排除的一般工作流程

在高级级别，当调试多Pod转发问题时，可以评估以下步骤：

1. 流是单播还是多目标流？请记住，即使预期流在工作状态下是单播，如果ARP未解析，它也是多目标流。
2. 流是路由还是桥接？传统上，从ACI角度来说，路由流是指目的MAC地址为ACI上配置的网关拥有的路由器MAC地址的任何流。此外，如果禁用ARP泛洪，则入口枝叶将基于目标IP地址进行路由。如果目的MAC地址不属于ACI，则交换机将根据MAC地址进行转发，或者遵循网桥域中配置的“未知单播”行为。
3. 入口枝叶是否正在丢弃流？fTriage和ELAM是确认这一点的最佳工具。

如果流是第3层单播：

1. 入口枝叶是否具有与源EPG相同的VRF中的目标IP的终端学习？如果是，它将始终优先于任何获知的路由。枝叶将直接转发到获取终端的隧道地址或出口接口。
2. 如果没有终端学习，入口枝叶是否具有已设置“无处不在”标志的目标路由？这表示目标子网配置为桥接域子网，且下一跳应是本地Pod中的主干代理。

3. 如果没有无处不在的路由，那么最后的手段将是通过L3Out获知的任何路由。此部分与单Pod L3Out转发相同。

如果流是第2层单播：

1. 入口枝叶是否具有与源EPG相同的网桥域中的目标MAC地址的终端学习？如果是，枝叶将转发到远程隧道IP或转发到获取终端的本地接口。
2. 如果在源网桥域中没有获取目的MAC地址，则枝叶将根据BD设置为“unknown-unicast”行为进行转发。如果设置为“泛洪”，则枝叶将泛洪到分配给网桥域的GIPo组播组。本地和远程Pod应获取一个泛洪副本。如果设置为“硬件代理”，则帧将发送到主干进行代理查找，并根据主干的COOP条目转发。

由于与BUM相比，单播的故障排除输出会有很大不同，因此在进入BUM之前会考虑单播的工作输出和场景。

多Pod单播故障排除工作流程

按照拓扑，浏览从leaf205上的10.0.2.100到leaf101上的10.0.1.100的流。

请注意，在继续此处之前，必须确认源是否已解析网关（路由流）或目标MAC地址（桥接流）的ARP

1.确认入口枝叶收到数据包。使用“工具”部分中显示的ELAM CLI工具以及4.2中提供的ereport输出。同时还会使用ELAM Assistant应用。

```
module-1# debug platform internal tah elam asic 0
module-1(DBG-elam)# trigger reset
module-1(DBG-elam)# trigger init in-select 6 out-select 1
module-1(DBG-elam-insel6)# set outer ipv4 src_ip 10.0.2.100 dst_ip 10.0.1.100
module-1(DBG-elam-insel6)# start
module-1(DBG-elam-insel6)# status
ELAM STATUS
===========
Asic 0 Slice 0 Status Armed
Asic 0 Slice 1 Status Triggered
```

请注意，ELAM已触发，确认入口交换机上已收到数据包。现在查看报告中的几个字段，因为输出内容非常丰富。

```
================================================================================================
===========
                                                    Captured Packet

================================================================================================
===========

------------------------------------------------------------------------------------------------
-----------
Outer Packet Attributes
------------------------------------------------------------------------------------------------
-----------
Outer Packet Attributes       : l2uc ipv4 ip ipuc ipv4uc
Opcode                        : OPCODE_UC

------------------------------------------------------------------------------------------------
```

```
-----------
Outer L2 Header
--------------------------------------------------------------------------------------------------
-----------
Destination MAC               : 0022.BDF8.19FF
Source MAC                    : 0000.2222.2222
802.1Q tag is valid           : yes( 0x1 )
CoS                           : 0( 0x0 )
Access Encap VLAN             : 1021( 0x3FD )

--------------------------------------------------------------------------------------------------
------------
Outer L3 Header
--------------------------------------------------------------------------------------------------
------------
L3 Type                       : IPv4
IP Version                    : 4
DSCP                          : 0
IP Packet Length              : 84 ( = IP header(28 bytes) + IP payload )
Don't Fragment Bit            : not set
TTL                           : 255
IP Protocol Number            : ICMP
IP CheckSum                   : 10988( 0x2AEC )
Destination IP                : 10.0.1.100
Source IP                     : 10.0.2.100
```

报告中包含更多有关数据包去向的信息，但ELAM助理应用目前对于解释此数据更有用。本章稍后将介绍此流程的ELAM Assistant输出。

2.入口枝叶是否将目标作为入口VRF中的终端？如果没有，是否有路由？

```
a-leaf205# show endpoint ip 10.0.1.100 detail
Legend:
s - arp              H - vtep             V - vpc-attached     p - peer-aged
R - peer-attached-rl B - bounce           S - static           M - span
D - bounce-to-proxy  O - peer-attached    a - local-aged       m - svc-mgr
L - local            E - shared-service
+----------------------------------+---------------+-----------------+--------------+----------
---+-----------------------------+
    VLAN/                           Encap           MAC Address       MAC Info/
Interface     Endpoint Group
    Domain                          VLAN            IP Address        IP Info
                  Info
+----------------------------------+---------------+-----------------+--------------+----------
---+-----------------------------+
```

上述命令中没有输出表示未获知目的IP。然后检查路由表。

```
a-leaf205# show ip route 10.0.1.100 vrf Prod:Vrf1
IP Route Table for VRF "Prod:Vrf1"
'*' denotes best ucast next-hop
'**' denotes best mcast next-hop
'[x/y]' denotes [preference/metric]
'%<string>' in via output denotes VRF <string>

10.0.1.0/24, ubest/mbest: 1/0, attached, direct, pervasive
  *via 10.0.120.34%overlay-1, [1/0], 01:55:37, static, tag 4294967294
       recursive next hop: 10.0.120.34/32%overlay-1
```

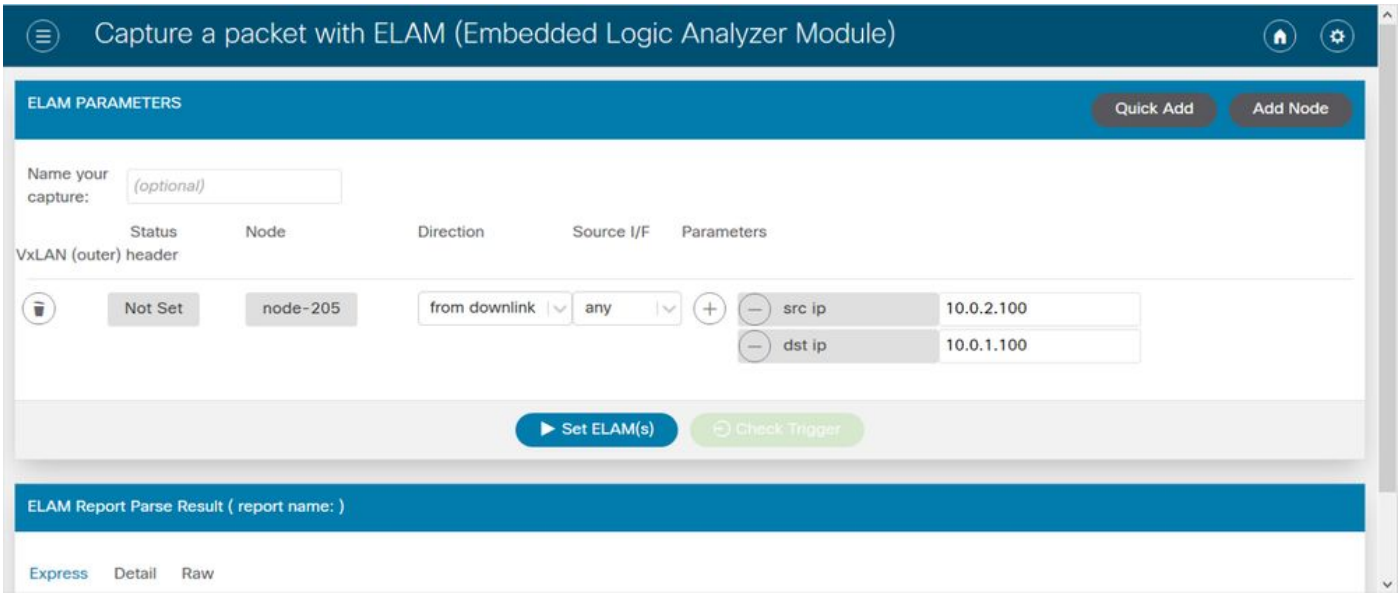
在上面的输出中，可以看到一个普遍标志，表明这是桥接域子网路由。下一跳应该是主干上的任播代理地址。

```
a-leaf205# show isis dtep vrf overlay-1 | grep 10.0.120.34
10.0.120.34        SPINE    N/A              PHYSICAL,PROXY-ACAST-V4
```

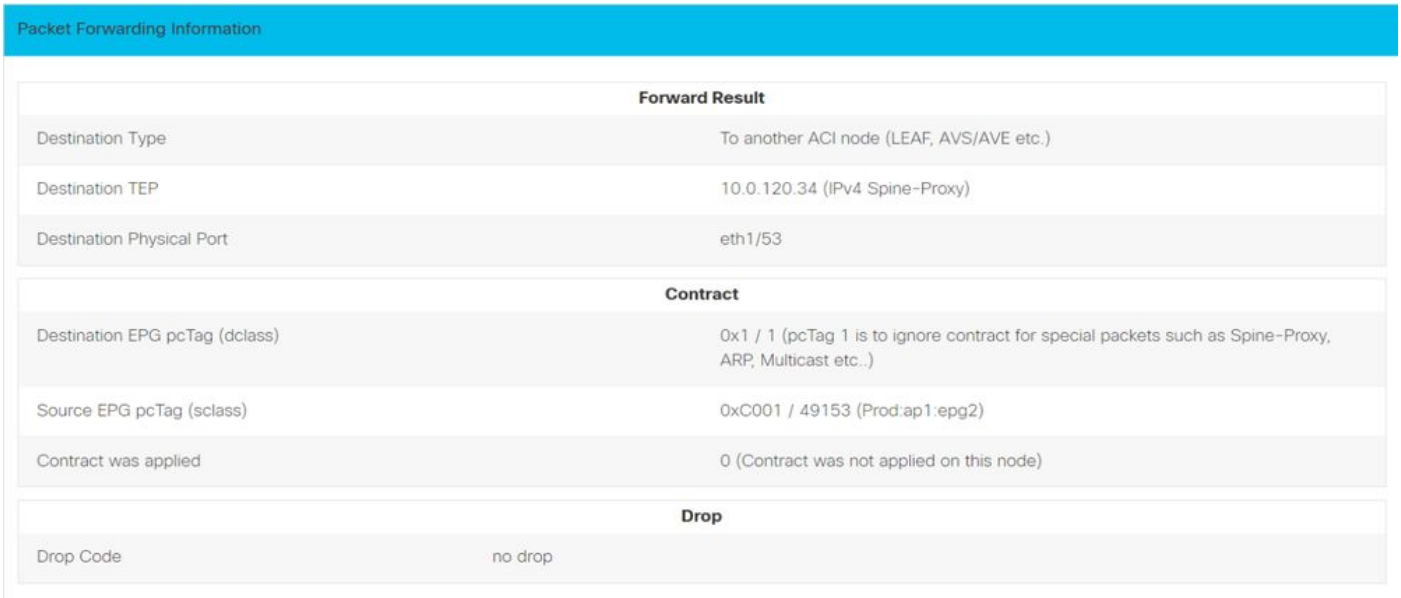
请注意，如果终端是在隧道或物理接口上获知的，这将优先，导致数据包直接在那里转发。有关详细信息，请参阅本书的“外部转发”一章。

使用ELAM Assistant确认上述输出中所见的转发决策。

ELAM助理配置



检验转发决策



以上输出显示入口枝叶正在将数据包转发到IPv4主干代理地址。这是预期会发生的情况。

3.在脊柱上确认目标IP存在于COOP中，以便代理请求生效。

有多种方法可以获取主干上的COOP输出，例如，使用“show coop internal info ip-db”命令查看它：

```
a-spine4# show coop internal info ip-db | grep -B 2 -A 15 "10.0.1.100"

------------------------------
IP address : 10.0.1.100
Vrf : 2392068 <-- This vnid should correspond to vrf where the IP is learned. Check operational
tab of the tenant vrfs
Flags : 0x2
EP bd vnid : 15728642
EP mac :  00:00:11:11:11:11
Publisher Id : 192.168.1.254
Record timestamp : 12 31 1969 19:00:00 0
Publish timestamp : 12 31 1969 19:00:00 0
Seq No: 0
Remote publish timestamp: 09 30 2019 20:29:07 9900483
URIB Tunnel Info
Num tunnels : 1
      Tunnel address : 10.0.0.34 <-- When learned from a remote pod this will be an External
Proxy TEP. We'll cover this more
      Tunnel ref count : 1

------------------------------
```

要在主干上运行的其他命令：

查询L2条目的COOP:

```
moquery -c coopEpRec -f 'coop.EpRec.mac=="00:00:11:11:22:22"
```

查询L3条目的COOP并获取父L2条目：

```
moquery -c coopEpRec -x rsp-subtree=children 'rsp-subtree-
filter=eq(coopIpv4Rec.addr,"192.168.1.1")' rsp-subtree-include=required
```

仅查询l3条目的COOP:

```
moquery -c coopIpv4Rec -f 'coop.Ipv4Rec.addr=="192.168.1.1"'
```

多个moquery的有用之处在于，它们还可以直接在APIC上运行，用户可以查看在命令行界面中记录的所有主干。

4.多Pod主干代理转发决策

如果主干的COOP条目指向本地Pod中的隧道，则转发基于传统ACI行为。

请注意，可以通过从APIC运行moquery **-c ipv4Addr -f 'ipv4.Addr.addr=="<tunnel address>"'，在交换矩阵中验证TEP的所有者**

在代理方案中，隧道下一跳是10.0.0.34。此IP地址的所有者是谁？:

```
a-apic1# moquery -c ipv4Addr -f 'ipv4.Addr.addr=="10.0.0.34"' | grep dn
dn               : topology/pod-1/node-1002/sys/ipv4/inst/dom-overlay-1/if-[lo9]/addr-
[10.0.0.34/32]
dn               : topology/pod-1/node-1001/sys/ipv4/inst/dom-overlay-1/if-[lo2]/addr-
[10.0.0.34/32]
```

此IP由Pod 1中的两个主干节点拥有。这是一个称为外部代理地址的特定IP。与ACI具有由Pod内的主干节点拥有的代理地址一样（请参阅本节的步骤2），也有分配给Pod本身的代理地址。可以通过运行以下命令来验证此接口类型：

```
a-apic1# moquery -c ipv4If -x rsp-subtree=children 'rsp-subtree-
filter=eq(ipv4Addr.addr,"10.0.0.34")' rsp-subtree-include=required

...
# ipv4.If
mode         : anycast-v4,external

# ipv4.Addr
 addr             : 10.0.0.34/32
 dn               : topology/pod-1/node-1002/sys/ipv4/inst/dom-overlay-1/if-[lo9]/addr-
[10.0.0.34/32]
```

“external”标志表示这是外部代理TEP。

5.检验主干上的BGP EVPN

应从主干上的BGP EVPN导入合作终端记录。以下命令可用于验证它是否在EVPN中（如果它已经与远程Pod外部代理TEP的下一跳在COOP中，则可以假设它来自EVPN）：

```
a-spine4# show bgp l2vpn evpn 10.0.1.100 vrf overlay-1
Route Distinguisher: 1:16777199
BGP routing table entry for [2]:[0]:[15728642]:[48]:[0000.1111.1111]:[32]:[10.0.1.100]/272,
version 689242 dest ptr 0xaf42a4ca
Paths: (2 available, best #2)
Flags: (0x000202 00000000) on xmit-list, is not in rib/evpn, is not in HW, is locked
Multipath: eBGP iBGP

 Path type: internal 0x40000018 0x2040 ref 0 adv path ref 0, path is valid, not best reason:
Router Id, remote nh not installed
 AS-Path: NONE, path sourced internal to AS
  192.168.1.254 (metric 7) from 192.168.1.102 (192.168.1.102)
    Origin IGP, MED not set, localpref 100, weight 0
    Received label 15728642 2392068
    Received path-id 1
    Extcommunity:
        RT:5:16
        SOO:1:1
        ENCAP:8
        Router MAC:0200.0000.0000

          Advertised path-id 1
 Path type: internal 0x40000018 0x2040 ref 1 adv path ref 1, path is valid, is best path, remote
nh not installed
 AS-Path: NONE, path sourced internal to AS
  192.168.1.254 (metric 7) from 192.168.1.101 (192.168.1.101)
    Origin IGP, MED not set, localpref 100, weight 0
    Received label 15728642 2392068
    Received path-id 1
    Extcommunity:
        RT:5:16
        SOO:1:1
        ENCAP:8
        Router MAC:0200.0000.0000

          Path-id 1 not advertised to any peer
```

请注意，上述命令也可以针对MAC地址运行。

-192.168.1.254是在多Pod设置期间配置的数据平面TEP。但请注意，即使在BGP中将其通告为NH，实际的下一跳将是外部代理TEP。

-192.168.1.101和。102是通告此路径的Pod 1主干节点。

6.验证目标Pod中主干上的COOP。

可以使用与前面相同的命令：

```
a-spine2# show coop internal info ip-db | grep -B 2 -A 15 "10.0.1.100"

------------------------------
IP address : 10.0.1.100
Vrf : 2392068
Flags : 0
EP bd vnid : 15728642
EP mac :  00:50:56:81:3E:E6
Publisher Id : 10.0.72.67
Record timestamp : 10 01 2019 15:46:24 502206158
Publish timestamp : 10 01 2019 15:46:24 524378376
Seq No: 0
Remote publish timestamp: 12 31 1969 19:00:00 0
URIB Tunnel Info
Num tunnels : 1
      Tunnel address : 10.0.72.67
      Tunnel ref count : 1

------------------------------
```

通过在APIC上运行以下命令验证隧道地址的所有者：

```
a-apic1# moquery -c ipv4Addr -f 'ipv4.Addr.addr=="10.0.72.67"'
Total Objects shown: 1

# ipv4.Addr
addr             : 10.0.72.67/32
childAction      :
ctrl             :
dn               : topology/pod-1/node-101/sys/ipv4/inst/dom-overlay-1/if-[lo0]/addr-
[10.0.72.67/32]
ipv4CfgFailedBmp :
ipv4CfgFailedTs  : 00:00:00:00.000
ipv4CfgState     : 0
lcOwn            : local
modTs            : 2019-09-30T18:42:43.262-04:00
monPolDn         : uni/fabric/monfab-default
operSt           : up
operStQual       : up
pref             : 0
rn               : addr-[10.0.72.67/32]
status           :
tag              : 0
type             : primary
vpcPeer          : 0.0.0.0
```

上述命令显示从COOP到枝叶101的隧道。这意味着枝叶101应该具有目标终端的本地学习。

7.验证出口枝叶具有本地学习。

这可以通过“show endpoint”命令完成：

```
a-leaf101# show endpoint ip 10.0.1.100 detail
Legend:
s - arp              H - vtep             V - vpc-attached     p - peer-aged
R - peer-attached-rl B - bounce           S - static           M - span
D - bounce-to-proxy  O - peer-attached    a - local-aged       m - svc-mgr
L - local            E - shared-service
+------------------------------------+--------------+-----------------+-------------+----------
---+----------------------------+
    VLAN/                           Encap           MAC Address       MAC Info/
Interface     Endpoint Group
    Domain                          VLAN            IP Address        IP
Info                       Info
+------------------------------------+--------------+-----------------+-------------+----------
---+----------------------------+
341                                     vlan-1075    0000.1111.1111 LV
po5                  Prod:ap1:epg1
Prod:Vrf1                               vlan-1075        10.0.1.100 LV
po5
```

请注意，终端已获知。应基于已设置VLAN标记1075的out port-channel 5转发数据包。

使用fTriage验证端到端流量

如本章“工具”部分所述，fTriage可用于映射现有端到端流量，并了解路径中的每台交换机对数据包执行的操作。这在大型和更复杂的部署（如多可配置设备）中尤其有用。

请注意，fTriage需要一些时间才能完全运行（可能需要15分钟）。

对示例流运行fTriage时：

```
a-apic1# ftriage route -ii LEAF:205 -dip 10.0.1.100 -sip 10.0.2.100
fTriage Status: {"dbgFtriage": {"attributes": {"operState": "InProgress", "pid": "7297",
"apicId": "1", "id": "0"}}}
Starting ftriage
Log file name for the current run is: ftlog_2019-10-01-16-04-15-438.txt
2019-10-01 16:04:15,442 INFO     /controller/bin/ftriage route -ii LEAF:205 -dip 10.0.1.100 -sip
10.0.2.100
2019-10-01 16:04:38,883 INFO     ftriage:     main:1165 Invoking ftriage with default password
and default username: apic#fallback\\admin
2019-10-01 16:04:54,678 INFO     ftriage:     main:839  L3 packet Seen on a-leaf205 Ingress:
Eth1/31  Egress: Eth1/53 Vnid: 2392068
2019-10-01 16:04:54,896 INFO     ftriage:     main:242  ingress encap string vlan-1021
2019-10-01 16:04:54,899 INFO     ftriage:     main:271  Building ingress BD(s), Ctx
2019-10-01 16:04:56,778 INFO     ftriage:     main:294  Ingress BD(s) Prod:Bd2
2019-10-01 16:04:56,778 INFO     ftriage:     main:301  Ingress Ctx: Prod:Vrf1
2019-10-01 16:04:56,887 INFO     ftriage:   pktrec:490  a-leaf205: Collecting transient losses
snapshot for LC module: 1
2019-10-01 16:05:22,458 INFO     ftriage:     main:933  SIP 10.0.2.100 DIP 10.0.1.100
2019-10-01 16:05:22,459 INFO     ftriage:  unicast:973  a-leaf205: <- is ingress node
2019-10-01 16:05:25,206 INFO     ftriage:  unicast:1215 a-leaf205: Dst EP is remote
2019-10-01 16:05:26,758 INFO     ftriage:     misc:657  a-leaf205: DMAC(00:22:BD:F8:19:FF) same
as RMAC(00:22:BD:F8:19:FF)
2019-10-01 16:05:26,758 INFO     ftriage:     misc:659  a-leaf205: L3 packet getting
routed/bounced in SUG
2019-10-01 16:05:27,030 INFO     ftriage:     misc:657  a-leaf205: Dst IP is present in SUG L3
tbl
2019-10-01 16:05:27,473 INFO     ftriage:     misc:657  a-leaf205: RwDMAC DIPo(10.0.72.67) is
one of dst TEPs ['10.0.72.67']
2019-10-01 16:06:25,200 INFO     ftriage:     main:622  Found peer-node a-spine3 and IF: Eth1/31
in candidate list
2019-10-01 16:06:30,802 INFO     ftriage:     node:643  a-spine3: Extracted Internal-port GPD
```

```
Info for lc: 1
2019-10-01 16:06:30,803 INFO     ftriage:     fcls:4414 a-spine3: LC trigger ELAM with IFS:
Eth1/31 Asic :3 Slice: 1 Srcid: 24
2019-10-01 16:07:05,717 INFO     ftriage:     main:839  L3 packet Seen on a-spine3 Ingress:
Eth1/31 Egress: LC-1/3 FC-24/0 Port-1 Vnid: 2392068
2019-10-01 16:07:05,718 INFO     ftriage:   pktrec:490  a-spine3: Collecting transient losses
snapshot for LC module: 1
2019-10-01 16:07:28,043 INFO     ftriage:      fib:332  a-spine3: Transit in spine
2019-10-01 16:07:35,902 INFO     ftriage:  unicast:1252 a-spine3: Enter dbg_sub_nexthop with
Transit inst: ig infra: False glbs.dipo: 10.0.72.67
2019-10-01 16:07:36,018 INFO     ftriage:  unicast:1417 a-spine3: EP is known in COOP (DIPo =
10.0.72.67)
2019-10-01 16:07:40,422 INFO     ftriage:  unicast:1458 a-spine3: Infra route 10.0.72.67 present
in RIB
2019-10-01 16:07:40,423 INFO     ftriage:     node:1331 a-spine3: Mapped LC interface: LC-1/3
FC-24/0 Port-1 to FC interface: FC-24/0 LC-1/3 Port-1
2019-10-01 16:07:46,059 INFO     ftriage:     node:460  a-spine3: Extracted GPD Info for fc: 24
2019-10-01 16:07:46,060 INFO     ftriage:     fcls:5748 a-spine3: FC trigger ELAM with IFS: FC-
24/0 LC-1/3 Port-1 Asic :0 Slice: 1 Srcid: 40
2019-10-01 16:08:06,735 INFO     ftriage:  unicast:1774 L3 packet Seen on FC of node: a-spine3
with Ingress: FC-24/0 LC-1/3 Port-1  Egress: FC-24/0 LC-1/3 Port-1 Vnid: 2392068
2019-10-01 16:08:06,735 INFO     ftriage:   pktrec:487  a-spine3: Collecting transient losses
snapshot for FC module: 24
2019-10-01 16:08:09,123 INFO     ftriage:     node:1339 a-spine3: Mapped FC interface: FC-24/0
LC-1/3 Port-1 to LC interface: LC-1/3 FC-24/0 Port-1
2019-10-01 16:08:09,124 INFO     ftriage:  unicast:1474 a-spine3: Capturing Spine  Transit pkt-
type L3 packet on egress LC on Node: a-spine3 IFS: LC-1/3 FC-24/0 Port-1
2019-10-01 16:08:09,594 INFO     ftriage:     fcls:4414 a-spine3: LC trigger ELAM with IFS: LC-
1/3 FC-24/0 Port-1 Asic :3 Slice: 1 Srcid: 48
2019-10-01 16:08:44,447 INFO     ftriage:  unicast:1510 a-spine3: L3 packet Spine egress
Transit pkt Seen on a-spine3 Ingress: LC-1/3 FC-24/0 Port-1 Egress: Eth1/29 Vnid: 2392068
2019-10-01 16:08:44,448 INFO     ftriage:   pktrec:490  a-spine3: Collecting transient losses
snapshot for LC module: 1
2019-10-01 16:08:46,691 INFO     ftriage:  unicast:1681 a-spine3: Packet is exiting the fabric
through {a-spine3: ['Eth1/29']} Dipo 10.0.72.67 and filter SIP 10.0.2.100 DIP 10.0.1.100
2019-10-01 16:10:19,947 INFO     ftriage:     main:716  Capturing L3 packet Fex: False on node:
a-spine1 IF: Eth2/25
2019-10-01 16:10:25,752 INFO     ftriage:     node:643  a-spine1: Extracted Internal-port GPD
Info for lc: 2
2019-10-01 16:10:25,754 INFO     ftriage:     fcls:4414 a-spine1: LC trigger ELAM with IFS:
Eth2/25 Asic :3 Slice: 0 Srcid: 24
2019-10-01 16:10:51,164 INFO     ftriage:     main:716  Capturing L3 packet Fex: False on node:
a-spine2 IF: Eth1/31
2019-10-01 16:11:09,690 INFO     ftriage:     main:839  L3 packet Seen on a-spine2 Ingress:
Eth1/31 Egress: Eth1/25 Vnid: 2392068
2019-10-01 16:11:09,690 INFO     ftriage:   pktrec:490  a-spine2: Collecting transient losses
snapshot for LC module: 1
2019-10-01 16:11:24,882 INFO     ftriage:      fib:332  a-spine2: Transit in spine
2019-10-01 16:11:32,598 INFO     ftriage:  unicast:1252 a-spine2: Enter dbg_sub_nexthop with
Transit inst: ig infra: False glbs.dipo: 10.0.72.67
2019-10-01 16:11:32,714 INFO     ftriage:  unicast:1417 a-spine2: EP is known in COOP (DIPo =
10.0.72.67)
2019-10-01 16:11:36,901 INFO     ftriage:  unicast:1458 a-spine2: Infra route 10.0.72.67 present
in RIB
2019-10-01 16:11:47,106 INFO     ftriage:     main:622  Found peer-node a-leaf101 and IF:
Eth1/54 in candidate list
2019-10-01 16:12:09,836 INFO     ftriage:     main:839  L3 packet Seen on a-leaf101 Ingress:
Eth1/54 Egress: Eth1/30 (Po5) Vnid: 11470
2019-10-01 16:12:09,952 INFO     ftriage:   pktrec:490  a-leaf101: Collecting transient losses
snapshot for LC module: 1
2019-10-01 16:12:30,991 INFO     ftriage:     nxos:1404 a-leaf101: nxos matching rule id:4659
scope:84 filter:65534
2019-10-01 16:12:32,327 INFO     ftriage:     main:522  Computed egress encap string vlan-1075
2019-10-01 16:12:32,333 INFO     ftriage:     main:313  Building egress BD(s), Ctx
```

```
2019-10-01 16:12:34,559 INFO     ftriage:     main:331  Egress Ctx Prod:Vrf1
2019-10-01 16:12:34,560 INFO     ftriage:     main:332  Egress BD(s): Prod:Bd1
2019-10-01 16:12:37,704 INFO     ftriage:  unicast:1252 a-leaf101: Enter dbg_sub_nexthop with
Local inst: eg infra: False glbs.dipo: 10.0.72.67
2019-10-01 16:12:37,705 INFO     ftriage:  unicast:1257 a-leaf101: dbg_sub_nexthop invokes
dbg_sub_eg for ptep
2019-10-01 16:12:37,705 INFO     ftriage:  unicast:1784 a-leaf101: <- is egress node
2019-10-01 16:12:37,911 INFO     ftriage:  unicast:1833 a-leaf101: Dst EP is local
2019-10-01 16:12:37,912 INFO     ftriage:     misc:657  a-leaf101: EP if(Po5) same as egr
if(Po5)
2019-10-01 16:12:38,172 INFO     ftriage:     misc:657  a-leaf101: Dst IP is present in SUG L3
tbl
2019-10-01 16:12:38,564 INFO     ftriage:     misc:657  a-leaf101: RW seg_id:11470 in SUG same
as EP segid:11470
fTriage Status: {"dbgFtriage": {"attributes": {"operState": "Idle", "pid": "0", "apicId": "0",
"id": "0"}}}
fTriage Status: {"dbgFtriage": {"attributes": {"operState": "Idle", "pid": "0", "apicId": "0",
"id": "0"}}}
```

fTriage中有大量数据。突出显示了一些最重要的字段。请注意，数据包的路径为“leaf205(Pod 2)> spine3(Pod 2)> spine2(Pod 1)> leaf101(Pod 1)”。 沿途作出的所有转发决策和合同查找也都会显示。

请注意，如果这是第2层流，则需要将fTriage的语法设置为如下内容：

```
ftriage bridge -ii LEAF:205 -dmac 00:00:11:11:22:22
```

EP不在COOP中的代理请求

在考虑特定故障场景之前，还需要讨论有关通过多Pod进行单播转发的问题。如果目标终端未知，请求被代理且终端不在COOP中，会发生什么情况？

在这种情况下，数据包/帧将发送到主干，并生成收集请求。

当主干生成收集请求时，原始数据包仍保留在请求中，但数据包接收ethertype 0xfff2，这是为收集保留的自定义Ethertype。因此，在Wireshark等数据包捕获工具中解释这些消息并不容易。

外部第3层目的地也设置为239.255.255.240，这是专门用于收集消息的保留组播组。这些流量应在交换矩阵中泛洪，并且已部署收集请求的目标子网的所有出口枝叶交换机都将生成ARP请求以解析目标。这些ARP从配置的BD子网IP地址发送（因此，如果在桥接域上禁用了单播路由，则代理请求无法解析无提示/未知端点的位置）。

可以通过以下命令验证在出口枝叶上接收收集消息以及随后生成的ARP和收到的ARP响应：

收集ARP验证

```
a-leaf205# show ip arp internal event-history event | grep -F -B 1 192.168.21.11
...
73) Event:E_DEBUG_DSF, length:127, at 316928 usecs after Wed May  1 08:31:53 2019
Updating epm ifidx: 1a01e000 vlan: 105 ip: 192.168.21.11, ifMode: 128 mac: 8c60.4f02.88fc   <<<
Endpoint is learned
75) Event:E_DEBUG_DSF, length:152, at 316420 usecs after Wed May  1 08:31:53 2019
log_collect_arp_pkt; sip = 192.168.21.11; dip = 192.168.21.254; interface = Vlan104;info = Garp
Check adj:(nil)  <<< Response received
77) Event:E_DEBUG_DSF, length:142, at 131918 usecs after Wed May  1 08:28:36 2019
log_collect_arp_pkt; dip = 192.168.21.11; interface = Vlan104;iod = 138; Info = Internal Request
Done   <<< ARP request is generated by leaf
```

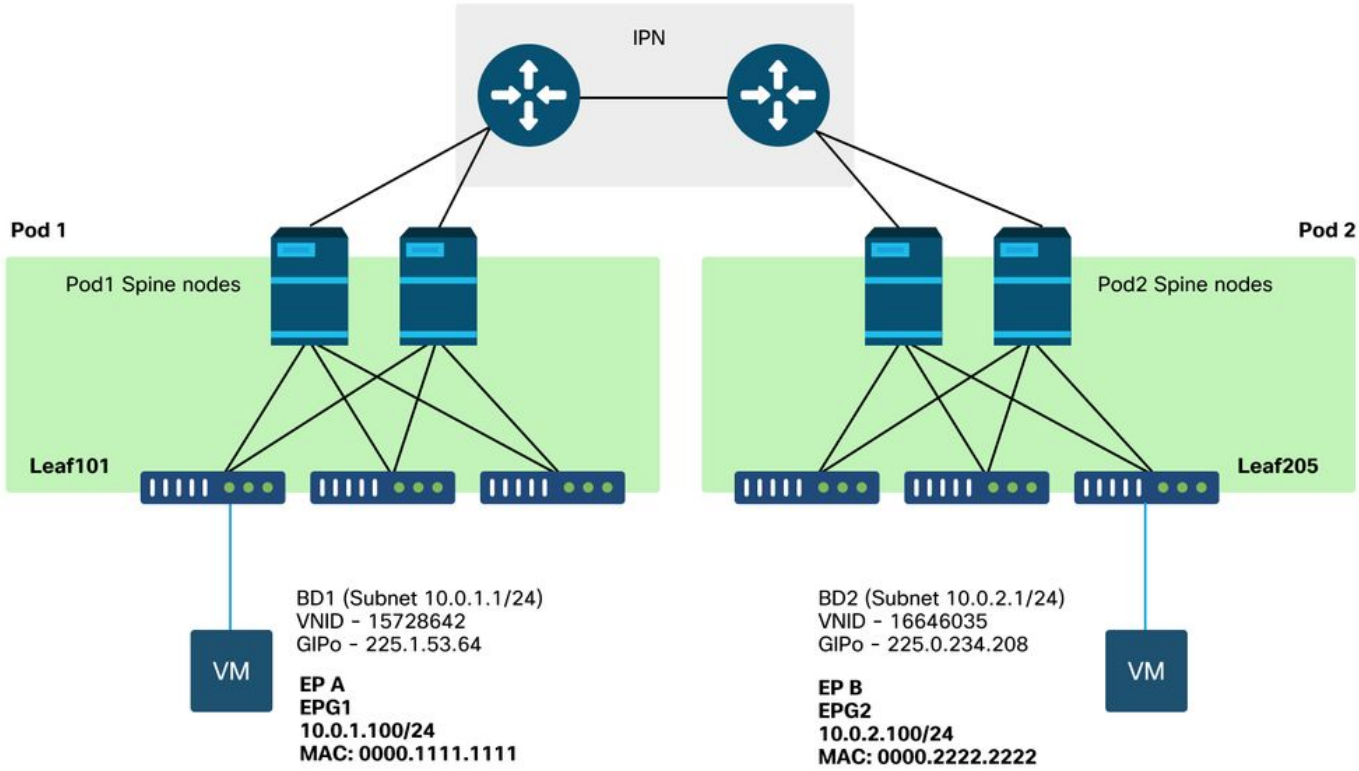
```
78) Event:E_DEBUG_DSF, length:136, at 131757 usecs after Wed May  1 08:28:36 2019   <<< Glean
received, Dst IP is in BD subnet
log_collect_arp_glean;dip = 192.168.21.11;interface = Vlan104;info = Received pkt Fabric-Glean:
1
79) Event:E_DEBUG_DSF, length:174, at 131748 usecs after Wed May  1 08:28:36 2019
log_collect_arp_glean; dip = 192.168.21.11; interface = Vlan104; vrf = CiscoLive2019:vrf1; info
= Address in PSVI subnet or special VIP   <<< Glean Received, Dst IP is in BD subnet
```

作为参考，发送到239.255.255.240的收集消息是此组需要包含在IPN上的双向PIM组范围中的原因。

多Pod故障排除场景#1（单播）

在以下拓扑中，EP B无法与EP A通信。

拓扑故障排除



请注意，在多Pod转发中出现的许多问题与单个Pod中发现的问题相同。因此，需要重点解决多Pod的特定问题。

遵循前面介绍的单播故障排除工作流程时，请注意请求是代理的，但Pod 2中的主干节点在COOP中没有目标IP。

原因：COOP中缺少终端

如前所述，系统会根据BGP EVPN信息填充远程Pod终端的COOP条目。因此，必须确定：

a.) 源Pod(Pod 2)主干是否包含在EVPN中？

```
a-spine4# show bgp l2vpn evpn 10.0.1.100 vrf overlay-1
<no output>
```

b.) 远程Pod(Pod 1)主干是否包含在EVPN中？

```
a-spine1# show bgp l2vpn evpn 10.0.1.100 vrf overlay-1
Route Distinguisher: 1:16777199    (L2VNI 1)
BGP routing table entry for [2]:[0]:[15728642]:[48]:[0050.5681.3ee6]:[32]:[10.0.1.100]/272,
version 11751 dest ptr 0xafbf8192
Paths: (1 available, best #1)
Flags: (0x00010a 00000000) on xmit-list, is not in rib/evpn
Multipath: eBGP iBGP

 Advertised path-id 1
 Path type: local 0x4000008c 0x0 ref 0 adv path ref 1, path is valid, is best path
 AS-Path: NONE, path locally originated
  0.0.0.0 (metric 0) from 0.0.0.0 (192.168.1.101)
    Origin IGP, MED not set, localpref 100, weight 32768
    Received label 15728642 2392068
    Extcommunity:
        RT:5:16

          Path-id 1 advertised to peers:
```

Pod 1主干已安装，下一跳IP为0.0.0.0;这意味着它是从COOP本地导出的。但请注意，“通告到对等体”部分不包括Pod 2主干节点。

c.) BGP EVPN是否在Pod之间启动？

```
a-spine4# show bgp l2vpn evpn summ vrf overlay-1

Neighbor            V      AS   MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
192.168.1.101   4 65000       57380    66362            0     0      0  00:00:21  Active
192.168.1.102   4 65000       57568    66357            0     0      0  00:00:22  Active
```

请注意，在上面的输出中，Pod之间的BGP EVPN对等连接已关闭。State/PfxRcd列中除数字值之外的任何值均表示邻接关系未启用。Pod 1 EP不通过EVPN学习，并且不会导入到COOP中。

如果发现此问题，请验证以下事项：

1. OSPF是否在主干节点和连接的IPN之间启动？
2. 主干节点是否通过OSPF获取远程主干IP的路由？
3. 通过IPN的完整路径是否支持巨型MTU?
4. 所有协议邻接关系是否稳定？

其他可能的原因

如果终端不在任何Pod的COOP数据库中，并且目标设备是静默主机（在交换矩阵中的任何枝叶交换机上未获知），请验证交换矩阵收集过程是否正常工作。要使此功能发挥作用：

- 必须在BD上启用单播路由。
- 目标必须位于BD子网中。
- IPN必须为239.255.255.240组提供组播路由服务。

组播部分将在下一部分详细介绍。

Multi-Pod广播、未知单播和组播(BUM)转发概述

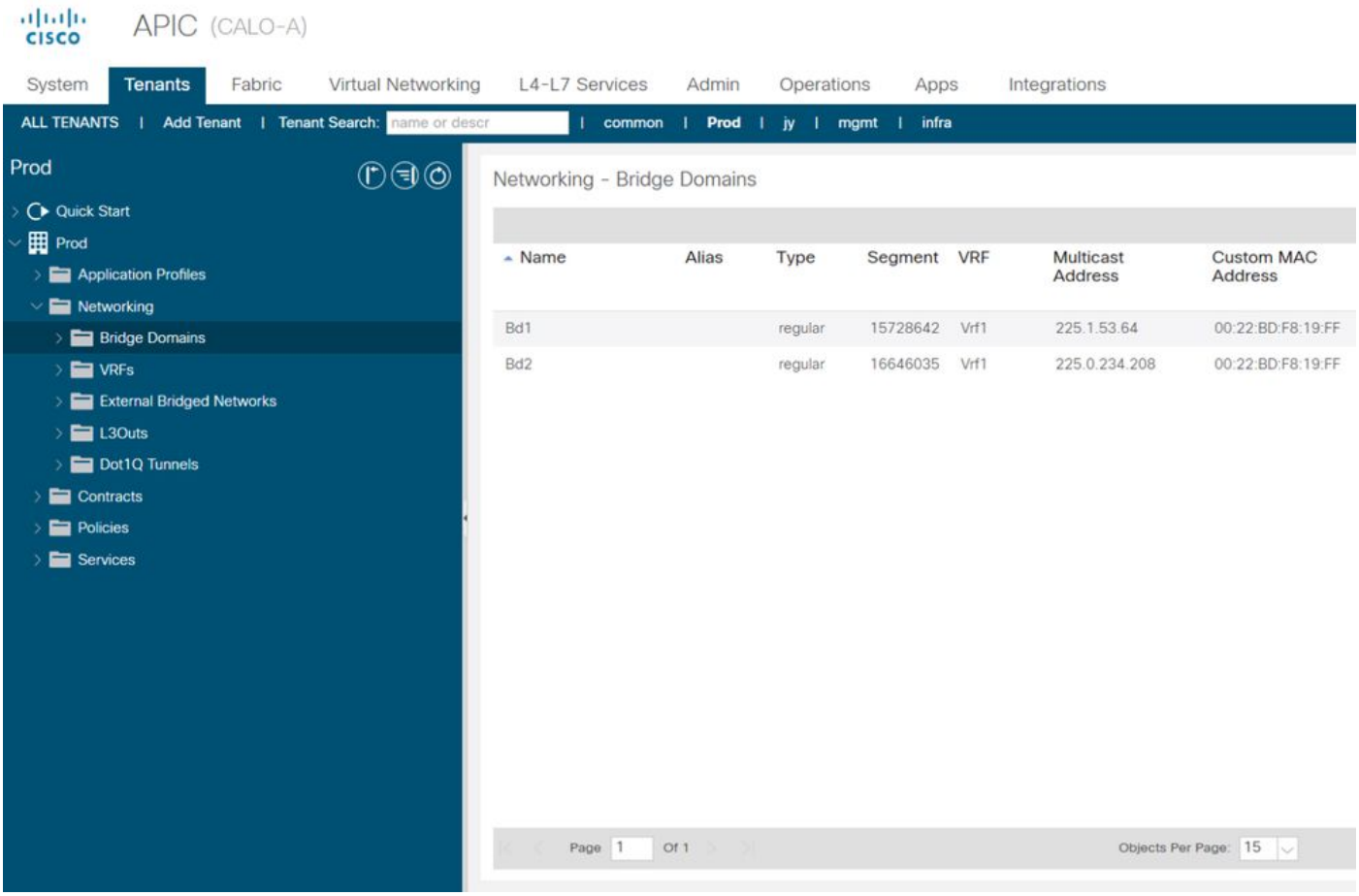
在ACI中，流量在许多不同场景中通过重叠组播组泛洪。例如，发生泛洪：

- 组播和广播流量。
- 必须泛洪的未知单播。
- 交换矩阵ARP收集消息。
- EP通告消息。

许多特性和功能依赖于BUM转发。

在ACI中，所有网桥域都分配了一个组播地址，称为组IP外部（或GIPo）地址。网桥域内必须泛洪的所有流量都泛洪到此GIPo。

GUI中的BD GIPo



可以直接在一个APIC上查询对象。

BD Moquery中的GIPo

```
a-apic1# moquery -c fvBD -f 'fv.BD.name=="Bd1"'
Total Objects shown: 1

# fv.BD
name                        : Bd1
OptimizeWanBandwidth        : no
annotation                  :
```

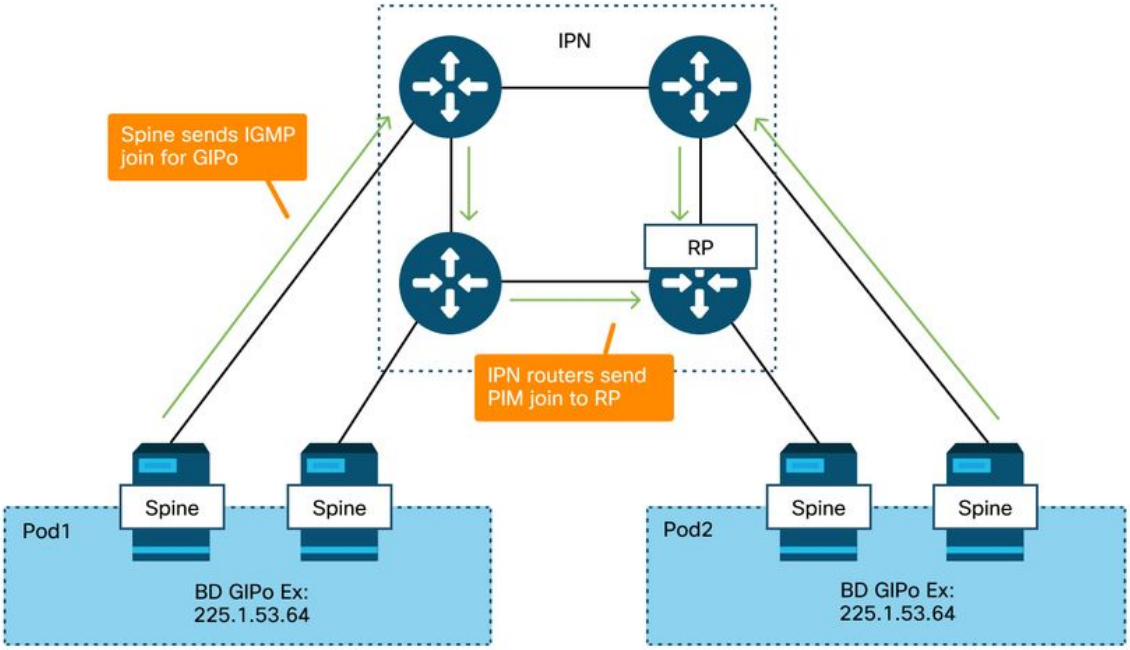
```
arpFlood                : yes
bcastP                  : 225.1.53.64
childAction             :
configIssues            :
descr                   :
dn                      : uni/tn-Prod/BD-Bd1
epClear                 : no
epMoveDetectMode        :
extMngdBy               :
hostBasedRouting        : no
intersiteBumTrafficAllow : no
intersiteL2Stretch      : no
ipLearning              : yes
ipv6McastAllow          : no
lcOwn                   : local
limitIpLearnToSubnets   : yes
llAddr                  : ::
mac                     : 00:22:BD:F8:19:FF
mcastAllow              : no
modTs                   : 2019-09-30T20:12:01.339-04:00
monPolDn                : uni/tn-common/monepg-default
mtu                     : inherit
multiDstPktAct          : bd-flood
nameAlias               :
ownerKey                :
ownerTag                :
pcTag                   : 16387
rn                      : BD-Bd1
scope                   : 2392068
seg                     : 15728642
status                  :
type                    : regular
uid                     : 16011
unicastRoute            : yes
unkMacUcastAct          : proxy
unkMcastAct             : flood
v6unkMcastAct           : flood
vmac                    : not-applicable
```

无论是否使用Multi-Pod，上述关于GIPo泛洪的信息都是正确的。与多Pod相关的这部分内容是IPN上的组播路由。

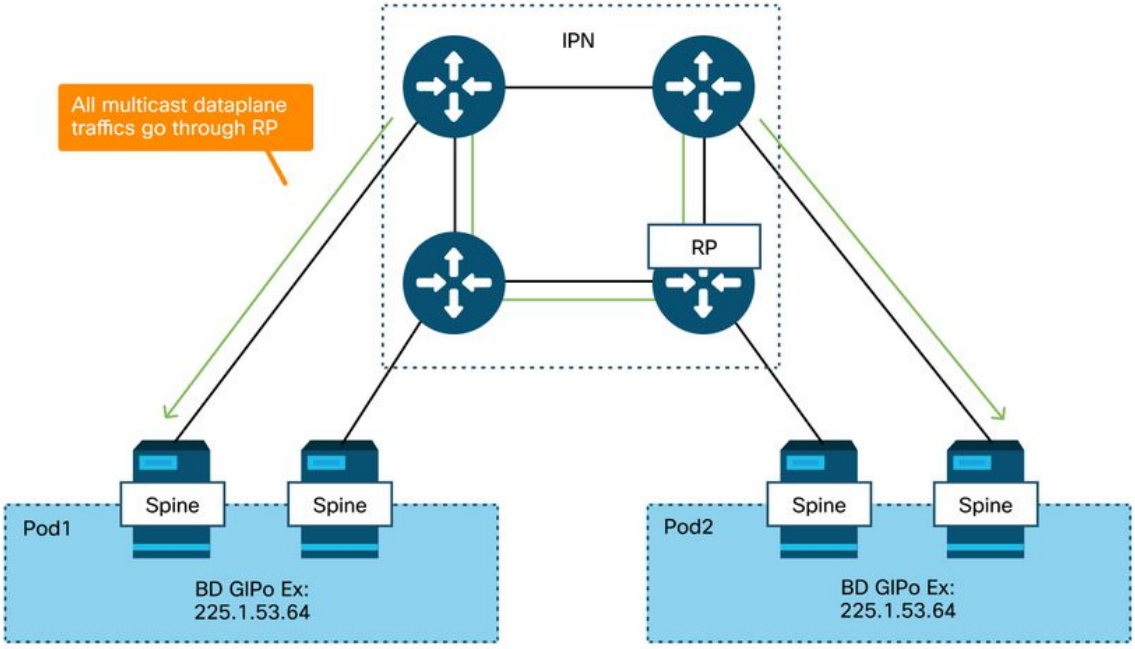
IPN组播路由涉及以下内容：

- 主干节点充当组播主机（仅限IGMP）。 它们不运行PIM。
- 如果BD部署在Pod中，该Pod的一个主干将在一个面向IPN的接口上发送IGMP加入。此功能跨所有主干节点和许多组上的面向IPN的接口进行条带化。
- IPN接收这些加入并向双向PIM RP发送PIM加入。
- 由于使用了PIM Bidir，因此没有(S，G)树。PIM Bidir中只使用(*,G)树。
- 发送到GIPo的所有数据平面流量都通过RP。

IPN组播控制平面



IPN组播数据平面

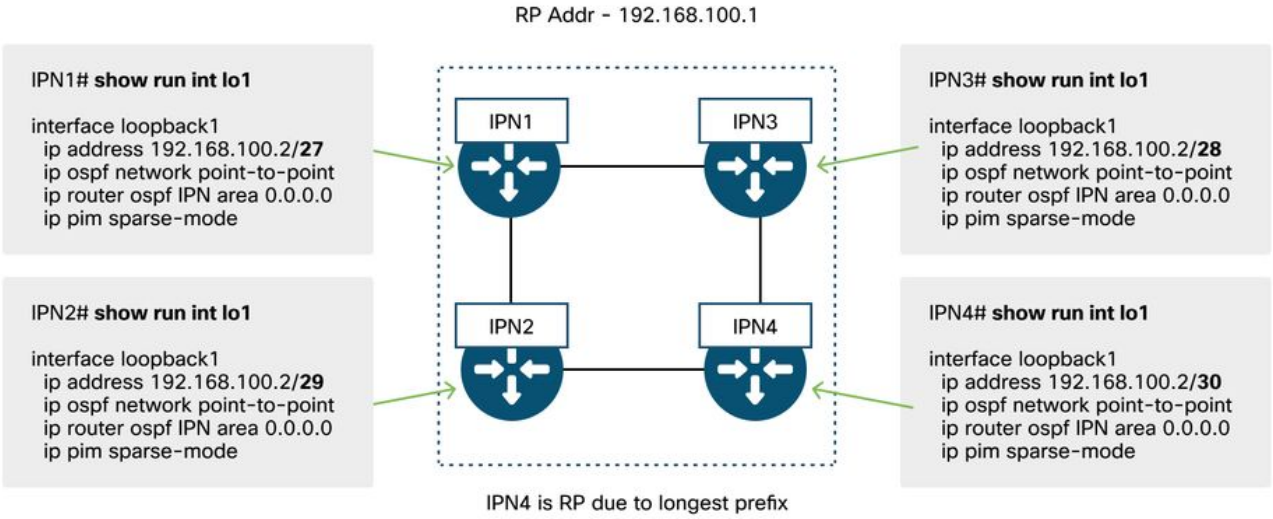


使用PIM Bidir的RP冗余的唯一方法是使用Phantom。本书的“Multi-Pod Discovery”部分对此进行了详细介绍。快速总结一下，请注意，对于虚拟RP:

- 所有IPN必须配置相同的RP地址。

- 任何设备上都不能存在确切的RP地址。
- 多台设备向包含虚拟RP IP地址的子网通告可达性。所通告的子网在子网长度上应有所差异，以便所有路由器就谁通告RP的最佳路径达成一致。如果此路径丢失，则收敛取决于IGP。

虚拟RP配置



Multi-Pod广播、未知单播和组播(BUM)故障排除工作流程

1.首先确认交换矩阵是否真正将流视为多目的地。

在以下常见示例中，流将在BD中泛洪：

- 帧是ARP广播，并且BD上启用了ARP泛洪。
- 该帧将发往组播组。请注意，即使启用了IGMP监听，流量仍会始终泛洪到GIPo上的交换矩阵。
- 流量发往ACI为其提供组播路由服务的组播组。
- 流是第2层（桥接流），目标MAC地址未知，并且BD上的未知单播行为设置为“泛洪”。

确定做出转发决策的最简单方法是使用ELAM。

2.确定BD GIPo。

请参阅本章前面讨论此问题的部分。主干ELAM也可以通过ELAM Assistant应用运行，以验证是否收到了泛洪流量。

3.检验IPN上该GIPo的组播路由表。

执行此操作的输出取决于所使用的IPN平台，但级别较高：

- 所有IPN路由器必须同意此GIP的RP和RPF必须指向此树。
- 连接到每个Pod的一台IPN路由器应该获得该组的IGMP加入。

多Pod故障排除场景#2（BUM流）

此场景将涵盖任何涉及未在多个Pod或BUM场景中解析ARP的场景（未知单播等）。

这里有几个可能的原因。

可能的原因1:多个路由器拥有PIM RP地址

在此场景中，入口枝叶泛洪流量（通过ELAM验证），源Pod接收并泛洪流量，但远程Pod无法接收该流量。对于某些BD来说，泛洪有效，但对于另一些不是。

在IPN上，为GIPo运行“show ip mroute <GIPo address>”以查看RPF树指向多个不同的路由器。

如果是这种情况，请检查以下项：

- 验证实际PIM RP地址是否未在任何位置配置。拥有该实际RP地址的任何设备都会看到其本地/32路由。
- 验证在幻影RP场景中，多个IPN路由器未通告RP的相同前缀长度。

可能的原因2:IPN路由器不学习RP地址的路由

就像第一个可能的原因一样，此处泛洪流量无法离开IPN。每个IPN路由器上的“show ip route <rp address>”输出仅显示本地配置的前缀长度，而不是显示其他路由器正在通告的前缀长度。

其结果是，即使未在任何位置配置实际RP IP地址，每台设备仍将其视为RP。

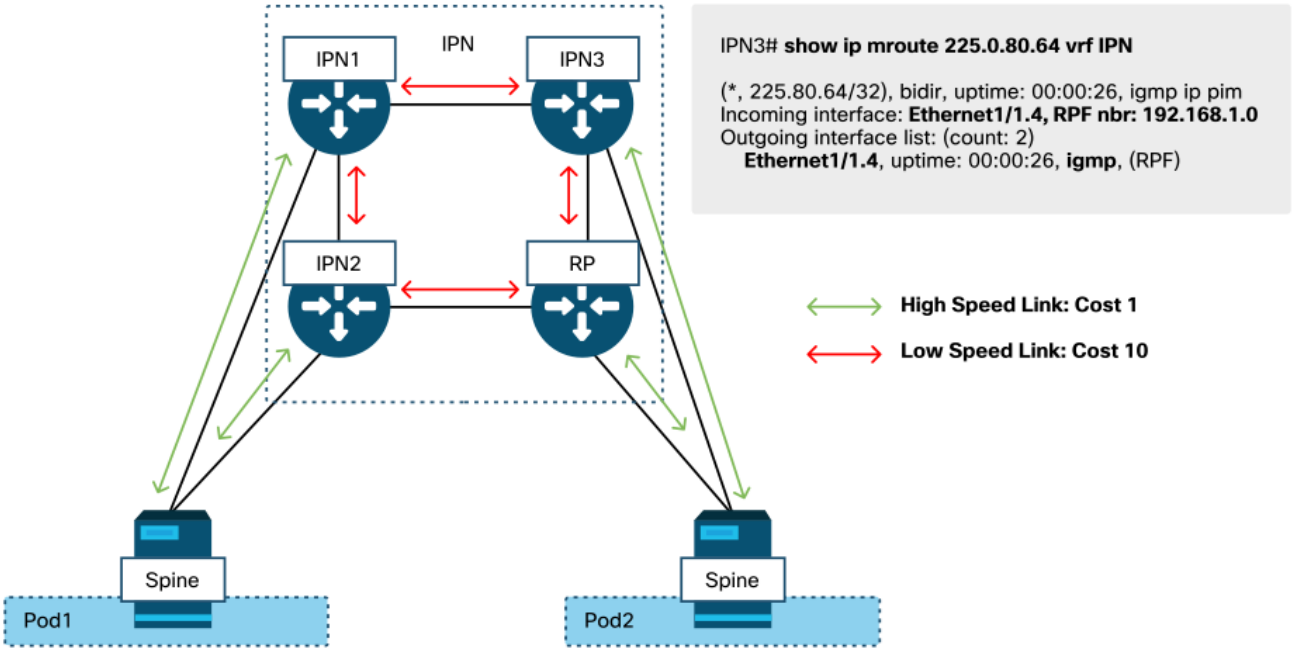
如果真是这样。应检查以下内容：

- 检验IPN路由器之间的路由邻接关系是否已启用。检验该路由是否位于实际协议数据库中（例如OSPF数据库）。
- 验证所有假设为候选RP的环回均配置为OSPF点对点网络类型。如果未配置此网络类型，则无论实际配置的内容如何，每台路由器都会始终通告/32前缀长度。

可能原因3:IPN路由器不安装GIPo路由或RPF指向ACI

如前所述，ACI不会在面向IPN的链路上运行PIM。这意味着IPN通向RP的最佳路径永远不应指向ACI。如果多个IPN路由器连接到同一个主干，则可能出现这种情况，而且通过主干可以发现比直接在IPN路由器之间发现更好的OSPF度量。

面向ACI的RPF接口



要解决此问题：

- 确保IPN路由器之间的路由协议邻接关系已启用。
- 将主干节点上面向IPN的链路的OSPF开销度量增加到使该度量不如IPN到IPN链路优先的值。

其它参考资料

在ACI软件4.0之前，外部设备在使用COS 6时遇到一些挑战。其中大多数问题已通过4.0增强功能解决，但有关详细信息，请参阅CiscoLive会话“BRKACI-2934 — 排除多可配置设备故障”和“服务质量”部分。

关于此翻译
思科采用人工翻译与机器翻译相结合的方式将此文档翻译成不同语言，希望全球的用户都能通过各自的语言得到支持性的内容。

请注意：即使是最好的机器翻译，其准确度也不及专业翻译人员的水平。

Cisco Systems, Inc. 对于翻译的准确性不承担任何责任，并建议您总是参考英文原始文档（已提供链接）。