在Catalyst 9000系列交换机上实施BGP EVPN DHCP第2层中继

目录

简介

本文档介绍如何配置、验证EVPN VxLAN DHCP L2中继功能并对其进行故障排除。

先决条件

要求

- 此功能用于任何使用DHCP的CGW类型部署
- 如果实施受保护分段，请查阅这些文档
  - 在Catalyst 9000系列交换机上实施BGP EVPN路由策略
  - 在Catalyst 9000系列交换机上实施BGP EVPN保护的重叠分段

使用的组件

本文档中的信息基于以下软件和硬件版本：

- Catalyst 9300
- Catalyst 9400
- Catalyst 9500
- Catalyst 9600
- Cisco IOS® XE 17.12.1及更高版本

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。如果您的网络处于活动状态，请确保您了解所有命令的潜在影响。

背景信息

文档详细信息

本文档可用于任何CGW部署，其中DHCP需要从没有SVI的枝叶中继到中央网关。

- 如果未使用受保护的分段，请使用文档中将SVI通告到交换矩阵的部分

如果要实施受保护分段，本文档是3个相互关联的文档的第2部分：

- 文档1：在Catalyst 9000系列交换机上实施BGP EVPN路由策略介绍了如何控制重叠中的BGP BUM流量，必须首先进行配置
- 文档2：在Catalyst 9000系列交换机上实施BGP EVPN保护的重叠分段在文档1的重叠设计和策略的基础上构建，描述“protected”关键字的实施。
- 文档3：本文档。在前两个文档之上构建，描述仅使用第2层枝叶和CGW实施DHCP中继的方式

L2中继行为

中继	监听	核心泛洪	访问泛洪	IPv4
是	是	否	是	• 选项82子选项：(1)代理电路ID (vni-mod-port)使用dhcp监听填充 • 您可以使用dhcp trust配置限制接入端 *推荐的型号
是	否	是	是	• 选项82子选项：(1)代理电路ID (vlan-mod-port)使用dhcp监听填充
否	是	否	是	• 选项82子选项：(1)代理电路ID (vni-mod-port)使用dhcp监听填充 • 您可以使用dhcp trust配置限制接入端
中继	监听	核心泛洪	访问泛洪	IPv6
是	是	是	是	• 选项82子选项：(1)代理电路ID (vni-mod-port)使用dhcp监听填充 • 您可以使用dhcp trust配置限制接入端
是	否	是	是	• 选项82子选项：(1)代理电路ID (vlan-mod-port)使用dhcp监听填充
否	是	是	是	• 选项82子选项：(1)代理电路ID (vni-mod-port)使用dhcp监听填充 • 您可以使用dhcp trust配置限制接入端
否	否	是	是	

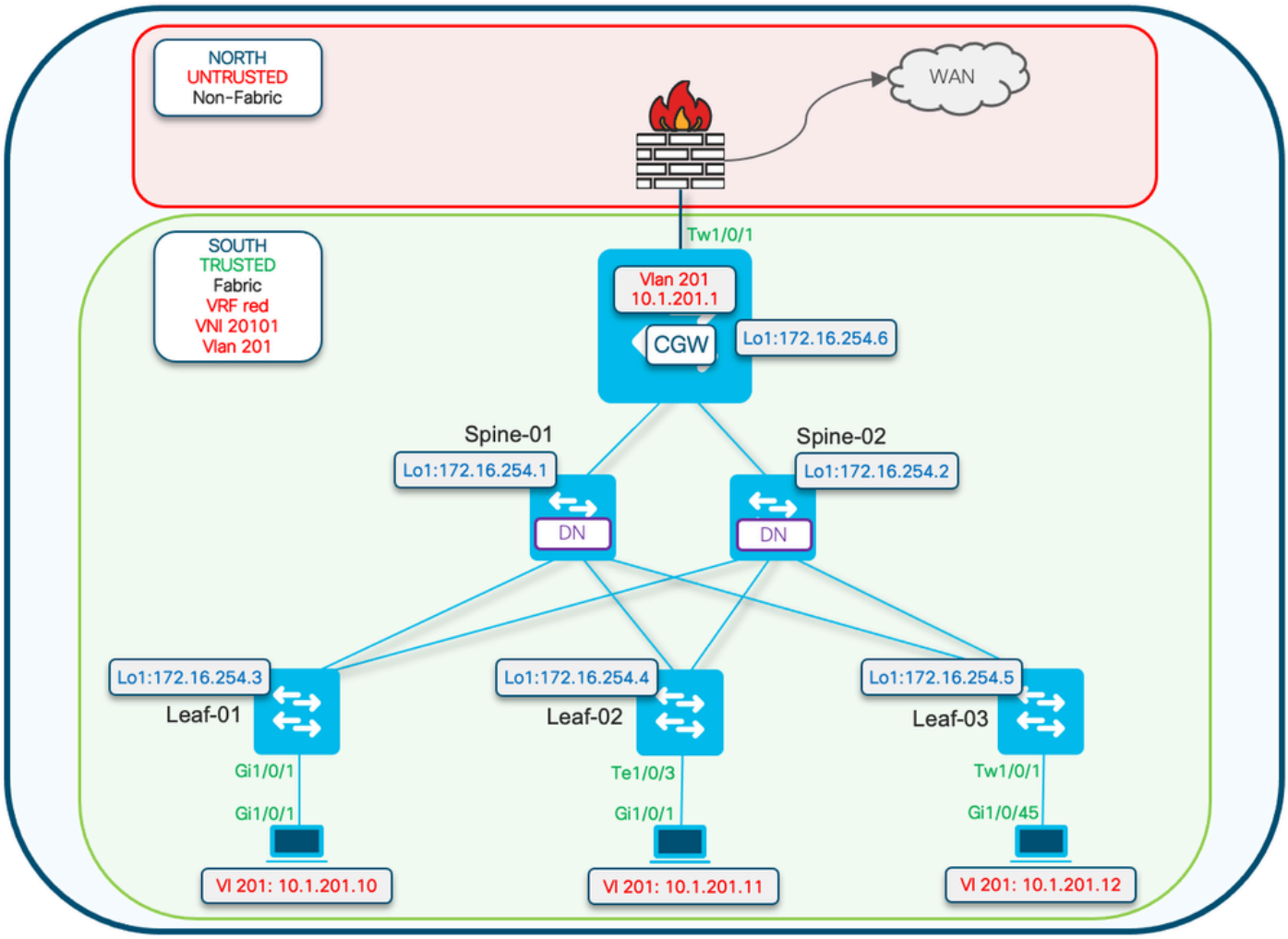
术语

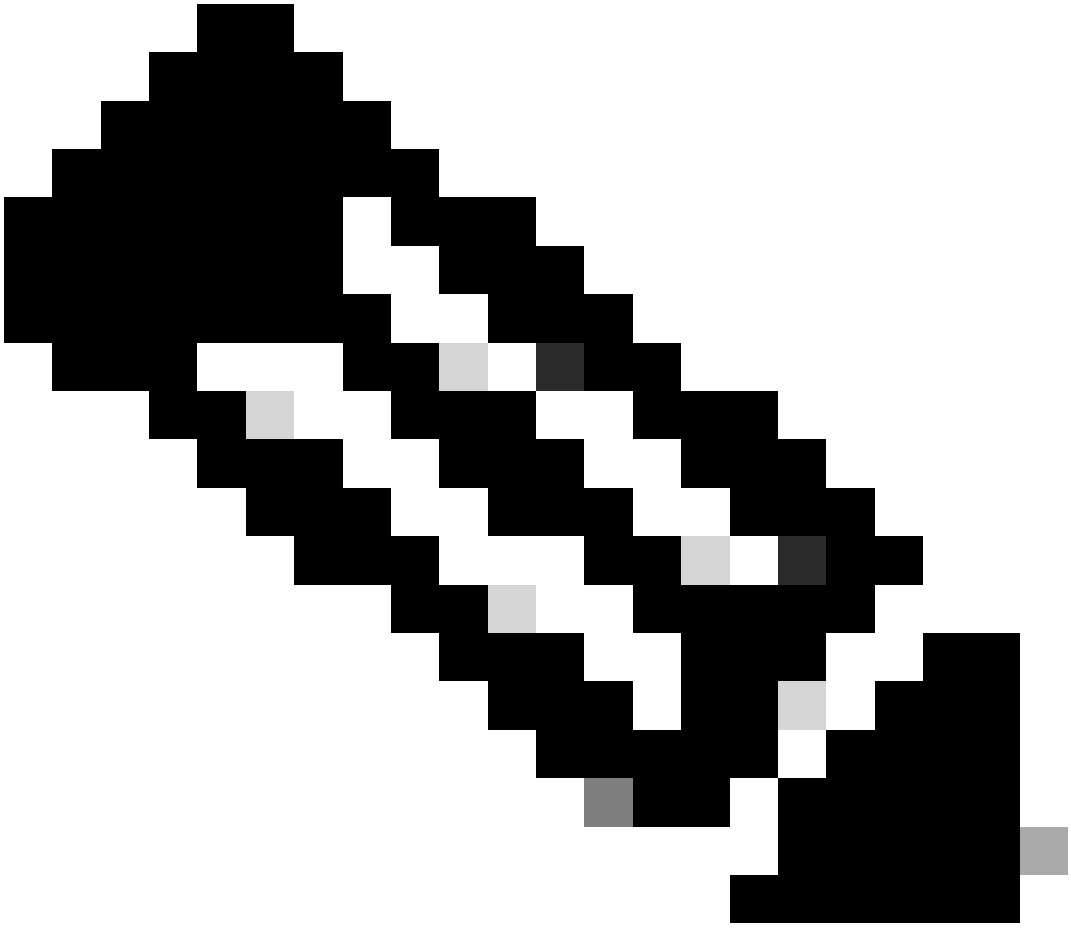
VRF	虚拟路由转发	定义与其他VRF和全局IPv4/IPv6路由域分离的第3层路由域
AF	地址系列	定义BGP处理的前缀类型和路由信息

AS	自治系统	一组属于一个网络或一组网络的可路由IP前缀，全部由单个实体或组织管理、控制和监督
EVPN	以太网虚拟专用网络	允许BGP传输第2层MAC和第3层IP信息的扩展是EVPN，并使用多协议边界网关协议(MP-BGP)作为协议来分发有关VXLAN重叠网络的可达性信息。
VXLAN	虚拟可扩展LAN（局域网）	VXLAN旨在克服VLAN和STP的固有局限性。推荐的IETF标准[RFC 7348]与VLAN提供相同的以太网第2层网络服务，但灵活性更高。从功能上讲，它是MAC-in-UDP封装协议，在第3层底层网络上作为虚拟重叠运行。
CGW	集中式网关	以及网关SVI不在每个枝叶上的EVPN的实施。相反，所有路由都由使用非对称IRB（集成路由和桥接）的特定枝叶完成
DEF网关	默认网关	通过“l2vpn evpn”配置部分下的“default-gateway advertise enable”命令添加到MAC/IP前缀的BGP扩展社区属性。
IMET (RT3)	包括组播以太网标记（路由）	也称为BGP类型3路由。此路由类型在EVPN中用于在VTEP之间传输BUM（广播/未知单播/组播）流量。
RT2	路由类型2	BGP MAC或MAC/IP前缀，表示主机MAC或网关MAC-IP
EVPN经理	EVPN管理器	用于各种其他组件的中央管理组件（例如：从SISF获知并向L2RIB发送信号）
SISF	交换机集成安全功能	EVPN使用的不可知主机跟踪表，用于了解枝叶上的本地主机
L2RIB	第2层路由信息库	在用于管理BGP、EVPN管理器、L2FIB之间交互的中间组件中
FED	转发引擎驱动程序	对ASIC（硬件）层进行编程
MATM	Mac地址表管理器	IOS MATM：仅安装本地地址和 FED MATM：硬件表，安装从控制平面获知的本地和远程地址，是硬件转发平面的一部分

配置（标准CGW部署）

网络图





注意：本部分介绍不使用受保护功能的标准CGW部署。

- 显示DHCP DORA数据包交换的调试仅在受保护网段示例中显示

L2 VTEP（枝叶）密钥详细信息

请求数据包来自客户端

- 使用Default gw advertised CGW mac。
- 如果存在多个gw，则使用第一个gw mac。
- 将外部广播MAC（客户端发起：DORA中的D和R）转换为单播GW MAC并转发到CGW

DHCP监听添加：选项82子选项：电路和RID

（RID由CGW上的响应数据包处理使用）。

(通知CGW其非本地和交换矩阵中继返回L2VTEP)

<#root>

```
Option: (82) Agent Information Option
        Length: 24
        Option 82 Suboption: (1) Agent Circuit ID
            Length: 12
            Agent Circuit ID: 010a000800000277501010000

        Option 82 Suboption: (2) Agent Remote ID

            Length: 8
          Agent Remote ID:
          000

682c7bf88700 <-- switch base mac 682c.7bf8.8700 (from 'show switch')
```

- 通过vxlan隧道从CGW接收的响应数据包。
- Leaf Strips选项82。
- 添加带有客户端源接口的绑定条目。（vxlan-mod-port提供客户端源接口）。
- 转发到客户端的响应数据包。

L3 VTEP (CGW)密钥详细信息

- 启用DHCP监听
- 在SVI中启用DHCP中继
- 从L2VTEP接收请求，并将其提供给中继。
- 中继添加其他选项82子选项（gi、服务器覆盖等）并发送到DHCP服务器。
- 来自dhcp服务器的DHCP响应首先进入RELAY组件。
- 在RELAY删除选项82参数（gi地址、服务器覆盖等）后，数据包将传递到dhcp监听组件。
- 监听组件检查RID（路由器ID），如果它不是本地路由器，则不会删除选项82子选项1和2。
- 交换矩阵中继（因为RID不是本地的）数据包直接转发到远程客户端。
- 使用客户端Mac并执行网桥插入。 硬件执行客户端mac查找，并将具有vxlan封装的数据包转发到始发L2VTEP。

L2VTEP

配置evpn实例

<#root>

```
Leaf-01#
```

show run | beg l2vpn evpn instance 201

```
l2vpn evpn instance 201 vlan-based
 encapsulation vxlan
 replication-type ingress
```

启用DHCP监听

<#root>

```
Leaf-01#
```

show run | sec dhcp snoop

```
ip dhcp snooping vlan 101,
```

201

```
ip dhcp snooping
```

CGW

配置evpn实例

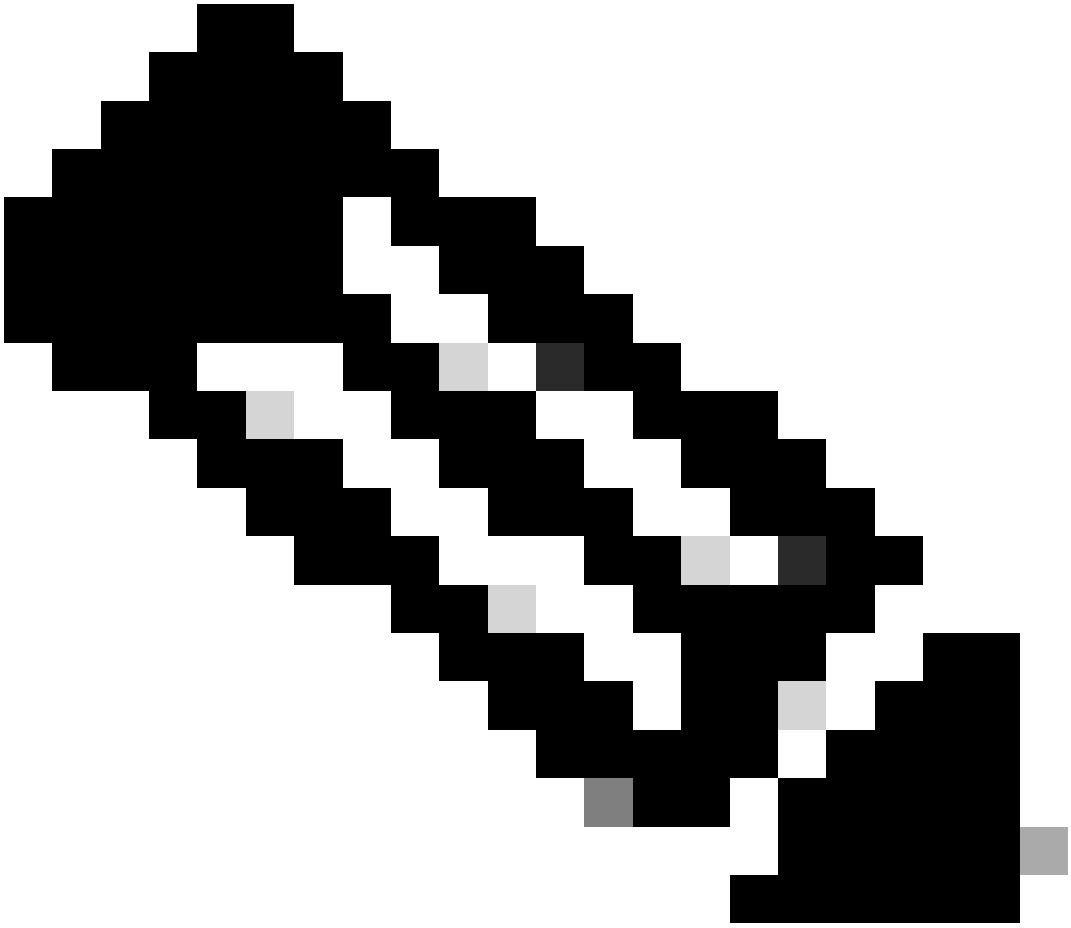
<#root>

```
Border#
```

sh run | s l2vpn evpn instance 201

```
l2vpn evpn instance 201 vlan-based
 encapsulation vxlan
 replication-type ingress
```

default-gateway advertise enable <-- Enable to add BGP DEF GW ext. community attribute



注意：DEF GW属性对于L2中继了解要将DHCP数据包封装和发送到谁至关重要。

启用DHCP监听

<#root>

```
Border#
sh run | s dhcp snoop

ip dhcp snooping vlan 101,
201

ip dhcp snooping
```

确保DHCP中继的配置正确以处理其他选项

<#root>

Border#

sh run int vl 201

Building configuration...

interface Vlan201
 mac-address 0000.beef.cafe
 vrf forwarding red

 ip dhcp relay information option vpn-id <-- Ensure the vrf info is passed to the server

ip dhcp relay source-interface Loopback0 <-- Sets the relay source interface to the loopback

 ip address 10.1.201.1 255.255.255.0

ip helper-address global 10.1.33.33   <-- In this scenario the DHCP server is in the global routing t

验证（标准CGW部署）

网关前缀（枝叶）

<#root>

Leaf-01#

sh bgp l2vpn evpn route-type 2 0 0000.beef.cafe 10.1.201.1

BGP routing table entry for [2][172.16.255.3:201][0][48][0000BEEFCAFE][32][10.1.201.1]/24, version 8964
Paths: (1 available, best #1,

table evi_201

)

<-- In the EVI context for the segment

Not advertised to any peer
Refresh Epoch 3
Local, imported path from [2][172.16.255.6:201][0][48][0000BEEFCAFE][32][10.1.201.1]/24 (global)
  172.16.255.6 (metric 30) (via default) from 172.16.255.1 (172.16.255.1)
    Origin incomplete, metric 0, localpref 100, valid, internal, best
    EVPN ESI: 00000000000000000000,

Label1 20101   <-- Correct segment ID

    Extended Community: RT:65001:201 ENCAP:8

EVPN DEF GW:0:0 <-- GW attribute added indicating this is GW prefix which L2 Relay uses

Originator: 172.16.255.6

, Cluster list: 172.16.255.1

<-- Learned from the Border (CGW)

    rx pathid: 0, tx pathid: 0x0
    Updated on Nov 14 2023 16:06:40 UTC

FED MATM (枝叶)

<#root>

Leaf-01#

show platform software fed switch active matm macTable vlan 201

VLAN MAC Type Seq# EC_Bi Flags machandle siHandle riHandl

---

201 0006.f601.cd42 0x1 32436 0 0 0x71e058dc3368 0x71e058655018 0x0

201 0006.f601.cd01 0x1 32437 0 0 0x71e058dae308 0x71e058655018 0x0

201 0000.beef.cafe 0x5000001

0 0 64 0x71e059177138 0x71e058eeb418 0x71e058df81f8 0x0

VTEP 172.16.255.6 adj_id 1371

No

<--- The GW MAC shows learnt via the Border Leaf Loopback with the right flags

Total Mac number of addresses:: 3
Summary:
Total number of secure addresses:: 0
Total number of drop addresses:: 0
Total number of lisp local addresses:: 0

Total number of lisp remote addresses:: 1 <---

*a_time=aging_time(secs) *e_time=total_elapsed_time(secs)
Type:

MAT_DYNAMIC_ADDR 0x1

MAT_STATIC_ADDR 0x2 MAT_CPU_ADDR 0x4 MAT_DISCARD_ADDR 0x8
MAT_ALL_VLANS 0x10 MAT_NO_FORWARD 0x20 MAT_IPMULT_ADDR 0x40 MAT_RES

```
MAT_DO_NOT_AGE           0x100  MAT_SECURE_ADDR          0x200  MAT_NO_PORT              0x400  MAT_DRO
MAT_DUP_ADDR            0x1000  MAT_NULL_DESTINATION    0x2000  MAT_DOT1X_ADDR          0x4000  MAT_ROU
MAT_WIRELESS_ADDR      0x10000  MAT_SECURE_CFG_ADDR    0x20000  MAT_OPQ_DATA_PRESENT   0x40000  MAT_WIR
MAT_DLR_ADDR          0x100000  MAT_MRP_ADDR          0x200000  MAT_MSRP_ADDR         0x400000  MAT_LIS

MAT_LISP_REMOTE_ADDR 0x1000000

  MAT_VPLS_ADDR        0x2000000

MAT_LISP_GW_ADDR     0x4000000                                  <-- these 3 values added = 0x5000001 (not
```

本地MAC（枝叶）

<#root>

```
Leaf-01#

show switch


Switch/Stack Mac Address : 682c.7bf8.8700 - Local Mac Address
Mac persistency wait time: Indefinite
                                             H/W   Current
Switch#   Role    Mac Address     Priority Version  State
------------------------------------------------------------------------------------
*1       Active

682c.7bf8.8700

     1      V01     Ready

<--- Use to validate the Agent ID in DHCP Option 82
```

DHCP监听（枝叶和CGW）

<#root>

```
Leaf-01#

show ip dhcp snooping


Switch DHCP snooping is enabled


Switch DHCP gleaning is disabled
DHCP snooping is configured on following VLANs:
101,201

DHCP snooping is operational on following VLANs:


101,201
```

```
Insertion of option 82 is enabled
circuit-id default format: vlan-mod-port
remote-id: 682c.7bf8.8700 (MAC)          <--- Leaf-01 adds the switch MAC to Option 82 to indicate to CG
```

```
CGW#
```

```
show ip dhcp snooping
```

```
Switch DHCP snooping is enabled
```

```
Switch DHCP gleaning is disabled
DHCP snooping is configured on following VLANs:
101,201
```

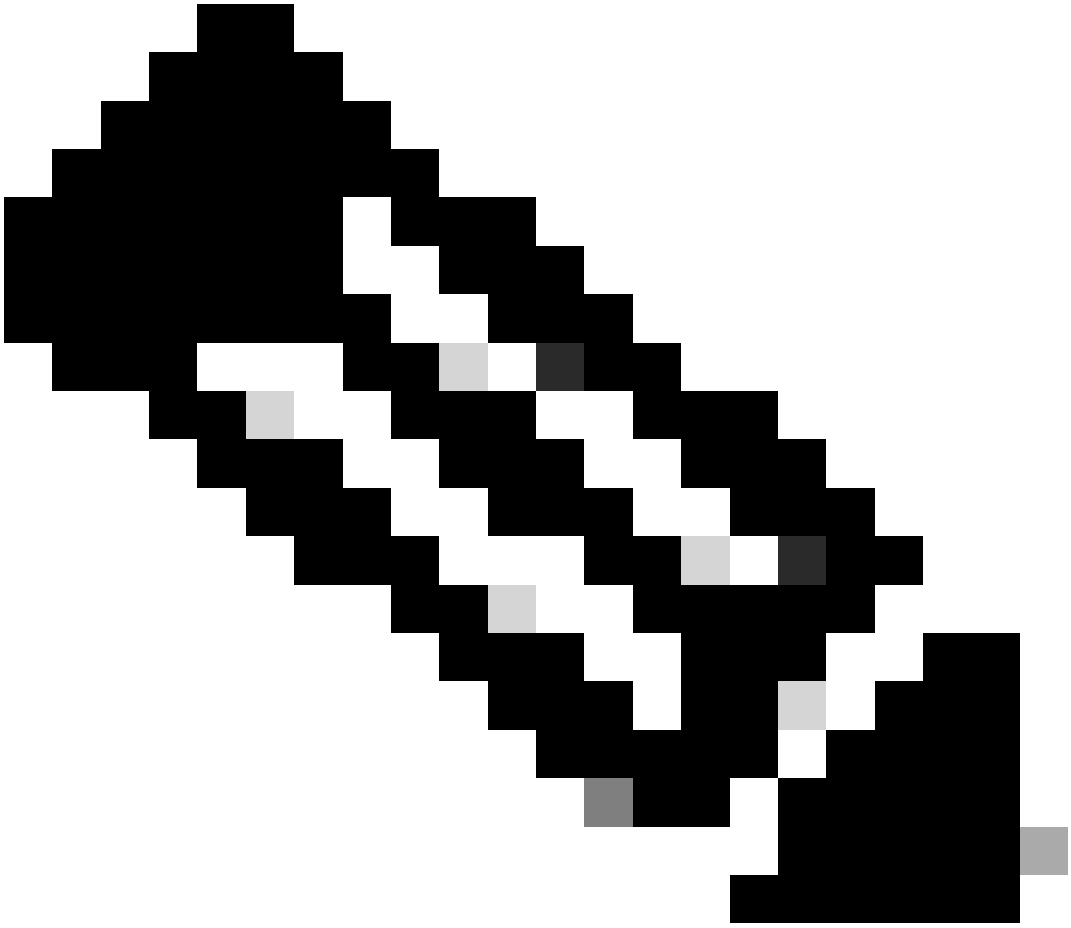
```
DHCP snooping is operational on following VLANs:
```

```
101,201
```

配置（部分隔离保护）

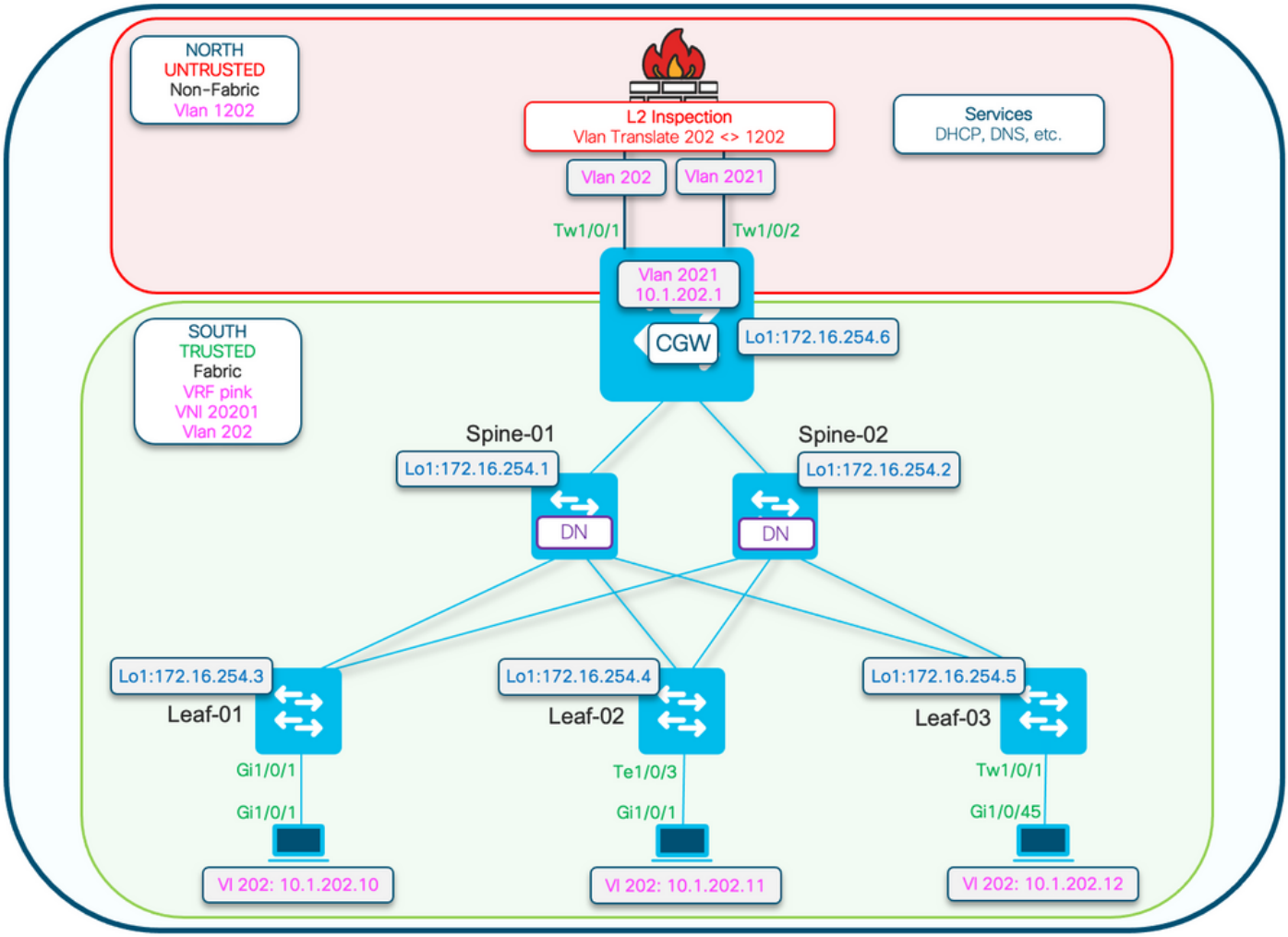
接入枝叶上的DHCP监听依靠来自CGW的默认网关路由来获知用于将DHCP数据包转发到的网关MAC。

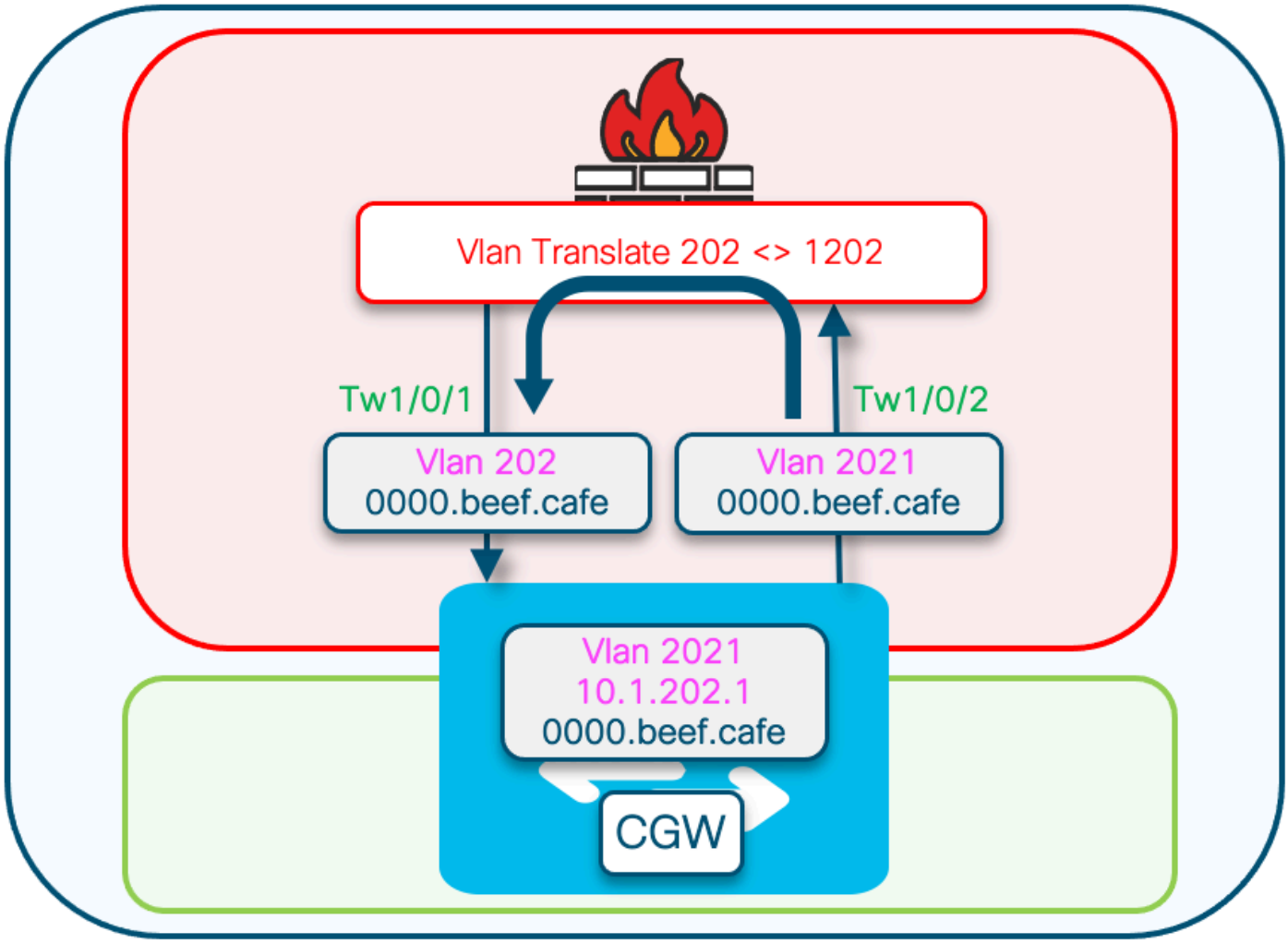
- 使用带有外部网关的部分隔离设计时，在CGW上需要额外的配置才能使用默认网关(DEF GW)属性通告MAC-IP RT2。



注：注：此部分还用于描述完全隔离受保护网段实施，该实施还使用通告到交换矩阵中的GW（与交换矩阵外部的GW）。

网络图


NORTH
UNTRUSTED
Non-Fabric
Vlan 1202
L2 Inspection
Vlan Translate 202 <> 1202
Services
DHCP, DNS, etc.
Vlan 202
Vlan 2021
Tw1/0/1
Tw1/0/2
Vlan 2021
10.1.202.1
CGW
Lo1:172.16.254.6
SOUTH
TRUSTED
Fabric
VRF pink
VNI 20201
Vlan 202
Spine-01
Lo1:172.16.254.1
DN
Spine-02
Lo1:172.16.254.2
DN
Lo1:172.16.254.3
Leaf-01
Lo1:172.16.254.4
Leaf-02
Lo1:172.16.254.5
Leaf-03
Gi1/0/1
Gi1/0/1
Te1/0/3
Gi1/0/1
Tw1/0/1
Gi1/0/45
Vl 202: 10.1.202.10
Vl 202: 10.1.202.11
Vl 202: 10.1.202.12



L2 VTEP（枝叶）密钥详细信息

请求数据包来自客户端

- 使用Default gw advertised CGW mac。
- 如果存在多个gw，则使用第一个gw mac。
- 将外部广播MAC（客户端发起：DORA中的D和R）转换为单播GW MAC并转发到CGW

DHCP监听添加：选项82子选项：电路和RID

（RID由CGW上的响应数据包处理使用）。

(通知CGW其非本地和交换矩阵中继返回L2VTEP)

<#root>

```
Option: (82) Agent Information Option
        Length: 24
```

```
        Option 82 Suboption: (1) Agent Circuit ID
            Length: 12
            Agent Circuit ID: 010a000800000277501010000

        Option 82 Suboption: (2) Agent Remote ID

            Length: 8
          Agent Remote ID:
          000

682c7bf88700 <-- switch base mac 682c.7bf8.8700 (from 'show switch')
```

- 通过vxlan隧道从CGW接收的响应数据包。
- Leaf Strips选项82。
- 添加带有客户端源接口的绑定条目。（vxlan-mod-port提供客户端源接口）。
- 转发到客户端的响应数据包。

L3 VTEP (CGW)密钥详细信息

- 启用DHCP监听
- 在SVI中启用DHCP中继
- 从L2VTEP接收请求，并将其提供给中继。
- 中继添加其他选项82子选项（gi、服务器覆盖等）并发送到DHCP服务器。
- 来自dhcp服务器的DHCP响应首先进入RELAY组件。
- 在RELAY删除选项82参数（gi地址、服务器覆盖等）后，数据包将传递到dhcp监听组件。
- 监听组件检查RID（路由器ID），如果它不是本地路由器，则不会删除选项82子选项1和2。
- 交换矩阵中继（因为RID不是本地的）数据包直接转发到远程客户端。
- 使用客户端Mac并执行网桥插入。 硬件执行客户端mac查找，并将具有vxlan封装的数据包转发到始发L2VTEP。

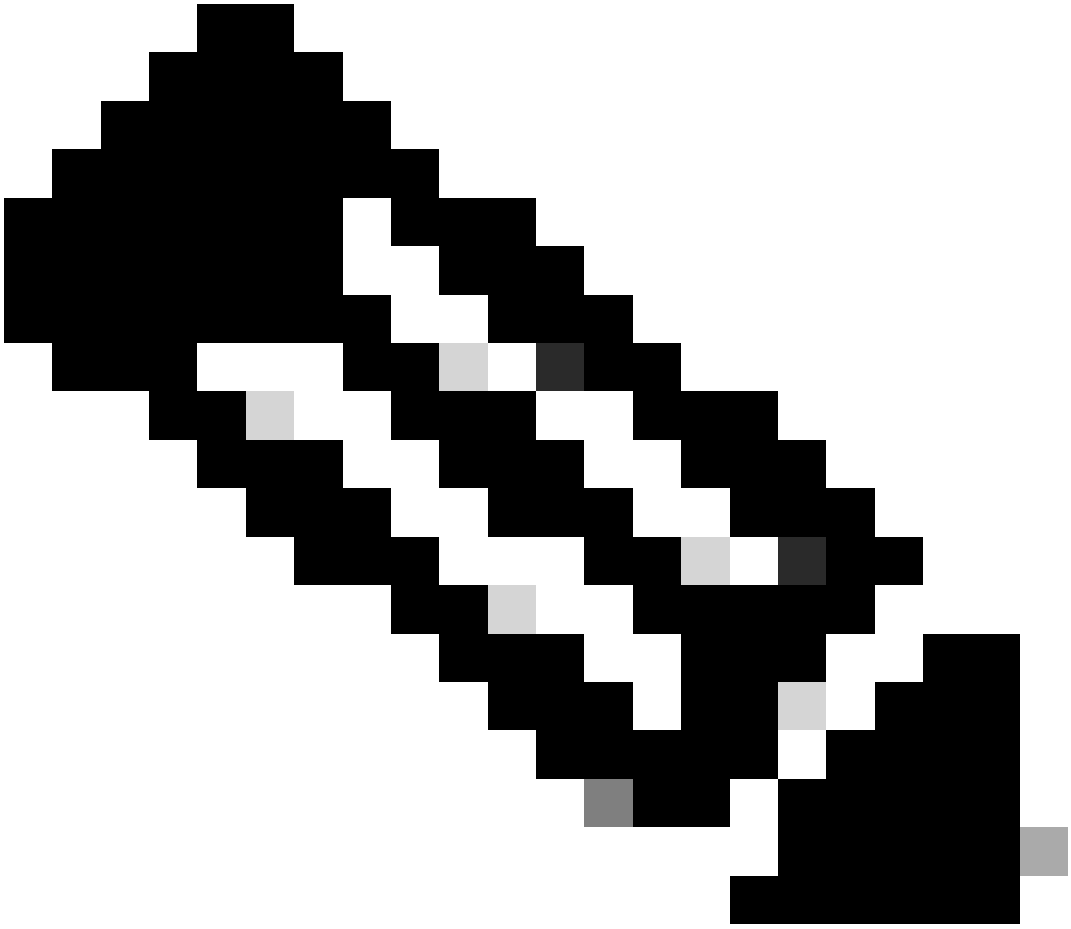
支持DHCP L2中继所需的步骤：

1. 启用ip local learning
2. 创建禁用收集功能的策略
3. 连接到外部网关EVI/VLAN
4. 将静态条目添加到外部网关mac-ip的设备跟踪表中
5. 创建BGP路由映射以匹配RT2 MAC-IP前缀并设置默认网关扩展社区

6. 将路由映射应用到BGP路由反射器邻居

7. 确保DHCP中继的配置正确以处理其他选项

8. 在交换矩阵VLAN和外部GW VLAN上配置DHCP监听



注意：无需在接入枝叶上进行配置更改即可支持带有外部网关的DHCP L2中继。

CGW

启用ip local learning

<#root>

CGW#

show running-config | beg l2vpn evpn instance 202

l2vpn evpn instance 202 vlan-based

```
 encapsulation vxlan
 replication-type ingress
 ip local-learning enable


<-- to advertise RT-2 with default gateway EC, ip local-learning must be enabled on the CGW.


    Use additional device-tracking policy shown in the next output to prevent MAC-IP binding flapping wh
 multicast advertise enable
<--- There is no default-gateway advertise enable cli here, as the SVI (Vlan 2021) used by this segment
```

创建禁用收集功能的策略

```
<#root>
device-tracking policy dt-no-glean <-- Configure device tracking policy to prevent MAC-IP flapping
 security-level glean
 no protocol ndp
 no protocol dhcp6
 no protocol arp
 no protocol dhcp4
```

连接到外部网关evi/vlan

```
<#root>
CGW#
show running-config | sec vlan config
vlan configuration 202
 member evpn-instance 202 vni 20201
 device-tracking attach-policy dt-no-glean <-- apply the new device tracking policy to the vlan configur
```

将静态条目添加到外部网关mac-ip的设备跟踪表中

```
<#root>
device-tracking binding vlan 202 10.1.202.1 interface TwentyFiveGigE1/0/1 0000.beef.cafe
<-- All static entries in device tracking table should be for external gateway mac-ip's.
```

```
    If there is any other static entry in device tracking table, match ip/ipv6 configurations in route m
```

创建BGP路由映射以匹配RT2 MAC-IP前缀并设置默认网关扩展社区

<#root>

```
route-map CGW_DEF_GW permit 10
 match evpn route-type 2-mac-ip <-- match RT2 type MAC-IP


 set extcommunity default-gw    <-- Set Default-gateway (DEF GW 0:0) extended community

route-map CGW_DEF_GW permit 20
```

将路由映射应用到BGP路由反射器邻居

<#root>

```
CGW#
sh run | sec router bgp


address-family l2vpn evpn
 neighbor 172.16.255.1 activate
 neighbor 172.16.255.1 send-community both
 neighbor 172.16.255.1
route-map CGW_DEF_GW out  <-- Sets the DEF GW Community when it advertises MAC-IP type RT2 to the RR


 neighbor 172.16.255.2 activate
 neighbor 172.16.255.2 send-community both
 neighbor 172.16.255.2
route-map CGW_DEF_GW out <-- Sets the DEF GW Community when it advertises MAC-IP type RT2 to the RR
```

确保DHCP中继的配置正确以处理其他选项

<#root>

```
CGW#
show run int vl 2021
Building configuration...

Current configuration : 315 bytes
!
interface Vlan2021
 mac-address 0000.beef.cafe
```

```
 vrf forwarding pink

 ip dhcp relay information option vpn-id   <-- Ensure the vrf info is passed to the server
 ip dhcp relay source-interface Loopback0  <-- sets the relay source interface to the loopback

 ip address 10.1.202.1 255.255.255.0

 ip helper-address global 10.1.33.33       <-- In this scenario the next hop to the DHCP server is in th

 no ip redirects
 ip local-proxy-arp
 ip route-cache same-interface
 no autostate
```

在交换矩阵VLAN和外部GW VLAN上配置DHCP监听

<#root>

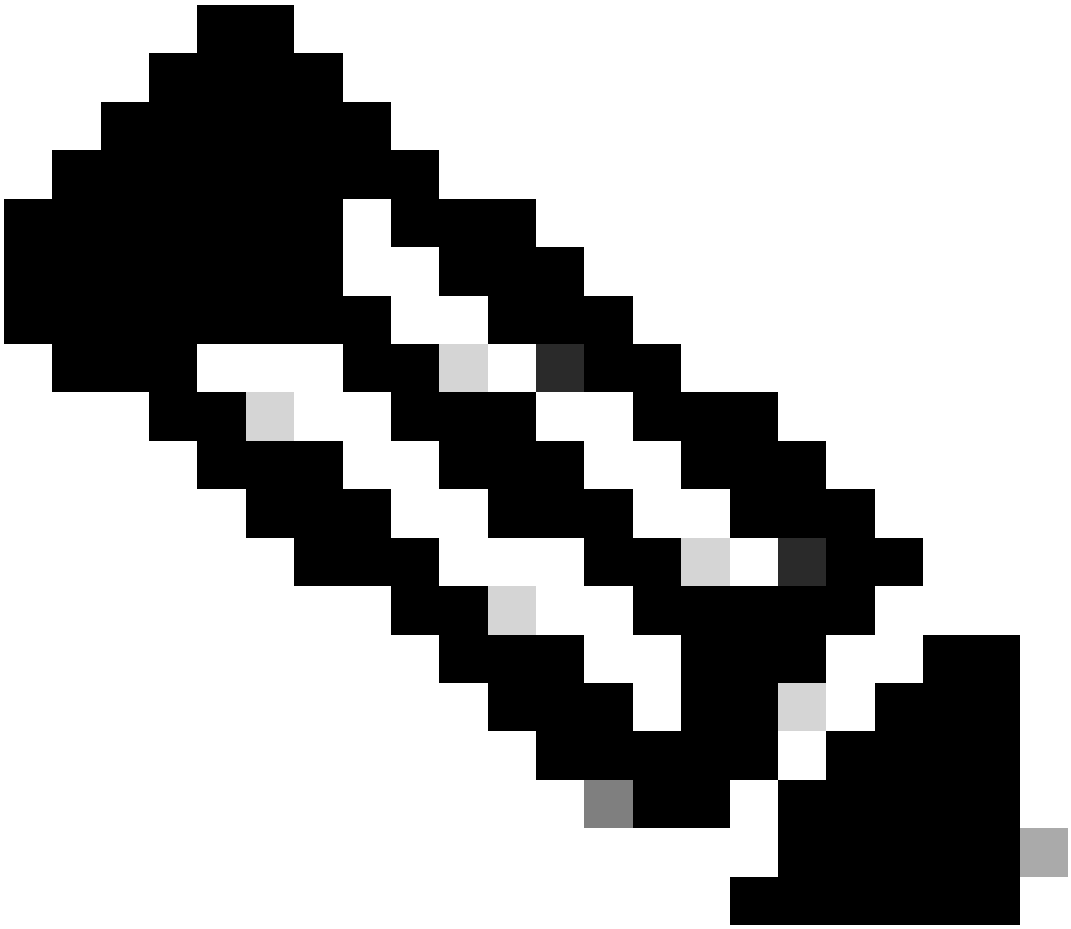
```
Leaf01#

sh run | s dhcp snoop

ip dhcp snooping vlan 202
ip dhcp snooping

CGW#

sh run | s dhcp snoop

ip dhcp snooping vlan 202,2021  <-- snooping is required in both the fabric vlan and the external GW vla

ip dhcp snooping
```

确保到DHCP服务器的上行链路在CGW上受信任

<#root>

```
CGW#

sh run int tw 1/0/1


interface TwentyFiveGigE1/0/1
 switchport trunk allowed vlan 202
 switchport mode trunk

 ip dhcp snooping trust

end

CGW#

sh run int tw 1/0/2


interface TwentyFiveGigE1/0/2
 switchport trunk allowed vlan 33,2021
 switchport mode trunk
```

```
 ip dhcp snooping trust
end
```



注意：由于服务器在防火墙设备上的放置方式，因此两个面向此设备的链路上都配置了信任。在放大图中，您可以看到此设计中的Offer同时到达Tw1/0/1和Tw1/0/2。

验证（部分隔离保护）

网关前缀（枝叶）

<#root>

```
Leaf01#
show bgp l2vpn evpn route-type 2 0 0000.beef.cafe 10.1.202.1
```

```
BGP routing table entry for [2][172.16.254.3:202][0][48][0000BEEFCAFE][32][10.1.202.1]/24, version 3411
Paths: (1 available, best #1, table evi_202)
  Not advertised to any peer
  Refresh Epoch 2
  Local, imported path from [2][172.16.254.6:202][0][48][0000BEEFCAFE][32][10.1.202.1]/24 (global)
    172.16.254.6 (metric 3) (via default) from 172.16.255.1 (172.16.255.1)
      Origin incomplete, metric 0, localpref 100, valid, internal, best
      EVPN ESI: 00000000000000000000, Label1 20201
      Extended Community: RT:65001:202 ENCAP:8
```

EVPN DEF GW:0:0 <-- GW attribute added indicating this is GW prefix which L2 Relay uses

```
      Originator: 172.16.255.6, Cluster list: 172.16.255.1
      rx pathid: 0, tx pathid: 0x0
      Updated on Sep 19 2023 19:57:25 UTC
```

FED MATM（枝叶）

确认枝叶已在硬件中安装CGW远程MAC

<#root>

```
Leaf01#
```

show platform software fed switch active matm macTable vlan 202

```
VLAN   MAC                 Type  Seq#    EC_Bi  Flags  machandle          siHandle           riHandl
-------------------------------------------------------------------------------------------------
202    0006.f601.cd01       0x1    1093      0      0  0x71e05918f138     0x71e05917a1a8     0x0

202    0006.f601.cd44       0x1   14309      0      0  0x71e058cdc208     0x71e05917a1a8     0x0
```

202

0000.beef.cafe 0x5000001

```
      0      0     64  0x71e058ee5d88      0x71e059195f88      0x71e059171678      0x0
```

<--- The GW MAC shows learnt via the Border Leaf Loopback

```
Total Mac number of addresses:: 3
Summary:
Total number of secure addresses:: 0
Total number of drop addresses:: 0
Total number of lisp local addresses:: 0
Total number of lisp remote addresses:: 1
*a_time=aging_time(secs)  *e_time=total_elapsed_time(secs)
Type:
```

MAT_DYNAMIC_ADDR 0x1

```
  MAT_STATIC_ADDR       0x2  MAT_CPU_ADDR          0x4  MAT_DISCARD_ADDR      0x8
MAT_ALL_VLANS          0x10  MAT_NO_FORWARD       0x20  MAT_IPMULT_ADDR       0x40  MAT_RES
MAT_DO_NOT_AGE        0x100  MAT_SECURE_ADDR     0x200  MAT_NO_PORT          0x400  MAT_DRO
MAT_DUP_ADDR         0x1000  MAT_NULL_DESTINATION 0x2000  MAT_DOT1X_ADDR      0x4000  MAT_ROU
MAT_WIRELESS_ADDR   0x10000  MAT_SECURE_CFG_ADDR 0x20000  MAT_OPQ_DATA_PRESENT 0x40000  MAT_WIR
MAT_DLR_ADDR       0x100000  MAT_MRP_ADDR       0x200000  MAT_MSRP_ADDR      0x400000  MAT_LIS
```

MAT_LISP_REMOTE_ADDR 0x1000000

MAT_VPLS_ADDR

0x2000000 MAT_LISP_GW_ADDR 0x4000000

<-- these 3 values added = 0x5000001 (note that 0x4000000 = GW type address

本地MAC（枝叶）

<#root>

Leaf01#

show switch

```
Switch/Stack Mac Address : 682c.7bf8.8700 - Local Mac Address
Mac persistency wait time: Indefinite
                                             H/W   Current
Switch#   Role    Mac Address     Priority Version  State
------------------------------------------------------------------------------------
*1       Active
```

682c.7bf8.8700

```
     1      V01     Ready
```

<-- this is the MAC that will be added to DHCP Agent Remote ID

DHCP监听（枝叶和CGW）

确认已在交换矩阵vlan中的枝叶上启用DHCP监听

<#root>

Leaf01#

show ip dhcp snooping

```
Switch DHCP snooping is enabled
Switch DHCP gleaning is disabled
DHCP snooping is configured on following VLANs:
202
```

DHCP snooping is operational on following VLANs: <-- Snooping on in the Fabric Vlan 202

```
<...snip...>
```

Insertion of option 82 is enabled
circuit-id default format: vlan-mod-port
remote-id: 682c.7bf8.8700 (MAC) <--- Remote ID (RID) inserted by Leaf to DHCP packets

```
<...snip...>
```

确认已在交换矩阵和外部网关vlan中的CGW上启用DHCP监听

<#root>

CGW#

show ip dhcp snooping

Switch DHCP snooping is enabled
Switch DHCP gleaning is disabled
DHCP snooping is configured on following VLANs:
202,2021

DHCP snooping is operational on following VLANs: <-- Snooping on in the Fabric and External GW Vlans 202,2021

<...snip...>

DHCP snooping trust/rate is configured on the following Interfaces:

Interface   Trusted   Allow option   Rate limit (pps)
-----------------------   -------   ------------   ----------------

TwentyFiveGigE1/0/1

             yes   yes   unlimited

<-- Trust set on ports the OFFER arrives on

Interface   Trusted   Allow option   Rate limit (pps)
-----------------------   -------   ------------   ----------------
  Custom circuit-ids:

TwentyFiveGigE1/0/2

              yes   yes   unlimited

<-- Trust set on ports the OFFER arrives on

  Custom circuit-ids:

确认已创建DHCP监听绑定

<#root>

Leaf01#

show ip dhcp snooping binding

MacAddress

IpAddress

        Lease(sec) Type   VLAN

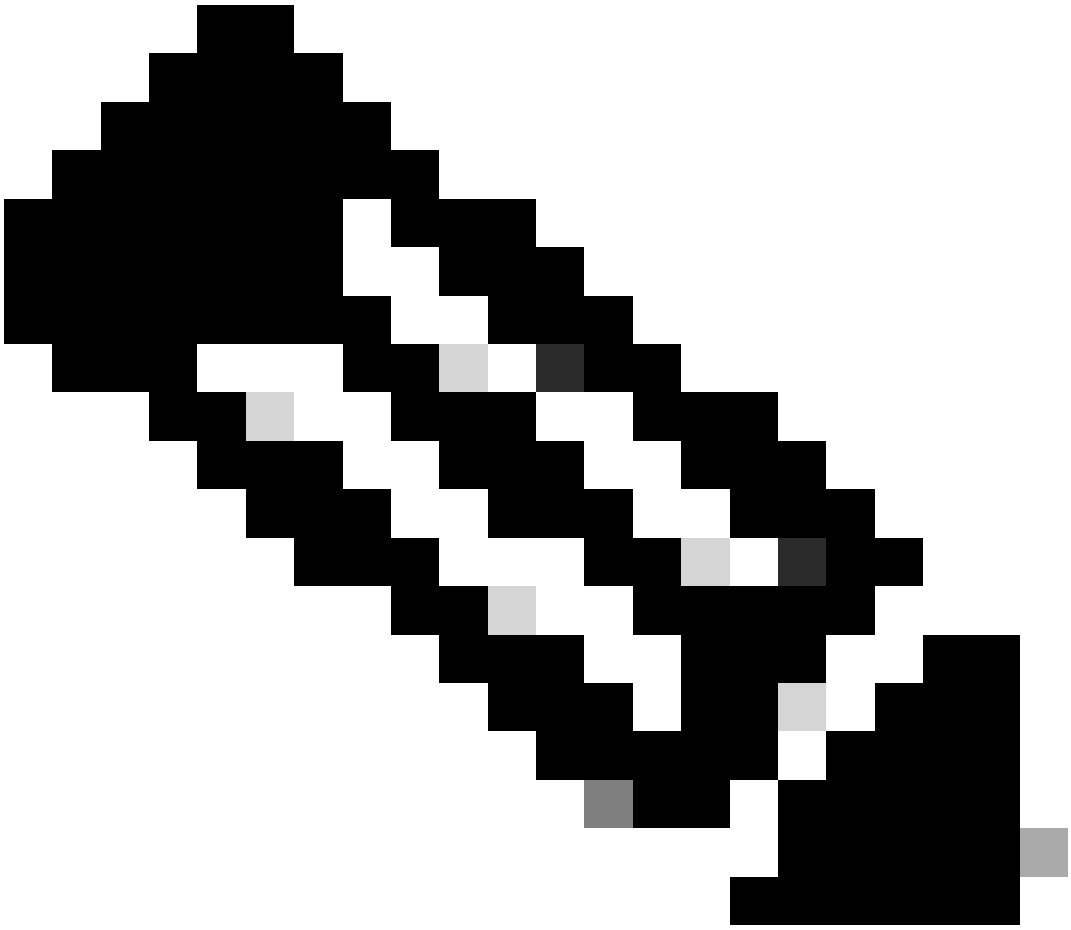
Interface

----------------- -------------- ---------- ------------- ---- --------------------

```
00:06:F6:01:CD:43
10.1.202.10
      34261       dhcp-snooping   202
GigabitEthernet1/0/1  <-- DHCP Snooping has created the binding
Total number of bindings: 1
```

故障排除（任何CGW类型）

调试有助于显示DHCP监听和L2中继进程如何处理DHCP数据包。



注意：这些调试可用于任何类型的使用带DCHP L2中继的CGW的部署。

DHCP监听调试（枝叶）

调试监听以确认数据包处理

<#root>

Leaf01#

debug ip dhcp snooping packet

DHCP Snooping Packet debugging is on

Leaf01#

show debugging

DHCP Snooping packet debugging is on

开始主机DHCP地址尝试

- 对于本文档，执行通过DHCP寻址的SVI的关闭/不关闭以触发DORA交换
- 对于Windows主机，可以执行ipconfig /release > ipconfig /renew

从show logging或终端窗口收集调试

DHCP 发现

发现来自面向主机的端口

<#root>

*Sep 19 20:16:31.164:

DHCP_SNOOPING: received new DHCP packet from input interface (GigabitEthernet1/0/1) <-- host facing por

*Sep 19 20:16:31.177:

DHCP_SNOOPING: process new DHCP packet, message type: DHCPDISCOVER, input interface: Gi1/0/1

, MAC da: ffff.ffff.ffff,

MAC sa: 0006.f601.cd43

, IP da: 255.255.255.255, IP sa: 0.0.0.0, DHCP ciaddr: 0.0.0.0, DHCP yiaddr: 0.0.0.0, DHCP siaddr: 0.0.
*Sep 19 20:16:31.177: DHCP_SNOOPING: add relay information option.
*Sep 19 20:16:31.177:

DHCP_SNOOPING: Encoding opt82 CID in vlan-mod-port format   <-- Option 82 encoding

*Sep 19 20:16:31.177: DHCP_SNOOPING:VxLAN : vlan_id 202 VNI 20201 mod 1 port 1
*Sep 19 20:16:31.177:

DHCP_SNOOPING: Encoding opt82 RID in MAC address format   <-- Encoding the switch Remote ID (loc

*Sep 19 20:16:31.177: DHCP_SNOOPING: binary dump of relay info option, length: 26 data:
0x52 0x18 0x1 0xC 0x1 0xA 0x0 0x8 0x0 0x0 0x4E 0xE9 0x1 0x1 0x0 0x0 0x2 0x8 0x0 0x6

0x68 0x2C 0x7B 0xF8 0x87 0x0 <-- the switch local MAC 682c.7bf8.8700

```
*Sep 19 20:16:31.177: DHCPS BRIDGE PAK: vlan=202 platform_flags=1
*Sep 19 20:16:31.177: DHCP_SNOOPING: bridge packet get invalid mat entry: FFFF.FFFF.FFFF, packet is flo
*Sep 19 20:16:31.177:

DHCP_SNOOPING: L2RELAY: sent unicast packet to default gw: 0000.beef.cafe vlan 0 src intf GigabitEtherne
```

DHCP 提供

发现优惠从交换矩阵隧道接口到达

<#root>

```
*Sep 19 20:16:33.180:

DHCP_SNOOPING: received new DHCP packet from input interface (Tunnel0)

*Sep 19 20:16:33.194:

DHCP_SNOOPING: process new DHCP packet, message type: DHCPOFFER, input interface: Tu0, MAC da: 0006.f601

, IP da: 255.255.255.255, IP sa: 10.1.202.1, DHCP ciaddr: 0.0.0.0, DHCP yiaddr: 10.1.202.18, DHCP siadd
*Sep 19 20:16:33.194: DHCP_SNOOPING: binary dump of option 82, length: 26 data:
0x52 0x18 0x1 0xC 0x1 0xA 0x0 0x8 0x0 0x0 0x4E 0xE9 0x1 0x1 0x0 0x0 0x2 0x8 0x0 0x6 0x68 0x2C 0x7B 0xF8
*Sep 19 20:16:33.194: DHCP_SNOOPING: binary dump of extracted circuit id, length: 14 data:
0x1 0xC 0x1 0xA 0x0 0x8 0x0 0x0 0x4E 0xE9 0x1 0x1 0x0 0x0
*Sep 19 20:16:33.194: DHCP_SNOOPING: binary dump of extracted remote id, length: 10 data:
0x2 0x8 0x0 0x6

0x68 0x2C 0x7B 0xF8 0x87 0x0                                    <-- the switch local MAC 682c.7bf8.8700

*Sep 19 20:16:33.194:  actual_fmt_cid OPT82_FMT_CID_VXLAN_MOD_PORT_INTF global_opt82_fmt_rid OPT82_FMT_
*Sep 19 20:16:33.194: dhcp_snooping_platform_is_local_dhcp_packet: VXLAN-MOD-PORT opt82 vni 20201, vlan
*Sep 19 20:16:33.194:

DHCP_SNOOPING: opt82 data indicates local packet      <-- switch found its own RID in Option 82 paramete


*Sep 19 20:16:33.194: DHCP_SNOOPING: remove relay information option.
*Sep 19 20:16:33.194: DHCP_SNOOPING opt82_fmt_cid_intf OPT82_FMT_CID_VXLAN_MOD_PORT_INTF opt82_fmt_cid_
*Sep 19 20:16:33.194:

DHCP_SNOOPING: VxLAN vlan_id 202 VNI 20201 mod 1 port 1

*Sep 19 20:16:33.194:

DHCP_SNOOPING: mod 1 port 1 idb Gi1/0/1 found for 0006.f601.cd43

*Sep 19 20:16:33.194: DHCP_SNOOPING: calling forward_dhcp_reply
*Sep 19 20:16:33.194: platform lookup dest vlan for input_if: Tunnel0, is  tunnel, if_output: NULL, if_
*Sep 19 20:16:33.194: DHCP_SNOOPING opt82_fmt_cid_intf OPT82_FMT_CID_VXLAN_MOD_PORT_INTF opt82_fmt_cid_
*Sep 19 20:16:33.194: DHCP_SNOOPING: VxLAN vlan_id 202 VNI 20201 mod 1 port 1
*Sep 19 20:16:33.194: DHCP_SNOOPING: mod 1 port 1 idb Gi1/0/1 found for 0006.f601.cd43
*Sep 19 20:16:33.194: DHCP_SNOOPING: vlan 202 after pvlan check
*Sep 19 20:16:33.207:

DHCP_SNOOPING: direct forward dhcp replyto output port: GigabitEthernet1/0/1.  <-- sending packet to hos
```

DHCP 请求

从面向主机的端口看到请求

<#root>

*Sep 19 20:16:33.209:

DHCP_SNOOPING: received new DHCP packet from input interface (GigabitEthernet1/0/1)

*Sep 19 20:16:33.222:

DHCP_SNOOPING: process new DHCP packet, message type: DHCPREQUEST

, input interface: Gi1/0/1, MAC da: ffff.ffff.ffff, MAC sa: 0006.f601.cd43, IP da: 255.255.255.255, IP
*Sep 19 20:16:33.222: DHCP_SNOOPING: add relay information option.
*Sep 19 20:16:33.222: DHCP_SNOOPING: Encoding opt82 CID in vlan-mod-port format
*Sep 19 20:16:33.222: DHCP_SNOOPING:VxLAN : vlan_id 202 VNI 20201 mod 1 port 1
*Sep 19 20:16:33.222: DHCP_SNOOPING: Encoding opt82 RID in MAC address format
*Sep 19 20:16:33.222: DHCP_SNOOPING: binary dump of relay info option, length: 26 data:
0x52 0x18 0x1 0xC 0x1 0xA 0x0 0x8 0x0 0x0 0x4E 0xE9 0x1 0x1 0x0 0x0 0x2 0x8 0x0 0x6 0x68 0x2C 0x7B 0xF8
*Sep 19 20:16:33.222: DHCPS BRIDGE PAK: vlan=202 platform_flags=1
*Sep 19 20:16:33.222: DHCP_SNOOPING: bridge packet get invalid mat entry: FFFF.FFFF.FFFF, packet is flo
*Sep 19 20:16:33.222:

DHCP_SNOOPING: L2RELAY: sent unicast packet to default gw: 0000.beef.cafe vlan 0 src intf GigabitEtherne

DHCP确认

发现确认从交换矩阵隧道接口到达

<#root>

*Sep 19 20:16:33.225:

DHCP_SNOOPING: received new DHCP packet from input interface (Tunnel0)

*Sep 19 20:16:33.238:

DHCP_SNOOPING: process new DHCP packet, message type: DHCPACK, input interface: Tu0, MAC da: 0006.f601.c

, IP da: 255.255.255.255, IP sa: 10.1.202.1, DHCP ciaddr: 0.0.0.0, DHCP yiaddr: 10.1.202.10, DHCP siadd
*Sep 19 20:16:33.238: DHCP_SNOOPING: binary dump of option 82, length: 26 data:
0x52 0x18 0x1 0xC 0x1 0xA 0x0 0x8 0x0 0x0 0x4E 0xE9 0x1 0x1 0x0 0x0 0x2 0x8 0x0 0x6 0x68 0x2C 0x7B 0xF8
*Sep 19 20:16:33.239: DHCP_SNOOPING: binary dump of extracted circuit id, length: 14 data:
0x1 0xC 0x1 0xA 0x0 0x8 0x0 0x0 0x4E 0xE9 0x1 0x1 0x0 0x0
*Sep 19 20:16:33.239: DHCP_SNOOPING: binary dump of extracted remote id, length: 10 data:
0x2 0x8 0x0 0x6 0x68 0x2C 0x7B 0xF8 0x87 0x0
*Sep 19 20:16:33.239: actual_fmt_cid OPT82_FMT_CID_VXLAN_MOD_PORT_INTF global_opt82_fmt_rid OPT82_FMT_
*Sep 19 20:16:33.239: dhcp_snooping_platform_is_local_dhcp_packet: VXLAN-MOD-PORT opt82 vni 20201, vlan
*Sep 19 20:16:33.239:

DHCP_SNOOPING: opt82 data indicates local packet

*Sep 19 20:16:33.239:

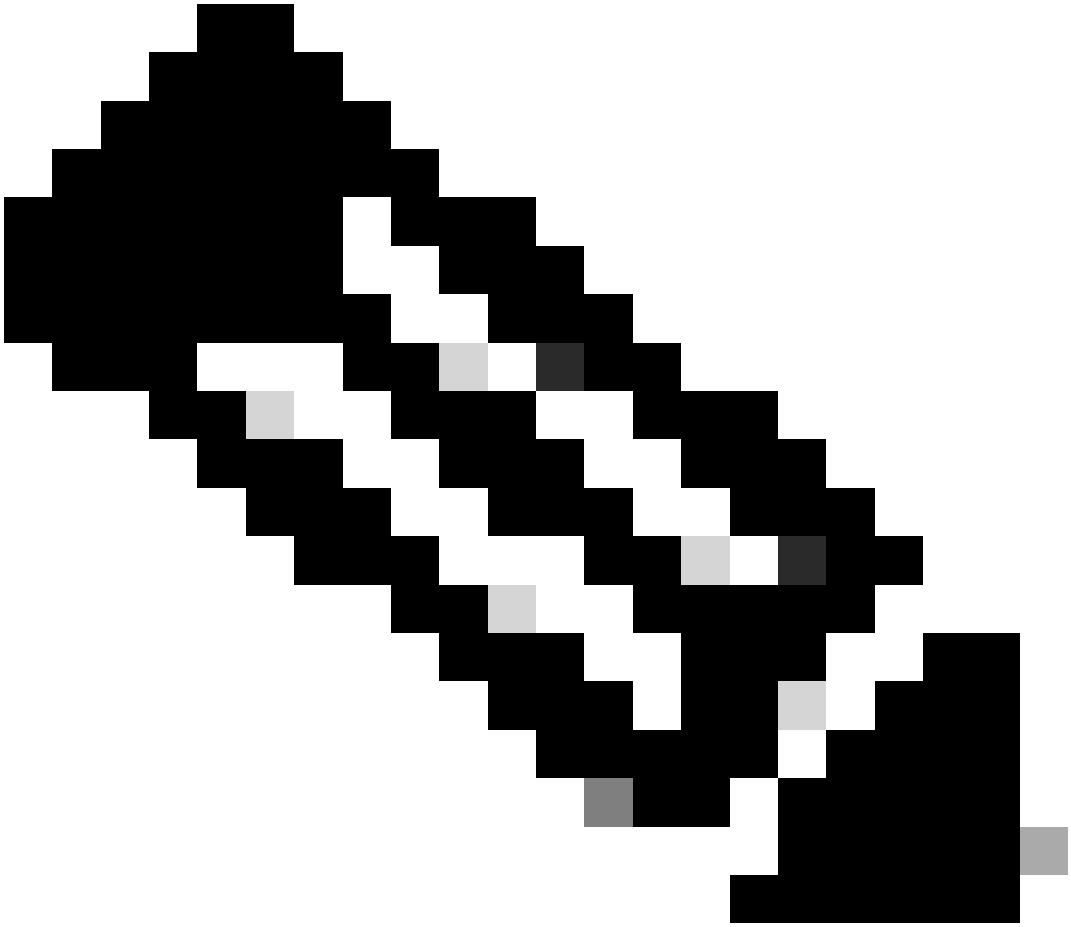
dhcp_snooping_platform_is_local_dhcp_packet: VXLAN-MOD-PORT opt82 vni 20201, vlan_id 202

```
*Sep 19 20:16:33.239: DHCP_SNOOPING: opt82 data indicates local packet
*Sep 19 20:16:33.239: DHCP_SNOOPING opt82_fmt_cid_intf OPT82_FMT_CID_VXLAN_MOD_PORT_INTF opt82_fmt_cid_
*Sep 19 20:16:33.239: DHCP_SNOOPING: VxLAN vlan_id 202 VNI 20201 mod 1 port 1
*Sep 19 20:16:33.239:
```

DHCP_SNOOPING: mod 1 port 1 idb Gi1/0/1 found for 0006.f601.cd43

```
*Sep 19 20:16:33.239: DHCP_SNOOPING: Reroute dhcp pak, message type: DHCPACK
*Sep 19 20:16:33.239: DHCP_SNOOPING: remove relay information option.
*Sep 19 20:16:33.239: DHCP_SNOOPING opt82_fmt_cid_intf OPT82_FMT_CID_VXLAN_MOD_PORT_INTF opt82_fmt_cid_
*Sep 19 20:16:33.239: DHCP_SNOOPING: VxLAN vlan_id 202 VNI 20201 mod 1 port 1
*Sep 19 20:16:33.239: DHCP_SNOOPING: mod 1 port 1 idb Gi1/0/1 found for 0006.f601.cd43
*Sep 19 20:16:33.239: DHCP_SNOOPING: calling forward_dhcp_reply
*Sep 19 20:16:33.239: platform lookup dest vlan for input_if: Tunnel0, is  tunnel, if_output: NULL, if_
*Sep 19 20:16:33.239: DHCP_SNOOPING opt82_fmt_cid_intf OPT82_FMT_CID_VXLAN_MOD_PORT_INTF opt82_fmt_cid_
*Sep 19 20:16:33.239: DHCP_SNOOPING: VxLAN vlan_id 202 VNI 20201 mod 1 port 1
*Sep 19 20:16:33.239: DHCP_SNOOPING: mod 1 port 1 idb Gi1/0/1 found for 0006.f601.cd43
*Sep 19 20:16:33.239: DHCP_SNOOPING: vlan 202 after pvlan check
*Sep 19 20:16:33.252:
```

DHCP_SNOOPING: direct forward dhcp replyto output port: GigabitEthernet1/0/1.



注意：这些调试会被截取。它们会生成数据包的内存转储，但调试结果的此部分注释不在本文档的讨论范围之内。

DHCP监听调试(CGW)

DHCP 发现

由于数据包在CGW上发送和接收的方式（在防火墙上迂回），调试会触发两次

从隧道接口上的交换矩阵到达并从Tw 1/0/1发送到交换矩阵VLAN 202中的防火墙

```
<#root>

*Apr 16 14:37:43.890:

 DHCP_SNOOPING: received new DHCP packet from input interface (Tunnel0)  <-- Discover sent from Leaf01 a


*Apr 16 14:37:43.901: DHCP_SNOOPING: process new DHCP packet, message type: DHCPDISCOVER, input interfa
*Apr 16 14:37:43.901: DHCPS BRIDGE PAK: vlan=202 platform_flags=1
*Apr 16 14:37:43.901:

DHCP_SNOOPING: bridge packet send packet to port: TwentyFiveGigE1/0/1, pak_vlan 202. <-- Sent to Firewal
```

从Vlan 2021中Tw 1/0/2上的防火墙到达，将发送到SVI和帮助程序到DHCP服务器

```
<#root>

*Apr 16 14:37:43.901:

DHCP_SNOOPING: received new DHCP packet from input interface (TwentyFiveGigE1/0/2) <-- Firewall sends di


*Apr 16 14:37:43.911: DHCP_SNOOPING: process new DHCP packet, message type: DHCPDISCOVER, input interfa
*Apr 16 14:37:43.911:

DHCPS BRIDGE PAK: vlan=2021 platform_flags=1      <-- Vlan discover seen is now 2021


*Apr 16 14:37:43.911:

DHCP_SNOOPING: Packet destined to SVI Mac:0000.beef.cafe


*Apr 16 14:37:43.911:

DHCP_SNOOPING: bridge packet send packet to cpu port: Vlan2021.  <-- Packet punted to CPU for handling b
```

DHCP 提供

从DHCP服务器返回SVI 2021，在此配置帮助程序并将其转发到防火墙

<#root>

*Apr 16 14:37:45.913:

DHCP_SNOOPING: received new DHCP packet from input interface (Vlan2021) <-- Arriving from the DHCP serv

*Apr 16 14:37:45.923:

DHCP_SNOOPING: process new DHCP packet, message type: DHCPOFFER, input interface: Vl2021

, MAC da: ffff.ffff.ffff, MAC sa: 0000.beef.cafe, IP da: 255.255.255.255, IP sa: 10.1.202.1, DHCP ciadd
*Apr 16 14:37:45.923: DHCP_SNOOPING: binary dump of option 82, length: 26 data:
0x52 0x18 0x1 0xC 0x1 0xA 0x0 0x8 0x0 0x0 0x4E 0xE9 0x1 0x1 0x0 0x0 0x2 0x8 0x0 0x6 0x68 0x2C 0x7B 0xF8
*Apr 16 14:37:45.924: DHCP_SNOOPING: binary dump of extracted circuit id, length: 14 data:
0x1 0xC 0x1 0xA 0x0 0x8 0x0 0x0 0x4E 0xE9 0x1 0x1 0x0 0x0
*Apr 16 14:37:45.924: DHCP_SNOOPING: binary dump of extracted remote id, length: 10 data:
0x2 0x8 0x0 0x6 0x68 0x2C 0x7B 0xF8 0x87 0x0
*Apr 16 14:37:45.924: actual_fmt_cid OPT82_FMT_CID_VXLAN_MOD_PORT_INTF global_opt82_fmt_rid OPT82_FMT_R
*Apr 16 14:37:45.924: dhcp_snooping_platform_is_local_dhcp_packet: VXLAN-MOD-PORT opt82 vni 20201, vlan
*Apr 16 14:37:45.924:

DHCP_SNOOPING: opt82 data indicates not a local packet

*Apr 16 14:37:45.924: DHCP_SNOOPING: can't parse option 82 data of the message,it is either in wrong fo

<-- This is expected even in working scenario (disregard it)

*Apr 16 14:37:45.924: DHCP_SNOOPING: calling forward_dhcp_reply
*Apr 16 14:37:45.924: platform lookup dest vlan for input_if: Vlan2021, is NOT tunnel, if_output: Vlan2
*Apr 16 14:37:45.924: DHCP_SNOOPING: vlan 2021 after pvlan check
*Apr 16 14:37:45.934:

DHCP_SNOOPING: direct forward dhcp replyto output port: TwentyFiveGigE1/0/2. <-- sending back toward the

从交换矩阵VLAN中的防火墙到达，并从CGW发送到交换矩阵到枝叶

<#root>

*Apr 16 14:37:45.934:

DHCP_SNOOPING: received new DHCP packet from input interface (TwentyFiveGigE1/0/1)

*Apr 16 14:37:45.944:

DHCP_SNOOPING: process new DHCP packet, message type: DHCPOFFER, input interface: Twe1/0/1

, MAC da: ffff.ffff.ffff, MAC sa: 0000.beef.cafe, IP da: 255.255.255.255, IP sa: 10.1.202.1, DHCP ciadd
*Apr 16 14:37:45.944: DHCP_SNOOPING: binary dump of option 82, length: 26 data:
0x52 0x18 0x1 0xC 0x1 0xA 0x0 0x8 0x0 0x0 0x4E 0xE9 0x1 0x1 0x0 0x0 0x2 0x8 0x0 0x6 0x68 0x2C 0x7B 0xF8
*Apr 16 14:37:45.944: DHCP_SNOOPING: binary dump of extracted circuit id, length: 14 data:
0x1 0xC 0x1 0xA 0x0 0x8 0x0 0x0 0x4E 0xE9 0x1 0x1 0x0 0x0
*Apr 16 14:37:45.944: DHCP_SNOOPING: binary dump of extracted remote id, length: 10 data:

```
0x2 0x8 0x0 0x6 0x68 0x2C 0x7B 0xF8 0x87 0x0
*Apr 16 14:37:45.944: actual_fmt_cid OPT82_FMT_CID_VXLAN_MOD_PORT_INTF global_opt82_fmt_rid OPT82_FMT_R
*Apr 16 14:37:45.944: dhcp_snooping_platform_is_local_dhcp_packet: VXLAN-MOD-PORT opt82 vni 20201, vlan
*Apr 16 14:37:45.945:
```

DHCP_SNOOPING: opt82 data indicates not a local packet

```
*Apr 16 14:37:45.945: DHCP_SNOOPING: EVPN enabled Ex GW:fabric relay can't parse option 82 data of the
*Apr 16 14:37:45.945: DHCP_SNOOPING: client address lookup failed to locate client interface, retry loo
*Apr 16 14:37:45.945: DHCP_SNOOPING: lookup packet destination port failed to get mat entry for mac: 00
*Apr 16 14:37:45.945:
```

DHCP_SNOOPING: L2RELAY: Ex GW unicast bridge packet to fabric: vlan id 202 from Twe1/0/1 <-- L2 RELAY f

DHCP 请求

<#root>

```
*Apr 16 14:37:45.967:
```

DHCP_SNOOPING: received new DHCP packet from input interface (Tunnel0)

```
*Apr 16 14:37:45.978:
```

DHCP_SNOOPING: process new DHCP packet, message type: DHCPREQUEST

```
, input interface: Tu0, MAC da: 0000.beef.cafe, MAC sa: 0006.f601.cd43, IP da: 255.255.255.255, IP sa:
*Apr 16 14:37:45.978: DHCPS BRIDGE PAK: vlan=202 platform_flags=1
*Apr 16 14:37:45.978:
```

DHCP_SNOOPING: bridge packet send packet to port: TwentyFiveGigE1/0/1, pak_vlan 202. <-- Send toward Fir

<#root>

```
*Apr 16 14:37:45.978:
```

DHCP_SNOOPING: received new DHCP packet from input interface (TwentyFiveGigE1/0/2) <-- Receive from Fire

```
*Apr 16 14:37:45.989:
```

DHCP_SNOOPING: process new DHCP packet, message type: DHCPREQUEST

```
, input interface: Twe1/0/2, MAC da: 0000.beef.cafe, MAC sa: 0006.f601.cd43, IP da: 255.255.255.255, IP
*Apr 16 14:37:45.989: DHCPS BRIDGE PAK: vlan=2021 platform_flags=1
*Apr 16 14:37:45.989: DHCP_SNOOPING: Packet destined to SVI Mac:0000.beef.cafe
*Apr 16 14:37:45.989:
```

DHCP_SNOOPING: bridge packet send packet to cpu port: Vlan2021. <-- Punt to CPU / DHCP helper

DHCP确认

<#root>

*Apr 16 14:37:45.990:

DHCP_SNOOPING: received new DHCP packet from input interface (Vlan2021) <-- Packet back to SVI from DHC

*Apr 16 14:37:46.000:

DHCP_SNOOPING: process new DHCP packet, message type: DHCPACK, input interface: Vl2021

, MAC da: ffff.ffff.ffff, MAC sa: 0000.beef.cafe, IP da: 255.255.255.255, IP sa: 10.1.202.1, DHCP ciadd
*Apr 16 14:37:46.000: DHCP_SNOOPING: binary dump of option 82, length: 26 data:
0x52 0x18 0x1 0xC 0x1 0xA 0x0 0x8 0x0 0x0 0x4E 0xE9 0x1 0x1 0x0 0x0 0x2 0x8 0x0 0x6 0x68 0x2C 0x7B 0xF8
*Apr 16 14:37:46.000: DHCP_SNOOPING: binary dump of extracted circuit id, length: 14 data:
0x1 0xC 0x1 0xA 0x0 0x8 0x0 0x0 0x4E 0xE9 0x1 0x1 0x0 0x0
*Apr 16 14:37:46.000: DHCP_SNOOPING: binary dump of extracted remote id, length: 10 data:
0x2 0x8 0x0 0x6 0x68 0x2C 0x7B 0xF8 0x87 0x0
*Apr 16 14:37:46.001: actual_fmt_cid OPT82_FMT_CID_VXLAN_MOD_PORT_INTF global_opt82_fmt_rid OPT82_FMT_R
*Apr 16 14:37:46.001: dhcp_snooping_platform_is_local_dhcp_packet: VXLAN-MOD-PORT opt82 vni 20201, vlan
*Apr 16 14:37:46.001:

DHCP_SNOOPING: opt82 data indicates not a local packet   <-- found this is coming from Leaf01 RID

*Apr 16 14:37:46.001: DHCP_SNOOPING: can't parse option 82 data of the message,it is either in wrong fo
*Apr 16 14:37:46.001: DHCP_SNOOPING: calling forward_dhcp_reply
*Apr 16 14:37:46.001: platform lookup dest vlan for input_if: Vlan2021, is NOT tunnel, if_output: Vlan2
*Apr 16 14:37:46.001: DHCP_SNOOPING: vlan 2021 after pvlan check
*Apr 16 14:37:46.011:

DHCP_SNOOPING: direct forward dhcp replyto output port: TwentyFiveGigE1/0/2. <-- Send to Firewall

<#root>

*Apr 16 14:37:46.011:

DHCP_SNOOPING: received new DHCP packet from input interface (TwentyFiveGigE1/0/1) <-- Coming back in f

*Apr 16 14:37:46.022:

DHCP_SNOOPING: process new DHCP packet, message type: DHCPACK, input interface: Twe1/0/1,

 MAC da: ffff.ffff.ffff, MAC sa: 0000.beef.cafe, IP da: 255.255.255.255, IP sa: 10.1.202.1, DHCP ciaddr
*Apr 16 14:37:46.022: DHCP_SNOOPING: binary dump of option 82, length: 26 data:
0x52 0x18 0x1 0xC 0x1 0xA 0x0 0x8 0x0 0x0 0x4E 0xE9 0x1 0x1 0x0 0x0 0x2 0x8 0x0 0x6 0x68 0x2C 0x7B 0xF8
*Apr 16 14:37:46.022: DHCP_SNOOPING: binary dump of extracted circuit id, length: 14 data:
0x1 0xC 0x1 0xA 0x0 0x8 0x0 0x0 0x4E 0xE9 0x1 0x1 0x0 0x0
*Apr 16 14:37:46.022: DHCP_SNOOPING: binary dump of extracted remote id, length: 10 data:
0x2 0x8 0x0 0x6 0x68 0x2C 0x7B 0xF8 0x87 0x0
*Apr 16 14:37:46.022: actual_fmt_cid OPT82_FMT_CID_VXLAN_MOD_PORT_INTF global_opt82_fmt_rid OPT82_FMT_R
*Apr 16 14:37:46.022: dhcp_snooping_platform_is_local_dhcp_packet: VXLAN-MOD-PORT opt82 vni 20201, vlan
*Apr 16 14:37:46.022:

DHCP_SNOOPING: opt82 data indicates not a local packet

*Apr 16 14:37:46.022: DHCP_SNOOPING: EVPN enabled Ex GW:fabric relay can't parse option 82 data of the
*Apr 16 14:37:46.022: DHCP_SNOOPING: client address lookup failed to locate client interface, retry loo
*Apr 16 14:37:46.022: DHCP_SNOOPING: lookup packet destination port failed to get mat entry for mac: 00

```
*Apr 16 14:37:46.022: DHCP_SNOOPING: can't find client's destination port, packet is assumed to be not
*Apr 16 14:37:46.022: DHCP_SNOOPING: client address lookup failed to locate client interface, retry loo
*Apr 16 14:37:46.022: DHCP_SNOOPING: lookup packet destination port failed to get mat entry for mac: 00
*Apr 16 14:37:46.022:
```

DHCP_SNOOPING: L2RELAY: Ex GW unicast bridge packet to fabric: vlan id 202 from Twe1/0/1 <-- Send packe

嵌入式捕获

使用EPC确认DHCP数据包交换和参数正确

- 这从CGW的角度显示，但可以在枝叶上重复此过程以验证数据包交换
- 此示例显示Discover，因为其他DHCP数据包的流程和分析相同

检查到枝叶环回的路由

<#root>

CGW#

show ip route 172.16.254.3

```
Routing entry for 172.16.254.3/32
  Known via "ospf 1", distance 110, metric 3, type intra area
  Last update from 172.16.1.25 on TwentyFiveGigE1/0/47, 2w6d ago
  Routing Descriptor Blocks:
  * 172.16.1.29, from 172.16.255.3, 2w6d ago,
```

via TwentyFiveGigE1/0/48

```
      Route metric is 3, traffic share count is 1
    172.16.1.25, from 172.16.255.3, 2w6d ago,
```

via TwentyFiveGigE1/0/47

```
      Route metric is 3, traffic share count is 1
```

配置捕获以在面向Leaf01的链路上运行

```
monitor capture 1 interface TwentyFiveGigE1/0/47 BOTH
monitor capture 1 interface TwentyFiveGigE1/0/48 BOTH
monitor capture 1 match any
monitor capture 1 buffer size 100
monitor capture 1 limit pps 1000
```

开始捕获、触发主机请求DHCP IP地址以及停止捕获

<#root>

```
monitor capture 1 start
  (have the host request dhcp ip)
monitor capture 1 stop
```

查看从DHCP发现开始的捕获结果（注意事务ID以确认这是相同的DORA事件）

<#root>

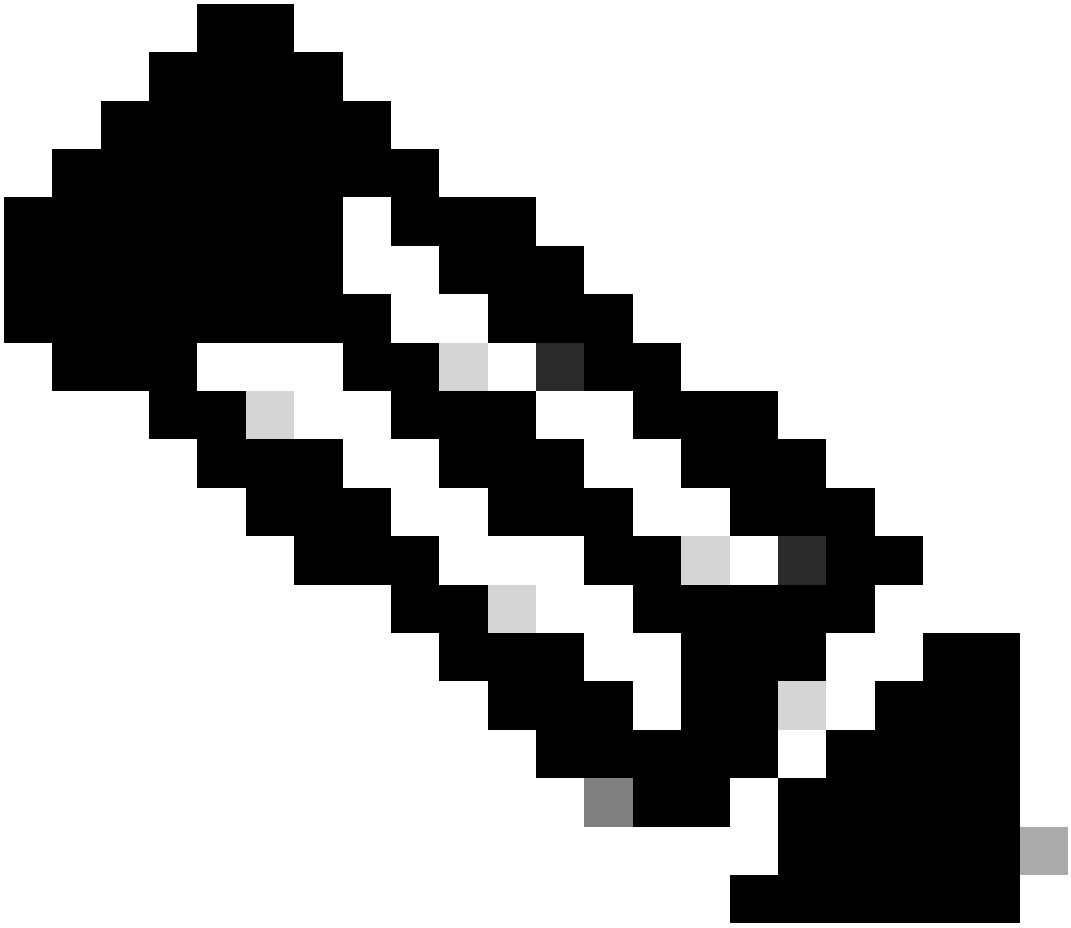
```
CGW#
show monitor cap 1 buff brief | i DHCP


 16
  12.737135      0.0.0.0 -> 255.255.255.255 DHCP 434
DHCP Discover
 -
Transaction ID 0x78b <-- Discover starts at Frame 16 with all same transaction ID
   18  14.740041   10.1.202.1 -> 255.255.255.255 DHCP 438 DHCP
Offer
    - Transaction ID
0x78b
   19  14.742741      0.0.0.0 -> 255.255.255.255 DHCP 452 DHCP
Request
  - Transaction ID
0x78b
   20  14.745646   10.1.202.1 -> 255.255.255.255 DHCP 438 DHCP
ACK
      - Transaction ID
0x78b
```

<#root>

```
CGW#
sh mon cap 1 buff detailed | b Frame 16
Frame 16:
 434 bytes on wire (3472 bits), 434 bytes captured (3472 bits) on interface /tmp/epc_ws/wif_to_ts_pipe,
    [Protocols in frame: eth:ethertype:ip:udp:vxlan:eth:ethertype:ip:udp:dhcp]
Ethernet II,
Src: dc:77:4c:8a:6d:7f
```

```
 (dc:77:4c:8a:6d:7f),

Dst: 10:f9:20:2e:9f:82

 (10:f9:20:2e:9f:82)

<-- Underlay Interface MACs

    Type: IPv4 (0x0800)
Internet Protocol Version 4,

Src: 172.16.254.3, Dst: 172.16.254.6

User Datagram Protocol, Src Port: 65281,

Dst Port: 4789                  <-- VXLAN Port

Virtual eXtensible Local Area Network
    VXLAN Network Identifier

(VNI): 20201                                <-- Correct VNI / Segment

    Reserved: 0
Ethernet II,

Src: 00:06:f6:01:cd:43

 (00:06:f6:01:cd:43),

Dst: 00:00:be:ef:ca:fe

 (00:00:be:ef:ca:fe)

<-- Inner Packet destined to CGW MAC


Internet Protocol Version 4, Src: 0.0.0.0, Dst: 255.255.255.255
User Datagram Protocol,

Src Port: 68, Dst Port: 67                      <-- DHCP ports


Dynamic Host Configuration Protocol (Discover)                          <-- DHCP Discover Packet


 Client MAC address: 00:06:f6:01:cd:43

 (00:06:f6:01:cd:43)
    Client hardware address padding: 00000000000000000000
    Server host name not given
    Boot file name not given
    Magic cookie: DHCP


 Option: (53) DHCP Message Type (Discover)

        Length: 1
        DHCP: Discover (1)
    Option: (57) Maximum DHCP Message Size
        Length: 2
        Maximum DHCP Message Size: 1152
    Option: (61) Client identifier
        Length: 27
        Type: 0
        Client Identifier: cisco-0006.f601.cd43-Vl202
    Option: (12) Host Name
```

```
        Length: 17


Host Name: 9300-HOST-3750X-2

    Option: (55) Parameter Request List
        Length: 8
        Parameter Request List Item: (1) Subnet Mask
        Parameter Request List Item: (6) Domain Name Server
        Parameter Request List Item: (15) Domain Name
        Parameter Request List Item: (44) NetBIOS over TCP/IP Name Server
        Parameter Request List Item: (3) Router
        Parameter Request List Item: (33) Static Route
        Parameter Request List Item: (150) TFTP Server Address
        Parameter Request List Item: (43) Vendor-Specific Information
    Option: (60) Vendor class identifier
        Length: 8
        Vendor class identifier: ciscopnp


Option: (82) Agent Information Option

        Length: 24
        Option 82 Suboption: (1) Agent Circuit ID
            Length: 12
            Agent Circuit ID: 010a000800004ee901010000


Option 82 Suboption: (2) Agent Remote ID

            Length: 8


Agent Remote ID:

000

6682c7bf88700  <-- switch base mac 682c.7bf8.8700 (from 'show switch')

    Option: (255) End
        Option End: 255
```



注意：可在任何枝叶或CGW上使用捕获工具来确定怀疑部分DHCP DORA交换失败的最后一个点。

验证监听统计错误

<#root>

```
Leaf01#

show ip dhcp snooping statistics detail

 Packets Processed by DHCP Snooping                  = 1288


Packets Dropped Because

   IDB not known                                     = 0
   Queue full                                        = 0
   Interface is in errdisabled                       = 0
   Rate limit exceeded                               = 0
   Received on untrusted ports                       = 0
```

```
   Nonzero giaddr                                  = 0
   Source mac not equal to chaddr                  = 0
   No binding entry                                = 0
   Insertion of opt82 fail                         = 0
   Unknown packet                                  = 0
   Interface Down                                  = 0
   Unknown output interface                        = 0
   Misdirected Packets                             = 0
   Packets with Invalid Size                       = 0
   Packets with Invalid Option                     = 0
```

<-- Look for any drop counter that is actively incrementing when the issue is seen.

验证DHCP监听的传送路径

- CoPP是丢弃传送路径中的数据包的主要组件

<#root>

Leaf01#

show platform hardware switch active qos queue stats internal cpu policer

```
                          CPU Queue Statistics
============================================================================================
                                          (default) (set)     Queue        Queue
QId
```

PlcIdx

```
  Queue Name                Enabled   Rate     Rate      Drop(Bytes)
```

Drop(Frames)

```
--------------------------------------------------------------------------------------------
17
```

6

DHCP Snooping

```
               Yes     400       400      0
```

0

```
              CPU Queue Policer Statistics
=============================================================================
```

Policer

```
    Policer Accept   Policer Accept  Policer Drop  Policer Drop
```

Index

```
         Bytes          Frames        Bytes          Frames
-----------------------------------------------------------------
```

```
6          472723           1288            0             0
```

用于定位可能发生数据包泛洪位置的另一个非常有用的命令是“show platform software fed switch active punt rates interfaces”

- 这对于查找发生泛洪的源接口非常有用，因为泛洪会使传送路径拥塞并影响合法的CPU绑定流量

<#root>

```
Leaf01#

show platform software fed switch active punt rates interfaces

Punt Rate on Interfaces Statistics

Packets per second averaged over 10 seconds, 1 min and 5 mins
================================================================================================
                              |           |  Recv |  Recv |  Recv |  Drop |  Drop  |  Drop
<-- Receive and drop rates for this port
 Interface Name               |  IF_ID    |  10s  |  1min |  5min |  10s  |  1min  |  5min
================================================================================================

GigabitEthernet1/0/1          0x0000000a

       2       2       2       0       0       0
<-- the port and its IF-ID which can be used in the next command


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

<#root>

```
Leaf01#

show platform software fed switch active punt rates interfaces 0xa  <-- From previous command (omit the

Punt Rate on Single Interfaces Statistics

Interface : GigabitEthernet1/0/1 [if_id: 0xA]

  Received                        Dropped
  --------                        -------
   Total          : 8032546         Total           : 0
   10 sec average : 2               10 sec average  : 0
    1 min average : 2                1 min average  : 0
    5 min average : 2                5 min average  : 0

Per CPUQ punt stats on the interface

 (rate averaged over 10s interval)
```

```
================================================================================
 Q  |          Queue                  |  Recv   |  Recv  |  Drop  |  Drop  |
 no |          Name                   |  Total  |  Rate  |  Total |  Rate  |
================================================================================
 17

 CPU_Q_DHCP_SNOOPING

                 1216        0        0        0
<...snip...>
```

DHCP监听客户端统计信息

使用此命令观察DHCP消息交换。可以在枝叶或CGW上运行以查看事件跟踪

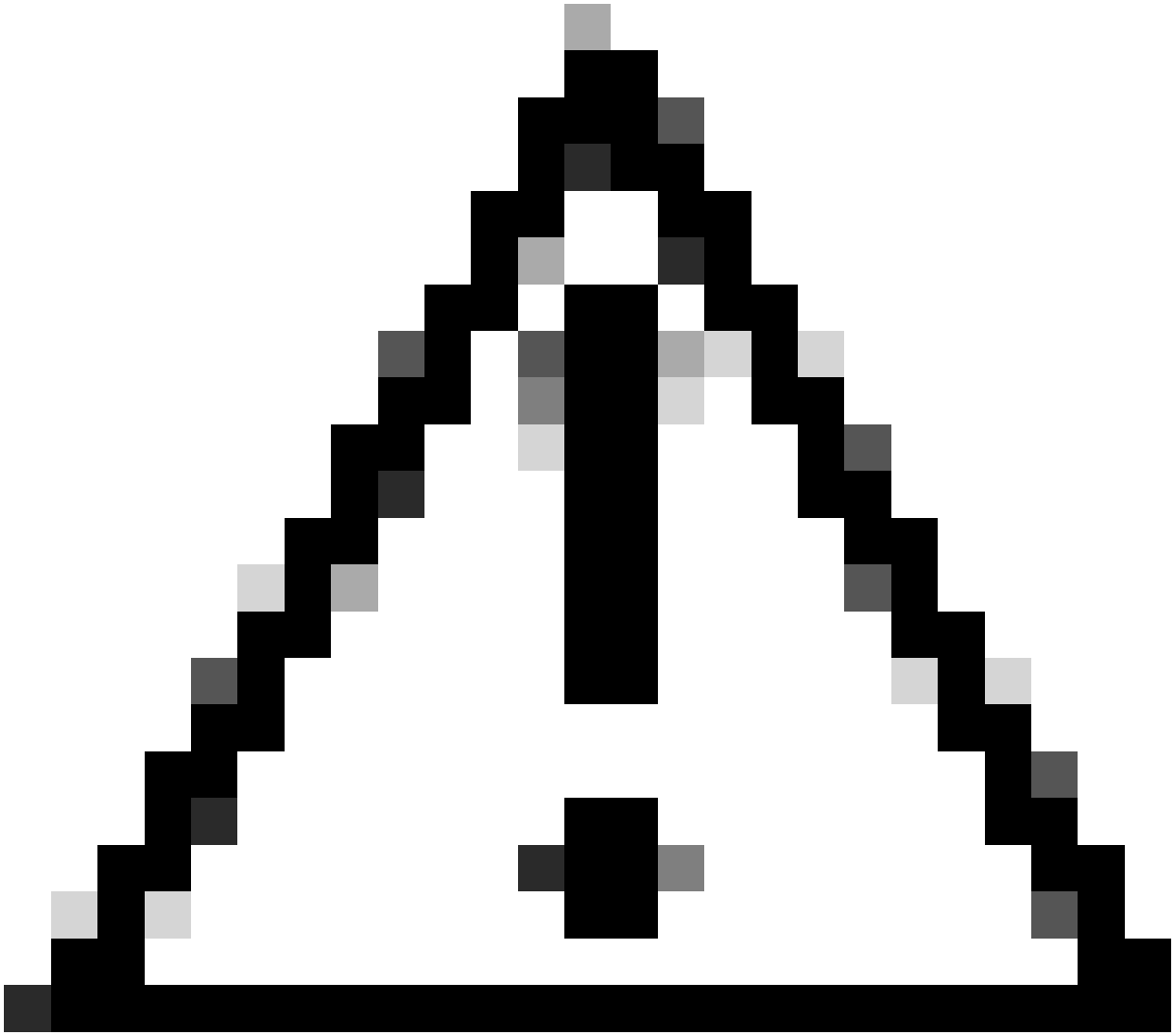
<#root>

```
Leaf01#

show platform dhcpsnooping client stats 0006.F601.CD43

DHCPSN: DHCP snooping server
DHCPD: DHCP protocol daemen
L2FWD: Transmit Packet to driver in L2 format
FWD: Transmit Packet to driver

(B): Dhcp message's response expected as 'B'roadcast
(U): Dhcp message's response expected as 'U'nicast

Packet Trace for client MAC 0006.F601.CD43:

Timestamp                Destination MAC  Destination Ip  VLAN  Message      Handler:Action
------------------------ ---------------- --------------- ----  ------------ ----------------
2023/09/28 14:53:59.866  FFFF.FFFF.FFFF   255.255.255.255 202  DHCPDISCOVER(B) PUNT:RECEIVED
2023/09/28 14:53:59.866  FFFF.FFFF.FFFF   255.255.255.255 202  DHCPDISCOVER(B) PUNT:TO_DHCPSN
2023/09/28 14:53:59.867  FFFF.FFFF.FFFF   255.255.255.255 202  DHCPDISCOVER(B) BRIDGE:RECEIVED
2023/09/28 14:53:59.867  0000.BEEF.CAFE   255.255.255.255 202  DHCPDISCOVER(B) L2INJECT:TO_FWD
2023/09/28 14:53:59.867  FFFF.FFFF.FFFF   255.255.255.255 202  DHCPDISCOVER(B) BRIDGE:TO_INJECT
2023/09/28 14:53:59.867  FFFF.FFFF.FFFF   255.255.255.255 202  DHCPDISCOVER(B) L2INJECT:TO_FWD
2023/09/28 14:54:01.871  0006.F601.CD43   255.255.255.255 202  DHCPOFFER(B)    PUNT:RECEIVED
2023/09/28 14:54:01.871  0006.F601.CD43   255.255.255.255 202  DHCPOFFER(B)    PUNT:TO_DHCPSN
2023/09/28 14:54:01.874  FFFF.FFFF.FFFF   255.255.255.255 202  DHCPREQUEST(B)  PUNT:RECEIVED
2023/09/28 14:54:01.874  FFFF.FFFF.FFFF   255.255.255.255 202  DHCPREQUEST(B)  PUNT:TO_DHCPSN
2023/09/28 14:54:01.874  FFFF.FFFF.FFFF   255.255.255.255 202  DHCPREQUEST(B)  BRIDGE:RECEIVED
2023/09/28 14:54:01.874  0000.BEEF.CAFE   255.255.255.255 202  DHCPREQUEST(B)  L2INJECT:TO_FWD
2023/09/28 14:54:01.874  FFFF.FFFF.FFFF   255.255.255.255 202  DHCPREQUEST(B)  BRIDGE:TO_INJECT
2023/09/28 14:54:01.874  FFFF.FFFF.FFFF   255.255.255.255 202  DHCPREQUEST(B)  L2INJECT:TO_FWD
2023/09/28 14:54:01.877  0006.F601.CD43   255.255.255.255 202  DHCPACK(B)      PUNT:RECEIVED
2023/09/28 14:54:01.877  0006.F601.CD43   255.255.255.255 202  DHCPACK(B)      PUNT:TO_DHCPSN
```

其他调试

```
debug ip dhcp server packet detail
debug ip dhcp server packet
```

```
debug ip dhcp server events
debug ip dhcp snooping packet
debug dhcp detail
```



警告：运行调试时务必小心！

相关信息

- 在Catalyst 9000系列交换机上实施BGP EVPN路由策略
- 在Catalyst 9000系列交换机上实施BGP EVPN保护的重叠分段
- 运行Catalyst 9000交换机上的DHCP监听并对其进行故障排除

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）。