

验证Nexus 9K FX/EX/GX上的MAC重写

目录

- [简介](#)
- [先决条件](#)
 - [要求](#)
 - [使用的组件](#)
- [拓扑](#)
- [背景](#)
- [检验MAC重写](#)
 - [检验下一跳MAC](#)
 - [检验路由表](#)
 - [验证Nexus转发决策 \(MAC重写 \)](#)
 - [验证下一个源MAC](#)
- [最终验证](#)

简介

本文档介绍如何在基于Tahoe的交换机上验证Nexus MAC重写。

先决条件

要求

Cisco 建议您了解以下主题：

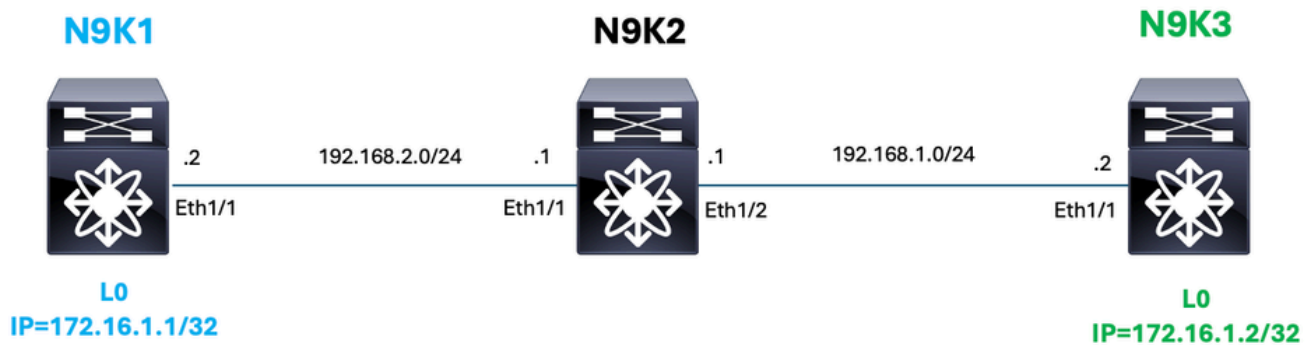
- NXOS平台
- 路由
- ELAM

使用的组件

名称	平台	version
N9K1	N9K-C93108TC-EX	9.3(10)
N9K2	N9K-C93108TC-EX	9.3(10)
N9K3	N9K-C93108TC-EX	9.3(10)

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。

拓扑



背景

在IP路由过程中，当数据包通过网络从源传输到目的地时，源和目的MAC地址会在每一跳更新，而源和目的IP地址保持不变。当数据包到达路由器时，路由器会检查目的IP地址，以确定通往最终目的地的路径上的下一跳。然后，路由器将数据包转发到下一跳，方法是将目的mac地址与下一跳接口的mac地址交换，并将源mac地址更新为其自己的传出接口mac地址。

此交换可确保数据包可以正确传送到本地网络中的下一台路由器或最终目的地。同时，源和目的IP地址在整个传输过程中保持不变，因为它们用于维护端到端路径并确保数据包通过可能的多个网段到达正确的目的地。

检验MAC重写

要验证基于Nexus 9K Tahoe的交换机上是否正确完成了mac重写，需要验证两个组件：下一跳mac地址和Nexus转发决策。

在本例中，Nexus正在接收具有所示源和目标mac的数据包，为了到达目标，将重写此mac:

</root>

```
N9K2# debug platform internal tah elam
N9K2(TAH-elam)# trigger init
Slot 1: param values: start asic 0, start slice 0, lu-a2d 1, in-select 6, out-select 0
N9K2(TAH-elam-insel6)# set outer ipv4 dst_ip 172.16.1.2
N9K2(TAH-elam-insel6)# start
N9K2(TAH-elam-insel6)# report
SUGARBOWL ELAM REPORT SUMMARY
slot - 1, asic - 0, slice - 0
=====
```

```
Incoming Interface: Eth1/1
Src Idx : 0x1, Src BD : 4100
Outgoing Interface Info: dmod 1, dpid 17
Dst Idx : 0x5, Dst BD : 4101
```

Packet Type: IPv4

Dst mac address:

70:0F:6A:95:1C:F9

Src mac address:

70:0F:6A:5E:6A:3F

Dst IPv4 address: 172.16.1.2

Src IPv4 address: 172.16.1.1

Ver = 4, DSCP = 0, Don't Fragment = 0

Proto = 1, TTL = 255, More Fragments = 0

Hdr len = 20, Pkt len = 84, Checksum = 0x9ebb

L4 Protocol : 1

ICMP type : 8

ICMP code : 0

检验下一跳MAC

在本文档中，为了验证下一跳MAC地址，可以检查路由和ARP表。

在本例中，我们将检查N9K2上如何重写目标172.16.1.2的mac地址。

检验路由表

```
N9K2# show ip route 172.16.1.2
```

```
<Snipped>
```

```
172.16.1.2/32, ubest/mbest: 1/0
```

```
*via 192.168.1.2, [1/0], 00:09:30, static
```

在找到作为出口接口的物理接口之前，需要检查通往下一跳的路由。

```
N9K2# show ip route 192.168.1.2
```

```
<Snipped>
```

```
192.168.1.2/32, ubest/mbest: 1/0, attached
```

```
*via 192.168.1.2, Eth1/2, [250/0], 00:12:23, am
```

一旦通过邻接管理器(am)发现并获知了连接的路由，您就可以继续检查下一跳的ARP条目

```
<#root>
```

```
N9K2# show ip arp 192.168.1.2
```

```
<Snipped>
```

```
IP ARP Table
```

```
Total number of entries: 1
```

Address	Age	mac Address	Interface	Flags
192.168.1.2	00:14:56			

700f.6a5e.6d5b

Ethernet1/2

700f.6a5e.6d5b 是Nexus必须用作路由重写的目标mac的mac。

验证Nexus转发决策 (MAC重写)

为了验证Nexus转发决策，需要捕获ELAM，具体在RW部分使用详细报告。

```
N9K2# debug platform internal tah elam
N9K2(TAH-elam)# trigger init
Slot 1: param values: start asic 0, start slice 0, lu-a2d 1, in-select 6, out-select 0
N9K2(TAH-elam-inse16)# set outer ipv4 dst_ip 172.16.1.2
N9K2(TAH-elam-inse16)# start
```

需要向目的地执行ping

```
N9K1# ping 172.16.1.2 source 172.16.1.1
PING 172.16.1.2 (172.16.1.2) from 172.16.1.1: 56 data bytes
64 bytes from 172.16.1.2: icmp_seq=0 ttl=253 time=0.906 ms
64 bytes from 172.16.1.2: icmp_seq=1 ttl=253 time=0.599 ms
64 bytes from 172.16.1.2: icmp_seq=2 ttl=253 time=0.589 ms
64 bytes from 172.16.1.2: icmp_seq=3 ttl=253 time=0.556 ms
64 bytes from 172.16.1.2: icmp_seq=4 ttl=253 time=0.55 ms

--- 172.16.1.2 ping statistics ---
5 packets transmitted, 5 packets received, 0.00% packet loss
round-trip min/avg/max = 0.55/0.64/0.906 ms
```

一旦数据包到达N9K2,Nexus必须查找(lu)到Tah L3邻接表以收集目标mac，Nexus必须将此mac置于重写(rw)中。

<#root>

N9K2

(TAH-elam-inse16)#

report detail

| i i

lurw

_vec.dst_addr

GBL_C++: [MSG] sug_lurw_vec.dst_addr:

0x700F6A5E6D5B

Nexus正在确定使用mac 700F6A5E6D5B作为数据包重写的目标mac地址，使用此信息可以确认Nexus正在执行正确的重写，即与从先前输出(70:0f:6a:5e:6d:5b)获取的目标mac地址匹配。

验证下一个源MAC

对于数据包的源mac地址，Nexus必须使用用于到达172.16.1.2的出口接口的mac地址，在本示例中，Nexus将使用接口以太网1/2 mac地址。

<#root>

```
N9K2# show interface ethernet 1/2 mac-address
```

```
-----  
Interface                mac-Address      Burn-in mac-Address  
-----  
Ethernet1/2  
  
700f.6a95.1cf9  
700f.6a95.1cfb
```

最终验证

验证可以在下一跳设备中完成，例如，可以在N9K3中运行ELAM

对于此示例以及之前的输出，Nexus应使用以下输出：

源mac:700f.6a95.1cf9

目标 MAC : 70:0f:6a:5e:6d:5b

<#root>

```
N9K3# debug platform internal tah elam  
N9K3(TAH-elam)# trigger init  
Slot 1: param values: start asic 0, start slice 0, lu-a2d 1, in-select 6, out-select 0  
N9K3(TAH-elam-inse16)# set outer ipv4 dst_ip 172.16.1.2  
N9K3(TAH-elam-inse16)# start  
N9K3(TAH-elam-inse16)# report  
SUGARBOWL ELAM REPORT SUMMARY  
slot - 1, asic - 0, slice - 0  
=====
```

```
Incoming Interface: Eth1/1  
Src Idx : 0x5, Src BD : 4101  
Outgoing Interface Info: dmod 0, dpid 0  
Dst Idx : 0x5bf, Dst BD : 4101
```

Packet Type: IPv4

Dst mac address:

70:0F:6A:5E:6D:5B

Src mac address:

70:0F:6A:95:1C:F9

Sup hit: 1, Sup Idx: 2788

Dst IPv4 address: 172.16.1.2

Src IPv4 address: 172.16.1.1

Ver = 4, DSCP = 0, Don't Fragment = 0

Proto = 1, TTL = 254, More Fragments = 0

Hdr len = 20, Pkt len = 84, Checksum = 0x9fc0

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）。