Verify MAC Rewrite on Nexus 9K FX/EX/GX

Contents

Introduction

This document describes how to verify Nexus MAC rewrite on Tahoe-based switches.

Prerequisites

Requirements

Cisco recommends that you have knowledge of these topics:

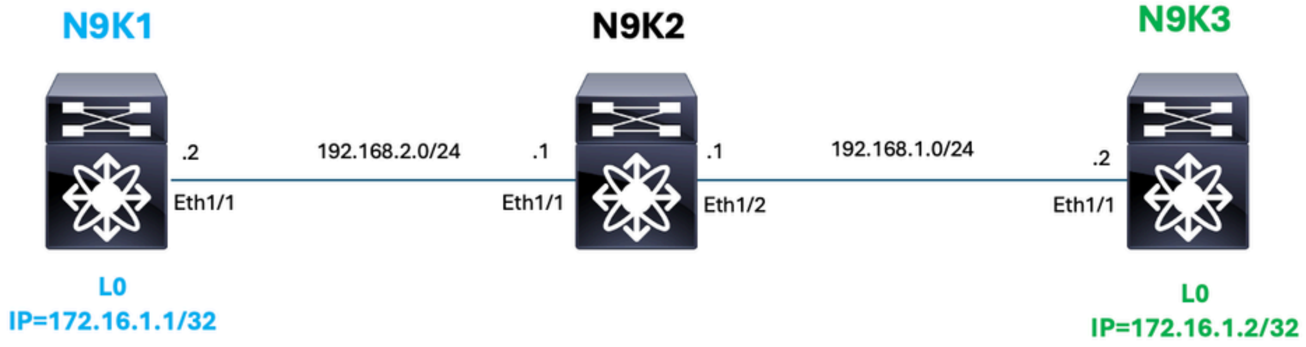
- NXOS Platform
- Routing
- ELAM

Components Used

Name	Paltform	Version
N9K1	N9K-C93108TC-EX	9.3(10)
N9K2	N9K-C93108TC-EX	9.3(10)
N9K3	N9K-C93108TC-EX	9.3(10)

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, ensure that you understand the potential impact of any command.

Topology



Background

In the process of IP routing, as packets traverse the network from the source to the destination, the source and destination mac addresses are updated at each hop, while the source and destination IP addresses remain unchanged. When a packet reaches a router, the router examines the destination IP address to determine the next hop on the path to its final destination. The router then forwards the packet to the next hop by swapping the destination mac address with the mac address of the next hops interface and updating the source mac address to its own outgoing interfaces mac address.

This swapping ensures that the packet can be correctly delivered to the next router or the final destination on the local network. Meanwhile, the source and destination IP addresses remain constant throughout the journey, as they are used to maintain the end-to-end path and ensure that the packet reaches the correct destination across potentially multiple network segments.

Verify MAC Rewrite

In order to verify if mac rewrite is done correctly on Nexus 9K Tahoe based switches, two components needs to be verified, next hop mac address and Nexus forwarding decision.

For this example, Nexus is be receiving packet with the showed source and destination macs, this **macs is be rewritten** in order to reach destination:

<#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
```

Verify Next-hop MAC

In this document in order to verify next hop mac address , routing and ARP tables can be checked.

For this example, on N9K2 is going to be examined on how its rewriting the mac address for destination **172.16.1.2**.

Verify Routing Table

```
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
```

Route towards next hop needs to be checked until a physical interface is found as egress interface.

```
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
```

Once an attached route is observed and learnt over Adjacency Manager (am), you can proceed to check ARP entry for that next hop

<#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 is the mac that Nexus must use as Destination mac for the routing rewrite.

Verify Nexus Forwarding Decision (MAC Rewrite)

In order to verify Nexus forwarding decision an ELAM capture needs to be taken, in specific the detailed report is used in the RW section.

```
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
```

A ping towards destination needs to be performed

```
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
```

Once packet reached N9K2, Nexus must do a look up (lu) into Tah L3 adjacency table to gather the destination mac, Nexus must place this mac in the rewrite (**rw**).

<#root>

```
N9K2

(TAH-elam-insel6)#

report detail

 | i i

lurw

_vec.dst_addr
GBL_C++:  [MSG]   sug_lurw_vec.dst_addr:

0x700F6A5E6D5B
```

Nexus is determining to use mac **700F6A5E6D5B** as destination mac address for the packet rewrite, with this information it can be confirmed that Nexus is performing the correct rewriting as is matching with the destination mac address taken from previous outputs (**70:0f:6a:5e:6d:5b**).

Verify Next Source MAC

For source mac address of the packet Nexus must use mac address of the egress interface that is using to reach 172.16.1.2, for this example Nexus is going to use interface ethernet 1/2 mac address.

<#root>

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

--------------------------------------------------------------------------------
Interface                  mac-Address     Burn-in mac-Address
--------------------------------------------------------------------------------
Ethernet1/2

700f.6a95.1cf9

  700f.6a95.1cfb
```

Final Verification

Verification can be done in the next hop device, for this example a ELAM can be run in N9K3

For this example with previous outputs it is expected for Nexus to use:

Source mac: **700f.6a95.1cf9**

Destination 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-insel6)# set outer ipv4 dst_ip 172.16.1.2
N9K3(TAH-elam-insel6)# start
N9K3(TAH-elam-insel6)# 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
```