Verify Nexus 9000 Series ARP and MAC Table Sync Behavior

Contents

Introduction

This document describes a specific synchronization behavior observed in the ARP and MAC address tables of Cisco Nexus 9000 series switches.

Prerequisites

Requirements

To fully benefit from the discussions in this document, the reader can have a foundational understanding of several key concepts and technologies:

1. Virtual Port Channel (vPC): Familiarity with the setup, configuration, and operational management of vPCs, as they are integral to understanding the networking scenarios described.
2. NX-OS Virtual Port Channel Peer Gateway Feature: Knowledge of how the peer gateway feature functions within a vPC setup, including its role in traffic forwarding and redundancy mechanisms.
3. Cisco Nexus Operating System (NX-OS): A working understanding of NX-OS, focusing on its command-line interface and typical configurations relevant to the Nexus 9000 series switches.

Components Used

- Switch Models: Nexus 3000 and Nexus 9000 series switches (First Generation only), which are central to illustrating the specific ARP and MAC table behavior due to their unique ASIC constraints.
- Virtual Port Channel (vPC): Configured to test synchronization behaviors across linked devices.
- vPC Peer-Gateway Feature: Activated within the vPC domain to investigate its influence on ARP and MAC synchronization.
- Non-vPC Layer 2 Trunk: Used to connect the Nexus peer devices.

- Non-vPC Switch Virtual Interfaces (SVIs): Configured to explore behaviors when user-defined MAC addresses are not used, highlighting the default handling of ARP and MAC address synchronization.
- Operating System: NX-OS version 7.0(3)I7(5).

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.

Background Information

In complex network environments, the synchronization of Address Resolution Protocol (ARP) and MAC address tables between interconnected devices is crucial for ensuring consistent data flow and network reliability. This guide aims to provide a comprehensive overview of these behaviors, supported by real-world lab observations and configurations, to aid in troubleshooting and configuring Nexus 9000 series switches effectively.

The ARP and MAC address synchronization issues detailed in this document are specific to certain configurations of Cisco Nexus 9000 series switches. These issues arise under two primary conditions:

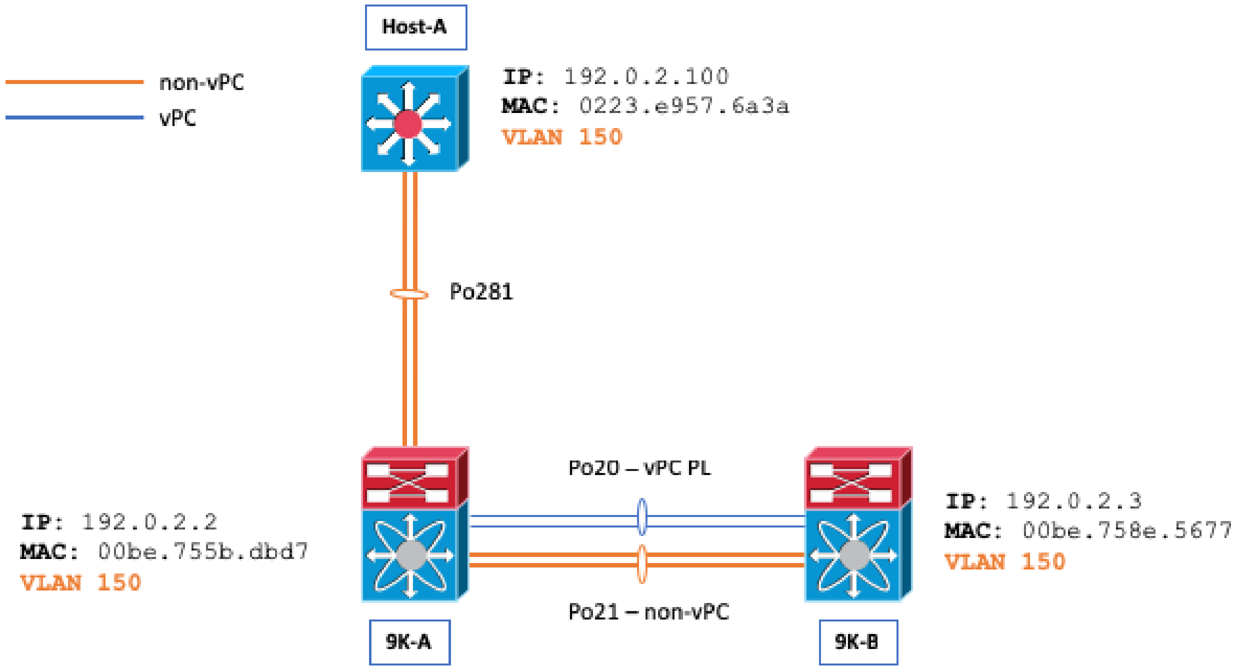
1. When Switch Virtual Interfaces (SVIs) are configured without user-defined MAC addresses.

2. When the Virtual Port Channel (vPC) peer-gateway feature is activated within the vPC domain settings.

This specific behavior is significant because it impacts how ARP entries are maintained despite corresponding MAC address entries potentially aging out or being explicitly cleared from the MAC address table. This can lead to inconsistencies in packet forwarding and network instability.

Furthermore, it is important to note that this behavior is due to an ASIC hardware limitation present only in the first-generation Nexus 9000 series switches. This limitation does not extend to the Nexus 9300 Cloud Scale models (EX, FX, GX, and C versions) introduced later. The issue has been recognized and cataloged under Cisco bug IDCSCuh94866.

Topology



Overview

Consider a network scenario where VLAN 150 is configured as a non-vPC VLAN, and both the ARP and MAC Address tables are initially empty between Host-A and Nexus 9000 switch B (N9K-B), and a ping is initiated from Host-A to N9K-B.

<#root>

```
Host-A#

ping 192.0.2.3

PING 192.0.2.3 (192.0.2.3): 56 data bytes
36 bytes from 192.0.2.100: Destination Host Unreachable
Request 0 timed out
64 bytes from 192.0.2.3: icmp_seq=1 ttl=254 time=1.011 ms
64 bytes from 192.0.2.3: icmp_seq=2 ttl=254 time=0.763 ms
64 bytes from 192.0.2.3: icmp_seq=3 ttl=254 time=0.698 ms
64 bytes from 192.0.2.3: icmp_seq=4 ttl=254 time=0.711 ms

--- 192.0.2.3 ping statistics ---
5 packets transmitted, 4 packets received, 20.00% packet loss
round-trip min/avg/max = 0.698/0.795/1.011 ms
```

This ping prompts Host-A to send an ARP request targeted at N9K-B. This request is broadcast out of port-channel 21 (Po21) on Nexus 9000 switch A (N9K-A), which is responsible for VLAN flooding. Simultaneously, the request is also tunneled via Cisco Fabric Services (CFS) across port-channel 20 (Po20). As a direct consequence, the MAC address table on N9K-B is updated to include the correct entry for Host-A, and an ARP entry is also established in the ARP table of N9K-B, pointing to Po21—the non-vPC Layer 2 trunk—as the interface for Host-A’s MAC address (0223.e957.6a3a).

```
<#root>

N9K-B#

show ip arp 192.0.2.100


Flags: * - Adjacencies learnt on non-active FHRP router
       + - Adjacencies synced via CFSoE
       # - Adjacencies Throttled for Glean
       CP - Added via L2RIB, Control plane Adjacencies
       PS - Added via L2RIB, Peer Sync
       RO - Re-Originated Peer Sync Entry
       D - Static Adjacencies attached to down interface

IP ARP Table
Total number of entries: 1
Address         Age       MAC Address     Interface       Flags
192.0.2.100   00:01:07  0223.e957.6a3a  Vlan150

N9K-B#

show mac address-table address  | i i 6a3a

*  150     0223.e957.6a3a   dynamic  0         F      F    Po21

N9K-B#

show ip arp detail | i 3a

192.0.2.100   00:03:22  0223.e957.6a3a  Vlan150

port-channel21

 <<<< Expected port-channel
```

The problem can be seen when the MAC Address of Host-A is removed from the MAC address table of N9K-B. This removal could occur for a variety of reasons, including the natural aging process of the MAC address, receipt of Spanning Tree Protocol (STP), Topology Change Notifications (TCNs), or manual interventions such as executing the **clearmac address-table dynamic** command.

```
<#root>

N9K-B#

show ip arp 192.0.2.100


Flags: * - Adjacencies learnt on non-active FHRP router
       + - Adjacencies synced via CFSoE
       # - Adjacencies Throttled for Glean
       CP - Added via L2RIB, Control plane Adjacencies
       PS - Added via L2RIB, Peer Sync
       RO - Re-Originated Peer Sync Entry
       D - Static Adjacencies attached to down interface

IP ARP Table
Total number of entries: 1
Address         Age       MAC Address     Interface       Flags
192.0.2.100   00:00:29     0223.e957.6a3a  Vlan150

<<< ARP remains populated
```

```
N9K-B#

show mac address-table address 0223.e957.6a3a

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, C - ControlPlane MAC, ~ - vsan
   VLAN     MAC Address      Type      age     Secure NTFY Ports
---------+-----------------+--------+---------+------+----+------------------

<empty, no MAC>

N9K-B#

ping 192.0.2.100

PING 192.0.2.100 (192.0.2.100): 56 data bytes
64 bytes from 192.0.2.100: icmp_seq=0 ttl=253 time=1.112 ms
64 bytes from 192.0.2.100: icmp_seq=1 ttl=253 time=0.647 ms
64 bytes from 192.0.2.100: icmp_seq=2 ttl=253 time=0.659 ms
64 bytes from 192.0.2.100: icmp_seq=3 ttl=253 time=0.634 ms
64 bytes from 192.0.2.100: icmp_seq=4 ttl=253 time=0.644 ms

--- 192.0.2.100 ping statistics ---
5 packets transmitted, 5 packets received, 0.00% packet loss
round-trip min/avg/max = 0.634/0.739/1.112 ms
```

Despite these deletions, it is noteworthy that ping traffic remains successful. However, the ARP entry for Host-A on N9K-B unexpectedly points to port-channel 20 (Po20—the vPC Peer Link), rather than port-channel 21 (Po21), which is the designated non-vPC Layer 2 trunk. This redirection occurs despite VLAN 150 being configured as a non-vPC VLAN, which leads to an inconsistency in expected traffic flow.

<#root>

```
N9K-B#

show ip arp detail | i i 6a3a


Flags: * - Adjacencies learnt on non-active FHRP router
       + - Adjacencies synced via CFSoE
       # - Adjacencies Throttled for Glean
       CP - Added via L2RIB, Control plane Adjacencies
       PS - Added via L2RIB, Peer Sync
       RO - Re-Originated Peer Sync Entry

IP ARP Table for context default
Total number of entries: 2
Address         Age       MAC Address     Interface        Physical Interface  Flags
192.0.2.100   00:15:54   0223.e957.6a3a    Vlan150          port-channel20

<<< Not Po21 once the issue is triggered.
```

You can use the **show ip arp internal event-history event** command on both Nexus 9000 switches to demonstrate that packets get tunneled via Cisco Fabric Services (CFS):

<#root>

```
N9K-B#

show ip arp internal event-history event | i i tunnel

    [116] [27772]: Tunnel Packets came with: vlan: 150, L2-SMAC :0223.e957.6a3a, L2-DMAC: 00be.758e.567
    [116] [27772]: Received tunneled packet on iod: Vlan150, physical iod: port-channel20

N9K-A#

show ip arp internal event-history event | i i tunnel

    [116] [28142]: Tunnel Packets sent with: vlan: 150, L2-SMAC :0223.e957.6a3a, L2-DMAC: 00be.758e.567
    [116] [28142]: Tunnel it to peer destined to remote SVI's Gateway MAC. Peer Gateway Enabled
```

You can also use the **debug ip arp** series of debug commands on 9K-B to detail this behavior as well:

<#root>

```
N9K-B#

debug logfile TAC_ARP


N9K-B#

debug ip arp packet


N9K-B#

debug ip arp event


N9K-B#

debug ip arp error



N9K-B#

show debug logfile TAC_ARP | beg "15:31:23"

2018 Oct 11 15:31:23.954433 arp: arp_send_request_internal: Our own address 192.0.2.3 on interface Vlan

2018 Oct 11 15:31:23.955221 arp: arp_process_receive_packet_msg: Received tunneled packet on iod: Vlan15
2018 Oct 11 15:31:23.955253 arp: arp_process_receive_packet_msg: Tunnel Packets came with: vlan: 150, L2
2018 Oct 11 15:31:23.955275 arp: (context 1) Receiving packet from Vlan150, logical interface Vlan150 ph
2018 Oct 11 15:31:23.955293 arp: Src 0223.e957.6a3a/192.0.2.100 Dst 00be.758e.5677/192.0.2.3


2018 Oct 11 15:31:23.955443 arp: arp_add_adj: arp_add_adj: Updating MAC on interface Vlan150, phy-inter
2018 Oct 11 15:31:23.955478 arp: arp_adj_update_state_get_action_on_add: Different MAC(0223.e957.6a3a)
2018 Oct 11 15:31:23.955576 arp: arp_add_adj: Entry added for 192.0.2.100, 0223.e957.6a3a, state 2 on i
2018 Oct 11 15:31:23.955601 arp: arp_add_adj: Adj info: iod: 77, phy-iod: 91, ip: 192.0.2.100, mac: 022
```

The ARP reply from Host-A first reaches Nexus 9000 switch A (N9K-A) and is then tunneled to Nexus 9000 switch B (N9K-B). Notably, N9K-A forwards the ARP reply to its control plane, leveraging the peer-gateway vPC domain enhancement. This configuration enables N9K-A to handle the routing of the packet for N9K-B, an operation typically not expected in a non-vPC VLAN setup.

```
<#root>

N9K-A#

ethanalyzer local interface inband display-filter arp limit-c 0


Capturing on inband
2018-10-11 15:32:47.378648 00:be:75:8e:56:77 -> ff:ff:ff:ff:ff:ff ARP Who has 192.0.2.100?  Tell 192.0.

<<<<


2018-10-11 15:32:47.379262 02:23:e9:57:6a:3a -> 00:be:75:8e:56:77 ARP 192.0.2.100 is at 02:23:e9:57:6a:
```

To validate the behavior of the ARP reply, the Ethanalyzer feature on NX-OS can be utilized. This tool confirms that the control plane of N9K-B does not directly observe this ARP reply, highlighting the specialized handling of ARP traffic in vPC configurations.

```
<#root>

N9K-B#

ethanalyzer local interface inband display-filter arp limit-c 0


Capturing on inband
2018-10-11 15:33:30.053239 00:be:75:8e:56:77 -> ff:ff:ff:ff:ff:ff ARP Who has 192.0.2.100?  Tell 192.0.
2018-10-11 15:34:16.817309 00:be:75:8e:56:77 -> ff:ff:ff:ff:ff:ff ARP Who has 192.0.2.100?  Tell 192.0.
2018-10-11 15:34:42.222965 00:be:75:8e:56:77 -> ff:ff:ff:ff:ff:ff ARP Who has 192.0.2.44?  Tell 192.0.2
<snip>
```

⚠ **Caution**: Depending on the sequence of events and circumstances, you could experience packet loss from N9K-B to Host-A.

```
<#root>

N9K-B#

ping 192.0.2.100

PING 192.0.2.100 (192.0.2.100): 56 data bytes
36 bytes from 192.0.2.3: Destination Host Unreachable
Request 0 timed out
Request 1 timed out
Request 2 timed out
Request 3 timed out
Request 4 timed out

--- 192.0.2.100 ping statistics ---
5 packets transmitted, 0 packets received, 100.00% packet loss
```

Conclusion and Workaround

The observed behavior, where ARP entries incorrectly reference the vPC peer link rather than the expected non-vPC trunk, typically occurs under specific circumstances: when User Defined MAC Addresses are not configured on non-vPC Switch Virtual Interfaces (SVIs), even if these SVIs are not used for routing adjacencies over a vPC.

This behavior only applies to First Generation Nexus 9000 switches.

To mitigate this issue, it is recommended to manually configure the MAC addresses for the impacted SVIs. Changing the MAC addresses can prevent the ARP misdirection from occurring, ensuring that the network functions as intended without relying on the vPC peer link in non-vPC scenarios.

Sample workaround below:

```
<#root>

N9K-A(config)#

interface Vlan150

N9K-A(config-if)#

mac-address 0000.aaaa.0030

N9K-A(config-if)#

end


N9K-B(config)#

interface Vlan150

N9K-B(config-if)#

mac-address 0000.bbbb.0030

N9K-B(config-if)#

end
```

Note: Due to a hardware limitation, you can only have 16 User Defined MAC Addresses Configured per device at a time. This is documented within the Cisco Nexus 9000 Series NX-OS Interfaces Configuration Guide as The switch has a limit of 16 user-defined MAC Addresses (MEv6/static).

Configuring beyond this limit can result in issues documented in Cisco bug ID CSCux84428

After the workaround is applied, the Ethanalyzer feature on NX-OS can be utilized to verify that Nexus 9000 switch A (N9K-A) no longer forwards the ARP Reply to its control plane, affirming the correct handling of ARP responses in the network.

<#root>

```
N9K-A#

ethanalyzer local interface inband display-filter arp limit-c 0


Capturing on inband
2018-10-11 15:36:11.675108 00:00:bb:bb:00:30 -> ff:ff:ff:ff:ff:ff ARP Who has 192.0.2.100?  Tell 192.0.2
```

Related Information

Reference the Create Topologies for Routing over Virtual Port Channel document for more information about Layer 2 non-vPC trunks, routing adjacencies, and SVI User Defined MAC requirements.