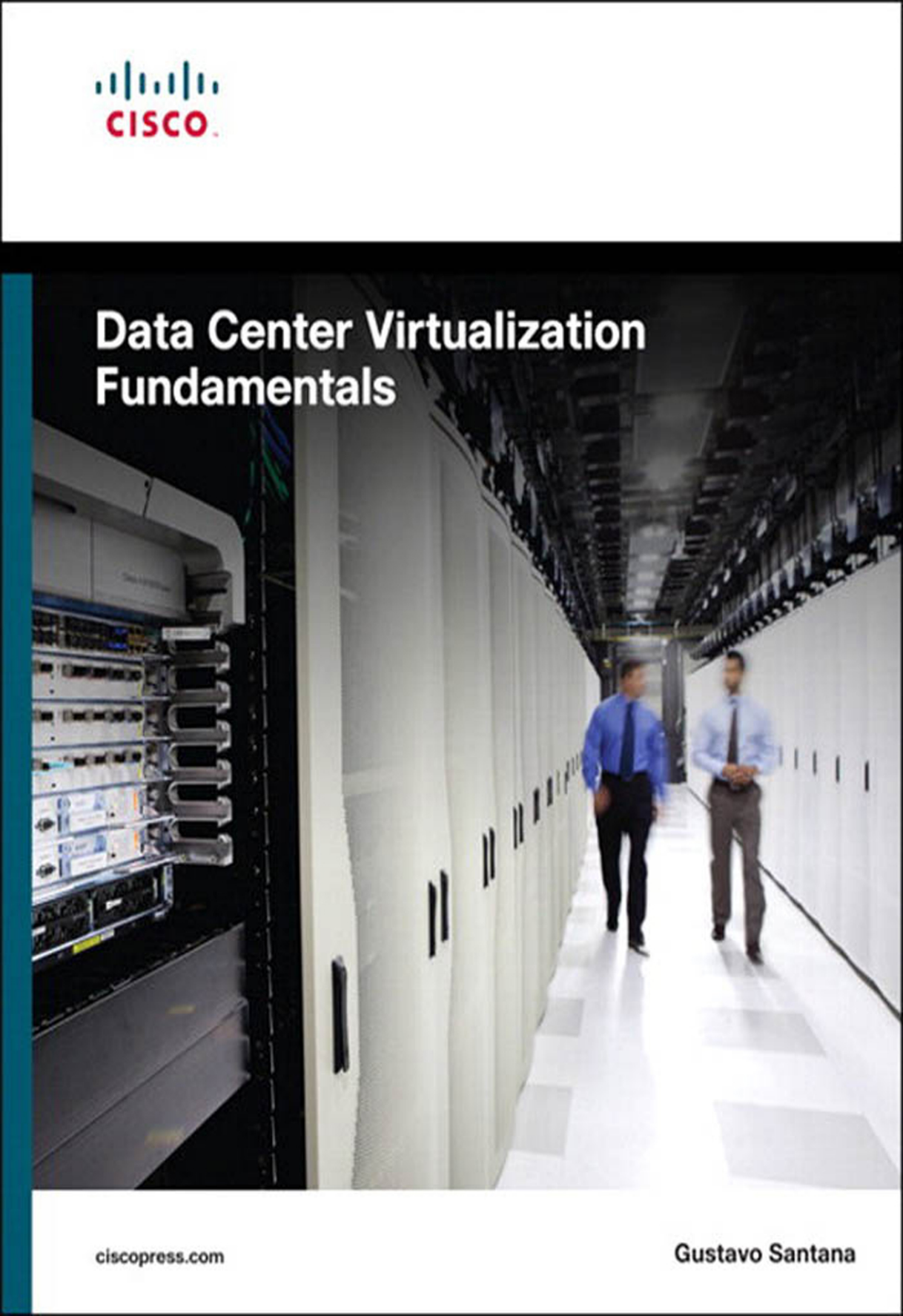
Data Center Virtualization Fundamentals
ciscopress.com
Gustavo Santana

Data Center Virtualization Fundamentals

Gustavo Alessandro Andrade Santana, CCIE No. 8806

Cisco Press

800 East 96th Street

Indianapolis, IN 46240

Note Because of the intense development in NX-OS, mapping enhancement features to Nexus platforms is almost like aiming at a moving target. Hence, I recommend that you verify whether the vPC features presented in this chapter are available in your hardware and software combination. My objective here is to explain the key concepts behind virtual PartChannels and not to present an outdated snapshot of Nexus capabilities.

Peer Link Failure and Orphan Ports

As explained in earlier section "Virtual PortChannels," you can expect the following consequences in the secondary vPC peer (N7K-2, in our topology) in a peer link failure:

- All the vPC ports are suspended.
- Orphan ports (Ethernet 1/11, for example) remain operational.

Example 6-15 illustrates this behavior after I have interrupted both connections in the peer link through a **shutdown** command executed in PortChannel 200 on N7K-1.

Example 6-15 *N7K-2 Port Status After Peer Link Failure*

```
! Verifying which interfaces were suspended after a peer-link failure
N7K-2# show interface brief | include vpc
Eth1/15   1   eth  trunk  down   suspended by vpc           auto(D) 10
Eth1/16   1   eth  trunk  down   suspended by vpc           auto(D) 10
Eth1/17   1   eth  trunk  down   suspended by vpc           auto(D) 10
Eth1/18   1   eth  trunk  down   suspended by vpc           auto(D) 10
Po10      1   eth  trunk  down   suspended by vpc            auto(D)  lacp
```

No surprises here: N7K-2 is still detecting that N7K-1 is active through the peer keepalive link; therefore, it will suspend every vPC port.

Note SVIs from VLANs present on vPC member ports are also suspended with the peer link failure.

Nevertheless, Ethernet 1/11 is still active, and at this moment, Orphan1 is completely isolated. And even if Orphan1 was deploying an active-standby NIC teaming policy, no failure would be detected in the active link.

To avoid such situations, it is possible to suspend the interface along with the vPC ports in a peer link failure scenario. The idea is that such failure will provoke at least a switchover to the standby interfaces of dual-homed servers.

The **vpc orphan-port suspend** command, configured in the Ethernet 1/11 on N7K-2, produces the result shown in Example 6-16.

Example 6-16 *N7K-2 Port Status After Another Peer Link and Orphan Port Configuration*

```
! Verifying which interfaces were suspended after a peer-link failure
N7K-2# show interface brief | include vpc
Eth1/11   1   eth  trunk  down    vpc peerlink is down        auto(D) --
Eth1/15   1   eth  trunk  down    suspended by vpc            auto(D) 10
Eth1/16   1   eth  trunk  down    suspended by vpc            auto(D) 10
Eth1/17   1   eth  trunk  down    suspended by vpc            auto(D) 10
Eth1/18   1   eth  trunk  down    suspended by vpc            auto(D) 10
Po10      1   eth  trunk  down    suspended by vpc             auto(D)  lacp
```

First-Hop Routing Protocols and Virtual PortChannels

Virtual PortChannel peers can also be configured as default gateways, as long as they both deploy Layer 3 capabilities. In truth, first-hop routing protocols (such as Hot Standby Router Protocol [HSRP] and Virtual Router Redundancy Protocol [VRRP]) can leverage the active-active behavior from vPC deployments in a way that was not possible with standard STP switches.

Figure 6-18 clarifies how both vPC peers are able to route IP packets directed to the virtual IP created by HSRP.

Note I had to enable **interface-vlan** and **hsrp** features before this configuration.

In Example 6-17, you can see that N7K-1 can route IP packets directed to the virtual default gateway.

Example 6-17 *N7K-1 MAC Addresses*

```
! Verifying N7K-1 HSRP role
N7K-1# show hsrp brief
                     P indicates configured to preempt.                     |
Interface  Grp Prio P State  Active addr  Standby addr Group addr
Vlan1      1   255  P Active local        10.1.1.2     10.1.1.254      (conf)
! Discovering the MAC address for VLAN1 SVI
N7K-1# show interface vlan 1 | include address
  Hardware is EtherSVI, address is  0026.980d.53c3
! Verifying HSRP Virtual MAC
N7K-1# show hsrp group 1 | include mac
  Virtual mac address is 0000.0c9f.f001 (Default MAC)
! Checking which MAC addresses can act as a Gateway in N7K-1
N7K-1# show mac address-table vlan 1
```

```
Legend: * - primary entry, G - Gateway MAC, (R) - Routed MAC, O - Overlay MAC
        age - seconds since last seen,+ - primary entry using vPC Peer-Link
 VLAN    MAC Address     Type     age    Secure NTFY Ports/SWID.SSID.LID
-------+---------------+------+-------+----+----+----------
G 1       0000.0c9f.f001  static     -      F    F  sup-eth1(R)
* 1       0026.980d.4343  static     -      F    F  vPC Peer-Link
G 1       0026.980d.53c3  static     -      F    F  sup-eth1(R)
[output suppressed]
```

```
N7K-1# show running-config interface vlan 1
[output suppressed]
interface Vlan1
  no shutdown
  no ip redirects
  ip address 10.1.1.1/24
  hsrp version 2
  hsrp 1
    preempt
    priority 255
    ip 10.1.1.254
```

```
N7K-2# show running-config interface vlan 1
[output suppressed]
interface Vlan1
  no shutdown
  noip redirects
  ip address 10.1.1.2/24
  hsrp version 2
  hsrp 1
    ip 10.1.1.254
```

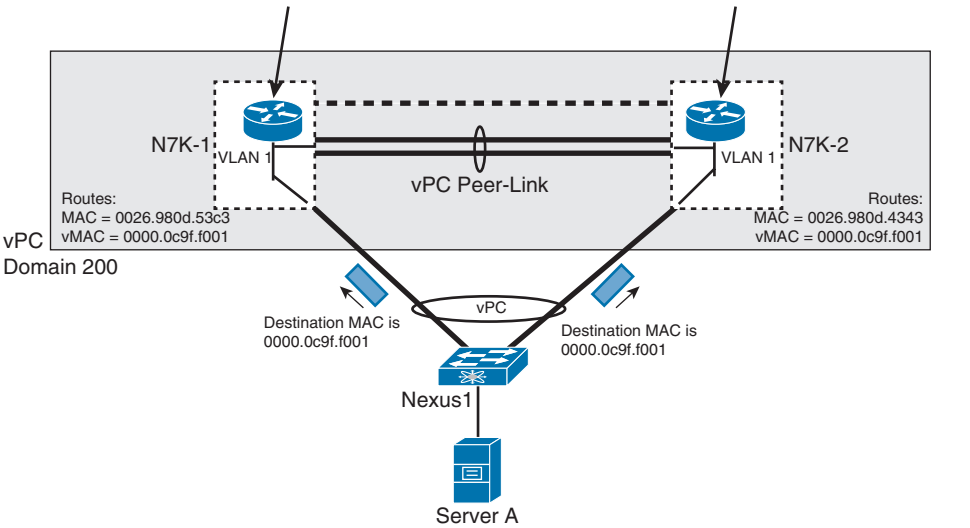

Figure 6-18 *HSRP and vPC*

From the output of the **show mac address-table** command, you can verify that N7K-1 can act as an IP gateway to frames directed to MAC addresses marked with the letter *G*, such as 0000.0c9f.f0001 (HSRP group virtual MAC) and 0026.980d.53c3 (VLAN 1 SVI MAC address in N7K-1).

In the same way, Example 6-18 illustrates that N7K-2 can act as a gateway to frames that are sent to MAC addresses 0000.0c9f.f0001 (HSRP group virtual MAC) and 0026.980d.4343 (VLAN 1 SVI MAC address in N7K-2).

Example 6-18 *N7K-2 MAC Addresses*

```
! Verifying N7K-2 HSRP role
N7K-2# show hsrp brief
                     P indicates configured to preempt.                     |
Interface  Grp Prio P State   Active addr Standby addr     Group addr
Vlan1      1   100    Standby 10.1.1.1    local            10.1.1.254      (conf)
! Discovering the MAC address for VLAN1 SVI
N7K-2# show interface vlan 1 | include address
  Hardware is EtherSVI, address is  0026.980d.4343
! Verifying HSRP Virtual MAC
N7K-2# show hsrp group 1 | include mac
  Virtual mac address is 0000.0c9f.f001 (Default MAC)
! Checking which MAC addresses can act as a Gateway in N7K-2
N7K-2# show mac address-table vlan 1
Legend: * - primary entry, G - Gateway MAC, (R) - Routed MAC, O - Overlay MAC
 age - seconds since last seen,+ - primary entry using vPC Peer-Link
 VLAN   MAC Address    Type    age   Secure NTFY Ports/SWID.SSID.LID
-------+---------------+------+-------+----+----+----------
G 1      0000.0c9f.f001  static     -     F  F  sup-eth1(R)
G 1      0026.980d.4343  static     -     F  F  sup-eth1(R)
* 1      0026.980d.53c3  static     -     F  F  vPC Peer-Link
[output suppressed]
```

Both Nexus switches can route packets to the HSRP virtual MAC address! As a consequence, the upstream routed traffic will be load balanced through the PortChannel hash algorithm in Nexus1.

However, some host devices deploy nonstandard behavior within their multiple Ethernet connections. For example, some Network Attached Storage (NAS), and servers with particular NIC teaming techniques, might not send any Address Resolution Protocol (ARP) frames to discover MAC addresses; they simply use the MAC address received on the original request to build its response. If such an NAS had its multiple interfaces connected to Nexus1 (from our topology), the replies to external clients would be directed to the *SVI MAC addresses* and not the HSRP MAC address.

This characteristic can generate the undesired vPC behavior described as follows:

- The NAS sends an IP packet directed to MAC address 0026.980d.53c3 (VLAN 1 SVI MAC address in N7K-1) using an interface connected to N7K-2 (because of a PortChannel hash decision).
- N7K-2 switches this frame to N7K-1 using its MAC address table.
- N7K-1 can *block* this packet if it was supposed to be forwarded out to a vPC.

Do you remember the vPC check rule? Some NX-OS switches (Nexus 7000 equipped with M1-series modules, for example) apply this rule for switched and *routed* traffic, causing packet drops when submitted to nonstandard traffic like the one I am describing now.

The *peer gateway* feature was designed to solve this awkward scenario. If both vPC peers are configured with this feature, each one of them will be able to route packets that are directed to its peer MAC address.

Figure 6-19 and Example 6-19 show that with peer gateway enabled, N7K-2 will be able to route the packet sent to the N7K-1 SVI MAC address.

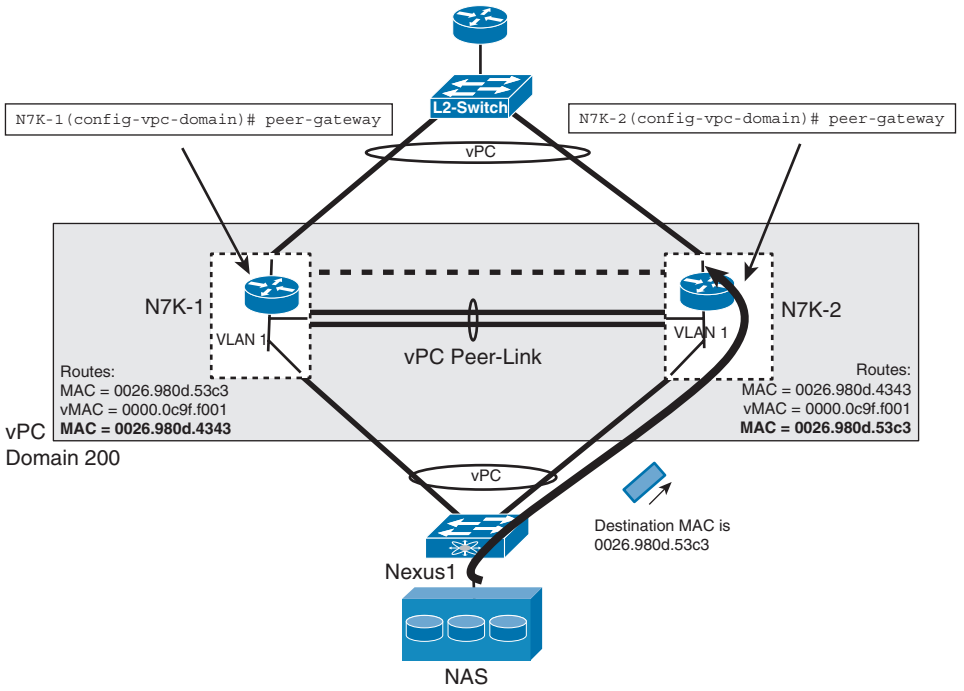

Figure 6-19 *Peer Gateway Configured in N7K-1 and N7K-2*

Example 6-19 *MAC Addresses in N7K-2 After Peer Gateway Is Enabled*

```
! Checking which MAC addresses can now act as a Gateway in N7K-2
N7K-2# show mac address-table vlan 1 | include "G"
        * - primary entry, G - Gateway MAC, (R) - Routed MAC, O - Overlay MAC
G 1         0000.0c9f.f001    static       -       F    F  sup-eth1(R)
G 1         0026.980d.4343    static       -       F    F  sup-eth1(R)
G 1         0026.980d.53c3    static       -       F    F  sup-eth1(R)
```

Of course, the same behavior is expected on N7K-1 regarding frames directed to the N7K-2 SVI MAC address.

Note As previously mentioned, I recommend that you consult the Cisco online documentation to verify how your specific NX-OS device behaves with routed unicast and multicast IP traffic on a vPC.

Routing Protocols and Virtual PortChannelsVirtual PortChannels are a Layer 2 multihop technology whose path decision depends on a hashing algorithm. On the other hand, when a Layer 3 device must choose a gateway among a set of equal-cost possible routers, another hash algorithm is used as well.

When applied to the same device, both independent algorithms can generate unexpected results in some vPC scenarios. For example, consider this hypothetical topology:

- Router1 is connected to our N7K-1 and N7K-2 switches using a vPC. Router1 is sending a packet to a remote subnet connected to another router (Router2).
- Router2 is also using a vPC created by N7K-1 and N7K-2.
- Router1 can use either N7K-1 or N7K-2 as gateways to the remote subnet. This is an equal-cost multipath decision that, for example, could result in a frame with N7K-1 MAC address as its destination.
- However, because Router1 is also using a vPC, the frame directed to the N7K-1 Layer 2 address can use an N7K-2–connected interface.
- N7K-2 switches the frame to N7K-1 SVI through the peer link.
- N7K-1 receives the frame and routes it to the vPC connected to Router2.
- Depending on its hardware characteristics, N7K-1 can apply the vPC check rule to this frame and block it.

This is not a problem of the vPC technology, but simply a design incompatibility between independent decision processes.

Similar to the NAS scenario presented in the previous section, this behavior can be fixed with the peer gateway feature. This capability can avoid the vPC check in N7K-1, because the traffic would be routed in N7K-2.

Although the vPC peer gateway capability can be a good solution to static routing in this case, the feature does not help when a routing protocol is deployed between a vPC-connected router and the vPC peers.

As shown in Figure 6-20, some routing protocols use packets with TTL of one that can be dropped by a vPC peer switch with peer gateway enabled. Hence, the routing protocol communication between the Layer 3 switch and N7K-1, passing through N7K-2 (Layer 2 hash decision), would not happen.