Q-in-VNI over VXLAN Fabric Deployment Guide

Cisco Confidential

The documentation set for this product strives to use bias-free language. For the purposes of this documentation set, bias-free is defined as language that does not imply discrimination based on age, disability, gender, racial identity, ethnic identity, sexual orientation, socioeconomic status, and intersectionality. Exceptions may be present in the documentation due to language that is hardcoded in the user interfaces of the product software, language used based on RFP documentation, or language that is used by a referenced third-party product.

Summary

In traditional network segmentation, the number of segments that can be created is limited to 4096 VLANs. This can be insufficient in environments, such as cloud data centers, where multiple tenants or customers may require their own virtual networks. VXLAN solves the problem of limited network segmentation and isolation in traditional VLAN-based networks.

Q-in-VNI over VXLAN addresses the requirement of limited network segmentation and isolation by stacking multiple VLANs above each other, allowing for even more virtual network segmentation and isolation. This provides a large number of virtual networks to be created, with the necessary flexibility and scalability for cloud computing environments and other situations where many virtual networks are required.

In summary, Q-in-VNI deployed using VXLAN EVPN as the transport network provides a highly efficient and scalable solution for delivering Layer 2 VPN services. It allows users to retain their existing VLAN structure while connecting to the service provider's network and provides high availability and redundancy with VXLAN EVPN's robust control plane.

Introduction to Q-in-Q

When a service provider offers services, each user’s network segmentation is preserved using Q-in-Q, also known as " double tagging" or " stacking VLANs”. This is a technique used to extend VLANs (Virtual LANs) across a transport network. It allows multiple customer VLANs to be tagged with a provider vlan, allowing service providers to offer Layer 2 VPN services to their customers. This allows customers to maintain their existing vlan structure while connecting to the service provider's network.

In a Q-in-Q deployment, customer frames are tagged with two vlan headers:

- A customer VLAN tag (C-tag), which identifies the customer’s VLAN within their local domain.
- A provider VLAN tag (S-tag), which identifies the provider’s VLAN used to transport the customer traffic across the provider network.

This approach enables service providers to transparently carry multiple customers’ VLANs over a shared backbone, maintaining isolation and improving scalability and security.

tags: a customer-facing vlan tag and a provider-facing vlan tag. The customer-facing vlan tag is used to identify the customer's vlan, while the provider-facing vlan tag is used to identify the provider's vlan. This allows the service provider to maintain the customer's vlan structure while connecting the customer's network to the provider's network. It can also be used to provide additional security by isolating customer VLANs from each other.

© 2023-25 Cisco and/or its affiliates. All rights reserved.

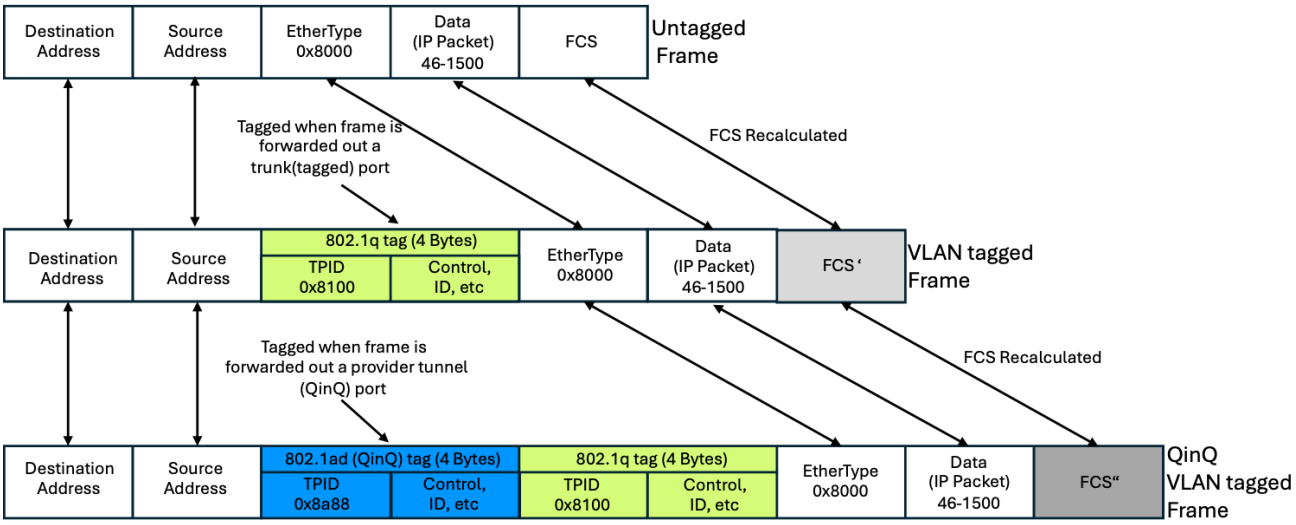


Figure 1. Q-in-Q packet header

Introduction to VXLAN (Virtual extensible LAN)

VXLAN (Virtual Extensible LAN) is a network virtualization technology that extends Layer 2 connections across a Layer 3 network. It uses a VXLAN header, which encapsulates the original Layer-2 frames, allowing them to be transmitted over a Layer 3 network. VXLAN ID (also known as the VXLAN Network Identifier or VNI) is a 24-bit identifier used to uniquely identify a virtual network.

The VXLAN ID in the header serves as the basis for building the VXLAN segment It is used to map the virtual network to a specific VXLAN segment, and to identify the destination of the traffic in the VXLAN network. All the endpoints that are part of the same VXLAN segment have the same VXLAN ID.

VXLAN addresses scalability limitations of traditional VLANs by allowing a large number of VLANs to be created. While traditional VLANs are limited to 4094, VXLANs can support up to 16 million unique VXLAN IDs (VNIs). This allows for a large number of virtual networks to be created, making it well suited for large-scale data center deployments.

This segmentation approach enables multi-tenancy by allowing different virtual networks to be isolated from each other, even if they are using the same physical infrastructure. This can be useful in a cloud computing environment, where different tenants may require their own virtual networks. Additionally, it can be useful where the network needs to be extended across multiple geographic locations, such as in disaster recovery scenarios.

© 2023-25 Cisco and/or its affiliates. All rights reserved.

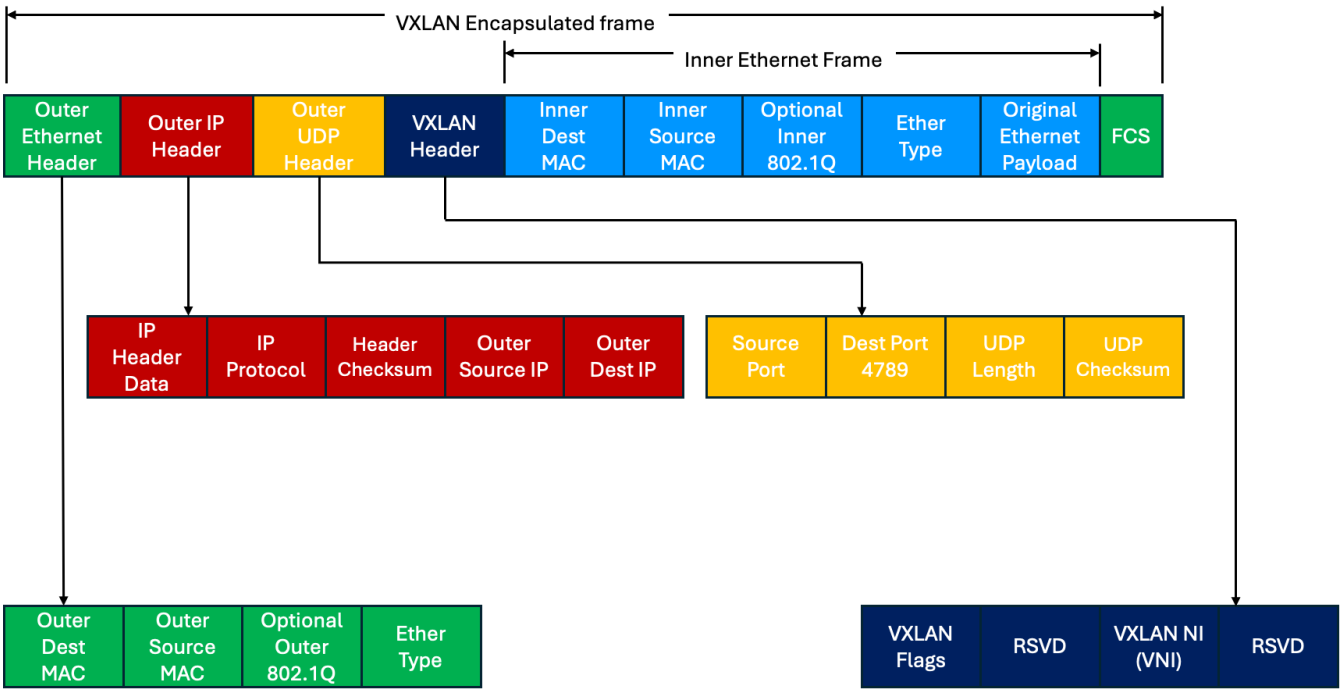


Figure 2. VXLAN Packet Header

Introduction to EVPN (Ethernet Virtual Private Network)

EVPN (Ethernet Virtual Private Network) is a control-plane technology that allows for the automatic discovery and configuration of virtual private networks (VPNs) using Multi-Protocol BGP (MP-BGP) routing protocol. It provides a way to exchange Layer 2 and Layer 3 information. The transport can be either VXLAN or MPLS. In this case study, VXLAN is used as a transport mechanism for the fabric infrastructure.

EVPN provides several benefits over traditional Layer 2 VPN technologies, including:

- Multi-homing support – EVPN allows for multiple links to be used for a single VPN, providing redundancy and high availability.
- Layer 3 and Layer 2 VPN services – EVPN supports both Layer 3 and Layer 2 VPN services, making it more versatile than traditional Layer 2 VPN technologies.
- Automatic discovery and configuration – EVPN uses a control-plane to automatically discover and configure VPNs, reducing the need for manual configuration and improving scalability.
- Interoperability – EVPN is designed to work with existing Layer 2 and Layer 3 protocols, making it easier to integrate into existing networks.
- EVPN can be used in different scenarios, such as data center interconnect, WAN connectivity, and service provider infrastructure. It can also be used in conjunction with VXLAN, which is known as VXLAN EVPN. This combination provides a way to extend Layer 2 connections across a Layer 3 network, while also providing the control-plane functionality using EVPN route types. EVPN route types are used to carry different types of information over the network. EVPN route types are carried in MP-BGP (Multiprotocol Border Gateway Protocol) updates and are used by the network's routing and switching equipment to make forwarding decisions.

© 2023-25 Cisco and/or its affiliates. All rights reserved.

EVPN Route types
Type-1 : Ethernet A-D Route
Type-2 : MAC Advertisement Route
Type-3 : Inclusive Multicast Route
Type -4 : Ethernet Segment Route
Type-5: IP prefix route

Figure 3. EVPN Route Types

Q-in-VNI with EVPN and VXLAN as transport

A Telco service provider VXLAN fabric infrastructure is CLOS-based architecture with a Leaf and Spine layer switches. All the leaf nodes will be connected to all the spine switches but the spine switches will not be connected to each other. Each leaf node or node pair will act as VXLAN Tunnel Endpoint (VTEP). The traffic entering the fabric will have a fixed latency with 3 hop to reach the end service.

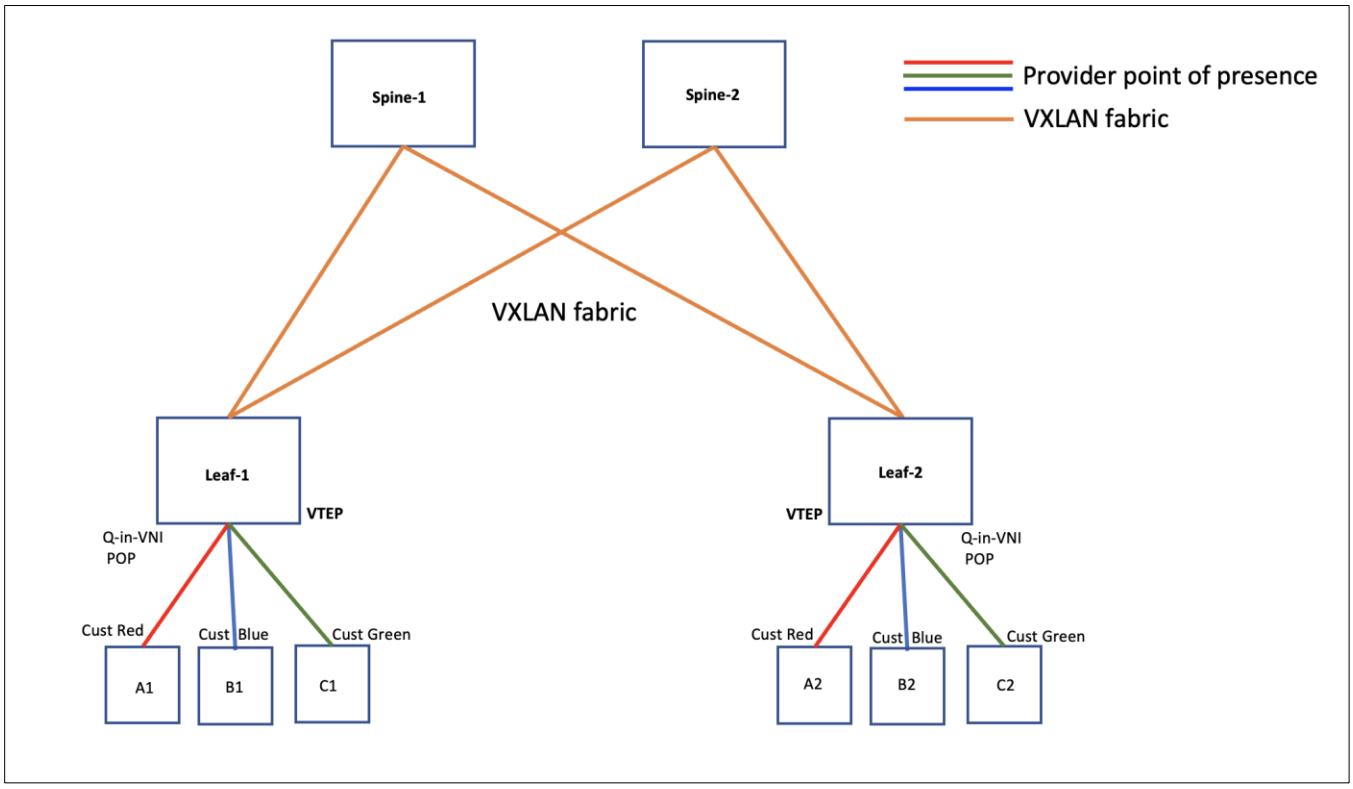


Figure 4. VXLAN topology

In the above figure the Service provider VXLAN fabric consists of 2 Leaf and 2 Spine Switches. The leaf switches are the point of presence for the service provider offering Q-in-Q services to customer Red, Blue and Green.

In this case study,

© 2023-25 Cisco and/or its affiliates. All rights reserved.

- Leaf1 and Leaf2 loopback addresses are advertised using OSPF routing protocol. The other recommended option is to use IS-IS as the Interior gateway protocol(IGP).
- On both Leaf switches, Loopback 1 is source interface for VXLAN NVE interface.
- BGP EVPN is configured between the Leaf and Spine nodes.
- Spine nodes act as the route reflector (RR) for the EVPN sessions.
- Customers Red, Blue, and green are configured with a unique Provider-vlan mapped to a unique layer 2 VNI across the VXLAN fabric.
- On the local leaf node, every customer endpoint (MAC) received on an interface configured with a Provider-vlan, is learned and advertised using BGP EVPN as type-2 route (MAC-Route).
- This type-2 route will also contain layer 2 VNI and the next hop information which is the VTEP IP address of the local leaf.

VXLAN EVPN fabric configuration sample shown below is from Leaf and Spine switches. Note that EVPN configuration under BGP when not explicitly configured, the route target for every VNI, import a,nd export will be set to auto by default.

Note: Partial configuration is shown below with few VLANs and VNIs but a few hundred VLANs and VNIs are used as part of this case study.

Table 1. VXLAN Configuration sample

LEAF-1 Configuration	LEAF-2 Configuration	SPINE-1 Configuration
vlan Configuration vlan 1300 vn-segment 13000 vlan 1301 vn-segment 13001 vlan 1302 vn-segment 13002 **BGP Configuration** router bgp 65001 router-id 10.102.0.3 neighbor 10.102.0.1 remote-as 65001 update-source loopback0 address-family l2vpn evpn send-community send-community extended evpn **OSPF Configuration** router ospf UNDER router-id 10.102.0.3 interface loopback0 ip router ospf UNDER area 0.0.0.0 interface loopback1 ip router ospf UNDER area 0.0.0.0 interface Ethernet1/49 ip ospf network point-to-point ip router ospf UNDER area 0.0.0.0 **VXLAN Configuration** interface nve1 switch(config-if)# no shutdown host-reachability protocol bgp source-interface loopback1	**vlan Configuration** vlan 1300 vn-segment 13000 vlan 1301 vn-segment 13001 vlan 1302 vn-segment 13002 **BGP Configuration** router bgp 65001 router-id 10.102.0.2 neighbor 10.102.0.1 remote-as 65001 update-source loopback0 address-family l2vpn evpn send-community send-community extended evpn **OSPF Configuration** router ospf UNDER router-id 10.102.0.2 interface loopback0 ip router ospf UNDER area 0.0.0.0 interface loopback1 ip router ospf UNDER area 0.0.0.0 interface Ethernet1/49 ip ospf network point-to-point ip router ospf UNDER area 0.0.0.0 **VXLAN Configuration** interface nve1 switch(config-if)# no shutdown host-reachability protocol bgp source-interface loopback1	**BGP Configuration** router bgp 65001 router-id 10.102.0.1 neighbor 10.102.0.2 remote-as 65001 update-source loopback0 address-family l2vpn evpn send-community send-community extended route-reflector-client neighbor 10.102.0.3 remote-as 65001 update-source loopback0 address-family l2vpn evpn send-community send-community extended route-reflector-client **OSPF Configuration** router ospf UNDER router-id 10.102.0.1 interface loopback0 ip router ospf UNDER area 0.0.0.0 interface loopback254 ip router ospf UNDER area 0.0.0.0 interface Ethernet1/5 ip ospf network point-to-point ip router ospf UNDER area 0.0.0.0 interface Ethernet1/6 ip ospf network point-to-point ip router ospf UNDER area

© 2023-25 Cisco and/or its affiliates. All rights reserved.

member vni 13000
  mcast-group 239.1.1.1
member vni 13001
  mcast-group 239.1.1.1
member vni 13002
  mcast-group 239.1.1.1
	member vni 13000
	  mcast-group 239.1.1.1
	member vni 13001
	  mcast-group 239.1.1.1
	member vni 13002
	  mcast-group 239.1.1.1
		0.0.0.0

The subsequent topics will cover various techniques used by service providers using Q-in-VNI over a VXLAN fabric as transport to provide VPN services.

Q-in-VNI

Q-in-VNI allows you to categorize network traffic by assigning it to a specific port. In a multi-tenant setup, you can designate a port to a tenant/customer and transmit/receive data using the VXLAN overlay. The provider interface is configured as an access port that uses 'switchport mode dot1q tunnel', which receives a dot1q tagged packet from the customer switch. The VXLAN encapsulated packet holds the customer's VLAN tag in its inner frame's L2 header.

Interface configuration:

```
switch(config-if)# interface Ethernet1/51
switch(config-if)# description Q-in-VNI for Customer Red
switch(config-if)# switchport
switch(config-if)# switchport mode dot1q-tunnel
switch(config-if)# switchport access vlan 300
switch(config-if)# spanning-tree port type edge
switch(config-if)# spanning-tree bpdufilter enable
switch(config-if)# mtu 9216
switch(config-if)# no shutdown

switch(config-if)# interface Ethernet1/52
switch(config-if)# description Q-in-VNI for Customer Blue
switch(config-if)# switchport
switch(config-if)# switchport mode dot1q-tunnel
switch(config-if)# switchport access vlan 301
switch(config-if)# spanning-tree port type edge
switch(config-if)# spanning-tree bpdufilter enable
switch(config-if)# mtu 9216
switch(config-if)# no shutdown

switch(config)# vlan 300
switch(config-vlan)# vn-segment 33000
switch(config)# vlan 301
switch(config-vlan)# vn-segment 33001
```

© 2023-25 Cisco and/or its affiliates. All rights reserved.

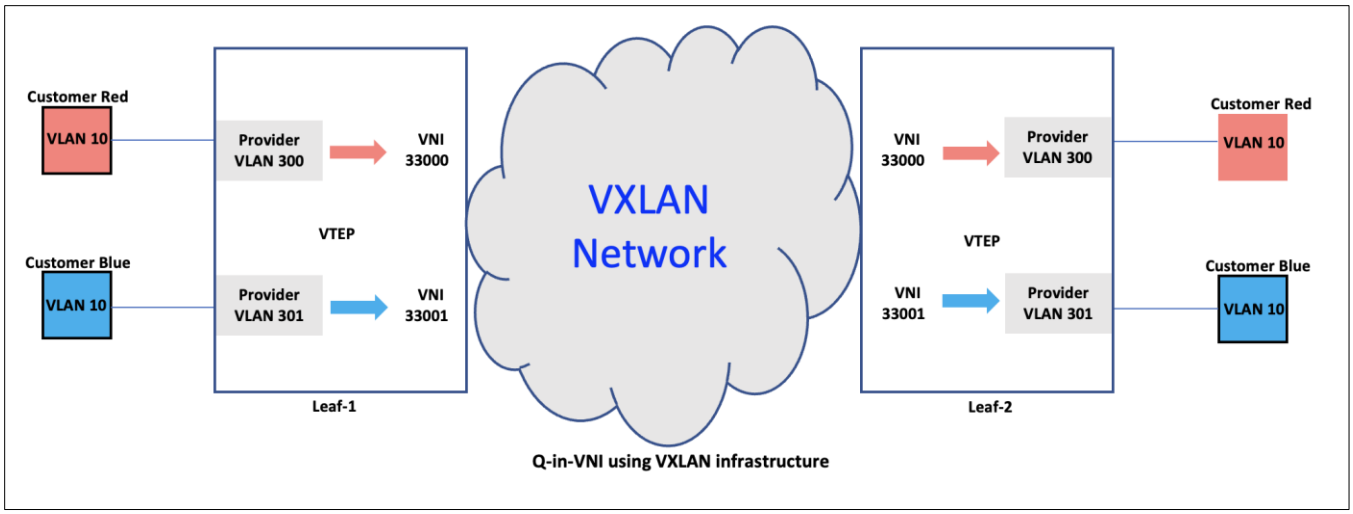


Figure 5. Q-in-VNI model

In the above figure, the user Red's traffic is forwarded over a VXLAN overlay network, which uses VXLAN headers to encapsulate the original dot1q frame on VLAN 10. This allows the traffic to be transmitted over a layer-3 network, while preserving the original layer-2 information. The VXLAN header includes information such as the Source VTEP and Destination VTEP, as well as the layer-2 VNI, which is used to forward the traffic to its destination. When the traffic reaches the LEAF-2 switch, the VXLAN header is removed (decapsulated), and the original VLAN 10 frame is sent out on the other side. The user's original frame is preserved throughout the journey, ensuring that the traffic is delivered to its intended destination intact.

Like user Red, user Blue communicates over VLAN 10, but it is configured as a Q-in-VNI port with provider VLAN 301 and a unique layer-2 VNI of 33001. The traffic is forwarded to the correct Q-in-VNI port-based on the matching provider VLAN 301. Despite having the same customer VLAN, each customer has a unique layer-2 VNI to ensure proper traffic forwarding and thereby maintaining multi tenancy.

A packet capture in the VXLAN fabric is shown below:

- SRC VTEP address 10.102.1.2 is LEAF-1 lo1 address (source-interface of NVE1 interface)
- DST VTEP address 10.102.1.1 is LEAF-2 lo1 address (source-interface of NVE1 interface)
- VXLAN packet UDP header DST Port is 4789
- Layer-2 VNI for Provider vlan 300 is 33000
- Encapsulated packet showing customer Red original packet with VLAN ID 10
- Capture also shows inner packet's SRC IP 10.1.1.2 and DST IP address 10.1.1.1

```
    [Protocols in frame: eth:ethertype:ip:udp:VXLAN:eth:ethertype:vlan:ethertype:ip:data]
Ethernet II, Src: a0:b4:39:0b:9e:0b, Dst: 00:fd:22:8d:1b:75
    Destination: 00:fd:22:8d:1b:75
        Address: 00:fd:22:8d:1b:75
    Source: a0:b4:39:0b:9e:0b
        Address: a0:b4:39:0b:9e:0b
    Type: IPv4 (0x0800)
```

© 2023-25 Cisco and/or its affiliates. All rights reserved.

```
Internet Protocol Version 4, Src: 10.102.1.2, Dst: 10.102.1.1
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
    Differentiated Services Field: 0xc0 (DSCP: CS6, ECN: Not-ECT)
        1100 00.. = Differentiated Services Codepoint: Class Selector 6 (48)
        .... ..00 = Explicit Congestion Notification: Not ECN-Capable Transport (0)
    Flags: 0x0000
        Fragment offset: 0
    Time to live: 255
    Protocol: UDP (17)
    Source: 10.102.1.2
    Destination: 10.102.1.1
User Datagram Protocol, Src Port: 13894, Dst Port: 4789
    Source Port: 13894
    Destination Port: 4789
Virtual eXtensible Local Area Network
    Flags: 0x0800, VXLAN Network ID (VNI)
        0... .... .... .... = GBP Extension: Not defined
        .... .... .0.. .... = Don't Learn: False
        .... 1... .... .... = VXLAN Network ID (VNI): True
        .... .... .... 0... = Policy Applied: False
        .000 .000 0.00 .000 = Reserved(R): 0x0000
    Group Policy ID: 0
    VXLAN Network Identifier (VNI): 33000
    Reserved: 0
Ethernet II, Src: 00:45:50:01:00:01, Dst: 00:85:54:01:01:01
    Destination: 00:85:54:01:01:01
        Address: 00:85:54:01:01:01
    Source: 00:45:50:01:00:01
        Address: 00:45:50:01:00:01
    Type: 802.1Q Virtual LAN (0x8100)
802.1Q Virtual LAN, PRI: 7, DEI: 0, ID: 10
    111. .... .... .... = Priority: Network Control (7)
    ...0 .... .... .... = DEI: Ineligible
    .... 0000 0000 1010 = ID: 10
    Type: IPv4 (0x0800)
Internet Protocol Version 4, Src: 10.1.1.2, Dst: 10.1.1.1
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
    Differentiated Services Field: 0xc0 (DSCP: CS6, ECN: Not-ECT)
        1100 00.. = Differentiated Services Codepoint: Class Selector 6 (48)
        .... ..00 = Explicit Congestion Notification: Not ECN-Capable Transport (0)
```

© 2023-25 Cisco and/or its affiliates. All rights reserved.

```
Time to live: 255
 Protocol: Unknown (253)
 Header checksum: 0xdab1 [validation disabled]
 [Header checksum status: Unverified]
 Source: 10.1.1.2
 Destination: 10.1.1.1
```

Selective Q-in-VNI with multiple provider VLANs

Selective Q-in-VNI is a VXLAN tunneling feature that allows a user-specific VLAN or range of user VLANs on a port to be associated with one specific provider VLAN. Packets that come in with a VLAN tag that matches any of the configured user VLANs on the port are tunneled across the VXLAN fabric using the properties of the service provider VNI mapped to the provider VLAN in the configuration. The VXLAN encapsulated packet carries the user VLAN tag as part of the L2 header of the inner frame.

The packets with a VLAN tag that is not present in the range of the configured user VLANs on a selective Q-in-VNI configured port are dropped. This includes the packets with a VLAN tag that matches the native VLAN on the port. A range of customer VLANs can also be selectively mapped to a single provider VLAN as shown in the configuration sample below.

Note: See limitations for native VLAN while configuring selective Q-in-VNI given in the reference section at the end of the document.

Interface configuration sample:

```
switch# config terminal
switch(config-if)# interface Ethernet1/52
switch(config-if)# description Selective Q-in-VNI with multiple provider Vlans
switch(config-if)# switchport
switch(config-if)# switchport mode trunk
switch(config-if)# switchport vlan mapping 10 dot1q-tunnel 100
switch(config-if)# switchport vlan mapping 11 dot1q-tunnel 101
switch(config-if)# switchport vlan mapping 12 dot1q-tunnel 102
.
..
...
switch(config-if)# switchport vlan mapping 58 dot1q-tunnel 148
switch(config-if)# switchport vlan mapping 59 dot1q-tunnel 149
switch(config-if)# switchport vlan mapping 800-820 dot1q-tunnel 150
switch(config-if)# switchport vlan mapping 821-840 dot1q-tunnel 151
switch(config-if)# switchport trunk allowed vlan 100-149,150,151
switch(config-if)# spanning-tree port type edge trunk
switch(config-if)# spanning-tree bpdufilter enable
switch(config-if)#  mtu 9216
switch(config-if)# no shutdown
```

© 2023-25 Cisco and/or its affiliates. All rights reserved.

```
switch# config terminal
switch(config) vlan 100
switch(config-vlan) vn-segment 30000
switch(config) vlan 101
switch(config-vlan) vn-segment 30001
switch(config) vlan 102
switch(config-vlan) vn-segment 30002
.
..
...
switch(config) vlan 120
switch(config-vlan) vn-segment 30020
switch(config) vlan 121
switch(config-vlan) vn-segment 30021
switch(config) vlan 122
switch(config-vlan) vn-segment 30022
```

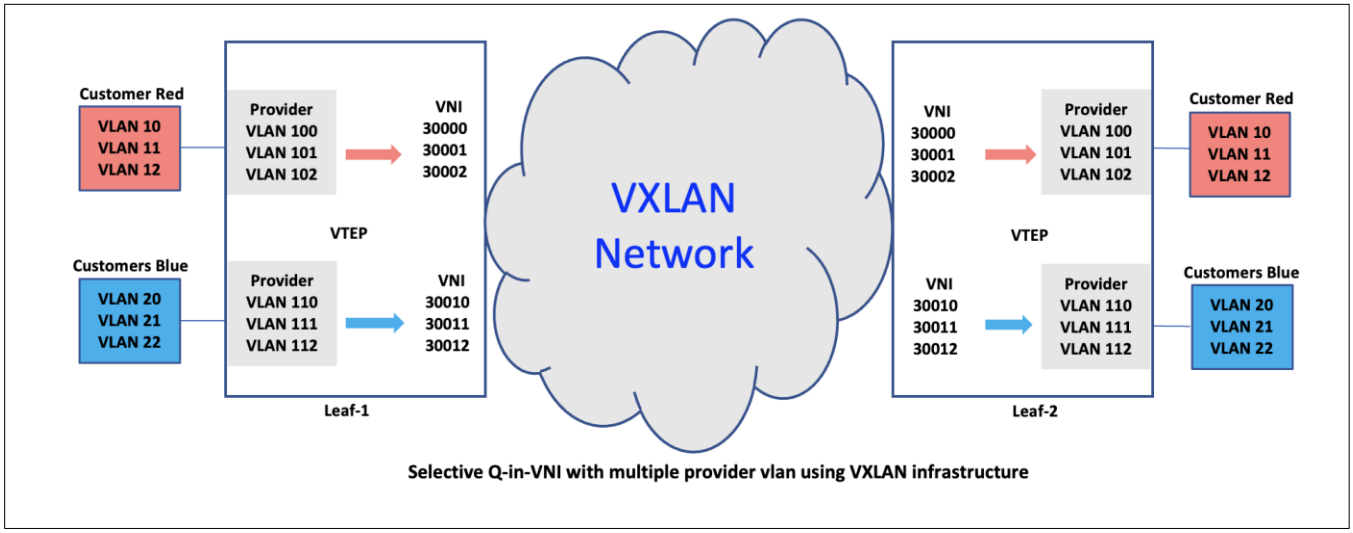


Figure 6. Selective Q-in-VNI model

In the above figure, the service provider switch is set up to map user VLAN 10 to provider VLAN 100, ensuring that traffic is forwarded to the correct destination. The VXLAN header includes the layer-2 VNI, in this case is 30000, which is used to route traffic over the network with the source VTEP Leaf-1 and destination VTEP Leaf-2. The encapsulation and decapsulation process is performed at each VTEP, ensuring that the original user frame remains intact as it travels over the VXLAN fabric. At the LEAF-2 switch, the VXLAN header is removed, and the inner frame (VLAN 10) is sent out on the egress side, facing the user. This allows the user Red's original frame to be preserved and delivered without any modifications, ensuring seamless and reliable communication over the network.

Similarly, user VLAN 11 is mapped to provider VLAN 101 with Layer-2 VNI 30001 and user VLAN 20 to provider VLAN 110 with Layer-2 VNI 30010, based on the configuration of the interface. This selective mapping of user VLANs to Q-in-VNI maintains multi-tenancy over the VXLAN network. In summary, each port on the service provider's switch can accommodate multiple customer VLANs, each connected to a

© 2023-25 Cisco and/or its affiliates. All rights reserved.

distinct provider VLAN with a unique layer-2 VNI. This directs traffic to the correct Q-in-VNI port, making efficient network communication.

Packet capture in the VXLAN fabric is shown below,

- SRC VTEP address 10.102.1.2 is LEAF-1 lo1 address (source-interface of NVE1 interface)
- DST VTEP address 10.102.1.1 is LEAF-2 lo1 address (source-interface of NVE1 interface)
- VXLAN packet UDP header DST Port is 4789
- Layer-2 VNI for Provider vlan 30002, which provider VLAN 102
- Encapsulated packet showing customer Red inner frame with VLAN ID 12
- Inner frames seen with SRC IP 102.1.1.2 and DST IP address 102.1.1.1

```
   [Protocols in frame: eth:ethertype:ip:udp:VXLAN:eth:ethertype:vlan:ethertype:ip:data]
Ethernet II, Src: a0:b4:39:0b:9e:0b, Dst: 00:fd:22:8d:1b:75
    Destination: 00:fd:22:8d:1b:75
        Address: 00:fd:22:8d:1b:75
    Source: a0:b4:39:0b:9e:0b
        Address: a0:b4:39:0b:9e:0b
    Type: IPv4 (0x0800)
Internet Protocol Version 4, Src: 10.102.1.2, Dst: 10.102.1.1
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
    Differentiated Services Field: 0xc0 (DSCP: CS6, ECN: Not-ECT)
        1100 00.. = Differentiated Services Codepoint: Class Selector 6 (48)
        .... ..00 = Explicit Congestion Notification: Not ECN-Capable Transport (0)
  Flags: 0x0000
        Fragment offset: 0
    Time to live: 255
    Protocol: UDP (17)
    Source: 10.102.1.2
    Destination: 10.102.1.1
User Datagram Protocol, Src Port: 21413, Dst Port: 4789
    Source Port: 21413
    Destination Port: 4789
Virtual eXtensible Local Area Network
    Flags: 0x0800, VXLAN Network ID (VNI)
        0... .... .... .... = GBP Extension: Not defined
        .... .... .0.. .... = Don't Learn: False
        .... 1... .... .... = VXLAN Network ID (VNI): True
        .... .... .... 0... = Policy Applied: False
        .000 .000 0.00 .000 = Reserved(R): 0x0000
    Group Policy ID: 0
```

© 2023-25 Cisco and/or its affiliates. All rights reserved.

```
    VXLAN Network Identifier (VNI): 30002
    Reserved: 0
Ethernet II, Src: 00:51:54:02:01:01, Dst: 00:85:56:02:01:02
    Destination: 00:85:56:02:01:02
        Address: 00:85:56:02:01:02
    Source: 00:51:54:02:01:01
        Address: 00:51:54:02:01:01
    Type: 802.1Q Virtual LAN (0x8100)
802.1Q Virtual LAN, PRI: 7, DEI: 0, ID: 12
    111. .... .... .... = Priority: Network Control (7)
    ...0 .... .... .... = DEI: Ineligible
    .... 0000 0000 1100 = ID: 12
    Type: IPv4 (0x0800)
Internet Protocol Version 4, Src: 102.1.1.2, Dst: 102.1.1.1
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
    Differentiated Services Field: 0xc0 (DSCP: CS6, ECN: Not-ECT)
        1100 00.. = Differentiated Services Codepoint: Class Selector 6 (48)
        .... ..00 = Explicit Congestion Notification: Not ECN-Capable Transport (0)
    Total Length: 490
    Identification: 0x8774 (34676)
    Flags: 0x0000
        Fragment offset: 0
    Time to live: 255
     Protocol: Unknown (253)
    Source: 102.1.1.2
    Destination: 102.1.1.1
```

VLAN mapping

VLAN mapping is the method of connecting a user VLAN to a service provider VLAN. In certain situations, VLAN translation may be necessary, such as when multiple users connect to the same physical switch using the same VLAN encapsulation but should not present on the same layer 2 segments. In these cases, the incoming VLAN is translated to a unique VLAN in the VXLAN fabric which is mapped to a VNI to extend the segment.

In this case study, the user VLAN 1000 is mapped to the provider VLAN 1300 on the interface. The provider VLAN 1300 has a layer-2 VNI of 13000.

Interface configuration showing multiple user VLANs mapped to translated provider VLANs:

```
switch# config terminal
switch(config-if)# interface Ethernet1/53
switch(config-if)# description Port Vlan Mapping
switch(config-if)# switchport
```

© 2023-25 Cisco and/or its affiliates. All rights reserved.

```
switch(config-if)# switchport mode trunk
switch(config-if)# switchport vlan mapping enable
switch(config-if)# switchport vlan mapping 1000 1300
switch(config-if)# switchport vlan mapping 1001 1301
switch(config-if)# switchport vlan mapping 1002 1302
switch(config-if)# switchport vlan mapping 1003 1303
switch(config-if)# switchport vlan mapping 1004 1304
.
..
...
switch(config-if)# switchport vlan mapping 1194 1494
switch(config-if)# switchport vlan mapping 1195 1495
switch(config-if)# switchport vlan mapping 1196 1496
switch(config-if)# switchport vlan mapping 1197 1497
switch(config-if)# switchport vlan mapping 1198 1498
switch(config-if)# switchport vlan mapping 1199 1499
switch(config-if)# switchport trunk allowed vlan 1300-1499
switch(config-if)# no shutdown

switch# config terminal
switch(config) vlan 1300
switch(config-vlan) vn-segment 13000
switch(config) vlan 1301
switch(config-vlan) vn-segment 13001
switch(config) vlan 1302
switch(config-vlan) vn-segment 13002
switch(config) vlan 1303
switch(config-vlan) vn-segment 13003
switch(config) vlan 1310
.
..
...
switch(config-vlan) vn-segment 13010
switch(config) vlan 1311
switch(config-vlan) vn-segment 13011
switch(config) vlan 1312
switch(config-vlan) vn-segment 13012
```

© 2023-25 Cisco and/or its affiliates. All rights reserved.

Vlan mapping with one Customer Tag

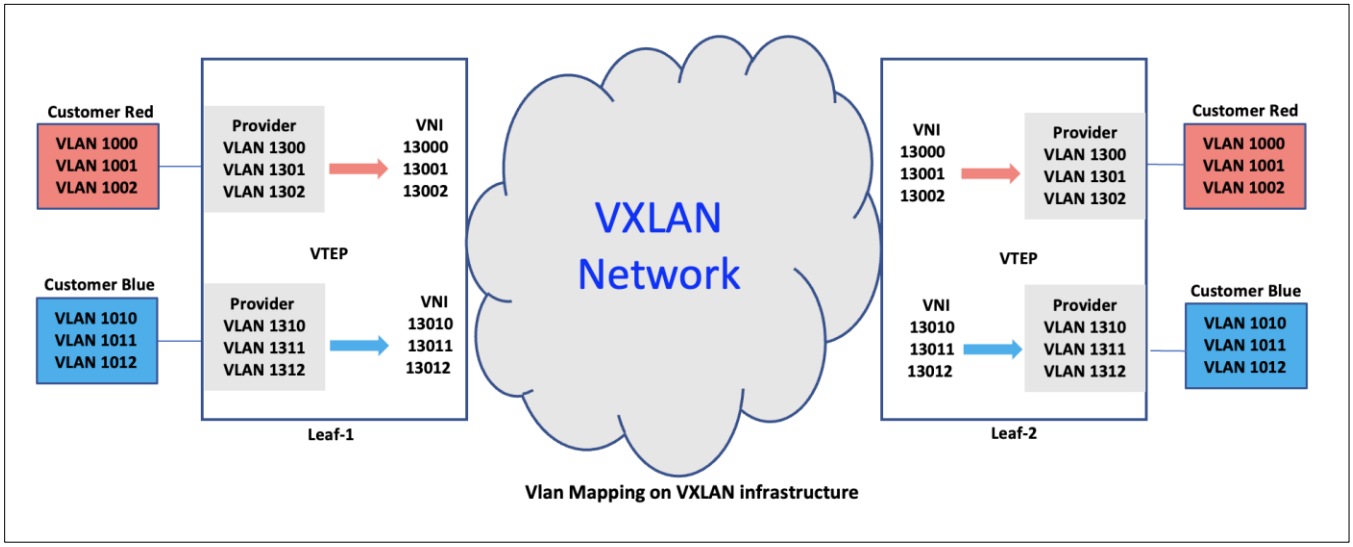


Figure 7. VLAN mapping on VXLAN infrastructure

In the above figure, the user VLANs 1000, 1001, 1002, and so on are mapped to provider VLANs 1300, 1301, 1302, respectively. The user is connected to a service provider switch set up as a VLAN-mapped Q-in-VNI port. User VLAN 1000 is mapped to provider VLAN 1300 which is configured with a layer-2 VNI of 13000.

When traffic on VLAN 1000 reaches the interface, it is encapsulated with a VXLAN header that includes layer-2 VNI 13000. This is because the interface is set up to translate user VLAN 1000 to provider VLAN 1300. The VXLAN header has a source VTEP address of the LEAF-1 switch and a destination VTEP address of the LEAF-2 switch. The packet is then sent to LEAF-2, where the VXLAN header is removed and the inner frame (on VLAN 1000) is sent out on the egress side facing the user.

This process ensures that the user VLAN 1000 traffic is preserved and transmitted securely between the LEAF-1 and LEAF-2 switches through the VXLAN fabric. The VXLAN header adds an extra layer of abstraction to the original user VLAN 1000 frame, allowing it to traverse the network and reach its destination, where the header is then removed, and the original frame is delivered to the user. This VXLAN-based approach enables efficient and secure communication between user VLANs and provider VLANs while preserving the user's original VLAN information.

A packet capture in the VXLAN fabric is shown below,

- SRC VTEP address 10.102.1.2 is LEAF-1 lo1 address (source-interface of NVE1 interface)
- DST VTEP address 10.102.1.1 is LEAF-2 lo1 address (source-interface of NVE1 interface)
- VXLAN packet UDP header DST Port is 4789
- Note that vlan 1000 which is translated is not seen packet capture
- Provider VLAN 1300, which is translated VLANs Layer-2 VNI 13000 seen
- Inner frames SRC IP 10.100.1.2 and DST IP address 10.100.1.1

```
    [Protocols in frame: eth:ethertype:ip:udp:VXLAN:eth:ethertype:ip:data]
Ethernet II, Src: a0:b4:39:0b:9e:0b, Dst: 00:fd:22:8d:1b:75
    Destination: 00:fd:22:8d:1b:75
```

© 2023-25 Cisco and/or its affiliates. All rights reserved.

```
        Address: 00:fd:22:8d:1b:75
    Source: a0:b4:39:0b:9e:0b
        Address: a0:b4:39:0b:9e:0b
    Type: IPv4 (0x0800)
Internet Protocol Version 4, Src: 10.102.1.2, Dst: 10.102.1.1
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
    Differentiated Services Field: 0xc0 (DSCP: CS6, ECN: Not-ECT)
        1100 00.. = Differentiated Services Codepoint: Class Selector 6 (48)
        .... ..00 = Explicit Congestion Notification: Not ECN-Capable Transport (0)
  Flags: 0x0000
     Fragment offset: 0
    Time to live: 255
    Protocol: UDP (17)
    Source: 10.102.1.2
    Destination: 10.102.1.1
User Datagram Protocol, Src Port: 42297, Dst Port: 4789
    Source Port: 42297
    Destination Port: 4789
tual eXtensible Local Area Network
    Flags: 0x0800, VXLAN Network ID (VNI)
        0... .... .... .... = GBP Extension: Not defined
        .... .... .0.. .... = Don't Learn: False
        .... 1... .... .... = VXLAN Network ID (VNI): True
        .... .... .... 0... = Policy Applied: False
        .000 .000 0.00 .000 = Reserved(R): 0x0000
    Group Policy ID: 0
    VXLAN Network Identifier (VNI): 13000
    Reserved: 0
Ethernet II, Src: 00:55:53:03:03:b9, Dst: 00:85:54:03:02:f1
    Destination: 00:85:54:03:02:f1
        Address: 00:85:54:03:02:f1
    Source: 00:55:53:03:03:b9
        Address: 00:55:53:03:03:b9
    Type: IPv4 (0x0800)
Internet Protocol Version 4, Src: 10.100.1.2, Dst: 10.100.1.1
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
    Differentiated Services Field: 0xc0 (DSCP: CS6, ECN: Not-ECT)
        1100 00.. = Differentiated Services Codepoint: Class Selector 6 (48)
        .... ..00 = Explicit Congestion Notification: Not ECN-Capable Transport (0)
    Total Length: 490
```

© 2023-25 Cisco and/or its affiliates. All rights reserved.

```
    Identification: 0x6f4f (28495)
    Flags: 0x0000
     Fragment offset: 0
    Time to live: 255
    Protocol: Unknown (253)
    Source: 10.100.1.2
    Destination: 10.100.1.1
```

Q-in-VNI Vlan mapping with two Customer Tag

In the packet capture shown below, user traffic was simulated with two C-Tags (i.e. two VLAN tags). The inner tag is VLAN 600, and the outer tag is VLAN 1000. According to the configuration, outer VLAN 1000 is mapped to provider VLAN 1300.

Packet capture in the VXLAN fabric is shown below:

- SRC VTEP address 10.102.1.2 is LEAF-1 lo1 address (source-interface of NVE1 interface)
- DST VTEP address 10.102.1.1 is LEAF-2 lo1 address (source-interface of NVE1 interface)
- VXLAN packet UDP header DST Port is 4789
- Note that VLAN 1000 which is translated is not seen packet capture
- Provider VLAN 1300, which is translated VLANs Layer-2 VNI 13000 seen
- Second user VLAN 600 is preserved inside the VXLAN packet.
- Inner frames seen with SRC IP 60.1.1.2 and DST IP address 60.1.1.1

```
Frame 1: 554 bytes on wire (4432 bits), 554 bytes captured (4432 bits) on interface ps-inb,
id 0
    [Protocols in frame: eth:ethertype:ip:udp:VXLAN:eth:ethertype:vlan:ethertype:ip:data]
Ethernet II, Src: a0:b4:39:0b:9e:0b, Dst: 00:fd:22:8d:1b:75
    Destination: 00:fd:22:8d:1b:75
        Address: 00:fd:22:8d:1b:75
    Source: a0:b4:39:0b:9e:0b
        Address: a0:b4:39:0b:9e:0b
    Type: IPv4 (0x0800)
Internet Protocol Version 4, Src: 10.102.1.2, Dst: 10.102.1.1
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
    Differentiated Services Field: 0xc0 (DSCP: CS6, ECN: Not-ECT)
        1100 00.. = Differentiated Services Codepoint: Class Selector 6 (48)
        .... ..00 = Explicit Congestion Notification: Not ECN-Capable Transport (0)
    Flags: 0x0000
    Fragment offset: 0
    Time to live: 255
    Protocol: UDP (17)
    Source: 10.102.1.2
    Destination: 10.102.1.1
```

© 2023-25 Cisco and/or its affiliates. All rights reserved.

```
User Datagram Protocol, Src Port: 50138, Dst Port: 4789
    Source Port: 50138
    Destination Port: 4789
Virtual eXtensible Local Area Network
    Flags: 0x0800, VXLAN Network ID (VNI)
        0... .... .... .... = GBP Extension: Not defined
        .... .... .0.. .... = Don't Learn: False
        .... 1... .... .... = VXLAN Network ID (VNI): True
        .... .... .... 0... = Policy Applied: False
        .000 .000 0.00 .000 = Reserved(R): 0x0000
    Group Policy ID: 0
    VXLAN Network Identifier (VNI): 13000
    Reserved: 0
Ethernet II, Src: 00:10:aa:00:04:85, Dst: 00:10:bb:00:04:86
    Destination: 00:10:bb:00:04:86
        Address: 00:10:bb:00:04:86
    Source: 00:10:aa:00:04:85
        Address: 00:10:aa:00:04:85
    Type: 802.1Q Virtual LAN (0x8100)
802.1Q Virtual LAN, PRI: 7, DEI: 0, ID: 600
    111. .... .... .... = Priority: Network Control (7)
    ...0 .... .... .... = DEI: Ineligible
    .... 0010 0101 1000 = ID: 600
    Type: IPv4 (0x0800)
Internet Protocol Version 4, Src: 60.1.1.2, Dst: 60.1.1.1
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
    Differentiated Services Field: 0xc0 (DSCP: CS6, ECN: Not-ECT)
        1100 00.. = Differentiated Services Codepoint: Class Selector 6 (48)
        .... ..00 = Explicit Congestion Notification: Not ECN-Capable Transport (0)
    Total Length: 486
    Identification: 0x13b8 (5048)
    Flags: 0x0000
     Fragment offset: 0
    Time to live: 255
    Protocol: Unknown (253)
    Source: 60.1.1.2
    Destination: 60.1.1.1
```

QinQ-QinVNI Multi tag

QinQ-QinVNI can provide an even greater level of network segmentation and flexibility. For example, a single physical network could have multiple Q-in-Q VLANs running, each with its own set of virtual

© 2023-25 Cisco and/or its affiliates. All rights reserved.

networks created using Q-in-VNI. This allows for a high degree of isolation between different groups of users or applications, while still allowing them to share the same physical infrastructure.

Additionally, QinQ-QinVNI can also be used to create virtual network overlays, which can be used to extend the reach of virtual networks across multiple physical locations, such as in a multi-tenant data center or cloud environment. This allows for more efficient use of resources and greater flexibility in managing and scaling virtual networks.

Overall, the combination of QinQ-QinVNI can provide a powerful toolset for network administrators to create, manage, and scaled virtual networks in a variety of different environments.

In this case study, there are two VLAN tags. The outer VLAN tag is the provider VLAN that maps to the layer-2 VNI in the VXLAN fabric. The inner VLAN has the user VLAN which is transparent to the interface configuration and the VXLAN fabric and is kept intact. The multi-tag allows multiple provider VLANs on the same physical interface as shown below.

QinQ-QinVNI Multi tag with one Customer Tag

Interface configuration sample:

```
switch# config terminal
switch(config-if)# interface Ethernet1/54
switch(config-if)# description Q-in-Q to Q-in-VNI
switch(config-if)# switchport
switch(config-if)# switch(config-if)# switchport mode trunk
switch(config-if)# switch(config-if)# switchport trunk allow-multi-tag
switch(config-if)# switchport trunk allowed vlan 50,100-149,300-301,1300-1499
switch(config-if)# spanning-tree port type edge trunk
switch(config-if)# mtu 9216
switch(config-if)# no shutdown

switch# config terminal
switch(config) vlan 100
switch(config-vlan) vn-segment 30000
switch(config) vlan 101
switch(config-vlan) vn-segment 30001
switch(config) vlan 102
switch(config-vlan) vn-segment 30002
switch(config) vlan 110
.
..
...
switch(config-vlan) vn-segment 30010
switch(config) vlan 111
switch(config-vlan) vn-segment 30011
switch(config) vlan 112
switch(config-vlan) vn-segment 30012
```

© 2023-25 Cisco and/or its affiliates. All rights reserved.

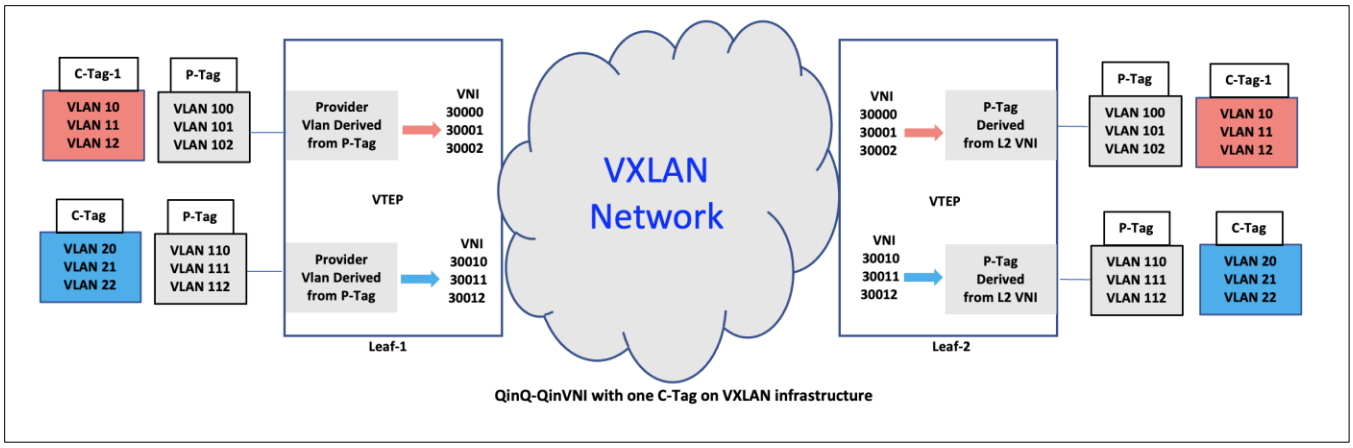


Figure 8. QinQ-QinVNI multi-tag with one C-tag model

Packet capture in the VXLAN fabric is shown below:

- SRC VTEP address 10.102.1.2 is LEAF-1 lo1 address (source-interface of NVE1 interface)
- DST VTEP address 10.102.1.1 is LEAF-2 lo1 address (source-interface of NVE1 interface)
- VXLAN packet UDP header DST Port is 4789
- Provider VLAN 100, Layer-2 VNI 30000 is used for forwarding this traffic
- Customer VLAN 10 is preserved inside the VXLAN packet.
- Inner frames seen with SRC IP 100.1.1.1 and DST IP address 100.1.1.2

```
    [Protocols in frame: eth:ethertype:ip:udp:VXLAN:eth:ethertype:vlan:ethertype:ip:data]
Ethernet II, Src: a0:b4:39:0b:a2:57, Dst: 00:fd:22:8d:1b:75
    Destination: 00:fd:22:8d:1b:75
        Address: 00:fd:22:8d:1b:75
    Source: a0:b4:39:0b:a2:57
        Address: a0:b4:39:0b:a2:57
    Type: IPv4 (0x0800)
Internet Protocol Version 4, Src: 10.102.1.1, Dst: 10.102.1.2
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
    Differentiated Services Field: 0xc0 (DSCP: CS6, ECN: Not-ECT)
        1100 00.. = Differentiated Services Codepoint: Class Selector 6 (48)
        .... ..00 = Explicit Congestion Notification: Not ECN-Capable Transport (0)
    Total Length: 540
    Identification: 0x55aa (21930)
    Flags: 0x0000
     Fragment offset: 0
    Time to live: 255
    Protocol: UDP (17)
  Source: 10.102.1.1
    Destination: 10.102.1.2
```

© 2023-25 Cisco and/or its affiliates. All rights reserved.

```
User Datagram Protocol, Src Port: 29571, Dst Port: 4789
    Source Port: 29571
    Destination Port: 4789
Virtual eXtensible Local Area Network
    Flags: 0x0800, VXLAN Network ID (VNI)
        0... .... .... .... = GBP Extension: Not defined
        .... .... .0.. .... = Don't Learn: False
        .... 1... .... .... = VXLAN Network ID (VNI): True
        .... .... .... 0... = Policy Applied: False
        .000 .000 0.00 .000 = Reserved(R): 0x0000
    Group Policy ID: 0
    VXLAN Network Identifier (VNI): 30000
    Reserved: 0
Ethernet II, Src: 00:85:54:02:01:02, Dst: 00:51:52:02:01:01
    Destination: 00:51:52:02:01:01
        Address: 00:51:52:02:01:01
    Source: 00:85:54:02:01:02
        Address: 00:85:54:02:01:02
    Type: 802.1Q Virtual LAN (0x8100)
802.1Q Virtual LAN, PRI: 7, DEI: 0, ID: 10
    111. .... .... .... = Priority: Network Control (7)
    ...0 .... .... .... = DEI: Ineligible
    .... 0000 0000 1010 = ID: 10
    Type: IPv4 (0x0800)
Internet Protocol Version 4, Src: 100.1.1.1, Dst: 100.1.1.2
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
    Differentiated Services Field: 0xc0 (DSCP: CS6, ECN: Not-ECT)
        1100 00.. = Differentiated Services Codepoint: Class Selector 6 (48)
        .... ..00 = Explicit Congestion Notification: Not ECN-Capable Transport (0)
    Total Length: 486
    Identification: 0xef59 (61273)
    Flags: 0x0000
     Fragment offset: 0
    Time to live: 255
    Protocol: Unknown (253)
    Source: 100.1.1.1
    Destination: 100.1.1.2
```

QinQ-QinVNI Multi tag with two Customer Tag

As discussed above, in this case, Q-in-VNI port is shown in the packet capture, sending traffic to another Q-in-VNI port with two user VLAN tags: outer VLAN 10 and inner VLAN 500. The provider VLAN for the Q-

© 2023-25 Cisco and/or its affiliates. All rights reserved.

in-VNI is 100, which is mapped to layer-2 VNI 30000. The packet capture displays both the outer and inner user VLANs, as well as the layer-2 VNI 30000. Again both the user VLANs are retained inside the fabric.

Interface Configuration is the same as the previous one:

```
switch# config terminal
switch(config-if)# interface Ethernet1/54
switch(config-if)# description Q-in-Q to Q-in-VNI
switch(config-if)# switchport
switch(config-if)# switch(config-if)# switchport mode trunk
switch(config-if)# switch(config-if)# switchport trunk allow-multi-tag
switch(config-if)# switchport trunk allowed vlan 50,100-149,300-301,1300-1499
switch(config-if)# spanning-tree port type edge trunk
switch(config-if)# mtu 9216
switch(config-if)# no shutdown

switch# config terminal
switch(config) vlan 100
switch(config-vlan) vn-segment 30000
switch(config) vlan 300
switch(config-vlan) vn-segment 33000
switch(config) vlan 301
switch(config-vlan) vn-segment 33001
```

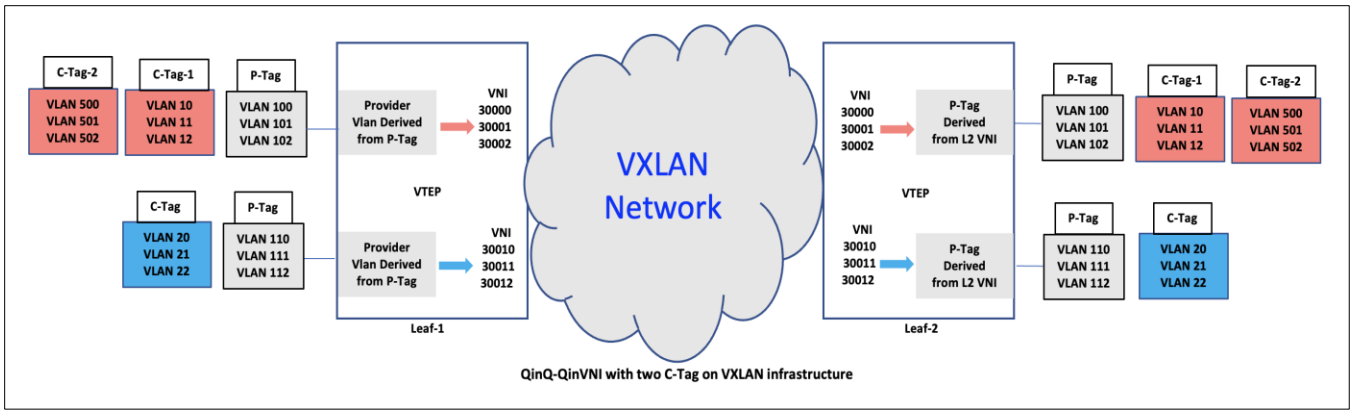


Figure 9. QinQ-QinVNI multi tag with two C-tag model

Packet capture in the VXLAN fabric is shown below,

- SRC VTEP address 10.102.1.2 is LEAF-1 lo1 address (source-interface of NVE1 interface)
- DST VTEP address 10.102.1.1 is LEAF-2 lo1 address (source-interface of NVE1 interface)
- VXLAN packet UDP header DST Port is 4789
- Provider VLAN 300, Layer-2 VNI 33000 is used for forwarding this traffic
- User VLAN 10, first C-tag is preserved inside the VXLAN packet.
- User VLAN 500, Second C-tag is also preserved inside the VXLAN packet.

© 2023-25 Cisco and/or its affiliates. All rights reserved.

- Inner frames seen with SRC IP 10.1.1.1 and DST IP address 10.1.1.2

```
 [Protocols in frame:
eth:ethertype:ip:udp:vxlan:eth:ethertype:vlan:ethertype:vlan:ethertype:arp]
Ethernet II, Src: a0:b4:39:0b:a2:57, Dst: 00:fd:22:8d:1b:75
    Destination: 00:fd:22:8d:1b:75
        Address: 00:fd:22:8d:1b:75
    Source: a0:b4:39:0b:a2:57
        Address: a0:b4:39:0b:a2:57
    Type: IPv4 (0x0800)
Internet Protocol Version 4, Src: 10.102.1.1, Dst: 10.102.1.2
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
    Differentiated Services Field: 0xe0 (DSCP: CS7, ECN: Not-ECT)
        1110 00.. = Differentiated Services Codepoint: Class Selector 7 (56)
        .... ..00 = Explicit Congestion Notification: Not ECN-Capable Transport (0)
Flags: 0x0000
        Fragment offset: 0
    Time to live: 255
    Protocol: UDP (17)
    Source: 10.102.1.1
    Destination: 10.102.1.2
User Datagram Protocol, Src Port: 53720, Dst Port: 4789
    Source Port: 53720
    Destination Port: 4789
Virtual eXtensible Local Area Network
    Flags: 0x0800, VXLAN Network ID (VNI)
        0... .... .... .... = GBP Extension: Not defined
        .... .... .0.. .... = Don't Learn: False
        .... 1... .... .... = VXLAN Network ID (VNI): True
        .... .... .... 0... = Policy Applied: False
        .000 .000 0.00 .000 = Reserved(R): 0x0000
    Group Policy ID: 0
    VXLAN Network Identifier (VNI): 33000
    Reserved: 0
Ethernet II, Src: 00:bb:94:00:04:89, Dst: 00:bb:94:00:04:88
    Destination: 00:bb:94:00:04:88
        Address: 00:bb:94:00:04:88
    Source: 00:bb:94:00:04:89
        Address: 00:bb:94:00:04:89
    Type: 802.1Q Virtual LAN (0x8100)
802.1Q Virtual LAN, PRI: 7, DEI: 0, ID: 10
    111. .... .... .... = Priority: Network Control (7)
```

© 2023-25 Cisco and/or its affiliates. All rights reserved.

```
    ...0 .... .... .... = DEI: Ineligible
    .... 0000 0000 1010 = ID: 10
    Type: 802.1Q Virtual LAN (0x8100)
802.1Q Virtual LAN, PRI: 7, DEI: 0, ID: 500
    111. .... .... .... = Priority: Network Control (7)
    ...0 .... .... .... = DEI: Ineligible
    .... 0001 1111 0100 = ID: 500
    Type: ARP (0x0806)
    Trailer: 00000000000000000000
Address Resolution Protocol (reply)
    Hardware type: Ethernet (1)
    Protocol type: IPv4 (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: reply (2)
    Sender MAC address: 00:bb:94:00:04:89
    Sender IP address: 10.1.1.1
    Target MAC address: 00:bb:94:00:04:88
    Target IP address: 10.1.1.2
```

QinVNI with Layer 2 Protocol Tunneling

Layer 2 Protocol Tunneling is the process of encapsulating a Layer 2 (L2) protocol within a different L2 protocol, so that it can be transmitted over a network that does not support the original L2 protocol. It is aimed to extend the reach of L2 protocols over networks that would otherwise not support them, by encapsulating the protocol in a compatible format.

Spanning Tree Protocol (STP), VLAN Trunking Protocol (VTP), Link Layer Discovery Protocol (LLDP), and Cisco Discovery Protocol (CDP) are all Layer 2 protocols that can be tunneled. These protocols are typically used to manage the behavior and configuration of switches in a local area network (LAN), and tunneling them can enable their use over networks that would not otherwise support them.

Tunneling these protocols can help to maintain consistency and manageability of the network, even when the network is spread across multiple physical locations or spans multiple underlying networking technologies.

© 2023-25 Cisco and/or its affiliates. All rights reserved.

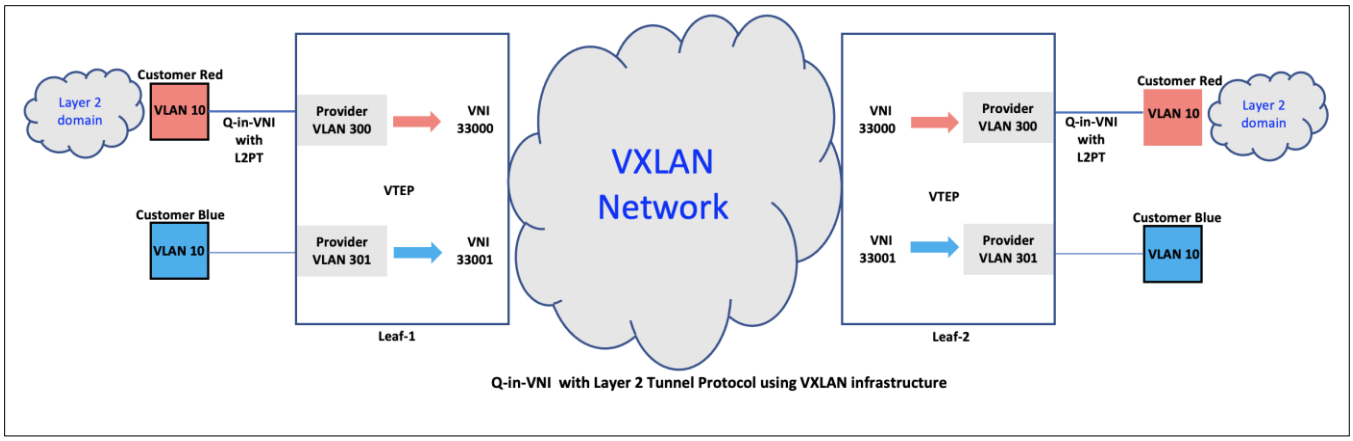


Figure 10. Q-in-VNI with Layer 2 Tunneling Protocol at provider edge

Interface configuration sample:

```
switch(config-if)# interface Ethernet1/51
switch(config-if)# description Q-in-VNI for Customer Red
switch(config-if)# switchport
switch(config-if)# switchport mode dot1q-tunnel
switch(config-if)# switchport access vlan 300
switch(config-if)# spanning-tree port type edge
switch(config-if)# l2protocol tunnel cdp
switch(config-if)# l2protocol tunnel stp
switch(config-if)# l2protocol tunnel vtp
switch(config-if)# l2protocol tunnel lldp
switch(config-if)# mtu 9216
switch(config-if)# no shutdown

switch# config terminal
switch(config) vlan 300
switch(config-vlan) vn-segment 33000

switch# show l2protocol tunnel
COS for Encapsulated Packets: 5
                        Shutdown    Drop    Encapsulation Decapsulation     Drop
  Port    Protocol      Threshold Threshold    Counter       Counter       Counter
-------- --------      --------- --------- ------------- ------------- -------------
Eth1/51  cdp             ----      ----            0             0             0
         stp             ----      ----          152             0             0
         vtp             ----      ----            0             0             0
         lldp            ----      ----           11             0             0
         ----            ----      ----         ----          ----          ----
         ----------      ----      ----         ----          ----          ----
```

© 2023-25 Cisco and/or its affiliates. All rights reserved.

Q-in-VNI with LACP Tunneling

LACP (Link Aggregation Control Protocol) tunneling over VXLAN is a method for transmitting LACP frames over a VXLAN network. LACP is used to bundle multiple physical links together to form a single logical link, providing increased bandwidth and redundancy.

When LACP is tunneled over VXLAN, LACP frames are encapsulated within VXLAN frames and transmitted over the VXLAN network. This allows LACP to be used to manage link aggregation in networks that use VXLAN as the underlying technology.

Tunneling LACP over VXLAN can be beneficial in large, geographically dispersed networks where multiple physical locations are connected over VXLAN. By encapsulating LACP frames within VXLAN frames, the protocol can be used to manage link aggregation across the WAN, helping to ensure high availability and resiliency of the network.

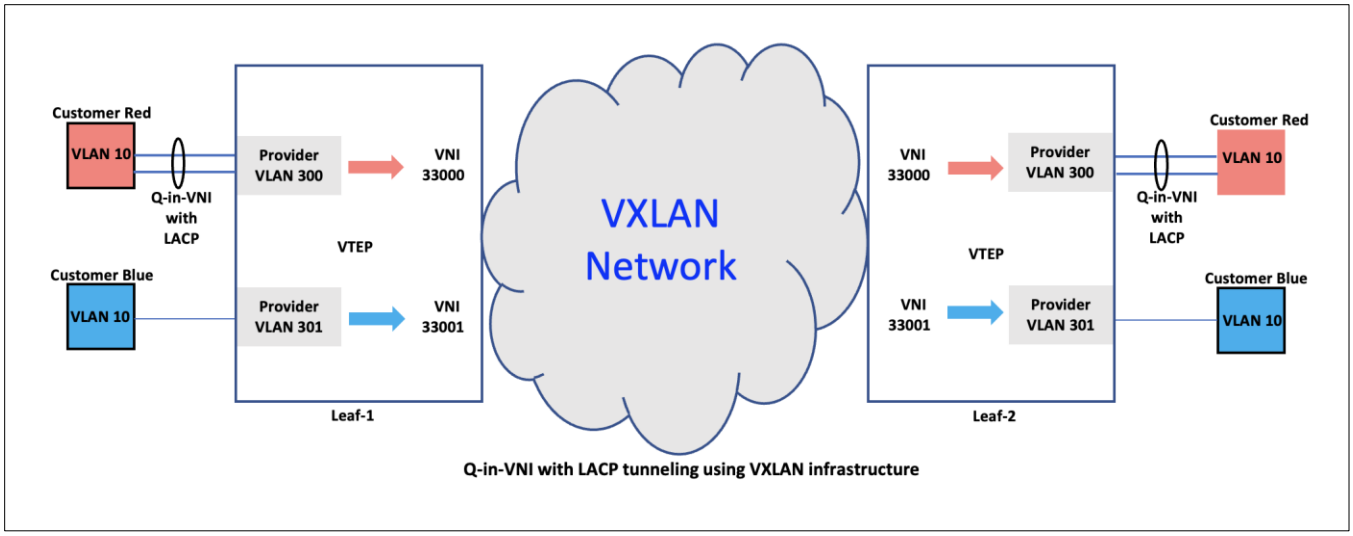


Figure 11. Q-in-VNI with LACP Tunneling at provider edge

Interface configuration sample:

```
switch# config terminal
switch(config)# interface ethernet 1/51
switch(config-if)# switchport mode dot1q-tunnel
switch(config-if)# switchport access vlan 10
switch(config-if)# spanning-tree bpdufilter enable
switch(config-if)# l2protocol tunnel lacp
switch(config-if)# interface nve1
switch(config-if)# overlay-encapsulation vxlan-with-tag tunnel-control-frames

switch# show l2protocol tunnel
COS for Encapsulated Packets: 5
                        Shutdown    Drop    Encapsulation Decapsulation     Drop
  Port    Protocol      Threshold Threshold    Counter       Counter       Counter
-------- --------       --------- --------- ------------- ------------- -------------
Eth1/51  cdp             ----      ----            0             0             0
```

© 2023-25 Cisco and/or its affiliates. All rights reserved.

```
stp                 ----       ----              0              0              0
vtp                 ----       ----              0              0              0
lldp                ----       ----              0              0              0
lacp                ----       ----              0              0              0
----------          ----       ----           ----           ----           ----
```

Support on Nexus 9000 Switches

The following table summarizes Q-in-VNI, Selective Q-in-VNI, QinQ-QinVNI, and L2PT enhancements across NX-OS releases.

NX-OS Release	Feature Detail	Supported Platforms	Enhancement/ Classification
10.4(3)F	QinQ-QinVNI, L2PT, and IPv6 underlay support	9364C-H1	Multi-tag, L2 protocol tunneling, IPv6 transport
10.4(2)F	Q-in-VNI, Selective Q-in-VNI, and QinQ-QinVNI support	93400LD-H1	Provider VLAN coexistence, IPv6 underlay
10.4(1)F	Q-in-VNI, Selective Q-in-VNI, QinQ-QinVNI, and L2PT support	9332D-H2R	vPC Fabric Peering, IPv6 underlay, multi-tag trunk
10.3(3)F	IPv6 underlay support for Q-in-VNI and Selective Q-in-VNI	9300-FX/FX2/FX3/GX/GX2	VXLAN EVPN IR with IPv6 transport
10.3(2)F	Q-in-VNI with L2PT support	9300-FX/FX2/FX3/GX/GX2	Multi-tag trunk and Ethertype configuration
10.2(3)F	Q-in-VNI coexistence with L3 subinterfaces, QinQ-QinVNI support	9300-GX2	Multi-tenant bridging, subinterface coexistence
9.3(5)	Initial Q-in-VNI and Selective Q-in-VNI introduction	9300-GX, FX/FX2	vPC Fabric Peering, dot1q-tunnel transit VLANs

Guidelines and limitations

The following sections include guidelines and limitations that apply to the deployment and configuration of Qin VNI scenarios to ensure proper operation, compatibility, and platform support:

For Q-in-VNI

- The `system dot1q-tunnel transit [vlan vlan-range]` command is required when running this feature on vPC VTEPs.
- Port VLAN mapping and Q-in-VNI cannot coexist on the same port.
- Port VLAN mapping and Q-in-VNI cannot coexist on a switch if the system dot1q-tunnel transit command is enabled.
- Beginning with Cisco NX-OS Release 9.3(5), they can coexist on the same switch but only on different ports and different provider VLANs, configured using the command:

  `system dot1q-tunnel transit vlan <vlan-range>`
- For proper operation during Layer 3 uplink failure scenarios on vPC VTEPs, configure a backup SVI and enter the command:

  `system nve infra-vlans backup-svi-vlan <vlan-id>`
- On Cisco Nexus 9000 Series switches with a Leaf Spine Engine (LSE), you can configure access, trunk, and dot1q-tunnel ports on different interfaces of the same switch.
- You cannot configure the same VLAN simultaneously for both dot1q-tunnel and trunk/access ports.

© 2023-25 Cisco and/or its affiliates. All rights reserved.

- Disable ARP suppression on the provider VNI to allow ARP traffic originating from a customer VLAN to flow.
- Q-in-VNI cannot coexist with a VTEP that has Layer 3 subinterfaces configured.
- Beginning with Cisco NX-OS Release 9.3(5), this limitation no longer applies to Nexus 9300-FX/FX2 and 9300-GX platform switches.
- When VLAN 1 is configured as the native VLAN with Selective Q-in-VNI and multiple provider tags, traffic on the native VLAN is dropped.
- Do not configure VLAN 1 as the native VLAN when the port is configured with Selective Q-in-VNI.
- If VLAN 1 is configured as a customer VLAN, its traffic is also dropped.
- The base port mode must be configured as a dot1q-tunnel port with an access VLAN.
- VNI mapping is required for the access VLAN on the port.

For Selective Q-in-VNI

- Supported on both vPC and non-vPC ports for Cisco Nexus 9300-FX/FXP/FX2/FX3 and 9300-GX platform switches.
- Starting with Cisco NX-OS Release 9.3(5), supports vPC Fabric Peering.
- You can configure Selective Q-in-VNI on one VTEP and plain Q-in-VNI on its VXLAN peer; mixed configuration on the same switch is also supported.
- This is an ingress VLAN tag-policing feature — only ingress policing applies within the configured VLAN range.
- Port VLAN mapping and Selective Q-in-VNI cannot coexist on the same port.
- If the `system dot1q-tunnel transit` command is enabled, port VLAN mapping and Selective Q-in-VNI cannot coexist on the same switch unless they use different ports and provider VLANs (specified by `system dot1q-tunnel transit vlan <vlan-range>`).
- On vPC switches, configure system dot1q-tunnel transit [vlan vlan-id] to preserve the inner Q-tag across the vPC peer link when one peer has an orphan port.
- The native VLAN on a Selective Q-in-VNI port cannot overlap with the customer VLAN range; such a configuration is rejected.
- The provider VLAN may overlap with the customer range (for example: `switchport vlan mapping 100-1000 dot1q-tunnel 200`).
- If VLAN 1 is configured as part of the customer VLAN range, its traffic will not be carried because VLAN 1 is the default native VLAN.
- Configure a dummy native VLAN outside the customer range if VLAN 1 traffic must be transported.
- To remove VLANs or ranges from an existing mapping, use:

  `no switchport vlan mapping <range> dot1q-tunnel <sp-vlan>`

For Q-in-VNI with L2PT

- Only two interfaces in the network can belong to a given tunnel VLAN.
- For vPC deployments, both vPC member ports on the vPC pair (and the MCT link) participate in the same tunnel VLAN.

© 2023-25 Cisco and/or its affiliates. All rights reserved.

- The same VLAN must not be tunneled on multiple interfaces.
- The l2protocol tunnel vxlan command is allowed only on trunk ports and requires the switchport trunk allow-multi-tag configuration to preserve multiple tags across the VXLAN fabric.
- The Cross Connect feature and l2protocol tunnel vxlan command cannot be used together on the same switch.
- Existing L2PT command options such as stp, cdp, or vtp are not supported when the l2protocol tunnel vxlan command is used.Beginning with Cisco NX-OS Release 10.3(3)F, Ethertype support for Q-in-VNI with L2PT is available on Cisco Nexus 9300-FX2/FX3/GX/GX2 ToR switches.Beginning with Cisco NX-OS Release 10.4(1)F, Ethertype support is available on Cisco Nexus 9332D-H2R switches.

For QinQ-QinVNI

- This feature is supported on Cisco Nexus 9300-FX/FX2/FX3 and 9300-GX platform switches.
- This feature supports vPC Fabric Peering.On a multi-tag port, provider VLANs must be part of the port configuration; they are used to derive the VNI for each packet.
- Untagged packets are associated with the native VLAN. If a native VLAN is not configured, they are associated with the default VLAN (VLAN 1).
- Packets arriving with an outermost provider VLAN tag that is not in the allowed VLAN range on a multi-tag port are dropped.
- Packets whose outermost provider VLAN tag matches the native VLAN are bridged or routed within the native VLAN's domain.
- QinQ-QinVNI supports VXLAN bridging only; VXLAN routing is not supported.
- Multicast data traffic containing more than two Q-tags is not supported when snooping is enabled on the VXLAN VLAN.
- At least one multi-tag trunk port allowing provider VLANs must be in the Up state on both vPC peers.
- Otherwise, traffic that traverses the peer-link for those provider VLANs will not carry all inner customer tags (C-tags).
- The `system dot1q-tunnel transit [vlan vlan-range]` command is required when running this feature on vPC VTEPs.

References

Cisco Configuration Guide for Q-in-VNI on NX-OS Release 10.6(x)

https://www.cisco.com/c/en/us/td/docs/dcn/nx-os/nexus9000/106x/configuration/vxlan/cisco-nexus-9000-series-nx-os-vxlan-configuration-guide-release-106x/configuring-q-in-vni-over-vxlan.html

VXLAN Design with Cisco Nexus 9300 Platform Switches

https://www.cisco.com/c/en/us/products/collateral/switches/nexus-9000-series-switches/white-paper-c11-732453.html

© 2023-25 Cisco and/or its affiliates. All rights reserved.

VXLAN Network with MP-BGP EVPN Control Plane Design Guide

https://www.cisco.com/c/en/us/products/collateral/switches/nexus-9000-series-switches/guide-c07-734107.html

Cisco Press: Building Data Centers with VXLAN BGP EVPN: A Cisco NX-OS Perspective

https://www.ciscopress.com/store/building-data-centers-with-VXLAN-bgp-evpn-a-cisco-nx-9781587144677

Configuring VXLAN BGP EVPN

https://www.cisco.com/c/en/us/td/docs/dcn/nx-os/nexus9000/106x/configuration/vxlan/cisco-nexus-9000-series-nx-os-vxlan-configuration-guide-release-106x/m_configuring_vxlan_bgp_evpn.html

Configuring VXLAN Multisite

https://www.cisco.com/c/en/us/td/docs/dcn/nx-os/nexus9000/106x/configuration/vxlan/cisco-nexus-9000-series-nx-os-vxlan-configuration-guide-release-106x/m_configuring_multisite_93x.html

Legal Information

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: www.cisco.com/go/trademarks. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1721R)

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

© 2023-2025 Cisco Systems, Inc. All rights reserved.