Virtual Extensible LAN

GIANLUCA REALI

Overview

Virtualization of servers causes challenges for Datacenter networks with traditional three-layer architecture

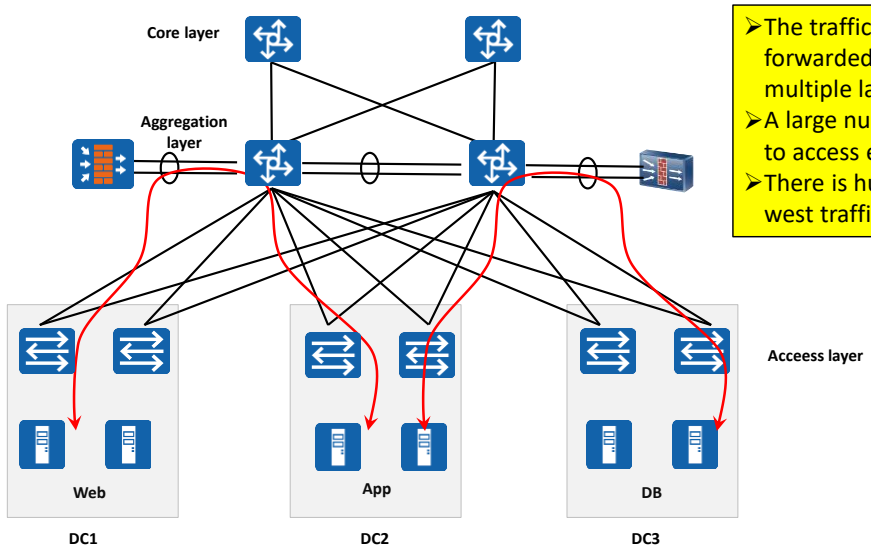
VXLAN can respond to these challenges.

Server virtualization greatly reduces IT construction and operations and maintenance (O&M) costs and improves service deployment flexibility.

Virtual machines (VMs) on a traditional data center network can only seamlessly migrate on Layer 2. If VMs migrate across a Layer 3 network, services will be interrupted.

The Virtual eXtensible Local Area Network (VXLAN) technology is introduced to improve VM migration flexibility, so that the large number of tenants are not limited by IP address changes and broadcast domains.

Challenge: Low Latency Requirements of Compute Nodes



- The traffic needs to be forwarded by switches at multiple layers.
- A large number of VMs need to access each other.
- There is huge delay in east-west traffic.

~~A large number of VMs are deployed on a physical server, causing huge traffic concurrency.~~

The data traffic model is converted from the traditional north-south traffic to east-west traffic.

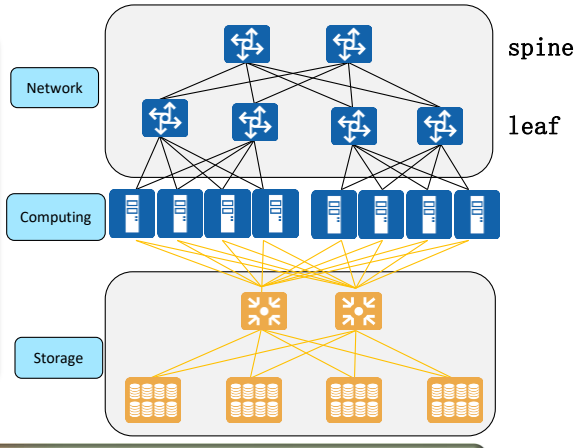
A large amount of many-to-one and many-to-many east-west traffic exists on the network.

The devices on the access and aggregation layers need to provide high processing capabilities.

Data Center Basic Concepts and Features

What Is a Data Center

- Data Center (DC)
- Core of the enterprise IT system
- Massive data computing, switching, and storage center
- Computing environment for key information, services, and applications
- Environment for centralized management and control of various data, applications, physical devices, and virtual devices



The best example of a switching fabric topology is the Spine-and-Leaf, which is commonly used as an underlay network.

A data center has four features: reliable, flexible, environmentally-friendly, and efficient.

A Data Center (DC) is a collection of complete and complex systems consisting of the computing system, auxiliary devices (such as the communications and storage system), data communication system, environment control devices, monitoring devices, and various security devices.

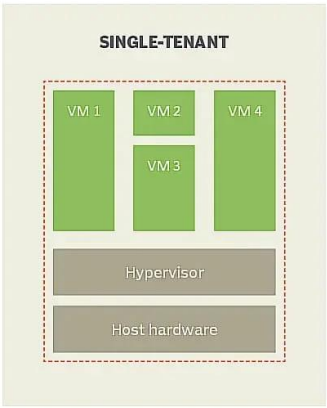
A data center stores, processes, transmits, switches, and manages information in a centralized mode within a physical space.

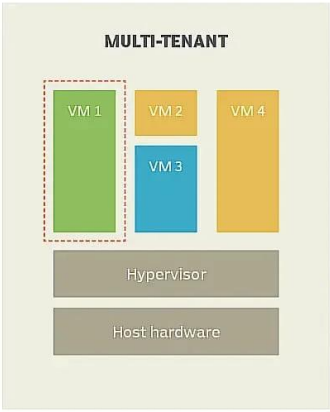
Key devices in a data center include servers, network devices, and storage devices.

Power supply system, air conditioning system, cabinets, fire protection system, monitoring system, and other systems that affect operating environment of the key devices are key physical facilities.

An Internet Data Center (IDC) is a center for data storage and processing on the Internet that involves the most intensive data exchange.

Architectural Alternatives





Tenancy in cloud computing refers to the sharing of computing resources in a private or public environment that is isolated from other users and kept secret. Tenancy in SaaS is divided into two types: single-tenant SaaS and multi-tenant SaaS.

https://www.cloudzero.com/blog/single-tenant-vs-multi-tenant/

In SaaS (Software-as-a-Service) , a single-tenant architecture is where a single instance of the software application, and its supporting infrastructure, serves a single customer (tenant).
The single instance of the software runs on a dedicated cloud server for just this one customer. So there is no sharing it with any other customer of the SaaS provider.

A multi-tenant architecture is where a single software instance and its supporting infrastructure are shared among two or more customers (tenants) at the same time. While each tenant's data is isolated from the other tenants', each customer shares a single database, software application, and SaaS server with the others.

meh leggi se vuoi capire ma meh

Possible challenge in a small/medium datacenter

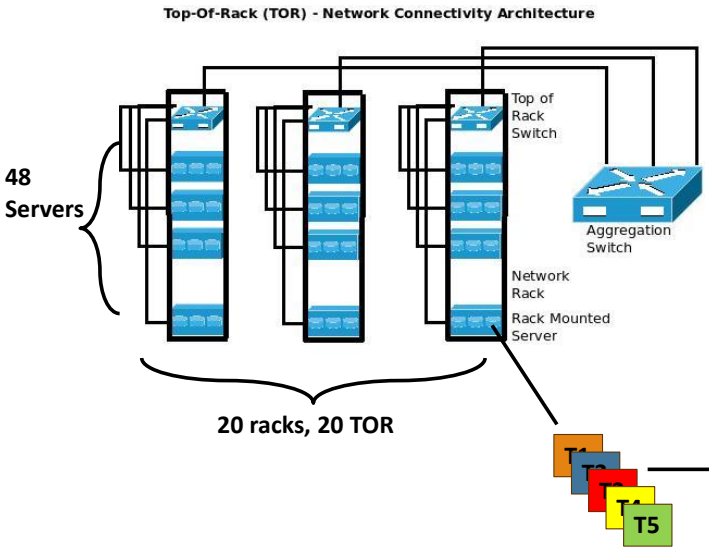
Assume a datacente with **20 racks** including **48 physical servers** connected to each access switch (TOR). Each of these servers includes **five different tenants** (dedicated virtualized environments) with their own virtual routing (VRF). **One tenant consists of three broadcast domains (VLAN)**, e.g. Presentation, Application and Database, each with two virtual machines backing up each other. The customer manages his own tenant and can define the VLANs IDs, the mac addresses of virtual machines and the IP address architecture.

The mobility of virtual machines is unlimited.

Virtual Routing and Forwarding (VRF) IP Technology allows users to configure multiple routing table instances to simultaneously co-exist within the same router.

meh ancora, è un esempio

Be quantitative!



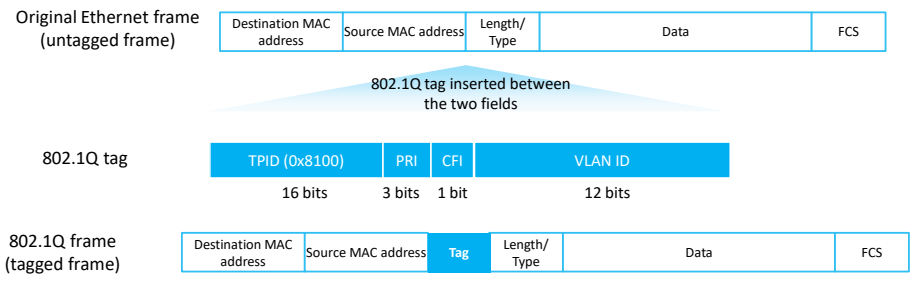
 # Physical servers: 20 (ToR) x 48 (port per ToR) = 960
 # VMs - Mac addresses - ARP entries:
960 (servers) x 30 (VM per server) =28800
 # Tenants / VRF: 960 hosts x 5 tenants = 4800
 # Broadcast Domains: 5 (tenant per server) x 3 (VLANs per tenant) x 960 (servers)=14400

UNLIMITED MOBILITY REQUIRED !!!

Virtual Routing and Forwarding (VRF) IP Technology allows users to configure multiple routing table instances to simultaneously co-exist within the same router.

Summary of identified iusses:

- **The isolation capability of traditional networks is limited**. VLAN is a mainstream network isolation technology. However, the VLAN tag field defined in **IEEE 802.1Q has only 12 bits** and can identify only a **maximum of 4096 VLANs**, which cannot meet the isolation requirements of a large number of tenants.



Summary of identified iusses:

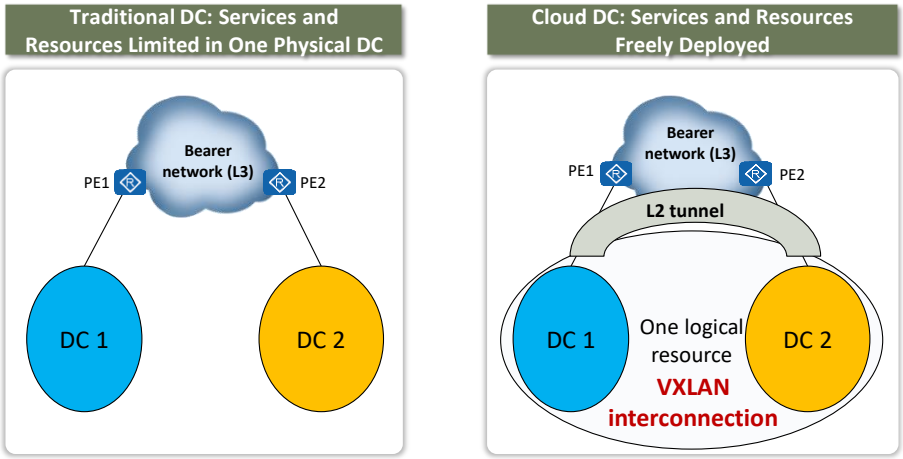
- **The VM migration scope is limited**. To ensure services continuity during VM migration, the IP addresses and MAC addresses of the VMs must remain unchanged before and after the migration. This means that **VM migration must occur in one Layer 2 domain**. However, VM migration in Layer 2 domains of traditional data center networks is limited to a small scope.

VXLAN solutions

- Limitation of network isolation capabilities: 24 bits isolation identifier, similar to VLAN ID, which is called VXLAN Network ID (VNI).
- Limitation of the VM migration scope: VXLAN encapsulates Ethernet packets in IP packets and transmits them over routes on a network to construct a large Layer 2 network. Therefore, VM migration is not restricted by the network architecture.

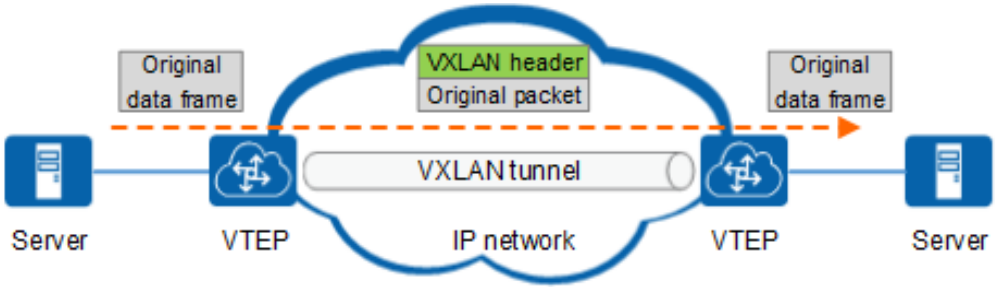
A VXLAN network is a virtual Layer 2 network constructed on a Layer 3 network to enable communication of hosts at Layer 2. Compared with VLAN, VXLAN has higher flexibility and scalability, and addresses the following issues:
Limitation of the VM scale imposed by network specifications; Data packets sent by VMs are encapsulated in IP data packets. The network is only aware of the encapsulated network parameters. This greatly reduces the number of MAC address entries required by large Layer 2 networks.
Limitation of network isolation capabilities VXLAN technology extends the number of isolation identifier bits to 24 bits, which greatly increases the number of tenants that can be isolated. Theoretically, up to 16 million tenants can be isolated. VXLAN introduces a network identifier similar to VLAN ID, which is called VXLAN Network ID (VNI). Each VNI has 24 bits and can identify up to 16 million tenants, meeting the isolation requirements of a large number of tenants.
Limitation of the VM migration scope imposed by the network architectureVXLAN encapsulates Ethernet packets in IP packets and transmits them over routes on a network to construct a large Layer 2 network. Therefore, VM migration is not restricted by the network architecture. In addition, a routed network has good scalability, self-healing capability, and load balancing capability.

Large Layer 2 Interconnection Among Multiple DCs - VXLAN



In this early stage, VM management and migration are completed on physical networks. Therefore, the east-west traffic in a DC is mainly Layer 2 traffic.
To extend the scale of a Layer 2 physical network and improve link utilization, large Layer 2 technologies, such as Transparent Interconnection of Lots of Links (TRILL) and Shortest Path Bridging (SPB) are developed.
As the virtualized DC scale expands and cloud-based management becomes popular, VM management and migration on physical networks can no longer meet virtualization requirements. As a result, overlay technologies such as VXLAN and Network Virtualization using Generic Routing Encapsulation (NVGRE) are developed.
In the overlay solution, east-west traffic on a physical network is gradually changed from Layer 2 traffic to Layer 3 traffic. In addition, the overlay solution changes the network topology from physical Layer 2 to logical Layer 2 and provides the logical Layer 2 division and management functions, better matching requirements of multiple tenants.
Overlay technologies such as VXLAN and NVGRE use MAC-in-IP encapsulation to solve limitations of physical networks, including the limit on the number of VLANs and MAC address entries supported by access switches. These technologies also provide a unified logical network management tool to enable policy migration during VM migration, greatly reducing network dependency during virtualization. These technologies attract major concern in network virtualization.

VXLAN network model



- Overlay network: VXLAN
- Underlay network: IP

Below IP, Spine-and-Leaf uses two layers:

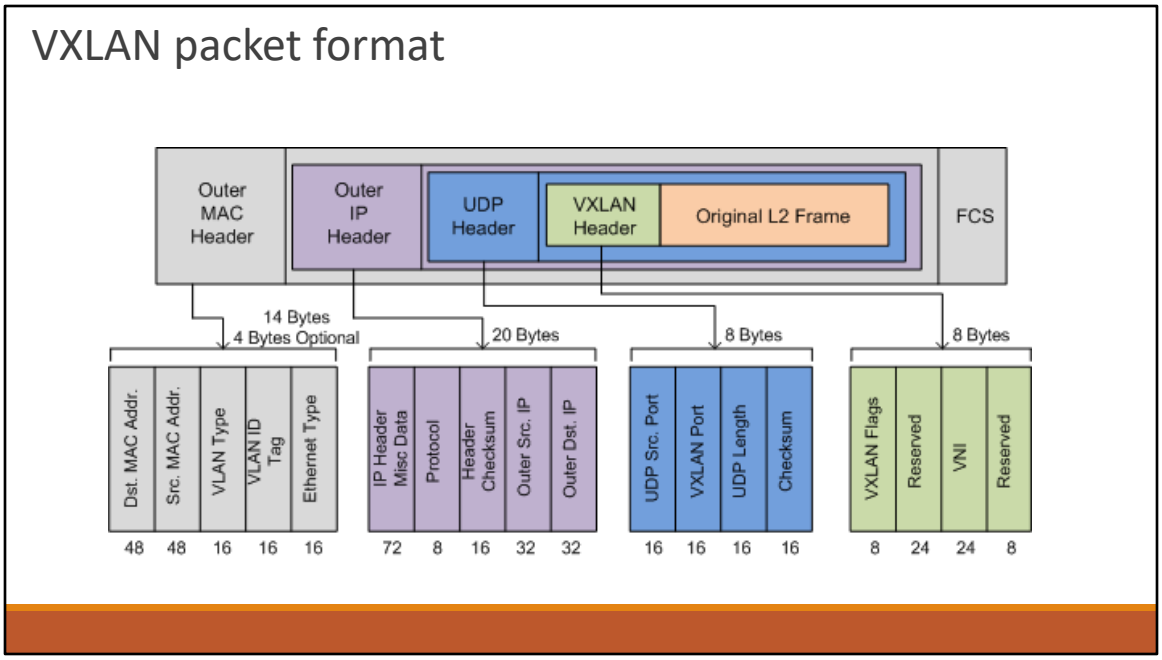
•**Spine:** The spine layer switches are only used to pass traffic through leaf switches. They are not aware of VxLAN.

•**Leaf:** The Leaf layer of switches interconnect the spine and the end points. The leaf layer switches create the VxLAN tunnels, encapsulation, and maps VLANs to VNI. The leaf switches that perform VxLAN functions are known as VTPEs (VxLAN Tunnel Endpoints

To address the preceding problems, overlay network technologies are gradually evolved to meet the network capability requirements of cloud computing. There are multiple overlay technologies, such as Virtual eXtensible Local Area Network (VXLAN), Network Virtualization using Generic Routing Encapsulation (NVGRE), and Stateless Transport Tunneling (STT). This document describes VXLAN that is the most widely used overlay technology.

VXLAN is one of the Network Virtualization over Layer 3 (NVO3) technologies defined by the Internet Engineering Task Force (IETF) and is essentially a tunneling technology. VXLAN adds the VXLAN header to an original data frame, encapsulates the frame into a UDP packet, and forwards the UDP packet in traditional IP network transmission mode. After the UDP packet arrives at the end point, the end point removes the outer header and sends the original data frame to the target terminal.

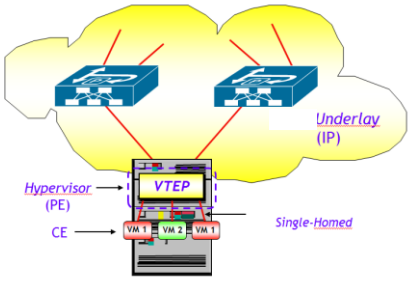
The end point of a VXLAN tunnel as shown in the Figure is called **VXLAN Tunnel Endpoint** (VTEP), which encapsulates and decapsulates VXLAN packets. A VXLAN tunnel is defined by a pair of VTEPs. The source VTEP encapsulates packets and sends them to the destination VTEP through the VXLAN tunnel. The destination VTEP decapsulates the received packets.



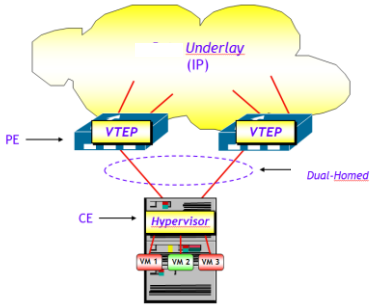
VXLAN Header: contains the 24-bit VNI field and 8-bit VXLAN flag bit. The other fields are reserved.
UDP Header: contains the destination port number fixed at 4789. The VXLAN header and the original Ethernet frame are used as UDP data.
Outer IP Header: The source IP address is the IP address of the source VTEP and the destination IP address is the IP address of the destination VTEP.
Outer MAC Header: The source MAC address is the MAC address of the source VTEP, and the destination MAC address is the MAC address of the next-hop device on the route to the destination VTEP.

Where is VTEP implemented?

Any endpoint like a host, switch, or router that supports VxLAN can be referred to as a VTPE (VxLAN Tunnel Endpoint).



The VTEP function is implemented by software and located in a *Hypervisor* (e.g. *KVM*, *Hyper-V*, *VMware NSX*, etc.).



The VTEP function is hardware implemented and located in a TOR switch

Any endpoint like a host, switch, or router that supports VxLAN can be referred to as a VTPE (VxLAN Tunnel Endpoint).

As the name implies, the job of VTEPs is to create and terminate tunnels between each other. In other words, they encapsulate and decapsulate VxLAN traffic.
a. How does a VTEP work?

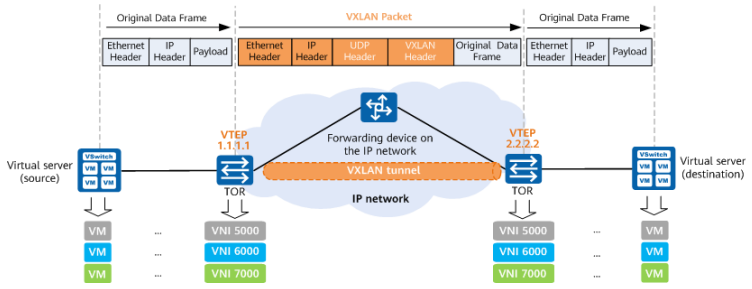
The VTPE is connected to the underlay network using a layer 3 IP address. VTPEs may have one or more VNIs associated with it.

When a layer 2 frame with the same VNI arrives at the ingress VTEP, it encapsulates the frame with a VxLAN and UDP/IP headers.

Then sends it over using the underlay IP network transport towards the egress VTPE for decapsulation.

The egress VTPE removes the IP and UDP headers and delivers the original layer 2 frame.

VXLAN network model

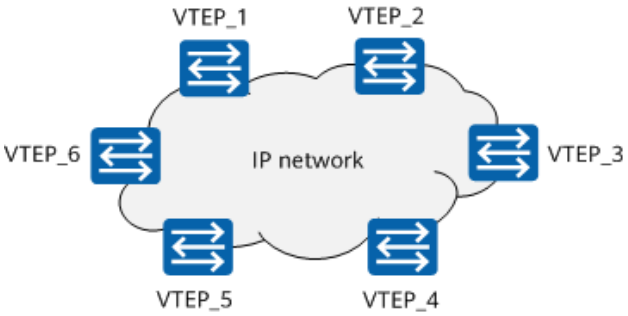


The VTEP is connected to the underlay network using a layer 3 IP address. **VTPEs may have one or more VNIs associated with it**.

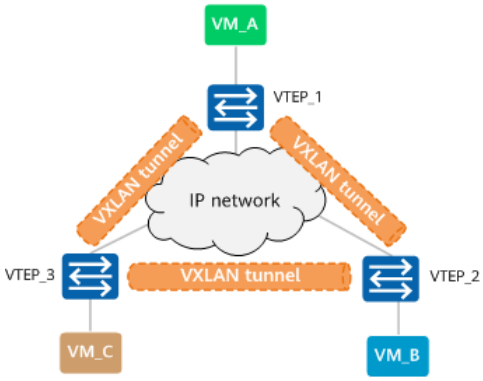
When a layer 2 frame with the same VNI arrives at the ingress VTEP, it encapsulates the frame with a VxLAN and UDP/IP headers.

https://support.huawei.com/enterprise/en/doc/EDOC1100086966

VXLAN network model

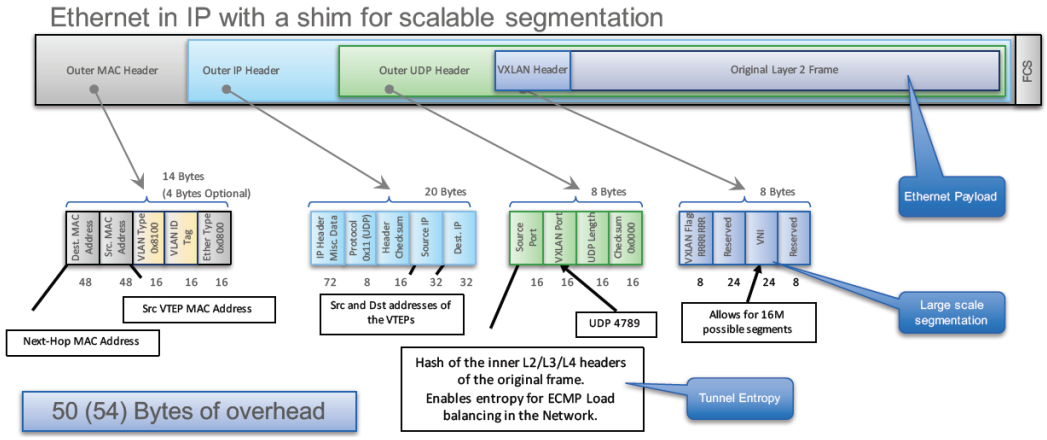


Assume that VMs each connected to VTEP_1, VTEP_2, and VTEP_3 respectively require large Layer 2 interconnection. Every two of VTEP_1, VTEP_2, and VTEP_3 then need to establish VXLAN tunnels between them,



The main problem to be addessed is L2-L3 address management and distribution over the VXLAN.

VXLAN packet format



RFC7348: IANA has assigned the value **4789** for the VXLAN UDP port, and this value SHOULD be used by default as the destination UDP port

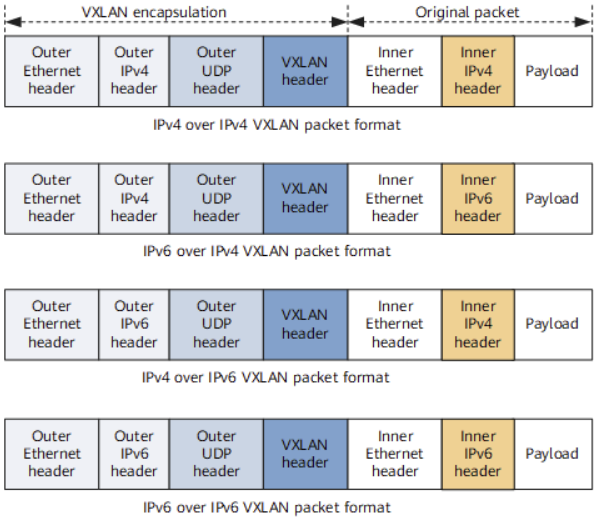
RFC7348:

VXLAN Header: This is an 8-byte field that has: - Flags (8 bits): where the I flag MUST be set to 1 for a valid VXLAN Network ID (VNI). The other 7 bits (designated "R") are reserved fields and MUST be set to zero on transmission and ignored on receipt. - VXLAN Segment ID/VXLAN Network Identifier (VNI): this is a 24-bit value used to designate the individual VXLAN overlay network on which the communicating VMs are situated. VMs in different VXLAN overlay networks cannot communicate with each other. - Reserved fields (24 bits and 8 bits): MUST be set to zero on transmission and ignored on receipt.

- Destination Port: IANA has assigned the value 4789 for the VXLAN UDP port, and this value SHOULD be used by default as the destination UDP port. Some early implementations of VXLAN have used other values for the destination port. To enable interoperability with these implementations, the destination port SHOULD be configurable. - Source Port: It is recommended that the UDP source port number be calculated using a hash of fields from the inner packet -- one example being a hash of the inner Ethernet frame's headers. This is to enable a level of entropy for the Equal-Cost Multipath (ECMP)/load- balancing of the VM-to-VM traffic across the VXLAN overlay. When calculating the UDP source port number in this manner, it is RECOMMENDED that the value be in the dynamic/private port range 49152-65535 [RFC6335]. Mahalingam, et al. Informational [Page 10] RFC 7348 VXLAN August 2014 - UDP Checksum: It SHOULD be transmitted as zero. When a packet is received with a UDP checksum of zero, it MUST be accepted for decapsulation. Optionally, if the encapsulating end point includes a non-zero UDP checksum, it MUST be correctly calculated across the entire packet including the IP header, UDP header, VXLAN header, and encapsulated MAC frame. When a decapsulating end point receives a packet with a non-zero checksum, it MAY choose to verify the checksum value. If it chooses to perform such verification, and the verification fails, the packet MUST be dropped. If the decapsulating destination chooses not to perform the verification, or performs it successfully, the packet MUST be accepted for decapsulation.

VXLAN packet format

The infrastructure network on which VXLAN tunnels are established is called the underlay network, and the service network carried over VXLAN tunnels are called the overlay network. The following combinations of underlay and overlay networks exist in VXLAN scenarios.



VXLAN Packet Forwarding Mechanism

Packets forwarded on a VXLAN network are classified into two types by forwarding mode:

- broadcast, unknown unicast, and multicast (**BUM**) packets
- known unicast packets.

Unknown-unicast traffic happens when a switch receives unicast traffic intended to be delivered to a destination that is not in its forwarding information base.

BUM packet forwarding

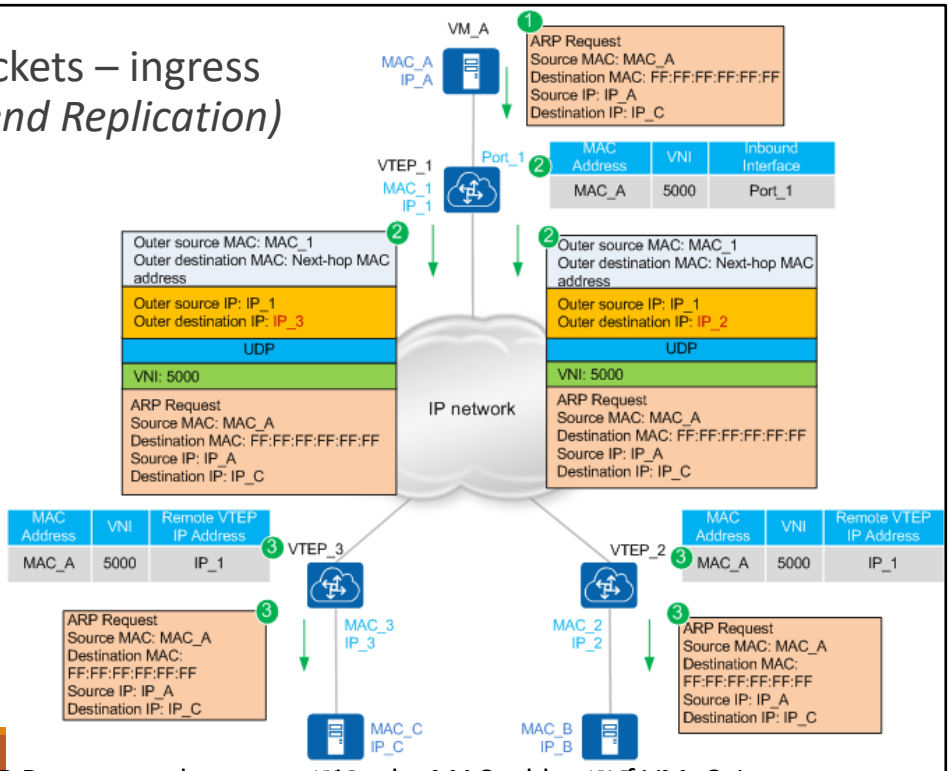
BUM packets can be forwarded in

- ingress replication
- multicast replication modes
- centralized replication

The most important problem in VXLAN implementation is the determination of MAC-to-VTEP tables

As with all switches, each leaf node creates a table. The table contains the MAC addresses of the devices connected to the leaf node's ports. What is different in VXLANs is the sharing of these tables. At an elementary level, the leaf nodes share their tables and associated VXLAN tunnel endpoint (VTEP) information with other leaf node VTEPs.
When a user device, for example, sends an egress packet to the leaf node to which it is connected, the leaf node checks its MAC-to-VTEP table.

Forwarding of BUM Packets – ingress replication (aka *Head-end Replication)*

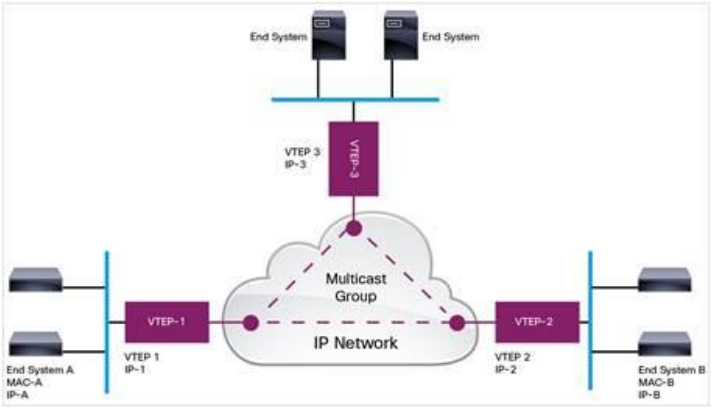


VM_A broadcasts an ARP Request packet, requesting the MAC address of VM_C. In the ARP Request packet, the source MAC address is MAC_A and the destination MAC address is all Fs.

After receiving the ARP Request packet, VTEP_1 determines the VNI and **ingress replication** list of the VXLAN tunnel. VTEP_1 replicates the ARP Request packet based on the ingress replication list, performs VXLAN encapsulation, and sends the encapsulated packet to each tunnel. In the encapsulated packet, the outer destination IP addresses are the IP addresses of the peer VTEPs (VTEP_2 and VTEP_3) respectively. The encapsulated packet is transmitted over the IP network based on the outer MAC address and IP address until it reaches the peer VTEPs.

After the packet reaches VTEP_2 and VTEP_3, VTEP_2 and VTEP_3 decapsulate it to obtain the original packet sent by VM_A. At the same time, VTEP_2 and VTEP_3 learn the mapping among MAC address of VM_A, VNI, and IP address of VTEP_1, and save the mapping in the local MAC address tables. Then, VTEP_2 and VTEP_3 send the original packet to the hosts in the corresponding Layer 2 domain.After receiving the ARP Request packet, VM_C sends an ARP Reply packet (while VM_B discards the ARP Request packet). Because VM_C has learned the MAC address of VM_A, the ARP Reply Packet is a known unicast packet.

Forwarding of BUM Packets – multicast replication



no

Forwarding of BUM Packets – multicast replication

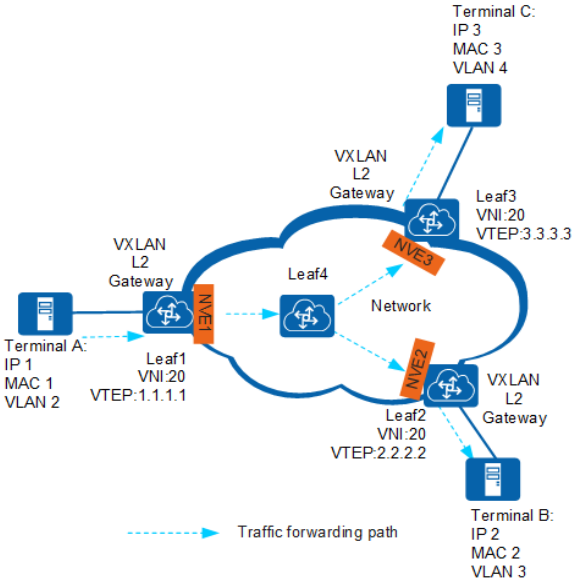
In multicast replication mode, all VTEPs with the same VNI join the same multicast group.

- **Local MAC learning**
- **Remote MAC learning**: according to the specifications of RFC 7348 it is based on a multicast routing protocol (PIM-SM / SSM or more often PIM-BiDir) on the IP underlay network
  - ➢ problems both in terms of scalability and management of the underlay network.

- In multicast replication mode, all VTEPs with the same VNI join the same multicast group. A multicast routing protocol, such as PIM, is used to create a multicast forwarding entry for the multicast group. When the source VTEP receives a BUM packet, it adds a multicast destination IP address, such as 225.0.0.1, to the BUM packet before sending the packet to the remote VTEPs based on the created multicast forwarding entry, reducing flooded packets.

Protocol-Independent Multicast (**PIM**) Sparse Mode (SM)
Bidirectional (BiDir PIM) **Protocol-Independent Multicast** (**PIM**)
source-specific multicast (*SSM*) PIM **Protocol-Independent Multicast** (**PIM**)

Forwarding of BUM Packets – multicast replication



Layer 2 packet initiated by Terminal A

DMAC	All F
SMAC	MAC1
VLAN Tag	2

Leaf 1-encapsulated VXLAN packet
Leaf1—>Leaf4

DMAC	Net MAC
SMAC	NVE1 MAC
SIP	1.1.1.1
DIP	225.0.0.1
UDP S_P	HASH
UDP D_P	4789
VNI	20
DMAC	All F
SMAC	MAC1

Packet forwarded by Leaf 4 in multicast mode
Leaf4—>Leaf2/Leaf3

DMAC	Net MAC
SMAC	NVE1 MAC
SIP	1.1.1.1
DIP	225.0.0.1
UDP S_P	HASH
UDP D_P	4789
VNI	20
DMAC	All F
SMAC	MAC1

Leaf 2/Leaf 3-decapsulated VXLAN packet

DMAC	All F
SMAC	MAC1
VLAN Tag	3

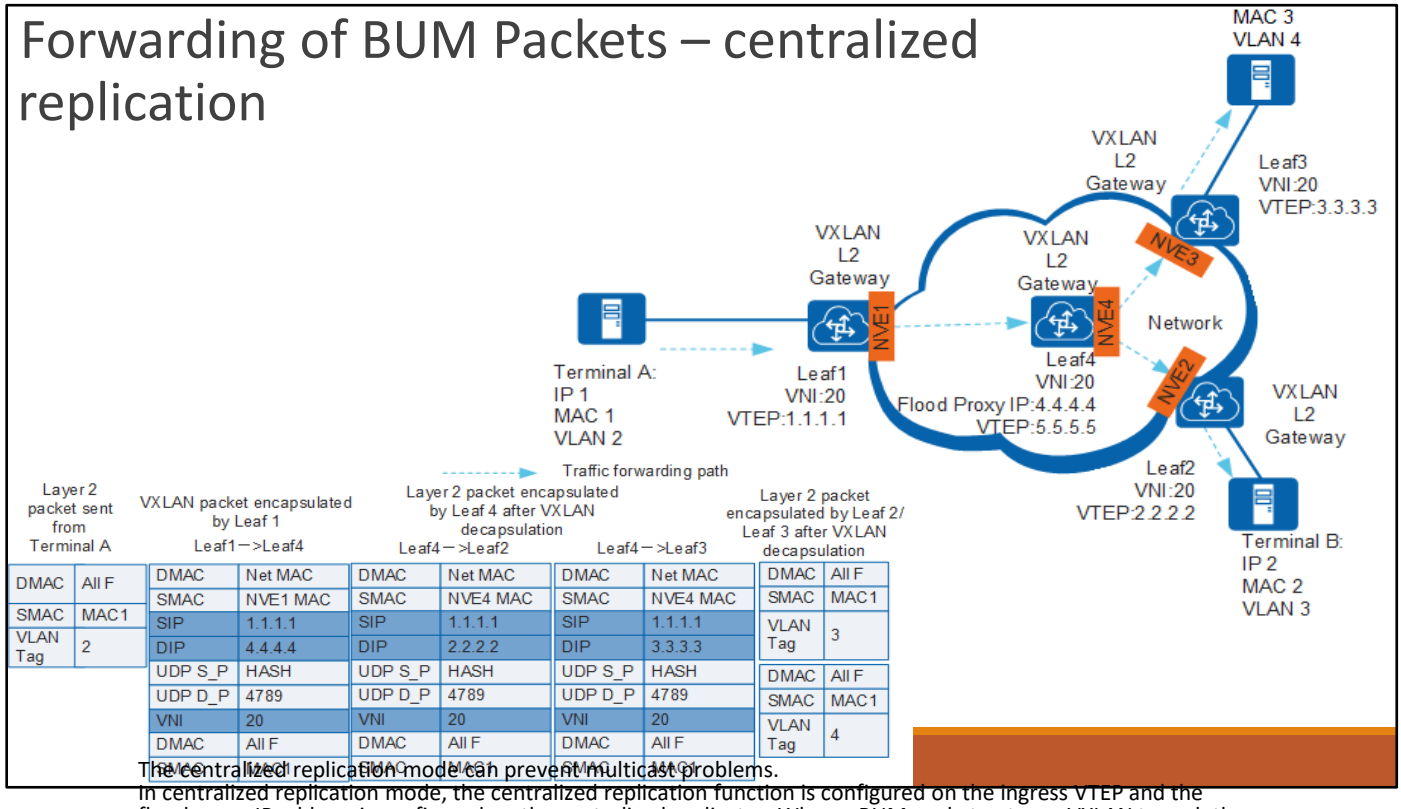
DMAC	All F
SMAC	MAC1
VLAN Tag	4

NVE (**Network Virtual Interface**): Logical interface where the encapsulation and de-encapsulation occur. The point where the VTEP function is implemented.

A bridge domain is a set of logical ports that share the same flooding or broadcast characteristics. Like a virtual LAN (VLAN), a bridge domain spans one or more ports of multiple devices.
In simple terms, a bridge domain is something that makes it possible to define a broadcast domain that is contained within a bridging device. It is a substitute for 802.1D bridge groups as well as 802.1Q VLAN bridging. The purpose of a bridge domain is to specify the broadcast domain number.

After receiving a packet from Terminal A, Leaf 1 determines the Layer 2 BD (Bridge Domain) of the packet **based on the access interface and VLAN information**.
Leaf 1's VTEP obtains the multicast replication address for the VNI based on the Layer 2 BD and performs VXLAN encapsulation. The encapsulated VXLAN packet is displayed as a multicast packet. The VTEP forwards it to Leaf 4 based on the matching multicast forwarding entry.
After receiving the multicast packet, Leaf 4 directly forwards it to Leaf 2 and Leaf 3 based on the matching multicast forwarding entry. NOTE: Leaf 4 acts as a non-gateway node and directly forwards multicast packets. Leaf 4 can be configured as a gateway node. In this case, Leaf 4 needs to forward multicast packets and decapsulate VXLAN packets. In this way, Leaf 4 is called a Bud node.
After the VTEP on Leaf 2/Leaf 3 receives the packet, it finds that the packet is a VXLAN packet after searching for the outbound interface (NVE interface) in a matching multicast forwarding entry. It checks the UDP destination port number, source and destination IP addresses, and VNI of the packet to determine the packet validity. After confirming that the packet is valid, the VTEP obtains the BD based on the VNI and decapsulates the VXLAN packet to obtain the inner Layer 2 packet.
Leaf 2/Leaf 3 checks the destination MAC address of the inner Layer 2 packet and finds it a BUM MAC address. Therefore, Leaf 2/Leaf 3 broadcasts the packet onto the network connected to the terminals (not the VXLAN tunnel side) in the Layer 2 BD. Specifically, Leaf 2/Leaf 3 finds the outbound interfaces and encapsulation information not related to the VXLAN tunnel, adds VLAN tags to the packet, and forwards the packet to Terminal B/Terminal C.

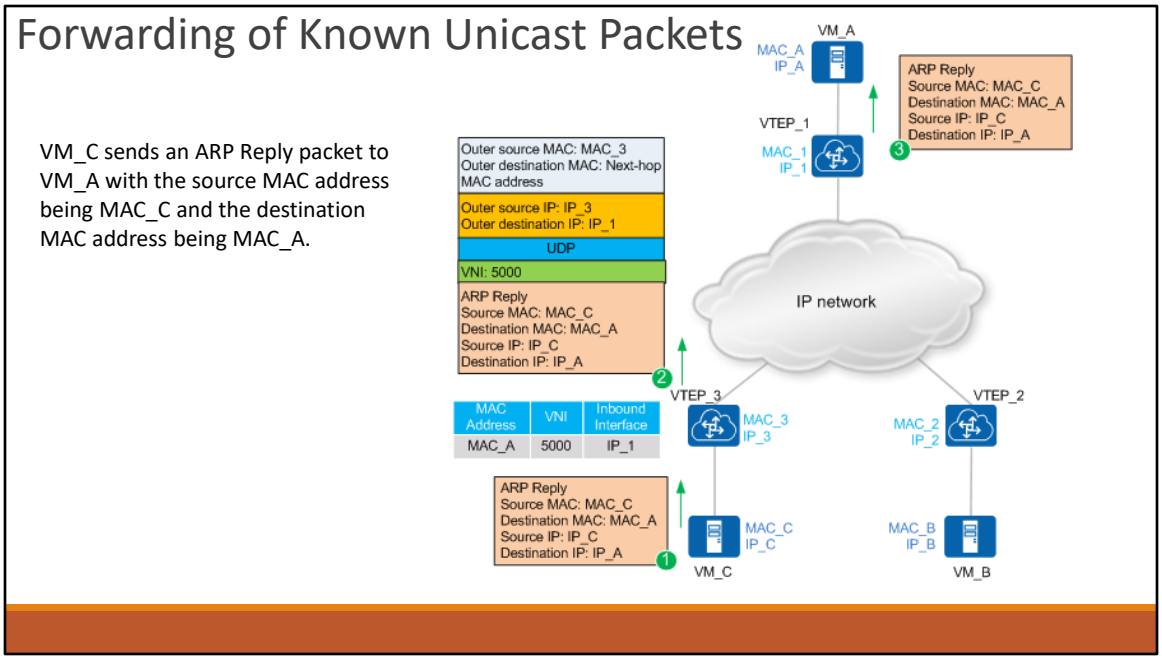
Note that it is not necessary to have a multicast tree for VXLAN, an aspect that could lead to serious scalability problems, being the VXLAN of the order of hundreds of thousands (in theory, as mentioned above, up to more than 16 million). Multicast trees shared by multiple VXLANs can be used safely, however the segregation of traffic between VXLANs is not affected since this depends exclusively on the value of the VNI. The downside to using shared multicast trees is that BUM traffic can also go to VTEPs that do not have hosts of a particular VNI, with consequent bandwidth waste.



The centralized replication mode can prevent multicast problems.

In centralized replication mode, the centralized replication function is configured on the ingress VTEP and the flood proxy IP address is configured on the centralized replicator. When a BUM packet enters a VXLAN tunnel, the ingress VTEP only needs to send one copy of the packet to the centralized replicator, reducing flooded traffic on the network. The centralized replicator is also called flood gateway. The centralized replicator decapsulates and encapsulates the BUM packet and sends it to each egress VTEP. When the BUM packet leaves the VXLAN tunnel, the egress VTEPs decapsulate the BUM packet. The figure shows the forwarding process of a BUM packet in centralized replication mode.

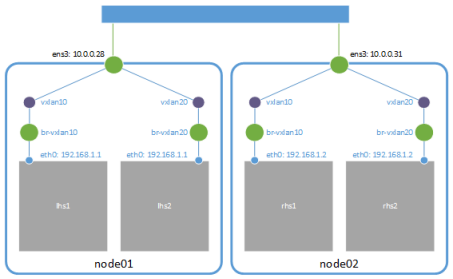
After Leaf 1 receives a packet from Terminal A, Leaf 1 determines the Layer 2 BD of the packet based on the access interface and VLAN information.
Leaf 1's VTEP obtains the centralized replication tunnel for the VNI based on the Layer 2 BD and performs VXLAN encapsulation. Leaf 1 then forwards the VXLAN packet through the outbound interface.
After Leaf 4 used as the centralized replicator receives the VXLAN packet, it checks the UDP destination port number, source and destination IP addresses, and VNI of the packet to determine the packet validity. After confirming that the packet is valid, the VTEP obtains the BD based on the VNI, decapsulates the VXLAN packet to obtain the inner Layer 2 packet, and then performs VXLAN encapsulation based on the matching ingress replication list. After VXLAN encapsulation, the outer source IP address is the VTEP address of Leaf 1. Therefore, MAC address learning among the VTEPs is not affected.
After the VTEP on Leaf 2/Leaf 3 receives the VXLAN packet, it checks the UDP destination port number, source and destination IP addresses, and VNI of the packet to determine the packet validity. After confirming that the packet is valid, the VTEP obtains the BD based on the VNI and decapsulates the VXLAN packet to obtain the inner Layer 2 packet.
Leaf 2/Leaf 3 checks the destination MAC address of the inner Layer 2 packet and finds it a BUM MAC address. Therefore, Leaf 2/Leaf 3 broadcasts the packet onto the network connected to the terminals (not the VXLAN tunnel side) in the Layer 2 BD. Specifically, Leaf 2/Leaf 3 finds the outbound interfaces and encapsulation information not related to the VXLAN tunnel, adds VLAN tags to the packet, and forwards the packet to Terminal B/Terminal C.



VM_C sends an ARP Reply packet to VM_A with the source MAC address being MAC_C and the destination MAC address being MAC_A.
After receiving the ARP Reply packet sent by VM_C, VTEP_3 determines the VNI and performs VXLAN encapsulation on the packet.Since VTEP_3 has learned the MAC address of VM_C, the outer destination IP address in the encapsulated packet is the IP address of the peer VTEP (VTEP_1). The encapsulated packet is transmitted over the IP network based on the outer MAC address and IP address until it reaches VTEP_1.
After the packet reaches VTEP_1, VTEP_1 decapsulates it to obtain the original packet sent by VM_C. Then, VTEP_1 sends the decapsulated packet to VM_A.

nah

Enjoy with Linux…



In each of the nodes

ip link add vxlan10 type vxlan id 10 group 239.1.1.1 dstport 0 dev ens3
ip link add br-vxlan10 type bridge
ip link set vxlan10 master br-vxlan10
ip link set vxlan10 up
ip link set br-vxlan10 up

ip link add vxlan20 type vxlan id 20 group 239.1.1.1 dstport 0 dev ens3
ip link add br-vxlan20 type bridge
ip link set vxlan20 master br-vxlan20
ip link set vxlan20 up
ip link set br-vxlan20 up

https://ilearnedhowto.wordpress.com/2017/02/16/how-to-create-overlay-networks-using-linux-bridges-and-vxlans/

First we create a vxlan port with VNI 10 that will use the device ens3 to multicast the UDP traffic using group 239.1.1.1 (using dstport 0 makes use of the default port).
Then we will create a bridge named br-vxlan10 to which we will bridge the previously created vxlan port.
Finally we will set both ports up

Flood and Learn (AKA Bridging)

- Flood and Learn was the original learning method for VxLAN (**all VTEPs with this VNI**). It is also known as bridging, as it's used to create virtual bridges (VNIs) between hosts.
- The other reason this is called bridging is that this is a layer-2 only solution. There is no built-in way to route between VNI's. If you need this, you must connect an external router, and let traffic pass through it.
- The flooding nature of this method limits its scalability. Also, as there is no control plane learning of VTEPs, its possible for a rogue VTEP to be added to the network. It could intercept and inject traffic.
- BUM traffic must be handled **by multicast**. There is no option for Head End replication in this case.
- While this method is still worth understanding, it is recommended to use control plane learning in production.

EVPN in VXLAN

- On VXLAN networks, VXLAN tunnels can be manually created by configuring VXLAN network identifiers (VNIs) and peer lists of VNIs. The configuration is difficult for users and requires heavy manual workload.
- **VXLAN does not provide a control plane**, and VTEP discovery and MAC addresses learning are implemented by traffic flooding on the data plane, resulting in high traffic volumes on DC networks.
- To address this problem, BGP operates in the control plane. A normal BGP deployment will share IP reachability information (routes). When integrated with VxLAN, it can also share MAC and VTEP reachability information.
- It is refered to as **Ethernet virtual private network – EVPN**.
- EVPN allows devices acting as VTEPs to exchange reachability information about endpoints as Layer 2 MAC addresses and Layer 3 IP addresses. As all the addresses are learned proactively, there's no need for flooding.
- All switches in the VxLAN topology need to run BGP EVPN. They don't all need to be running VTEPs. An example is the spine switches in the spine/leaf topology.

VXLAN does not provide the control plane, and VTEP discovery and host information (IP and MAC addresses, VNIs, and gateway VTEP IP address) learning are implemented by traffic flooding on the data plane, resulting in high traffic volumes on VXLAN networks. To address this problem, VXLAN uses EVPN as the control plane. EVPN allows VTEPs to exchange BGP EVPN routes to implement automatic VTEP discovery and host information advertisement, preventing unnecessary traffic flooding.

meh

EVPN in VXLAN

- Traditional BGP-4 peers use Update messages to exchange routing information. An Update message can advertise reachable routes with the same path attribute. These routes are carried in the **Network Layer Reachability Information (NLRI)** field.
- BGP-4 can manage only IPv4 unicast routing information, so **MP-BGP was developed to support multiple network layer protocols**, such as IPv6 and multicast. MP-BGP extends NLRI based on BGP-4. After extension, the description of the address family is added to NLRI to differentiate network layer protocols, such as the IPv6 unicast address family and VPN instance address family.
- Similarly, EVPN uses the MP-BGP mechanism and defines a **new sub-address family, EVPN address family,** in the L2VPN address family. In the EVPN address family, a new type of NLRI is added, that is, **EVPN NLRI**. EVPN NLRI defines several types of BGP EVPN routes, which can carry information such as the host IP address, MAC address, VNI, and VRF.
- After a VTEP learns the IP address and MAC address of a connected host, the VTEP can send the information to other VTEPs through MP-BGP routes. This way, learning of host IP address and MAC address information can be implemented on the control plane, suppressing traffic flooding on the data plane.

The Network Layer Reachability Information (NLRI) is exchanged between BGP routers using UPDATE messages. An NLRI is composed of a LENGTH and a PREFIX. The length is a network mask in CIDR notation (eg. /25) specifying the number of network bits, and the prefix is the Network address for that subnet

The NLRI is unique to BGP version 4 and allows BGP to carry supernetting information, as well as perform aggregation.

The NLRI would look something like one of these:

/25, 204.149.16.128
/23, 206.134.32
/8, 10

EVPN in VXLAN

Using EVPN as the control plane of VXLAN has the following advantages:

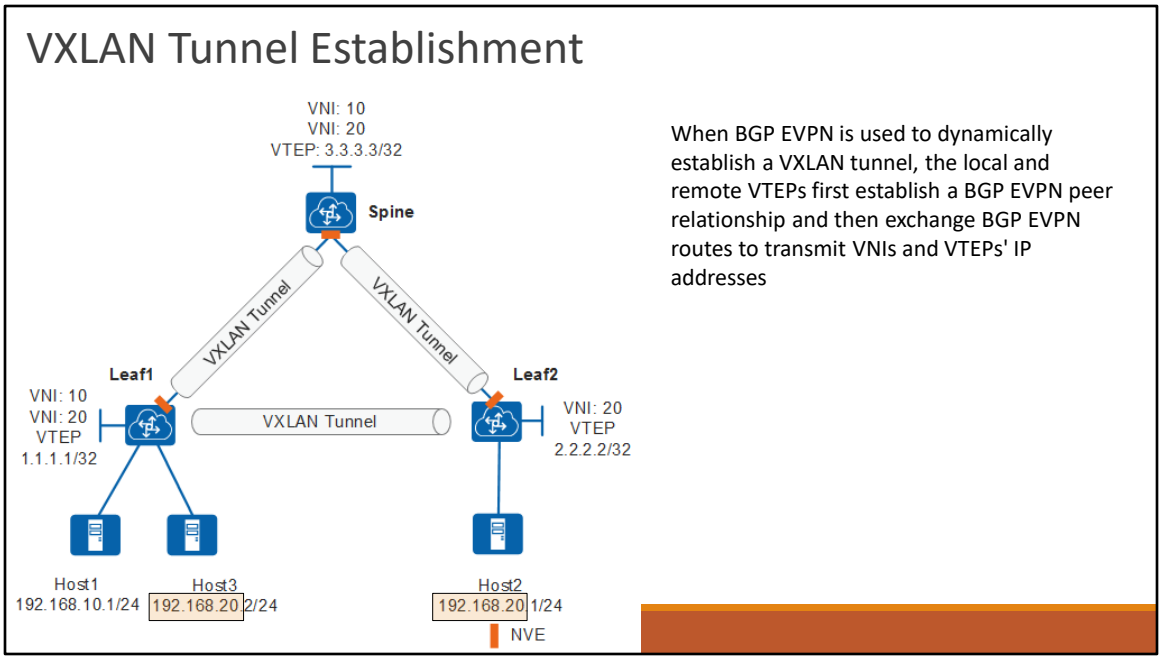
- VTEPs can be automatically discovered and VXLAN tunnels can be automatically established, simplifying network deployment and expansion.
- EVPN can advertise both Layer 2 MAC address information and Layer 3 routing information.
- Flooding traffic is reduced on the network. When a host comes online, it announces its MAC address. This can also happen at other times with Gratuitus ARP messages. The local switch will add the MAC into the local BGP database. This is then sent to its peers as a BGP update.

- Five Types of EVPN Routes (RFC 7432):

Type 1	Ethernet auto-discovery (A-D) route
Type 2	MAC/IP advertisement route
Type 3	Inclusive multicast Ethernet tag route
Type 4	Ethernet segment route
Type 5	IP prefix route

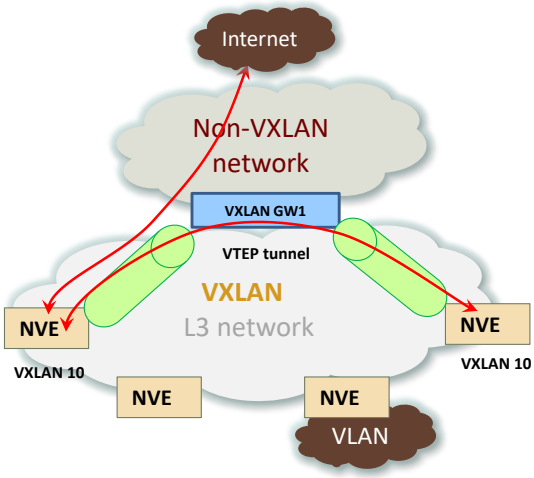
https://support.huawei.com/enterprise/en/doc/EDOC1100168670#:~:text=In%20the%20EVPN%20address%20family,address%2C%20VNI%2C%20and%20VRF.

ripetitivo, recap breve



A VXLAN tunnel is identified by a pair of VTEP IP addresses. During VXLAN tunnel establishment, the local and remote VTEPs attempt to obtain the IP addresses of each other. A VXLAN tunnel can be established if the IP addresses obtained are reachable at Layer 3. When BGP EVPN is used to dynamically establish a VXLAN tunnel, the local and remote VTEPs first establish a BGP EVPN peer relationship and then exchange BGP EVPN routes to transmit VNIs and VTEPs' IP addresses.
On the network shown in the figure, Leaf 1 connects to Host 1 and Host 3; Leaf 2 connects to Host 2; Spine functions as a Layer 3 gateway. To allow Host 3 and Host 2 to communicate, establish a VXLAN tunnel between Leaf 1 and Leaf 2. To allow Host 1 and Host 2 to communicate, establish a VXLAN tunnel between Leaf 1 and Spine and between Spine and Leaf 2. Although Host 1 and Host 3 both connect to Leaf 1, they belong to different subnets and must communicate through the Layer 3 gateway (Spine). Therefore, a VXLAN tunnel is also required between Leaf 1 and Spine.

VXLAN Gateways



To implement Layer 3 interworking, a Layer 3 gateway must be deployed on a VXLAN.

Similar to a VXLAN NVE, a Layer 3 VXLAN gateway provides mappings between the VXLAN packet header and IP packet header.

Different VLANs need to communicate with each other through Layer 3 gateways. Similarly, Layer 3 gateways are also required for communication between VXLANs with different VNIs.

In the typical spine-leaf VXLAN networking, Layer 3 VXLAN gateways can be classified into centralized gateways and distributed gateways based on their deployment locations.

The point where the VTEP function is implemented is referred to as NVE, *Network Virtualization Endpoint*. We can consider it in the VTEP interface.

An NVE is a functional module at the server virtualization layer, enabling VMs to use virtualization software to establish VTEP tunnels.

An NVE can also be a VXLAN-capable access switch that provides the VXLAN gateway service to multiple tenants in a centralized manner.

A VXLAN gateway can implement communication between tenants on different VXLANs, as well as between VXLAN users and non-VXLAN users. This function is similar to that of a VLANIF interface.

VXLAN Gateways

- Both the Layer 2 VXLAN gateway and Layer 3 VXLAN gateway are used to implement connection between a VXLAN networks. **Integrated Routing and Bridging (IRB)** is supported in VTEPs when BGP is used. This means that each switch with a VTEP can also behave as a router.
- VXLAN gateways can be deployed in centralized or distributed mode.
- **Centralized VXLAN Gateway Mode**
  - In this mode, Layer 3 gateways are configured on one device. Traffic across network segments is forwarded through Layer 3 gateways to implement centralized traffic management
- **Distributed VXLAN Gateway Mode**
  - In the distributed VXLAN gateway networking, each leaf node functions as a VTEP and as a Layer 3 VXLAN gateway.

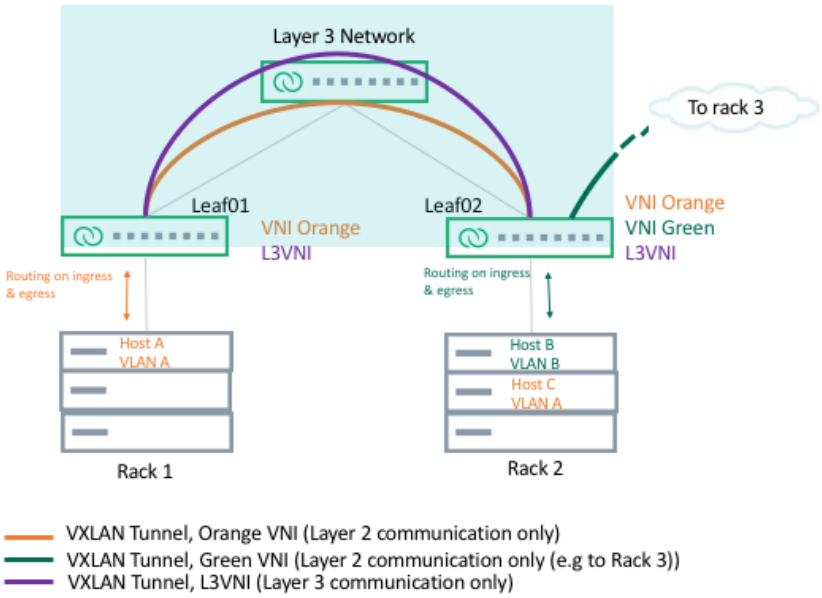
VNI EVPN Types

Two types of VNI's are used which is one for L2 operations and one for L3 operations. They are referred to as layer-2 VNI's (L2VNI) or layer-3 VNI's (L3VNI). BGP distributes both of these to all peers.

- **L2VNI** is the bridge domain. This is for bridging hosts on the same layer-2 segment. Essentially, it is the VxLAN equivalent of a VLAN. It is recommended to keep this one-to-one relation between L2VNI and VLAN's
- **L3VNI** can be used to route between L2VNI's, so this will have an IP associated which is used for routing purposes. The ingress or egress VTEP can perform routing. This is called **Symmetric IRB**. Another form of routing called **Asymmetric IRB**, uses the ingress VTEP for routing and bridging, while the egress VTEP can only do bridging.

MAC-VRF: A Virtual Routing and Forwarding table for storing Media Access Control (MAC) addresses on a VTEP for a specific tenant.
The L2VNI is in charge of forwarding traffic within the same VNI ("VLAN") between two switches. This is part of the overlay, and from a user's perspective the network is behaving as one big switch.

VNI Types - symmetric EVPN IRB



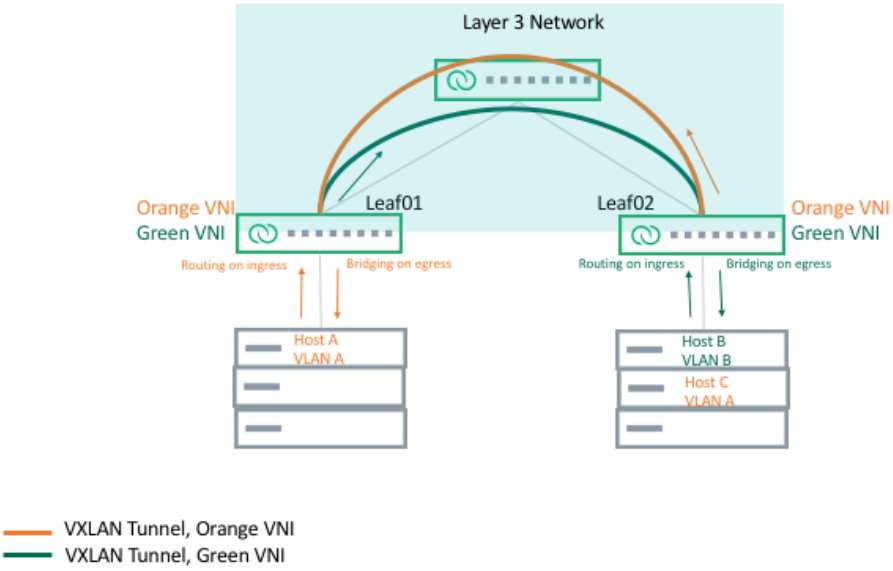
https://cumulusnetworks.com/blog/asymmetric-vs-symmetric-model/

The symmetric model routes and bridges on both the ingress and the egress leafs. This results in bi-directional traffic being able to travel on the same VNI, hence the symmetric name. However, **a new specialty transit VNI is used for all routed VXLAN traffic, called the L3VNI**. **All traffic that needs to be routed will be routed onto the L3VNI**, tunneled across the layer 3 Infrastructure, routed off the L3VNI to the appropriate VLAN and ultimately bridged to the destination.

Now consider the scenario with a symmetric model, as shown above. Let's say Host A on VLAN A needs to communicate with Host B on VLAN B. Since the destination is a different subnet from Host A, Host A sends the frame to its default gateway, which is Leaf01. Leaf01 recognizes that the destination MAC address is itself and will use the routing table to route the packet to the L3VNI and nexthop Leaf02. The VXLAN-encapsulated packet will have the egress leaf's MAC as the destination MAC address and this L3VNI as the VNI. Leaf02 performs VXLAN decapsulation and recognizes that the destination MAC address is itself and routes the packet on to the destination VLAN, to reach the destination host. The return traffic will be routed similarly over the same L3VNI.

With symmetric model, the leaf switches only need to host the VLANs and the corresponding VNIs that are located on its rack, as well as the L3VNI and its associated VLAN, since the ingress leaf switch doesn't need to know the destination VNI. The ability to host only the local VNIs (plus one extra) helps with scale. However, **the configuration is more complex as an extra VXLAN tunnel and VLAN in your network are required**. The data plane traffic is also more complex as an **extra routing hop occurs and could cause extra latency**. Multitenancy requires one L3VNI per VRF, and all switches participating in that VRF must be configured with the same L3VNI. The L3VNI is used by the egress leaf to identify the VRF in which to route the packet.

VNI Types - asymmetric EVPN IRB



When a switch performs the VTEP functions, it is referred to as a VxLAN Gateway.

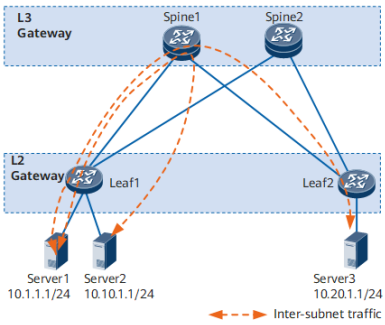
The switches can perform VxLAN encapsulation/decapsulation and can also translate the VLAN ID to VNI.

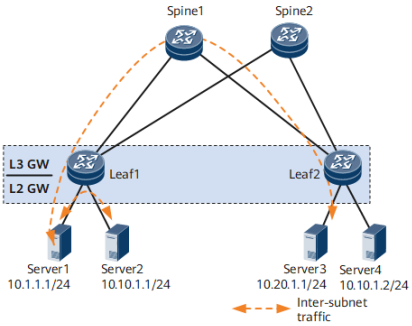
The VxLAN gateway creates the tunnel to the destination VTEP (either host or gateway), so the hosts and IP infrastructure are not aware of the existence of VxLAN.

https://developer.nvidia.com/blog/using-vxlan-routing-with-evpn-through-asymmetric-or-symmetric-models/

The asymmetric model allows **routing and bridging on the VXLAN tunnel ingress**, but **only bridging on the egress**. This results in bi-directional VXLAN traffic traveling on different VNIs in each direction (always the destination VNI) across the routed. Host A wants to communicate with Host B, which is located on a different VLAN and a different rack, thus reachable via a different VNI. Since Host B is on a different subnet from Host A, Host A sends the frame to its default gateway, which is Leaf01 (this is generally an Anycast Gateway, but we can cover that in a later post). Leaf01 recognizes that the destination MAC address is itself, looks up the routing table and routes the packet to the Green VNI while still on Leaf01. Leaf01 then tunnels the frame in the Green VNI to Leaf02. Leaf02 removes the VXLAN header from the frame, and bridges the frame to Host B. Likewise, the return traffic would behave similarly. Host B sends a frame to Leaf02. Leaf02 recognizes its own destination MAC address and routes the packet to the Orange VNI on Leaf02. The packet is tunneled within the Orange VNI to Leaf01. Leaf01 removes the VXLAN header from the frame and bridges it to Host A. With the asymmetric model, **all the required source and destination VNIs (e.g. orange and green) must be present on each leaf, even if that leaf doesn't have a host in that VLAN in its rack**. This may increase the number of IP/MAC addresses the leaf must hold, which results in somewhat limited scale. However, in many instances, all VNIs in the network are configured on all leaves anyway to allow VM mobility and to simplify configuration of the network as a whole, in which case asymmetric model is desirable.
astructure.

VXLAN Gateways

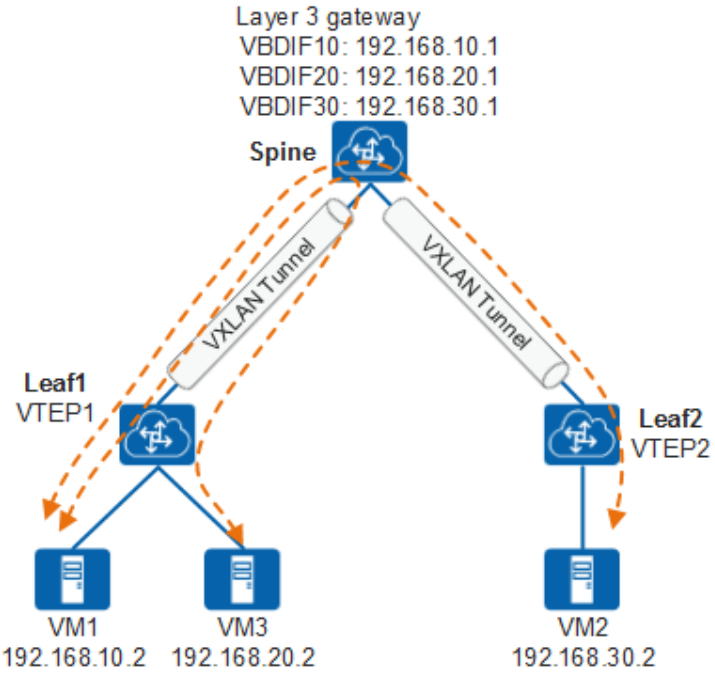




Centralized Gateway

Centralized VXLAN gateway deployment has its advantages and disadvantages.

- Advantage: Inter-segment traffic can be centrally managed, and gateway deployment and management is easy.
- Disadvantages:
  - Forwarding paths are not optimal. Inter-segment Layer 3 traffic of data centers connected to the same Layer 2 gateway must be transmitted to the centralized Layer 3 gateway for forwarding.
  - The ARP entry specification is a bottleneck. ARP entries must be generated for tenants on the Layer 3 gateway. However, only a limited number of ARP entries are allowed by the Layer 3 gateway, impeding data center network expansion.



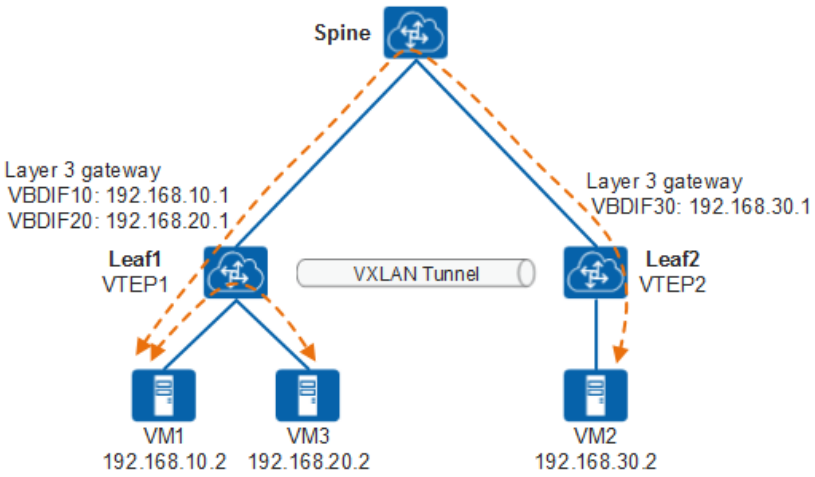
VBDIF interfaces are Layer 3 logical interfaces. A VBDIF interface is a virtual interface based on a bridge domain and supporting Layer 3 features. VBDIF interfaces implement communication between BDs, between BD and non-BD networks, and between BD and Layer 3 networks. After creating VBDIF interfaces, you can configure Layer 3 features on these interfaces.

In the centralized gateway networking, Layer 3 gateways are centrally deployed on one spine node, as shown in the following figure. Inter-subnet traffic is forwarded through the Layer 3 gateways.

In the centralized gateway networking, inter-subnet traffic can be centrally managed. Gateway deployment and management are simple, but inter-subnet traffic of VMs on the same leaf node needs to be forwarded by the spine node. Therefore, the traffic forwarding path is not optimal. In addition, all entries of terminals whose traffic is forwarded at Layer 3 need to be generated on the spine node. However, the spine node supports only a limited number of entries. When the number of tenants increases, it may become a network bottleneck.

Distributed Gateway

In the distributed gateway scenario, a control plane is required to transmit host routes between Layer 3 gateways to ensure communication between hosts. To meet this requirement, Ethernet VPN (EVPN) is introduced as the VXLAN control plane.



Distributed VXLAN gateways use the spine-leaf network. In this networking, leaf nodes, which can function as Layer 3 VXLAN gateways, are used as VTEPs to establish VXLAN tunnels. Spine nodes are unaware of the VXLAN tunnels and only forward VXLAN packets between different leaf nodes.

Layer 3 VXLAN gateways are deployed on leaf nodes to enable inter-subnet communication of VMs on the same leaf node. In this way, traffic is directly forwarded by the leaf nodes without passing through the spine node. This conserves bandwidth resources. Unlike centralized Layer 3 gateways that need to learn ARP entries of all hosts, a leaf node in the distributed VXLAN gateway scenario only needs to learn the ARP entries of hosts connected to itself. This eliminates the bottleneck caused by limited ARP entry specifications in the centralized Layer 3 gateway scenario and improves network expansion capabilities.
In the distributed gateway scenario, a control plane is required to transmit host routes between Layer 3 gateways to ensure communication between hosts. To meet this requirement, Ethernet VPN (EVPN) is introduced as the VXLAN control plane. By referring to the BGP/MPLS IP VPN mechanism, EVPN defines several types of BGP EVPN routes by extending BGP. It advertises routes on the network to implement automatic VTEP discovery and host address learning.

Sources

https://reissromoli.com/it/formazione/catalogo/38-reissblog/vxlan-pc.html
https://support.huawei.com/enterprise/en/doc/EDOC1100086966
https://support.huawei.com/enterprise/en/doc/EDOC1100092936/53f4c6d2/evpn-vxlan-fundamentals
https://support.huawei.com/enterprise/en/doc/EDOC1000180270/4d97d2db/example-for-configuring-vxlan-using-bgp-evpn-to-enable-communication-among-users-on-the-same-network-segment
https://support.huawei.com/enterprise/en/doc/EDOC1100004365/3e18cbdb/centralized-vxlan-gateway-deployment-using-bgp-evpn
https://networkdirection.net/articles/routingandswitching/vxlanoverview/vxlanaddresslearning/