

# H3C S6520X-HI[EI][SI] & S6520-SI & S5560X-HI & S5000-EI & MS4600 Switch Series VXLAN Configuration Guide

This configuration guide is applicable to the following switches and software versions:

H3C S6520X-HI switch series (Release 6308 and later)

H3C S6520X-EI switch series (Release 6308 and later)

H3C S6520X-SI switch series (Release 6308 and later)

H3C S6520-SI switch series (Release 6308 and later)

H3C S5560X-HI switch series (Release 6308 and later)

H3C S5000-EI switch series (Release 6308 and later)

H3C MS4600 switch series (Release 6308 and later)

New H3C Technologies Co., Ltd.  
<http://www.h3c.com>

Software version: Release 63xx  
Document version: 6W101-20201015

**Copyright © 2020, New H3C Technologies Co., Ltd. and its licensors**

**All rights reserved**

No part of this manual may be reproduced or transmitted in any form or by any means without prior written consent of New H3C Technologies Co., Ltd.

**Trademarks**

Except for the trademarks of New H3C Technologies Co., Ltd., any trademarks that may be mentioned in this document are the property of their respective owners.

**Notice**

The information in this document is subject to change without notice. All contents in this document, including statements, information, and recommendations, are believed to be accurate, but they are presented without warranty of any kind, express or implied. H3C shall not be liable for technical or editorial errors or omissions contained herein.

# Preface

This configuration guide describes the fundamentals and configuration procedures for VXLAN features.

This preface includes the following topics about the documentation:

- [Audience](#).
- [Conventions](#).
- [Documentation feedback](#).

## Audience

This documentation is intended for:

- Network planners.
- Field technical support and servicing engineers.
- Network administrators.

## Conventions

The following information describes the conventions used in the documentation.

### Command conventions

| Convention       | Description  |
|------------------|--|
| <b>Boldface</b>  | <b>Bold</b> text represents commands and keywords that you enter literally as shown.   |
| <i>Italic</i>    | <i>Italic</i> text represents arguments that you replace with actual values.   |
| [ ]              | Square brackets enclose syntax choices (keywords or arguments) that are optional.  |
| { x   y   ... }  | Braces enclose a set of required syntax choices separated by vertical bars, from which you select one.   |
| [ x   y   ... ]  | Square brackets enclose a set of optional syntax choices separated by vertical bars, from which you select one or none.                                  |
| { x   y   ... }* | Asterisk marked braces enclose a set of required syntax choices separated by vertical bars, from which you select a minimum of one.                      |
| [ x   y   ... ]* | Asterisk marked square brackets enclose optional syntax choices separated by vertical bars, from which you select one choice, multiple choices, or none. |
| &<1-n>           | The argument or keyword and argument combination before the ampersand (&) sign can be entered 1 to n times.  |
| #                | A line that starts with a pound (#) sign is comments.  |

### GUI conventions

| Convention      | Description   |
|-----------------|---|
| <b>Boldface</b> | Window names, button names, field names, and menu items are in Boldface. For example, the <b>New User</b> window opens; click <b>OK</b> . |
| >               | Multi-level menus are separated by angle brackets. For example, <b>File &gt; Create &gt; Folder</b> .                                     |

## Symbols

| Convention  | Description  |
|---|--|
|  <b>WARNING!</b>   | An alert that calls attention to important information that if not understood or followed can result in personal injury.   |
|  <b>CAUTION:</b>   | An alert that calls attention to important information that if not understood or followed can result in data loss, data corruption, or damage to hardware or software. |
|  <b>IMPORTANT:</b> | An alert that calls attention to essential information.  |
| <b>NOTE:</b>  | An alert that contains additional or supplementary information.  |
|  <b>TIP:</b>       | An alert that provides helpful information.  |

## Network topology icons

| Convention  | Description  |
|---|--|
|    | Represents a generic network device, such as a router, switch, or firewall.  |
|    | Represents a routing-capable device, such as a router or Layer 3 switch.   |
|    | Represents a generic switch, such as a Layer 2 or Layer 3 switch, or a router that supports Layer 2 forwarding and other Layer 2 features. |
|  | Represents an access controller, a unified wired-WLAN module, or the access controller engine on a unified wired-WLAN switch.              |
|  | Represents an access point.  |
|  | Represents a wireless terminator unit.   |
|  | Represents a wireless terminator.  |
|  | Represents a mesh access point.  |
|  | Represents omnidirectional signals.  |
|  | Represents directional signals.  |
|  | Represents a security product, such as a firewall, UTM, multiservice security gateway, or load balancing device.                           |
|  | Represents a security module, such as a firewall, load balancing, NetStream, SSL VPN, IPS, or ACG module.                                  |

## Examples provided in this document

Examples in this document might use devices that differ from your device in hardware model, configuration, or software version. It is normal that the port numbers, sample output, screenshots, and other information in the examples differ from what you have on your device.

# Documentation feedback

You can e-mail your comments about product documentation to [info@h3c.com](mailto:info@h3c.com).

We appreciate your comments.

# Contents

|   |           |
|---|-----------|
| <b>VXLAN overview</b> .....   | <b>1</b>  |
| VXLAN benefits.....   | 1         |
| VXLAN network model.....  | 1         |
| VXLAN packet format.....  | 2         |
| VXLAN working mechanisms.....   | 3         |
| Generic VXLAN network establishment and forwarding process.....             | 3         |
| VXLAN tunnel establishment and assignment.....                              | 3         |
| Assignment of traffic to VXLANs.....  | 3         |
| MAC learning.....   | 4         |
| Unicast forwarding.....   | 5         |
| Flood.....  | 6         |
| Access modes of VSIs.....   | 7         |
| ARP and ND flood suppression.....   | 7         |
| VXLAN IP gateways.....  | 8         |
| Protocols and standards.....  | 8         |
| <b>Configuring basic VXLAN features</b> .....                               | <b>9</b>  |
| Restrictions: System operating mode restriction.....                        | 9         |
| VXLAN tasks at a glance.....  | 9         |
| Prerequisites for VXLAN.....  | 9         |
| Setting the forwarding mode for VXLANs.....                                 | 9         |
| Creating a VXLAN on a VSI.....  | 10        |
| Configuring a VXLAN tunnel.....   | 11        |
| Manually assigning VXLAN tunnels to a VXLAN.....                            | 12        |
| Assigning customer frames to a VSI.....                                     | 12        |
| Restrictions and guidelines for configuring traffic assignment methods..... | 12        |
| Mapping a static Ethernet service instance to a VSI.....                    | 13        |
| Mapping dynamic Ethernet service instances to VSIs.....                     | 13        |
| Configuring VLAN-based VXLAN assignment.....                                | 15        |
| Managing MAC address entries.....   | 15        |
| About MAC address entry management.....                                     | 15        |
| Configuring static MAC address entries.....                                 | 16        |
| Disabling remote-MAC address learning.....                                  | 16        |
| Enabling local-MAC logging.....   | 16        |
| Setting the destination UDP port number of VXLAN packets.....               | 17        |
| Disabling flooding for a VSI.....   | 17        |
| Enabling ARP flood suppression.....   | 18        |
| Enabling VXLAN packet statistics.....                                       | 18        |
| Enabling packet statistics for a VSI.....                                   | 18        |
| Enabling packet statistics for an AC.....                                   | 19        |
| Enabling packet statistics for VXLAN tunnels.....                           | 19        |
| Display and maintenance commands for VXLANs.....                            | 20        |
| VXLAN configuration examples.....   | 21        |
| Example: Configuring a unicast-mode VXLAN.....                              | 21        |
| <b>Configuring VXLAN IP gateways</b> .....                                  | <b>26</b> |
| About VXLAN IP gateways.....  | 26        |
| VXLAN IP gateways separated from VTEPs.....                                 | 26        |
| Centralized VXLAN IP gateway deployment.....                                | 27        |
| Distributed VXLAN IP gateway deployment.....                                | 28        |
| VXLAN IP gateway tasks at a glance.....                                     | 33        |
| Prerequisites for VXLAN IP gateway configuration.....                       | 33        |
| Configuring a centralized VXLAN IP gateway.....                             | 33        |
| Restrictions and guidelines.....  | 33        |
| Configuring a gateway interface on a centralized VXLAN IP gateway.....      | 33        |
| Assigning a subnet to a VSI.....  | 34        |
| Configuring a distributed VXLAN IP gateway.....                             | 34        |

|   |           |
|---|-----------|
| Restrictions and guidelines for distributed VXLAN IP gateway configuration .....        | 34        |
| Configuring a gateway interface on a distributed VXLAN IP gateway .....                 | 35        |
| Enabling dynamic ARP or ND entry synchronization for distributed VXLAN IP gateways..... | 35        |
| Assigning a subnet to a VSI.....  | 36        |
| Managing ARP entries and ND entries .....   | 36        |
| Adding a static ARP entry .....   | 36        |
| Disabling remote ARP or ND learning for VXLANs.....                                     | 37        |
| Configuring a VSI interface .....   | 37        |
| Configuring optional parameters for a VSI interface .....                               | 37        |
| Restoring the default settings of the VSI interface.....                                | 38        |
| Display and maintenance commands for VXLAN IP gateways.....                             | 38        |
| VXLAN IP gateway configuration examples.....  | 39        |
| Example: Configuring a centralized VXLAN IP gateway.....                                | 39        |
| Example: Configuring distributed VXLAN IPv4 gateways .....                              | 44        |
| Example: Configuring distributed VXLAN IPv6 gateways .....                              | 54        |
| <b>Configuring the VTEP as an OVSDB VTEP .....</b>                                      | <b>65</b> |
| About OVSDB VTEP.....   | 65        |
| Protocols and standards .....   | 65        |
| Restrictions and guidelines: OVSDB VTEP configuration.....                              | 65        |
| OVSDB VTEP tasks at a glance .....  | 65        |
| Prerequisites for OVSDB VTEP configuration .....  | 66        |
| Setting up an OVSDB connection to a controller .....                                    | 66        |
| About OVSDB connection types .....  | 66        |
| Restrictions and guidelines for OVSDB controller connection setup .....                 | 66        |
| Prerequisites for OVSDB controller connection setup .....                               | 66        |
| Configuring active SSL connection settings.....   | 66        |
| Configuring passive SSL connection settings.....  | 67        |
| Configuring active TCP connection settings .....  | 67        |
| Configuring passive TCP connection settings.....  | 67        |
| Enabling the OVSDB server.....  | 68        |
| Enabling the OVSDB VTEP service.....  | 68        |
| Specifying a global source address for VXLAN tunnels.....                               | 68        |
| Specifying a VTEP access port.....  | 69        |
| Disabling the ACLs issued by the OVSDB controller .....                                 | 69        |
| OVSDB VTEP configuration examples .....   | 69        |
| Example: Configuring a unicast-mode VXLAN .....   | 69        |

# VXLAN overview

Virtual eXtensible LAN (VXLAN) is a MAC-in-UDP technology that provides Layer 2 connectivity between distant network sites across an IP network. VXLAN is typically used in data centers and the access layer of campus networks for multitenant services.

The device supports only IPv4-based VXLAN. IPv6-based VXLAN is not supported.

## VXLAN benefits

VXLAN provides the following benefits:

- **Support for more virtual switched domains than VLANs**—Each VXLAN is uniquely identified by a 24-bit VXLAN ID. The total number of VXLANs can reach 16777216 ( $2^{24}$ ). This specification makes VXLAN a better choice than 802.1Q VLAN to isolate traffic for user terminals.
- **Easy deployment and maintenance**—VXLAN requires deployment only on the edge devices of the transport network. Devices in the transport network perform typical Layer 3 forwarding.

## VXLAN network model

As shown in [Figure 1](#), a VXLAN is a virtual Layer 2 network (known as the overlay network) built on top of an existing physical Layer 3 network (known as the underlay network). The overlay network encapsulates inter-site Layer 2 frames into VXLAN packets and forwards the packets to the destination along the Layer 3 forwarding paths provided by the underlay network. The underlay network is transparent to tenants, and geographically dispersed sites of a tenant are merged into a Layer 2 network.

The transport edge devices assign user terminals to different VXLANs, and then forward traffic between sites for user terminals by using VXLAN tunnels. Supported user terminals include PCs, wireless terminals, and VMs on servers.

---

### NOTE:

This document uses VMs as examples to describe the mechanisms of VXLAN. The mechanisms do not differ between different kinds of user terminals.

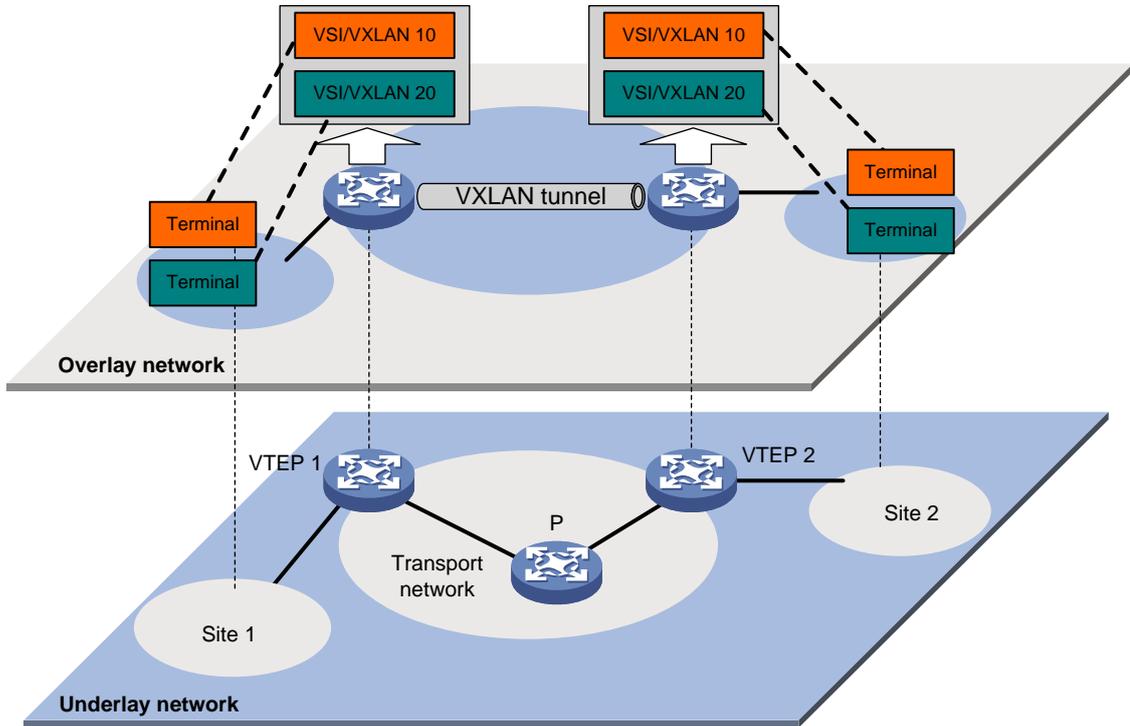
---

The transport edge devices are VXLAN tunnel endpoints (VTEP). The VTEP implementation of the device uses ACs, VSIs, and VXLAN tunnels to provide VXLAN services.

- **VSI**—A virtual switch instance is a virtual Layer 2 switched domain. Each VSI provides switching services only for one VXLAN. VSIs learn MAC addresses and forward frames independently of one another. VMs in different sites have Layer 2 connectivity if they are in the same VXLAN.
- **Attachment circuit (AC)**—An AC is a physical or virtual link that connects a VTEP to a local site. Typically, ACs are Ethernet service instances that are associated with the VSI of a VXLAN. Traffic received from an AC is assigned to the VSI associated with the AC. Ethernet service instances are created on site-facing Layer 2 interfaces. An Ethernet service instance matches a list of custom VLANs by using a frame match criterion.
- **VXLAN tunnel**—Logical point-to-point tunnels between VTEPs over the transport network. Each VXLAN tunnel can trunk multiple VXLANs.

VTEPs encapsulate VXLAN traffic in the VXLAN, outer UDP, and outer IP headers. The devices in the transport network forward VXLAN traffic only based on the outer IP header.

Figure 1 VXLAN network model

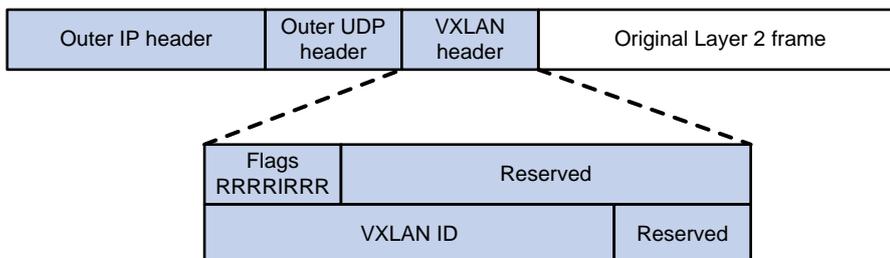


## VXLAN packet format

As shown in Figure 2, a VTEP encapsulates a frame in the following headers:

- **8-byte VXLAN header**—VXLAN information for the frame.
  - **Flags**—If the I bit is 1, the VXLAN ID is valid. If the I bit is 0, the VXLAN ID is invalid. All other bits are reserved and set to 0.
  - **24-bit VXLAN ID**—Identifies the VXLAN of the frame. It is also called the virtual network identifier (VNI).
- **8-byte outer UDP header for VXLAN**—The default VXLAN destination UDP port number is 4789.
- **20-byte outer IP header**—Valid addresses of VTEPs or VXLAN multicast groups on the transport network. Devices in the transport network forward VXLAN packets based on the outer IP header.

Figure 2 VXLAN packet format



# VXLAN working mechanisms

## Generic VXLAN network establishment and forwarding process

The VTEP uses the following process to establish the VXLAN network and forward an inter-site frame:

1. Discovers remote VTEPs, establishes VXLAN tunnels, and assigns the VXLAN tunnels to VXLANs.
2. Assigns the frame to its matching VXLAN if the frame is sent between sites.
3. Performs MAC learning on the VXLAN's VSI.
4. Forwards the frame through VXLAN tunnels.

This section describes this process in detail. For intra-site frames in a VSI, the system performs typical Layer 2 forwarding, and it processes 802.1Q VLAN tags as described in "[Access modes of VSIs.](#)"

## VXLAN tunnel establishment and assignment

To provide Layer 2 connectivity for a VXLAN between two sites, you must create a VXLAN tunnel between the sites and assign the tunnel to the VXLAN.

### VXLAN tunnel establishment

VXLAN supports manual and automatic VXLAN tunnel establishment.

- **Manual creation**—Manually create a VXLAN tunnel interface, and specify the tunnel source and destination IP addresses on the peer VTEPs.
- **Automatic creation**—Configure Ethernet Virtual Private Network (EVPN) to automatically discover VTEPs and set up VXLAN tunnels. For more information about EVPN, see *EVPN Configuration Guide*.

### VXLAN tunnel assignment

VXLAN supports manual and automatic VXLAN tunnel assignment.

- **Manual assignment**—Manually assign VXLAN tunnels to VXLANs.
- **Automatic assignment**—Run EVPN to automatically assign VXLAN tunnels to VXLANs. For more information about EVPN, see *EVPN Configuration Guide*.

## Assignment of traffic to VXLANs

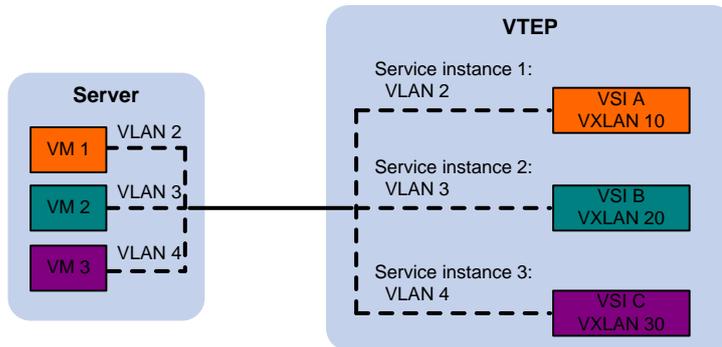
### Traffic from the local site to a remote site

The VTEP uses the following methods to assign customer frames to a VXLAN:

- **Ethernet service instance-to-VSI mapping**—This method uses the frame match criterion of an Ethernet service instance to match a list of VLANs on a site-facing Layer 2 interface. The frame match criterion specifies the characteristics of traffic from the VLANs, such as tagging status and VLAN IDs. The VTEP assigns customer traffic to a VXLAN by mapping the Ethernet service instance to a VSI.
- **VLAN-based VXLAN assignment**—This method maps a VLAN to a VXLAN. The VTEP assigns all frames of the VLAN to the VXLAN.

As shown in Figure 3, Ethernet service instance 1 matches VLAN 2 and is mapped to VSI A (VXLAN 10). When a frame from VLAN 2 arrives, the VTEP assigns the frame to VXLAN 10, and looks up VSI A's MAC address table for the outgoing interface.

**Figure 3 Identifying traffic from the local site**



### Traffic from a remote site to the local site

When a frame arrives at a VXLAN tunnel, the VTEP uses the VXLAN ID in the frame to identify its VXLAN.

## MAC learning

The VTEP performs source MAC learning on the VSI as a Layer 2 switch.

- For traffic from the local site to the remote site, the VTEP learns the source MAC address before VXLAN encapsulation.
- For traffic from the remote site to the local site, the VTEP learns the source MAC address after removing the VXLAN header.

A VSI's MAC address table includes the following types of MAC address entries:

- **Local MAC**—MAC entries learned from the local site. The outgoing interfaces for the MAC address entries are site-facing interfaces.
  - **Static**—Manually added MAC entries.
  - **Dynamic**—Dynamically learned MAC entries.
- **Remote MAC**—MAC entries learned from a remote site, including static and dynamic MAC entries. The outgoing interfaces for the MAC addresses are VXLAN tunnel interfaces.
  - **Static**—Manually added MAC entries.
  - **Dynamic**—MAC entries learned in the data plane from incoming traffic on VXLAN tunnels. The learned MAC addresses are contained in the inner Ethernet header.
  - **BGP EVPN**—MAC entries advertised through BGP EVPN. For more information, see *EVPN Configuration Guide*.
  - **OpenFlow**—MAC entries issued by a remote controller through OpenFlow. For more information, see *OpenFlow Configuration Guide*.
  - **OVSDDB**—MAC entries issued by a remote controller through OVSDDB.

The following shows the priority order of different types of remote MAC address entries:

- a. Static MAC address entries, and MAC address entries issued by a remote controller through OpenFlow or OVSDDB. These types of entries have the same priority and overwrite each other.
- b. MAC address entries advertised through BGP EVPN.
- c. Dynamic MAC address entries.

# Unicast forwarding

## Intra-site unicast forwarding

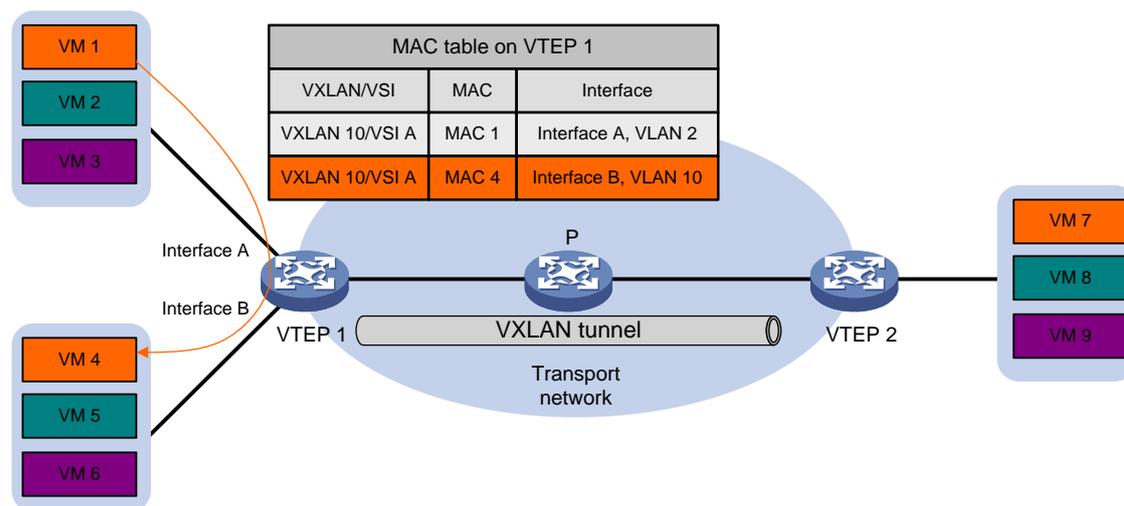
The VTEP uses the following process to forward a known unicast frame within a site:

1. Identifies the VSI of the frame.
2. Looks up the destination MAC address in the VSI's MAC address table for the outgoing interface.
3. Sends the frame out of the matching outgoing interface.

As shown in Figure 4, VTEP 1 forwards a frame from VM 1 to VM 4 within the local site in VLAN 10 as follows:

1. Identifies that the frame belongs to VSI A when the frame arrives at Interface A.
2. Looks up the destination MAC address (MAC 4) in the MAC address table of VSI A for the outgoing interface.
3. Sends the frame out of the matching outgoing interface (Interface B) to VM 4 in VLAN 10.

Figure 4 Intra-site unicast

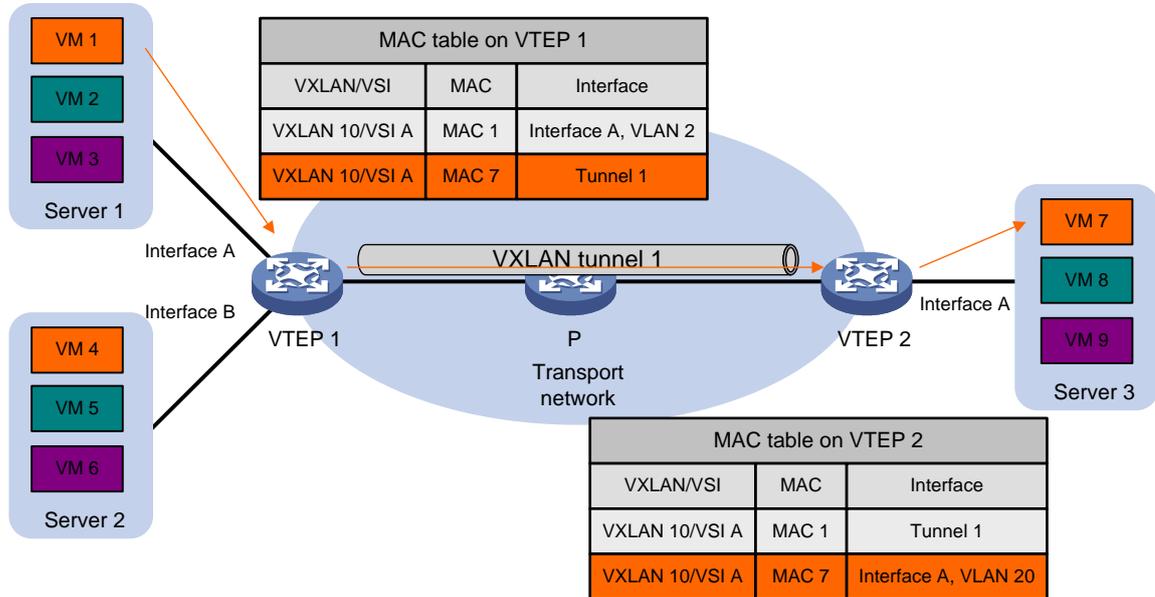


## Inter-site unicast forwarding

The following process (see Figure 5) applies to a known unicast frame between sites:

1. The source VTEP encapsulates the Ethernet frame in the VXLAN/UDP/IP header. In the outer IP header, the source IP address is the source VTEP's VXLAN tunnel source IP address. The destination IP address is the VXLAN tunnel destination IP address.
2. The source VTEP forwards the encapsulated packet out of the outgoing VXLAN tunnel interface found in the VSI's MAC address table.
3. The intermediate transport devices (P devices) forward the frame to the destination VTEP by using the outer IP header.
4. The destination VTEP removes the headers on top of the inner Ethernet frame. It then performs MAC address table lookup in the VXLAN's VSI to forward the frame out of the matching outgoing interface.

**Figure 5 Inter-site unicast**



## Flood

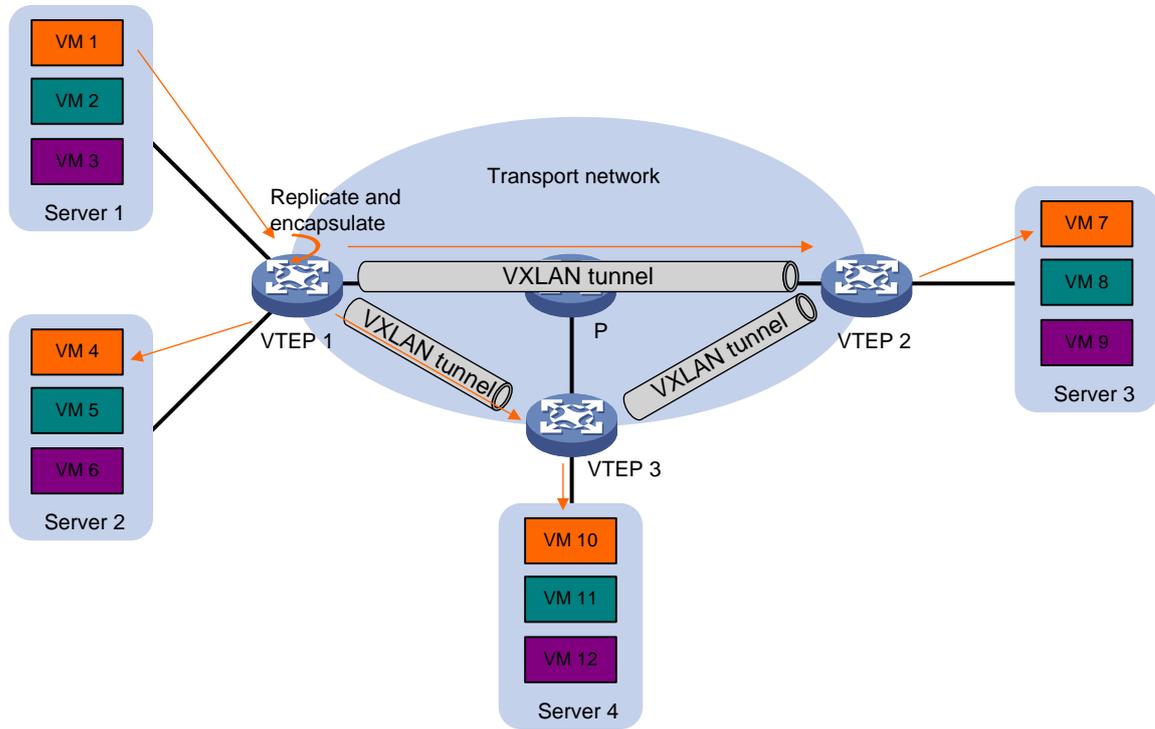
The source VTEP floods a broadcast, multicast, or unknown unicast frame to all site-facing interfaces and VXLAN tunnels in the VXLAN, except for the incoming interface. Each destination VTEP floods the inner Ethernet frame to all site-facing interfaces in the VXLAN. To avoid loops, the destination VTEPs do not flood the frame back to VXLAN tunnels.

VXLAN supports unicast mode (also called head-end replication).

### Unicast mode (head-end replication)

As shown in Figure 6, the source VTEP replicates the flood frame, and then sends one replica to the destination IP address of each VXLAN tunnel in the VXLAN.

**Figure 6 Unicast mode**



## Access modes of VSIs

The access mode of a VSI determines how the VTEP processes the 802.1Q VLAN tags in the Ethernet frames. The device supports the VLAN access mode. In this mode, Ethernet frames received from or sent to the local site must contain 802.1Q VLAN tags.

- For an Ethernet frame received from the local site, the VTEP removes all its 802.1Q VLAN tags before forwarding the frame.
- For an Ethernet frame destined for the local site, the VTEP adds 802.1Q VLAN tags to the frame before forwarding the frame.

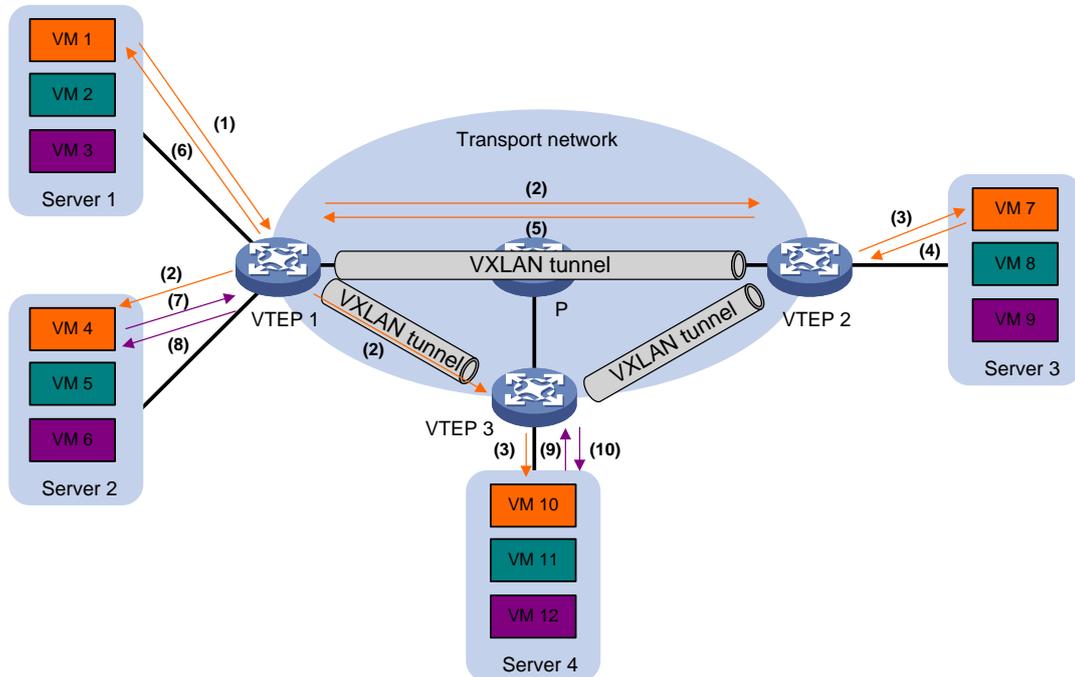
In VLAN access mode, VXLAN packets sent between sites do not contain 802.1Q VLAN tags. You can use different 802.1Q VLANs to provide the same service in different sites.

## ARP and ND flood suppression

ARP or ND flood suppression reduces ARP request broadcasts or ND request multicasts by enabling the VTEP to reply to ARP or ND requests on behalf of VMs.

As shown in [Figure 7](#), this feature snoops ARP or ND packets to populate the ARP or ND flood suppression table with local and remote MAC addresses. If an ARP or ND request has a matching entry, the VTEP replies to the request on behalf of the VM. If no match is found, the VTEP floods the request to both local and remote sites.

**Figure 7 ARP flood suppression**



The following uses ARP flood suppression as an example to explain the flood suppression workflow:

1. VM 1 sends an ARP request to obtain the MAC address of VM 7.
2. VTEP 1 creates a suppression entry for VM 1, and floods the ARP request in the VXLAN.
3. VTEP 2 and VTEP 3 de-encapsulate the ARP request. The VTEPs create a suppression entry for VM 1, and broadcast the request in the local site.
4. VM 7 sends an ARP reply.
5. VTEP 2 creates a suppression entry for VM 7 and forwards the ARP reply to VTEP 1.
6. VTEP 1 de-encapsulates the ARP reply, creates a suppression entry for VM 7, and forwards the ARP reply to VM 1.
7. VM 4 sends an ARP request to obtain the MAC address of VM 1 or VM 7.
8. VTEP 1 creates a suppression entry for VM 4 and replies to the ARP request.
9. VM 10 sends an ARP request to obtain the MAC address of VM 1.
10. VTEP 3 creates a suppression entry for VM 10 and replies to the ARP request.

## VXLAN IP gateways

A VXLAN IP gateway provides Layer 3 forwarding services for VMs in VXLANs. A VXLAN IP gateway can be an independent device or be collocated with a VTEP. For more information about VXLAN IP gateway placement, see "[Configuring VXLAN IP gateways.](#)"

## Protocols and standards

RFC 7348, *Virtual eXtensible Local Area Network (VXLAN): A Framework for Overlaying Virtualized Layer 2 Networks over Layer 3 Networks*

# Configuring basic VXLAN features

## Restrictions: System operating mode restriction

Before you configure VXLAN settings, perform the following tasks:

1. Execute the `switch-mode 1` command in system view to set the system operating mode to VXLAN.
2. Save the running configuration to the next-startup configuration file.
3. Reboot the device.

For more information about the system operating mode, see device management in *Fundamentals Configuration Guide*.

## VXLAN tasks at a glance

To configure basic VXLAN settings, perform the following tasks on VTEPs:

1. [Setting the forwarding mode for VXLANs](#)
2. [Creating a VXLAN on a VSI](#)
3. [Configuring a VXLAN tunnel](#)
4. [Manually assigning VXLAN tunnels to a VXLAN](#)
5. [Assigning customer frames to a VSI](#)
6. (Optional.) [Managing MAC address entries](#)
  - o [Configuring static MAC address entries](#)
  - o [Disabling remote-MAC address learning](#)
  - o [Enabling local-MAC logging](#)
7. (Optional.) [Setting the destination UDP port number of VXLAN packets](#)
8. (Optional.) Reducing flood traffic in the transport network
  - o [Disabling flooding for a VSI](#)
  - o [Enabling ARP flood suppression](#)
9. [Enabling VXLAN packet statistics](#)

## Prerequisites for VXLAN

Configure a routing protocol on the devices in the transport network to make sure the VTEPs can reach one another.

## Setting the forwarding mode for VXLANs

### About VXLAN forwarding modes

A VXLAN tunnel supports the following modes:

- **Layer 3 forwarding mode**—The device uses the ARP table (IPv4 network) or ND table (IPv6 network) to forward traffic for VXLANs.
- **Layer 2 forwarding mode**—The device uses the MAC address table to forward traffic for VXLANs.

If the device is a VTEP, enable Layer 2 forwarding for VXLANs. If the device is a VXLAN IP gateway, enable Layer 3 forwarding for VXLANs. For more information about VXLAN IP gateways, see "[Configuring VXLAN IP gateways.](#)"

## Restrictions and guidelines

You must delete all VSIs, VSI interfaces, and VXLAN tunnel interfaces before you can change the forwarding mode. As a best practice, finish VXLAN network planning and determine the VXLAN forwarding mode of each device before performing your configuration, and set the VXLAN forwarding mode before other VXLAN settings.

## Procedure

1. Enter system view.  
**system-view**
2. Set the forwarding mode of VXLANs.
  - o Enable Layer 2 forwarding.  
**undo vxlan ip-forwarding**
  - o Enable Layer 3 forwarding.  
**vxlan ip-forwarding**By default, Layer 3 forwarding is enabled for VXLANs.

# Creating a VXLAN on a VSI

1. Enter system view.  
**system-view**
2. Enable L2VPN.  
**l2vpn enable**  
By default, L2VPN is disabled.
3. Create a VSI and enter VSI view.  
**vsi** *vsi-name*
4. Enable the VSI.  
**undo shutdown**  
By default, a VSI is enabled.
5. Create a VXLAN and enter VXLAN view.  
**vxlan** *vxlan-id*  
You can create only one VXLAN on a VSI.  
The VXLAN ID must be unique for each VSI.
6. (Optional.) Configure VSI parameters:
  - a. Return to VSI view.  
**quit**
  - b. Configure a VSI description.  
**description** *text*  
By default, a VSI does not have a description.
  - c. Set the MTU for the VSI.  
**mtu** *size*  
The default MTU for a VSI is 1500 bytes.
  - d. Enable MAC address learning for the VSI.

**mac-learning enable**

By default, MAC address learning is enabled for a VSI.

# Configuring a VXLAN tunnel

## About manual VXLAN tunnel creation

When you manually create a VXLAN tunnel, specify addresses on the local VTEP and the remote VTEP as the tunnel source and destination addresses, respectively.

## Restrictions and guidelines

As a best practice, do not configure multiple VXLAN tunnels to use the same source and destination IP addresses.

Make sure the following VXLAN tunnels are not associated with the same VXLAN when they have the same tunnel destination IP address:

- A VXLAN tunnel automatically created by EVPN.
- A manually created VXLAN tunnel.

For more information about EVPN, see *EVPN Configuration Guide*.

This task provides basic VXLAN tunnel configuration. For more information about tunnel configuration and commands, see *Layer 3—IP Services Configuration Guide* and *Layer 3—IP Services Command Reference*.

## Procedure

1. Enter system view.

**system-view**

2. (Optional.) Specify a global source IP address for VXLAN tunnels.

**tunnel global source-address** *ip-address*

By default, no global source IP address is specified for VXLAN tunnels.

A VXLAN tunnel uses the global source address if you do not specify a source interface or source address for the tunnel.

3. Create a VXLAN tunnel interface and enter tunnel interface view.

**interface tunnel** *tunnel-number* **mode vxlan**

The endpoints of a tunnel must use the same tunnel mode.

4. Specify a source address for the tunnel. Choose one of the following methods:

- Specify a source IP address for the tunnel.

**source** *ipv4-address*

The specified IP address is used in the outer IP header of tunneled VXLAN packets.

- Specify a source interface for the tunnel.

**source** *interface-type interface-number*

The primary IP address of the specified interface is used in the outer IP header of tunneled VXLAN packets.

By default, no source IP address or source interface is specified for a tunnel.

Do not perform this step if you are using OVSD for VXLAN tunnel management.

For a multicast-mode VXLAN, the source IP address cannot be a loopback interface's address, and the source interface cannot be a loopback interface.

5. Specify a destination IP address for the tunnel.

**destination** *ipv4-address*

By default, no destination IP address is specified for a tunnel.

Specify the remote VTEP's IP address. This IP address will be the destination IP address in the outer IP header of tunneled VXLAN packets.

# Manually assigning VXLAN tunnels to a VXLAN

## About VXLAN tunnel manual assignment

To provide Layer 2 connectivity for a VXLAN between two sites, you must assign the VXLAN tunnel between the sites to the VXLAN.

You can assign multiple VXLAN tunnels to a VXLAN, and configure a VXLAN tunnel to trunk multiple VXLANs. For a unicast-mode VXLAN, the system floods unknown unicast, multicast, and broadcast traffic to each tunnel associated with the VXLAN.

## Restrictions and guidelines

For full Layer 2 connectivity in the VXLAN, make sure the VXLAN contains the VXLAN tunnel between each pair of sites in the VXLAN.

## Procedure

1. Enter system view.  
`system-view`
2. Enter VSI view.  
`vsi vsi-name`
3. Enter VXLAN view.  
`vxlan vxlan-id`
4. Assign VXLAN tunnels to the VXLAN.  
`tunnel { tunnel-number [ backup-tunnel tunnel-number ] | all }`  
By default, a VXLAN does not contain any VXLAN tunnels.

| Parameter                                | Description   |
|--|---|
| <code>backup-tunnel tunnel-number</code> | Specifies a backup tunnel. When the primary VXLAN tunnel is operating correctly, the backup VXLAN tunnel does not forward traffic. When the primary VXLAN tunnel goes down, traffic is switched to the backup VXLAN tunnel. |

# Assigning customer frames to a VSI

## Restrictions and guidelines for configuring traffic assignment methods

VLAN-based VXLAN assignment is mutually exclusive with the manually created Ethernet service instances. To create these Ethernet service instances, you must first disable VLAN-based VXLAN assignment by using the `undo vxlan vlan-based` command. To enable VLAN-based VXLAN assignment, you must first delete all Ethernet service instances.

# Mapping a static Ethernet service instance to a VSI

## About static Ethernet service instance mappings

A static Ethernet service instance matches a list of VLANs on a site-facing interface. The VTEP assigns customer traffic from the VLANs to a VXLAN by mapping the Ethernet service instance to a VSI.

## Restrictions and guidelines

Link aggregation group membership is mutually exclusive with Ethernet service instance-to-VSI mappings on a Layer 2 interface. Do not associate a VSI with an Ethernet service instance on a Layer 2 interface if the interface is in an aggregation group. Do not assign a Layer 2 interface to an aggregation group if the interface is configured with Ethernet service instances of VSIs.

Ethernet service instance bindings of VSIs are mutually exclusive with QinQ and VLAN mapping on a Layer 2 Ethernet interface or Layer 2 aggregate interface. Do not configure these features simultaneously on the same interface. Otherwise, the features cannot take effect.

Do not configure VLAN mapping, QinQ, or MAC-based VLAN on a Layer 2 Ethernet interface or Layer 2 aggregate interface that acts as the traffic outgoing interface of a VXLAN tunnel. Otherwise, the features cannot take effect.

For information about the frame match criterion configuration restrictions and guidelines of Ethernet service instances, see *VXLAN Command Reference*.

## Procedure

1. Enter system view.  
**system-view**
  2. Enter interface view.
    - o Enter Layer 2 Ethernet interface view.  
**interface** *interface-type* *interface-number*
    - o Enter Layer 2 aggregate interface view.  
**interface bridge-aggregation** *interface-number*
  3. Create an Ethernet service instance and enter Ethernet service instance view.  
**service-instance** *instance-id*
  4. Configure a frame match criterion. Choose one of the following options:
    - o Match frames tagged with the specified outer 802.1Q VLAN ID.  
**encapsulation s-vid** *vlan-id*
    - o Match any 802.1Q untagged frames.  
**encapsulation untagged**

By default, an Ethernet service instance does not contain a frame match criterion.
  5. Map the Ethernet service instance to a VSI.  
**xconnect vsi** *vsi-name* [ **access-mode** *vlan* ] [ **track** *track-entry-number*&<1-3> ]
- By default, an Ethernet service instance is not mapped to any VSI.

# Mapping dynamic Ethernet service instances to VSIs

## About dynamic Ethernet service instance mappings

The 802.1X or MAC authentication feature can use the authorization VSI, the guest VSI, the Auth-Fail VSI, and the critical VSI to control the access of users to network resources. When assigning a user to a VSI, 802.1X or MAC authentication sends the VXLAN feature the VSI

information and the user's access information, including access interface, VLAN, and MAC address. Then the VXLAN feature creates a dynamic Ethernet service instance for the user and maps it to the VSI. For more information about 802.1X authentication and MAC authentication, see *Security Configuration Guide*.

A dynamic Ethernet service instance supports the following traffic match modes:

- **VLAN-based mode**—Matches frames by VLAN ID.
- **MAC-based mode**—Matches frames by VLAN ID and source MAC address.

By default, dynamic Ethernet service instances use VLAN-based traffic match mode. To use MAC-based traffic match mode for dynamic Ethernet service instances, you must enable MAC authentication or 802.1X authentication that uses MAC-based access control.

### Configuring the VLAN-based traffic match mode

To use the VLAN-based traffic match mode, configure 802.1X authentication or MAC authentication and perform one of the following tasks:

- Configure the guest VSI, Auth-Fail VSI, or critical VSI on the 802.1X- or MAC authentication-enabled interface.
- Issue an authorization VSI to an 802.1X or MAC authentication user from a remote AAA server.

Then, the device will automatically create a dynamic Ethernet service instance for the 802.1X or MAC authentication user and map the Ethernet service instance to a VSI.

For more information about configuring 802.1X authentication and MAC authentication, see *Security Configuration Guide*.

### Configuring the MAC-based traffic match mode

1. Enter system view.  
**system-view**
2. Enter interface view.
  - Enter Layer 2 Ethernet interface view.  
**interface** *interface-type* *interface-number*
  - Enter Layer 2 aggregate interface view.  
**interface bridge-aggregation** *interface-number*
3. Enable MAC-based traffic match mode for dynamic Ethernet service instances on the interface.  
**mac-based ac**

By default, MAC-based traffic match mode is disabled for dynamic Ethernet service instances, and VLAN-based traffic match mode is used for dynamic Ethernet service instances.

For more information about this command, see *VXLAN Command Reference*.

4. Enable MAC authentication or 802.1X authentication that uses MAC-based access control.  
To use the MAC-based traffic match mode, configure MAC authentication or 802.1X authentication that uses MAC-based access control and perform one of the following tasks:
  - Configure the guest VSI, Auth-Fail VSI, or critical VSI on the 802.1X- or MAC authentication-enabled interface.
  - Issue an authorization VSI to an 802.1X or MAC authentication user from a remote AAA server.

Then, the device will automatically create a dynamic Ethernet service instance for the 802.1X or MAC authentication user and map the Ethernet service instance to a VSI.

For more information about configuring 802.1X authentication and MAC authentication, see *Security Configuration Guide*.

# Configuring VLAN-based VXLAN assignment

## About VLAN-based VXLAN assignment

VLAN-based VXLAN assignment enables the device to assign all traffic of a VLAN to a VXLAN. If you enable this feature and map a VLAN to a VXLAN, the device automatically performs the following operations:

1. Creates an Ethernet service instance that uses the VLAN ID as its instance ID on each interface in the VLAN. The matching outer VLAN ID of the Ethernet service instances is the VLAN ID.
2. Maps the Ethernet service instances to the VSI of the VXLAN.

## Restrictions and guidelines

Do not configure this feature together with EVPN distributed relay. For information about EVPN distributed relay, see *EVPN Configuration Guide*.

If you map a VLAN to a VXLAN, the VTEP cannot perform non-VXLAN Layer 2 forwarding in the VLAN. Also, the VLAN interface of the VLAN cannot perform Layer 3 forwarding.

The Ethernet service instance creation or deletion time is affected by the number of VLANs mapped to a VXLAN and the number of trunk ports assigned to the VLANs. The larger the numbers, the longer the time. During AC creation or deletion, other operations are queued.

## Prerequisites

Use the `vxlan` command to create the VXLAN to which a VLAN is mapped.

## Procedure

1. Enter system view.  
`system-view`
2. Enable VLAN-based VXLAN assignment.  
`vxlan vlan-based`  
By default, VLAN-based VXLAN assignment is disabled.
3. Create a VLAN and enter VLAN view.  
`vlan vlan-id`  
Do not specify VLAN 1 for VLAN-based VXLAN assignment.
4. Map the VLAN to a VXLAN.  
`vxlan vni vxlan-id`  
By default, a VLAN is not mapped to a VXLAN.  
Do not map a VLAN to the L3 VXLAN ID of EVPN.

# Managing MAC address entries

## About MAC address entry management

Local-MAC address entries are dynamically learned. You can log local MAC addresses and local-MAC changes.

Remote-MAC address entries have a variety of types, including manually added entries and dynamically learned entries.

# Configuring static MAC address entries

## Restrictions and guidelines

Do not configure static remote-MAC entries for VXLAN tunnels that are automatically established by using EVPN.

- EVPN re-establishes VXLAN tunnels if the transport-facing interface goes down and then comes up. If you have configured static remote-MAC entries, the entries are deleted when the tunnels are re-established.
- EVPN re-establishes VXLAN tunnels if you perform configuration rollback. If the tunnel IDs change during tunnel re-establishment, configuration rollback fails, and static remote-MAC entries on the tunnels cannot be restored.

For more information about EVPN, see *EVPN Configuration Guide*.

## Procedure

1. Enter system view.

```
system-view
```

2. Add a static remote-MAC address entry.

```
mac-address static mac-address interface tunnel tunnel-number vsi  
vsi-name
```

For the setting to take effect, make sure the VSI's VXLAN has been specified on the VXLAN tunnel.

# Disabling remote-MAC address learning

## About disabling remote-MAC address learning

When network attacks occur, disable remote-MAC address learning to prevent the device from learning incorrect remote MAC addresses. You can manually add static remote-MAC address entries.

## Procedure

1. Enter system view.

```
system-view
```

2. Disable remote-MAC address learning.

```
vxlan tunnel mac-learning disable
```

By default, remote-MAC address learning is enabled.

# Enabling local-MAC logging

## About local-MAC logging

When the local-MAC logging feature is enabled, the VXLAN module immediately sends a log message with its local MAC addresses to the information center. When a local MAC address is added or removed, a log message is also sent to the information center to report the local-MAC change.

With the information center, you can set log message filtering and output rules, including output destinations. For more information about configuring the information center, see *Network Management and Monitoring Configuration Guide*.

## Procedure

1. Enter system view.

**system-view**

2. Enable local-MAC logging.

**vxlan local-mac report**

By default, local-MAC logging is disabled.

## Setting the destination UDP port number of VXLAN packets

1. Enter system view.

**system-view**

2. Set a destination UDP port for VXLAN packets.

**vxlan udp-port** *port-number*

By default, the destination UDP port number is 4789 for VXLAN packets.

You must configure the same destination UDP port number on all VTEPs in a VXLAN.

## Disabling flooding for a VSI

### About VSI flooding

By default, the VTEP floods broadcast, unknown unicast, and unknown multicast frames received from the local site to the following interfaces in the frame's VXLAN:

- All site-facing interfaces except for the incoming interface.
- All VXLAN tunnel interfaces.

When receiving broadcast, unknown unicast, and unknown multicast frames on VXLAN tunnel interfaces, the device floods the frames to all site-facing interfaces in the frames' VXLAN.

To confine a kind of flood traffic, disable flooding for that kind of flood traffic on the VSI bound to the VXLAN.

To exclude a remote MAC address from the remote flood suppression done by using this feature, enable selective flood for the MAC address. The VTEP will flood the frames destined for the MAC address to remote sites.

### Procedure

1. Enter system view.

**system-view**

2. Enter VSI view.

**vsi** *vsi-name*

3. Disable flooding for the VSI.

**flooding disable** { **all** | { **broadcast** | **unknown-multicast** | **unknown-unicast** } \* } [ **all-direction** ]

By default, flooding is enabled for a VSI.

The **broadcast**, **unknown-multicast**, and **unknown-unicast** keywords are available only in Release 6316P01 and later.

4. (Optional.) Enable selective flood for a MAC address.

**selective-flooding mac-address** *mac-address*

By default, selective flood is disabled for all MAC addresses.

This command is available only in Release 6316P01 and later.

# Enabling ARP flood suppression

## Restrictions and guidelines

The aging timer is fixed at 25 minutes for ARP flood suppression entries. If the suppression table is full, the VTEP stops learning new entries. For the VTEP to learn new entries, you must wait for old entries to age out, or use the **reset arp suppression vsi** command to clear the table.

If the **flooding disable** command is configured, set the MAC aging timer to a higher value than the aging timer for ARP flood suppression entries on all VTEPs. This setting prevents the traffic blackhole that occurs when a MAC address entry ages out before its ARP flood suppression entry ages out. To set the MAC aging timer, use the **mac-address timer** command.

When remote ARP learning is disabled for VXLANs, the device does not use ARP flood suppression entries to respond to ARP requests received on VXLAN tunnels.

## Procedure

1. Enter system view.  
**system-view**
2. Enter VSI view.  
**vsi vsi-name**
3. Enable ARP flood suppression.  
**arp suppression enable**  
By default, ARP flood suppression is disabled.

# Enabling VXLAN packet statistics

## Enabling packet statistics for a VSI

### Restrictions and guidelines

To display the packet statistics for a VSI, use the **display l2vpn vsi verbose** command in any view.

To clear the packet statistics for a VSI, use the **reset l2vpn statistics vsi** command in user view.

### Procedure

1. Enter system view.  
**system-view**
2. Enter VSI view.  
**vsi vsi-name**
3. Enable packet statistics for the VSI.  
**statistics enable**  
By default, the packet statistics feature is disabled for all VSIs.

# Enabling packet statistics for an AC

## Restrictions and guidelines

For the **statistics enable** command to take effect on an Ethernet service instance, you must configure a frame match criterion for the Ethernet service instance and map it to a VSI. When you modify the frame match criterion or VSI mapping, the packet statistics of the instance are cleared.

## Enabling packet statistics for an Ethernet service instance

1. Enter system view.  
**system-view**
2. Enter interface view.
  - o Enter Layer 2 Ethernet interface view.  
**interface interface-type interface-number**
  - o Enter Layer 2 aggregate interface view.  
**interface bridge-aggregation interface-number**
3. Enter Ethernet service instance view.  
**service-instance instance-id**
4. Enable packet statistics for the Ethernet service instance.  
**statistics enable**

By default, the packet statistics feature is disabled for all Ethernet service instances.

## Enabling packet statistics for Ethernet service instances of a VLAN

1. Enter system view.  
**system-view**
2. Enter VLAN view.  
**vlan vlan-id**
3. Enable packet statistics for Ethernet service instances of the VLAN.  
**ac statistics enable**

By default, packet statistics are disabled for Ethernet service instances of a VLAN.

This feature enables packet statistics for the Ethernet service instances that are automatically created for VLAN-based VXLAN assignment. Before you enable this feature, you must use the **vxlan vlan-based** command to enable VLAN-based VXLAN assignment.

# Enabling packet statistics for VXLAN tunnels

## About packet statistics of VXLAN tunnels

VXLAN tunnels can be manually or automatically created. For manually created VXLAN tunnels, you can enable packet statistics on a per-tunnel interface basis. For automatically created VXLAN tunnels, you can enable packet statistics globally in system view.

To display the packet statistics for a VXLAN tunnel, use the **display interface tunnel** command in any view.

To clear the packet statistics for a VXLAN tunnel, use the **reset counters interface tunnel** command in user view.

## Enabling packet statistics for a manually created VXLAN tunnel

1. Enter system view.  
**system-view**

2. Enter VXLAN tunnel interface view.  
**interface tunnel** *tunnel-number* [ **mode vxlan** ]
3. Enable packet statistics for the tunnel.  
**statistics enable**

By default, the packet statistics feature is disabled for manually created VXLAN tunnels.

### Enabling packet statistics for automatically created VXLAN tunnels

1. Enter system view.  
**system-view**
2. Enable packet statistics for automatically created VXLAN tunnels.  
**tunnel statistics vxlan auto** [ **destination** *ipv4-address* ]

By default, the packet statistics feature is disabled for automatically created VXLAN tunnels.

This command enables the device to collect packet statistics for VXLAN tunnels that are automatically created by EVPN or OVSDb. For more information about EVPN, see *EVPN Configuration Guide*. For more information about OVSDb, see "[Configuring the VTEP as an OVSDb VTEP](#)."

## Display and maintenance commands for VXLANs

Execute **display** commands in any view and **reset** commands in user view.

| Task  | Command   |
|---|---|
| Display ARP flood suppression entries on VSIs.        | <b>display arp suppression vsi</b> [ <b>name</b> <i>vsi-name</i> ] [ <b>slot</b> <i>slot-number</i> ] [ <b>count</b> ]  |
| Display information about tunnel interfaces.          | <b>display interface</b> [ <b>tunnel</b> [ <i>number</i> ] ] [ <b>brief</b> [ <b>description</b>   <b>down</b> ] ]  |
| Display ND flood suppression entries.                 | <b>display ipv6 nd suppression vsi</b> [ <b>name</b> <i>vsi-name</i> ] [ <b>slot</b> <i>slot-number</i> ] [ <b>count</b> ]  |
| Display MAC address entries for VSIs.                 | <b>display l2vpn mac-address</b> [ <b>vsi</b> <i>vsi-name</i> ] [ <b>dynamic</b> ] [ <b>count</b>   <b>verbose</b> ]  |
| Display information about Ethernet service instances. | <b>display l2vpn service-instance</b> [ <b>interface</b> <i>interface-type interface-number</i> [ <b>service-instance</b> <i>instance-id</i> ] ] [ <b>verbose</b> ] |
| Display information about VSIs.                       | <b>display l2vpn vsi</b> [ <b>name</b> <i>vsi-name</i> ] [ <b>verbose</b> ]   |
| Display VXLAN tunnel information for VXLANs.          | <b>display vxlan tunnel</b> [ <b>vxlan</b> <i>vxlan-id</i> ]  |
| Clear ARP flood suppression entries on VSIs.          | <b>reset arp suppression vsi</b> [ <b>name</b> <i>vsi-name</i> ]  |
| Clear dynamic MAC address entries on VSIs.            | <b>reset l2vpn mac-address</b> [ <b>vsi</b> <i>vsi-name</i> ]   |
| Clear packet statistics on ACs.                       | <b>reset l2vpn statistics ac</b> [ <b>interface</b> <i>interface-type interface-number</i> <b>service-instance</b> <i>instance-id</i> ]                             |
| Clear packet statistics on VSIs.                      | <b>reset l2vpn statistics vsi</b> [ <b>name</b> <i>vsi-name</i> ]   |

---

**NOTE:**

For more information about the `display interface tunnel` command, see tunneling commands in *Layer 3—IP Services Command Reference*.

---

# VXLAN configuration examples

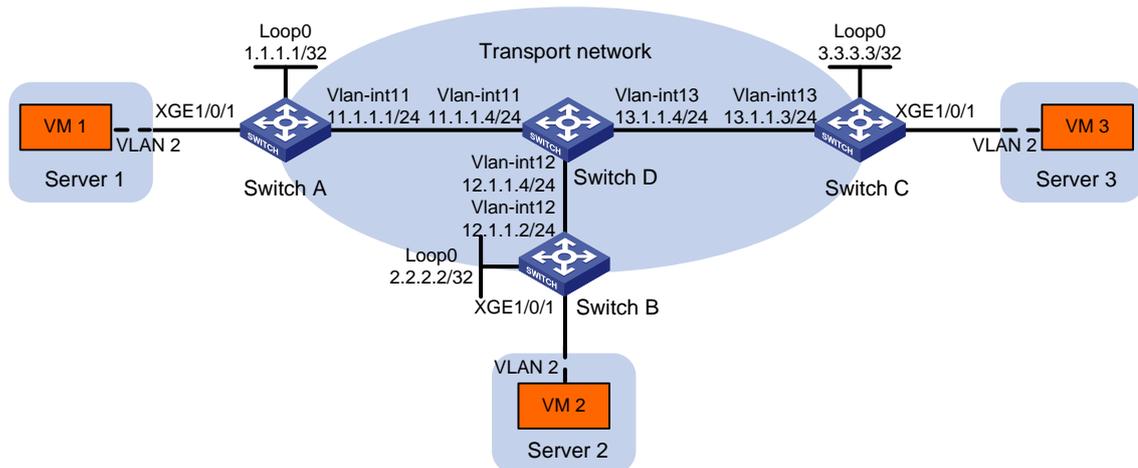
## Example: Configuring a unicast-mode VXLAN

### Network configuration

As shown in [Figure 8](#):

- Configure VXLAN 10 as a unicast-mode VXLAN on Switch A, Switch B, and Switch C to provide Layer 2 connectivity for the VMs across the network sites.
- Manually establish VXLAN tunnels and assign the tunnels to VXLAN 10.
- Enable remote-MAC address learning.

**Figure 8 Network diagram**



### Procedure

1. Set the system operating mode to VXLAN on Switch A, Switch B, and Switch C. This step uses Switch A as an example.

```
<SwitchA> system-view
```

```
[SwitchA] switch-mode 1
```

Reboot device to make the configuration take effect.

```
[SwitchA] quit
```

```
<SwitchA> reboot
```

Start to check configuration with next startup configuration file, please wait..

.....DONE!

Current configuration may be lost after the reboot, save current configuration?

```
[Y/N]:y
```

This command will reboot the device. Continue? [Y/N]:y

2. Configure IP addresses and unicast routing settings:  
# Assign IP addresses to interfaces, as shown in [Figure 8](#). (Details not shown.)

# Configure OSPF on all transport network switches (Switches A through D). (Details not shown.)

### 3. Configure Switch A:

# Enable L2VPN.

```
<SwitchA> system-view
[SwitchA] l2vpn enable
```

# Enable Layer 2 forwarding for VXLANs.

```
[SwitchA] undo vxlan ip-forwarding
```

# Create VSI **vpna** and VXLAN 10.

```
[SwitchA] vsi vpna
[SwitchA-vsi-vpna] vxlan 10
[SwitchA-vsi-vpna-vxlan-10] quit
[SwitchA-vsi-vpna] quit
```

# Assign an IP address to Loopback 0. The IP address will be used as the source IP address of the VXLAN tunnels to Switch B and Switch C.

```
[SwitchA] interface loopback 0
[SwitchA-Loopback0] ip address 1.1.1.1 255.255.255.255
[SwitchA-Loopback0] quit
```

# Create a VXLAN tunnel to Switch B. The tunnel interface name is **Tunnel 1**.

```
[SwitchA] interface tunnel 1 mode vxlan
[SwitchA-Tunnel1] source 1.1.1.1
[SwitchA-Tunnel1] destination 2.2.2.2
[SwitchA-Tunnel1] quit
```

# Create a VXLAN tunnel to Switch C. The tunnel interface name is **Tunnel 2**.

```
[SwitchA] interface tunnel 2 mode vxlan
[SwitchA-Tunnel2] source 1.1.1.1
[SwitchA-Tunnel2] destination 3.3.3.3
[SwitchA-Tunnel2] quit
```

# Assign Tunnel 1 and Tunnel 2 to VXLAN 10.

```
[SwitchA] vsi vpna
[SwitchA-vsi-vpna] vxlan 10
[SwitchA-vsi-vpna-vxlan-10] tunnel 1
[SwitchA-vsi-vpna-vxlan-10] tunnel 2
[SwitchA-vsi-vpna-vxlan-10] quit
[SwitchA-vsi-vpna] quit
```

# On Ten-GigabitEthernet 1/0/1, create Ethernet service instance 1000 to match VLAN 2.

```
[SwitchA] interface ten-gigabitethernet 1/0/1
[SwitchA-Ten-GigabitEthernet1/0/1] port link-type trunk
[SwitchA-Ten-GigabitEthernet1/0/1] port trunk permit vlan 2
[SwitchA-Ten-GigabitEthernet1/0/1] service-instance 1000
[SwitchA-Ten-GigabitEthernet1/0/1-srv1000] encapsulation s-vid 2
```

# Map Ethernet service instance 1000 to VSI **vpna**.

```
[SwitchA-Ten-GigabitEthernet1/0/1-srv1000] xconnect vsi vpna
[SwitchA-Ten-GigabitEthernet1/0/1-srv1000] quit
[SwitchA-Ten-GigabitEthernet1/0/1] quit
```

### 4. Configure Switch B:

# Enable L2VPN.

```
<SwitchB> system-view
```

```

[SwitchB] l2vpn enable
# Enable Layer 2 forwarding for VXLANs.
[SwitchB] undo vxlan ip-forwarding
# Create VSI vpna and VXLAN 10.
[SwitchB] vsi vpna
[SwitchB-vsi-vpna] vxlan 10
[SwitchB-vsi-vpna-vxlan-10] quit
[SwitchB-vsi-vpna] quit
# Assign an IP address to Loopback 0. The IP address will be used as the source IP address of
the VXLAN tunnels to Switch A and Switch C.
[SwitchB] interface loopback 0
[SwitchB-Loopback0] ip address 2.2.2.2 255.255.255.255
[SwitchB-Loopback0] quit
# Create a VXLAN tunnel to Switch A. The tunnel interface name is Tunnel 2.
[SwitchB] interface tunnel 2 mode vxlan
[SwitchB-Tunnel2] source 2.2.2.2
[SwitchB-Tunnel2] destination 1.1.1.1
[SwitchB-Tunnel2] quit
# Create a VXLAN tunnel to Switch C. The tunnel interface name is Tunnel 3.
[SwitchB] interface tunnel 3 mode vxlan
[SwitchB-Tunnel3] source 2.2.2.2
[SwitchB-Tunnel3] destination 3.3.3.3
[SwitchB-Tunnel3] quit
# Assign Tunnel 2 and Tunnel 3 to VXLAN 10.
[SwitchB] vsi vpna
[SwitchB-vsi-vpna] vxlan 10
[SwitchB-vsi-vpna-vxlan-10] tunnel 2
[SwitchB-vsi-vpna-vxlan-10] tunnel 3
[SwitchB-vsi-vpna-vxlan-10] quit
[SwitchB-vsi-vpna] quit
# On Ten-GigabitEthernet 1/0/1, create Ethernet service instance 1000 to match VLAN 2.
[SwitchB] interface ten-gigabitethernet 1/0/1
[SwitchB-Ten-GigabitEthernet1/0/1] port link-type trunk
[SwitchB-Ten-GigabitEthernet1/0/1] port trunk permit vlan 2
[SwitchB-Ten-GigabitEthernet1/0/1] service-instance 1000
[SwitchB-Ten-GigabitEthernet1/0/1-srv1000] encapsulation s-vid 2
# Map Ethernet service instance 1000 to VSI vpna.
[SwitchB-Ten-GigabitEthernet1/0/1-srv1000] xconnect vsi vpna
[SwitchB-Ten-GigabitEthernet1/0/1-srv1000] quit
[SwitchB-Ten-GigabitEthernet1/0/1] quit

```

## 5. Configure Switch C:

```

# Enable L2VPN.
<SwitchC> system-view
[SwitchC] l2vpn enable
# Enable Layer 2 forwarding for VXLANs.
[SwitchC] undo vxlan ip-forwarding
# Create VSI vpna and VXLAN 10.

```

```

[SwitchC] vsi vpna
[SwitchC-vsi-vpna] vxlan 10
[SwitchC-vsi-vpna-vxlan-10] quit
[SwitchC-vsi-vpna] quit
# Assign an IP address to Loopback 0. The IP address will be used as the source IP address of
the VXLAN tunnels to Switch A and Switch B.
[SwitchC] interface loopback 0
[SwitchC-Loopback0] ip address 3.3.3.3 255.255.255.255
[SwitchC-Loopback0] quit
# Create a VXLAN tunnel to Switch A. The tunnel interface name is Tunnel 1.
[SwitchC] interface tunnel 1 mode vxlan
[SwitchC-Tunnel1] source 3.3.3.3
[SwitchC-Tunnel1] destination 1.1.1.1
[SwitchC-Tunnel1] quit
# Create a VXLAN tunnel to Switch B. The tunnel interface name is Tunnel 3.
[SwitchC] interface tunnel 3 mode vxlan
[SwitchC-Tunnel3] source 3.3.3.3
[SwitchC-Tunnel3] destination 2.2.2.2
[SwitchC-Tunnel3] quit
# Assign Tunnel 1 and Tunnel 3 to VXLAN 10.
[SwitchC] vsi vpna
[SwitchC-vsi-vpna] vxlan 10
[SwitchC-vsi-vpna-vxlan-10] tunnel 1
[SwitchC-vsi-vpna-vxlan-10] tunnel 3
[SwitchC-vsi-vpna-vxlan-10] quit
[SwitchC-vsi-vpna] quit
# On Ten-GigabitEthernet 1/0/1, create Ethernet service instance 1000 to match VLAN 2.
[SwitchC] interface ten-gigabitethernet 1/0/1
[SwitchC-Ten-GigabitEthernet1/0/1] port link-type trunk
[SwitchC-Ten-GigabitEthernet1/0/1] port trunk permit vlan 2
[SwitchC-Ten-GigabitEthernet1/0/1] service-instance 1000
[SwitchC-Ten-GigabitEthernet1/0/1-srv1000] encapsulation s-vid 2
# Map Ethernet service instance 1000 to VSI vpna.
[SwitchC-Ten-GigabitEthernet1/0/1-srv1000] xconnect vsi vpna
[SwitchC-Ten-GigabitEthernet1/0/1-srv1000] quit
[SwitchC-Ten-GigabitEthernet1/0/1] quit

```

## Verifying the configuration

1. Verify the VXLAN settings on the VTEPs. This example uses Switch A.

# Verify that the VXLAN tunnel interfaces on the VTEP are up.

```

[SwitchA] display interface tunnel 1
Tunnel1
Current state: UP
Line protocol state: UP
Description: Tunnel1 Interface
Bandwidth: 64 kbps
Maximum transmission unit: 1464
Internet protocol processing: Disabled

```

```

Last clearing of counters: Never
Tunnel source 1.1.1.1, destination 2.2.2.2
Tunnel protocol/transport UDP_VXLAN/IP
Last 300 seconds input rate: 0 bytes/sec, 0 bits/sec, 0 packets/sec
Last 300 seconds output rate: 0 bytes/sec, 0 bits/sec, 0 packets/sec
Input: 0 packets, 0 bytes, 0 drops
Output: 0 packets, 0 bytes, 0 drops

```

**# Verify that the VXLAN tunnels have been assigned to the VXLAN.**

```
[SwitchA] display l2vpn vsi verbose
```

```
VSI Name: vpna
```

```

VSI Index           : 0
VSI State           : Up
MTU                 : 1500
Bandwidth           : -
Broadcast Restrain  : -
Multicast Restrain  : -
Unknown Unicast Restrain: -
MAC Learning        : Enabled
MAC Table Limit     : -
MAC Learning rate   : -
Drop Unknown        : -
Flooding            : Enabled
Statistics          : Disabled
VXLAN ID            : 10

```

```
Tunnels:
```

| Tunnel Name | Link ID   | State | Type   | Flood proxy |
|-------------|-----------|-------|--------|-------------|
| Tunnel1     | 0x5000001 | Up    | Manual | Disabled    |
| Tunnel2     | 0x5000002 | Up    | Manual | Disabled    |

```
ACs:
```

| AC               | Link ID | State | Type   |
|------------------|---------|-------|--------|
| XGE1/0/1 srv1000 | 0       | Up    | Manual |

**# Verify that the VTEP has learned the MAC addresses of remote VMs.**

```
<SwitchA> display l2vpn mac-address
```

| MAC Address    | State   | VSI Name | Link ID/Name | Aging |
|----------------|---------|----------|--------------|-------|
| cc3e-5f9c-6cdb | Dynamic | vpna     | Tunnel1      | Aging |
| cc3e-5f9c-23dc | Dynamic | vpna     | Tunnel2      | Aging |

```

--- 2 mac address(es) found ---

```

2. Verify that VM 1, VM 2, and VM 3 can ping each other. (Details not shown.)

# Configuring VXLAN IP gateways

## About VXLAN IP gateways

The following are available IP gateway placement designs for VXLANs:

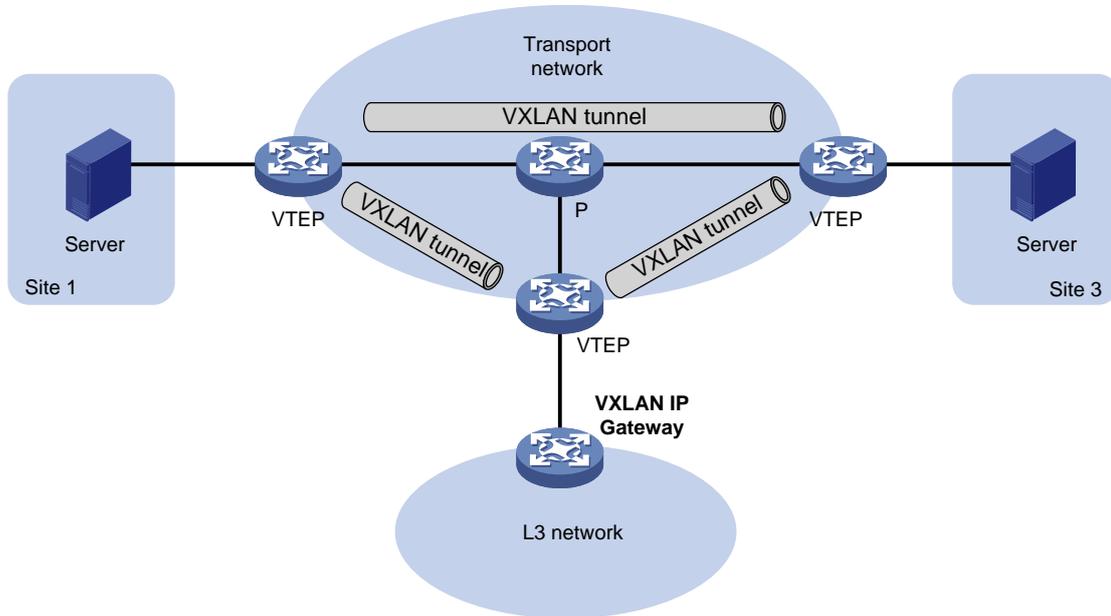
- **VXLAN IP gateways separated from VTEPs**—Use a VXLAN-unaware device as a gateway to the external network for VXLANs. On the gateway, you do not need to configure VXLAN settings.
- **VXLAN IP gateways collocated with VTEPs**—Include the following placement designs:
  - **Centralized VXLAN IP gateway deployment**—Use one VTEP to provide Layer 3 forwarding for VXLANs. Typically, the gateway-collocated VTEP connects to other VTEPs and the external network. To use this design, make sure the IP gateway has sufficient bandwidth and processing capability. Centralized VXLAN IP gateways provide services only for IPv4 networks.
  - **Centralized VXLAN gateway group deployment**—Use one VTEP group that contains redundant centralized VXLAN IP gateways to provide reliable gateway services for VXLANs.
  - **Distributed VXLAN IP gateway deployment**—Deploy one VXLAN IP gateway on each VTEP to provide Layer 3 forwarding for VXLANs at their respective sites. This design distributes the Layer 3 traffic load across VTEPs. However, its configuration is more complex than the centralized VXLAN IP gateway design. Distributed gateways can provide services for both IPv4 and IPv6 networks.

In a collocation design, the VTEPs use virtual Layer 3 VSI interfaces as gateway interfaces to provide services for VXLANs.

## VXLAN IP gateways separated from VTEPs

As shown in [Figure 9](#), an independent VXLAN IP gateway connects a Layer 3 network to a VTEP. VMs send Layer 3 traffic in Layer 2 frames to the gateway through VXLAN tunnels. When the tunneled VXLAN packets arrive, the VTEP terminates the VXLANs and forwards the inner frames to the gateway. In this gateway placement design, the VTEP does not perform Layer 3 forwarding for VXLANs.

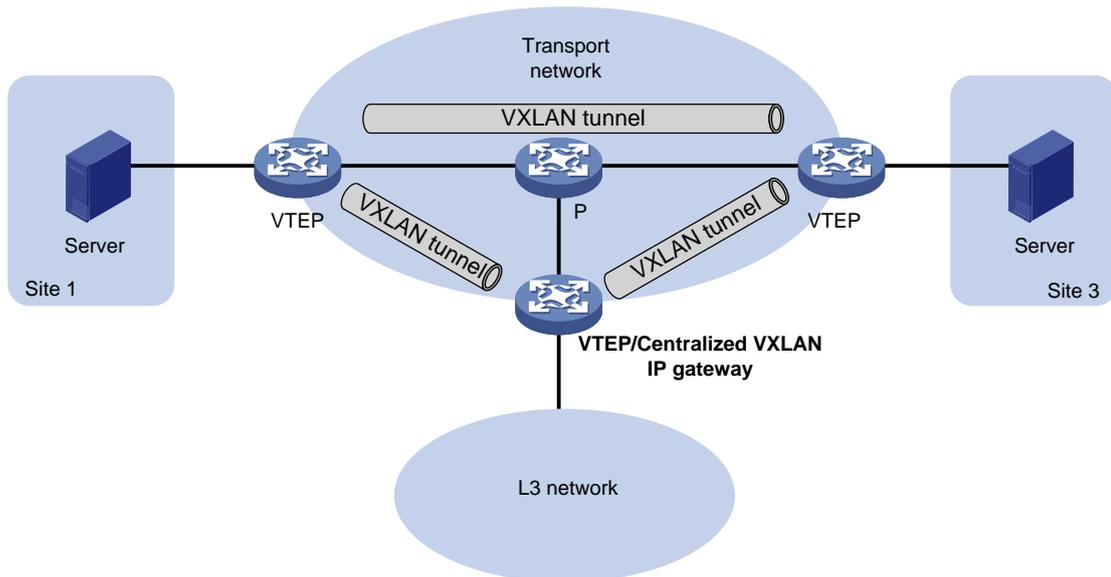
**Figure 9 VXLAN IP gateway separated from VTEPs**



## Centralized VXLAN IP gateway deployment

As shown in [Figure 10](#), a VTEP acts as a gateway for VMs in the VXLANs. The VTEP both terminates the VXLANs and performs Layer 3 forwarding for the VMs.

**Figure 10 Centralized VXLAN IP gateway placement design**



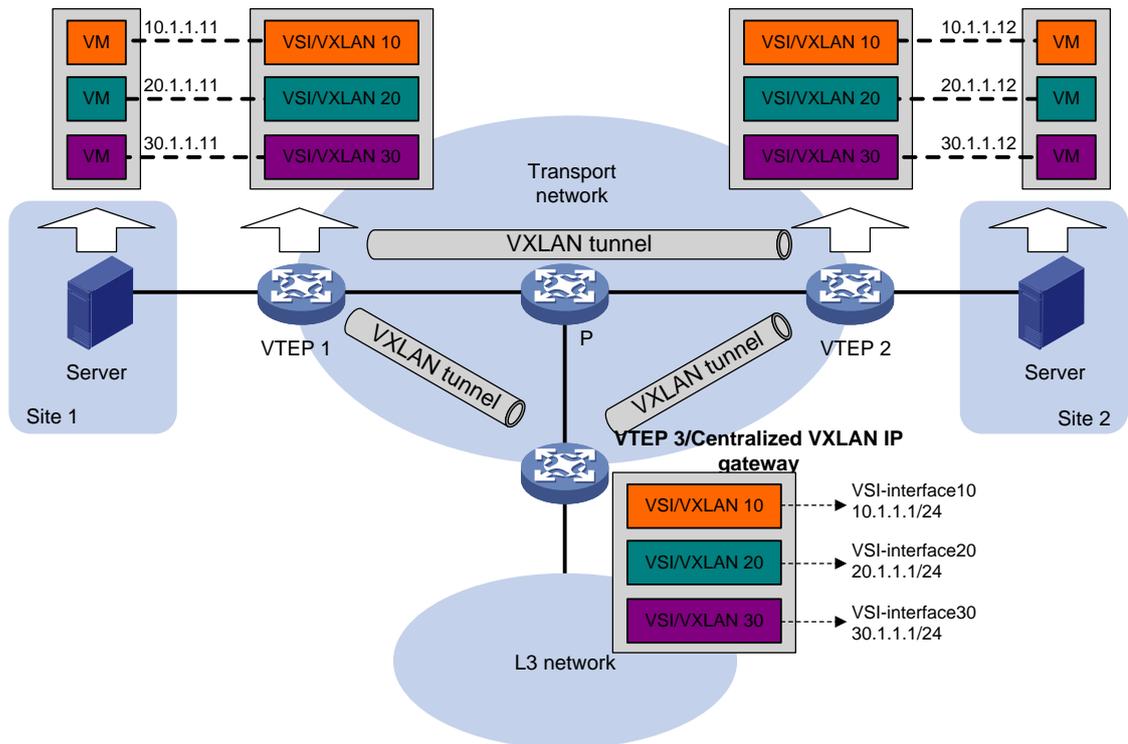
As shown in [Figure 11](#), the network uses the following process to forward Layer 3 traffic from VM 10.1.1.11 to the Layer 3 network:

1. The VM sends an ARP request to obtain the MAC address of the gateway (VTEP 3) at 10.1.1.1.
2. VTEP 1 floods the ARP request to all remote VTEPs.
3. VTEP 3 de-encapsulates the ARP request, creates an ARP entry for the VM, and sends an ARP reply to the VM.

4. VTEP 1 forwards the ARP reply to the VM.
5. The VM learns the MAC address of the gateway, and sends the Layer 3 traffic to the gateway.
6. VTEP 3 removes the VXLAN encapsulation and inner Ethernet header for the traffic, and forwards the traffic to the destination node.

Inter-VXLAN forwarding is the same as this process except for the last step. At the last step of inter-VLAN forwarding, the gateway replaces the source-VXLAN encapsulation with the destination-VXLAN encapsulation, and then forwards the traffic.

**Figure 11 Example of centralized VXLAN IP gateway deployment**



## Distributed VXLAN IP gateway deployment

### About distributed VXLAN IP gateway deployment

As shown in [Figure 12](#), each site's VTEP acts as a gateway to perform Layer 3 forwarding for the VXLANs of the local site. A VTEP acts as a border gateway to the Layer 3 network for the VXLANs.

**Figure 12 Distributed VXLAN IP gateway placement design**

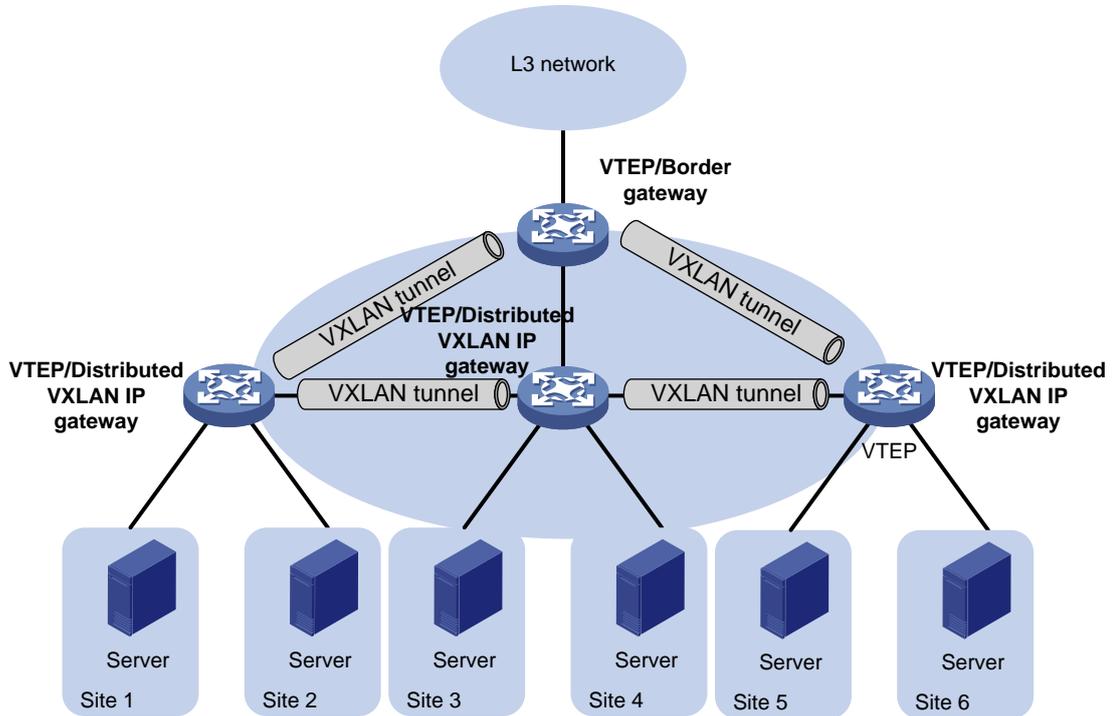
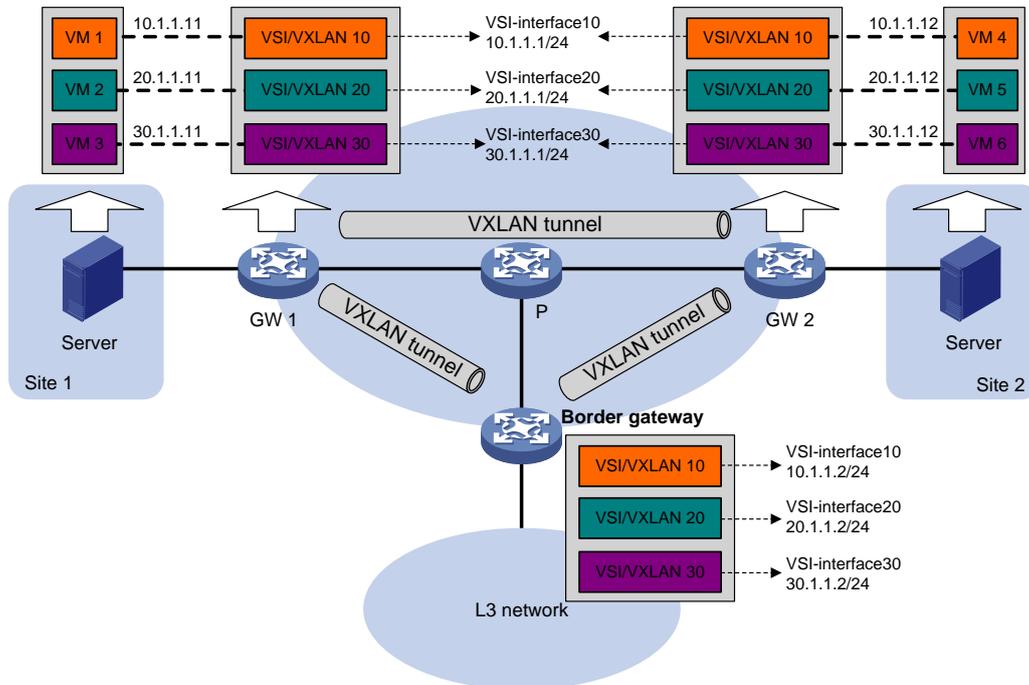


Figure 13 shows an example of distributed VXLAN IP gateway deployment. Create VSI interfaces on each distributed VXLAN IP gateway and the border gateway as gateway interfaces. Assign the same IP address to the same VSI interface on the distributed VXLAN IP gateways. Enable one of the following features on a distributed VXLAN IP gateway:

- ARP flood suppression. The gateway performs Layer 2 forwarding based on MAC address entries and performs Layer 3 forwarding based on ARP entries.
- Local proxy ARP or local ND proxy. The gateway performs Layer 3 forwarding based on ARP or ND entries. The following sections use distributed VXLAN IP gateways enabled with the local proxy ARP or local ND proxy feature to describe the forwarding processes for intra-VXLAN traffic, inter-VXLAN traffic, and traffic from a VXLAN to an external network.

A distributed VXLAN IP gateway can generate ARP or ND entries by a variety of methods. The following sections use dynamically learned ARP or ND entries to describe the forwarding processes.

**Figure 13 Example of distributed VXLAN IP gateway deployment**



### Intra-VXLAN traffic forwarding between sites

As shown in Figure 13, the network uses the following process to forward traffic in a VXLAN between sites (for example, from VM 1 to VM 4 in VXLAN 10):

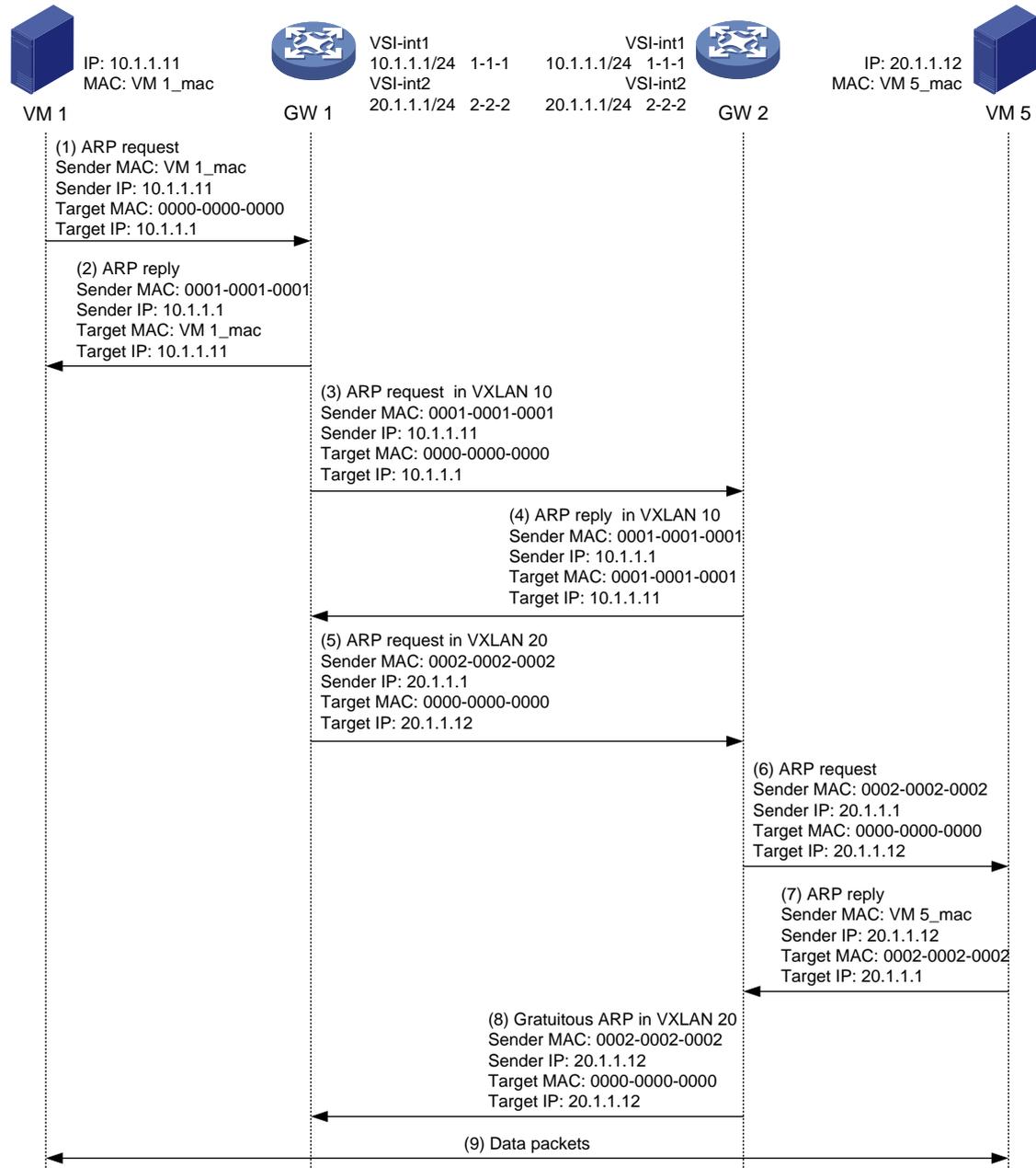
1. VM 1 sends an ARP request to obtain the MAC address of VM 4.
2. GW 1 performs the following operations:
  - a. Creates an ARP entry for VM 1 and replies with the MAC address of VSI-interface 10 (the gateway interface for VXLAN 10).
  - b. Replaces the sender MAC address of the ARP request with the MAC address of VSI-interface 10, and then floods the request to all sites in VXLAN 10.
3. VM 1 creates an ARP entry for VM 4. The MAC address in the entry is the MAC address of VSI-interface 10 on GW 1.
4. GW 2 (the VTEP for VM 4) performs the following operations:
  - a. De-encapsulates the ARP request and creates an ARP entry for VM 1. The entry contains VM 1's IP address (10.1.1.11), the MAC address of VSI-interface 10 on GW 1, and the incoming tunnel interface.
  - b. Replaces the sender MAC address of the request with the MAC address of VSI-interface 10 on GW 2, and then floods the request to the local site in VXLAN 10.
5. VM 4 creates an ARP entry for VM 1, and then sends a reply to GW 2. The MAC address in the ARP entry is the MAC address of VSI-interface 10 on GW 2.
6. GW 2 performs the following operations:
  - a. Creates an ARP entry for VM 4.
  - b. Replaces the sender MAC address of the reply with the MAC address of VSI-interface 10 on GW 2, and sends the reply to GW 1.
7. GW 1 de-encapsulates the ARP reply and creates an ARP entry for VM 4. The entry contains VM 4's IP address (10.1.1.12), the MAC address of VSI-interface 10 on GW 2, and the incoming tunnel interface.
8. For subsequent traffic between VM 1 and VM 4, GW 1 and GW 2 use their respective ARP tables to make the forwarding decision.

## Inter-VXLAN traffic forwarding between sites

As shown in [Figure 14](#), the network uses the following process to forward traffic between VXLANs (for example, from VM 1 in VXLAN 10 to VM 5 in VXLAN 20):

1. VM 1 sends an ARP request to obtain the MAC address of the gateway at 10.1.1.1.
2. GW 1 creates an ARP entry for VM 1 and replies with the MAC address of VSI-interface 10 (the gateway interface for VXLAN 10) so VM 1 will send the packets destined for VM 5 to GW 1.
3. GW 1 sends an ARP request to the local and remote sites in VXLAN 10. In the ARP request, the sender IP address is 10.1.1.11, and the sender MAC address is the MAC address of VSI-interface 10 on GW 1.
4. GW 2 performs the following operations:
  - a. De-encapsulates the ARP request and creates an ARP entry for VM 1. The entry contains IP address 10.1.1.11 and MAC address of VSI-interface 10 on GW 1, and the incoming tunnel interface.
  - b. Replaces the sender MAC address of the request with the MAC address of VSI-interface 10 on GW 2, and then floods the request to the local site in VXLAN 10.
  - c. Sends an ARP reply to GW 1. The reply contains IP address 10.1.1.1 and MAC address of VSI-interface 10 on GW 2).
5. When sending an ARP request in VXLAN 10, GW 1 also sends an ARP request to the local and remote sites in VXLAN 20 to obtain the MAC address of VM 5. In the ARP request, the sender IP address is 20.1.1.1, and the sender MAC address is the MAC address of VSI-interface 20 on GW 1.
6. GW 2 de-encapsulates the ARP request of VXLAN 20, replaces the sender MAC address of the request with the MAC address of VSI-interface 20 on GW 2, and then floods the request to the local site in VXLAN 20.
7. VM 5 creates an ARP entry for GW 2, and then sends a reply to GW 2. The entry contains IP address 20.1.1.1 and MAC address of VSI-interface 20 on GW 2.
8. GW 2 performs the following operations:
  - a. Creates an ARP entry for VM 5.
  - b. Sends a gratuitous ARP packet to the local and remote sites. In the packet, the sender IP address is 20.1.1.12, and the sender MAC address is the MAC address of VSI-interface 20 on GW 2.
9. GW 1 de-encapsulates the gratuitous ARP packet and creates an ARP entry for VM 5. The entry contains VM 5's IP address 20.1.1.12, the MAC address of VSI-interface 20 on GW 2, and the incoming tunnel interface.
10. For subsequent traffic between VM 1 and VM 5, GW 1 and GW 2 use their respective ARP tables to make the forwarding decision.

**Figure 14 Inter-VXLAN traffic forwarding between sites**



### VXLAN-to-external network traffic forwarding

As shown in [Figure 13](#), the network uses the following process to forward traffic from a VXLAN to the Layer 3 network (for example, from VM 1 to the host at 50.1.1.1):

1. VM 1 sends an ARP request to obtain the MAC address of the gateway at 10.1.1.1.
2. GW 1 creates an ARP entry for VM 1 and replies with the MAC address of VSI-interface 10 (the gateway interface for VXLAN 10).
3. VM 1 sends a packet destined for the host to GW 1.
4. GW 1 performs the following operations:
  - a. Searches the IP routing policies or routing table for the next hop. In this example, the next hop for the packet is 10.1.1.2 (the border gateway).

- b. Floods an ARP request to the local and remote sites in VXLAN 10 to obtain the MAC address of 10.1.1.2.
5. The border gateway de-encapsulates the ARP request, creates an ARP entry for GW 1, and tunnels a reply to GW 1.
6. GW 1 de-encapsulates the ARP reply and creates an ARP entry for 10.1.1.2.
7. GW 1 sends the packet destined for the host to the border gateway.
8. The border gateway de-encapsulates the packet and forwards it to the host.

## VXLAN IP gateway tasks at a glance

To configure a VXLAN IP gateway, perform the following tasks:

1. Configure a VXLAN IP gateway  
Choose one of the following tasks:
  - o [Configuring a centralized VXLAN IP gateway](#)
  - o [Configuring a distributed VXLAN IP gateway](#)
2. (Optional.) [Managing ARP entries and ND entries](#)
3. (Optional.) [Configuring a VSI interface](#)

## Prerequisites for VXLAN IP gateway configuration

Before you configure a centralized or distributed VXLAN IP gateway, you must perform the following tasks on VTEPs:

- Enable Layer 3 forwarding for VXLANs.
- Create VSIs and VXLANs.
- Configure VXLAN tunnels and assign them to VXLANs.

## Configuring a centralized VXLAN IP gateway

### Restrictions and guidelines

Do not execute the `local-proxy-arp enable` command on a centralized VXLAN IP gateway.

### Configuring a gateway interface on a centralized VXLAN IP gateway

1. Enter system view.  
**system-view**
2. Create a VSI interface and enter VSI interface view.  
**interface vsi-interface** *vsi-interface-id*
3. Assign an IPv4 address to the VSI interface.  
**ip address** *ip-address* { *mask* | *mask-length* }  
By default, no IPv4 address is assigned to a VSI interface.
4. Return to system view.  
**quit**

5. Enter VSI view.  
`vsi vsi-name`
6. Specify a gateway interface for the VSI.  
`gateway vsi-interface vsi-interface-id`  
By default, no gateway interface is specified for a VSI.

## Assigning a subnet to a VSI

### About subnet assignment to a VSI

Perform this task on VSIs that share a gateway interface. This task enables the VSI interface to identify the VSI of a packet.

You can assign a maximum of eight IPv4 and IPv6 subnets to a VSI. Make sure these subnets are on the same network as one of the IP addresses on the gateway interface.

For VSIs that share a gateway interface, the subnets must be unique.

If you remove the gateway interface from the VSI, the VSI's subnet settings are automatically deleted.

### Procedure

1. Enter system view.  
`system-view`
2. Enter VSI view.  
`vsi vsi-name`
3. Assign a subnet to the VSI.  
`gateway subnet ipv4-address wildcard-mask`  
By default, no subnet exists on a VSI.

## Configuring a distributed VXLAN IP gateway

### Restrictions and guidelines for distributed VXLAN IP gateway configuration

For a VXLAN that requires access to the external network, specify the VXLAN's VSI interface on the border gateway as the next hop by using one of the following methods:

- Configure a static route.
- Configure a routing policy, and apply the policy by using the `apply default-next-hop` command. For more information about configuring routing policies, see routing policy configuration in *Layer 3—IP Routing Configuration Guide*.

If both ARP flood suppression and local proxy ARP are enabled on a distributed VXLAN IP gateway, only local proxy ARP takes effect. As a best practice, do not use these features together on distributed VXLAN IP gateways. For more information about ARP flood suppression, see "[Enabling ARP flood suppression](#)."

Make sure a VSI interface uses the same MAC address to provide service on distributed VXLAN IP gateways connected to IPv4 sites. Make sure a VSI interface uses different link-local addresses to provide service on distributed VXLAN IP gateways connected to both IPv4 and IPv6 sites.

# Configuring a gateway interface on a distributed VXLAN IP gateway

1. Enter system view.  
**system-view**
2. Create a VSI interface and enter VSI interface view.  
**interface vsi-interface vsi-interface-id**
3. Assign an IP address to the VSI interface.  
IPv4:  
**ip address ip-address { mask | mask-length } [ sub ]**  
IPv6:  
See IPv6 basics in *Layer 3—IP Services Configuration Guide*.  
By default, no IP address is assigned to a VSI interface.
4. Specify the VSI interface as a distributed gateway.  
**distributed-gateway local**  
By default, a VSI interface is not a distributed gateway.
5. Enable local proxy ARP or local ND proxy.  
IPv4:  
**local-proxy-arp enable [ ip-range startIP to endIP ]**  
By default, local proxy ARP is disabled.  
For more information about this command, see proxy ARP commands in *Layer 3—IP Services Command Reference*.  
IPv6:  
**local-proxy-nd enable**  
By default, local ND proxy is disabled.  
For more information about this command, see IPv6 basics commands in *Layer 3—IP Services Command Reference*.
6. Bring up the VSI interface.  
**undo shutdown**  
By default, a VSI interface is up.
7. Return to system view.  
**quit**
8. Enter VSI view.  
**vsi vsi-name**
9. Specify the VSI interface as the gateway interface for the VSI.  
**gateway vsi-interface vsi-interface-id**  
By default, no gateway interface is specified for a VSI.

## Enabling dynamic ARP or ND entry synchronization for distributed VXLAN IP gateways

### About dynamic ARP or ND entry synchronization for distributed VXLAN IP gateways

When local proxy ARP or local ND proxy is enabled on distributed VXLAN IP gateways, enable this feature for all gateways to have the same ARP or ND entries.

A controller or the EVPN feature can also synchronize ARP or ND entries among distributed VXLAN IP gateways. When you use a controller or the EVPN feature, do not enable dynamic ARP or ND entry synchronization.

### Enabling dynamic ARP entry synchronization

1. Enter system view.

```
system-view
```

2. Enable dynamic ARP entry synchronization for distributed VXLAN IP gateways.

```
arp distributed-gateway dynamic-entry synchronize
```

By default, dynamic ARP entry synchronization is disabled for distributed VXLAN IP gateways.

### Enabling dynamic ND entry synchronization

1. Enter system view.

```
system-view
```

2. Enable dynamic ND entry synchronization for distributed VXLAN IP gateways.

```
ipv6 nd distributed-gateway dynamic-entry synchronize
```

By default, dynamic ND entry synchronization is disabled for distributed VXLAN IP gateways.

## Assigning a subnet to a VSI

### About subnet assignment to a VSI

Perform this task on VSIs that share a gateway interface. This task enables the VSI interface to identify the VSI of a packet.

You can assign a maximum of eight IPv4 and IPv6 subnets to a VSI. Make sure these subnets are on the same network as one of the IP addresses on the gateway interface.

For VSIs that share a gateway interface, the subnets must be unique.

If you remove the gateway interface from the VSI, the VSI's subnet settings are automatically deleted.

### Procedure

1. Enter system view.

```
system-view
```

2. Enter VSI view.

```
vsi vsi-name
```

3. Assign a subnet to the VSI.

```
gateway subnet { ipv4-address wildcard-mask | ipv6-address prefix-length }
```

By default, no subnet exists on a VSI.

## Managing ARP entries and ND entries

### Adding a static ARP entry

#### About static ARP entries

A VXLAN IP gateway can dynamically learn ARP entries and use manually configured static ARP entries.

## Procedure

1. Enter system view.

**system-view**

2. Add a static local-ARP entry.

```
arp static ip-address mac-address vsi-interface vsi-interface-id  
interface-type interface-number service-instance instance-id vsi  
vsi-name [ vpn-instance vpn-instance-name ]
```

For more information about this command, see ARP commands in *Layer 3—IP Services Command Reference*.

3. Add a static remote-ARP entry.

```
arp static ip-address mac-address vsi-interface vsi-interface-id  
tunnel number vsi vsi-name [ vpn-instance vpn-instance-name ]
```

For more information about this command, see ARP commands in *Layer 3—IP Services Command Reference*.

# Disabling remote ARP or ND learning for VXLANs

## About remote ARP and ND learning

By default, the device learns ARP or ND information of remote user terminals from packets received on VXLAN tunnel interfaces. To save resources on VTEPs in an SDN transport network, you can temporarily disable remote ARP or ND learning when the controller and VTEPs are synchronizing entries. After the entry synchronization is completed, enable remote ARP or ND learning.

## Restrictions and guidelines

As a best practice, disable remote ARP or ND learning for VXLANs only when the controller and VTEPs are synchronizing entries.

## Procedure

1. Enter system view.

**system-view**

2. Disable remote ARP learning.

```
vxlan tunnel arp-learning disable
```

By default, remote ARP learning is enabled for VXLANs.

3. Disable remote ND learning.

```
vxlan tunnel nd-learning disable
```

By default, remote ND learning is enabled for VXLANs.

# Configuring a VSI interface

## Configuring optional parameters for a VSI interface

1. Enter system view.

**system-view**

2. Enter VSI interface view.

```
interface vsi-interface vsi-interface-id
```

3. Assign a MAC address to the VSI interface.

```
mac-address mac-address
```

By default, the MAC address of VSI interfaces is the bridge MAC address + 26.

4. Configure the description of the VSI interface.

**description** *text*

The default description of a VSI interface is *interface-name* plus **Interface** (for example, **Vsi-interface100 Interface**).

5. Set the MTU for the VSI interface.

**mtu** *size*

The default MTU is 1444 bytes.

6. Set the expected bandwidth for the VSI interface.

**bandwidth** *bandwidth-value*

The default expected bandwidth (in kbps) equals the interface baudrate divided by 1000.

The expected bandwidth is an informational parameter used only by higher-layer protocols for calculation. You cannot adjust the actual bandwidth of an interface by using this command.

7. Set an ARP packet sending rate limit for the VSI interface.

**arp send-rate** *pps*

By default, the ARP packet sending rate is not limited for a VSI interface.

## Restoring the default settings of the VSI interface

### Restrictions and guidelines

#### CAUTION:

This operation might interrupt ongoing network services. Make sure you are fully aware of the impact of this operation when you perform it on a live network.

This operation might fail to restore the default settings for some commands for reasons such as command dependencies or system restrictions. Use the **display this** command in interface view to identify these commands. Use their **undo** forms or follow the command reference to restore their default settings. If your restoration attempt still fails, follow the error message instructions to resolve the problem.

### Procedure

1. Enter system view.  
**system-view**
2. Enter VSI interface view.  
**interface vsi-interface** *vsi-interface-id*
3. Restore the default settings of the VSI interface.  
**default**

## Display and maintenance commands for VXLAN IP gateways

Execute **display** commands in any view.

| Task                                      | Command   |
|---|---|
| Display information about VSI interfaces. | <b>display interface</b> [ <b>vsi-interface</b> |

| Task | Command   |
|------|---|
|      | [ vsi-interface-id ] [ brief [ description   down ] ] |

# VXLAN IP gateway configuration examples

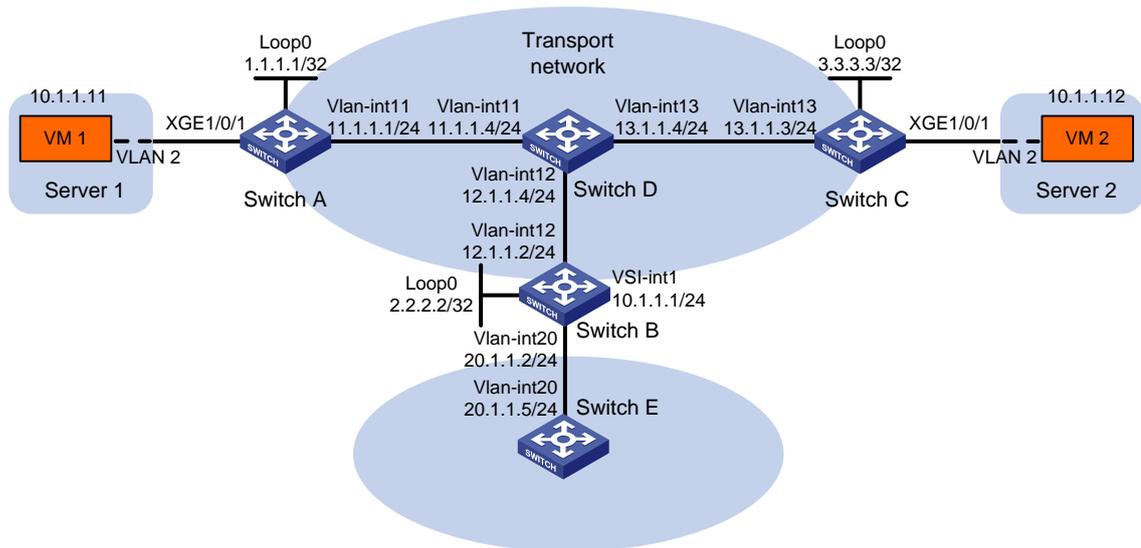
## Example: Configuring a centralized VXLAN IP gateway

### Network configuration

As shown in [Figure 15](#):

- Configure VXLAN 10 as a unicast-mode VXLAN on Switch A, Switch B, and Switch C to provide connectivity for the VMs across the network sites.
- Configure a centralized VXLAN IP gateway on Switch B to provide gateway services for VXLAN 10.
- Manually establish VXLAN tunnels and assign the tunnels to VXLAN 10.
- Enable remote-MAC address learning.

**Figure 15 Network diagram**



### Procedure

1. Set the system operating mode to VXLAN on Switch A, Switch B, and Switch C. This step uses Switch A as an example.

```

<SwitchA> system-view
[SwitchA] switch-mode 1
Reboot device to make the configuration take effect.
[SwitchA] quit
<SwitchA> reboot
Start to check configuration with next startup configuration file, please wait..
.....DONE!
Current configuration may be lost after the reboot, save current configuration?
[Y/N]:y

```

This command will reboot the device. Continue? [Y/N]:y

2. On VM 1 and VM 2, specify 10.1.1.1 as the gateway address. (Details not shown.)
3. Configure IP addresses and unicast routing settings:
  - # Assign IP addresses to interfaces, as shown in [Figure 15](#). (Details not shown.)
  - # Configure OSPF on all transport network switches (Switches A through D). (Details not shown.)
  - # Configure OSPF to advertise routes to networks 10.1.1.0/24 and 20.1.1.0/24 on Switch B and Switch E. (Details not shown.)
4. Configure Switch A:
  - # Enable L2VPN.

```
<SwitchA> system-view
[SwitchA] l2vpn enable
```
  - # Enable Layer 2 forwarding for VXLANs.

```
[SwitchA] undo vxlan ip-forwarding
```
  - # Create VSI **vpna** and VXLAN 10.

```
[SwitchA] vsi vpna
[SwitchA-vsi-vpna] vxlan 10
[SwitchA-vsi-vpna-vxlan-10] quit
[SwitchA-vsi-vpna] quit
```
  - # Assign an IP address to Loopback 0. The IP address will be used as the source IP address of the VXLAN tunnels to Switch B and Switch C.

```
[SwitchA] interface loopback 0
[SwitchA-Loopback0] ip address 1.1.1.1 255.255.255.255
[SwitchA-Loopback0] quit
```
  - # Create a VXLAN tunnel to Switch B. The tunnel interface name is **Tunnel 1**.

```
[SwitchA] interface tunnel 1 mode vxlan
[SwitchA-Tunnel1] source 1.1.1.1
[SwitchA-Tunnel1] destination 2.2.2.2
[SwitchA-Tunnel1] quit
```
  - # Create a VXLAN tunnel to Switch C. The tunnel interface name is **Tunnel 2**.

```
[SwitchA] interface tunnel 2 mode vxlan
[SwitchA-Tunnel2] source 1.1.1.1
[SwitchA-Tunnel2] destination 3.3.3.3
[SwitchA-Tunnel2] quit
```
  - # Assign Tunnel 1 and Tunnel 2 to VXLAN 10.

```
[SwitchA] vsi vpna
[SwitchA-vsi-vpna] vxlan 10
[SwitchA-vsi-vpna-vxlan-10] tunnel 1
[SwitchA-vsi-vpna-vxlan-10] tunnel 2
[SwitchA-vsi-vpna-vxlan-10] quit
[SwitchA-vsi-vpna] quit
```
  - # On Ten-GigabitEthernet 1/0/1, create Ethernet service instance 1000 to match VLAN 2.

```
[SwitchA] interface ten-gigabitethernet 1/0/1
[SwitchA-Ten-GigabitEthernet1/0/1] port link-type trunk
[SwitchA-Ten-GigabitEthernet1/0/1] port trunk permit vlan 2
[SwitchA-Ten-GigabitEthernet1/0/1] service-instance 1000
[SwitchA-Ten-GigabitEthernet1/0/1-srv1000] encapsulation s-vid 2
```

**# Map Ethernet service instance 1000 to VSI `vpna`.**

```
[SwitchA-Ten-GigabitEthernet1/0/1-srv1000] xconnect vsi vpna
[SwitchA-Ten-GigabitEthernet1/0/1-srv1000] quit
[SwitchA-Ten-GigabitEthernet1/0/1] quit
```

**5. Configure Switch B:**

**# Enable L2VPN.**

```
<SwitchB> system-view
[SwitchB] l2vpn enable
```

**# Create VSI `vpna` and VXLAN 10.**

```
[SwitchB] vsi vpna
[SwitchB-vsi-vpna] vxlan 10
[SwitchB-vsi-vpna-vxlan-10] quit
[SwitchB-vsi-vpna] quit
```

**# Assign an IP address to Loopback 0. The IP address will be used as the source IP address of the VXLAN tunnels to Switch A and Switch C.**

```
[SwitchB] interface loopback 0
[SwitchB-Loopback0] ip address 2.2.2.2 255.255.255.255
[SwitchB-Loopback0] quit
```

**# Create a VXLAN tunnel to Switch A. The tunnel interface name is **Tunnel 2**.**

```
[SwitchB] interface tunnel 2 mode vxlan
[SwitchB-Tunnel2] source 2.2.2.2
[SwitchB-Tunnel2] destination 1.1.1.1
[SwitchB-Tunnel2] quit
```

**# Create a VXLAN tunnel to Switch C. The tunnel interface name is **Tunnel 3**.**

```
[SwitchB] interface tunnel 3 mode vxlan
[SwitchB-Tunnel3] source 2.2.2.2
[SwitchB-Tunnel3] destination 3.3.3.3
[SwitchB-Tunnel3] quit
```

**# Assign Tunnel 2 and Tunnel 3 to VXLAN 10.**

```
[SwitchB] vsi vpna
[SwitchB-vsi-vpna] vxlan 10
[SwitchB-vsi-vpna-vxlan-10] tunnel 2
[SwitchB-vsi-vpna-vxlan-10] tunnel 3
[SwitchB-vsi-vpna-vxlan-10] quit
[SwitchB-vsi-vpna] quit
```

**# Create VSI-interface 1 and assign the interface an IP address. The IP address will be used as the gateway address for VXLAN 10.**

```
[SwitchB] interface vsi-interface 1
[SwitchB-Vsi-interfacel] ip address 10.1.1.1 255.255.255.0
[SwitchB-Vsi-interfacel] quit
```

**# Specify VSI-interface 1 as the gateway interface for VSI `vpna`.**

```
[SwitchB] vsi vpna
[SwitchB-vsi-vpna] gateway vsi-interface 1
[SwitchB-vsi-vpna] quit
```

**6. Configure Switch C:**

**# Enable L2VPN.**

```
<SwitchC> system-view
[SwitchC] l2vpn enable
```

```

# Enable Layer 2 forwarding for VXLANs.
[SwitchC] undo vxlan ip-forwarding
# Create VSI vpna and VXLAN 10.
[SwitchC] vsi vpna
[SwitchC-vsi-vpna] vxlan 10
[SwitchC-vsi-vpna-vxlan-10] quit
[SwitchC-vsi-vpna] quit
# Assign an IP address to Loopback 0. The IP address will be used as the source IP address of
the VXLAN tunnels to Switch A and Switch B.
[SwitchC] interface loopback 0
[SwitchC-Loopback0] ip address 3.3.3.3 255.255.255.255
[SwitchC-Loopback0] quit
# Create a VXLAN tunnel to Switch A. The tunnel interface name is Tunnel 1.
[SwitchC] interface tunnel 1 mode vxlan
[SwitchC-Tunnel1] source 3.3.3.3
[SwitchC-Tunnel1] destination 1.1.1.1
[SwitchC-Tunnel1] quit
# Create a VXLAN tunnel to Switch B. The tunnel interface name is Tunnel 3.
[SwitchC] interface tunnel 3 mode vxlan
[SwitchC-Tunnel3] source 3.3.3.3
[SwitchC-Tunnel3] destination 2.2.2.2
[SwitchC-Tunnel3] quit
# Assign Tunnel 1 and Tunnel 3 to VXLAN 10.
[SwitchC] vsi vpna
[SwitchC-vsi-vpna] vxlan 10
[SwitchC-vsi-vpna-vxlan-10] tunnel 1
[SwitchC-vsi-vpna-vxlan-10] tunnel 3
[SwitchC-vsi-vpna-vxlan-10] quit
[SwitchC-vsi-vpna] quit
# On Ten-GigabitEthernet 1/0/1, create Ethernet service instance 1000 to match VLAN 2.
[SwitchC] interface ten-gigabitethernet 1/0/1
[SwitchC-Ten-GigabitEthernet1/0/1] port link-type trunk
[SwitchC-Ten-GigabitEthernet1/0/1] port trunk permit vlan 2
[SwitchC-Ten-GigabitEthernet1/0/1] service-instance 1000
[SwitchC-Ten-GigabitEthernet1/0/1-srv1000] encapsulation s-vid 2
# Map Ethernet service instance 1000 to VSI vpna.
[SwitchC-Ten-GigabitEthernet1/0/1-srv1000] xconnect vsi vpna
[SwitchC-Ten-GigabitEthernet1/0/1-srv1000] quit
[SwitchC-Ten-GigabitEthernet1/0/1] quit

```

## Verifying the configuration

1. Verify the VXLAN IP gateway settings on Switch B:
 

```

# Verify that the VXLAN tunnel interfaces are up on Switch B.
[SwitchB] display interface tunnel 2
Tunnel2
Current state: UP
Line protocol state: UP
Description: Tunnel2 Interface

```

```

Bandwidth: 64 kbps
Maximum transmission unit: 1464
Internet protocol processing: Disabled
Last clearing of counters: Never
Tunnel source 2.2.2.2, destination 1.1.1.1
Tunnel protocol/transport UDP_VXLAN/IP
Last 300 seconds input rate: 0 bytes/sec, 0 bits/sec, 0 packets/sec
Last 300 seconds output rate: 0 bytes/sec, 0 bits/sec, 0 packets/sec
Input: 0 packets, 0 bytes, 0 drops
Output: 0 packets, 0 bytes, 0 drops

```

**# Verify that VSI-interface 1 is up.**

```

[SwitchB] display interface vsi-interface 1
Vsi-interfacel
Current state: UP
Line protocol state: UP
Description: Vsi-interfacel Interface
Bandwidth: 1000000 kbps
Maximum transmission unit: 1444
Internet address: 10.1.1.1/24 (primary)
IP packet frame type: Ethernet II, hardware address: 0011-2200-0102
IPv6 packet frame type: Ethernet II, hardware address: 0011-2200-0102
Physical: Unknown, baudrate: 1000000 kbps
Last clearing of counters: Never
Last 300 seconds input rate: 0 bytes/sec, 0 bits/sec, 0 packets/sec
Last 300 seconds output rate: 0 bytes/sec, 0 bits/sec, 0 packets/sec
Input: 0 packets, 0 bytes, 0 drops
Output: 0 packets, 0 bytes, 0 drops

```

**# Verify that the VXLAN tunnels have been assigned to the VXLAN, and VSI-interface 1 is the gateway interface of VSI vpna.**

```

[SwitchB] display l2vpn vsi verbose
VSI Name: vpna
VSI Index          : 0
VSI State          : Up
MTU                : 1500
Bandwidth          : -
Broadcast Restrain : -
Multicast Restrain : -
Unknown Unicast Restrain: -
MAC Learning       : Enabled
MAC Table Limit    : -
MAC Learning rate  : -
Drop Unknown       : -
Flooding           : Enabled
Statistics         : Disabled
Gateway interface  : VSI-interface 1
VXLAN ID           : 10
Tunnels:
  Tunnel Name      Link ID   State   Type      Flood proxy

```

|         |           |    |        |          |
|---------|-----------|----|--------|----------|
| Tunnel2 | 0x5000002 | Up | Manual | Disabled |
| Tunnel3 | 0x5000003 | Up | Manual | Disabled |

# Verify that Switch B has created ARP entries for the VMs.

```
[SwitchB] display arp
```

| Type: S-Static | D-Dynamic      | O-Openflow | R-Rule            | M-Multiport | I-Invalid |
|----------------|----------------|------------|-------------------|-------------|-----------|
| IP address     | MAC address    | VLAN/VSI   | Interface/Link ID | Aging       | Type      |
| 20.1.1.5       | 000c-29c1-5e46 | 20         | Vlan20            | 19          | D         |
| 10.1.1.11      | 0000-1234-0001 | 0          | Tunnel2           | 20          | D         |
| 10.1.1.12      | 0000-1234-0002 | 0          | Tunnel3           | 19          | D         |

# Verify that Switch B has created FIB entries for the VMs.

```
[SwitchB] display fib 10.1.1.11
```

```
Destination count: 1 FIB entry count: 1
```

```
Flag:
```

| U:Usable         | G:Gateway | H:Host | B:Blackhole        | D:Dynamic | S:Static |
|------------------|-----------|--------|--------------------|-----------|----------|
| R:Relay          | F:FRR     |        |                    |           |          |
| Destination/Mask | Nexthop   | Flag   | OutInterface/Token | Label     |          |
| 10.1.1.11/32     | 10.1.1.11 | UH     | Vs11               |           | Null     |

## 2. Verify that the VMs can access the WAN:

# Verify that VM 1 and VM 2 can ping each other. (Details not shown.)

# Verify that VM 1, VM 2, and VLAN-interface 20 (20.1.1.5) on Switch E can ping each other. (Details not shown.)

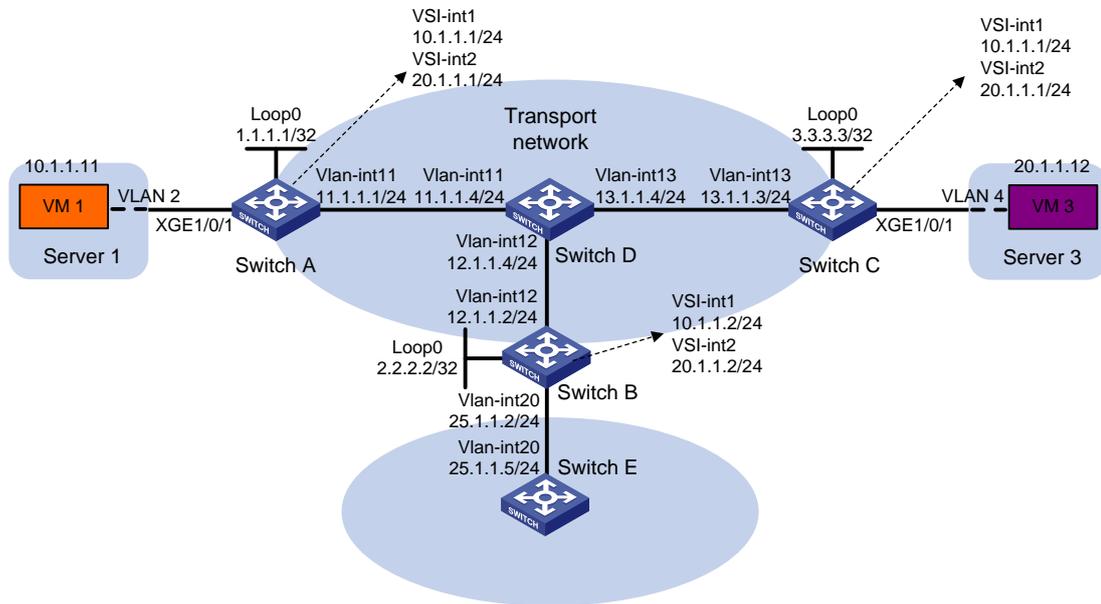
# Example: Configuring distributed VXLAN IPv4 gateways

## Network configuration

As shown in [Figure 16](#):

- Configure VXLAN 10 and VXLAN 30 as unicast-mode VXLANs on Switch A, Switch B, and Switch C to provide connectivity for the VMs across the network sites.
- Manually establish VXLAN tunnels and assign the tunnels to the VXLANs.
- Configure distributed VXLAN IP gateways on Switch A and Switch C to forward traffic between the VXLANs.
- Configure Switch B as a border gateway to forward traffic between the VXLANs and the WAN connected to Switch E.

Figure 16 Network diagram



## Procedure

1. Set the system operation mode to VXLAN on Switch A, Switch B, and Switch C. This step uses Switch A as an example.

```
<SwitchA> system-view
[SwitchA] switch-mode 1
Reboot device to make the configuration take effect.
[SwitchA] quit
<SwitchA> reboot
Start to check configuration with next startup configuration file, please wait..
.....DONE!
Current configuration may be lost after the reboot, save current configuration?
[Y/N]:y
This command will reboot the device. Continue? [Y/N]:y
```

2. On VM 1 and VM 3, specify 10.1.1.1 and 20.1.1.1 as the gateway address, respectively. (Details not shown.)
3. Configure IP addresses and unicast routing settings:
  - # Assign IP addresses to interfaces, as shown in Figure 16. (Details not shown.)
  - # Configure OSPF on all transport network switches (Switches A through D). (Details not shown.)
  - # Configure OSPF to advertise routes to networks 10.1.1.0/24, 20.1.1.0/24, and 25.1.1.0/24 on Switch B and Switch E. (Details not shown.)

4. Configure Switch A:

```
# Enable L2VPN.
<SwitchA> system-view
[SwitchA] l2vpn enable
# Create VSI vpna and VXLAN 10.
[SwitchA] vsi vpna
[SwitchA-vsi-vpna] vxlan 10
[SwitchA-vsi-vpna-vxlan-10] quit
```

```

[SwitchA-vsi-vpna] quit
# Create VSI vpnc and VXLAN 30.
[SwitchA] vsi vpnc
[SwitchA-vsi-vpnc] vxlan 30
[SwitchA-vsi-vpnc-vxlan-30] quit
[SwitchA-vsi-vpnc] quit
# Assign an IP address to Loopback 0. The IP address will be used as the source IP address of the VXLAN tunnels to Switch B and Switch C.
[SwitchA] interface loopback 0
[SwitchA-Loopback0] ip address 1.1.1.1 255.255.255.255
[SwitchA-Loopback0] quit
# Create a VXLAN tunnel to Switch B. The tunnel interface name is Tunnel 1.
[SwitchA] interface tunnel 1 mode vxlan
[SwitchA-Tunnel1] source 1.1.1.1
[SwitchA-Tunnel1] destination 2.2.2.2
[SwitchA-Tunnel1] quit
# Create a VXLAN tunnel to Switch C. The tunnel interface name is Tunnel 2.
[SwitchA] interface tunnel 2 mode vxlan
[SwitchA-Tunnel2] source 1.1.1.1
[SwitchA-Tunnel2] destination 3.3.3.3
[SwitchA-Tunnel2] quit
# Assign Tunnel 1 and Tunnel 2 to VXLAN 10.
[SwitchA] vsi vpna
[SwitchA-vsi-vpna] vxlan 10
[SwitchA-vsi-vpna-vxlan-10] tunnel 1
[SwitchA-vsi-vpna-vxlan-10] tunnel 2
[SwitchA-vsi-vpna-vxlan-10] quit
[SwitchA-vsi-vpna] quit
# Assign Tunnel 1 and Tunnel 2 to VXLAN 30.
[SwitchA] vsi vpnc
[SwitchA-vsi-vpnc] vxlan 30
[SwitchA-vsi-vpnc-vxlan-30] tunnel 1
[SwitchA-vsi-vpnc-vxlan-30] tunnel 2
[SwitchA-vsi-vpnc-vxlan-30] quit
[SwitchA-vsi-vpnc] quit
# On Ten-GigabitEthernet 1/0/1, create Ethernet service instance 1000 to match VLAN 2.
[SwitchA] interface ten-gigabitethernet 1/0/1
[SwitchA-Ten-GigabitEthernet1/0/1] port link-type trunk
[SwitchA-Ten-GigabitEthernet1/0/1] port trunk permit vlan 2
[SwitchA-Ten-GigabitEthernet1/0/1] service-instance 1000
[SwitchA-Ten-GigabitEthernet1/0/1-srv1000] encapsulation s-vid 2
# Map Ethernet service instance 1000 to VSI vpna.
[SwitchA-Ten-GigabitEthernet1/0/1-srv1000] xconnect vsi vpna
[SwitchA-Ten-GigabitEthernet1/0/1-srv1000] quit
[SwitchA-Ten-GigabitEthernet1/0/1] quit
# Create VSI-interface 1 and assign the interface an IP address and a MAC address. The IP address will be used as the gateway address for VXLAN 10.
[SwitchA] interface vsi-interface 1

```

```

[SwitchA-Vsi-interface1] ip address 10.1.1.1 255.255.255.0
[SwitchA-Vsi-interface1] mac-address 1-1-1
# Specify VSI-interface 1 as a distributed gateway and enable local proxy ARP on the interface.
[SwitchA-Vsi-interface1] distributed-gateway local
[SwitchA-Vsi-interface1] local-proxy-arp enable
[SwitchA-Vsi-interface1] quit
# Create VSI-interface 2 and assign the interface an IP address and a MAC address. The IP
address will be used as the gateway address for VXLAN 30.
[SwitchA] interface vsi-interface 2
[SwitchA-Vsi-interface2] ip address 20.1.1.1 255.255.255.0
[SwitchA-Vsi-interface2] mac-address 2-2-2
# Specify VSI-interface 2 as a distributed gateway and enable local proxy ARP on the interface.
[SwitchA-Vsi-interface2] distributed-gateway local
[SwitchA-Vsi-interface2] local-proxy-arp enable
[SwitchA-Vsi-interface2] quit
# Enable dynamic ARP entry synchronization for distributed VXLAN IP gateways.
[SwitchA] arp distributed-gateway dynamic-entry synchronize
# Specify VSI-interface 1 as the gateway interface for VSI vpna.
[SwitchA] vsi vpna
[SwitchA-vsi-vpna] gateway vsi-interface 1
[SwitchA-vsi-vpna] quit
# Specify VSI-interface 2 as the gateway interface for VSI vpnc.
[SwitchA] vsi vpnc
[SwitchA-vsi-vpnc] gateway vsi-interface 2
[SwitchA-vsi-vpnc] quit
# Configure a PBR policy for VXLAN 10. Set the policy name to vxlan10, and set the next hop
to 10.1.1.2 (VSI-interface 1 on Switch B).
[SwitchA] acl advanced 3000
[SwitchA-acl-ipv4-adv-3000] rule 0 permit ip
[SwitchA-acl-ipv4-adv-3000] quit
[SwitchA] policy-based-route vxlan10 permit node 5
[SwitchA-pbr-vxlan10-5] if-match acl 3000
[SwitchA-pbr-vxlan10-5] apply next-hop 10.1.1.2
[SwitchA-pbr-vxlan10-5] quit
# Configure a PBR policy for VXLAN 30. Set the policy name to vxlan30, and set the next hop
to 20.1.1.2 (VSI-interface 2 on Switch B).
[SwitchA] policy-based-route vxlan30 permit node 5
[SwitchA-pbr-vxlan30-5] if-match acl 3000
[SwitchA-pbr-vxlan30-5] apply next-hop 20.1.1.2
[SwitchA-pbr-vxlan30-5] quit
# Apply policies vxlan10 and vxlan30 to VSI-interface 1 and VSI-interface 2, respectively.
[SwitchA] interface vsi-interface 1
[SwitchA-Vsi-interface1] ip policy-based-route vxlan10
[SwitchA-Vsi-interface1] quit
[SwitchA] interface vsi-interface 2
[SwitchA-Vsi-interface2] ip policy-based-route vxlan30
[SwitchA-Vsi-interface2] quit
# Configure a default route. Set the next hop to 10.1.1.2 (VSI-interface 1 on Switch B).

```

```
[SwitchA] ip route-static 0.0.0.0 0 10.1.1.2
```

## 5. Configure Switch B:

### # Enable L2VPN.

```
<SwitchB> system-view
```

```
[SwitchB] l2vpn enable
```

### # Create VSI **vpna** and VXLAN 10.

```
[SwitchB] vsi vpna
```

```
[SwitchB-vsi-vpna] vxlan 10
```

```
[SwitchB-vsi-vpna-vxlan-10] quit
```

```
[SwitchB-vsi-vpna] quit
```

### # Create VSI **vpnc** and VXLAN 30.

```
[SwitchB] vsi vpnc
```

```
[SwitchB-vsi-vpnc] vxlan 30
```

```
[SwitchB-vsi-vpnc-vxlan-30] quit
```

```
[SwitchB-vsi-vpnc] quit
```

### # Assign an IP address to Loopback 0. The IP address will be used as the source IP address of the VXLAN tunnels to Switch A and Switch C.

```
[SwitchB] interface loopback 0
```

```
[SwitchB-Loopback0] ip address 2.2.2.2 255.255.255.255
```

```
[SwitchB-Loopback0] quit
```

### # Create a VXLAN tunnel to Switch A. The tunnel interface name is **Tunnel 2**.

```
[SwitchB] interface tunnel 2 mode vxlan
```

```
[SwitchB-Tunnel2] source 2.2.2.2
```

```
[SwitchB-Tunnel2] destination 1.1.1.1
```

```
[SwitchB-Tunnel2] quit
```

### # Create a VXLAN tunnel to Switch C. The tunnel interface name is **Tunnel 3**.

```
[SwitchB] interface tunnel 3 mode vxlan
```

```
[SwitchB-Tunnel3] source 2.2.2.2
```

```
[SwitchB-Tunnel3] destination 3.3.3.3
```

```
[SwitchB-Tunnel3] quit
```

### # Assign Tunnel 2 to VXLAN 10.

```
[SwitchB] vsi vpna
```

```
[SwitchB-vsi-vpna] vxlan 10
```

```
[SwitchB-vsi-vpna-vxlan-10] tunnel 2
```

```
[SwitchB-vsi-vpna-vxlan-10] quit
```

```
[SwitchB-vsi-vpna] quit
```

### # Assign Tunnel 3 to VXLAN 30.

```
[SwitchB] vsi vpnc
```

```
[SwitchB-vsi-vpnc] vxlan 30
```

```
[SwitchB-vsi-vpnc-vxlan-30] tunnel 3
```

```
[SwitchB-vsi-vpnc-vxlan-30] quit
```

```
[SwitchB-vsi-vpnc] quit
```

### # Create VSI-interface 1 and assign the interface an IP address.

```
[SwitchB] interface vsi-interface 1
```

```
[SwitchB-Vsi-interfacel] ip address 10.1.1.2 255.255.255.0
```

```
[SwitchB-Vsi-interfacel] quit
```

### # Create VSI-interface 2 and assign the interface an IP address.

```
[SwitchB] interface vsi-interface 2
[SwitchB-Vsi-interface2] ip address 20.1.1.2 255.255.255.0
[SwitchB-Vsi-interface2] quit
```

**# Specify VSI-interface 1 as the gateway interface for VSI `vpna`.**

```
[SwitchB] vsi vpna
[SwitchB-vsi-vpna] gateway vsi-interface 1
[SwitchB-vsi-vpna] quit
```

**# Specify VSI-interface 2 as the gateway interface for VSI `vpnc`.**

```
[SwitchB] vsi vpnc
[SwitchB-vsi-vpnc] gateway vsi-interface 2
[SwitchB-vsi-vpnc] quit
```

## 6. Configure Switch C:

**# Enable L2VPN.**

```
<SwitchC> system-view
[SwitchC] l2vpn enable
```

**# Create VSI `vpna` and VXLAN 10.**

```
[SwitchC] vsi vpna
[SwitchC-vsi-vpna] vxlan 10
[SwitchC-vsi-vpna-vxlan-10] quit
[SwitchC-vsi-vpna] quit
```

**# Create VSI `vpnb` and VXLAN 30.**

```
[SwitchC] vsi vpb
[SwitchC-vsi-vpb] vxlan 30
[SwitchC-vsi-vpb-vxlan-30] quit
[SwitchC-vsi-vpb] quit
```

**# Assign an IP address to Loopback 0. The IP address will be used as the source IP address of the VXLAN tunnels to Switch A and Switch B.**

```
[SwitchC] interface loopback 0
[SwitchC-Loopback0] ip address 3.3.3.3 255.255.255.255
[SwitchC-Loopback0] quit
```

**# Create a VXLAN tunnel to Switch A. The tunnel interface name is **Tunnel 1**.**

```
[SwitchC] interface tunnel 1 mode vxlan
[SwitchC-Tunnel1] source 3.3.3.3
[SwitchC-Tunnel1] destination 1.1.1.1
[SwitchC-Tunnel1] quit
```

**# Create a VXLAN tunnel to Switch B. The tunnel interface name is **Tunnel 3**.**

```
[SwitchC] interface tunnel 3 mode vxlan
[SwitchC-Tunnel3] source 3.3.3.3
[SwitchC-Tunnel3] destination 2.2.2.2
[SwitchC-Tunnel3] quit
```

**# Assign Tunnel 1 and Tunnel 3 to VXLAN 10.**

```
[SwitchC] vsi vpna
[SwitchC-vsi-vpna] vxlan 10
[SwitchC-vsi-vpna-vxlan-10] tunnel 1
[SwitchC-vsi-vpna-vxlan-10] tunnel 3
[SwitchC-vsi-vpna-vxlan-10] quit
[SwitchC-vsi-vpna] quit
```

**# Assign Tunnel 1 and Tunnel 3 to VXLAN 30.**

```
[SwitchC] vsi vpnb
[SwitchC-vsi-vpn] vxlan 30
[SwitchC-vsi-vpn-vxlan-30] tunnel 1
[SwitchC-vsi-vpn-vxlan-30] tunnel 3
[SwitchC-vsi-vpn-vxlan-30] quit
[SwitchC-vsi-vpn] quit
```

**# On Ten-GigabitEthernet 1/0/1, create Ethernet service instance 1000 to match VLAN 4.**

```
[SwitchC] interface ten-gigabitethernet 1/0/1
[SwitchC-Ten-GigabitEthernet1/0/1] port link-type trunk
[SwitchC-Ten-GigabitEthernet1/0/1] port trunk permit vlan 4
[SwitchC-Ten-GigabitEthernet1/0/1] service-instance 1000
[SwitchC-Ten-GigabitEthernet1/0/1-srv1000] encapsulation s-vid 4
```

**# Map Ethernet service instance 1000 to VSI vpnb.**

```
[SwitchC-Ten-GigabitEthernet1/0/1-srv1000] xconnect vsi vpnb
[SwitchC-Ten-GigabitEthernet1/0/1-srv1000] quit
[SwitchC-Ten-GigabitEthernet1/0/1] quit
```

**# Create VSI-interface 1 and assign the interface an IP address and a MAC address. The IP address will be used as the gateway address for VXLAN 10.**

```
[SwitchC] interface vsi-interface 1
[SwitchC-Vsi-interface1] ip address 10.1.1.1 255.255.255.0
[SwitchC-Vsi-interface1] mac-address 1-1-1
```

**# Specify VSI-interface 1 as a distributed gateway and enable local proxy ARP on the interface.**

```
[SwitchC-Vsi-interface1] distributed-gateway local
[SwitchC-Vsi-interface1] local-proxy-arp enable
[SwitchC-Vsi-interface1] quit
```

**# Enable dynamic ARP entry synchronization for distributed VXLAN IP gateways.**

```
[SwitchC] arp distributed-gateway dynamic-entry synchronize
```

**# Specify VSI-interface 1 as the gateway interface for VSI vpna.**

```
[SwitchC] vsi vpna
[SwitchC-vsi-vpna] gateway vsi-interface 1
[SwitchC-vsi-vpna] quit
```

**# Create VSI-interface 2 and assign the interface an IP address and a MAC address. The IP address will be used as the gateway address for VXLAN 30.**

```
[SwitchC] interface vsi-interface 2
[SwitchC-Vsi-interface2] ip address 20.1.1.1 255.255.255.0
[SwitchC-Vsi-interface2] mac-address 2-2-2
```

**# Specify VSI-interface 2 as a distributed gateway and enable local proxy ARP on the interface.**

```
[SwitchC-Vsi-interface2] distributed-gateway local
[SwitchC-Vsi-interface2] local-proxy-arp enable
[SwitchC-Vsi-interface2] quit
```

**# Specify VSI-interface 2 as the gateway interface for VSI vpnb.**

```
[SwitchC] vsi vpnb
[SwitchC-vsi-vpn] gateway vsi-interface 2
[SwitchC-vsi-vpn] quit
```

**# Configure a PBR policy for the VXLANs. Set the policy name to vxlan and set the next hop to 20.1.1.2 (VSI-interface 1 on Switch B).**

```
[SwitchC] acl advanced 3000
```

```

[SwitchC-acl-ipv4-adv-3000] rule 0 permit ip
[SwitchC-acl-ipv4-adv-3000] quit
[SwitchC] policy-based-route vxlan permit node 5
[SwitchC-pbr-vxlan-5] if-match acl 3000
[SwitchC-pbr-vxlan-5] apply next-hop 20.1.1.2
[SwitchC-pbr-vxlan-5] quit
# Apply policy vxlan to VSI-interface 2.
[SwitchC] interface vsi-interface 2
[SwitchC-Vsi-interface2] ip policy-based-route vxlan
[SwitchC-Vsi-interface2] quit
# Configure a default route. Set the next hop to 20.1.1.2 (VSI-interface 1 on Switch B).
[SwitchC] ip route-static 0.0.0.0 0 20.1.1.2

```

## Verifying the configuration

1. Verify the VXLAN IP gateway settings on Switch A:

**# Verify that the VXLAN tunnel interfaces are up on Switch A.**

```

[SwitchA] display interface tunnel 2
Tunnel2
Current state: UP
Line protocol state: UP
Description: Tunnel2 Interface
Bandwidth: 64 kbps
Maximum transmission unit: 1464
Internet protocol processing: Disabled
Last clearing of counters: Never
Tunnel source 1.1.1.1, destination 3.3.3.3
Tunnel protocol/transport UDP_VXLAN/IP
Last 300 seconds input rate: 0 bytes/sec, 0 bits/sec, 0 packets/sec
Last 300 seconds output rate: 0 bytes/sec, 0 bits/sec, 0 packets/sec
Input: 0 packets, 0 bytes, 0 drops
Output: 0 packets, 0 bytes, 0 drops

```

**# Verify that VSI-interface 1 is up.**

```

[SwitchA] display interface vsi-interface 1
Vsi-interfacel
Current state: UP
Line protocol state: UP
Description: Vsi-interfacel Interface
Bandwidth: 1000000 kbps
Maximum transmission unit: 1444
Internet address: 10.1.1.1/24 (primary)
IP packet frame type: Ethernet II, hardware address: 0001-0001-0001
IPv6 packet frame type: Ethernet II, hardware address: 0001-0001-0001
Physical: Unknown, baudrate: 1000000 kbps
Last clearing of counters: Never
Last 300 seconds input rate: 0 bytes/sec, 0 bits/sec, 0 packets/sec
Last 300 seconds output rate: 0 bytes/sec, 0 bits/sec, 0 packets/sec
Input: 0 packets, 0 bytes, 0 drops
Output: 0 packets, 0 bytes, 0 drops

```

# Verify that the VXLAN tunnels have been assigned to VXLAN 10, and VSI-interface 1 is the gateway interface for VSI **vpna**.

```
[SwitchA] display l2vpn vsi name vpna verbose
```

VSI Name: vpna

```
VSI Index          : 0
VSI State          : Up
MTU                : 1500
Bandwidth          : -
Broadcast Restrain : -
Multicast Restrain : -
Unknown Unicast Restrain: -
MAC Learning       : Enabled
MAC Table Limit    : -
MAC Learning rate  : -
Drop Unknown       : -
Flooding           : Enabled
Statistics         : Disabled
Gateway Interface  : VSI-interface 1
VXLAN ID           : 10
```

Tunnels:

| Tunnel Name | Link ID   | State | Type   | Flood proxy |
|-------------|-----------|-------|--------|-------------|
| Tunnel1     | 0x5000001 | Up    | Manual | Disabled    |
| Tunnel2     | 0x5000002 | Up    | Manual | Disabled    |

ACs:

| AC               | Link ID | State | Type   |
|------------------|---------|-------|--------|
| XGE1/0/1 srv1000 | 0       | Up    | Manual |

# Verify that Switch A has created ARP entries for the VMs.

```
[SwitchA] display arp
```

| Type: S-Static | D-Dynamic      | O-Openflow | R-Rule            | M-Multiport | I-Invalid |
|----------------|----------------|------------|-------------------|-------------|-----------|
| IP address     | MAC address    | VLAN/VSI   | Interface/Link ID | Aging       | Type      |
| 11.1.1.4       | 000c-29c1-5e46 | 11         | XGE1/0/1          | 19          | D         |
| 10.1.1.2       | 3c8c-400d-867a | 0          | Tunnel1           | 20          | D         |
| 10.1.1.11      | 0cda-41b5-cf09 | 0          | 0                 | 20          | D         |
| 20.1.1.12      | 0001-0001-0001 | 1          | Tunnel2           | 19          | D         |

## 2. Verify the configuration on the border gateway Switch B:

# Verify that the VXLAN tunnel interfaces are up on Switch B.

```
[SwitchB] display interface tunnel 2
```

Tunnel2

Current state: UP

Line protocol state: UP

Description: Tunnel2 Interface

Bandwidth: 64 kbps

Maximum transmission unit: 1464

Internet protocol processing: Disabled

Last clearing of counters: Never

Tunnel source 2.2.2.2, destination 1.1.1.1

Tunnel protocol/transport UDP\_VXLAN/IP

Last 300 seconds input rate: 0 bytes/sec, 0 bits/sec, 0 packets/sec

```
Last 300 seconds output rate: 0 bytes/sec, 0 bits/sec, 0 packets/sec
Input: 0 packets, 0 bytes, 0 drops
Output: 0 packets, 0 bytes, 0 drops
```

**# Verify that VSI-interface 1 is up.**

```
[SwitchB] display interface vsi-interface 1
```

```
Vsi-interfacel
```

```
Current state: UP
```

```
Line protocol state: UP
```

```
Description: Vsi-interfacel Interface
```

```
Bandwidth: 1000000 kbps
```

```
Maximum transmission unit: 1444
```

```
Internet address: 10.1.1.2/24 (primary)
```

```
IP packet frame type: Ethernet II, hardware address: 0011-2200-0102
```

```
IPv6 packet frame type: Ethernet II, hardware address: 0011-2200-0102
```

```
Physical: Unknown, baudrate: 1000000 kbps
```

```
Last clearing of counters: Never
```

```
Last 300 seconds input rate: 0 bytes/sec, 0 bits/sec, 0 packets/sec
```

```
Last 300 seconds output rate: 0 bytes/sec, 0 bits/sec, 0 packets/sec
```

```
Input: 0 packets, 0 bytes, 0 drops
```

```
Output: 0 packets, 0 bytes, 0 drops
```

**# Verify that the VXLAN tunnels have been assigned to VXLAN 10, and VSI-interface 1 is the gateway interface for VSI vpna.**

```
[SwitchB] display l2vpn vsi name vpna verbose
```

```
VSI Name: vpna
```

```
VSI Index          : 0
VSI State          : Up
MTU                : 1500
Bandwidth          : -
Broadcast Restrain : -
Multicast Restrain : -
Unknown Unicast Restrain: -
MAC Learning       : Enabled
MAC Table Limit    : -
MAC Learning rate  : -
Drop Unknown       : -
Flooding           : Enabled
Statistics         : Disabled
Gateway interface  : VSI-interface 1
VXLAN ID           : 10
```

```
Tunnels:
```

| Tunnel Name | Link ID   | State | Type   | Flood proxy |
|-------------|-----------|-------|--------|-------------|
| Tunnel2     | 0x5000002 | Up    | Manual | Disabled    |

**# Verify that Switch B has created ARP entries for the VMs.**

```
[SwitchB] display arp
```

| Type: S-Static | D-Dynamic      | O-Openflow | R-Rule            | M-Multiport | I-Invalid |
|----------------|----------------|------------|-------------------|-------------|-----------|
| IP address     | MAC address    | VLAN/VSI   | Interface/Link ID | Aging       | Type      |
| 12.1.1.4       | 0000-fc00-00ab | 12         | Vlan12            | 14          | D         |
| 25.1.1.5       | 4431-9234-24bb | 20         | Vlan20            | 17          | D         |

```

10.1.1.1          0001-0001-0001 0          Tunnel2          17    D
10.1.1.11         0001-0001-0001 0          Tunnel2          20    D
20.1.1.1          0002-0002-0002 1          Tunnel3          17    D
20.1.1.12         0002-0002-0002 1          Tunnel3          20    D

```

# Verify that Switch B has created FIB entries for the VMs.

```
[SwitchB] display fib 10.1.1.11
```

```
Destination count: 1 FIB entry count: 1
```

```
Flag:
```

```
U:Usable G:Gateway H:Host B:Blackhole D:Dynamic S:Static
```

```
R:Relay F:FRR
```

```

Destination/Mask  Nexthop          Flag    OutInterface/Token  Label
10.1.1.11/32     10.1.1.11       UH      Vsi1                 Null

```

```
[SwitchB] display fib 20.1.1.12
```

```
Destination count: 1 FIB entry count: 1
```

```
Flag:
```

```
U:Usable G:Gateway H:Host B:Blackhole D:Dynamic S:Static
```

```
R:Relay F:FRR
```

```

Destination/Mask  Nexthop          Flag    OutInterface/Token  Label
20.1.1.12/32     20.1.1.12       UH      Vsi3                 Null

```

### 3. Verify that the network connectivity for VMs meets the requirements:

# Verify that VM 1 and VM 3 can ping each other. (Details not shown.)

# Verify that VM 1 and VM 3 can ping VLAN-interface 20 (25.1.1.5) on Switch E for WAN access. (Details not shown.)

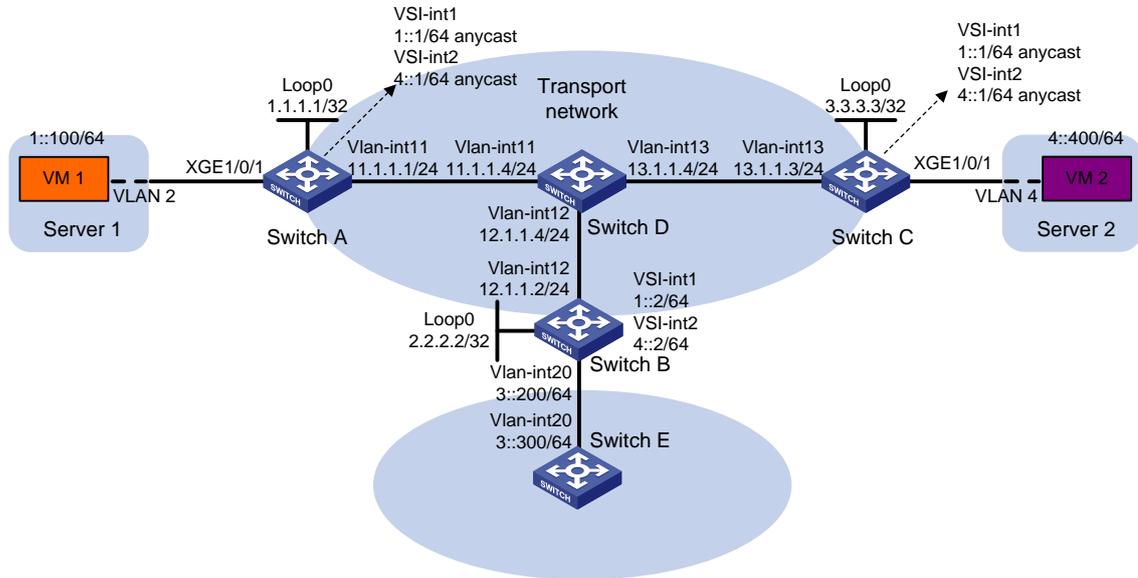
## Example: Configuring distributed VXLAN IPv6 gateways

### Network configuration

As shown in [Figure 17](#):

- Configure VXLAN 10 and VXLAN 20 as unicast-mode VXLANs on Switch A, Switch B, and Switch C to provide connectivity for the VMs across the network sites.
- Manually establish VXLAN tunnels and assign the tunnels to the VXLANs.
- Configure distributed VXLAN IP gateways on Switch A and Switch C to forward traffic between the VXLANs.
- Configure Switch B as a border gateway to forward traffic between the VXLANs and the WAN connected to Switch E.

Figure 17 Network diagram



## Procedure

1. Set the system operation mode to VXLAN on Switch A, Switch B, and Switch C. This step uses Switch A as an example.

```
<SwitchA> system-view
[SwitchA] switch-mode 1
Reboot device to make the configuration take effect.
[SwitchA] quit
<SwitchA> reboot
Start to check configuration with next startup configuration file, please wait..
.....DONE!
Current configuration may be lost after the reboot, save current configuration?
[Y/N]:y
This command will reboot the device. Continue? [Y/N]:y
```

2. On VM 1 and VM 2, specify 1::1 and 4::1 as the gateway address, respectively. (Details not shown.)
3. Configure IP addresses and unicast routing settings:
  - # Assign IP addresses to interfaces, as shown in Figure 17. (Details not shown.)
  - # Configure OSPF on all transport network switches (switches A through D). (Details not shown.)
  - # Configure OSPFv3 to advertise routes to networks 1::/64, 4::/64, and 3::/64 on Switch B and Switch E. (Details not shown.)

4. Configure Switch A:

```
# Enable L2VPN.
<SwitchA> system-view
[SwitchA] l2vpn enable
# Create VSI vpna and VXLAN 10.
[SwitchA] vsi vpna
[SwitchA-vsi-vpna] vxlan 10
[SwitchA-vsi-vpna-vxlan-10] quit
[SwitchA-vsi-vpna] quit
```

**# Create VSI `vpnb` and VXLAN 20.**

```
[SwitchA] vsi vpng
[SwitchA-vsi-vpng] vxlan 20
[SwitchA-vsi-vpng-vxlan-20] quit
[SwitchA-vsi-vpng] quit
```

**# Assign an IP address to Loopback 0. The IP address will be used as the source IP address of the VXLAN tunnels to Switch B and Switch C.**

```
[SwitchA] interface loopback 0
[SwitchA-Loopback0] ip address 1.1.1.1 255.255.255.255
[SwitchA-Loopback0] quit
```

**# Create a VXLAN tunnel to Switch B. The tunnel interface name is `Tunnel 1`.**

```
[SwitchA] interface tunnel 1 mode vxlan
[SwitchA-Tunnel1] source 1.1.1.1
[SwitchA-Tunnel1] destination 2.2.2.2
[SwitchA-Tunnel1] quit
```

**# Create a VXLAN tunnel to Switch C. The tunnel interface name is `Tunnel 2`.**

```
[SwitchA] interface tunnel 2 mode vxlan
[SwitchA-Tunnel2] source 1.1.1.1
[SwitchA-Tunnel2] destination 3.3.3.3
[SwitchA-Tunnel2] quit
```

**# Assign Tunnel 1 and Tunnel 2 to VXLAN 10.**

```
[SwitchA] vsi vpna
[SwitchA-vsi-vpna] vxlan 10
[SwitchA-vsi-vpna-vxlan-10] tunnel 1
[SwitchA-vsi-vpna-vxlan-10] tunnel 2
[SwitchA-vsi-vpna-vxlan-10] quit
[SwitchA-vsi-vpna] quit
```

**# Assign Tunnel 1 and Tunnel 2 to VXLAN 20.**

```
[SwitchA] vsi vpng
[SwitchA-vsi-vpng] vxlan 20
[SwitchA-vsi-vpng-vxlan-20] tunnel 1
[SwitchA-vsi-vpng-vxlan-20] tunnel 2
[SwitchA-vsi-vpng-vxlan-20] quit
[SwitchA-vsi-vpng] quit
```

**# On Ten-GigabitEthernet 1/0/1, create Ethernet service instance 1000 to match VLAN 2.**

```
[SwitchA] interface ten-gigabitethernet 1/0/1
[SwitchA-Ten-GigabitEthernet1/0/1] port link-type trunk
[SwitchA-Ten-GigabitEthernet1/0/1] port trunk permit vlan 2
[SwitchA-Ten-GigabitEthernet1/0/1] service-instance 1000
[SwitchA-Ten-GigabitEthernet1/0/1-srv1000] encapsulation s-vid 2
```

**# Map Ethernet service instance 1000 to VSI `vpna`.**

```
[SwitchA-Ten-GigabitEthernet1/0/1-srv1000] xconnect vsi vpna
[SwitchA-Ten-GigabitEthernet1/0/1-srv1000] quit
[SwitchA-Ten-GigabitEthernet1/0/1] quit
```

**# Enable dynamic ND entry synchronization for distributed VXLAN IP gateways.**

```
[SwitchA] ipv6 nd distributed-gateway dynamic-entry synchronize
```

**# Create VSI-interface 1 and assign the interface an IPv6 anycast address. The IP address will be used as the gateway address for VXLAN 10.**

```

[SwitchA] interface vsi-interface 1
[SwitchA-Vsi-interface1] ipv6 address 1::1/64 anycast
# Specify VSI-interface 1 as a distributed gateway and enable local ND proxy on the interface.
[SwitchA-Vsi-interface1] distributed-gateway local
[SwitchA-Vsi-interface1] local-proxy-nd enable
[SwitchA-Vsi-interface1] quit
# Specify VSI-interface 1 as the gateway interface for VSI vpna.
[SwitchA] vsi vpna
[SwitchA-vsi-vpna] gateway vsi-interface 1
[SwitchA-vsi-vpna] quit
# Create VSI-interface 2 and assign the interface an IPv6 anycast address. The IP address will be used as the gateway address for VXLAN 20.
[SwitchA] interface vsi-interface 2
[SwitchA-Vsi-interface2] ipv6 address 4::1/64 anycast
# Specify VSI-interface 2 as a distributed gateway and enable local ND proxy on the interface.
[SwitchA-Vsi-interface2] distributed-gateway local
[SwitchA-Vsi-interface2] local-proxy-nd enable
[SwitchA-Vsi-interface2] quit
# Specify VSI-interface 2 as the gateway interface for VSI vpb.
[SwitchA] vsi vpb
[SwitchA-vsi-vpb] gateway vsi-interface 2
[SwitchA-vsi-vpb] quit
# Configure an IPv6 static route. Set the destination address to 3::/64 and the next hop to 1::2.
[SwitchA] ipv6 route-static 3:: 64 1::2

```

## 5. Configure Switch B:

```

# Enable L2VPN.
<SwitchB> system-view
[SwitchB] l2vpn enable
# Create VSI vpna and VXLAN 10.
[SwitchB] vsi vpna
[SwitchB-vsi-vpna] vxlan 10
[SwitchB-vsi-vpna-vxlan-10] quit
[SwitchB-vsi-vpna] quit
# Create VSI vpb and VXLAN 20.
[SwitchB] vsi vpb
[SwitchB-vsi-vpb] vxlan 20
[SwitchB-vsi-vpb-vxlan-20] quit
[SwitchB-vsi-vpb] quit
# Assign an IP address to Loopback 0. The IP address will be used as the source IP address of the VXLAN tunnels to Switch A and Switch C.
[SwitchB] interface loopback 0
[SwitchB-Loopback0] ip address 2.2.2.2 255.255.255.255
[SwitchB-Loopback0] quit
# Create a VXLAN tunnel to Switch A. The tunnel interface name is Tunnel 2.
[SwitchB] interface tunnel 2 mode vxlan
[SwitchB-Tunnel2] source 2.2.2.2
[SwitchB-Tunnel2] destination 1.1.1.1
[SwitchB-Tunnel2] quit

```

**# Create a VXLAN tunnel to Switch C. The tunnel interface name is Tunnel 3.**

```
[SwitchB] interface tunnel 3 mode vxlan
[SwitchB-Tunnel3] source 2.2.2.2
[SwitchB-Tunnel3] destination 3.3.3.3
[SwitchB-Tunnel3] quit
```

**# Assign Tunnel 2 and Tunnel 3 to VXLAN 10.**

```
[SwitchB] vsi vpna
[SwitchB-vsi-vpna] vxlan 10
[SwitchB-vsi-vpna-vxlan-10] tunnel 2
[SwitchB-vsi-vpna-vxlan-10] tunnel 3
[SwitchB-vsi-vpna-vxlan-10] quit
[SwitchB-vsi-vpna] quit
```

**# Assign Tunnel 2 and Tunnel 3 to VXLAN 20.**

```
[SwitchB] vsi vpb
[SwitchB-vsi-vpb] vxlan 20
[SwitchB-vsi-vpb-vxlan-20] tunnel 2
[SwitchB-vsi-vpb-vxlan-20] tunnel 3
[SwitchB-vsi-vpb-vxlan-20] quit
[SwitchB-vsi-vpb] quit
```

**# Create VSI-interface 1 and assign the interface an IPv6 address.**

```
[SwitchB] interface vsi-interface 1
[SwitchB-Vsi-interfacel] ipv6 address 1::2/64
[SwitchB-Vsi-interfacel] quit
```

**# Create VSI-interface 2 and assign the interface an IPv6 address.**

```
[SwitchB] interface vsi-interface 2
[SwitchB-Vsi-interface2] ipv6 address 4::2/64
[SwitchB-Vsi-interface2] quit
```

**# Specify VSI-interface 1 as the gateway interface for VSI vpna.**

```
[SwitchB] vsi vpna
[SwitchB-vsi-vpna] gateway vsi-interface 1
[SwitchB-vsi-vpna] quit
```

**# Specify VSI-interface 2 as the gateway interface for VSI vpb.**

```
[SwitchB] vsi vpb
[SwitchB-vsi-vpb] gateway vsi-interface 2
[SwitchB-vsi-vpb] quit
```

## **6. Configure Switch C:**

**# Enable L2VPN.**

```
<SwitchC> system-view
[SwitchC] l2vpn enable
```

**# Create VSI vpna and VXLAN 10.**

```
[SwitchC] vsi vpna
[SwitchC-vsi-vpna] vxlan 10
[SwitchC-vsi-vpna-vxlan-10] quit
[SwitchC-vsi-vpna] quit
```

**# Create VSI vpb and VXLAN 20.**

```
[SwitchC] vsi vpb
[SwitchC-vsi-vpb] vxlan 20
```

```

[SwitchC-vsi-vpnb-vxlan-20] quit
[SwitchC-vsi-vpnb] quit
# Assign an IP address to Loopback 0. The IP address will be used as the source IP address of the VXLAN tunnels to Switch A and Switch B.
[SwitchC] interface loopback 0
[SwitchC-Loopback0] ip address 3.3.3.3 255.255.255.255
[SwitchC-Loopback0] quit
# Create a VXLAN tunnel to Switch A. The tunnel interface name is Tunnel 1.
[SwitchC] interface tunnel 1 mode vxlan
[SwitchC-Tunnel1] source 3.3.3.3
[SwitchC-Tunnel1] destination 1.1.1.1
[SwitchC-Tunnel1] quit
# Create a VXLAN tunnel to Switch B. The tunnel interface name is Tunnel 3.
[SwitchC] interface tunnel 3 mode vxlan
[SwitchC-Tunnel3] source 3.3.3.3
[SwitchC-Tunnel3] destination 2.2.2.2
[SwitchC-Tunnel3] quit
# Assign Tunnel 1 and Tunnel 3 to VXLAN 10.
[SwitchC] vsi vpna
[SwitchC-vsi-vpna] vxlan 10
[SwitchC-vsi-vpna-vxlan-10] tunnel 1
[SwitchC-vsi-vpna-vxlan-10] tunnel 3
[SwitchC-vsi-vpna-vxlan-10] quit
[SwitchC-vsi-vpna] quit
# Assign Tunnel 1 and Tunnel 3 to VXLAN 20.
[SwitchC] vsi vpb
[SwitchC-vsi-vpnb] vxlan 20
[SwitchC-vsi-vpnb-vxlan-20] tunnel 1
[SwitchC-vsi-vpnb-vxlan-20] tunnel 3
[SwitchC-vsi-vpnb-vxlan-20] quit
[SwitchC-vsi-vpnb] quit
# On Ten-GigabitEthernet 1/0/1, create Ethernet service instance 1000 to match VLAN 4.
[SwitchC] interface ten-gigabitethernet 1/0/1
[SwitchC-Ten-GigabitEthernet1/0/1] port link-type trunk
[SwitchC-Ten-GigabitEthernet1/0/1] port trunk permit vlan 4
[SwitchC-Ten-GigabitEthernet1/0/1] service-instance 1000
[SwitchC-Ten-GigabitEthernet1/0/1-srv1000] encapsulation s-vid 4
# Map Ethernet service instance 1000 to VSI vpb.
[SwitchC-Ten-GigabitEthernet1/0/1-srv1000] xconnect vsi vpb
[SwitchC-Ten-GigabitEthernet1/0/1-srv1000] quit
[SwitchC-Ten-GigabitEthernet1/0/1] quit
# Enable dynamic ND entry synchronization for distributed VXLAN IP gateways.
[SwitchC] ipv6 nd distributed-gateway dynamic-entry synchronize
# Create VSI-interface 1 and assign the interface an IPv6 anycast address. The IP address will be used as the gateway address for VXLAN 10.
[SwitchC] interface vsi-interface 1
[SwitchC-Vsi-interfacel] ipv6 address 1::1/64 anycast
# Specify VSI-interface 1 as a distributed gateway and enable local ND proxy on the interface.

```

```

[SwitchC-Vsi-interface1] distributed-gateway local
[SwitchC-Vsi-interface1] local-proxy-nd enable
[SwitchC-Vsi-interface1] quit
# Specify VSI-interface 1 as the gateway interface for VSI vpna.
[SwitchC] vsi vpna
[SwitchC-vsi-vpna] gateway vsi-interface 1
[SwitchC-vsi-vpna] quit
# Create VSI-interface 2 and assign the interface an IPv6 anycast address. The IP address will be used as the gateway address for VXLAN 20.
[SwitchC] interface vsi-interface 2
[SwitchC-Vsi-interface2] ipv6 address 4::1/64 anycast
# Specify VSI-interface 2 as a distributed gateway and enable local ND proxy on the interface.
[SwitchC-Vsi-interface2] distributed-gateway local
[SwitchC-Vsi-interface2] local-proxy-nd enable
[SwitchC-Vsi-interface2] quit
# Specify VSI-interface 2 as the gateway interface for VSI vpb.
[SwitchC] vsi vpb
[SwitchC-vsi-vpb] gateway vsi-interface 2
[SwitchC-vsi-vpb] quit
# Configure an IPv6 static route. Set the destination address to 3::/64 and the next hop to 4::2.
[SwitchC] ipv6 route-static 3:: 64 4::2

```

## Verifying the configuration

1. Verify the distributed VXLAN IP gateway settings on Switch A:

**# Verify that the VXLAN tunnel interfaces are up on Switch A.**

```

[SwitchA] display interface tunnel 2
Tunnel2
Current state: UP
Line protocol state: UP
Description: Tunnel2 Interface
Bandwidth: 64 kbps
Maximum transmission unit: 1464
Internet protocol processing: Disabled
Last clearing of counters: Never
Tunnel source 1.1.1.1, destination 3.3.3.3
Tunnel protocol/transport UDP_VXLAN/IP
Last 300 seconds input rate: 0 bytes/sec, 0 bits/sec, 0 packets/sec
Last 300 seconds output rate: 0 bytes/sec, 0 bits/sec, 0 packets/sec
Input: 0 packets, 0 bytes, 0 drops
Output: 0 packets, 0 bytes, 0 drops

```

**# Verify that the VSI interfaces are up.**

```

[SwitchA] display interface vsi-interface 1
Vsi-interfacel
Current state: UP
Line protocol state: UP
Description: Vsi-interfacel Interface
Bandwidth: 1000000 kbps
Maximum transmission unit: 1444

```

```

Internet protocol processing: Disabled
IP packet frame type: Ethernet II, hardware address: 0011-2200-0102
IPv6 packet frame type: Ethernet II, hardware address: 0011-2200-0102
Physical: Unknown, baudrate: 1000000 kbps
Last clearing of counters: Never
Last 300 seconds input rate: 0 bytes/sec, 0 bits/sec, 0 packets/sec
Last 300 seconds output rate: 0 bytes/sec, 0 bits/sec, 0 packets/sec
Input: 0 packets, 0 bytes, 0 drops
Output: 0 packets, 0 bytes, 0 drops
# Verify that the VXLAN tunnels have been assigned to the VXLANs, and the VSI interfaces are the gateway interfaces for the VSIs.

```

```
[SwitchA] display l2vpn vsi verbose
```

```
VSI Name: vpna
```

```

VSI Index           : 0
VSI State           : Up
MTU                 : 1500
Bandwidth           : -
Broadcast Restrain  : -
Multicast Restrain  : -
Unknown Unicast Restrain: -
MAC Learning        : Enabled
MAC Table Limit     : -
MAC Learning rate   : -
Drop Unknown        : -
Flooding            : Enabled
Statistics          : Disabled
Gateway Interface   : VSI-interface 1
VXLAN ID            : 10

```

```
Tunnels:
```

| Tunnel Name | Link ID   | State | Type   | Flood proxy |
|-------------|-----------|-------|--------|-------------|
| Tunnel1     | 0x5000001 | Up    | Manual | Disabled    |
| Tunnel2     | 0x5000002 | Up    | Manual | Disabled    |

```
ACs:
```

| AC               | Link ID | State | Type   |
|------------------|---------|-------|--------|
| XGE1/0/1 srv1000 | 0       | Up    | Manual |

```
VSI Name: vpb
```

```

VSI Index           : 0
VSI State           : Up
MTU                 : 1500
Bandwidth           : -
Broadcast Restrain  : -
Multicast Restrain  : -
Unknown Unicast Restrain: -
MAC Learning        : Enabled
MAC Table Limit     : -
MAC Learning rate   : -
Drop Unknown        : -
Flooding            : Enabled

```

```

Statistics                : Disabled
Gateway Interface        : VSI-interface 2
VXLAN ID                 : 20
Tunnels:
  Tunnel Name            Link ID   State  Type      Flood proxy
  -----
  Tunnel1                0x5000001  Up     Manual    Disabled
  Tunnel2                0x5000002  Up     Manual    Disabled

```

**# Verify that Switch A has created neighbor entries for the VMs.**

```

[SwitchA] display ipv6 neighbors all
Type: S-Static   D-Dynamic   O-Openflow   R-Rule   I-Invalid
IPv6 address          Link layer   VID   Interface   State T   Age
1::2                  3c8c-400d-867a 0     Tunnel1     STALE D   7
1::100                0001-0000-0047 0     0           STALE D   22
4::400                0002-0000-0047 1     Tunnel2     REACH D   5
FE80::201:FF:FE00:47 0001-0000-0047 0     Tunnel1     REACH D   30
FE80::202:FF:FE00:0  0002-0000-0000 1     Tunnel2     REACH D   27
FE80::202:FF:FE00:47 0002-0000-0047 0     0           DELAY D   5

```

**# Verify that Switch A has created FIB entries for the VMs.**

```

[SwitchA] display ipv6 fib 4::400
Destination count: 1 FIB entry count: 1
Flag:
  U:Usable   G:Gateway   H:Host   B:Blackhole   D:Dynamic   S:Static
  R:Relay    F:FRR
Destination: 4::400                               Prefix length: 128
Nexthop      : 4::400                               Flags: UH
Time stamp   : 0x2c                                 Label: Null
Interface    : Tunnel2                             Token: Invalid

```

```

[SwitchA] display ipv6 fib 3::300
Destination count: 1 FIB entry count: 1
Flag:
  U:Usable   G:Gateway   H:Host   B:Blackhole   D:Dynamic   S:Static
  R:Relay    F:FRR
Destination: 3::                               Prefix length: 40
Nexthop      : 1::2                               Flags: USGR
Time stamp   : 0x23                               Label: Null
Interface    : Tunnel1                             Token: Invalid

```

**2. Verify the configuration on the border gateway Switch B:**

**# Verify that the VXLAN tunnel interfaces are up on Switch B.**

```

[SwitchB] display interface tunnel 2
Tunnel2
Current state: UP
Line protocol state: UP
Description: Tunnel2 Interface
Bandwidth: 64 kbps
Maximum transmission unit: 1464
Internet protocol processing: Disabled
Last clearing of counters: Never
Tunnel source 2.2.2.2, destination 1.1.1.1

```

```
Tunnel protocol/transport UDP_VXLAN/IP
Last 300 seconds input rate: 0 bytes/sec, 0 bits/sec, 0 packets/sec
Last 300 seconds output rate: 0 bytes/sec, 0 bits/sec, 0 packets/sec
Input: 0 packets, 0 bytes, 0 drops
Output: 0 packets, 0 bytes, 0 drops
```

**# Verify that the VSI interfaces are up.**

```
[SwitchB] display interface Vsi-interface 1
Vsi-interfacel
Current state: UP
Line protocol state: UP
Description: Vsi-interfacel Interface
Bandwidth: 1000000 kbps
Maximum transmission unit: 1444
Internet protocol processing: Disabled
IP packet frame type: Ethernet II, hardware address: 0011-2200-0102
IPv6 packet frame type: Ethernet II, hardware address: 0011-2200-0102
Physical: Unknown, baudrate: 1000000 kbps
Last clearing of counters: Never
Last 300 seconds input rate: 0 bytes/sec, 0 bits/sec, 0 packets/sec
Last 300 seconds output rate: 0 bytes/sec, 0 bits/sec, 0 packets/sec
Input: 0 packets, 0 bytes, 0 drops
Output: 0 packets, 0 bytes, 0 drops
```

**# Verify that the VXLAN tunnels have been assigned to the VXLANs, and the VSI interfaces are the gateway interfaces for the VSIs.**

```
[SwitchB] display l2vpn vsi name vpna verbose
VSI Name: vpna
VSI Index          : 0
VSI State          : Up
MTU                : 1500
Bandwidth          : -
Broadcast Restrain : -
Multicast Restrain : -
Unknown Unicast Restrain: -
MAC Learning       : Enabled
MAC Table Limit    : -
MAC Learning rate  : -
Drop Unknown       : -
Flooding           : Enabled
Statistics         : Disabled
Gateway interface  : VSI-interface 1
VXLAN ID           : 10
Tunnels:
  Tunnel Name      Link ID   State  Type      Flood proxy
  Tunnel1         0x5000001 Up      Manual    Disabled
  Tunnel2         0x5000002 Up      Manual    Disabled
```

**# Verify that Switch B has created neighbor entries for the VMs.**

```
[SwitchB] display ipv6 neighbors all
Type: S-Static    D-Dynamic    O-Openflow    R-Rule    I-Invalid
```

| IPv6 address         | Link layer     | VID | Interface | State | T | Age |
|----------------------|----------------|-----|-----------|-------|---|-----|
| 3::300               | 0003-0000-0047 | 20  | Vlan20    | DELAY | D | 3   |
| FE80::203:FF:FE00:47 | 0003-0000-0047 | 20  | Vlan20    | STALE | D | 222 |
| 1::100               | 0001-0000-0047 | 0   | Tunnel2   | STALE | D | 232 |
| 4::400               | 0002-0000-0047 | 1   | Tunnel3   | REACH | D | 3   |
| FE80::201:FF:FE00:0  | 0001-0000-0000 | 0   | Tunnel2   | STALE | D | 237 |
| FE80::201:FF:FE00:47 | 0001-0000-0047 | 20  | Vlan20    | STALE | D | 222 |
| FE80::202:FF:FE00:0  | 0002-0000-0000 | 1   | Tunnel3   | STALE | D | 345 |

# Verify that Switch B has created FIB entries for the VMs.

[SwitchB] display ipv6 fib 1::100

Destination count: 1 FIB entry count: 1

Flag:

U:Usable G:Gateway H:Host B:Blackhole D:Dynamic S:Static  
R:Relay F:FRR

Destination: 1::100

Prefix length: 128

Nexthop : 1::100

Flags: UH

Time stamp : 0x21

Label: Null

Interface : Tunnel2

Token: Invalid

[SwitchB] display ipv6 fib 4::400

Destination count: 1 FIB entry count: 1

Flag:

U:Usable G:Gateway H:Host B:Blackhole D:Dynamic S:Static  
R:Relay F:FRR

Destination: 4::

Prefix length: 64

Nexthop : ::

Flags: U

Time stamp : 0x19

Label: Null

Interface : Tunnel3

Token: Invalid

### 3. Verify the network connectivity for the VMs:

# Verify that VM 1 and VM 2 can ping each other. (Details not shown.)

# Verify that VM 1, VM 2, and VLAN-interface 20 (3::300) on Switch E can ping each other. (Details not shown.)

# Configuring the VTEP as an OVSDb VTEP

## About OVSDb VTEP

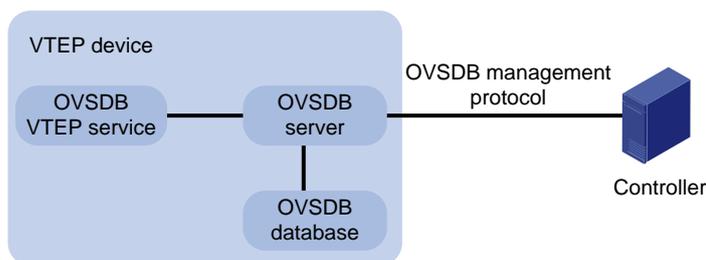
An H3C network virtualization controller can use the Open vSwitch Database (OVSDb) management protocol to deploy and manage VXLANs on VTEPs. To work with a controller, you must configure the VTEP as an OVSDb VTEP.

As shown in [Figure 18](#), an OVSDb VTEP stores all of its VXLAN settings in the form of entries in an OVSDb database. The OVSDb database, OVSDb VTEP service, and the controller interact through the OVSDb server. The controller communicates with the OVSDb server through the OVSDb protocol to manage the OVSDb database. The OVSDb VTEP service reads and writes data in the OVSDb database through the OVSDb server.

The OVSDb VTEP service performs the following operations to manage the VXLAN settings on the VTEP:

- Converts data in the OVSDb database into VXLAN configuration and deploys the configuration to the VTEP. For example, create or remove a VXLAN or VXLAN tunnel.
- Adds site-facing interface information and the global source address of VXLAN tunnels to the OVSDb database. The information is reported to the controller by the OVSDb server.

**Figure 18 OVSDb network model**



## Protocols and standards

RFC 7047, *The Open vSwitch Database Management Protocol*

## Restrictions and guidelines: OVSDb VTEP configuration

You can configure a VTEP both at the CLI and through a controller. As a best practice, do not manually remove the VXLAN configuration issued by the controller.

## OVSDb VTEP tasks at a glance

To configure OVSDb VTEPs, perform the following tasks:

1. [Setting up an OVSDb connection to a controller](#)
  - [Configuring active SSL connection settings](#)
  - [Configuring passive SSL connection settings](#)

- Configuring active TCP connection settings
- Configuring passive TCP connection settings
- 2. Enabling the OVSDB server
- 3. Enabling the OVSDB VTEP service
- 4. Specifying a global source address for VXLAN tunnels
- 5. Specifying a VTEP access port
- 6. (Optional.) Disabling the ACLs issued by the OVSDB controller

## Prerequisites for OVSDB VTEP configuration

Before you configure the VTEP as an OVSDB VTEP, enable L2VPN by using the `l2vpn enable` command.

Before you set up SSL connections to controllers, you must configure SSL as described in *Security Configuration Guide*.

## Setting up an OVSDB connection to a controller

### About OVSDB connection types

The OVSDB server supports the following types of OVSDB connections:

- **Active SSL connection**—The OVSDB server initiates an SSL connection to the controller.
- **Passive SSL connection**—The OVSDB server accepts the SSL connection from the controller.
- **Active TCP connection**—The OVSDB server initiates a TCP connection to the controller.
- **Passive TCP connection**—The OVSDB server accepts the TCP connection from the controller.

### Restrictions and guidelines for OVSDB controller connection setup

When you set up OVSDB connections, follow these restrictions and guidelines:

- You can set up multiple OVSDB connections. For the device to establish the connections, you must enable the OVSDB server. You must disable and then re-enable the OVSDB server if it has been enabled.
- You must specify the same PKI domain and CA certificate file for all active and passive SSL connections.

### Prerequisites for OVSDB controller connection setup

Make sure you have configured a PKI domain before specifying it for SSL. For more information about configuring a PKI domain, see *Security Configuration Guide*.

### Configuring active SSL connection settings

1. Enter system view.  
`system-view`

2. Specify a PKI domain for SSL.

```
ovsdb server pki domain domain-name
```

By default, no PKI domain is specified for SSL.

3. (Optional.) Specify a CA certificate file for SSL.

```
ovsdb server bootstrap ca-certificate ca-filename
```

By default, SSL uses the CA certificate file in the PKI domain.

If the specified CA certificate file does not exist, the device obtains a self-signed certificate from the controller. The obtained file uses the name specified for the *ca-filename* argument.

4. Set up an active SSL connection.

```
ovsdb server ssl ip ip-address port port-number
```

By default, the device does not have active OVSDDB SSL connections.

You can set up a maximum of eight OVSDDB SSL connections.

## Configuring passive SSL connection settings

1. Enter system view.

```
system-view
```

2. Specify a PKI domain for SSL.

```
ovsdb server pki domain domain-name
```

By default, no PKI domain is specified for SSL.

3. (Optional.) Specify a CA certificate file for SSL.

```
ovsdb server bootstrap ca-certificate ca-filename
```

By default, SSL uses the CA certificate file in the PKI domain.

If the specified CA certificate file does not exist, the device obtains a self-signed certificate from the controller. The obtained file uses the name specified for the *ca-filename* argument.

4. Enable the device to listen for SSL connection requests.

```
ovsdb server pssl [ port port-number ]
```

By default, the device does not listen for SSL connection requests.

You can specify only one port to listen for OVSDDB SSL connection requests.

## Configuring active TCP connection settings

1. Enter system view.

```
system-view
```

2. Set up an active TCP connection.

```
ovsdb server tcp ip ip-address port port-number
```

By default, the device does not have active OVSDDB TCP connections.

You can set up a maximum of eight active OVSDDB TCP connections.

## Configuring passive TCP connection settings

1. Enter system view.

```
system-view
```

2. Enable the device to listen for TCP connection requests.

```
ovsdb server ptcp [ port port-number ]
```

By default, the device does not listen for TCP connection requests.  
You can specify only one port to listen for OVSDB TCP connection requests.

## Enabling the OVSDB server

### Prerequisites

Make sure you have complete OVSDB connection setup before you enable the OVSDB server. If you change OVSDB connection settings after the OVSDB server is enabled, you must disable and then re-enable the OVSDB server for the change to take effect.

### Procedure

1. Enter system view.  
**system-view**
2. Enable the OVSDB server.  
**ovsdb server enable**  
By default, the OVSDB server is disabled.

## Enabling the OVSDB VTEP service

1. Enter system view.  
**system-view**
2. Enable the OVSDB VTEP service.  
**vtep enable**  
By default, the OVSDB VTEP service is disabled.

## Specifying a global source address for VXLAN tunnels

### About the global VXLAN tunnel source address

The VTEP reports the global VXLAN tunnel source address to the controller for VXLAN tunnel setup.

### Restrictions and guidelines

For correct VXLAN deployment and VTEP management, do not manually specify tunnel-specific source addresses for VXLAN tunnels if OVSDB is used.

### Procedure

1. Enter system view.  
**system-view**
2. Specify a global source address for VXLAN tunnels.  
**tunnel global source-address *ip-address***  
By default, no global source address is specified for VXLAN tunnels.

# Specifying a VTEP access port

## About specifying a VTEP access port

For the controller to manage a site-facing interface, you must specify the interface as a VTEP access port.

### Procedure

1. Enter system view.  
`system-view`
2. Enter interface view.  
`interface interface-type interface-number`
3. Specify the interface as a VTEP access port.  
`vtep access port`

By default, an interface is not a VTEP access port.

# Disabling the ACLs issued by the OVSDB controller

## About disabling the ACLs issued by the OVSDB controller

Perform this task on a VTEP to disable all the ACLs issued by the OVSDB controller in order to save ACL resources on the VTEP.

### Prerequisites

Before you perform this task, you must enable the OVSDB VTEP service by using the `vtep enable` command.

### Procedure

1. Enter system view.  
`system-view`
2. Disable the ACLs issued by the OVSDB controller.  
`vtep acl disable`

By default, the ACLs issued by the OVSDB controller are enabled on a VTEP.

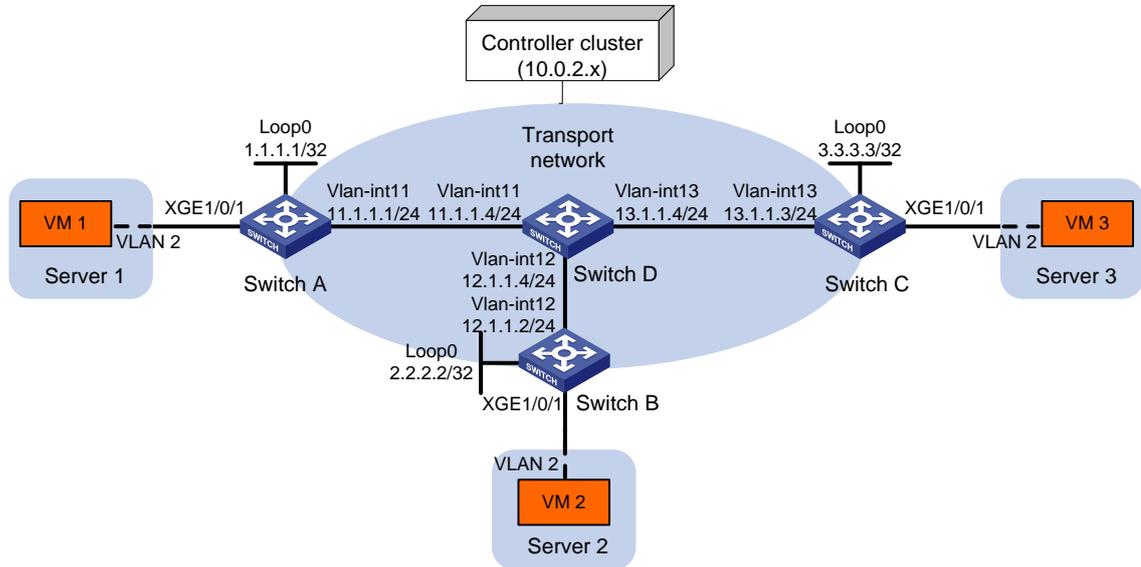
# OVSDB VTEP configuration examples

## Example: Configuring a unicast-mode VXLAN

### Network configuration

As shown in [Figure 19](#), configure the controller cluster to deploy unicast-mode VXLAN 10 to Switch A, Switch B, and Switch C to provide Layer 2 connectivity for the VMs across the network sites.

**Figure 19 Network diagram**



## Procedure

1. Set the system operating mode to VXLAN on Switch A, Switch B, and Switch C. This step uses Switch A as an example.

```
<SwitchA> system-view
```

```
[SwitchA] switch-mode 1
```

Reboot device to make the configuration take effect.

```
[SwitchA] quit
```

```
<SwitchA> reboot
```

Start to check configuration with next startup configuration file, please wait..

.....DONE!

Current configuration may be lost after the reboot, save current configuration?

```
[Y/N]:y
```

This command will reboot the device. Continue? [Y/N]:y

2. Configure IP addresses and unicast routing settings:
  - # Assign IP addresses to interfaces, as shown in [Figure 19](#). (Details not shown.)
  - # Configure OSPF on all transport network switches (Switches A through D). (Details not shown.)

3. Configure Switch A:

# Enable L2VPN.

```
<SwitchA> system-view
```

```
[SwitchA] l2vpn enable
```

# Configure active TCP connection settings.

```
[SwitchA] ovssdb server tcp ip 10.0.2.15 port 6632
```

# Enable the OVSSDB server.

```
[SwitchA] ovssdb server enable
```

# Enable the OVSSDB VTEP service.

```
[SwitchA] vtep enable
```

# Assign an IP address to Loopback 0. Specify the IP address as the global source address for VXLAN tunnels.

```
[SwitchA] interface loopback 0
```

```
[SwitchA-LoopBack0] ip address 1.1.1.1 255.255.255.255
[SwitchA-LoopBack0] quit
[SwitchA] tunnel global source-address 1.1.1.1
```

**# Specify site-facing interface Ten-GigabitEthernet 1/0/1 as a VTEP access port.**

```
[SwitchA] interface ten-gigabitethernet 1/0/1
[SwitchA-Ten-GigabitEthernet1/0/1] vtep access port
[SwitchA-Ten-GigabitEthernet1/0/1] quit
```

#### **4. Configure Switch B:**

**# Enable L2VPN.**

```
<SwitchB> system-view
[SwitchB] l2vpn enable
```

**# Configure active TCP connection settings.**

```
[SwitchB] ovssdb server tcp 10.0.2.15 port 6632
```

**# Enable the OVSSDB server.**

```
[SwitchB] ovssdb server enable
```

**# Enable the OVSSDB VTEP service.**

```
[SwitchB] vtep enable
```

**# Assign an IP address to Loopback 0. Specify the IP address as the global source address for VXLAN tunnels.**

```
[SwitchB] interface loopback 0
[SwitchB-LoopBack0] ip address 2.2.2.2 255.255.255.255
[SwitchB-LoopBack0] quit
[SwitchB] tunnel global source-address 2.2.2.2
```

**# Specify site-facing interface Ten-GigabitEthernet 1/0/1 as a VTEP access port.**

```
[SwitchB] interface ten-gigabitethernet 1/0/1
[SwitchB-Ten-GigabitEthernet1/0/1] vtep access port
[SwitchB-Ten-GigabitEthernet1/0/1] quit
```

#### **5. Configure Switch C:**

**# Enable L2VPN.**

```
<SwitchC> system-view
[SwitchC] l2vpn enable
```

**# Configure active TCP connection settings.**

```
[SwitchC] ovssdb server tcp ip 10.0.2.15 port 6632
```

**# Enable the OVSSDB server.**

```
[SwitchC] ovssdb server enable
```

**# Enable the OVSSDB VTEP service.**

```
[SwitchC] vtep enable
```

**# Assign an IP address to Loopback 0. Specify the IP address as the global source address for VXLAN tunnels.**

```
[SwitchC] interface loopback 0
[SwitchC-LoopBack0] ip address 3.3.3.3 255.255.255.255
[SwitchC-LoopBack0] quit
[SwitchC] tunnel global source-address 3.3.3.3
```

**# Specify site-facing interface Ten-GigabitEthernet 1/0/1 as a VTEP access port.**

```
[SwitchC] interface ten-gigabitethernet 1/0/1
[SwitchC-Ten-GigabitEthernet1/0/1] vtep access port
[SwitchC-Ten-GigabitEthernet1/0/1] quit
```

## 6. Configure VXLAN settings on the controller. (Details not shown.)

### Verifying the configuration

1. Verify the VXLAN settings on the VTEPs. This example uses Switch A.

# Verify that the VXLAN tunnel interfaces on the VTEP are up.

```
[SwitchA] display interface tunnel 1
Tunnell
Current state: UP
Line protocol state: UP
Description: Tunnell Interface
Bandwidth: 64 kbps
Maximum transmission unit: 1464
Internet protocol processing: Disabled
Last clearing of counters: Never
Tunnel source 1.1.1.1, destination 2.2.2.2
Tunnel protocol/transport UDP_VXLAN/IP
Last 300 seconds input rate: 0 bytes/sec, 0 bits/sec, 0 packets/sec
Last 300 seconds output rate: 0 bytes/sec, 0 bits/sec, 0 packets/sec
Input: 0 packets, 0 bytes, 0 drops
Output: 0 packets, 0 bytes, 0 drops
```

# Verify that the VXLAN tunnels have been assigned to the VXLAN.

```
[SwitchA] display l2vpn vsi verbose
VSI Name: evpn2014
  VSI Index          : 0
  VSI State          : Up
  MTU                : 1500
  Bandwidth          : -
  Broadcast Restrain : -
  Multicast Restrain : -
  Unknown Unicast Restrain: -
  MAC Learning       : Enabled
  MAC Table Limit    : -
  MAC Learning rate  : -
  Drop Unknown       : -
  Flooding           : Enabled
  Statistics         : Disabled
  VXLAN ID           : 10
  Tunnels:
    Tunnel Name      Link ID   State  Type      Flood proxy
    -----
    Tunnell          0x5000001 Up      Manual    Disabled
    Tunnel2          0x5000002 Up      Manual    Disabled
  ACs:
    AC                Link ID   State  Type
    -----
    XGE1/0/1 srv2    0         Up     Manual
```

# Verify that the VTEP has learned the MAC addresses of remote VMs.

```
<SwitchA> display l2vpn mac-address
MAC Address      State      VSI Name      Link ID/Name  Aging
-----
cc3e-5f9c-6cdb   Dynamic    evpn2014      Tunnell       Aging
cc3e-5f9c-23dc   Dynamic    evpn2014      Tunnel2       Aging
```

--- 2 mac address(es) found ---

2. Verify that VM 1, VM 2, and VM 3 can ping each other. (Details not shown.)