Installing and Configuring the VXLAN Gateway

This chapter contains the following sections:

Information About the VXLAN Gateway Deployment

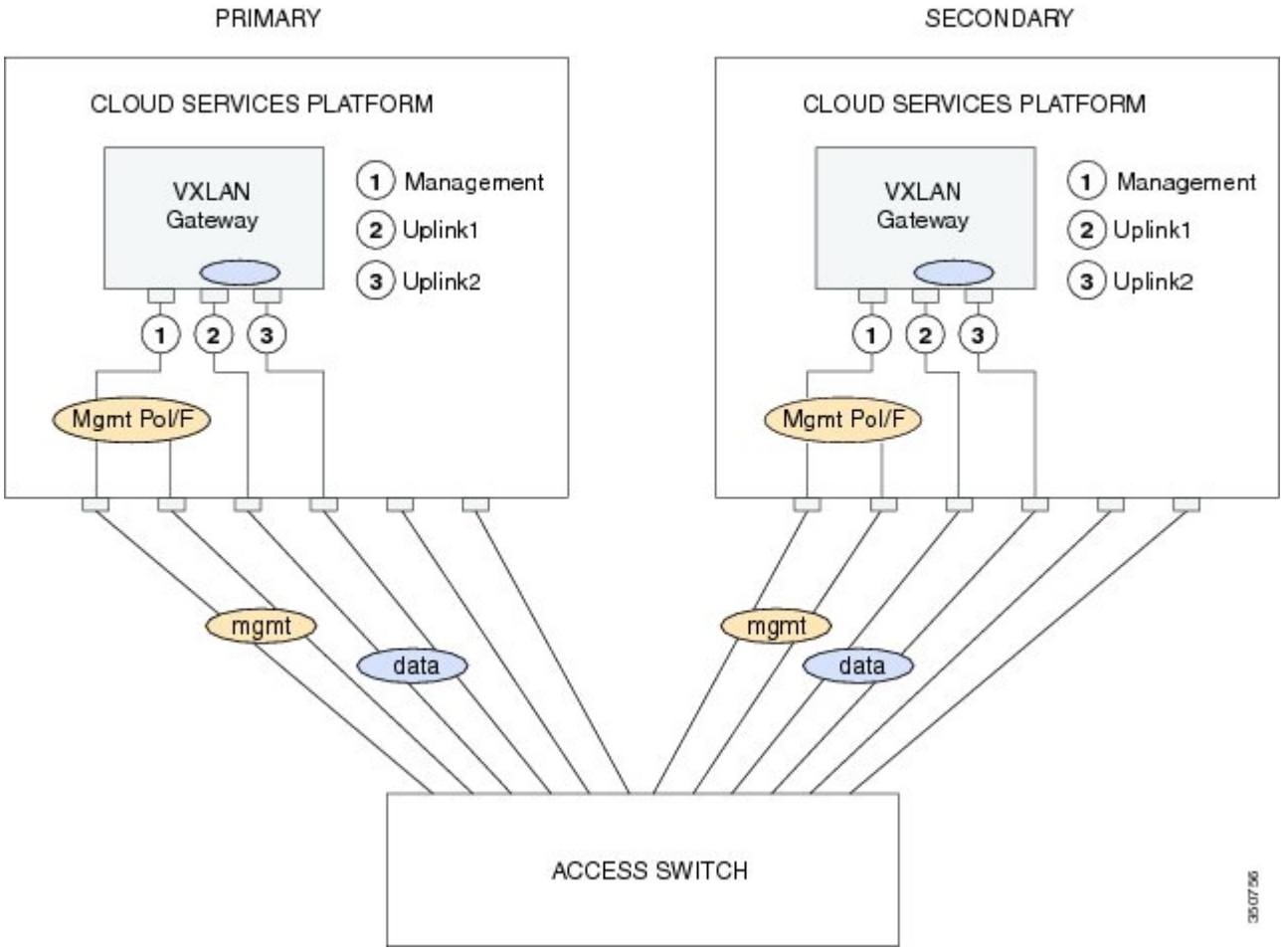
The Virtual Extensible Local Area Network (VXLAN) Gateway has the following deployment requirements:

- To configure the VXLAN Gateway, you must install the Advanced Edition license on the Cisco Nexus 1000V switch.
- You can only deploy the VXLAN Gateway as a VM. In this release of the Cisco Nexus 1000V for KVM, the VXLAN Gateway cannot be deployed on a Cisco Nexus Cloud Services Platform (Cisco Nexus 1010 or Cisco Nexus 1110 platforms).
- vCPU and memory requirements:
  - Two vCPUs for each Virtual Machine (VM)
  - 4-GB RAM
  - 10-GB virutal disk space
- Network requirements:
  - Management interface

- Gateway uplink interface
- VTEP interface

This figure shows the VXLAN Gateway deployment.

Figure 1: VXLAN Gateway Deployment on the Cloud Services Platform (CSP)



Guidelines and Limitations for Cisco Nexus 1000V VXLAN Gateway

VXLAN Gateways have the following configuration guidelines and limitations:

- You must configure the VSM to use the Layer 3 control. We strongly recommend that VSM Layer 3 control is through mgmt 0. For more information about Layer 3 control, see the *Cisco Nexus 1000V System Management Configuration Guide*.
- A single VSM can manage a maximum of eight VXLAN Gateway high availability (HA) clusters.

- You must configure the HA mode of the VXLAN Gateway as standalone or primary/secondary so that when you bring up the VXLAN Gateway, the HA state is either active or standby and the VXLAN-to-VLAN mappings are either active or pending. If you do not configure an HA role for the VXLAN Gateway, when you bring up the VXLAN Gateway, the HA state is unknown, and VXLAN-to-VLAN traffic is not processed.
- The VXLAN Gateway virtual machine (VM) uses two vCPUs; one vCPU for management traffic and one vCPU for the data interface.

Steps to Install and Configure VXLAN Gateway

There are several key steps to install and configure a VXLAN Gateway.

Procedure

	Command or Action	Purpose
Step 1	Create two port profiles on the switch (VSM): one for the uplinks on the gateway and one for the VTEP interface.	See Configuring a Port Profile for the Uplink on the VXLAN Gateway, on page 4 and Configuring a Port Profile for the VTEP on the VXLAN Gateway, on page 6.
Step 2	Deploy the VXLAN Gateway.	To deploy the VXLAN Gateway as a VM, see Installing the VXLAN Gateway as a VM Using Juju Charms, on page 7.
Step 3	Create the VXLAN Gateway data and management interfaces.	Choose one of the following methods: • Configuring the VXLAN Gateway Data and Management Interfaces Using OpenStack dashboard, on page 8 • Configuring the VXLAN Gateway Data and Management Interfaces Using the OpenStack CLI, on page 13
Step 4	Configure high availability on the VSMs. You must configure the HA mode of the VXLAN Gateway as standalone or primary/secondary so that when you bring up the VXLAN Gateway, the HA state is either active or standby and the VXLAN-to-VLAN mappings are either active or pending. If you do not configure an HA role for the VXLAN Gateway, when you bring up the VXLAN Gateway, the HA state is unknown, and VXLAN-to-VLAN traffic is not processed.	See Configuring the VXLAN Gateway HA Mode as Standalone, on page 16.
Step 5	Set up the VXLAN-to-VLAN mappings on the VXLAN Gateway.	See Managing the VXLAN-to-VLAN Mappings on the VXLAN Gateway, on page 17.

	Command or Action	Purpose
Step 6	Verify the VXLAN Gateway installation and configuration.	(Optional) See Verifying the VXLAN Gateway Configuration, on page 19.

Configuring Port Profiles on the VSM

You must create port profiles on the VSM before you can install and configure the VXLAN Gateway.

Configuring a Port Profile for the Uplink on the VXLAN Gateway

Before installing the VXLAN Gateway, you must create two port profiles on the switch (VSM), one for the uplinks on the gateway and one for the VXLAN Tunnel Endpoint (VTEP) interface.

Before You Begin

- Ensure that the VSM is configured in the Advanced mode by entering the **svs switch edition advanced** configuration command to enable Advanced mode.
- Ensure that the VXLAN feature is enabled on the VSM by entering the **feature segmentation** configuration command to enable VXLANs on the VSM.
- Ensure that the VXLAN Gateway is enabled on the VSM by entering the **feature vxlan-gateway** configuration command.
- Ensure that the interfaces of the upstream switch are configured with a matching port channel and VLAN trunk configuration.

Procedure

	Command or Action	Purpose
Step 1	vsm# **configure terminal**	Enters global configuration mode.
Step 2	vsm# **encapsulation profile segment** *name*	Creates an encapsulation profile to contain the VLAN-to-VXLAN mappings.
Step 3	vsm(config-vxlan-encap-prof)# **dot1q** *VLAN-ID* **bridge-domain** *bd-name*	Maps a VLAN to a VXLAN. The VXLAN is specified through the bridge-domain name. **Note** The bridge-domain name and VLAN-ID you provide are not created during the port-profile configuration. The bridge-domain name and the VLAN ID that you provide should be in an active state or the mapping is held in an inactive state until you create the bridge-domain name and VLAN ID.

	Command or Action	Purpose
		Note Repeat this step to specify additional mappings.
Step 4	vsm(config)# **port profile type ethernet** *name*	Creates a port profile of type ethernet for the VXLAN Gateway uplink. **Note** You must provide a port-profile name when prompted while executing the setup script to configure the VXLAN Gateway.
Step 5	vsm(config-port-prof)# **switchport mode trunk**	Designates that the interfaces are to be used as trunking ports. A trunk port transmits untagged packets for the native VLAN and transmits encapsulated, tagged packets for all other VLANs.
Step 6	vsm(config-port-prof)# **switchport trunk allowed vlan** *vlan list*	Specifies the list of VLANs allowed on the gateways uplink. This list should consist of all the mapped VLANs and the VLAN for the VTEP virtual interface.
Step 7	vsm(config-port-prof)# **mtu** *mtu size in bytes*	Designates the MTU size. For VXLAN traffic to be functional, you must set the MTU size as 1550. If you do not set the MTU size, the default of 1500 is used. The size must be an even number between 1500 and 9000. The MTU configured on an interface takes precedence over the MTU configured on a port profile.
Step 8	vsm(config-port-prof)# **service instance** *1-4096*	(Optional) Defines a place holder for mappings. The range is from 1 to 4096. **Note** You do not need to execute the service instance and the encapsulation command at this stage to bring up the gateway. These commands are optional and you can add the mappings later once the port profiles are configured.
Step 9	vsm(config-port-prof-svc)#**encapsulation profile** *name*	Specifies the encapsulation profile for the port profile.
Step 10	vsm(config-port-prof-srv)# **exit**	(Optional) Exits from the service instance mode.
Step 11	vsm(config-port-prof)# **no shutdown**	Administratively enables all ports in the profile.
Step 12	vsm(config-port-prof)# **state enabled**	Enables the port profile and applies its configuration to the assigned ports.
Step 13	vsm(config-port-prof)# **publish port-profile**	Publishes the port profile.

This example shows how to configure and display the gateway mappings:

```
vsm# configuration terminal
vsm(config)# port-profile type ethernet gw-uplink
vsm(config)# switchport mode trunk
vsm(config)# switchport trunk allowed vlan 1545
vsm(config)# mtu 1550
vsm(config-port-prof)# service instance 1
vsm(config-port-prof)# encapsulation profile gw_mappings
vsm(config-port-prof-srv)# encapsulation profile segment gw-segment
vsm(config-port-prof-srv)# encapsulation dot1q 753 bridge-domain bd-753
vsm(config-port-prof-srv)# exit
vsm(config-port-prof)# channel-group auto
vsm(config-port-prof)# no shutdown
vsm(config-port-prof)# state enabled
vsm(config-port-prof)# publish port-profile
```

Configuring a Port Profile for the VTEP on the VXLAN Gateway

You can create a port profile that can be applied to the VTEP virtual interface on the VXLAN Gateway.

Procedure

	Command or Action	Purpose
Step 1	vsm# **configure terminal**	Enters global configuration mode.
Step 2	vsm(config) # **port-profile type vethernet** *port-profile name*	Configures a port profile for the VTEP on the VXLAN gateway. **Note** You must provide a port profile name when prompted while executing the setup script to configure the VXLAN Gateway.
Step 3	vsm(config-port-prof) # **switchport mode access**	Designates that the interfaces are to be used as a trunking ports. A trunk port transmits untagged packets for the native VLAN and transmits encapsulated, tagged packets for all other VLANs.
Step 4	vsm(config-port-prof) # **switchport access vlan** *vlan-id-access*	Assigns an access VLAN ID to this port profile. The VLAN ID provided must be added to the allowed VLAN set of the uplink port profile. This VLAN should not be mapped to any VXLAN. **Note** If you do not specify a VLAN ID, VLAN 1 is used automatically.
Step 5	vsm(config-port-prof) # **capability vxlan**	Configures the capability VXLAN feature on the specified virtual ethernet port and enables encapsulation and decapsulation of VXLAN packets.

	Command or Action	Purpose
Step 6	vsm(config-port-prof) # **transport ip address** *ip-address* **netmask** *network mask* [**gateway** *ip-address*]	Configures the IP address, netmask, and gateway for the VTEP. **Note** If you have VTEPs that are in different subnets, you must specify the gateway IP address. If a gateway is not provided, the VXLAN Gateway uses ARP to reach the remote VTEP.
Step 7	vsm(config-port-prof)# **no shutdown**	Administratively enables all ports in the profile.
Step 8	vsm(config-port-prof)# **state enabled**	Enables the port profile and applies its configuration to the assigned ports.
Step 9	vsm(config-port-prof)# **publish port-profile**	Publishes the port profile.

This example displays how to configure a VTEP on the VXLAN gateway:

```
vsm# configure terminal
vsm(config)# port-profile type vethernet gw-vtep
vsm(config-port-prof)# switchport mode access
vsm(config-port-prof)# switchport access vlan 760
vsm(config-port-prof)# capability vxlan
vsm(config-port-prof)# transport ip address 192.0.2.1 255.255.255.0 gateway 192.0.2.254
vsm(config-port-prof)# no shutdown
vsm(config-port-prof)# state enabled
vsm(config-port-prof)# publish port-profile
```

Installing VXLAN Gateway

Installing the VXLAN Gateway as a VM Using Juju Charms

Before You Begin

- Ensure that the server on which you create the VXLAN Gateway VM meets the minimum requirements. See Networking Requirements and Networking Requirements.
- The source of the VXLAN Gateway image is configured in your VXLAN Gateway charm mapping.file. For information, see Cisco Nexus 1000V for KVM VXLAN Gateway Charm Parameters.
- You have created two port profiles on the switch (VSM): one for the uplinks on the gateway and one for the VTEP interface. For details, see Configuring a Port Profile for the Uplink on the VXLAN Gateway, on page 4 and Configuring a Port Profile for the VTEP on the VXLAN Gateway, on page 6.
- If you are not using DHCP to configure IP addresses on the VTEPS, you must have configured static IP addresses on the VTEPs. To do this, you must have configured the vtep_config parameter in a custom mapping file for the VEM charm. For information, see Cisco Nexus 1000V for KVM VEM Charm Parameters.

Ensure that the Nova cloud controller is started before deploying the VXLAN Gateway charm. To start the Nova cloud controller, use the **juju status nova-cloud-controller** command.

Procedure

Step 1 Configure the source of the VXLAN Gateway image in your VXLAN Gateway charm mapping file. For more information, see Cisco Nexus 1000V for KVM VXLAN Gateway Charm Parameters.

Step 2 Download the VXLAN Gateway software image *VXGW-image*.qcow.

Step 3 Deploy the VXLAN Gateway.
juju deploy [**config=***config-file*] [**--to=***destination*]
--repository=/opt/cisco/n1kv/charms/jujucharm-n1k/charms local:precise/vxlan-gateway

Step 4 Make sure the VXLAN Gateway image is present by entering the following OpenStack command:
glance image-list

Configuring the VXLAN Gateway Data and Management Interfaces

Configuring the VXLAN Gateway Data and Management Interfaces Using OpenStack dashboard

Guidelines and Limitations for the OpenStack Dashboard

The OpenStack dashboard has the following guidelines and limitations when you use it to create virtual networks for Cisco Nexus 1000V for KVM:

- Network profile creation by an administrator is not supported in Cisco Nexus 1000V for KVM Release 5.2(1)SK3(2.2) software and highter. Network profiles are automatically created for each network type.
- To create a network profile and associate it with a tenant, you must log in to the OpenStack dashboard as a user with admin privileges. Any user can use a network profile that is associated with a tenant.
- You cannot create policy profiles or assign them to a tenant in OpenStack dashboard. You must first create them as part of the port profiles in the VSM. The OpenStack dashboard retrieves them from the VSM and displays them on the **Router** dashboard.
- When there are multiple VSMs, the port profile must be configured on all the VSMs.

Steps to Configure the VXLAN Gateway Interfaces Using the OpenStack Dashboard

Before You Begin

Create one or more port profiles in the VSM. These port profiles are displayed as policy profiles in OpenStack dashboard. For more information, see Configuring a Port Profile for the Uplink on the VXLAN Gateway, on page 4 and Configuring a Port Profile for the VTEP on the VXLAN Gateway, on page 6.

Procedure

Step 1 Create a network profile of type trunk.
See Creating a Trunk Network Profile Using OpenStack Dashboard, on page 9.

Step 2 Create a network profile of type VLAN.
See Creating a VLAN Network Profile Using OpenStack Dashboard, on page 10.

Step 3 Create one or more networks for the network profiles.
See Creating a Network Using the OpenStack Dashboard, on page 11.

Step 4 Create a subnet for each network.
See Creating a Subnet for a Network Using the OpenStack Dashboard, on page 11.

You do not need to create a port for the network. OpenStack dashboard creates a port for the network when you launch the instance.

Step 5 Create and launch the VM instance.
See Booting a VXLAN Gateway VM, on page 12.

Creating a Trunk Network Profile Using OpenStack Dashboard

Before You Begin

- Create one or more policy profiles as part of the port profiles in the VSM.
- Create one or more tenants in OpenStack dashboard.

Procedure

Step 1 In the **Navigation** pane, click the **Router** dashboard.

Step 2 In the **Cisco Nexus 1000v** panel, click **Create Network Profile**.

Step 3 In the **Create Network Profile** dialog box, do the following:

a) In the **Name** field, enter a unique name for the network profile.
The name can have a maximum length of 255 characters and can contain uppercase or lowercase characters, numerals, and special characters such as an "at" sign (@), ampersand (&), and exclamation point (!).

b) From the **Segment Type** drop-down list, choose **Trunk**.

c) From the **Sub Type** drop-down list, choose **VLAN**.

d) From the **Project** multi-select list, choose one or more tenants that you want to associate with this network profile.
e) Click **Create Network Profile**.

OpenStack dashboard creates the network profile and then updates the OpenStack Neutron database and the VSM.

What to Do Next

Create one or more networks.

Creating a VLAN Network Profile Using OpenStack Dashboard

Before You Begin

- Create one or more policy profiles as part of the port profiles in the VSM.
- Create one or more tenants in OpenStack dashboard.

Procedure

Step 1 In the **Navigation** pane, click the **Router** dashboard.

Step 2 In the **Cisco Nexus 1000v** panel, click **Create Network Profile**.

Step 3 In the **Create Network Profile** dialog box, do the following:

a) In the **Name** field, enter a unique name for the network profile.
The name can have a maximum length of 255 characters and can contain uppercase or lowercase characters, numerals, and special characters such as an "at" sign (@), ampersand (&), and exclamation point (!).

b) From the **Segment Type** drop-down list, choose **VLAN**.
Separate the first and last segments in the range with a hyphen (-). For example, enter a range of 80-86. For a segment type of **VLAN**, the range can be from 1 to 3967 or from 4048 to 4093.

c) In the **Segment Range** field, enter the segment range for the network profile.
For example, enter a range of 80-86. For a segment type of **VLAN**, the range can be from 1 to 3967 or from 4048 to 4093.

d) In the **Physical Network** field, enter the name of the associated physical network.
e) From the **Project** multi-select list, choose one or more tenants that you want to associate with this network profile.
f) Click **Create Network Profile**.

OpenStack dashboard creates the network profile and then updates the OpenStack Neutron database and the VSM.

What to Do Next

Create one or more networks.

Creating a Network Using the OpenStack Dashboard

You must perform this procedure twice: once to create a network with the trunk network profile and again to create a network with the VLAN network profile.

Procedure

Step 1 In the **Navigation** pane, click **Admin** > **Networks**.

Step 2 In the **Networks** panel, click **Create Network**.

Step 3 In the **Create Network** dialog box, do the following:

a) In the **Name** field, enter a unique name for the network.
The name can have a maximum length of 255 characters and can contain uppercase or lowercase characters, numerals, and special characters such as an "at" sign (@), ampersand (&), and exclamation point (!).

b) From the **Project** drop-down list, choose a tenant that you want to associate with this network.
This tenant should be the same tenant that you use for the network profile.

c) Check the **Admin State** check box.
You can accept the defaults for the **Shared** check box and the **External Network** check box. By default, these check boxes are unchecked.

d) Click **Create Network**.

Step 4 For the VXLAN Gateway, perform this procedure again to create another network and choose the VLAN network profile that you already created.

What to Do Next

Create a subnet for the network.

Creating a Subnet for a Network Using the OpenStack Dashboard

You must perform this procedure twice: once to create a subnet in the trunk network that you already created and again to create a subnet in the VLAN network that you already created.

Note Do not enable DHCP on these subnets.

Procedure

Step 1 In the **Navigation** pane, click **Admin** > **Networks**.

Step 2 In the **Networks** panel, click the network to which you want to add a subnet.
For the VXLAN Gateway, choose the trunk network that you already created.

Step 3 In the **Create Subnet** dialog box, click the **Subnet** tab an do the following:

a) In the **Name** field, enter a unique name for the subnet.

The name can have a maximum length of 255 characters and can contain uppercase or lowercase characters, numerals, and special characters such as an "at" sign (@), ampersand (&), and exclamation point (!).

b) In the **Network Address** field, enter the address for the subnet.
The subnet address must be in classless interdomain routing (CIDR) format. For example, 192.168.0.0/16.

c) From the **IP Version** drop-down list, choose IPv4.

d) (Optional) In the **Gateway IP** field, enter a gateway IP address for the subnet.

Step 4 Optionally, click the **Subnet Detail** tab and do the following:

a) (Optional) Click the **Enable DHCP** checkbox.

b) Enter one or more allocation pools in the **Allocation Pools** text box.

c) Enter one or more name servers in the **DNS Name Servers** text box.

d) Enter one or more host routes in the **Host Routes** text box.

Step 5 Click **Create** to create the subnet.

Step 6 For the VXLAN Gateway, perform this procedure again to create another network and choose the VLAN network that you already created.

Booting a VXLAN Gateway VM

You must manually create the port IDs and launch the VM instance using OpenStack commands.

Procedure

Step 1 **neutron port-create** *trunk-network-name* **--n1kv:profile_id** *VSM-policy-profile-id*
Creates the trunk port ID for the VXLAN Gateway data interface.

Step 2 **neutron port-create** *vxgw-mgmt-network-name* **--n1kv:profile_id** *VSM-policy-profile-id*
Creates the management port ID for the VXLAN Gateway management interface.

Step 3 **nova boot --image** *image_id* **--flavor 3 nic port-id=** *trunk_port-id* **nic port-id=** *access_port-id vxlan_gateway_name*
Launches the VXLAN Gateway VM.

Note In this command, you must specify the first NIC as a trunk port and the second NIC as an access port.

Note You must specify flavor 3, because flavor 3 provides the two vCPUs that the VM requires.

Step 4 **nova list**
Lists the active servers.

Configuring the VXLAN Gateway Data and Management Interfaces Using the OpenStack CLI

Steps to Configure the VXLAN Gateway Interfaces Using the OpenStack CLI

Before You Begin

Create one or more port profiles in the VSM. For more information, see Configuring a Port Profile for the Uplink on the VXLAN Gateway, on page 4 and Configuring a Port Profile for the VTEP on the VXLAN Gateway, on page 6.

Procedure

	Command or Action	Purpose
Step 1	If you have not already done so, log in to the OpenStack CLI as a user with admin privileges.	—
Step 2	Create the VXLAN Gateway data interface.	For details, see Creating the VXLAN Gateway Data Interface, on page 13.
Step 3	Create theVXLAN Gateway management interface.	For details, see Creating a VXLAN Gateway Management Interface, on page 14.
Step 4	Boot the VXLAN Gateway VM.	For details, see Booting a VXLAN Gateway VM, on page 12.

Creating the VXLAN Gateway Data Interface

Procedure

	Command or Action	Purpose
Step 1	**neutron net-create** *network_name*	Creates a network using the network profile.
Step 2	**neutron subnet-create** *network_name cidr_blocks* **--name** *subnet_name* **--disable-dhcp**	Creates a subnet for the network profile. **Note** Although you configure the VXLAN Gateway data interface manually, the **create-subnet** command still requires that you specify an IP address block. However, this value is not used and is ignored.
Step 3	**neutron cisco-policy-profile-list**	Lists the policy profiles available in neutron.
Step 4	**neutron port-create** *network_name* **n1kv:profile** *cisco_policy_profile_id*	Creates a trunk port. For *network_name*, specify the name of the network that you created in Step 3. For

	Command or Action	Purpose
		cisco_policy_profile_id, specify the profile with the mode trunk from the list of available profiles obtained in Step 3. **Note** To create a second instance of the VXLAN Gateway virtual machine (for example, for an HA pair), perform this step again.

Creating a VXLAN Gateway Management Interface

Procedure

	Command or Action	Purpose
Step 1	**neutron net-create** *network_name*	Creates an access network.
Step 2	**neutron subnet-create** *network_name cidr_blocks* **--name** *subnet_name*	Creates a subnet for the access network segment profile.
Step 3	**neutron cisco-policy-profile-list**	Lists the policy profiles available in Neutron.
Step 4	**neutron port-create** *network_name* **n1kv:profile** *cisco_policy_profile_id*	Creates an access port. Use a Cisco policy profile ID from the list of available profiles listed in Step 5.

Booting a VXLAN Gateway VM

You must manually launch a VM instance using the OpenStack CLI.

Procedure

	Command or Action	Purpose
Step 1	Launch the VXLAN Gateway VM.	**nova boot --image** *image_id* **--flavor 3 --nic port-id=** *trunk_port-id* **--nic port-id=** *access_port-id vxlan_gateway_name* **Note** In this command, you must specify the first NIC as a trunk port and the second NIC as an access port. Also, you must specify flavor 3 (m1.medium), because flavor 3 provides the two vCPUs that the VM requires.
Step 2	List the active servers.	**nova list**

Example of VXLAN Gateway Interface Configuration

This example shows how to configure a VXLAN Gateway VM with a trunk network profile named aug-net and a VLAN network profile named test-2.

```
root@server:~# neutron cisco-credential-create 172.23.181.101 n1kv --tenant-id
1234-1234-1234-1234 --user_name admin --password mypwd
root@server:~# neutron cisco-network-profile-create aug-net-prof trunk sub_type vlan
root@server:~# neutron net-create aug-net --n1kv:profile_id
999e6dc-7a55-4c4a-839e-9aa22957ba99
root@server:~# neutron subnet-create aug-net 110.10.10.0/24 --name augsubnet --disable-dhcp
root@server:~# neutron cisco-policy-profile-list
 +--------------------------------------+---------------+
| id                                   | name          |
+--------------------------------------+---------------+
| 0dc6bb6d-2929-466f-abed-6e4c0f25c3be | test-prof     |
| 4c4455e5-95d6-4185-80a1-a028e27fc8bd | trk-policy    |
| 63964f53-178c-440a-8f09-301ce021498d | acc-policy    |
| 652fbfd7-0c40-41d0-bc34-5a0fda4b2ca7 | dhcp_pp       |
| 964fb14c-59cc-48d4-a4d4-abc52965fb7c | q-n1kv-policy |
+--------------------------------------+---------------+
root@server:~# neutron port-create aug-net --n1kv:profile_id
4c4455e5-95d6-4185-80a1-a028e27fc8bd
root@server:~# neutron cisco-network-profile-create test2 vlan --segment_range 400-499
--physical_network q-n1k
root@server:~# neutron net-create q-n1k --n1kv:profile_id 2e464910-f452-4865-9793-6bd70a4c0cf6
root@server:~# neutron subnet-create q-n1k 110.10.10.0/24 --name q-110.10.10.0/24
root@server:~# neutron cisco-policy-profile-list
+--------------------------------------+---------------------------+
| id                                   | name                      |
+--------------------------------------+---------------------------+
| 5cdd9b93-c109-4455-a740-746a9d59c340 | NSM_template_vlan         |
| 691b2497-b435-495a-b140-e3ebe0835e75 | test-2                    |
| 7fc8d34b-cadd-4ca0-a85c-e18aecedd308 | NSM_template_segmentation |
| 83cc0410-7781-42b5-abb1-3b00921bfcc5 | test-1                    |
+--------------------------------------+---------------------------+
root@server:~# neutron port-create q-n1k --n1kv:profile_id
99cc0410-7781-42b5-abb1-3b00921bfcc5
root@server:~# nova boot --image vxlan-gateway --flavor 3 --nic
port-id=d341926c-21ca-48cd-ae18-c51f899f6d3f
root@server:~#
```

Configuring High Availability

VXLAN Gateway and High Availability

The operation of high availability (HA) involves the following terminology:

- Cluster—A cluster is a pair of gateway modules that operate together as a single HA module. Each cluster is distinguished by a unique cluster ID. A gateway module that is deployed in a standalone mode of operation is assigned a dummy cluster ID of 0.
- HA role—The gateway modules that make up an HA cluster are assigned separate roles. One is designated as primary and the other as secondary. This role decides which of the two modules goes to active state first and which stays in a standby state. These states persist until the active module fails. In the event of any failure in the active gateway module, the standby gateway module detects the failure and moves to the active state. This way, one of the two modules is always providing active service.
- HA state— At any given time, only one gateway module from a given cluster is actively performing the gateway function. The other module stays in the standby state pending the failure of the active module.

A gateway module can be in the active or standby state. In addition, there is a transient initial state called the Init state. In this state, a gateway is either waiting to be assigned a role or negotiating its state with its peer.

After a gateway module is installed and brought up, the VSM assigns a role to the gateway module and can result in one of the following transitions:

- Unconfigured-Init to Standalone-Active
- Unconfigured-Init to Primary-Active
- Unconfigured-Init to Secondary-Standby
- Standalone-Active to Primary-Active
- Standalone-Active to Secondary-Active

For all other combinations, we recommend that you first fall back to the Unconfigured-Init mode by using the **no service** *VXLAN Gateway module* command and then proceed to the desired role or states.

Note Roles must be preassigned to module numbers in the VSM. When a VXLAN gateway is attached to the VSM on that module, it inherits the role and state that are assigned by the VSM.

You must configure the HA mode of the VXLAN Gateway as standalone or primary/secondary so that when you bring up the VXLAN Gateway, the HA state is either active or standby and the VXLAN-to-VLAN mappings are either active or pending. If you do not configure an HA role for the VXLAN Gateway, when you bring up the VXLAN Gateway, the HA state is unknown, and VXLAN-to-VLAN traffic is not processed.

Configuring the VXLAN Gateway HA Mode as Standalone

You can create a service module in a standalone mode. Perform these steps on the VSM.

Before You Begin

Roles must be preassigned to module numbers in the VSM. When a VXLAN gateway is attached to the VSM on that module, it inherits the role and state that are assigned by the VSM.

Procedure

	Command or Action	Purpose
Step 1	vsm(config)# **service** *mod* **role standalone**	Configures the service module as standalone active.
Step 2	vsm(config)# **show module service**	Displays the service module number, cluster ID, role, HA mode, and status.

This example shows how to display the cluster ID mapping and the details about active, standby, and standalone service modules:

```
vsm(config)# show module service
Mod  Cluster-id  Role            HA Mode       Status
---  ----------  ------------    ----------    -------
36        0      Standalone      Standalone    Active
```

Configuring the VXLAN Gateway as an HA Pair

You can create a service module as a HA pair. Perform these steps on the VSM.

Before You Begin

You must create a second instance of the VXLAN Gateway VM.

Procedure

	Command or Action	Purpose
Step 1	vsm(config)# **service** *modNo1* **role** *primary* **ha-cluster** *clusterNo*	Configures the service module in HA and adds a primary service module to a cluster.
Step 2	vsm(config)# **service** modNo2 **role** *secondary* **ha-cluster** *clusterNo*	Configures another service module as secondary in the same cluster.
Step 3	vsm(config)# **show module service-module**	Displays the service module number, cluster ID, role, HA mode, and status.

This example shows how to display the cluster ID mapping and the details about active, standby, and standalone service modules:

```
vsm(config)# show module service
Mod  Cluster-id  Role        HA Mode    Status
--- ---------- ----------- ---------- -------
9    1        Primary          HA       Active
10   1        Secondary        HA       Standby
```

To switch over between the active and standby VXLAN gateway, enter the following command on the VSM:

```
vsm# service ha-cluster <1-8> switchover
```

Managing the VXLAN-to-VLAN Mappings on the VXLAN Gateway

The VLAN-to-VXLAN mappings that are configured on a gateway module can be managed by editing the port profile applied on the gateway uplink modules. To add or remove a mapping, perform these steps on the VSM.

Procedure

	Command or Action	Purpose
Step 1	vsm# **configure terminal**	Enters global configuration mode.
Step 2	vsm(config)# **encapsulation profile segment** *name*	Creates an encapsulation profile to contain the VLAN-to-VXLAN mappings.
Step 3	vsm(config-vxlan-encap-prof)# **dot1q** *VLAN-ID* **bridge-domain** *bd-name*	Maps a VLAN to a VXLAN. The VXLAN is specified through the bridge-domain name. **Note** The bridge-domain name and VLAN ID that you provide are not created during the port-profile configuration. The bridge-domain name and the VLAN ID that you provide should be in an active state or the mapping is held in an inactive state until you create the bridge-domain name and VLAN ID. **Note** Repeat this step to specify additional mappings. **Note** To remove a mapping, use the **no** form of this command.
Step 4	vsm(config-vxlan-encap-prof)# **exit**	Exits the current configuration mode.
Step 5	vsm(config)# **port-profile** *port-profile-name*	Specifies the name of the port-profile applied to the VXLAN Gateway uplink interface.
Step 6	vsm(config-port-prof)# **service instance** *1 to 4096*	Defines a place holder for mappings. The range is from 1 to 4096. **Note** Port profiles that contain the **service instance** keyword cannot be used for a non-VXLAN gateway module.
Step 7	vsm(config-port-prof-srv)# **encapsulation profile** *name*	Assigns the specified encapsulation profile to the port profile.
Step 8	vsm(config-port-prof-srv)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

This example shows how to configure VXLAN-to-VLAN mappings on the VXLAN gateway:

```
vsm# configure terminal
vsm(config)# encapsulation profile segment mgmt_mappings
vsm(config-vxlan-encap-prof)# dot1q 1555 bridge-domain vxlan6000
vsm(config-vxlan-encap-prof)# dot1q 1557 bridge-domain vxlan6002
vsm(config-vxlan-encap-prof)# dot1q 1558 bridge-domain vxlan6003
vsm(config-vxlan-encap-prof)# dot1q 1559 bridge-domain vxlan6004
vsm(config-vxlan-encap-prof)# exit
vsm(config)# port-profile Uplink-All-VXGW
vsm(config-port-prof)# service instance 2
```

```
vsm(config-port-prof)# encapsulation profile mgmt_mappings
vsm(config-prot-prof-srv)# copy running-config startup-config
vsm(config)# show run port-profile Uplink-All-VXGW
port-profile type ethernet Uplink-All-VXGW
  switchport mode trunk
  switchport trunk allowed vlan 1545-1575,1577-1605
  mtu 1550
  service instance 2
    encapsulation dot1q 1555 bridge-domain vxlan6000
    encapsulation dot1q 1557 bridge-domain vxlan6002
    encapsulation dot1q 1558 bridge-domain vxlan6003
    encapsulation dot1q 1559 bridge-domain vxlan6004
   no shutdown
  state enabled
vsm(config)# show port-profile usage
port-profile Uplink-All-VXGW
 port-channel1
 port-channel5
 Ethernet7/1
 Ethernet7/3
vsm(config)# show run interface ethernet 7/1 expand-port-profile
interface Ethernet7/1
  switchport mode trunk
  switchport trunk allowed vlan 1545-1575,1577-1605
  mtu 1550
  channel-group auto mode active
  service instance 2
    no shutdown
    encapsulation dot1q 1557 bridge-domain vxlan6002
    encapsulation dot1q 1555 bridge-domain vxlan6000
    encapsulation dot1q 1558 bridge-domain vxlan6003
    no shutdown
```

Verifying the VXLAN Gateway Configuration

To display the VXLAN gateway (GW) installation and configuration information, perform one of the following tasks on the VSM:

Command	Purpose
show running-config port-profile gw-uplink	Displays the configuration of the port profile assigned to the VXLAN gateway uplinks.
show running-config port-profile gw-vtep	Displays the configuration of the port profile assigned to the VXLAN VTEP.
show module	Displays the VXLAN gateway service modules.
show module service	Verifies the role of the VXLAN gateway module and displays the cluster ID mapping and the details about active, standby, and standalone service modules.
show vxlan gateway interface	Displays if the VTEPs are configured properly.
show interface vethernet 6	Displays if both the VTEP Virtual Ethernet interfaces are in up state.

Command	Purpose
show port-channel summary	Displays if the port channels are up for gateway service modules.
show bridge-domain mappings	Displays VXLAN gateway mappings.
show switch edition	Displays if the VSM is in Advanced mode.
show feature	Displays if the VXLAN gateway is enabled on the VSM.
show virtual-service-blade summary **Note** You must enter this command from the Cloud Services Platform.	Displays the status of the VXLAN gateway VSB as it transitions from the VSB DEPLOY IN PROGRESS to VSB POWERED ON.
show virtual-service-blade **Note** You must enter this command from the Cloud Services Platform.	Displays the VXLAN Gateway configuration.
show encapsulation profile	Displays the VLAN-to-VXLAN mappings for all encapsulation profiles or for the specified encapsulation profile.

This example shows how to display the status of the VXLAN gateway VSB:

```
CSP# show virtual-service-blade summary

--------------------------------------------------------------------------------
Name                 HA-Role     HA-Status   Status                    Location
--------------------------------------------------------------------------------
VXLAN-GW             PRIMARY     ACTIVE      VSB POWERED ON            PRIMARY
VXLAN-GW             SECONDARY   ACTIVE      VSB POWERED ON            SECONDARY
```

This example shows how to display the VXLAN gateway configuration:

```
CSP# show virtual-service-blade
virtual-service-blade VXLAN-GW
  Description:
  Slot id:        1
  Host Name:      VXLAN-GW-DOCS
  Management IP:  192.168.1.104
  VSB Type Name :  vx-gw-1.5
  Configured vCPU:           3
  Operational vCPU:           3
  Configured Ramsize:        2048
  Operational Ramsize:        2048
  Disksize:       3
  Heartbeat:      154764

  Legends:   P -  Passthrough
  ------------------------------------------------------------------------------
   Interface           Type         MAC     VLAN    State   Uplink-Int
                                                   Pri  Sec Oper  Adm
  ------------------------------------------------------------------------------
  VsbEthernet1/1  gw-uplink1 0002.3d71.a303        up   up Gi3(P) Gi3(P)
  VsbEthernet1/2  management 0002.3d71.a302  751   up   up Gi1    Gi1
  VsbEthernet1/3  gw-uplink2 0002.3d71.a304        up   up Gi4(P) Gi4(P)
        internal          NA        NA       NA    up   up
  HA Role: Primary
```

```
    HA Status: ACTIVE
    Status:       VSB POWERED ON
    Location:     PRIMARY
    SW version:
  HA Role: Secondary
    HA Status: ACTIVE
    Status:       VSB POWERED ON
    Location:     SECONDARY
    SW version:
  VSB Info:
    Domain ID : 405
```

This example shows how to display the port-profile configuration assigned to the VXLAN Gateway uplinks:

```
vsm# show running-config port-profile gw-uplink

port-profile type ethernet gw-uplink
  switchport mode trunk
  switchport trunk allowed vlan 1,81,751-760
  mtu 1550
  channel-group auto mode active
  no shutdown
  state enabled
```

This example shows how to display the port-profile configuration assigned to the VXLAN VTEP:

```
vsm# show running-config port-profile gw-vtep

port-profile type vethernet gw-vtep
  switchport mode access
  switchport access vlan 760
  capability vxlan
  transport ip address 182.168.1.253 255.255.255.0 gateway 182.168.1.1
  no shutdown
  state enabled
```

This example shows how to display the VXLAN Gateway service modules as soon as they are online:

```
vsm# show module
Mod  Ports  Module-Type                      Model               Status
---  -----  --------------------------------  ------------------  ------------
1    0      Virtual Supervisor Module         Nexus1000V          ha-standby
2    0      Virtual Supervisor Module         Nexus1000V          active *
11   332    Virtual Ethernet Module           NA                  ok
18   332    Virtual Ethernet Module           NA                  ok
20   13     Virtual Service Module            VXLAN Gateway       ok
21   13     Virtual Service Module            VXLAN Gateway       ok
22   13     Virtual Service Module            VXLAN Gateway       ok

Mod  Sw                Hw
---  ----------------  ------------------------------------------------
1    5.2(1)SK1(2.1)    0.0
2    5.2(1)SK1(2.1)    0.0
11   5.2(1)SK1(2.1)    Linux 3.2.0-23-generic
18   5.2(1)SK1(2.1)    Linux 3.8.0-29-generic
20   5.2(1)SK1(2.1)    Linux 2.6.27.10
21   5.2(1)SK1(2.1)    Linux 2.6.27.10
22   5.2(1)SK1(2.1)    Linux 2.6.27.10

Mod  Server-IP        Server-UUID                           Server-Name
---  ---------------  ------------------------------------  --------------------
1    192.0.2.166      NA                                    NA
2    192.0.2.166      NA                                    NA
11   192.0.2.147      47DE0F4A-8072-0E4D-A218-84A2C058FF3C  vxlan-gw-ubuntu
18   192.0.2.54       EE04B285-6BA6-5642-AC8E-EDB12D07811B  vxlan-gw-ubuntu2
20   192.0.2.90       7a4f6171-3235-4fd2-9797-80149df55171  ab-vxgw-1110
21   192.0.2.53       48ad3898-401f-4377-8de1-dcfa22df78ec  VxLanGW
22   192.0.2.55       5a996a7e-fa85-4630-89e5-18a3a3b413b8  VxLanGW
* this terminal session
```

This example shows how to display the cluster ID mapping and the details about active, standby, and standalone service modules:

```
vsm# show module service
Mod  Cluster-id  Role        HA Mode    Status
--- ---------- ----------- ---------- -------
9    1        Primary          HA       Active
10   1        Secondary        HA       Standby
```

This example shows how to display the module for virtual Ethernet interface binding:

```
vsm(config-if)# show vxlan gateway interface
-----------------------------------------------------------------
Port  IPAddress  Netmask  Gateway Mod Status Role
-----------------------------------------------------------------
Veth6  192.0.2.253  255.255.255.0 192.168.1.1 9  up Active
Veth22  192.0.2.253  255.255.255.0 192.168.1.1 10  up Standby
```

This example shows how to display whether both the VTEP virtual Ethernet interfaces are in up state:

```
vsm# show interface vethernet 6
Vethernet6 is up
  Port description is VXLANGW VTEP, Network Adapter 1
  Hardware: Virtual, address: 0002.3d71.a303 (bia 0002.3d71.a303)
  Owner is VM "VXLANGW VTEP", adapter is Network Adapter 1
  Active on module 9
  Port-Profile is gw-vtep
  Port mode is access
  5 minute input rate 8 bits/second, 0 packets/second
  5 minute output rate 0 bits/second, 0 packets/second
  Rx
    6 Input Packets 6 Unicast Packets
    0 Multicast Packets 588 Broadcast Packets
    468 Bytes
  Tx
    34321 Output Packets 34321 Unicast Packets
    33609 Multicast Packets 24 Broadcast Packets 33633 Flood Packets
    2193700 Bytes
    0 Input Packet Drops 0 Output Packet Drops

vsm# show interface vethernet 22
Vethernet22 is up
  Port description is VXLANGW VTEP, Network Adapter 1
  Hardware: Virtual, address: 0002.3d71.a383 (bia 0002.3d71.a383)
  Owner is VM "VXLANGW VTEP", adapter is Network Adapter 1
  Active on module 10
  Port-Profile is gw-vtep
  Port mode is access
  5 minute input rate 8 bits/second, 0 packets/second
  5 minute output rate 0 bits/second, 0 packets/second
  Rx
    6 Input Packets 6 Unicast Packets
    0 Multicast Packets 25 Broadcast Packets
    468 Bytes
  Tx
    33742 Output Packets 33742 Unicast Packets
    33609 Multicast Packets 133 Broadcast Packets 33742 Flood Packets
    2158956 Bytes
    0 Input Packet Drops 0 Output Packet Drops
```

This example shows how to display whether the port channels are up for VXLAN Gateway service modules:

```
vsm# show port-channel summary
Flags:  D - Down        P - Up in port-channel (members)
        I - Individual  H - Hot-standby (LACP only)
        s - Suspended   r - Module-removed
        S - Switched    R - Routed
        U - Up (port-channel)
--------------------------------------------------------------------------------
Group Port-       Type     Protocol  Member Ports
      Channel
--------------------------------------------------------------------------------
```

```
1     Po1(SU)     Eth      NONE      Eth3/3(P)    Eth3/4(P)    Eth3/5(P)
                                     Eth3/6(P)
2     Po2(SU)     Eth      NONE      Eth4/3(P)    Eth4/4(P)    Eth4/5(P)
                                     Eth4/6(P)
3     Po3(SU)     Eth      NONE      Eth5/3(P)    Eth5/4(P)    Eth5/5(P)
                                     Eth5/6(P)
4     Po4(SU)     Eth      NONE      Eth6/3(P)    Eth6/4(P)    Eth6/5(P)
                                     Eth6/6(P)
5     Po5(SU)     Eth      NONE      Eth7/3(P)    Eth7/4(P)    Eth7/5(P)
                                     Eth7/6(P)
6     Po6(SU)     Eth      NONE      Eth8/4(P)
7     Po7(SU)     Eth      LACP      Eth9/1(P)    Eth9/3(P)
8     Po8(SU)     Eth      LACP      Eth10/1(P)   Eth10/3(P)
```

This example shows how to display VXLAN Gateway mappings:

```
vsm# show bridge-domain mappings
--------------------------------------------------------------------------------
Interface       Module  Serv Inst  Vlan  BD-Name
--------------------------------------------------------------------------------
port-channel7   9       753        753   bd-753
port-channel8   10      753        753   bd-753
```

This example shows how to display the IP address for module binding:

```
vsm(config-if)# show module service mgmt-int
 --------------------------------------------------------------------------------
Mod Interface-Name IP-address Speed MTU
--------------------------------------------------------------------------------
4 Mgmt0 10.10.10.2             0     0
5 Mgmt0 10.10.10.3             0     0
Remember the management IP address user installs gateway with
(in this example 10.10.10.2, which occupies module slot 4)
```

This example shows how to display whether the VSM is in Advanced mode:

```
vsm# show switch edition
Switch Edition: Advanced

Advanced Features
Feature Name    Feature State
-----------------------------
vxlan-gateway   enabled

Licenses Available: 1020
Licenses In Use: 4
License Expiry Date: 13 Jun 2013
```

This example shows how to display whether the VXLAN Gateway is enabled on the VSM:

```
vsm# show feature
Feature Name          Instance  State
--------------------  --------  --------
cts                   1         enabled
dhcp-snooping         1         enabled
http-server           1         enabled
lacp                  1         enabled
netflow               1         disabled
network-segmentation  1         enabled
port-profile-roles    1         disabled
private-vlan          1         disabled
segmentation          1         enabled
sshServer             1         enabled
tacacs                1         disabled
telnetServer          1         disabled
vtracker              1         enabled
vxlan-gateway         1         enabled
```

Perform one of the following tasks on the VXLAN Gateway. If your VSM is on Layer 3 through management and your gateway is also on the same management subnet, use the **attach module** *service module number* command to acces the gateway CLI. If your VSM is on Layer 3 through control, you can access the gateway CLI from any machine on that control subnet. This example shows the VSM which is on Layer 3 control.

Command	Purpose
show redundancy config	Displays the high availability status.

This example shows how to display the HA status:

```
gw# show redundancy config

HA Manager Node Information:

    Cluster Node Count: 2

    Local Node:
      state          : Active
      HA mode        : High Availability
      uuid           : 56fa6753-4dc5-4a7d-ad07-cc817114f838
      cluster_id     : 1
      node_priority  : 2
      node_type      : VXLAN Gateway
      ipaddr [mgmt]  : 192.168.1.104

    Peer Node 1:
      state          : Standby
      uuid           : 4cbd05df-b3e5-468a-9497-89aa3fae8153
      node_type      : VXLAN Gateway
      ipaddr [mgmt]  : 192.168.1.105
```

This example shows how to display the VLAN-to-VXLAN mappings for all encapsulation profiles:

```
gw# show encapsulation profile

--------------------------------------------------------------------------------
Vlan Bridge-domain
--------------------------------------------------------------------------------
2100 segment5050
2055 segment5031
2056 segment5032
2057 segment5033
2058 segment5034
```