Virtualization Design

• VMM Integration with UCS-B, on page 1
• VMM Integration with AVS or VDS, on page 3
• VMM Domain Resolution Immediacy, on page 6
• OpenStack and Cisco ACI, on page 8

VMM Integration with UCS-B

About VMM Integration with UCS-B

Virtual Machine Manager (VMM) integration allows for the Cisco Application Centric Infrastructure (ACI) fabric to extend network policy and policy group definitions into virtual switches residing on a hypervisor. This integration allows for automation of certain steps that typically create delays in the deployment of virtual and compute resources. The integration is done by allowing the ACI fabric to configure automatically the required fabric side and hypervisor virtual switch encapsulation to ensure matching definitions.

When it comes to ACI and UCS-B interaction, the specific design of the UCS-B has to be taken into account. That is, the fact that the two fabric interconnects are never part of a single logical switch, and the concept of a designated receiver and how the designated receiver is determined on these fabric interconnects. A leaf switch connected to a set of end hosts (compute resources) is commonly referred to as a "leaf node." This terminology will be used throughout this section.

Prerequisites for VMM Integration with UCS-B

Virtual Machine Manager (VMM) integration with UCS-B has the following prerequisites:

• The Virtual Machine Manager (VMM) must be deployed.

• The VMM must be reachable through out-of-band or in-band management from the Application Policy Infrastructure Controller (APIC).

• The VMM must have some hosts integrated into its domain.

• The UCS vNICs must be configured to use either CDP or LLDP. Both protocols cannot be configured, but one is required.

• The block of VLANs to be utilized must be created on UCS and applied only to the leaf node-facing uplinks and the integrated hosts vNICs.
Guidelines and Limitations for VMM Integration with UCS-B

For UCS-B integration, you must take into account the following limitations:

- The only supported OS load balancing mechanism for UCS-B is "Route Based on Originating Virtual Port ID." This equates to the vSwitch policy of "MAC-pinning" within Cisco Application Centric Infrastructure (ACI).

- If utilizing a disjointed Layer 2 domain on the UCS (essentially certain VLANs of certain interfaces), you must have performed proper VLAN pruning on the fabric interconnects. By default, the UCS allows configured VLANs on all interfaces. VLAN trunking is associated with the designated receiver (DR) within the UCS POD. Only one interface (port/port-channel) per VLAN is selected as the DR. There will be endpoint retention issues if the selected DR interface is not one that is connected to the ACI fabric.

- CDP or LLDP is required for most VMM integration deployment scenarios. ACI utilizes these neighbors to resolve virtual host IDs from the end hosts to the leaf nodes. If neighborship is not formed under these scenarios, the leaf node will not push policy to allow for a communication path into the ACI fabric.

CDP and LLDP are not required when integrating with Cisco AVS.

Recommended Configuration Procedure for VMM Integration with UCS-B

Although VMM integration aids in configuration by automating the VLAN assignment for both the endpoint group and the port group, there are certain configurations that must still be completed manually or there will be connectivity issues:

Procedure

---

**Step 1**

All intermediate devices should have the dynamic block range of VLANs allowed. In the case of UCS, this means that the user must still navigate the UCS Manager and allow the range of configured VLANs on all VNICs and uplink ports that are going to the ACI fabric.

**Example:**

The design asks to use VLANs 100-200 for VMM integration with UCS-B. The user must go into UCSM and perform the following tasks:

a) Create VLANs 100-200.
b) Allow the VLANs on the Uplink interfaces.
c) Prune the VLANs from undesired uplink interfaces.
d) Allow the VLANs on the vNICs of all hosts that will be integrated.

**Step 2**

In the APIC GUI, create a MAC-pinning port channel policy.

a) On the menu bar, choose Fabric > Access Policies.
b) In the Navigation pane, choose Interface Policies > Policies > Port Channel Policies.
c) In the Work pane, choose Actions > Create Port Channel Policy.
d) In the Create Port Channel Policy dialog box, fill out the fields as necessary

This policy must be associated to the attachable access entity profile as a vSwitch port channel policy to take effect. This only changes the vSwitch port channel policy, not the port channel policy that is associated with the physical interfaces that are utilized by the end hosts.
Step 3  Associate the port channel policy to the attachable access entity profile as a vSwitch port channel policy.
   a)  On the menu bar, choose Fabric > Access Policies.
   b)  In the Navigation pane, choose Global Policies > Attachable Access Entity Profiles > AAEP_name.
   c)  In the Work pane, choose Actions > Config vSwitch Policies.
   d)  In the Config vSwitch Policies dialog box, fill out the fields as necessary.

Verifying the VMM Integration with UCS-B Configuration

Procedure

Step 1  Verify the node neighbors by using SSH to connect to the leaf node and run either the show cdp neighbors or show lldp neighbors command, depending on what configuration is used within this deployment.
Step 2  Verify neighborship directly on the fabric interconnects to ensure that the hypervisor vNICs are forming a neighborship through CDP or LLDP.
Step 3  Verify compute node VLAN programming by using SSH to connect to the node and running the show VLAN extended command.

Additional References for VMM Integration with UCS-B

For additional information on VMM Integration, go to the following URL:

VMM Integration with AVS or VDS

About VMM Integration with AVS or VDS

The integration of Cisco ACI with virtualized servers using a VMware vSphere Distributed Switch (VDS) or Cisco Application Virtual Switch (AVS) provides more control of the virtual environment from the Application Policy Infrastructure Controller (APIC). The APIC aggregates the information from virtualized servers, allowing the administrator to see where virtual machines are located in the fabric, the locations where the virtualized hosts are attached, and more.

With VDS, certain levels of configuration get pushed from the APIC as opposed to manually configuring them directly on the VDS. The configuration pushed from the APIC includes port groups and various port group settings. The VDS on its own can only be deployed utilizing VLANs.

The AVS is derived from the Cisco Nexus 1000v Platform. Similar in control, the APIC pushes port groups, port group settings, and other features that can be utilized, including, but not limited to, the distributed firewall and microsegmentation.
Prerequisites for VMM Integration with AVS or VDS

This section lists the prerequisites for Virtual Machine Manager (VMM) integration with AVS or VDS:

• Make a decision on whether or not to use VLAN or VXLAN encapsulation or multicast groups.

• A virtual machine manager must be already deployed, such as vCenter.

• The VMM must be accessible by the Application Policy Infrastructure Controller (APIC) by either out-of-band or in-band management.

• For Cisco Application Virtual Switch (AVS) deployment, a vSphere Installation Bundle (VIB) must be installed on all Hypervisor hosts to be added to the AVS.

• For a VXLAN deployment, know whether or not intermediate devices have Internet Group Management Protocol (IGMP) snooping on or off by default.

Guidelines and Limitations for VMM Integration with AVS or VDS

• When utilizing VLANs for VMM integration, regardless of Cisco Application Virtual Switch (AVS) or VMware vSphere Distributed Switch (VDS), the range of VLANs to be used for port groups must be manually allowed on any intermediate devices.

• For VMM integration with VLANs and a resolution immediacy of “On Demand” or “Immediate,” there can be a maximum of one hop in between the hosts and the compute node.

• For VMM integration with VXLAN, only the infrastructure VLAN needs to be allowed on all intermediate devices.

• For VMM integration with VXLAN, if the infra bridge domain subnet is set as a Querier, the intermediate devices must have Internet Group Management Protocol (IGMP) snooping enabled for traffic to pass properly.

Log in to the Advance Mode in the APIC GUI, choose Tenants > Tenant infra > Networking > Bridge Domains > default > Subnets > 10.0.0.30/27

• For VMM Integration with VXLAN and UCS-B, IGMP snooping is enabled on the UCS-B by default. Therefore, ensure that the Querier IP is enabled for the infra bridge domain. The other option is to disable IGMP snooping on UCS and disable the querier IP on the infra bridge domain.

Verifying the VMM Integration with AVS or VDS

The following procedures verify that the Cisco Application Virtual Switch (AVS) has been installed on the VMware ESXi hypervisor.

Verifying the Virtual Switch Status

This section describes how to verify the virtual switch status.

Procedure

Step 1 Log in to the VMware vSphere Client.
Step 2 Choose **Networking**.
Step 3 Open the folder for the data center and click the virtual switch.
Step 4 Click the **Hosts** tab. The VDS Status and Status fields display the virtual switch status. The VDS status should be "Up" to indicate that OpFlex communication has been established.

---

**Verifying the vNIC Status**

This section describes how to verify the vNIC status.

**Procedure**

<table>
<thead>
<tr>
<th>Step</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td>In VMware vSphere Client, click the <strong>Home</strong> tab.</td>
</tr>
<tr>
<td>Step 2</td>
<td>Choose <strong>Hosts and Clusters</strong>.</td>
</tr>
<tr>
<td>Step 3</td>
<td>Click the host.</td>
</tr>
<tr>
<td>Step 4</td>
<td>Click the <strong>Configuration</strong> tab.</td>
</tr>
<tr>
<td>Step 5</td>
<td>In the <strong>Hardware</strong> panel, choose <strong>Networking</strong>.</td>
</tr>
<tr>
<td>Step 6</td>
<td>In the <strong>View</strong> field, click the <strong>vSphere Distributed Switch</strong> button.</td>
</tr>
<tr>
<td>Step 7</td>
<td>Click <strong>Manage Virtual Adapters</strong>. The vmk1 displays as a virtual adapter and lists an IP address.</td>
</tr>
<tr>
<td>Step 8</td>
<td>Click the newly created vmk interface to display the vmknic status.</td>
</tr>
</tbody>
</table>

**Note** Allow approximately 20 seconds for the vmk to receive an IP address through DHCP.

---

**Additional References for VMM Integration with AVS or VDS**

For additional information on virtualization within ACI:


For additional information on ACI Integration and configuration with AVS:


For additional information on ACI integration with VMware:


For additional information on the AVS distributed firewall:

http://www.cisco.com/c/en/us/td/docs/switches/datacenter/aci/apic/sw/1-x/virtualization/b_ACI_Virtualization_Guide_1_3_x/b_ACI_Virtualization_Guide_1_3_x_chapter_0101.html#concept_E89432FC9DDF4F45A3AFB0EA826A7DEA
VMM Domain Resolution Immediacy

About VMM Domain Resolution Immediacy

Resolution immediacy determines at which point to push endpoint group policies to a compute node for end host usage. These policies include VLAN/VXLAN binding, contracts, and filters. Due to the dynamic nature of a VMM domain, most of the policy will wait for an indication of usage (as a trigger) before programming these values. There are certain scenarios where you will want to force programming onto the leaf node before usage. This section discusses both scenarios.

Prerequisites for VMM Domain Resolution Immediacy

The three resolution immediacies are defined as follows:

• Pre-provision—Specifies that a policy (for example, VLAN, VXLAN binding, contracts, or filters) is downloaded to a leaf switch even before a hypervisor is attached to the VMware vSphere Distributed Switch (VDS), thereby pre-provisioning the configuration on the switch.

• Immediate—Specifies that endpoint group policies (including contracts and filters) are downloaded to the associated leaf switch software upon hypervisor attachment to a VDS. LLDP or OpFlex permissions are used to resolve the hypervisor to leaf node attachments.

• On Demand—Specifies that a policy (for example, VLAN, VXLAN bindings, contracts, or filters) is pushed to the leaf node only when a pNIC attaches to the hypervisor connector and a virtual machine is placed in the port group (endpoint group).

Guidelines and Limitations for VMM Domain Resolution Immediacy

At a high level, the least strict definition of a policy comes from the "Pre-Provision" setting. This is essentially a static path, in that the resolution will program the endpoint group policies on all of the interfaces that are linked to that VMM domain as soon as the configuration is made. The resolution is not checking for any level of usage, and will program these interfaces even if the interfaces never get used. This option will pre-provision the VLAN on all ports using the AEP. If an AEP is tied to multiple domains, then the VLAN is pushed to all of the domains in that AEP.

The next level of definition comes from the "Immediate" setting. A resolution set to "Immediate" is only checking for hypervisor attachment to the vSphere Distributed Switch (VDS).

"On-Demand" is the strictest setting, as this has two checks in place to ensure that the policy is only programmed when truly in use. The resolution is looking for the following things:

• Hypervisor attachment to the Application Policy Infrastructure Controller (APIC)-provisioned VDS.

• VM assignment to a port group that was configured from an endpoint group within the APIC.

The value in having a stricter resolution immediacy means that various configurations can be staged from an APIC configuration view without having to worry about resource utilization until truly needed (VM attachment to a port group). However, there are certain virtualization scenarios where this is not ideal and the setting of "Pre-Provision" is truly needed. One such scenario is when migrating a hypervisor management VMK over
to the VDS from a standard vSwitch. Another scenario would be if the NICs of the attached hosts do not support either CDP or LLDP.

**Recommended Configuration Procedure for VMM Domain Resolution Immediacy**

When implementing a VMM domain for virtual machine traffic, a resolution immediacy of “On-Demand” or “Immediate” generally suffices. However, when planning to migrate a hypervisor management VMK over to an in-band VLAN through the VMware vSphere Distributed Switch (VDS), use the “Pre-Provision” immediacy. There are certain configurations that are specific to utilizing “Pre-Provision”:

**Procedure**

<table>
<thead>
<tr>
<th>Step 1</th>
<th>Choose a VLAN to be pre-provisioned.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 2</td>
<td>Add the chosen VLAN to a separate range (encap block) within the VLAN pool that is associated with the target VMM domain. The block where this VLAN is added must have the allocation mode set to <strong>Static Allocation</strong>. A static encap block can reside within a dynamic pool specifically for the purpose of using pre-provision.</td>
</tr>
<tr>
<td>Step 3</td>
<td>Create an endpoint group within the desired tenant.</td>
</tr>
<tr>
<td>Step 4</td>
<td>Verify that the bridge domain associated with the management endpoint group is also associated with a VRF.</td>
</tr>
<tr>
<td>Step 5</td>
<td>Associate the VMM domain to the target endpoint group.</td>
</tr>
<tr>
<td>Step 6</td>
<td>Use resolution immediacy <strong>Pre-Provision</strong>.</td>
</tr>
<tr>
<td>Step 7</td>
<td>Specify the management VLAN in the <strong>Port Encap</strong> field of the VM domain profile association.</td>
</tr>
</tbody>
</table>

As a result, the Application Policy Infrastructure Controller (APIC) creates a port group within the VDS with the specified VLAN. The APIC also pushes the endpoint group policies onto the leaf switches that are associated with the VMM domain and Attachable Access Entity Profile (AAEP).

**Verifying the VMM Domain Resolution Immediacy Configuration**

This section describes how to verify the VMM Domain Resolution Immediacy Configuration.

**Procedure**

VLAN programming can be verified by logging into the compute node CLI and running the following command:

```bash
show vlan extended
```

Depending on the immediacy, certain criteria must be met before you will see the VLAN programmed on any interfaces.
Additional References for VMM Domain Resolution Immediacy

For additional Information on resolution immediacy:


For additional information on pre-provision and management VMK:


OpenStack and Cisco ACI

About OpenStack and Cisco ACI

OpenStack defines a flexible software architecture for creating cloud-computing environments. The reference software-based implementation of OpenStack allows for multiple Layer 2 transports including VLAN, GRE, and VXLAN. The Neutron project within OpenStack can also provide software based Layer 3 forwarding. When utilized with ACI, the ACI fabric provides an integrated Layer 2/3 VXLAN-based overlay networking capability that can offload network encapsulation processing from the compute nodes onto the top-of-rack or ACI leaf switches. This architecture provides the flexibility of software overlay networking in conjunction with the performance and operational benefits of hardware-based networking.

Extending OpFlex to the Compute Node

OpFlex is an open and extensible policy protocol designed to transfer declarative networking policies such as those used in Cisco ACI to other devices. Utilizing OpFlex, the policy model native to ACI can be extended all the way down into the virtual switches running on OpenStack Nova compute hosts. This OpFlex extension to the compute host allows ACI to use Open vSwitch (OVS) to support common OpenStack features such as source NAT (SNAT) and floating IP addresses in a distributed manner.

The ACI OpenStack drivers support two distinct modes of deployment. The first approach is based on the Neutron API and Modular Layer 2 (ML2), which are designed to provide common constructs such as network, router, and security groups that are familiar to Neutron users. The second approach is native to the group-based policy abstractions for OpenStack, which are closely aligned with the declarative policy model used in Cisco ACI.

ACI with OpenStack Physical Architecture

A typical architecture for an ACI fabric with an OpenStack deployment consists of a Nexus 9000 spine/leaf topology, an APIC cluster, and a group of servers to run the various control and compute components of OpenStack. An ACI external routed network connection as a Layer 3 connection outside of the fabric can be used to provide connectivity outside the OpenStack cloud.
OpFlex Software Architecture

The Modular Layer 2 (ML2) framework in OpenStack allows the integration of networking services based on TypeDrivers and MechanismDrivers. Common networking type drivers include local, flat, VLAN, and VXLAN. OpFlex is added as a new network type through ML2, with an actual packet encapsulation of either VXLAN or VLAN on the host defined in the OpFlex configuration. A mechanism driver is enabled to communicate networking requirements from the Neutron servers to the Cisco APIC cluster. The APIC mechanism driver translates Neutron networking elements such as a network (segment), subnet, router, or external network into APIC constructs within the ACI policy model.

The OpFlex software stack also currently utilizes Open vSwitch (OVS), and local software agents on each OpenStack compute host that communicate with the Neutron servers and OVS. An OpFlex proxy from the ACI leaf switch exchanges policy information with the Agent-OVS instance in each compute host, effectively extending the ACI switch fabric and policy model into the virtual switch.
Logical OpenStack Topology

The logical topology diagram in the following figure illustrates the connections to OpenStack network segments from Neutron/controller servers and compute hosts, including the distributed Neutron services.
The management/API network for OpenStack can be connected to servers using an additional virtual NIC/sub-interface on a common uplink with tenant networking to the ACI fabric, or by way of a separate physical interface.

**Mapping OpenStack and ACI Constructs**

Cisco ACI uses a policy model to enable network connectivity between endpoints attached to the fabric. OpenStack Neutron uses more traditional Layer 2 and Layer 3 networking concepts to define networking configuration. The OpFlex ML2 driver translates the Neutron networking requirements into the necessary ACI policy model constructs to achieve the desired connectivity. The **OpenStack GBP Objects and Corresponding APIC Objects** table illustrates the OpenStack Neutron constructs and the corresponding APIC policy objects that will be configured when they are created. In the case of GBP deployment, the policies have a direct mapping to the ACI policy model.

**Table 1: OpenStack Neutron Objects and Corresponding APIC Objects**

<table>
<thead>
<tr>
<th>Neutron Object</th>
<th>APIC Object</th>
</tr>
</thead>
<tbody>
<tr>
<td>(Neutron Instance)</td>
<td>VMM Domain</td>
</tr>
<tr>
<td>Neutron Object</td>
<td>APIC Object</td>
</tr>
<tr>
<td>------------------------</td>
<td>----------------------------------</td>
</tr>
<tr>
<td>Project</td>
<td>Tenant + Application Network Profile</td>
</tr>
<tr>
<td>Network</td>
<td>EPG + Bridge Domain</td>
</tr>
<tr>
<td>Subnet</td>
<td>Subnet</td>
</tr>
<tr>
<td>Security Group + Rule</td>
<td>N/A (Iptables rules maintained per host)</td>
</tr>
<tr>
<td>Router</td>
<td>Contract</td>
</tr>
<tr>
<td>Network:external</td>
<td>L3Out/Outside EPG</td>
</tr>
</tbody>
</table>

Table 2: OpenStack GBP Objects and Corresponding APIC Objects

<table>
<thead>
<tr>
<th>GBP Object</th>
<th>APIC Object</th>
</tr>
</thead>
<tbody>
<tr>
<td>Policy Target</td>
<td>Endpoint</td>
</tr>
<tr>
<td>Policy Group</td>
<td>Endpoint Group (fvAEPg)</td>
</tr>
<tr>
<td>Policy Classifier</td>
<td>Filter (vzFilter)</td>
</tr>
<tr>
<td>Policy Action</td>
<td>--</td>
</tr>
<tr>
<td>Policy Rule</td>
<td>Subject (vzSubj)</td>
</tr>
<tr>
<td>Policy Ruleset</td>
<td>Contract (vzBrCP)</td>
</tr>
<tr>
<td>L2 Policy</td>
<td>Bridge Domain (fvBD)</td>
</tr>
<tr>
<td>L3 Policy</td>
<td>Context (fvCtx)</td>
</tr>
</tbody>
</table>

Prerequisites for OpenStack and Cisco ACI

This section lists the prerequisites for OpenStack and Cisco ACI:

- Target audience—Working knowledge of Linux, intended OpenStack distribution, ACI policy model and GUI-based APIC configuration.

- ACI fabric—ACI fabric installed and initialized with a minimum APIC version of 1.1(4e) and NX-OS version of 11.1(4e). For basic guidelines on initializing a new ACI fabric, see the relevant documentation. For communication between multiple leaf pairs, the fabric must have a BGP route reflector enabled to use an OpenStack external network.

- Servers—Controller and Compute servers connected to the fabric, preferably using NIC bonding and a vPC. In most cases the Controller does not need to be connected to fabric.

- L3-Out—For external connectivity, one or more Layer 3 outs configured on the ACI.

- VLAN mode—for VLAN mode, a non-overlapping VLAN pool of sufficient size should be allocated ahead of time.
Guidelines and Limitations for OpenStack and Cisco ACI

This section describes the guidelines and limitations for OpenStack and Cisco Application Centric Infrastructure (ACI).

**Scalability Guidelines**

There is a 1:1 correlation between the OpenStack tenant and the ACI tenant, and for each OpenStack tenant, the plugin automatically creates ACI tenants named according to the following convention:

```
convention_apic_system_id_openstack_tenant_name
```

You should consider the scalability parameters for supporting the number of required tenants.

Calculate the fabric scale limits for endpoint groups, bridge domains, tenants, and contracts before deployment. Doing so will limit the number of tenant/projects networks and routers that can be created in OpenStack.

There are per leaf and per fabric limits. Make sure to check the scalability parameters for the deployed release before deployment. In the case of GBP deployment, it can take twice as many endpoint groups and bridge domains than ML2 mode. The following tables list the Application Policy Infrastructure Controller (APIC) resources that are needed for each OpenStack resource in GBP and ML2 configurations.

**Table 3: OpenStack GBP and ACI Resources**

<table>
<thead>
<tr>
<th>GBP Resource</th>
<th>APIC Resources Consumed</th>
</tr>
</thead>
<tbody>
<tr>
<td>L3 Policy</td>
<td>1 context</td>
</tr>
<tr>
<td>L2 Policy</td>
<td>1 bridge domain</td>
</tr>
<tr>
<td></td>
<td>1 endpoint group</td>
</tr>
<tr>
<td></td>
<td>2 contract</td>
</tr>
<tr>
<td>Policy Group</td>
<td>1 endpoint group</td>
</tr>
<tr>
<td>Ruleset</td>
<td>1 contract</td>
</tr>
<tr>
<td>Classifier</td>
<td>2 filters (forward and reverse)</td>
</tr>
<tr>
<td></td>
<td><strong>Note</strong> 5 overhead classifiers are created.</td>
</tr>
</tbody>
</table>

**Table 4: OpenStack ML2 and ACI Resources**

<table>
<thead>
<tr>
<th>ML2 Resource</th>
<th>APIC Resources Consumed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Network</td>
<td>1 bridge domain</td>
</tr>
<tr>
<td></td>
<td>1 endpoint group</td>
</tr>
<tr>
<td>Router</td>
<td>1 contract</td>
</tr>
<tr>
<td>Security Groups</td>
<td>N/A (no filters are used)</td>
</tr>
</tbody>
</table>
Availability Guidelines

For redundancy, use bonded interfaces (vPCs) by connecting 2 interfaces to two leaf switches and creating a vPC in ACI.

You should deploy redundant OpenStack controller nodes to avoid a single point of failure.

The external network should also be designed to avoid a single point of failure and service interruption.

NAT/External Network Operations

The OpFlex driver software brings the capability to support external network connectivity and Network Address Translation (NAT) functions in a distributed manner using the local OVS instance on each OpenStack compute node. This distributed approach increases the availability of the overall solution and offloads the central processing of NAT from the Neutron server Layer 3 agent that is used in the reference implementation. You can also provide direct external connectivity without NAT or with a mix of NAT and non-NAT external connectivity.

Subnets Required for NAT

Contrary to the standard Neutron approach, three distinct IP subnets are required to take full advantage of external network functionality with the OpFlex driver.

- Link Subnet—This subnet represents the actual physical connection to the external next-hop router outside of the fabric to be assigned to a routed interface, sub-interface, or SVI.
- Source-NAT Subnet—This subnet is used for Port Address Translation (PAT), allowing multiple virtual machines to share an outside-routable IP address. A single IP address is assigned to each compute host and Layer 4 port number manipulation is used to maintain unique session traffic.
- Floating IP Subnet—The term "floating IP" in OpenStack is used when a virtual machine instance is allowed to claim a distinct static NAT address to support inbound connections to the virtual machine from outside of the cloud. The floating IP subnet is the subnet assigned within OpenStack to the Neutron external network entity.

For information about the external connectivity in OpFlex plugin, see the Cisco ACI with OpenStack OpFlex Architectural Overview document:


Optimized DHCP and Metadata Proxy Operations

The OpFlex driver software stack provides optimized traffic flow and distributed processing to provide DHCP and metadata proxy services for virtual machine instances. These services are designed to keep as much processing and packet traffic local to the compute host. The distributed elements communicate with centralized functions to ensure system consistency. You should enable optimized DHCP and metadata services when deploying the OpFlex plugin for OpenStack.

For information about how these optimized services work, see the Cisco ACI with OpenStack OpFlex Architectural Overview document:

**Physical Interfaces**

OpFlex uses the untagged fabric interface for an uplink trunk in VLAN mode. This means the fabric interface cannot be used for PXE, since PXE usually requires an untagged interface. If you require PXE in a VLAN mode deployment, you must use a separate interface for PXE. This interface can be connected through ACI or an external switch. This issue is not present in VXLAN mode since tunnels are created using the tagged interface for infra VLAN.

**Layer 4 to Layer 7 Services**

Service insertion in OpenStack is done through a physical domain or device package. Check customer requirements and the plugin mode (GBP, ML2) to plan how service insertion/chaining will be done. The OpenStack Neutron project also defines Layer 4 to Layer 7 extension APIs, such as LBaaS, FWaaS, and VPNaaS. The availability of these extensions depends on device vendors. Check the vendor for the availability of these extensions.

**Blade servers**

When deploying on the blade servers, you must make sure there is no intermediate switch between the fabric and the physical server interfaces. Check the OpenStack ACI plugin release notes to make sure the configuration is supported. At the time of this writing, there is limited support for B-Series blade servers and the support is limited to VLAN mode only.

**Verifying the OpenStack Configuration**

The following procedure verifies the OpenStack configuration:

**Procedure**

<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>Verify that a VMM domain was created for the OpenStack system ID defined during installation. The nodes connected to the fabric, running OpFlex agent, should be visible under Hypervisors. The virtual machines running on the hypervisor should be visible upon selecting that hypervisor. All networks created for this tenant should also be visible under the DVS submenu and selecting the network should show you all endpoints connected to that network.</td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>Look at the health score and faults for the entity to verify correct operation. If the hypervisors are not visible or show as disconnected, check the OpFlex connectivity.</td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td>Verify that there is a tenant created for the OpenStack tenant/project. All of the networks created in OpenStack should show up as endpoint groups and corresponding bridge domains. Choose the Operational tab for the endpoint group to show all of the endpoints for that endpoint group.</td>
</tr>
<tr>
<td><strong>Step 4</strong></td>
<td>Choose the Health Score tab and Faults tab to make sure that there are no issues.</td>
</tr>
</tbody>
</table>

**Configuration Examples for OpenStack and Cisco ACI**

This section describes the configuration examples for OpenStack and Cisco Application Centric Infrastructure (ACI).
Optimized Metadata and DHCP

In the configuration file, the optimized DHCP is enabled by default in OpFlex OpenStack plugin. To disable the optimized DHCP, add the following line:

```
enable_optimized_dhcp = False
```

In the configuration file, the optimized metadata service is disabled by default. To enable the optimized metadata, add the following line:

```
enable_optimized_metadata = True
```

For more information, see the Cisco ACI with OpenStack OpFlex Deployment Guide for Ubuntu:


For more information, see the Cisco ACI with OpenStack OpFlex Deployment Guide for Red Hat:


External Network/NAT Configuration

External network connectivity is defined by adding "apic_external_network" section to the configuration file. For example:

```
[apic_external_network:DC-Out]
preexisting=True
external_epg=DC-Out-EPG
host_pool_cidr=10.104.11.1/24
```

The `host_pool_cidr` defines the SNAT subnet. The floating IP subnet is defined by creating an external network in Neutron, or an external policy in GBP. The name of the external network or policy should use the same name as "apic_external_network" defined in the file (in this case "DC-Out").

It is possible to disable NAT by adding `enable_nat = False` in the above section. You can have multiple external networks using different Layer 3 Out on ACI, and have a mix of NAT and non-NAT external networks.

For more information on external network configuration, see the Cisco ACI with OpenStack OpFlex Deployment Guide for Ubuntu:


For more information on external network configuration, see the Cisco ACI with OpenStack OpFlex Deployment Guide for Red Hat:


Network configuration for GBP

In GBP deployment, network subnets for policy groups are carved out of the `default_ip_pool` defined in the plugin configuration file. For example:

```
[group_policy_implicit_policy]
default_ip_pool = 192.168.0.0/16
```

The above pool will be used to allocate networks for created policy groups. You must make sure that the pool is large enough for the intended number of groups.
Additional references for Openstack and Cisco ACI

For more information, see the following documents:

- Cisco ACI with OpenStack OpFlex Architectural Overview

- Cisco ACI with OpenStack OpFlex Deployment Guide for Ubuntu

- Cisco ACI with OpenStack OpFlex Deployment Guide for Red Hat

- Cisco ACI Installation Guide for Mirantis OpenStack
Additional references for Openstack and Cisco ACI