Cisco Nexus Fabric Enabler Installation

Installation Overview

Apart from OpenStack installation, the Nexus Fabric Enabler also needs to be installed. As described previously, OpenStack can be installed in multiple ways. Apart from installation, the OpenStack configuration file needs to have certain settings for the solution to work. The changes in the configuration file are needed for the following:

- Configuring OpenStack networking to work in conjunction with the VXLAN BGP EVPN farbic (Programmable Fabric) or DFA. This is done when OpenStack is installed for the first time. This step is dependent on the distribution. For example, RHEL OSP 7 may have a different command to perform this operation than Mirantis.
- Enabling notification functionality in OpenStack, which can be received by the Nexus Fabric Enabler. This step is done by the installer provided by the Nexus Fabric Enabler. This step is mostly common across distributions.

The Nexus Fabric Enabler solution is officially qualified with RHEL OSP 7. The solution is also tested with DevStack, which is not intended for production. This chapter starts with a section on Red Hat OSP installation that provides pointers on *configuring OpenStack networking to work in conjunction with Programmable Fabric or DFA*. Then, the installation mechanism of Nexus Fabric Enabler is explained in detail.

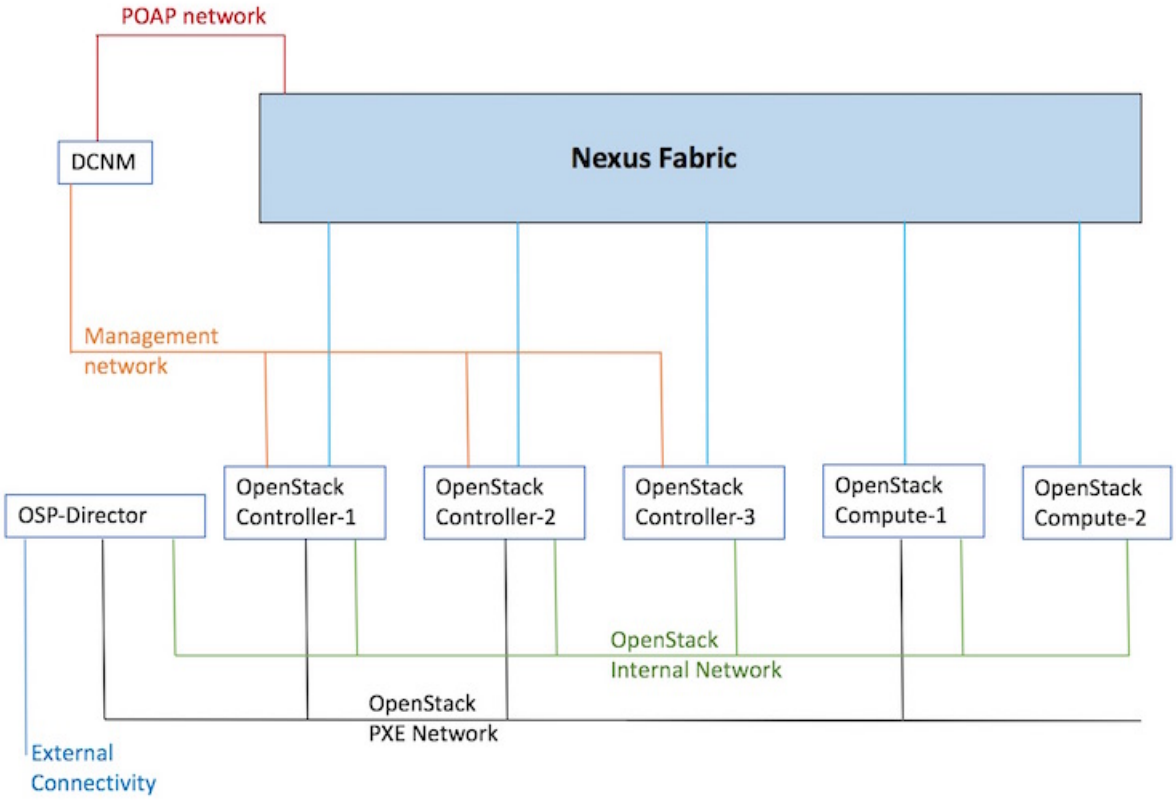
OpenStack Installation

Topology

Note **Important—**A sample topology is shown below. This is a critical step as it lays down the foundation for installation. It is highly recommended that you follow the same wiring scheme. This topology is based on RHEL OSP 7/8 with the Nexus fabric, Fabric Enabler, and DCNM. In the topology, the interface that is connected to the cluster should be operationally up (the corresponding command is **sudo ifconfig eth0 up**).

DCNM is also connected to the OpenStack control node through an IP network. Refer the OSP installation guide for more information on the OpenStack internal and PXE network.

Figure 1: Sample topology



RHEL OSP 7 Installation

Note Refer the official Red Hat documentation on how to install the Overcloud and the Undercloud.

Before deploying the Overcloud, implement the following steps to ensure compatibility with Nexus Fabric Enabler.

1. The Neutron type drivers must include *local*. Add these Neutron Type Drivers in *network-environment.yaml*, the network environment file—*local*, *flat*, *vlan*, *gre*, and *vxlan*.
2. Disable Neutron tunneling with the -- *neutron-disable-tunneling* option.
3. Set the Neutron Network Type to *local* with the -- *neutron-disable-tunneling* option.
4. Set the Neutron bridge mapping with the -- *neutron-bridge-mappings ethd:br-ethd* option.

An example for an Overcloud deployment command that is compatible with Nexus Fabric Enabler is given below:

```
openstack overcloud deploy --templates \

-e /usr/share/openstack-tripleo-heat-templates/environments/network-isolation.yaml \
-e /home/stack/templates/network-environment.yaml \
-e /home/stack/templates/storage-environment.yaml \
--control-flavor control \
--compute-flavor compute \
--ntp-server clock.cisco.com \
--neutron-network-type local \
--neutron-disable-tunneling \
--neutron-bridge-mappings ethd:br-ethd \
--compute-scale 2 \
--verbose
```

HA for RHEL OSP 7

High availability (HA) provides continuous operation. The RHEL OSP7 director provides high availability to an OpenStack Platform environment through the controller node cluster. The director installs a set of the same components on each controller node and manages them as one whole service. Having a cluster provides a fallback in case of operational failures on a single controller node.

The following example shows how to install the Overcloud with redundant controllers:

```
openstack overcloud deploy --templates \

-e /usr/share/openstack-tripleo-heat-templates/environments/network-isolation.yaml \
-e /home/stack/templates/network-environment.yaml \
-e /home/stack/templates/storage-environment.yaml \
--control-flavor control \
--compute-flavor compute \
--ntp-server clock.cisco.com \
--neutron-network-type local \
--neutron-disable-tunneling \
--neutron-bridge-mappings ethd:br-ethd \
--control-scale 3 \
--compute-scale 2 \
--verbose
```

Note For HA to work, you need a minimum of three controllers.

Note If compute nodes are connected to fabric leaf nodes through a port-channel/bond, make sure the Linux bond interfaces are used, since OVS bond interfaces will not work with Fabric Enabler. Create bond interfaces before installing and running the Openstack Fabric Enabler.

Verification of Configuration Files

This section is generally independent of the OpenStack installation method. But, check with the distribution as to where the configuration files are placed, and if the name of the configuration file has changed. As described earlier, the type_drivers should be set to local. *openvswitch* is used as the vswitch for the VXLAN BGP EVPN Programmable Fabric or DFA solution. Verify the following:

1. Ensure that *./etc/neutron/plugins/ml2/ml2_conf.ini* is configured as follows:

```
[ml2]
type_drivers = local mechanism_drivers = openvswitch

[ovs]
bridge_mappings = ethd:br-ethd

([ml2_type_flat], [ml2_type_vlan], [ml2_type_gre] and [ml2_type_vxlan] sections should
not be specified)
```

2. Ensure that *./etc/neutron/neutron.conf* is configured as follows:

```
core_plugin = neutron.plugins.ml2.plugin.Ml2Plugin
```

   Ensure that the tunnel type is not set. The *keystone_authtoken* should be similar to the following setting:

```
[keystone_authtoken]
signing_dir = /var/cache/neutron
auth_uri = http://<ip address of controller>:5000/v2.0
cafile = /opt/stack/data/ca-bundle.pem
identity_uri = http://<ip address of controller>:35357
auth_host = <ip address of controller>
auth_port = 35357
auth_protocol = http
admin_tenant_name = service
admin_user = neutron
admin_password = password
```

3. Ensure that *./etc/nova/ nova.conf* has the *keystone_authtoken* section(s) set similar to the following

```
[keystone_authtoken]
signing_dir = /var/cache/nova
admin_password = password
admin_user = nova
admin_tenant_name = service
auth_uri = http://<ip address of controller>:5000/v2.0
cafile = /opt/stack/data/ca-bundle.pem
identity_uri = http://<ip address of controller>:35357
auth_protocol = http
auth_port = 35357
auth_host = <ip address of controller>
```

Note Different OpenStack distributions may have the configuration file placed in a different directory. Or, the name of the configuration file can be different. Ensure that the appropriate configuration file has the above contents set correctly.

Cisco Nexus Fabric Enabler Installation

The sections below provides Cisco Nexus Fabric Enabler installation information.

Prerequisites

The following pre-requisite is applicable for RHEL OSP 7/8 based setups.

Note All the changes to Neutron and Nova files as well as extra rpm installation, Neutron DB patching for the firewall, disabling selinux, etc, that are described for a non-HA setup are still applicable.

- Cisco Nexus Fabric Enabler requires RabbitMQ to listen to requests made to the VIP address. You should configure this manually as other OpenStack components do not need them.

Installation in all the nodes (Fresh install)

1. Login to the OSP 7 Director Node for an RHEL OSP 7 or OSP 8 setup. In case of a non-production setup like DevStack, login to the controller node.
2. git clone -b rel_2_0_0 https://github.com/CiscoSystems/fabric_enabler ofe
3. cd ofe
4. Edit enabler_conf.ini as given in the next section.
5. The following command will install the Fabric Enabler on all the controller and compute nodes. This command will also install the Fabric Enabler in case of a HA setup. The two scenarios involving proxy requirement are given below:

   - If a proxy for reaching the Internet is not needed, use these commands:
     ```
     python setup_enabler.py  --vendor-os-release=rhel-osp7 --mysql-user=root
     --mysql-host=localhost
     ```
   - If a proxy for reaching the Internet is needed, use these commands:
     ```
     python setup_enabler.py  --vendor-os-release=rhel-osp7 --mysql-user=root
     --mysql-host=localhost --https-proxy=<proxy>
     ```
     <proxy> is the https proxy, such as *https://proxy.esl.cisco.com:80* from Cisco Labs.

   In case of HA setups for RHEL OSP 7 or OSP 8, the Enabler server will be started by Pacemaker. The Enabler server will be running in one of the servers. In case of a crash or the controller node going down, Pacemaker will ensure to restart the Enabler server in one of the available controllers. The Enabler agent and lldpad will be started in all the HA controller nodes and compute nodes.

Installation on a single node

Typically, this is needed when a new compute node is added after installing the Fabric Enabler on all the nodes:

1. Login to the OSP7 Director Node in case of RHEL OSP 7 or OSP 8 setups. In case of a non-production setup like DevStack, login to the controller node.
2. git clone -b rel_2_0_0 https://github.com/CiscoSystems/fabric_enabler ofe
3. cd ofe
4. Edit enabler_conf.ini as given in the next section.
5. Use the Nova list to get a list of controllers and compute node IP addresses.
6. For each compute node that is newly added, install the Fabric Enabler using these commands:

```
python setup_enabler.py --compute-name=<compute ip address> --remote-user=heat-admin
--https-proxy=<proxy>
```

Upgrading Fabric Enabler to the new version

The following are needed when you want to install a new version of the Fabric Enabler:

1. Ensure proxy variables are set properly.
2. Update the Fabric Enabler git repository using the following command:

```
git pull
```

Note If the git repository is not available, a new one needs to be cloned.

3. Ensure *enabler_conf.ini* is up to date. If needed, copy it from a controller.
4. If a proxy for reaching the internet is not needed, use these commands:

```
python setup_enabler.py  --vendor-os-release=rhel-osp7 --upgrade=True
```

5. If a proxy for reaching the internet is needed, use these commands::

```
python setup_enabler.py  --vendor-os-release=rhel-osp7 --https-proxy=<proxy> --upgrade=True
```

   a. More upgrade options/pointers are given below:

   - The upgrade process will upgrade and restart the Cisco Nexus Fabric Enabler server and agent.
   - At times, lldpad also needs to be restarted. If so, add *--restart-lldpad=True* to the above commands.
   - It is possible to update a single controller or compute note by using the *--control-name* and *--compute-name* options, respectively.

enabler_conf.ini

The following table describes the sections and fields in the *enabler_conf.in* file.

Note Unless specified as *(Optional)* in the Field Name column, the field is mandatory.

Table 1: Section—[General]

Field Name	Description	Default Value	Fabric Type
compute_user (Optional)	Compute node user name that can be used with the **ssh** command for remote logins. Also, the user must be a sudoer assuming all compute nodes have the same	Not Applicable	VXLAN BGP EVPN (Programmable Fabric) DFA
compute_passwd (Optional)	Compute node password that can be used with the **ssh** command for remote logins.	Not Applicable	VXLAN BGP EVPN DFA
node (Optional)	A *comma* separated list of hosts for which a static uplink is configured. The node name should be a fully qualified domain name (such as **host1.example.com**)	Not Applicable	VXLAN BGP EVPN DFA
node_uplink (Optional)	A comma separated list of uplink ports on the server connected to the leaf switch. This parameter and the *node* parameter are mandatory if a static uplink is desired.	Not Applicable	VXLAN BGP EVPN DFA

ucs_fi_evb_dmac (Optional)	If OpenStack is running in any UCS FI blade server, enter the EVB DMAC address that is configured in the fabric. The Fabric Enabler software running in the node will detect if it is a UCS FI blade server, but the interface connected to the switch has be included in the 'node_uplink' configuration, along with the node.	01:80:c2:12:34:56	VXLAN BGP EVPN DFA

This is the section about Cisco DCNM, which you have installed separately and set the right access credentials to be used by the OpenStack Cisco Nexus Fabric Enabler. Ensure that the *gateway_mac* value matches your POAP template setting in Cisco DCNM for your leaf switch, and you use the right range of segment IDs administrated by your Fabric Manager.

Table 2: Section—[dcnm]

Field Name	Description	Default Value	Fabric Type
dcnm_ip	IP address of the DCNM. It should be reachable from the OpenStack controller node.	Not Applicable	VXLAN BGP EVPN DFA
dcnm_user	DCNM server login credentials.	Not Applicable	VXLAN BGP EVPN DFA
dcnm_amqp_user	DCNM server RabbitMQ messaging credentials.	Not Applicable	VXLAN BGP EVPN DFA
dcnm_password	DCNM server password.	Not Applicable	VXLAN BGP EVPN DFA
gateway_mac (Optional)	Gateway MAC address. This should be the same as the MAC address configured on the leaf switch nodes.	20:20:00:00:00:AA	VXLAN BGP EVPN DFA

orchestrator_id (Optional)	Orchestartor ID used for registering the segment ID range on DCNM. If there are multiple setups using the same DCNM, ensure different orchestrator IDs are used.	Openstack Controller	VXLAN BGP EVPN DFA
segmentation_id_min	The minimum Segment ID value. It is a 24 bit integer value.	4097	VXLAN BGP EVPN DFA
segmentation_id_max	The maximum Segment ID value. It is a 24 bit integer value.	16777216	VXLAN BGP EVPN DFA
segmentation_reuse_timeout (Optional)	Duration after which an *available* segment ID can be reused. Once a segment ID is released, it will only be reused after 1 hour. If this functionality is not needed, enter a value of 0. Alternatively, to change the default value of 1 hour, uncomment the below and enter a different number (as an integer value).	1 hour	VXLAN BGP EVPN DFA
dcnm_net_ext (Optional)	The suffix of a network name when it is created by DCNM. This is usable for a scenario when network creation is done in DCNM and the Fabric Enabler populates this in OpenStack.	(DCNM)	VXLAN BGP EVPN DFA
dcnm_dhcp_leases (Optional)	The lease file name of the DHCP server on the DCNM.	/var/lib/dhcpd/dhcpd.leases	VXLAN BGP EVPN DFA
default_cfg_profile (Optional)	Default configuration profile when creating a network in DCNM.	defaultNetworkEvpnProfile defaultNetworkUniversalTfProfile	VXLAN BGP EVPN DFA
default_vrf_profile (Optional)	Default VRF profile name for a partition in DCNM.	vrf-common-evpn vrf-common-default	VXLAN BGP EVPN DFA

Table 3: Section—[dfa_rpc]

Field Name	Description	Default Value	Fabric Type
transport_url	Transport URL parameter for RPC.	Not Applicable An example is given below: `transport_url='rabbit://username:password@%(ip)s:5672//'` The *ip* address is of the controller when there is only one controller. In a HA environment with multiple controllers in a cluster, the IP address is the virtual IP address (VIP) of the controller cluster. You should replace the *username* and *password* based on your setting. These credentials should be the same as the one you used to configure RabbitMQ, by default available in the location */etc/rabbitmq/rabbitmq.config*.	VXLAN BGP EVPN DFA

Table 4: Section—[dfa_mysql]

Field Name	Description	Default Value	Fabric Type
connection	MYSQL DB connection option	Not Applicable An example is given below: `connection=mysql://username:password@localhost/cisco_dfa?charset=utf8` The *localhost* is applicable if there is only one controller. In a HA environment with multiple controllers in a cluster, a localhost will be replaced with the *VIP* of the controller cluster. You should replace the *username* and *password* based on your setting.	VXLAN BGP EVPN DFA

Table 5: Section—[dfa_notify]

Field Name	Description	Default Value	Fabric Type

cisco_dfa_notify_queue (Optional)	Notification queue name for DFA enabler. service_name: keystone and neutron.	cisco_dfa_%(service_name)s_notify	VXLAN BGP EVPN DFA

Table 6: Section—[dfa_log]

Field Name	Description	Default Value	Fabric Type
log_file (Optional)	Log file name. DEPRECATED (use Log file prefix instead). If log file name and directory is not specified, the default is the standard output.	fabric_enabler.log	VXLAN BGP EVPN DFA
log_file_prefix (Optional)	The prefix will be used by Fabric Enabler processes to create log files. If the default prefix of fabric_enabler is used, the Fabric Enabler server's log files will be fabric_enabler_server.log and the Fabric Enabler agent's log file will be fabric_enabler_agent.log	fabric_enabler	VXLAN BGP EVPN DFA
log_dir (Optional)	The directory name for the log file.	Current directory	VXLAN BGP EVPN DFA
log_level (Optional)	Enabler debugging output level. Set to DEBUG to see the debbugging output	WARNING	VXLAN BGP EVPN DFA

Table 7: Section—[dfa_agent]

Field Name	Description	Default Value	Fabric Type
integration_bridge (Optional)	OVS Neutron Agent related configuration. Ensure that this is the same as what is configured for the OVS Neutron Agent.	br-int	VXLAN BGP EVPN DFA

external_dfa_bridge (Optional)	OVS Neutron Agent related configuration. Ensure that this is the same as what is configured for the OVS Neutron Agent.	br-ethd	VXLAN BGP EVPN DFA

Table 8: Section—[vdp]

Field Name	Description	Default Value	Fabric Type
mgrid2 (Optional)	Refer to IEEE 801.1QBG standard documentation.	0	VXLAN BGP EVPN DFA
typeid (Optional)	Refer to IEEE 801.1QBG standard documentation.	0	VXLAN BGP EVPN DFA
typeidver (Optional)	Refer to IEEE 801.1QBG standard documentation.	0	VXLAN BGP EVPN DFA
vsiidfrmt (Optional)	Refer to IEEE 801.1QBG standard documentation.	5	VXLAN BGP EVPN DFA
hints (Optional)	Refer to IEEE 801.1QBG standard documentation.	Not Applicable	VXLAN BGP EVPN DFA
filter (Optional)	Refer to IEEE 801.1QBG standard documentation.	4	VXLAN BGP EVPN DFA
vdp_sync_timeout (Optional)	Query to lldpad for every VSI. If a VSI is not present in lldpad, an associate request is sent to lldpad.	15	VXLAN BGP EVPN DFA

Note

- Ensure that the segment ID does not overlap with other segment ID ranges (for example, the Cisco DCNM segment ID uses the default 30,000 to 49,999).
- It requires all control and compute nodes that have the same username and password, and this is your Linux account on the servers, to run as control/compute nodes.

Post Installation

Verify the Cisco Nexus Fabric Enabler server, Cisco Nexus Fabric Enabler agent, lldpad, and the existence of notification queues, as shown below:

Cisco Nexus Fabric Enabler server verification

On an RHEL OSP HA setup, a sample output is shown below on a controller:

```
[heat-admin@overcloud-controller-0 ~]$ sudo pcs resource | grep fabric-enabler-server
 fabric-enabler-server  (systemd:fabric-enabler-server):       Started overcloud-controller-1
```

The command displays the controller node where the Fabric Enabler server is running. Alternatively, the following commands can be used on the controller where the Fabric Enabler server is running:

- sudo systemctl status fabric-enabler-server [For a Redhat/CentOs based controller]
- sudo status fabric-enabler-server [For a Ubuntu based one]
- ps –ef | grep fabric-enabler-server [Any setup]

Cisco Nexus Fabric Enabler agent verification

The Enabler agent runs on all controller and compute nodes. Run the below commands on a node where verification is needed. This sample is specific to Red Hat based setups that have system based startup scripts.

```
[heat-admin@overcloud-controller-0 ~]$ sudo systemctl status fabric-enabler-agent
â fabric-enabler-agent.service - Cluster Controlled fabric-enabler-agent
   Loaded: loaded (/usr/lib/systemd/system/fabric-enabler-agent.service; enabled; vendor
preset: disabled)
  Drop-In: /run/systemd/system/fabric-enabler-agent.service.d
           ââ50-pacemaker.conf
   Active: active (running) since Fri 2016-09-23 01:31:01 EDT; 1 weeks 0 days ago
```

Alternatively, the following commands can be used:

- sudo status fabric-enabler-agent [On Ubuntu based servers]
- ps –ef | grep fabric-enabler-agent [Any setup]

lldpad verification

lldpad runs on all controller and compute nodes. Run the below commands on the compute node where verification is needed. The sample is specific to Red Hat based setups that have system based startup scripts.

```
[heat-admin@overcloud-controller-0 ~]$ sudo systemctl status lldpad
â lldpad.service - Cluster Controlled lldpad
   Loaded: loaded (/usr/lib/systemd/system/lldpad.service; enabled; vendor preset: disabled)

  Drop-In: /run/systemd/system/lldpad.service.d
           ââ50-pacemaker.conf
   Active: active (running) since Sun 2016-08-07 22:05:27 EDT; 1 months 22 days ago
 Main PID: 6041 (lldpad)
   CGroup: /system.slice/lldpad.service
           ââ6041 /usr/sbin/lldpad –t
```

Alternatively, the following commands can be used:

- sudo status lldpad [On Ubuntu based servers]

- ps –ef | grep lldpad [Any setups]

Verify existence of notification queues

```
sudo rabbitmqctl list_queues | grep cisco cisco_dfa_keystone_notify.info 0
cisco_dfa_neutron_notify.info 0
```