Dell EMC Red Hat OpenStack NFV Solution

Deployment Guide

Version 6.0

Dell EMC Validated Solutions

Contents

List of Figures

List of Tables

Trademarks

Notes, Cautions, and Warnings

A **Note** indicates important information that helps you make better use of your system.

A **Caution** indicates potential damage to hardware or loss of data if instructions are not followed.

A **Warning** indicates a potential for property damage, personal injury, or death.

This document is for informational purposes only and may contain typographical errors and technical inaccuracies. The content is provided as is, without express or implied warranties of any kind.

Chapter 1

Overview

Topics:

- *Summary*
- *Intended Audience*
- *Prerequisites*
- *Document Organization*

The Dell EMC Red Hat OpenStack NFV Solution Index of Documents contains links to all documents that you should read in order to successfully deploy the Dell EMC Red Hat OpenStack NFV Solution. The documents are listed in the order in which they are used.

In addition, the Index of Documents contains links to deployment scripts, tools, and workbooks.

Summary

This guide provides a detailed set of instructions on how to deploy the Dell EMC Red Hat OpenStack NFV Solution. This guide is intended to be followed in the order in which it is organized.

Intended Audience

This guide is written for OpenStack administrators or deployment engineers who are responsible for installation and ongoing operation of OpenStack clusters. It assumes that the reader is familiar with:

- OpenStack
- Red Hat Enterprise Linux (RHEL)
- Red Hat OpenStack Platform (RHOSP) documentation
- Networking and system administration

Prerequisites

In order to deploy the Dell Red Hat OpenStack Platform 9 Solution on Dell hardware, the following prerequisites must be met:

- Successful hardware setup and configuration, per the *Dell EMC Red Hat OpenStack NFV Solution Hardware Deployment Guide*
- Must have a valid subscription and outbound network access to Red Hat's Content Delivery Network.

Document Organization

Deployment of the Dell EMC Red Hat OpenStack NFV Solution on Dell hardware begins with installation of the Solution Admin Host (SAH). Before proceeding to the SAH installation there are some prerequisites that must first be satisfied, such as obtaining valid Red Hat subscriptions and gathering of proxy information.

Dell EMC recommends that you first review this document to familiarize yourself with terms and concepts before proceeding. Another document that would be helpful to review and print out is the *Dell EMC Red Hat OpenStack NFV Solution Workbook*. The workbook contains essential information that you will need as you proceed through this document.

To deploy the Dell Red Hat OpenStack Platform, complete the following tasks:

1. Determine the *Red Hat Subscription Manager Pool IDs*
2. *Solution Admin Host Deployment* on page 14
3. *RHEL OSP Director Node Deployment* on page 21
4. *Red Hat Ceph Storage Admin Node Deployment* on page 27
5. *Provisioning the Nodes* on page 31
6. *Red Hat Ceph Storage Integration* on page 48

Note: Performing all of these tasks is very complex, so please take your time and follow the steps closely.

Chapter 2

Red Hat Subscriptions

Topics:

Once all prerequisites have been met, you must determine the appropriate Red Hat subscription entitlements for each cluster node.

Determining Pool IDs

To determine the pool IDs, you must have an existing server that is registered to the Red Hat Hosted Services. This server must also be registered using the same credentials as the ones being used in this environment.

1. Once the server is correctly registered, execute the following command to see the available subscription pools.

   ```
   # subscription-manager list --all --available
   ```

   The command will output a list of available pools. Each section of information lists what the subscription provides, its pool ID, how many are available, the type of system it is for, as well as other information.
2. Determine the correct pool ID needed for this environment and take note of it.

   Note: Pay close attention to the **System Type**. The System Type can be *Virtual* or *Physical*. If necessary you can use a physical license for a virtual node. However, you cannot use a virtual license for a physical node.

```
# subscription-manager list --all --available

[OUTPUT ABBREVIATED]

Subscription Name: Red Hat Cloud Infrastructure, Standard (8-sockets)
Provides:          Red Hat Beta
                   Red Hat OpenStack Beta
                   JBoss Enterprise Application Platform
                   Red Hat Software Collections (for RHEL Server)
                   Red Hat Enterprise Virtualization
                   Oracle Java (for RHEL Server)
                   Red Hat OpenStack
                   Red Hat Enterprise MRG Messaging
                   Red Hat Enterprise Linux Server
                   Red Hat Enterprise Linux High Availability (for RHEL
 Server)
                   Red Hat Software Collections Beta (for RHEL Server)
                   Red Hat Enterprise Linux Load Balancer (for RHEL Server)
                   Red Hat CloudForms
SKU:               MCT2861
Pool ID:           aaaa111bbb222ccc333ddd444eee5556
Available:         7
Suggested:         1
Service Level:     Standard
Service Type:      L1-L3
Multi-Entitlement: No
Ends:              09/23/2015
System Type:       Physical

[OUTPUT ABBREVIATED]
```

The above output shows a subscription that contains the Red Hat OpenStack entitlement. The required entitlement types for each node are shown in *Table 1: Red Hat Subscription Entitlements* on page 12.

Table 1: Red Hat Subscription Entitlements

Node Role	Entitlement	System Type
Automation Control System	Red Hat Enterprise Linux Server	physical

Node Role	Entitlement	System Type
Solution Admin Host	Red Hat Enterprise Linux Server	physical
Director Node	Red Hat OpenStack	virtual
Red Hat Ceph Storage Admin Node	Red Hat Ceph Storage Calamari	physical (no virtual available at this time)
Controller Node	Red Hat OpenStack	physical
Compute Node	Red Hat OpenStack	physical
Storage Node	Red Hat Ceph Storage	physical

Chapter 3

Solution Admin Host Deployment

Topics:

The SAH hosts at least two (2) virtual machines (VMs):

- **Red Hat OpenStack Platform Director node** - Used for hosting Red Hat OSP Director deployment software, configuring OpenStack, and deployment of OpenStack software to the Controller, Compute, and Storage nodes
- **Red Hat Ceph Storage Admin Node -** Used for hosting the Calamari WEB based management platform

Solution Admin Host Deployment Overview and Prerequisites

Installation of the Dell EMC Solution Admin Host begins with the installation of Red Hat Enterprise Linux Server 7. The *osp-sah.ks* kickstart file is provided to assist automation of this process. The installation process can be accomplished using different processes (CD-ROM, CD image, or via a PXE installation). Instructions for how to include the kickstart file are provided later in this document.

This kickstart file performs the following steps when properly configured:

- Partitions the system
- Sets SELinux to *permissive* mode
- Disables `firewalld`, and uses `iptables`
- Disables NetworkManager
- Configures networking, including:
    - Bonding
    - Bridges
    - Static IP addresses
    - Gateway
    - Name resolution
    - NTP service
- Registers the system using the Red Hat Subscription Manager

Additionally, there are some requirements that must be satisfied prior to installation of the OS:

- A Red Hat subscription license
- Access to the Subscription Manager hosts

Note: If your network configuration/firewall require them, you must provide the proxy values in order to access Red Hat's Subscription Manager servers.

The osp-sah.ks Kickstart File

The *Dell-OSP-6.0.0.tgz* archive file (see *File References* on page 50) contains the *osp-sah.ks* kickstart file, and must be customized for the environment into which it is being installed.

Note: All edits that usually require changes are in the section marked **CHANGEME** and **END of CHANGEME**. Do not make other edits outside of these lines.

There are many changes that you will need to make, so a brief description of the SAH networks might help clarify the need for the variables in *Figure 1: Solution Admin Host Internal Network Fabric* on page 16.

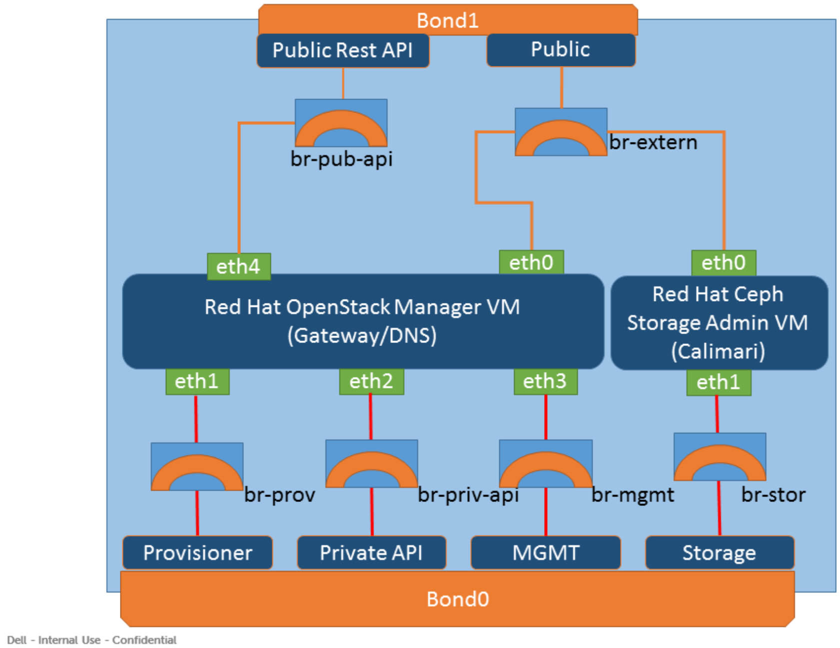

Figure 1: Solution Admin Host Internal Network Fabric

There are 4 network interface files, created during the OS installation, that are required by the SAH:

- `ifcfg-em1`
- `ifcfg-em2`
- `ifcfg-p1p1`
- `ifcfg-p1p2`

Note: The interfaces names (`ifcfg-em1`, `ifcfg-em2`, `ifcfg-p1p1` and `ifcfg-p1p2`) might be different on different system configurations, so these might need to change for your hardware configuration.

These are combined to build the bond interfaces (`bond0` and `bond1`). Interfaces `em1` and `p1p1` (**private_ifaces**) are combined to form `bond0` (**private_bond_name**) which is used by both the provisioning and storage networks. Interfaces `em2` and `p1p2` (**public_ifaces**) are combined to form `bond1` (**public_bond_name**). The other bond variables listed in the table are the **public_boot_opts, public_bond_opts, private_boot_opts** and **private_bond_opts**. These four variables can be left as is unless your configuration requires their modification.

Two VLANs are created to segregate the provisioning and storage traffic across the private network. For our example, we set the **provision_bond_name** to *bond0.120* and **storage bond_name** to *bond0.170* and the **provision_boot_opts** and **storage boot_opts** to *"onboot none vlan"*. You must configure the values for these variables to match your configuration.

The SAH requires three bridges to allow network traffic to pass through the SAH to and from the VM instances. The **public_bridge_boot_opts, provision_bridge_boot_opts** and **storage_bridge_boot_opts** variables are the last changes you must make to the kickstart file. For these variables you must set the boot options, IP address and netmask. The `osp-sah.ks` kickstart file provides an example of the format required.

To customize the kickstart file:

1. Extract the contents of the archive file using the `tar -zxf` command or a Windows® archive utility (*7zip*, *WinRaR*, etc).

2. Extract the contents of the *dell-mgmt-node.tgz* file, where you will find the *osp-sah.ks* that you will change as described below.
3. Set the following variables:

Table 2: Kickstart File Variables

Variable	Description
HostName	The FQDN of the server, e.g., sah.acme.com.
SystemPassword	The root user password for the system.
SubscriptionManagerUser	The user credential when registering with Subscription Manager.
SubscriptionManagerPassword	The user password when registering with Subscription Manager.
SubscriptionManagerPool	The pool ID used when attaching the system to an entitlement.
SubscriptionManagerProxy	Optional proxy server to use when attaching the system to an entitlement.
SubscriptionManagerProxyPort	Optional port for the proxy server.
SubscriptionManagerProxyUser	Optional user name for the proxy server.
SubscriptionManagerProxyPassword	Optional password for the proxy server.
Gateway	The default gateway for the system.
NameServers	A comma-separated list of nameserver IP addresses.
NTPServers	A comma-separated list of time servers. This can be IP addresses or FQDNs.
TimeZone	The time zone in which the system resides.
anaconda_interface	The public interface that allows connection to Red Hat Subscription services.
extern_bond_name	The name of the bond that provides access to the external network.
extern_boot_opts	The boot options for the bond on the external network. Typically, there no need to change this variable.
extern_bond_opts	The bonding options for the bond on the external network. Typically, there no need to change this variable.
extern_ifaces	A space delimited list of interface names to bond together for the bond on the external network.
internal_bond_name	The name of the bond that provides access for all internal networks.
internal_boot_opts	The boot options for the bond on the internal network. Typically, there no need to change this variable.

Variable	Description
internal_bond_opts	The bonding options for the bond on the internal network. Typically, there no need to change this variable.
internal_ifaces	A space delimited list of interface names to bond together for the bond on the internal network.
mgmt_bond_name	The VLAN interface name for the management network.
mgmt_boot_opts	The boot options for the management VLAN interface. Typically, there no need to change this variable.
prov_bond_name	The VLAN interface name for the provisioning network.
prov_boot_opts	The boot options for the provisioning VLAN interface. Typically, there no need to change this variable.
stor_bond_name	The VLAN interface name for the storage network.
stor_boot_opts	The boot options for the storage VLAN interface. Typically, there no need to change this variable.
pub_api_bond_name	The VLAN interface name for the public API interface.
pub_api_boot_opts	The boot options for the public API VLAN interface. Typically, there no need to change this variable.
priv_api_bond_name	The VLAN interface name for the private API interface.
priv_api_boot_opts	The boot options for the private API VLAN interface. Typically, there no need to change this variable.
br_extern_boot_opts	The bonding options, IP address and netmask for the external bridge.
br_mgmt_boot_opts	The bonding options, IP address and netmask for the management bridge.
br_prov_boot_opts	The bonding options, IP address and netmask for the provisioning bridge.
br_stor_boot_opts	The bonding options, IP address and netmask for the storage bridge.
br_pub_api_boot_opts	The bonding options, IP address and netmask for the public API bridge.
br_priv_api_boot_opts	The bonding options, IP address and netmask for the private API bridge.

Making the Kickstart File Available for Installation

This procedure places the kickstart file in the top level of a USB image and makes it available for installation. This is useful if you are using the iDRAC virtual media to install.

Note: The USB key must be formatted as *vfat* or *ext2*.

The following example performs these steps:

1. Creates a *ks* image
2. Mounts the image through a loopback device
3. Copies the *osp-sah.ks* file to the image
4. Unmounts the image from the system

The resulting image can be used as removable media for PXE boot or iDRAC. As this is an example, please refer to the man pages or the reference manual for further information on the commands.

There are several options for presenting the *osp-sah.ks* to the OS installation. Below are two ways of preparing the image prior to presenting it to the OS installation for inclusion into the installation process:

- *Preparing an Image File for use with iDRAC (Option 1)* on page 19
- *Preparing a USB Key for Physical Boot (Option 2)* on page 19

Preparing an Image File for use with iDRAC (Option 1)

1. From an existing RHEL 7.2 system, create a USB image:

```
mkfs.vfat -C ks_usb.img 1024
```

2. Mount the image:

```
mount -o loop ks_usb.img /mnt
```

3. Place the `osp-sah.ks` file into the image:

```
cp osp-sah.ks /mnt
```

4. Unmount the image:

```
sync; umount /mnt
```

5. Make the image file, ks_usb.img, available using the *Map the image as Removable Media* option on the iDRAC.

   Note: If only one physical hard disk is presented to the server, the device name presented to the installer should be `sdb`.

Preparing a USB Key for Physical Boot (Option 2)

1. From an existing RHEL 7.2 system, format a USB key:

```
mkfs.ext3 /dev/sdb
```

2. Mount the USB key.

```
mount /dev/sdb /mnt
```

3. Place the `osp-sah.ks` file onto the USB key:

```
cp osp-sah.ks /mnt
```

4. Unmount the image:

   Note: If only one physical hard disk is presented to the server, the device name presented to the installer should be `sdb`.

   ```
   sync; umount /mnt
   ```

Presenting the Image to the RHEL OS Installation Process

1. Boot the Solution Admin Host using the Red Hat Enterprise Server 7.x installation media.
   a. At the installation menu, select the **Install** option. *Do not press the [Enter] key*.
   b. Press the **Tab** key.
   c. Move the cursor to the end of the line that begins with `vmlinuz`.
   d. Append the following to the end of the line:

      Note: The device `sdb` can change, depending upon the quantity of disks being presented to the installation environment. These instructions assume that a single disk is presented. If otherwise, adjust accordingly.

      ```
      ks=hd:sdb:/osp-sah.ks
      ```
2. Press the **[Enter]** key to start the installation.

Note: It may take a few minutes before progress is seen on the screen. Press the ESC key at the memory check to speed up the process.

Next Steps

After the SAH is installed:

1. Copy the ISO of the **Red Hat Enterprise Linux Server 7 installation DVD** to the `/store/data/iso` directory. **Only RHEL 7.2 is supported**. This ISO is used to install the RHEL OSP Director Node and Red Hat Ceph Storage Admin Node.
2. Set up the Director Node by following the procedures in *RHEL OSP Director Node Deployment* on page 21.

Chapter 4

RHEL OSP Director Node Deployment

Topics:

The deployment of the RHEL OSP Director Node (Director Node) is performed using the `deploy-director-vm.sh` script. This script creates a kickstart file and then executes the `virt-install` command to install the system.

The Director Node Kickstart file

The generated kickstart script performs the following steps:

- Partitions the system
- Sets SELinux to enforcing mode
- Configures `iptables` to ensure the following services can pass traffic:
  - HTTP
  - HTTPS
  - DNS
  - TFTP
  - TCP port 8140
- Configures networking, including:
  - Static IP addresses
  - Gateway
  - Name resolution
  - NTP time service
- Registers the system using the Red Hat Subscription Manager
- Installs the RHEL OSP Director installer
  - Configures the RHEL OSP Director installer to not install the EPEL repository

Setup

To set up the Director Node deployment:

1. Log into the SAH node as the *root* user.
2. Ensure that a copy of the **Red Hat Enterprise Linux Server 7 Installation DVD** ISO (RHEL 7.2) is in the `/store/data/iso` directory.
3. Download the *dell-mgmt-node* archive (zip or tgz) file and extract the contents of the archive file into the */root* directory as per the example below:

```
# cd /root
# tar zxvf /PATH/TO/FILE/dell-mgmt-node.tgz
```

Configuration

To configure the Director Node deployment:

1. Edit the configuration file, named `director.cfg`, in the `/root/mgmt` directory.
2. Set the following variables in the `director.cfg` file:

Table 3: Director Node Configuration Parameters

Parameter	Description
rootpassword	The root user password for the system.
timezone	The timezone the system is in.

Parameter	Description
smuser	The user credential when registering with Subscription Manager.
smpassword	The user password when registering with Subscription Manager. The password must be enclosed in single quotes if it contains certain special characters.
smpool	The pool ID used when attaching the system to an entitlement. **Note:** Edit the line with the smpool-changeme. The # smpool line is an example only.
hostname	The FQDN of the Director Node.
gateway	The default gateway for the system.
nameserver	A comma-separated list of nameserver IP addresses.
ntpserver	A comma-separated list of time servers. This can consist of IP addresses or FQDNs.
user	The ID of an admin user to create to use for installing RHEL OSP Director. Default admin user is *stack*.
password	The password for the admin user.
eth0	This line specifies the IP address and network mask for the external network. The line begins with `eth0`, followed by at least one space, the IP address of the VM on the external network, another set of spaces, and then the network mask.
eth1	This line specifies the IP address and network mask for the provisioning network. The line begins with `eth1`, followed by at least one space, the IP address of the VM on the provisioning network, another set of spaces, and then the network mask.
eth2	This line specifies the IP address and network mask for the management network. The line begins with `eth2`, followed by at least one space, the IP address of the VM on the management network, another set of spaces, and then the network mask.
eth3	This line specifies the IP address and network mask for the private API network. The line begins with `eth3`, followed by at least one space, the IP address of the VM on the management network, another set of spaces, and then the network mask.

Parameter	Description
eth4	This line specifies the IP address and network mask for the public API network. The line begins with `eth4`, followed by at least one space, the IP address of the VM on the management network, another set of spaces, and then the network mask.

Installing the RHEL OSP Director Node

To install the Director Node:

1. Invoke the `deploy-director-vm.sh` script.
   a. Pass `director.cfg` as the first parameter.
   b. Pass the full path to the Red Hat Enterprise Linux Server 7 Installation media as the second parameter.

```
# ./deploy-director-vm.sh director.cfg /store/data/iso/rhel-server-7.2-
x86_64-dvd.iso
Starting install...
Retrieving file .treeinfo...
| 3.2 kB 00:00:00
Retrieving file vmlinuz...
| 7.9 MB 00:00:00
Retrieving file initrd.img...
| 64 MB 00:00:00
Creating storage file director.img
| 16 GB 00:00:00
Creating domain...
| 0 B 00:00:00
Domain installation still in progress. You can reconnect to
the console to complete the installation process.
```

   The installation will begin, but no console will be displayed.
2. To display the console:
   a. Ensure that you are logged into a GUI environment.
   b. Open a terminal.
   c. Enter the following command:

   Note: If you are connected to the Director Node using a Windows® system, you must install and configure an **Xwin Server** before executing `virt-viewer director` to see the output.

```
virt-viewer director
```

3. A console for the Director Node will open.
4. After the Director Node completes the installation, it will power itself off.
5. The power state of the Director Node can be viewed using the `virsh list --all` command:

```
Id Name State
----------------------------------------------------
2 director shut off
```

6. The Director Node can be started using the following command:

```
# virsh start director
```

Installing RHEL OSP Director

To install RHEL OSP Director:

1. Log into the Director Node using the user name and password specified in *director.cfg*.
2. Download the `dell-pilot-deploy.tgz` archive file. See *Solution Files* on page 51 for a list of solution files for the Dell EMC Red Hat OpenStack Cloud Solution.
3. Extract the contents of the `tar` archive file into the home directory of the user:

```
$ cd
$ tar xzvf /PATH/TO/FILE/dell-pilot-deploy.tgz
```

4. Edit the `~/pilot/undercloud.conf` configuration file.
5. Set the following variables in `undercloud.conf`:

Table 4: Undercloud Configuration Parameters

Parameter	Description
local_ip	The IP address and prefix of the Director Node on the provisioning network in CIDR format (xx.xx.xx.xx/yy). This must be the IP address used for *eth1* in *director.cfg*. The prefix used here must correspond to the netmask for *eth1* as well (usually *24*).
masquerade_network	The network address and prefix of the Director Node on the provisioning network in CIDR format (xx.xx.xx.xx/yy). This must be the network used for *eth1* in *director.cfg*. The prefix used here must correspond to the netmask for *eth1* as well (usually *24*).
dhcp_start	The starting IP address on the provisioning network to use for OpenStack cloud nodes. **Note:** Ensure the IP address of the Director Node is not included.
dhcp_end	The ending IP address on the provisioning network to use for OpenStack cloud nodes.
network_cidr	The network and prefix in CIDR format for the Neutron managed network for Overcloud instances.
network_gateway	The network gateway for Neutron-managed Overcloud instances.
inspection_iprange	An IP address range on the provisioning network to use during node inspection. Note that this should not overlap with the *dhcp_start/dhcp_end* range.

6. Set the passwords, if desired, in the `[auth]` section; otherwise passwords will be randomly generated.

Note: The *undercloud_heat_encryption_key* parameter **must** be either 16, 24, or 32 characters in length in order for RHEL OSP Director to successfully create a stack.

7. Determine the IP address of a DNS server for the Overcloud nodes to use.
8. Execute the following command:

```
$ ~/pilot/install-director.sh <dns_ip> <subscription_manager_user>
 <subscription_manager_pass> <subscription_manager_poolid>
```

Where:

- `dns_ip` = IP address of the DNS server
- `subscription_manager_user` = Red Hat Subscription Manager user, as specified in *Red Hat Subscription Manager Pool IDs*
- `subscription_manager_pass` = Red Hat Subscription Manager user password, as specified in *Red Hat Subscription Manager Pool IDs*
- `subscription_manager_poolid` = ID of the **Red Hat Ceph Storage** pool, as specified in *Red Hat Subscription Manager Pool IDs*

The RHEL OSP Director installation can take approximately 30 minutes to complete.

Note: The installation log is available at *~/pilot/install-director.log*.

Using Red Hat OpenStack Platform Director

The RHEL OSP Director installer creates an *rc* file for using the CLI commands, and a file containing all passwords.

1. Before running any CLI commands, first `source` the *rc* file:

```
source ~/stackrc
```

2. All of the passwords are in the `undercloud-passwords.conf` file.

Next Steps

After the RHEL OSP Director Node is installed:

1. Install the Red Hat Ceph Storage Admin Node by following the procedures in *Red Hat Ceph Storage Admin Node Deployment* on page 27.

Chapter 5

Red Hat Ceph Storage Admin Node Deployment

Topics:

The deployment of the Red Hat Ceph Storage Admin Node is performed using the `deploy-ceph-vm.sh` script. This script creates a kickstart file and then executes the `virt-install` command to install the system.

The Red Hat Ceph Storage Admin Node Kickstart File

The generated kickstart script performs the following steps:

- Partitions the system
- Sets SELinux to enforcing mode
- Configures `iptables` to run on the system and disables `firewalld`
- Configures networking, including:
  - Static IP addresses
  - Gateway
  - Name resolution
  - NTP time service
- Registers the system using the Red Hat Subscription Manager

Setup

To set up the Red Hat Ceph Storage Admin Node deployment:

1. Log into the SAH node as the *root* user.
2. Ensure that a copy of the **Red Hat Enterprise Linux Server 7 Installation DVD** ISO (RHEL 7.2) is in the `/store/data/iso` directory.

   Several steps in this document use files to configure the environment. See *File References* on page 50 for a list of required files.

   Note: Installation of these files should have been completed earlier in *RHEL OSP Director Node Deployment* on page 21.

Configuration

To configure the Red Hat Ceph Storage Admin Node deployment:

1. Edit the `ceph.cfg` configuration file, in the `/root/mgmt` directory.
2. Set the following variables in `ceph.cfg`:

Table 5: Red Hat Ceph Storage Admin Node Configuration Variables

Parameter	Description
rootpassword	The root user password for the Red Hat Ceph Storage Admin Node.
timezone	The timezone in which the Red Hat Ceph Storage Admin Node is located.
smuser	The user credential when registering with Subscription Manager.
smpassword	The user password when registering with Subscription Manager. The password must be enclosed in single quotes if it contains certain special characters.

Parameter	Description
smpool	The pool ID used when attaching the Red Hat Ceph Storage Admin Node to an entitlement. **Note:** Edit the line with the smpool-changeme. The # smpool line is an example only.
hostname	The FQDN of the Red Hat Ceph Storage Admin Node.
gateway	The default gateway for the Red Hat Ceph Storage Admin Node.
nameserver	A comma-separated list of nameserver IP addresses.
ntpserver	A comma-separated list of time servers. This can consist of IP addresses or FQDNs.
eth0	This line specifies the IP address and network mask for the external network. The line begins with `eth0`, followed by at least one space, the IP address, another set of spaces, and then the network mask.
eth1	This line specifies the IP address and network mask for the storage network. The line begins with `eth1`, followed by at least one space, the IP address, another set of spaces, and then the network mask.

Installing the Red Hat Ceph Storage Admin Node

To install the Red Hat Ceph Storage Admin Node:

1. Invoke the `deploy-ceph-vm.sh` script.
   a. Pass **`ceph.cfg`** as the first parameter.
   b. Pass the **full path to the Red Hat Enterprise Linux Server 7 Installation media** as the second option.

```
# ./deploy-ceph-vm.sh ceph.cfg /store/data/iso/rhel-server-7.2-x86_64-
dvd.iso

Starting install...
Retrieving file .treeinfo...
| 3.2 kB  00:00:00
Retrieving file vmlinuz...
| 7.9 MB  00:00:00
Retrieving file initrd.img...
|  64 MB  00:00:00
Creating storage file ceph.img
|  16 GB  00:00:00
Creating domain...
|    0 B  00:00:00
Domain installation still in progress. You can reconnect to
the console to complete the installation process.
```

   The installation begins, but no console is displayed.

2. To display the console:
   a. Ensure you are logged into a GUI environment.
   b. Open a terminal.
   c. Enter the following command:

      Note: If you are connected to the RHEL OSP Director Node using a Windows® system, you need to install **Xwin Server** before executing `virt-viewer ceph`.

      ```
      # virt-viewer ceph
      ```
3. A console for the Red Hat Ceph Storage Admin Node will open.
4. After the Red Hat Ceph Storage Admin Node completes the installation, it will power itself off.
5. The power state of the Red Hat Ceph Storage Admin Node can be viewed using the virsh list --all command:

   ```
   # virsh list --all
   ```

   You will see output similar to the following:

   ```
   Id    Name                           State
    ----------------------------------------------------
    2     ceph                          shut off
   ```
6. You can start the Red Hat Ceph Storage Admin Node by entering the following command:

   ```
   # virsh start ceph
   ```

Next Steps

After the Red Hat Ceph Storage Admin Node is installed:

1. Provision the Controller, Compute, and Storage nodes by following the procedures in *Provisioning the Nodes* on page 31.

Chapter

6

Provisioning the Nodes

Topics:

This topic describes provisioning the Dell EMC Red Hat OpenStack Cloud Solution cluster nodes.

Node Discovery

To discover the nodes in the cluster:

1. Log into the Director Node using the user name and password specified in *director.cfg*.
2. Determine the node iDRAC IP address(es) to scan on the management network.

   Note: The IP address(es) should only include nodes that you wish to provision. It should not include other nodes, such as the SAH.
3. Navigate to the *discover_nodes* subdirectory:

   ```
   $ cd ~/pilot/discover_nodes
   ```
4. Discover the nodes using a range of IP addresses, a space-separated list of IP addresses, or a CIDR block, the iDRAC user and the iDRAC password. For example:

   ```
   $ ./discover_nodes.py x.x.x.x-y.y.y.y -u root -p calvin \
    > ~/instackenv.json
   ```

   ```
   $ ./discover_nodes.py x.x.x.x x.x.x.y x.x.x.z -u root -p calvin \
    > ~/instackenv.json
   ```

   ```
   $ ./discover_nodes.py x.x.x.0/24 -u root -p calvin \
    > ~/instackenv.json
   ```

   a. Optional arguments include:
      - `-h`, `--help` — Display help for `discover_nodes.py`
      - `-u USERNAME`, `--u USERNAME` — Specify the iDRAC access user name. Default = *root*.
      - `-p PASSWORD`, `--p PASSWORD` — Specify the iDRAC access user password Default = *calvin*.
      - `-n {1,10,25,40,50,100}`, `--nics {1,10,25,40,50,100}` — Specify the link speed of the provisioning network interfaces in gigabits per second (Gbps). Default = *1*.
5. When discovery is complete, examine *~/instackenv.json* to verify that it contains an entry for every cluster node to be provisioned, and no other nodes:

   ```
   $ less ~/instackenv.json
   ```

   a. If `discover_nodes.py` cannot identify a provisioning network interface for a node(s), the node's entry in *~/instackenv.json* will indicate that; the value for its `mac` attribute will be `"FIXME and rerun discover_nodes"`.

      a. Ensure that the interface(s) are up and provisioned correctly.
      b. Then rerun `discover_nodes.py`.
6. Load the discovered nodes into RHEL OSP Director:

   ```
   $ openstack baremetal import --json ~/instackenv.json
   ```
7. Power off the nodes and set them to PXE boot:

   ```
   $ ~/pilot/prep_overcloud_nodes.py
   ```
8. Assign the kernel and ramdisk to the nodes:

   ```
   $ openstack baremetal configure boot
   ```

9. Launch introspection of the nodes:

```
$ openstack baremetal introspection bulk start
```

Note: Node introspection can take approximately 10 minutes to complete.

Assigning Node Roles

Before deployment, nodes are assigned the roles that they will perform. This in turn causes the appropriate BIOS and RAID configuration to occur on the nodes prior to provisioning.

To assign the roles to the nodes in RHEL OSP Director:

1. Make a list of the IP addresses of the iDRAC interfaces on all nodes.
2. Decide what role each node will perform. Node roles include:
- controller
- compute
- storage

3. Log into the Director Node.
4. Navigate to the *~/pilot* directory:

```
cd ~/pilot
```

5. For each node, run the following command:

```
./assign_role.py <IP> <role>
```

a. `<IP>` is the *IP address of the node's iDRAC*.
b. `<role>` is the role that the node will perform (*controller*, *compute*, or *storage*).

a. To place a given node role on a specific server, specify the node role with an index number as follows. This example places the *controller-0* role on the server *192.168.110.125*:

```
~/pilot/assign_role.py 192.168.110.125 controller-0
```

Note: The indices must start at 0 for each role, and must increment by 1 for each node in that role. For example, *controller-0*, *controller-1*, *controller-2*, *compute-0*, *compute-1*, etc. Node placement is performed **for all nodes or none**; if you specify an index for one node, you must specify indices for all nodes.

Note: When assigning the role of *storage* to a node, the output will display the size of the OS drive (the size of the RAID configured for the Flex Bay drives). This is an indicator to the admin that the assignment was correct.

Configuring Node Interfaces

At a high level, configuring nodes network interfaces in the Dell EMC Red Hat OpenStack NFV Solution consists of editing a small set of YAML files so that their networking parameters conform to your environment.

The following sections present the networking information you can change:

- *Network Environment Parameters* on page 34
- *Controller Node Bond and VLAN Parameters* on page 35
- *Compute Node Bond and VLAN Parameters* on page 36

- *Storage Node Bond and VLAN Parameters* on page 37

Network Environment Parameters

To configure network environment parameters:

1. On the RHEL OSP Director node, navigate to the *~/pilot/templates/* directory.
2. Open the *network-environment.yaml* file in an editor.
3. Search for the term, *CHANGEME*, to go to the lines in which changes can be made.
4. Make changes, as required, to the parameters listed in *Table 6: network-environment.yaml File Parameters* on page 34.

Note: Each parameter has a default value, which may or may not require a change depending upon your environment.

Table 6: network-environment.yaml File Parameters

Parameter Name	Default Value	Description
ManagementNetCidr	192.168.110.0/24	Classless Inter-Domain Routing (CIDR) block for the Management network
InternalApiNetCidr	192.168.140.0/24	CIDR block for the Private API network.
TenantNetCidr	192.168.130.0/24	CIDR block for the Tenant network. Not used unless you wish to configure Generic Routing Encapsulation (GRE) or VXLAN networks
StorageNetCidr	192.168.170.0/24	CIDR block for the Storage network.
StorageMgmtNetCidr	192.168.180.0/24	CIDR block for the Storage Clustering network.
ExternalNetCidr	192.168.190.0/24	CIDR block for the External network.
InternalApiAllocationPools	`[{'start': '192.168.140.20', 'end': '192.168.140.200'}]`	IP address range for the Private API network.
TenantAllocationPools	`[{'start': '192.168.130.20', 'end': '192.168.130.200'}]`	IP address range for the Tenant network. Not used unless you wish to configure Generic Routing Encapsulation (GRE) or VXLAN networks.
StorageAllocationPools	`[{'start': '192.168.170.20', 'end': '192.168.170.200'}]`	IP address range for the Storage network.
StorageMgmtAllocationPools	`[{'start': '192.168.180.20', 'end': '192.168.180.200'}]`	IP address range for the Storage Clustering. network

Parameter Name	Default Value	Description
ExternalAllocationPools	`[{'start': '192.168.190.20', 'end': '192.168.190.120'}]`	IP address range for the External network.
ExternalInterfaceDefaultRoute	192.168.190.1	Router gateway on the External network.
ProvisioningNetworkGateway	192.168.120.1	The IP address of the gateway on the Provisioning network, which allows access to the Management network.
ControlPlaneDefaultRoute	192.168.120.123	Router gateway on the provisioning network (or Undercloud IP address).
ControlPlaneSubnetCidr	24	CIDR of the control plane network.
EC2MetadataIp	192.168.120.123	IP address of the Undercloud.
DnsServers	`["8.8.8.8", "8.8.4.4"]`	DNS servers for the Overcloud nodes to use (maximum 2). **Note:** This list should include the DNS server passed to the `install-director.sh` command.
InternalApiNetworkVlanID	140	VLAN ID of the Private API network.
StorageNetworkVlanID	170	VLAN ID of the Storage network.
StorageMgmtNetworkVlanID	180	VLAN ID of the Storage Clustering network.
TenantNetworkVlanID	130	VLAN ID of the Tenant network. Not used unless you wish to configure Generic Routing Encapsulation (GRE) or VXLAN networks.
ExternalNetworkVlanID	190	VLAN ID of the External network.
BondInterfaceOptions	`"mode=802.3ad miimon=100"`	Bonding mode for all nodes.[1]
NeutronExternalNetworkBridge	`""`	Empty string for External VLAN, or *br-ex* if on the native VLAN.

Controller Node Bond and VLAN Parameters

To configure Controller node bond and VLAN parameters:

1. On the RHEL OSP Director node, navigate to the *~/pilot/templates/nic-configs* directory.
2. Open the *controller.yaml* file in an editor.
3. Search for the term, *CHANGEME*, to go to the lines in which changes can be made.

[1] If you wish to use different bonding modes for different roles, delete the *BondInterfaceOptions*, and specify the bonding options per role type in the *nic-configs* files.

4. Make changes, as required, to the parameters listed in *Table 7: controller.yaml File Parameters* on page 36.

Note: Each parameter has a default value, which may or may not require a change depending upon your environment.

Table 7: controller.yaml File Parameters

Parameter Name	Default Value	Description
BondInterfaceOptions	" "	Interface bonding mode. For example, `lacp=active` and/or `bond_mode=balance-slb`.
Provisioning Interface Name	em3	Name of the provisioning network interface.
bond0	em1	First interface to include in bond0.
	p3p1	Second interface to include in bond0.
bond1	em2	First interface to include in bond1.
	p3p2	Second interface to include in bond1.

The Controller NIC template assumes that the iDRACs are on the Management network. If the iDRACs are on the Provisioning network, then no routing is required; you must perform the following procedure to remove the associated route.

1. Remove the route that references the *ManagementNetCidr* parameter from the `routes:` section, indicated by the bold italicized example below:

```
routes:
 -
 ip_netmask: 169.254.169.254/32
 next_hop: {get_param: EC2MetadataIp}
 # The following route is used to route from the provisioning
 # network to the management network.  If the iDRACs are on the
 # provisioning network, then remove this route.
 -
   ip_netmask: {get_param: ManagementNetCidr}
   next_hop: {get_param: ProvisioningNetworkGateway}
```

Compute Node Bond and VLAN Parameters

To configure Compute node bond and VLAN parameters:

1. On the RHEL OSP Director node, navigate to the *~/pilot/templates/nic-configs* directory.
2. Open the *compute.yaml* file in an editor.
3. Search for the term, *CHANGEME*, to go to the lines in which changes can be made.
4. Make changes, as required, to the parameters listed in *Table 8: compute.yaml File Parameters* on page 37.

Note: Each parameter has a default value, which may or may not require a change depending upon your environment.

Table 8: compute.yaml File Parameters

Parameter Name	Default Value	Description
BondInterfaceOptions	" "	Interface bonding mode. For example, `lacp=active` and/or `bond_mode=balance-slb`.
Provisioning Interface Name	em3	Name of the provisioning network interface.
bond0	em1	First interface to include in bond0.
	p3p1	Second interface to include in bond0.
bond1	em2	First interface to include in bond1.
	p3p2	Second interface to include in bond1.

Storage Node Bond and VLAN Parameters

To configure Storage node bond and VLAN parameters:

1. On the RHEL OSP Director node, navigate to the *~/pilot/templates/nic-configs* directory.
2. Open the *ceph-storage.yaml* file in an editor.
3. Search for the term, *CHANGEME*, to go to the lines in which changes can be made.
4. Make changes, as required, to the parameters listed in *Table 9: ceph-storage.yaml File Parameters* on page 37.

Note: Each parameter has a default value, which may or may not require a change depending upon your environment.

Table 9: ceph-storage.yaml File Parameters

Parameter Name	Default Value	Description
BondInterfaceOptions	" "	Interface bonding mode. For example, `lacp=active` and/or `bond_mode=balance-slb`.
Provisioning Interface Name	em3	Name of the provisioning network interface.
bond0	em1	First interface to include in bond0.
	p2p1	Second interface to include in bond0.
bond1	em2	First interface to include in bond1.
	p2p2	Second interface to include in bond1.

Configuring Dell Storage

Dell Storage PS Series and SC Series storage servers can be integrated into the Dell EMC Red Hat OpenStack NFV Solution as backends for OpenStack Block Storage (Cinder). This topic describes the following prerequisites and required parameters:

- *PS Series Configuration (Optional)* on page 38
- *SC Series Configuration (Optional)* on page 39

PS Series Configuration (Optional)

These options apply if the Dell Storage PS Series is included in the Dell EMC Red Hat OpenStack NFV Solution as one of the Cinder backends. Requirements for Dell Storage PS Series Configuration include:

- Configure the Dell Storage PS Series according to the Dell EMC Red Hat OpenStack NFV Solution Reference Architecture.
- The Dell Storage PS Series should have access to the Storage Network VLAN. The Controller nodes and the Compute nodes use the Storage Network VLAN to interact with the Dell Storage PS Series through the iSCSI OpenStack driver.

Note: For more information about the OpenStack driver configuration or post-deployment configuration, refer to the *OpenStack Volume Driver Documentation*.

To configure the Dell Storage PS Series as storage backend(s) along with Red Hat Ceph Storage, see:

- *Single PS Series Backend* on page 38
- *Multiple PS Series Backends* on page 38
- *PS Series Configuration Parameters* on page 39

Single PS Series Backend

To configure a single Dell Storage PS Series server as a storage backend along with Red Hat Ceph Storage:

1. On the Director Node, open the *~/pilot/templates/dell-cinder-backends.yaml* file in a text editor.
2. Change any applicable values listed in *Table 10: PS Series Parameters* on page 39.

Multiple PS Series Backends

To configure multiple Dell Storage PS Series servers as storage backends along with Red Hat Ceph Storage:

1. On the Director Node, open the *~/pilot/templates/dell-cinder-backends.yaml* file in a text editor.
2. Make a copy of the configuration section for each Dell Storage PS Series backend.
3. In each new section, change any applicable parameter values for the backends as listed in *Table 10: PS Series Parameters* on page 39.
   a. Assign a unique section name for each backend (e.g., *eqlx1*, *eqlx2*, etc.).
   b. Assign a unique name for the *volume_backend_name*.
   c. List the section names in the *cinder_user_enabled_backends* array with the other enabled Dell Storage PS Series backends.

The configuration example below is for two Dell Storage PS Series backend arrays:

```
...
#EQLX
#Backend1
eqlx1/volume_backend_name:
  value: eqlx1
```

```
eqlx1/volume_driver:
  value: cinder.volume.drivers.eqlx.DellEQLSanISCSIDriver
eqlx1/san_ip:
  ...
#Backend2
eqlx2/volume_backend_name:
  value: eqlx2
eqlx2/volume_driver:
  value: cinder.volume.drivers.eqlx.DellEQLSanISCSIDriver
eqlx2/san_ip:
 ...
#EQLX-END
 ...
cinder_user_enabled_backends: ['eqlx1','eqlx2',...]
```

PS Series Configuration Parameters

PS Series configuration parameters include:

Table 10: PS Series Parameters

Parameter Name	Description
eqlx_san_ip	String containing SAN IP address
eql_san_login	String containing SAN login ID
eqlx_san_password	String containing SAN IP password
eqlx_san_thin_provision	Boolean: if set to *true*, thin provisioning enabled
eqlx_group_name	String containing Storage Array Group Name
eqlx_pool	String containing Storage Pool Name
eqlx_use_chap	Boolean: if set to *true*, CHAP authentication enabled
eqlx_chap_login	String containing CHAP Account Name
eqlx_chap_password	String containing CHAP Account Password
cinder_user_enabled_backends	Array of strings containing enabled backends. List valid backends only

Note: For more information about using Dell Storage PS Series in a Red Hat OpenStack Platform Overcloud, see *https://access.redhat.com/documentation/en/red-hat-openstack-platform/version-8/dell-equallogic-back-end-guide/*.

SC Series Configuration (Optional)

These options apply if the Dell Storage SC Series is included in the Dell EMC Red Hat OpenStack NFV Solution as one of the Cinder backends. Requirements for Dell Storage SC Series Configuration include:

- Dell Storage Center(s) with Dell Storage Enterprise Manager Platform.
- Documents and software can be downloaded from *https://portal.compellent.com/*.
- The Dell Storage SC Series is configured with the Dell Storage Enterprise Manager Node, according to the Dell EMC Red Hat OpenStack NFV Solution Reference Architecture.
- The Dell Storage Enterprise Manager Node should have access to the Storage Network VLAN and External Network. The Controller nodes will use the Storage Network VLAN to access the Dell Storage Enterprise Manager Node for management of volumes and snapshots.

- Dell Storage Center should have access to the Storage Network VLAN. The Compute nodes must have access to the Dell Storage SC Series through the Dell Storage Center iSCSI ports, in order for the iSCSI driver on that node to interact with the volumes associated with Virtual Machines hosted by that node.
- Dell Storage Center Front-End Connectivity Mode should be configured to use virtual port mode.

Note: For more information about the OpenStack driver configuration or post-deployment configuration, refer to the *OpenStack Volume Driver Documentation*.

To configure the Dell Storage SC Series as storage backend(s) along with Red Hat Ceph Storage, see:

- *Single SC Series Backend* on page 40
- *Multiple SC Series Backends* on page 40
- *PS Series Configuration Parameters* on page 39

Single SC Series Backend

To configure a single Dell Storage SC Series server as a storage backend along with Red Hat Ceph Storage:

1. On the Director Node, open the *~/pilot/templates/dell-cinder-backends.yaml* file in a text editor
2. Change any applicable values listed in *Table 11: SC Series Parameters* on page 41.

Multiple SC Series Backends

To configure multiple Dell Storage SC Series servers as storage backends along with Red Hat Ceph Storage:

1. On the Director Node, open the *~/pilot/templates/dell-cinder-backends.yaml* file in a text editor.
2. Make a copy of the configuration section for each Dell Storage SC Series backend.
3. Change any applicable parameter values for the backends as listed in *Table 11: SC Series Parameters* on page 41.
   a. Assign a unique section name for each backend (e.g., *dellsc1*, *dellsc2*, etc.).
   b. Assign a unique name for the *volume_backend_name*.
   c. List the section names in the *cinder_user_enabled_backends* array with the other enabled Dell Storage SC Series backends.

The configuration example below is for two Dell Storage SC Series backend arrays:

```
        ...
        #DELLSC
        #Backend1
        dellsc1/volume_backend_name:
        value: dellsc1
        dellsc1/volume_driver:
        value:
cinder.volume.drivers.dell.dell_storagecenter_iscsi.DellStorageCenterISCSIDriver
        dellsc1/san_ip:
        ...
        #Backend2
        dellsc1/volume_backend_name:
        value: dellsc2
        dellsc1/volume_driver:
        value:
cinder.volume.drivers.dell.dell_storagecenter_iscsi.DellStorageCenterISCSIDriver
        dellsc1/san_ip:
        ...
        #DELLSC-END
        ...
        cinder_user_enabled_backends: [...,'dellsc1','dellsc1']
```

SC Series Configuration Parameters

SC Series configuration parameters include:

Table 11: SC Series Parameters

Parameter Name	Description
dellsc_san_ip	String containing Enterprise Manager IP address
dellsc_san_login	String containing Enterprise Manager login ID
dellsc_san_password	String containing Enterprise Manager password
dellsc_iscsi_port	String containing iSCSI API port
dellsc_iscsi_ip_address	String containing virtual port iSCSI IP address
dellsc_sc_api_port	String containing Enterprise Manager API port
dellsc_sc_ssn	String containing the Storage Center serial numbers to use
dellsc_server_folder	String containing server folder in which to place new server definition
dellsc_volume_folder	String containing volume folder in which to place new volume
cinder_user_enabled_backends	Array of strings containing enabled backends. List valid backends only

Note: For more information about using Dell Storage SC Series in a Red Hat OpenStack Platform Overcloud, see *https://access.redhat.com/documentation/en/red-hat-openstack-platform/8/dell-storage-center-back-end-guide/dell-storage-center-back-end-guide*.

Configuring Red Hat Ceph Storage

Red Hat Ceph Storage interacts with physical or logical storage units, or Object Storage Devices (OSDs). This topic describes how multiple OSDs are configured across the Red Hat Ceph Storage nodes.

Some points to consider:

- The RHOSP Controllers are configured to be highly available (HA).
- The RHOSP Storage back end is configured to use Red Hat Ceph Storage via the RBD client library.
- Red Hat Ceph Storage pools will be created for RHOSP images and volumes.
- All node access will be controlled via the `cephx` authentication protocol.

Red Hat Ceph Storage Configuration Parameters

To configure Red Hat Ceph Storage parameters:

1. On the RHEL OSP Director node, navigate to the *~/pilot/templates/overrides/puppet/hieradata* directory.
2. Open the *ceph.yaml* file in an editor.
3. Search for the term, *CHANGEME*, to go to the lines in which changes can be made.
4. Make changes, as required, to the parameters listed in *Table 12: ceph.yaml File Parameters* on page 42.

   Note: Each parameter has a default value, which may or may not require a change depending upon your environment.

Table 12: ceph.yaml File Parameters

Parameter	Default Value	Description
osd_pool_default_pg_num	256	Default number of placement groups in the pool
osd_pool_default_pgp_num	256	Default number of placement groups for placement in the pool
osds	See *OSD Configuration Parameters* on page 42.	List of drives to be used as OSDs and journals
ceph_pool_pgs: volumes	1024	Each pool's *pg_num* and *pgp_num* values, using *http://ceph.com/pgcalc* for guidance.
ceph_pool_pgs: vms	256	
ceph_pool_pgs: images	256	
ceph_pool_pgs: .rgw.buckets	512	

OSD Configuration Parameters

Table 13: OSD Configuration Parameters on page 42 displays the default Dell EMC Red Hat OpenStack NFV Solution's OSDs layout. If your configuration differs from that displayed below, modify the *osds* parameter in *ceph.yaml* according to the guidance contained within the *ceph.yaml* comments.

For the 3.5" drive configuration, several journal devices are shared between multiple OSDs. The OSDs sharing a journal device should be placed on the SSD drives. The data devices, where the data is stored on HDD drives, are associated with journal devices and are also shared between the OSDs. For the 2.5" drive configuration, no dedicated journal devices are used.

Table 13: OSD Configuration Parameters

OSD Host	Journal Device	Data Devices
3.5" R730xd	/dev/sda	/dev/sdd, /dev/sde, /dev/sdf, /dev/sdg
	/dev/sdb	/dev/sdh, /dev/sdi, /dev/sdj, /dev/sdk
	/dev/sdc	/dev/sdl, /dev/sdm, /dev/sdn, /dev/sdo
2.5 " R730xd	N/A	/dev/sda, /dev/sdb, /dev/sdc, /dev/sdd, /dev/sde /dev/sdf, /dev/sdg, /dev/sdh, /dev/sdi, /dev/shj, /dev/sdk, /dev/sdl, /dev/sdm, /dev/sdn, /dev/sdo, /dev/sdp, /dev/sdq, /dev/sdr, /dev/sds, /dev/sdt, /dev/sdu, /dev/sdv, /dev/sdw, /dev/sdx

Caution: Any existing data on the drives will be destroyed upon Overcloud deployment.

Configuring Local Ephemeral Storage

Every VM created in OpenStack has a virtual disk that the operating system is installed upon. Ephemeral Storage refers to this OS disk and where it is stored. An ephemeral disk is created when a VM is created, and destroyed when a VM is destroyed.

By default the Dell EMC Red Hat OpenStack NFV Solution stores ephemeral disks in Red Hat Ceph Storage. You can store the ephemeral disk on the Compute Node that hosts the VM; this concept is known as Local Ephemeral Storage.

Local Ephemeral Storage Configuration Parameters

To configure the Dell EMC Red Hat OpenStack NFV Solution to use local ephemeral storage instead of Red Hat Ceph Storage:

1. Open the *~/pilot/templates/dell-environment.yaml* file in an editor.
2. Change the *NovaEnableRbdBackend* parameter from *true* to *false*.
3. Save the file.

Note: Local ephemeral storage currently supports the images_type values *rbd* (shared storage like Red Hat Ceph Storage) or *default* (local Compute node storage). If you choose *default*, then the Instance HA feature is not supported.

You can now proceed to *Configuring the Overcloud Domain Name* on page 43.

Configuring the Overcloud Domain Name

To configure the domain name for Overcloud nodes:

1. Edit the *~/pilot/templates/dell-environment.yaml* file.
2. Search for the *CloudDomain* parameter.
3. Change the value of the *CloudDomain* parameter to the domain name you want to use for the Overcloud.
4. Save the file.

You can now proceed to *Deploying the Overcloud* on page 43.

Deploying the Overcloud

Once you have completed editing the environment and nodes networking YAML configuration files, you can begin deploying the Overcloud.

To deploy the Overcloud:

1. Remove all removable media, such as USB drives.
2. Log onto the RHEL OSP Director node using the user name and password specified in *director.cfg*.
3. Deploy the Overcloud by executing the following command (optional arguments are enclosed in square brackets):

```
$ cd ~/pilot
$ ./deploy-overcloud.py [-h][--controllers <NUM_CONTROLLERS>] \
--computes <NUM_COMPUTES> --storage <NUM_STORAGE> \
--vlans <VLAN_RANGE> [--ntp <NTP_SERVER_FQDN>] [--timeout <MINUTES>] \
[--overcloud_name <OVERCLOUD_NAME>] [--enable_eqlx] [--enable_dellsc] \
[--static_ips] [--static_vips] [--node_placement]
```

   a. <vlanRange> is of this format: <startingVlanNumber>:<endingVlanNumber>. For example, --vlans 201:219.
   b. The deployment specifies *3* Controller nodes. Therefore, you must have assigned the *controller* role using the assign_role.py command to three or more nodes. See *Assigning Node Roles* on page 33.
   c. If you specify *2* for the number of Compute nodes, then you must have assigned the *compute* role using the assign_role.py command to two or more nodes.
   d. If you specify *3* for the number of Storage nodes, then you must have assigned the *storage* role using the assign_role.py command to three or more nodes.

4. Optionally, specify whether you are using Dell Storage PS Series or SC Series as storage backends for OpenStack Block Storage (Cinder) by adding `--enable_eqlx` or `--enable_dellsc` to the `deploy-overcloud.py` command.
5. Optionally, specify the assignment of Overcloud nodes' static IP addresses, as described in *Static IP Addresses (Optional)*, by adding `--static_ips` to the `deploy-overcloud.py` command.
6. Optionally, specify the assignment of Overcloud nodes' static VIP addresses, as described in *Static Virtual IP Addresses (Optional)* , by adding `--static_vips` to the `deploy-overcloud.py` command.
7. Optionally, specify node placement, as described in *Assigning Node Roles* on page 33, by adding `--node_placement` to the `deploy-overcloud.py` command.

   Note: The `deploy-overcloud.py` command will take some time to complete, since it is installing all of the nodes in the OpenStack cluster. It can take approximately 60-180 minutes to complete depending on the options used above.
8. Once the Overcloud has been successfully deployed, use the `agent_fencing.sh` command to enable fencing, which allows the system to detect and correct problematic nodes in the Overcloud. It can also be used to disable fencing. The syntax is:

```
$ agent_fencing.sh <idrac_user> <idrac_password> < [ enable | disable ] >
```

   a. Example to enable fencing:

```
$ ~/pilot/agent_fencing.sh root calvin enable
```

   b. Example to disable fencing:

```
$ ~/pilot/agent_fencing.sh root calvin disable
```

   c. Example output from pcs status after fencing has been enabled.:

```
# [heat-admin@controller-0 ~]$ sudo pcs status
controller-1-ipmi  (stonith:fence_ipmilan):  Started controller-2
controller-0-ipmi  (stonith:fence_ipmilan):  Started controller-1
controller-2-ipmi  (stonith:fence_ipmilan):  Started controller-0
```

You can now proceed to *Registering Overcloud Nodes with CDN* on page 44.

Registering Overcloud Nodes with CDN

If your environment requires the ability to update the Overcloud nodes' software packages, you must first register them with the Red Hat Content Delivery Network (CDN). Overcloud nodes include:

- Controller Nodes
- Compute Nodes
- Red Hat Ceph Storage Nodes

The RHEL repositories listed in *Table 14: Overcloud Nodes CDN Registration Parameters* on page 45 are those necessary for updating the RPMs that are installed on the nodes, if required in your environment. Although permissible, you do not need to modify the repository values.

Registering Overcloud Nodes

To register the Overcloud nodes:

1. Edit the *~/pilot/subscription.json* file, changing any applicable values listed in *Table 14: Overcloud Nodes CDN Registration Parameters* on page 45.

a. If you would rather not put CDN and/or proxy credentials into the JSON file, you can alternatively pass them in on the command line. See *Passing CDN and Proxy Credentials via the Command Line* on page 47.

b. If you are using a proxy, be sure to remove the leading underscore from `"_proxy_credentials"` in the *~/pilot/subscription.json* file. See *Example subscription.json File* on page 46.

Table 14: Overcloud Nodes CDN Registration Parameters

Parameter	Value
CDN Credentials	
cdn_username	CDN username
cdn_password	CDN password
Proxy Credentials	
proxy_url	Proxy hostname:port
proxy_username	Proxy username
proxy_password	Proxy password
Roles	
control	
pool_ids	Pool ID (requires a Red Hat OpenStack Platform subscription)
repos	RHEL repositories: • rhel-7-server-rpms • rhel-7-server-extras-rpms • rhel-7-server-openstack-9-rpms • rhel-7-server-openstack-9-director-rpms • rhel-7-server-rh-common-rpms • rhel-ha-for-rhel-7-server-rpms • rhel-7-server-rhceph-1.3-tools-rpms • rhel-7-server-rhceph-1.3-mon-rpms • rhel-7-server-rhceph-1.3-osd-rpms
compute	
pool_ids	Pool ID (requires a Red Hat OpenStack Platform subscription)
repos	RHEL repositories: • rhel-7-server-rpms • rhel-7-server-extras-rpms • rhel-7-server-openstack-9-rpms • rhel-7-server-openstack-9-director-rpms • rhel-7-server-rh-common-rpms • rhel-ha-for-rhel-7-server-rpms
ceph-storage	

Parameter	Value
pool_ids	Pool ID (requires a Red Hat OpenStack Platform subscription and a Red Hat Ceph Storage subscription)
repos	RHEL repositories: • rhel-7-server-rpms • rhel-7-server-extras-rpms • rhel-7-server-openstack-9-rpms • rhel-7-server-openstack-9-director-rpms • rhel-7-server-rh-common-rpms • rhel-ha-for-rhel-7-server-rpms • rhel-7-server-rhceph-1.3-tools-rpms • rhel-7-server-rhceph-1.3-osd-rpms • rhel-7-server-rhceph-1.3-osd-rpms

2. Navigate to the *~/pilot* directory:

```
# cd ~/pilot
```

3. Register the Overcloud nodes by executing the following command:

```
# ./register_overcloud.py
```

Example subscription.json File

```
{
    "cdn_credentials": {
        "cdn_username": "username",
        "cdn_password": "password"
    },
    "_comment": [ "If using a proxy, remove the leading underscore from",
                  "_proxy_credentials below and fill in the following proxy",
                  "information." ],
    "_proxy_credentials": {
        "proxy_url": "hostname:port",
        "proxy_username": "username",
        "proxy_password": "password"
    },
    "roles": {
        "control": {
            "pool_ids": [ "openstack_pool_id",
                          "ceph_pool_id" ],
            "repos": [ "rhel-7-server-rpms",
                       "rhel-7-server-extras-rpms",
                       "rhel-7-server-openstack-9-rpms",
                       "rhel-7-server-openstack-9-director-rpms",
                       "rhel-7-server-rh-common-rpms",
                       "rhel-ha-for-rhel-7-server-rpms",
                       "rhel-7-server-rhceph-1.3-tools-rpms",
                       "rhel-7-server-rhceph-1.3-mon-rpms",
                       "rhel-7-server-rhceph-1.3-osd-rpms" ]
        },
        "compute": {
            "pool_ids": [ "openstack_pool_id" ],
            "repos": [ "rhel-7-server-rpms",
                       "rhel-7-server-extras-rpms",
                       "rhel-7-server-openstack-9-rpms",
                       "rhel-7-server-openstack-9-director-rpms",
```

```
                       "rhel-7-server-rh-common-rpms",
                       "rhel-ha-for-rhel-7-server-rpms" ]
        },
        "ceph-storage": {
            "pool_ids": [ "openstack_pool_id",
                          "ceph_pool_id" ],
            "repos": [ "rhel-7-server-rpms",
                       "rhel-7-server-extras-rpms",
                       "rhel-7-server-openstack-9-rpms",
                       "rhel-7-server-openstack-9-director-rpms",
                       "rhel-7-server-rh-common-rpms",
                       "rhel-ha-for-rhel-7-server-rpms",
                       "rhel-7-server-rhceph-1.3-tools-rpms",
                       "rhel-7-server-rhceph-1.3-mon-rpms",
                       "rhel-7-server-rhceph-1.3-osd-rpms"]
        }
    }
}
```

Passing CDN and Proxy Credentials via the Command Line

To pass CDN and/or proxy credentials via the command line:

1. Navigate to the ~/pilot directory:

```
# cd ~/pilot
```

2. Register the Overcloud nodes by executing the following command:

```
# ./register_overcloud.py
```

   a. Optional arguments to `register_overcloud.py` include:
      - `--cdn_username <username>`
      - `--cdn_password <password>`
      - `--proxy_url <host>:<port>`
      - `--proxy_username <username>`
      - `--proxy_password <password>`

Troubleshooting Node Provisioning

If you encounter difficulties during the node provisioning process, you can find troubleshooting tips and information at:

- *https://access.redhat.com/documentation/en/red-hat-openstack-platform/8/director-installation-and-usage/chapter-11-troubleshooting-director-issues*

Chapter 7

Red Hat Ceph Storage Integration

Topics:

This topic provides instructions for installing and configuring Calamari.

Installing and Configuring Calamari

Calamari is the management and monitoring service for Red Hat Ceph Storage. This section describes how to how to install Calamari to the previously-deployed Red Hat Ceph Storage Admin Node (see *Red Hat Ceph Storage Admin Node Deployment* on page 27).

Note: The Ceph Object Gateway is automatically installed and configured by Heat templates.

Configuring the Red Hat Ceph Storage Admin Node and Calamari Client Nodes

To configure the Red Hat Ceph Storage Admin Node and Calamari Client nodes:

1. Log onto the Director Node as the *admin_user*, or the user as configured in *Table 3: Director Node Configuration Parameters* on page 22 and change into the pilot directory:

```
# cd ~/pilot
```

2. Execute the `config_calamari_nodes.sh` script, which:

- Installs SSH keys
- Configures the hosts files, salt files, and Calamari files
- Restarts services
- Executes the Calamari initialization scripts that enables you to use Calamari services

```
# ./config_calamari_nodes.sh <calamari_node_ip> <root_password>
```

When this script has completed the configuration process:

3. Use a web browser to connect to *http://<calamari_node_ip>*.
4. Log into Calamari as the *root* user, with the password that you supplied to the `config_calamari_nodes.sh` script.

Appendix A

File References

Topics:

- *Solution Files*

This appendix lists documents and script archives that are required to install and deploy the Dell EMC Red Hat OpenStack Cloud Solution. Please contact your Dell EMC representative for copies if required.

Solution Files

Dell EMC Red Hat OpenStack Cloud Solution files include:

- *Dell-OSP-6.0.0.tgz* - Contains all solution documentation and scripts
- *Dell EMC Red Hat OpenStack NFV Solution BOM Guide*
- *Dell EMC Red Hat OpenStack NFV Solution Deployment Guide*
- *Dell EMC Red Hat OpenStack NFV Solution Hardware Deployment Guide*
- *Dell EMC Red Hat OpenStack NFV Solution Workbook*
- *Dell EMC Red Hat OpenStack NFV Solution Reference Architecture*
- *Dell EMC Red Hat OpenStack NFV Solution Release Notes*
- *dell-mgmt-node.tgz* - Contains helper scripts for SAH node installation
- *dell-pilot-deploy.tgz* - Contains helper scripts for RHOSP installation

Appendix B

Updating RPMs on Version Locked Nodes

Topics:

- *Updating the RPMs*
- *Director Node Version Locking*

At a high level, updating RPMs on a version locked node (RHEL OSP Director Node or Red Hat Ceph Storage Admin Node):

1. Identifies the RPMs that need to be updated.
2. Removes them from the version lock list on that node.
3. Updates the RPMs.
4. Adds the updated RPMs back into the version lock list.

This topic provides detailed information to perform those steps.

Updating the RPMs

To update the RPMs:

Note: All of the following commands should be run as the **root** user.

1. Produce a list of RPMs that are version locked on a node:
   a. Login to a node.
   b. Execute the following command to produce a list of RPMs that are version locked:

```
# yum versionlock list
```

2. Identify the RPMs to be updated from the output of the above command.
3. Remove the selected RPMs from the version lock list:
   a. Execute the following command, substituting *VLockListEntry* with an RPM name from the output of the `versionlock list` command above:

      Note: The *VLockListEntry* must **exactly** match an RPM name in the output of the `yum versionlock list` command.

```
# yum versionlock delete VLockListEntry
```

   b. Repeat for each RPM.
4. Update each of the selected RPMs:
   a. Execute the following command for an RPM, substituting *RPMNameWithoutVersion* with the name of the RPM **without** the version number:

```
# yum update RPMNameWithoutVersion
```

   b. Repeat for each subsequent RPM
5. Add each of the selected RPMs back into the version lock list:
   a. Execute the following command, again substituting *RPMNameWithoutVersion* with the name of the RPM **without** the version number:

```
# yum versionlock add RPMNameWithoutVersion
```

Director Node Version Locking

The Dell EMC Red Hat OpenStack NFV Solution version 6.0 includes the following files, in the version lock list file, that are used to install the base system and the Undercloud:

- openstack-ironic-api-4.2.2-4.el7ost.*
- openstack-ironic-common-4.2.2-4.el7ost.*
- openstack-ironic-conductor-4.2.2-4.el7ost.*

The Solution requires a set of custom RPMs, that are not yet available on the CDN, to be installed during the deployment process. These RPMs allow upgrading from the versions above, and fix a known Ironic issue:

- openstack-ironic-api-4.2.3-1.el7ost.noarch.rpm
- openstack-ironic-common-4.2.3-1.el7ost.noarch.rpm
- openstack-ironic-conductor-4.2.3-1.el7ost.noarch.rpm

These RPMs are automatically updated when you execute the `install-director.sh` installation script. The lock file on the Director Node is also updated with those versions.

Appendix C

OpenStack Operations Functional Test (Optional)

Topics:

- *Creating Neutron Networks in the Overcloud*
- *Manual RHOSP Test*
- *Scripted RHOSP Sanity Test*

This is an optional section. It includes instructions for creating the networks and testing a majority of your RHOSP environment using Glance configured with Red Hat Ceph Storage, PS Series, SC Series or any backend. These command line instructions are working examples that are found on the OpenStack website (*http://docs.openstack.org/juno/install-guide/install/apt/content/neutron_initial-tenant-network.html*).

Creating Neutron Networks in the Overcloud

The following example commands create the required tenant and public networks, and their network interfaces. You must complete them prior to creating instances and volumes, and testing of the functional operations of OpenStack.

Note: The following commands and those in the following section should be executed on the Director Node.

1. Log into the Director Node using the user name and password specified when creating the node and source the *overcloudrc* file, or the name of the stack defined when deploying the overcloud :

```
# cd ~/
# source overcloudrc
```

2. Create the tenant network by executing the following commands:

Note: Replace *tenant_network_name* with your desired values. (e.g., `neutron net-create tenant_net --shared`).

```
# neutron net-create <tenant_network_name> --shared
```

3. Create the tenant subnet on the tenant network:

Note: Replace *tenant_network_name*, *vlan_network*, *vlan_name* and *vlan_gateway* with your desired values (e.g., `neutron subnet-create tenant_net 192.168.201.0/24 --name tenant_201`).

```
# neutron subnet-create <tenant_network_name> <vlan_network> \
--name <vlan_name>
```

4. Create the router:

Note: Replace *tenant_router* with your desired values (e.g., `neutron router-create tenant_201_router`).

```
# neutron router-create <tenant_router>
```

5. Before you add the tenant network interface, you will need the subnets ID. Execute the following command to display them:

```
# neutron net-list
```

The displayed output will be similar to the following (example truncated for brevity):

```
+--------------------------------------+-------------+--------------- ...
| id                                   | name        | subnets        ...
+--------------------------------------+-------------+--------------- ...
| 52411536-ec43-402f-9736-4cabdc8c875d | tenant_net        | 7329d413 ...
| 0af01763-539e-41c7-ac32-abbaa62ee575 | HA network tenant | bdae0b72 ...
+--------------------------------------+-------------+--------------  ...
```

6. Add the tenant network interface between the router and the tenant network:

Note: Replace *tenant_router* and *subnets_id* with your desired values (e.g., `neutron router-interface-add tenant_201_router 7329d413-ac23-56cf-8867-133b5ff8fc12`).

```
# neutron router-interface-add <tenant_router> <subnets_id>
```

7. Create the external network by executing the following commands:

> **Note:** Replace *external_network_name* and *external_vlan_id* with your desired value. (e.g., `neutron net-create public --router:external --provider:network_type vlan --provider:physical_network physext --provider:segmentation_id 191`).

```
# neutron net-create <external_network_name> --router:external \
--provider:network_type vlan --provider:physical_network physext \
--provider:segmentation_id <external_vlan_id>
```

8. Create the external subnet with floating IP addresses on the external network:

> **Note:** Replace *external_subnet_name*, *start_ip*, *end_ip*, *external_network_name*, *external_vlan_network* and *external_gateway* with your desired values (e.g., `neutron subnet-create --name external_sub --allocation-pool start=192.168.191.2,end=192.168.191.30 --disable-dhcp --gateway 192.168.191.254 public 192.168.191.0/24`).

```
# neutron subnet-create --name <external_subnet_name> \
--allocation-pool start=<start_ip>,end=<end_ip> \
--disable-dhcp --gateway <gateway_ip> <external_network_name>
 <external_vlan_network>
```

9. Set the external network gateway for the router:

> **Note:** Replace *tenant_router_name* with the router name *external_nework_name* with the external network name (e.g., `neutron router-gateway-set tenant_201_router public`).

```
# neutron router-gateway-set <tenant_router_name> <external_network_name>
```

Manual RHOSP Test

This example uses the Cirros image to test high-level functional operations of OpenStack.

1. Log into the Director Node using the user name and password specified when creating the node, or the name of the stack defined when deploying the Overcloud.
2. Download the Cirros image:

```
# wget http://download.cirros-cloud.net/0.3.3/cirros-0.3.3-x86_64-disk.img
```

3. Source your Overcloud credentials:

```
# cd ~/
# source <overcloud_name>rc
```

4. Create and upload the Glance image:

```
# glance image-create --name <image_name> --is-public true \
--disk-format format> --container-format bare \
--file <file_path>
```

For example:

```
# glance image-create --name "cirros image" \
--is-public true --disk-format qcow2 --container-format bare \
--file cirros-0.3.3-x86_64-disk.img
```

5. List available images to verify that your image was uploaded successfully:

```
# glance image-list
```

6. To view more detailed information about an image, use the identifier of the image from output of the `glance image-list` command above:

```
# glance image-show <id>
```

7. Launch an instance using the boot image that you uploaded:

   a. Get the ID of the flavor you will use:

   ```
   # nova flavor-list
   ```

   b. Get the image ID:

   ```
   # nova image-list
   ```

   c. Get the tenant network ID:

   ```
   # nova network-list
   ```

   d. Generate a key pair. The command below generates a new key pair; if you try using an existing key pair in the command, it fails.

   Note: `MY_KEY.pem` is an output file created by the `nova keypair-add` command, and will be used later.

   ```
   # nova keypair-add <key_name> > MY_KEY.pem
   ```

   e. Create an instance using the `nova boot` command.

   Note: Change the *IDs* to your IDs from Steps 7a-c, and the *nameofinstance* and the *key_name* from Step 7c:

   ```
   # nova boot --flavor <flavor_id> --key_name <key_name> \
   --image <imageid>  --nic <net-id=<tenantNetID> <nameofinstance>
   ```

   For example:

   ```
   # nova boot --flavor 2 --key_name key_name \
   --image 0bde34f6-fba6-4174-a3ea-ff2a7918de2e \
   --nic net-id=52411536-ec43-402f-9736-4cabdc8c875d  cirros-test
   ```

   f. List the instance you created:

   ```
   # nova list
   ```

8. If you have multiple backends, create a Cinder volume type for each backend. Get the `<volume_backend_name>` from the */etc/cinder/cinder.conf* file on the Controller node.

```
 # cinder type-create <type_name>
 # cinder type-key <type_name> set
 volume_backend_name=<volume_backend_name>
```

For example:

```
# cinder type-create rbd_backend
# cinder type-key rbd_backend set volume_backend_name=tripleo_ceph

# cinder type-create eqlx_backend
# cinder type-key eqlx_backend set volume_backend_name=eqlx
```

```
# cinder type-create dellsc_backend
# cinder type-key dellsc_backend set volume_backend_name=dellsc
```

Multiple PS Series backend example:

```
# cinder type-create eqlx1_backend
# cinder type-key eqlx1_backend set volume_backend_name=eqlx1

# cinder type-create eqlx2_backend
# cinder type-key eqlx2_backend set volume_backend_name=eqlx2
```

9. Create a new volume to test the Cinder volumes:

> **Note:** If you have multiple backends defined, you must append the optional arguments `--volume-type <type-name>` from Step 7 to the command below.

```
# cinder create --display-name <name> <sizeinGB>
```

For example:

```
# cinder create --display-name vol_test1 1
```

a. List the Cinder volumes:

```
# cinder list
```

b. Attach the volume to the instance, specifying the server ID and the volume ID.

> **Note:** Replace the *server_id* with the ID returned from the `nova list` command, and replace the *volume_id* with the ID returned from the `cinder list` command, from the previous steps.

```
# nova volume-attach <server_id> <volume_id> <device>
```

For example:

```
# nova volume-attach 84c6e57d-a6b1-44b6-81eb-fcb36afd31b5 \
573e024d-5235-49ce-8332-be1576d323f8 /dev/vdb
```

10. Access the instance.

a. Find the active Controller by executing the following commands from the Director Node node:

```
# cd ~/
# source stackrc
# nova list (make note of the controllers ips)
# ssh heat-admin@<controller ip>
# sudo -i
# pcs status
```

The displayed output will be similar to the following:

```
Cluster name: tripleo_cluster
Last updated: Wed Apr 6 20:48:10 2016
Last change: Mon Apr 4 18:49:20 2016 by root via cibadmin on overcloud-
controller-1
Stack: corosync
Current DC: overcloud-controller-1 (version 1.1.13-10.el7_2.2-44eb2dd) -
 partition with quorum
3 nodes and 112 resources configured
```

b. Initiate an SSH session to the active Controller, as **heat-admin**.
c. Find the instances by executing the following command:

```
# sudo -i
# ip netns
```

The displayed output will be similar to the following:

```
qrouter-21eba0b0-b849-4083-ac40-44b794744e9f
qdhcp-f4a2c88f-1bc9-4785-b070-cc82d7c334f4
```

d. Access an instance namespace by executing the following command:

```
# ip netns exec <namespace> bash
```

For example:

```
# ip netns exec qdhcp-f4a2c88f-1bc9-4785-b070-cc82d7c334f4 bash
```

e. Verify that the namespace is the desired tenant network, by executing the following command:

```
# ip a
```

The displayed output will be similar to the following:

```
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN
link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
inet 127.0.0.1/8 scope host lo
valid_lft forever preferred_lft forever
inet6 ::1/128 scope host
valid_lft forever preferred_lft forever
19: tap05a22fb4-4f: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc
 noqueue state UNKNOWN
link/ether fa:16:3e:99:b9:88 brd ff:ff:ff:ff:ff:ff
inet 192.168.201.2/24 brd 192.168.201.255 scope global tap05a22fb4-4f ->
 Tenant network
valid_lft forever preferred_lft forever
inet6 fe80::f816:3eff:fe99:b988/64 scope link
valid_lft forever preferred_lft forever
```

f. Ping the IP address of the instance.
g. SSH into the instance, as **root**, using the keypair generated above:

```
# ssh -i MY_KEY.pem root@<ip>
```

11. Format the drive and access it.

a. List storage devices:

```
# fdisk -l
```

b. Format the drive:

```
# mkfs.ext3 /dev/vdb
```

c. Mount the device, access it, and then unmount it:

```
# mkdir ~/mydrive
# mount /dev/vdb ~/mydrive
# cd ~/mydrive
# touch helloworld.txt
# ls
```

```
# umount ~/mydrive
```

Scripted RHOSP Sanity Test

As an alternative to manually testing your deployment script, we provide `sanity_test.sh`, which tests all of the basic functionality outlined in *Creating Neutron Networks in the Overcloud* on page 56 and *Manual RHOSP Test* on page 57.

To run the sanity test script:

1. Log into the Director Node using the user name and password specified when creating the node, or the name of the stack defined when deploying the Overcloud.
2. From your home directory, execute the `sanity_test.sh` script:

```
# cd ~/
# ./pilot/deployment-validation/sanity_test.sh
```

3. If you wish to clean the environment once the `sanity_test.sh` script has run successfully:

```
# cd ~/
# ./pilot/deployment-validation/sanity_test.sh clean
```

Appendix D

Enabling Huge Pages pre-deployment or post-deployment

Topics:

- *Huge Pages Overview*
- *Predeployment*
- *Post deployment - Enabling and Deploying Huge Pages*

This appendix provide instructions for enabling Huge Pages pre-deployment or post-deplyment

Huge Pages Overview

The Dell EMC Red Hat OpenStack NFV Solution provides the ability to enable Huge page support on all Nova compute nodes in the solution.

Hugepages allow the Linux kernel to utilize the multiple page size capabilities of modern hardware architectures. Linux uses pages as the basic unit of memory, where physical memory is partitioned and accessed using the basic page unit. The default page size is 4096 Bytes in the x86 architecture.

This document provides the instructions to configure this feature at the time of deployment (Deployment time enablement (DTE)) or post deployment.

Caution: BEFORE you use this facility - please note that the scripts described in this Appendix WILL REBOOT the compute nodes in the solution several times. You are strongly advised to save all work, stop services, and/or ask users to log out before proceeding any further.

Warning: Please do not reboot any nodes when the optional feature scripts are being used.

Note: This facility allows you to create a new Nova flavor with default values of 4 vCPUs and 4GB RAM. If you would like to enable Huge Pages on a flavor of different size, then create the desired flavor prior to following the steps described below.

Prerequisites

In order to deploy the Dell EMC Red Hat OpenStack NFV Solution on Dell hardware, the following prerequisites must be met:

- Successful hardware setup and configuration, per the *Dell EMC Red Hat OpenStack NFV Solution Hardware Deployment Guide*
- Must have a valid subscription and outbound network access to Red Hat's Content Delivery Network.
- Addition changes to the settings_sample.ini are required for automated deployment see the *Dell EMC Red Hat OpenStack NFV Solution Automated Deployment Guide - Version 6.0* for details.

Predeployment

The Dell EMC Red Hat OpenStack NFV Solution provides provides the ability to enable Huge Page support on all Nova compute nodes during deployment of the solution. This section provides the instructions to enable the feature when the solution is being deployed.

Enabling Huge Pages during Deployment

To enable Huge Pages:

1. Open an SSH session to the SAH node.
2. Log in as the root user.
3. Edit the settings_sample.ini (e.g. acme.ini) file. Change Enable_hugepages to true from the default of false. Change the settings based on the desired huge page size. *flavorname* can be an existing or newly created flavor.
    - For a Huge Page size of 1GB:

```
Enable_hugepages=true
hpsize=1GB
hpgnum=96
hpg_flavor_name=flavor_name
```

- For a Huge Page size of 2 MB:

```
Enable_hugepages=true
hpsize=2MB
hpgnum=49152
hpg_flavor_name=flavor_name
```

4. After making these changes, follow the steps in the Dell EMC Red Hat OpenStack NFV Solution Automated Deployment Guide

This script has pre-set default values for the number of HugePages allocated for the two sizes. For 1GB pages, exactly 96 HugePages are allocated. For 2MB pages, exactly 49,152 HugePages are allocated. These values cannot be changed in this version of the script.

Note: As a best-practice while using this script, please ensure that the combined RAM requirement of all VMs that need to use HugePages adds up to 1/2 of the total amount of memory space addresable by the number of HugePages allocated. Consider that the minimum RAM requirement of each compute node in this solution is 128GB. With that in mind, the following example can illustrates this best-practice: since this script enables 96 HugePages of size 1GB, in a compute node, a pool of VMs requiring 1GB HugePages could be configured as: 8 VMs, each with RAM requirement of 6GB. This would use 8 x 6GB = 48GB RAM for the VMs, which is 1/2 of 96GB addresseable via 1GB HugePages. Similar calculations apply to 2MB HugePages.

The script sets aside 8GB of RAM for use by the Host OS on each compute node.

An example of the sample.ini is provided in below.

Post deployment - Enabling and Deploying Huge Pages

The Dell EMC Red Hat OpenStack NFV Solution provides the ability to enable Linux HugePage support on all Nova compute nodes in the solution. This section provides the instructions to enable and disable the feature after the solution is deployed.. The script resides on the director node and using information obtained from the director node (the number and IP addresses of the controller and compute nodes in the solution) this script carries out commands to enable/disable HugePage support.

Note: Please note the pre-requisites for this facility. Locally install the "enable_hugepages.py" script supplied with the solution on the director node in the solution set up, and make it executable.

```
# cd <sourcedir>
# tar xvzf Dell-NFV-6.0.tgz
# cp <sourcedir>/enable_hugepages.py \
       /opt/Dell/NFV/6.0/enable_hugepages.py
# chmod 755 /opt/Dell/NFV/6.0/enable_hugepages.py
```

Enabling Huge Pages

The enable_hugepages script to enable HugePages using action=set is invoked as follows:

```
$ python enable_hugepages.py \
         --openstack_admin_username=OPENSTACK_ADMIN_USERNAME \
         --openstack_admin_password=OPENSTACK_ADMIN_PASSWORD \
         --authentication_url=AUTHENTICATION_URL \
        [--director_install_user=DIRECTOR_INSTALL_USER] \
         --flavor_name=FLAVOR_NAME \
        [-- hpgsize HPGSIZE]  \
```

```
                 [--action {set}] \
                 [--logfile LOGFILE]


           Enable Huge Pages in Dell NFV Platform
                  -h, --help
                    show this help message and exit
                  --openstack_admin_username OPENSTACK_ADMIN_USERNAME
                    username of the undercloud
                  --openstack_admin_password OPENSTACK_ADMIN_PASSWORD
                    password of the undercloud
                  --authentication_url AUTHENTICATION_URL
                    authentication url of the undercloud
                  --director_install_user DIRECTOR_INSTALL_USER
                    user for undercloud/overcloud installation
                  --flavor_name FLAVOR_NAME
                    name of flavor to be created
                 --hpgsize HPGSIZE
                     size of hugepage (s) 'size in MB/GB'
                  --action {set}
                    set  the custom flavor
                  --logfile LOGFILE
                    name of the logfile
```

For enable_hugepages help, please execute the following command:

```
         $ python enable_hugepages.py --help
```

Notes on the action=set invocation of this script:

The script will apply HugePages pinning enablement to all the compute nodes in the solution.

If a Nova flavor already exists with the same name as supplied by the script, then the script will apply the enablement to the flavor. If the flavor does not exist in the solution, then it is created and Huge Pages are enabled on it.

This script has pre-set default values for the number of HugePages allocated for the two sizes. For 1GB pages, exactly 96 HugePages are allocated. For 2MB pages, exactly 49,152 HugePages are allocated. These values cannot be changed in this version of the script.

Note: As a best-practice while using this script, please ensure that the combined RAM requirement of all VMs that need to use HugePages adds up to 1/2 of the total amount of memory space addresable by the number of HugePages allocated. Consider that the minimum RAM requirement of each compute node in this solution is 128GB. With that in mind, the following example can illustrates this best-practice: since this script enables 96 HugePages of size 1GB, in a compute node, a pool of VMs requiring 1GB HugePages could be configured as: 8 VMs, each with RAM requirement of 6GB. This would use 8 x 6GB = 48GB RAM for the VMs, which is 1/2 of 96GB addresseable via 1GB HugePages. Similar calculations apply to 2MB HugePages.

The script sets aside 8GB of RAM for use by the Host OS on each compute node.

After successful execution of the script, the following message is displayed at the end of the logfile (or in the terminal if a logfile name was not given). This message indicates that the script execution is successful and no further steps are required.

```
       Script executed successfully. Please check the logs for details.
```

Disabling HugePages

The enable_hugepages script disablement of HugePages using action=remove is invoked as follows:

```
$ python enable_hugepages.py \
--openstack_admin_username=OPENSTACK_ADMIN_USERNAME \
--openstack_admin_password=OPENSTACK_ADMIN_PASSWORD \
--authentication_url=AUTHENTICATION_URL \
[--director_install_user=DIRECTOR_INSTALL_USER] \
--flavor_name=FLAVOR_NAME \ [--hpgsize HPGSIZE] \
[--action {remove}] \ [--logfile LOGFILE] \ [--remove_flavor]

Disable Huge Pages in Dell NFV Platform

-h, --help
show this help message and exit
--openstack_admin_username OPENSTACK_ADMIN_USERNAME username of the
 undercloud
--openstack_admin_password OPENSTACK_ADMIN_PASSWORD password of the
 undercloud
--authentication_url AUTHENTICATION_URL authentication url of the undercloud
--director_install_user DIRECTOR_INSTALL_USER user for undercloud/overcloud
 installation
--flavor_name FLAVOR_NAME
name of flavor to be created
--hpgsize HPGSIZE
size of hugepage(s) 'size in MB/GB'
--action {remove}
remove the custom flavor
--logfile LOGFILE
name of the logfile
--remove_flavor
Flag to remove flavor
```

For enable_hugepages help, execute the following command:

```
$ python enable_hugepages.py --help
```

Notes on the action=remove invocation of this script:

The script will remove the HugePages enablement from all compute nodes in the solution.

The script will also remove the named flavor if the optional --remove_flavor flag is given. You are asked to confirm whether you intended to remove the flavor.

Making Changes to HugePages Settings in a Flavor

Note: To make changes to the HugePages settings in a flavor, this facility requires you to first use action=remove, followed by action=set with a changed set of parameters.

After Successful Execution

After successful execution of the script, the following message is displayed at the end of the logfile (or in the terminal if a logfile name was not given). This message indicates that the script execution is successful and no further steps are required.

```
Script executed successfully. Please check the logs for details.
```

Checking OpenStack Nova Hypervisor Health

Following either action={set | remove} modes of execution of the script, you are strongly encouraged to check the health of the hypervisors across the pool of compute nodes in the solution. Use the command below on a controller node to list all compute node hypervisors.

```
# nova hypervisor-list
```

The report printed on the terminal must show all compute node hypervisors marked with a 'up' state.

```
+----+---------------------+-------+---------+
| ID | Hypervisor hostname | State | Status  |
+----+---------------------+-------+---------+
| 1  | ComputeNode0        | up    | enabled |
| 2  | ComputeNode1        | up    | enabled |
| 3  | ComputeNode2        | up    | enabled |
+----+---------------------+-------+---------+
```

Example of settings_sample.ini

```
##########################################################################
#                                                                        #
#Copy & rename this file for your own stamp.                             #
#Review ALL settings below, pay particular attention to paths/ip's etc..#
#                                                                        #
##########################################################################

[Cluster Settings]
# Only developers should set to false.
enable_version_locking=true

# This pathname must be the full path to the properties file which
# describes the cluster. You should copy *this* sample settings file
# (settings_sample.ini) and the sample properties file
# (sample.properties) to another directory, and customize them for your
# cluster. Then use the path to your customized properties file here.
cluster_nodes_configuration_file=path to your .properties file

# Domain name for the cluster (i.e., mycluster.lab)
domain=domain.net

# DRAC credentials with IPMI privilege for the nodes
ipmi_user=root
ipmi_password=xxxxxxx

# User for the undercloud/overcloud installation
director_install_user=osp_admin
director_install_user_password=xxxxxxx

# Name of the Overcloud.
# The nodes hostnames will be prepended with the given name and a dash
overcloud_name=overcloud
use_internal_repo=false
# Semi-colon ( ; ) separated list of internal repos to use, if needed.
 Typically
```

```
# not used.
internal_repos_locations=CHANGEME_INTERNAL_REPO_URL

# Subscription Manager account info for registering Red Hat subscriptions
subscription_manager_user=xxxxxxx
subscription_manager_password=xxxxxxxxxxxxxx

# The following pool IDs provide different collections of repositories.
# Each is labeled with possible subscription names.

# Red Hat Enterprise Linux Self-Supported Business Partner NFR
subscription_manager_pool_sah=xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

# Red Hat Enterprise Linux OpenStack Platform Self-Supported Business
 Partner NFR
subscription_manager_pool_vm_rhel=xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

# Red Hat Ceph Storage Business Partner Self-Supported NFR, (Physical Node)
subscription_manager_vm_ceph=xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

subscription_check_retries=20
ntp_servers=0.centos.pool.ntp.org
time_zone=America/Chicago

# Use static ip adresses for the overcloud nodes if set to true (ips need to
 be defined in the .properties)
# Use dhcp if set to false (ips not required in the .properties)
overcloud_static_ips=true

# External network details
external_netmask=255.255.255.0
external_gateway=10.7.101.1

# Nova public network details
public_api_network=192.168.190.0/24
public_api_vlanid=190
public_api_gateway=192.168.190.1
public_api_netmask=255.255.255.0
public_api_allocation_pool_start=192.168.190.121
public_api_allocation_pool_end=192.168.190.250

# Private API network details
private_api_network=192.168.140.0/24
private_api_vlanid=140
private_api_netmask=255.255.255.0
private_api_allocation_pool_start=192.168.140.121
private_api_allocation_pool_end=192.168.140.250

# Storage network details
storage_network=192.168.170.0/24
storage_vlanid=170
storage_netmask=255.255.255.0
storage_allocation_pool_start=192.168.170.125
storage_allocation_pool_end=192.168.170.250

# Provisioning network details
provisioning_network=192.168.120.0/24
provisioning_vlanid=120
provisioning_netmask=255.255.255.0
provisioning_gateway=192.168.120.1
provisioning_net_dhcp_start=192.168.120.121
provisioning_net_dhcp_end=192.168.120.250
discovery_ip_range=192.168.120.21,192.168.120.120
```

```
# Storage cluster network details
storage_cluster_network=192.168.180.0/24
storage_cluster_vlanid=180
storage_cluster_allocation_pool_start=192.168.180.121
storage_cluster_allocation_pool_end=192.168.180.250

# Management network details
management_network=192.168.110.0/24
managment_vlanid=110
managment_netmask=255.255.255.0
name_server=8.8.8.8

# Tenant network details
# Not used unless you wish to configure Generic Routing Encapsulation (GRE)
 networks.
tenant_network=192.168.130.0/24
tenant_network_allocation_pool_start=192.168.130.121
tenant_network_allocation_pool_end=192.168.130.250

# Nova Private network details
tenant_vlan_range=201:220


# Use static VIPs ip addresses for the overcloud
use_static_vips=true

# The following VIP settings apply if the above use_static_vips is enabled.

# VIP for the redis service on the Private API api network
# Note that this IP must lie OUTSIDE the private_api_allocation_pool_start/
end range
redis_vip=192.168.140.251

# VIP for the provisioning network
# Note that this IP must lie INSIDE the provisioning_net_dhcp_start/end
 range
# but cannot be the first IP in that range
provisioning_vip=192.168.120.250

# VIP for the Private API network
# Note that this IP must lie INSIDE the private_api_allocation_pool_start/
end range
private_api_vip=192.168.140.250

# VIP for the Public API network
# Note that this IP must lie INSIDE the public_api_allocation_pool_start/end
 range
public_api_vip=192.168.190.250

# VIP for the Storage network
# Note that this IP must lie INSIDE the storage_allocation_pool_start/end
 range
storage_vip=192.168.170.250

# VIP for the Storage cluster network
# The Storage Clustering network is not connected to the controller nodes,
# so the VIP for this network must be mapped to the provisioning network
# Note that this IP must lie INSIDE the provisioning_net_dhcp_start/range
 but
# cannot be the first IP in that range
storage_cluster_vip=192.168.120.249
```

```
# iDRAC doesn't always play nice due to a bug. Workaround built in the
 deployer already, but occasionally fails.
# Option below is to use a custom instack.json (i.e., not using idracula)
use_custom_instack_json=false
custom_instack_json=n/a

# Bonding options configuration by node type
controller_bond_opts=802.3ad miimon=100
compute_bond_opts=802.3ad miimon=100
storage_bond_opts=802.3ad miimon=100


# Overcloud deployment timeout value - default is 120mns, but can be tweaked
 here if required.
overcloud_deploy_timeout=120


# Interfaces per node type
controller_bond0_interfaces=em1 p1p1
controller_bond1_interfaces=em2 p1p2
controller_provisioning_interface=em3

compute_bond0_interfaces=em1 p1p1
compute_bond1_interfaces=em2 p1p2
compute_provisioning_interface=em3

storage_bond0_interfaces=em1 p2p1
storage_bond1_interfaces=em2 p2p2
storage_provisioning_interface=em3

#Default driver is DRAC.
use_ipmi_driver=true

# EQLX backend settings, if applicable
enable_eqlx_backend = false
eqlx_backend_name=CHANGEME_EQLX_GROUP_NAME
eqlx_san_ip=CHANGEME_SAN_IP
eqlx_san_login=CHANGEME_SAN_USERNAME
eqlx_san_password=CHANGEME_SAN_PASSWORD
eqlx_ch_login=CHANGEME_CHAP_USERNAME
eqlx_ch_pass=CHANGEME_CHAP_PASSWORD
eqlx_group_n=CHANGEME_EQLX_GROUP_NAME
eqlx_thin_provisioning=true
eqlx_pool=default
eqlx_use_chap=false

# Provide NFV feature enablement parameters here
#
# Enter value of enable_hugepages as true/True and false/False
enable_hugepages=true
#
# 'hpgsize' (size of HugePages) can only be 1GB or 2MB, and must be
 specified precisely as noted.
# 'hpgnum' (# of HugePages) can only be 96 for 1GB pages, and 49152 for 2MB
 pages.
# No other values are accepted in this version.
hpgsize=1GB
hpgnum=96
# hpgsize=2MB
# hpgnum=49152
#
# Specify the flavor to which HugePage enablement must be applied:
# Use only Uppercase and Lowercase letters, digits, '.', '-' and '_' in
 flavor names.
```

```
hpg_flavor_name=m1.test

# NUMA
# Enter value of enable_numa as true/True and false/False
enable_numa=true
#
# 'hostos_cpus' should be an enumerated list of cpus for dedicated use of
 the HostOS (KVM).
# Suggested best-practice: dedicate CPUs 0-7 for the use for the HostOS. The
 rest can be used
# to run the Compute Node's VMs.
# The enumerated list must be a comma-separated list of +ve integers with no
 whitespace characters.
hostos_cpus=0,1,2,3,4,5,6,7
#
# Specify the flavor to which NUMA enablement must be applied:
# Use only Uppercase and Lowercase letters, digits, '.', '-' and '_' in
 flavor names.
numa_flavor_name=m1.test

# Compellent parameters. See the Deployment Guide for description of the
 parameters.
enable_dellsc_backend=false
dellsc_backend_name=CHANGEME
dellsc_api_port=3033
dellsc_iscsi_ip_address=CHANGEME
dellsc_iscsi_port=3260
dellsc_san_ip=CHANGEME
dellsc_san_login=CHANGEME
dellsc_san_password=CHANGEME
dellsc_ssn=CHANGEME
dellsc_server_folder=cmpl_iscsi_servers
dellsc_volume_folder=cmpl_iscsi_volumes

# Set to true to enable Nova usage of Ceph for ephemeral storage.
# If set to false, Nova uses the storage local to the compute.
enable_rbd_backend=true

# Set to true to enable fencing
# Please refer to the following document for more details on fencing :
# Dell Red Hat OpenStack Cloud Solution Deployment Guide - Version 6.0
enable_fencing=false

# Set to true to enable instance HA
# Note : fencing must also be enabled (setting above)
# Please refer to following technical document for more details on Instance
 HA :
# Using Instance High Availability in the Dell Red Hat OpenStack Cloud
 Solution - Version 6.0
enable_instance_ha=false

[Bastion Settings]
rhl72_iso=/var/www/html/RH7/RHEL-7.2-20151030.0-Server-x86_64-dvd1.iso
rhel_install_location=http://192.168.120.11/pub
sah_kickstart=/var/www/html/ks.cfg
cloud_repo_dir=/pathto/cloud_repo
# If you want the sanity script to run on deployment completion (Appendix C,
 etc.), you may do so.
run_sanity=false
# If you want to run Tempest post-deployment, you may do so. The sanity
 script must also run to create networks for Tempest.
run_tempest=false
tempest_smoke_only=true
# RDO cloud images
```

```
# Available to download @ https://access.redhat.com/downloads/content/191/
ver=8/rhel---7/8/x86_64/product-software
discovery_ram_disk_image=/pathto/discovery-ramdisk-7.1.0-39.tar
overcloud_image=/pathto/overcloud-full-7.1.0-39.tar
# if option below is enabled, images will be pulled fom the cdn (and the
 above x2 settings ignored)
pull_images_from_cdn=true
```

Appendix E

Enabling Numa pre-deployment or post-deployment

Topics:

This appendix provide instructions for enabling Numa pre-deployment or post-deplyment

NUMA Overview

The Dell EMC Red Hat OpenStack NFV Solution provides the ability to enable Linux NUMA and CPU pinning support on all Nova compute nodes in the the solution.

Non-uniform memory access or NUMA allows multiple CPUs to share local memory, which improves performance due to improved memory access times.

This document provides the instructions to configure this feature at the time of deployment (Deployment time enablement (DTE)) or post deployment.

Caution: BEFORE you use this facility - please note that the scripts described in this Appendix WILL REBOOT the compute nodes in the solution several times. You are strongly advised to save all work, stop all services, and/or ask users to log out before proceeding any further.

Warning: Please do not reboot any nodes when the optional feature scripts are being used.

Note: This facility allows you to create a new Nova flavor with default values of 4 vCPUs and 4GB RAM. If you would like to enable NUMA and CPU pinning on a flavor of different size, then create the desired flavor prior to following the steps described below.

Prerequisites

In order to deploy the Dell EMC Red Hat OpenStack NFV Solution on Dell hardware, the following prerequisites must be met:

- Successful hardware setup and configuration, per the *Dell EMC Red Hat OpenStack NFV Solution Hardware Deployment Guide*
- Must have a valid subscription and outbound network access to Red Hat's Content Delivery Network.
- Addition changes to the settings_sample.ini are required for automated deployment see the *Dell EMC Red Hat OpenStack NFV Solution Automated Deployment Guide - Version 6.0* for details.

Predeployment

The Dell EMC Red Hat OpenStack NFV Solution provides the ability to enable Linux NUMA and CPU pinning support on all Nova compute nodes during deployment of the solution. This section provides the instructions to enable the feature when the solution is being deployed.

Enabling NUMA with DTE

To enable Numa:

1. Open an SSH session to the SAH node.
2. Log in as the root user.
3. Edit the settings_sample.ini (e.g. acme.ini) file. *flavorname* can be an existing or newly created flavor.

```
Enable_numa=true
numa_flavor_name=flavorname
```

4. After making these changes, follow the steps in the Dell EMC Red Hat OpenStack NFV Solution Automated Deployment Guide

The script sets aside CPU numbers 0, 2, 4, 6 on CPU socket-0, and CPU numbers 1, 3, 5, 7 on CPU socket-1 for exclusive use by the Host OS on each compute node. The rest of the CPUs (8-47) are available for use by and for pinning of tenant VMs on the compute nodes.

An example of the sample.ini is provided in Appendex D.

Post deployment - Enabling and Deploying NUMA

The Dell EMC Red Hat OpenStack NFV Solution provides the ability to enable Linux NUMA and CPU pinning support on all Nova compute nodes in the solution. This section provides the instructions to enable and disable this feature after solution is deployed. The script resides on the director node and using information obtained from the director node (the number and IP addresses of the controller and compute nodes in the solution) this script carries out commands to enable/disable NUMA/CPU Pinning support.

Note: Please note the pre-requisites for this facility. Locally install the "enable_numa.py" script supplied with the solution on the director node in the solution set up, and make it executable.

```
# cd <sourcedir>
# tar xvzf Dell-NFV-6.0.tgz
# cp <sourcedir>/enable_numa.py \
       /opt/Dell/NFV/6.0/enable_numa.py
# chmod 755 /opt/Dell/NFV/6.0/enable_numa.py
```

Enabling NUMA Optimization and CPU Pinning

The enable_numa script to enable NUMA and CPU pinning using action=set is invoked as follows:

```
$ python enable_numa.py \
         --openstack_admin_username=OPENSTACK_ADMIN_USERNAME \
         --openstack_admin_password=OPENSTACK_ADMIN_PASSWORD \
         --authentication_url=AUTHENTICATION_URL \
        [--director_install_user=DIRECTOR_INSTALL_USER] \
         --flavor_name=FLAVOR_NAME \
        [--action {set}] \
        [--logfile LOGFILE]

  Enable NUMA in Dell NFV Platform
         -h, --help
           show this help message and exit
         --openstack_admin_username OPENSTACK_ADMIN_USERNAME
           username of the undercloud
         --openstack_admin_password OPENSTACK_ADMIN_PASSWORD
           password of the undercloud
         --authentication_url AUTHENTICATION_URL
           authentication url of the undercloud
         --director_install_user DIRECTOR_INSTALL_USER
           user for undercloud/overcloud installation
         --flavor_name FLAVOR_NAME
           name of flavor to be created
         --action {set}
           set  the custom flavor
         --logfile LOGFILE
           name of the logfile
```

For enable_numa help, please execute the following command:

```
$ python enable_numa.py --help
```

Notes on the action=set invocation of this script:

The script will apply NUMA and CPU pinning enablement to all the compute nodes in the solution.

If a Nova flavor already exists with the same name as supplied by the script, then the script will apply the enablement to the flavor. If the flavor does not exist in the solution, then it is created and NUMA and CPU pinning are enabled on it.

The script sets aside CPU numbers 0, 2, 4, 6 on CPU socket-0, and CPU numbers 1, 3, 5, 7 on CPU socket-1 for exclusive use by the Host OS on each compute node. The rest of the CPUs (8-47) are available for use by and for pinning of tenant VMs on the compute nodes.

After successful execution of the script, the following message is displayed at the end of the logfile (or in the terminal if a logfile name was not given). This message indicates that the script execution is successful and no further steps are required.

```
Script executed successfully. Please check the logs for details.
```

Disabling NUMA Optimization and CPU Pinning

The enable_numa script disabling of NUMA and CPU pinning using action=remove is invoked as follows:

```
$ python enable_numa.py \
                --openstack_admin_username=OPENSTACK_ADMIN_USERNAME \
                --openstack_admin_password=OPENSTACK_ADMIN_PASSWORD \
                --authentication_url=AUTHENTICATION_URL \
               [--director_install_user=DIRECTOR_INSTALL_USER] \
                --flavor_name=FLAVOR_NAME \
               [--action {remove}] \
               [--logfile LOGFILE] \
               [--remove_flavor]

         Disable NUMA in Dell NFV Platform
                --h, --help
                  show this help message and exit
                --openstack_admin_username OPENSTACK_ADMIN_USERNAME
                  username of the undercloud
                --openstack_admin_password OPENSTACK_ADMIN_PASSWORD
                  password of the undercloud
                --authentication_url AUTHENTICATION_URL
                  authentication url of the undercloud
                --director_install_user DIRECTOR_INSTALL_USER
                  user for undercloud/overcloud installation
                --flavor_name FLAVOR_NAME
                  name of flavor to be created
                --action {remove}
                  remove the custom flavor
                --logfile LOGFILE
                  name of the logfile
                --remove_flavor
                  flag to remove flavor
```

For enable_numa help, execute the following command:

```
$ python enable_numa.py --help
```

Notes on the action=remove invocation of this script:

The script disables the NUMA and CPU pinning enablement from all compute nodes in the solution.

The script will also remove the named flavor if the optional --remove_flavor flag is given. You are asked to confirm whether you intended to remove the flavor.

Making Changes to NUMA Settings in a Flavor

Note: The NUMA and CPU pinning facility supports only two options: action=set and action=remove for a flavor. No further changes can be made to the settings.

After Successful Execution

After successful execution of the script, the following message is displayed at the end of the logfile (or in the terminal if a logfile name was not given). This message indicates that the script execution is successful and no further steps are required.

```
Script executed successfully. Please check the logs for details.
```

Checking OpenStack Nova Hypervisor Health

Following either action={set | remove} modes of execution of the script, you are strongly encouraged to check the health of the hypervisors across the pool of compute nodes in the solution. Use the command below on a controller node to list all compute node hypervisors.

```
# nova hypervisor-list
```

The report printed on the terminal must show all compute node hypervisors marked with a 'up' state.

```
+----+---------------------+-------+---------+
| ID | Hypervisor hostname | State | Status  |
+----+---------------------+-------+---------+
| 1  | ComputeNode0        | up    | enabled |
| 2  | ComputeNode1        | up    | enabled |
| 3  | ComputeNode2        | up    | enabled |
+----+---------------------+-------+---------+
```

Appendix F

IPv6 Addressing for Tenant VM (optional)

Topics:

This appendix provide instructions for enabling IPv6 addressing on Tenant VM

Enabling Tenant VM IPv6 Addressing (optional)

The Dell EMC Red Hat OpenStack NFV Solution provides the ability to enable tenant VM IPv6 addressing support on all Nova Compute nodes in the solution. This Appendix provides the instructions to enable the feature. There is no special script needed for this enablement.

Note: ATTENTION! The user must have superuser privileges to execute the following steps.

Alternative 1: CLI-based approach

There are two steps to take to enable tenant VM IPv6 addressing capability.

STEP 1: On a Controller node in the solution, create a network using the command:

```
# neutron net-create <network-name>
```

The Neutron facility will print a table of field-value parameters for the new network <network-name>, after it is created.

STEP 2: On the Controller node in the solution, create a new subnet using the command:

```
$ neutron subnet-create --name <network-name> \
  --ip-version 6 IPv6_NETWORK <network_CIDR> \
  --ipv6_ra_mode slaac \
  --ipv6_address_mode slaac
```

The Neutron facility will print a table of field-value parameters for the new subnet <network-name>, after it is created.

Now, create the desired tenant VM instance, and assign its NIC to use the IPv6 network that was just created.

Alternative 2: UI-based approach

Take the following steps to enable tenant VMs to communicate using IPv6.

1. Log into the Horizon UI of OpenStack
2. Select 'Project' on the top toolbar
3. Select 'Network'
4. Then, select 'Networks'
5. Click the 'Create Networks' button

These steps will lead you to the following screen shown in *Figure 2: OpenStack Horizon UI form used to enable IPv6 communication between tenant VMs* on page 80. Enter appropriate values in the form fields, as they pertain to your solution deployment.

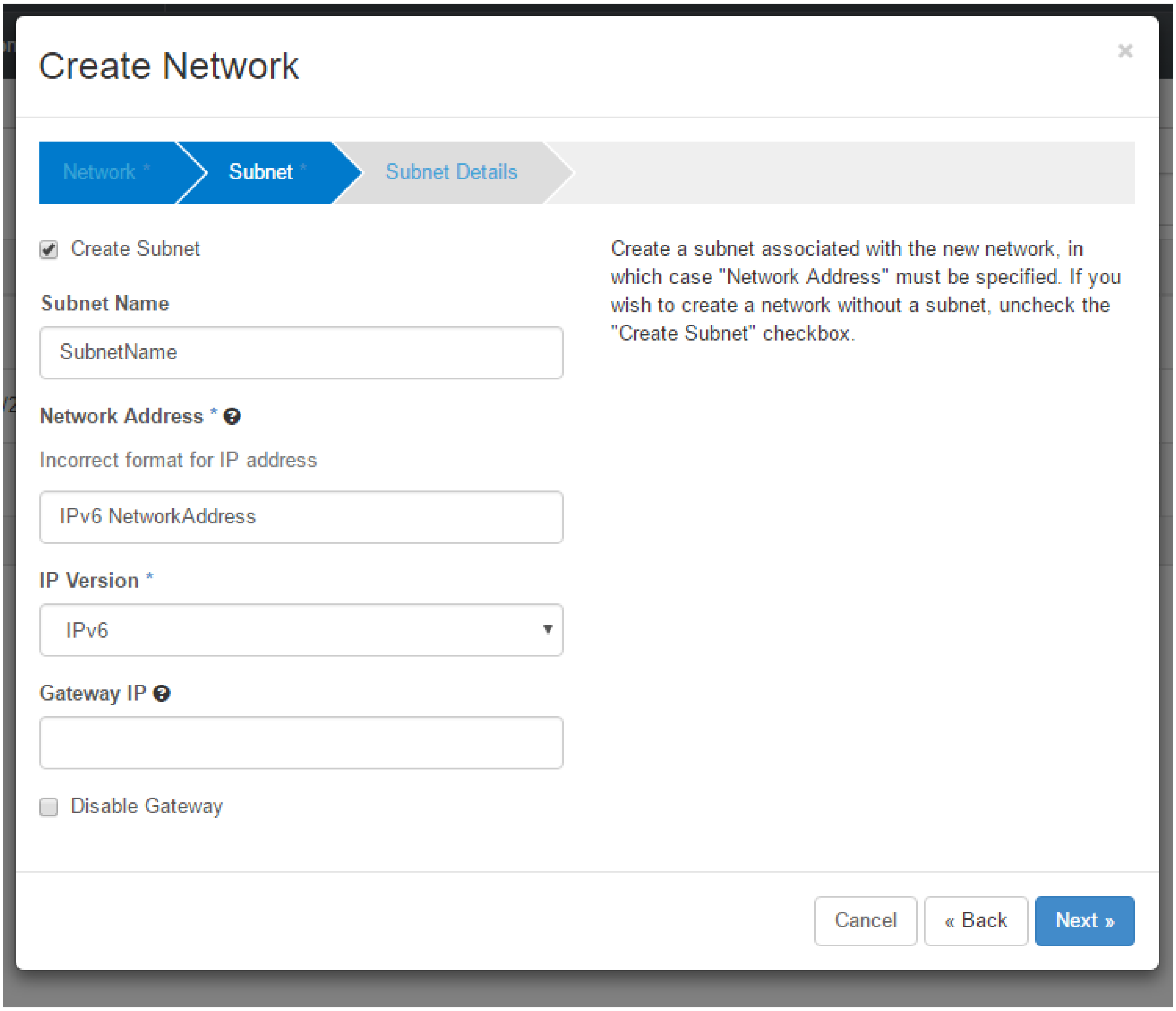

Figure 2: OpenStack Horizon UI form used to enable IPv6 communication between tenant VMs

Appendix G

Security-Enhanced Linux

Topics:

- *Notes on Enablement of Security-Enhanced (SELinux) Protection Capability*

This appendix provides basic introduction for SELinux

Notes on Enablement of Security-Enhanced (SELinux) Protection Capability

Starting with version 6.0, the Dell EMC Red Hat OpenStack NFV Solution provides enablement of Security-Enhanced Linux (SELinux) protection on all Controller and Compute nodes in the solution. This feature is enabled by default, and does not require the end-user to take any actions. If you need additional services or implementation help, please contact your Dell EMC sales representative.

The solution components that have been enabled for SELinux protection are listed below. SELinux is enabled by default and runs in enforcing mode. No separate actions are necessary.

```
Glance
HAProxy
KeepAlive
Keystone
MongoDB
MySQL
Neutron
Nova
OVS
RabbitMQ
Redis
Swift
```

Appendix H

References

Topics:

Additional information can be obtained at *http://www.dell.com/nfv* or by e-mailing *nfv@dell.com*.

If you need additional services or implementation help, please contact your Dell EMC sales representative.

To Learn More

For more information on the Dell EMC Red Hat OpenStack Cloud Solution visit *http://www.dell.com/learn/us/en/04/solutions/red-hat-openstack*.