OPNFV Installation instructions (Apex)

Release draft (c5750be)

OPNFV

February 15, 2016

CONTENTS

Contents:

CHAPTER

ONE

ABSTRACT

This document describes how to install the Bramaputra release of OPNFV when using Apex as a deployment tool covering it's limitations, dependencies and required system resources.

CHAPTER

TWO

LICENSE

Bramaputra release of OPNFV when using Apex as a deployment tool Docs (c) by Tim Rozet (Red Hat) and Dan Radez (Red Hat)

Bramaputra release of OPNFV when using Apex as a deployment tool Docs are licensed under a Creative Commons Attribution 4.0 International License. You should have received a copy of the license along with this. If not, see <http://creativecommons.org/licenses/by/4.0/>.

CHAPTER

THREE

INTRODUCTION

This document describes the steps to install an OPNFV Bramaputra reference platform, as defined by the Genesis Project using the Apex installer.

The audience is assumed to have a good background in networking and Linux administration.

CHAPTER

FOUR

PREFACE

Apex uses the RDO Manager Open Source project as a server provisioning tool. RDO Manager is the RDO Project implimentation of OpenStack's Triple-O project. The Triple-O image based life cycle installation tool provisions an OPNFV Target System (3 controllers, n number of compute nodes) with OPNFV specific configuration provided by the Apex deployment tool chain.

The Apex deployment artifacts contain the necessary tools to deploy and configure an OPNFV target system using the Apex deployment toolchain. These artifacts offer the choice of using the Apex bootable ISO (`opnfv-apex-bramaputra.iso`) to both install CentOS 7 and the nessesary materials to deploy or the Apex RPM (`opnfv-apex.rpm`) which expects installation to a CentOS 7 libvirt enabled host. The RPM contains a collection of configuration file, prebuilt disk images, and the automatic deployment script (`opnfv-deploy`).

An OPNFV install requires a "Jumphost" in order to operate. The bootable ISO will allow you to install a customized CentOS 7 release to the Jumphost, which includes the required packages needed to run `opnfv-deploy`. If you already have a Jumphost with CentOS 7 installed, you may choose to skip the ISO step and simply install the (`opnfv-apex.rpm`) RPM. The RPM is the same RPM included in the ISO and includes all the necessary disk images and configuration files to execute an OPNFV deployment. Either method will prepare a host to the same ready state for OPNFV deployment.

`opnfv-deploy` instantiates an RDO Manager Instack VM server using libvirt as its provider. This VM is then configured and used to provision the OPNFV target deployment (3 controllers, n compute nodes). These nodes can be either virtual or bare metal. This guide contains instructions for installing either method.

CHAPTER

FIVE

TRIPLE-O DEPLOYMENT ARCHITECTURE

Apex is based on RDO Manager which is the RDO Project's implementation of the OpenStack Triple-O project. It is important to understand the basics of a Triple-O deployment to help make decisions that will assist in successfully deploying OPNFV.

Triple-O stands for OpenStack On OpenStack. This means that OpenStack will be used to install OpenStack. The target OPNFV deployment is an OpenStack cloud with NFV features built-in that will be deployed by a smaller all-in-one deployment of OpenStack. In this deployment methodology there are two OpenStack installations. They are referred to as the undercloud and the overcloud. The undercloud is used to deploy the overcloud.

The undercloud is the all-in-one installation of OpenStack that includes baremetal provisioning. RDO Manager's deployment of the undercloud is call Instack. Instack will be deployed as a virtual machine on a jumphost. This VM is pre-built and distributed as part of the Apex RPM.

The overcloud is OPNFV. Configuration will be passed into Instack and Instack will use OpenStack's orchestration component call Heat to execute a deployment will provision the target nodes to become OPNFV.

CHAPTER

SIX

SETUP REQUIREMENTS

6.1 Jumphost Requirements

The Jumphost requirements are outlined below:

1. CentOS 7 (from ISO or self-installed).
2. Root access.
3. libvirt virtualization support.
4. minimum 2 networks and maximum 6 networks, multiple NIC and/or VLAN combinations are supported. This is virtualized for a VM deployment.
5. The Bramaputra Apex RPM.
6. 16 GB of RAM for a bare metal deployment, 56 GB of RAM for a VM deployment.

6.2 Network Requirements

Network requirements include:

1. No DHCP or TFTP server running on networks used by OPNFV.
2. 2-6 separate networks with connectivity between Jumphost and nodes.
    - Control Plane Network (Provisioning)
    - Private / Internal Network*
    - External Network
    - Storage Network*
3. Lights out OOB network access from Jumphost with IPMI node enabled (bare metal deployment only).
4. Admin or public network has Internet access, meaning a gateway and DNS availability.

** These networks can be combined with each other or all combined on the Control Plane network.*
** Non-External networks will be consolidated to the Control Plane network if not specifically configured.*

6.3 Bare Metal Node Requirements

Bare metal nodes require:

1. IPMI enabled on OOB interface for power control.
2. BIOS boot priority should be PXE first then local hard disk.
3. BIOS PXE interface should include Control Plane network mentioned above.

6.4 Execution Requirements (Bare Metal Only)

In order to execute a deployment, one must gather the following information:

1. IPMI IP addresses for the nodes.
2. IPMI login information for the nodes (user/pass).
3. MAC address of Control Plane / Provisioning interfaces of the overcloud nodes.

CHAPTER

SEVEN

INSTALLATION HIGH-LEVEL OVERVIEW - BARE METAL DEPLOYMENT

The setup presumes that you have 6 bare metal servers and have already setup network connectivity on at least 2 interfaces for all servers via a TOR switch or other network implementation.

The physical TOR switches are **not** automatically configured from the OPNFV reference platform. All the networks involved in the OPNFV infrastructure as well as the provider networks and the private tenant VLANs needs to be manually configured.

The Jumphost can be installed using the bootable ISO or by other means including the (`opnfv-apex`) RPM and virtualization capabilities. The Jumphost should then be configured with an IP gateway on its admin or public interface and configured with a working DNS server. The Jumphost should also have routable access to the lights out network.

`opnfv-deploy` is then executed in order to deploy the Instack VM. `opnfv-deploy` uses three configuration files in order to know how to install and provision the OPNFV target system. The information gathered under section Execution Requirements (Bare Metal Only) is put into the YAML file (`/etc/opnfv-apex/inventory.yaml`) configuration file. Deployment options are put into the YAML file (`/etc/opnfv-apex/deploy_settings.yaml`). Networking definitions gathered under section Network Requirements are put into the YAML file (`/etc/opnfv-apex/network_settings.yaml`). `opnfv-deploy` will boot the Instack VM and load the target deployment configuration into the provisioning toolchain. This includes MAC address, IPMI, Networking Environment and OPNFV deployment options.

Once configuration is loaded and Instack is configured it will then reboot the nodes via IPMI. The nodes should already be set to PXE boot first off the admin interface. The nodes will first PXE off of the Instack PXE server and go through a discovery/introspection process.

Introspection boots off of custom introspection PXE images. These images are designed to look at the properties of the hardware that is booting off of them and report the properties of it back to the Instack node.

After introspection Instack will execute a Heat Stack Deployment to being node provisioning and configuration. The nodes will reboot and PXE again off the Instack PXE server to provision each node using the Glance disk images provided by Instack. These disk images include all the necessary packages and configuration for an OPNFV deployment to execute. Once the node's disk images have been written to disk the nodes will boot off the newly written disks and execute cloud-init which will execute the final node configuration. This configuration is largly completed by executing a puppet apply on each node.

CHAPTER

EIGHT

INSTALLATION HIGH-LEVEL OVERVIEW - VM DEPLOYMENT

The VM nodes deployment operates almost the same way as the bare metal deployment with a few differences. `opnfv-deploy` still deploys an Instack VM. In addition to the Instack VM a collection of VMs (3 control nodes + 2 compute for an HA deployment or 1 control node and 1 compute node for a Non-HA Deployment) will be defined for the target OPNFV deployment. The part of the toolchain that executes IPMI power instructions calls into libvirt instead of the IPMI interfaces on baremetal servers to operate the power managment. These VMs are then provisioned with the same disk images and configuration that baremetal would be.

To RDO Manager these nodes look like they have just built and registered the same way as bare metal nodes, the main difference is the use of a libvirt driver for the power management.

CHAPTER

NINE

INSTALLATION GUIDE - BARE METAL DEPLOYMENT

WARNING: Baremetal documentation is not complete. WARNING: The main missing instructions are r elated to bridging the networking for the undercloud to the physical underlay network for the overcloud to be deployed to.

This section goes step-by-step on how to correctly install and provision the OPNFV target system to bare metal nodes.

9.1 Install Bare Metal Jumphost

1a. If your Jumphost does not have CentOS 7 already on it, or you would like to do a fresh install, then download the Apex bootable ISO from OPNFV artifacts <http://artifacts.opnfv.org/>.

1b. If your Jump host already has CentOS 7 with libvirt running on it then install the opnfv-apex RPM from OPNFV artifacts <http://artifacts.opnfv.org/>.

2a. Boot the ISO off of a USB or other installation media and walk through installing OPNFV CentOS 7. The ISO comes prepared to be written directly to a USB drive with dd as such:

```
dd if=opnfv-apex.iso of=/dev/sdX bs=4M
```

Replace /dev/sdX with the device assigned to your usb drive. Then select the USB device as the boot media on your Jumphost

2b. Install the RDO Release RPM and the opnfv-apex RPM:

```
sudo yum install -y https://www.rdoproject.org/repos/rdo-release.rpm
opnfv-apex-{version}.rpm
```

The RDO Project release repository is needed to install OpenVSwitch, which is a dependency of opnfv-apex. If you do not have external connectivity to use this repository you need to download the OpenVSwitch RPM from the RDO Project repositories and install it with the opnfv-apex RPM.

3. After the operating system and the opnfv-apex RPM are installed, login to your Jumphost as root.
4. Configure IP addresses on the interfaces that you have selected as your networks.
5. Configure the IP gateway to the Internet either, preferably on the public interface.
6. Configure your `/etc/resolv.conf` to point to a DNS server (8.8.8.8 is provided by Google).

9.2 Creating a Node Inventory File

IPMI configuration information gathered in section Execution Requirements (Bare Metal Only) needs to be added to the `inventory.yaml` file.

1. Edit `/etc/apex-opnfv/inventory.yaml`.
2. The nodes dictionary contains a definition block for each baremetal host that will be deployed. 1 or more compute nodes and 3 controller nodes are required. (The example file contains blocks for each of these already). It is optional at this point to add more compute nodes into the node list.
3. Edit the following values for each node:
    - `mac_address`: MAC of the interface that will PXE boot from Instack
    - `ipmi_ip`: IPMI IP Address
    - `ipmi_user`: IPMI username
    - `ipmi_password`: IPMI password
    - `ipmi_type`: Power Management driver to use for the node
    - `cpus`: (Introspected*) CPU cores available
    - `memory`: (Introspected*) Memory available in Mib
    - `disk`: (Introspected*) Disk space available in Gb
    - `arch`: (Introspected*) System architecture
    - `capabilities`: (Optional**) Intended node role (profile:control or profile:compute)

- Introspection looks up the overcloud node's resources and overrides these value. You can

leave default values and Apex will get the correct values when it runs introspection on the nodes.

** If capabilities profile is not specified then Apex will select node's roles in the OPNFV cluster in a non-deterministic fashion.

9.3 Creating the Settings Files

Edit the 2 settings files in /etc/opnfv-apex/. These files have comments to help you customize them.

1. deploy_settings.yaml This file includes basic configuration options deployment.
2. network_settings.yaml This file provides Apex with the networking information that satisfies the prerequisite Network Requirements. These are specific to your environment.

9.4 Running `opnfv-deploy`

You are now ready to deploy OPNFV using Apex! `opnfv-deploy` will use the inventory and settings files to deploy OPNFV.

Follow the steps below to execute:

1. Execute opnfv-deploy `sudo opnfv-deploy [ --flat | -n network_setttings.yaml ] -i instackenv.json -d deploy_settings.yaml` If you need more information about the options that can be passed to opnfv-deploy use `opnfv-deploy --help` –flat will collapse all networks onto a single nic, -n network_settings.yaml allows you to customize your networking topology.
2. Wait while deployment is executed. If something goes wrong during this part of the process, it is most likely a problem with the setup of your network or the information in your configuration files. You will also notice different outputs in your shell.

3. The message "Overcloud Deployed" will display when When the deployment is complete. Just above this message there will be a URL that ends in port http://<host>:5000. This url is also the endpoint for the OPNFV Horizon Dashboard if connected to on port 80.

CHAPTER

TEN

INSTALLATION HIGH-LEVEL OVERVIEW - VIRTUAL DEPLOYMENT

The VM nodes deployment operates almost the same way as the bare metal deploymen t with a few differences. `opnfv-deploy` still deploys an Instack VM. In addition to t he Instack VM a collection of VMs (3 control nodes + 2 compute for an HA deployment or 1 contr ol node and 1 compute node for a Non-HA Deployment) will be defined for the target OPNFV dep loyment. The part of the toolchain that executes IPMI power instructions calls into libvi rt instead of the IPMI interfaces on baremetal servers to operate the power managment. These VMs are then provisioned with the same disk images and configuration that baremetal would be.

To RDO Manager these nodes look like they have just built and registered the sam e way as bare metal nodes, the main difference is the use of a libvirt driver for the pow er management.

CHAPTER

ELEVEN

INSTALLATION GUIDE - VIRTUAL DEPLOYMENT

This section goes step-by-step on how to correctly install and provision the OPNFV target system to VM nodes.

11.1 Install Jumphost

Follow the instructions in the Install Bare Metal Jumphost section.

11.2 Running `opnfv-deploy`

You are now ready to deploy OPNFV! `opnfv-deploy` has virtual deployment capability that includes all of the configuration nessesary to deploy OPNFV with no modifications.

If no modifications are made to the included configurations the target environment will deploy with the following architecture:

- 1 Instack VM
- The option of 3 control and 2 compute VMs (HA Deploy / default) or 1 control and 1 compute VM (Non-HA deploy / pass -n)
- 2 networks, one for provisioning, internal API, storage and tenant networking traffic and a second for the external network

Follow the steps below to execute:

1. `sudo opnfv-deploy --virtual [ --no-ha ]`
2. It will take approximately 30 minutes to stand up instack, define the target virtual machines, configure the deployment and execute the deployment. You will notice different outputs in your shell.
3. When the deployment is complete you will see "Overcloud Deployed"

11.3 Verifying the Setup - VMs

To verify the set you can follow the instructions in the Verifying the Setup section.

Before you get started following these instructions you will need to add IP addresses on the networks that have been created for the External and provisioning networks. By default the External network is 192.168.37.0/24 and the provisioning network is 192.0.2.0/24. To access these networks simply add an IP to brbm and brbm1 and set their link to up. This will provide a route from the hypervisor into the virtual networks acting as OpenStack's underlay network in the virtual deployment.

```
ip addr add 192.0.2.252/24 dev brbm
ip link set up dev brbm
ip addr add 192.168.37.252/24 dev brbm1
ip link set up dev brbm1
```

Once these IP addresses are assigned and the links are up the gateways on the overcloud's networks should be pingable and read to be SSHed to.

```
ping 192.0.2.1
ping 192.168.37.1
```

Now continue with the Verifying the Setup section.

CHAPTER

TWELVE

VERIFYING THE SETUP

Once the deployment has finished, the OPNFV deployment can be accessed via the Instack node. From the jump host ssh to the instack host and become the stack user. Alternativly ssh keys have been setup such that the root user on the jump host can ssh to Instack directly as the stack user.

```
ssh root@192.0.2.1
su - stack
```

Once connected to Instack as the stack user look for two keystone files that can be used to interact with the undercloud and the overcloud. Source the appropriate RC file to interact with the respective OpenStack deployment.

`source stackrc` (undercloud / Instack)
`source overcloudrc` (overcloud / OPNFV)

The contents of these files include the credentials for the administrative user for Instack and OPNFV respectivly. At this point both Instack and OPNFV can be interacted with just as any OpenStack installation can be. Start by listing the nodes in the undercloud that were used to deploy the overcloud.

```
source stackrc
openstack server list
```

The control and compute nodes will be listed in the output of this server list command. The IP addresses that are listed are the control plane addresses that were used to provision the nodes. Use these IP addresses to connect to these nodes. Initial authentication requires using the user heat-admin.

```
ssh heat-admin@192.0.2.7
```

To begin creating users, images, networks, servers, etc in OPNFV source the overcloudrc file or retrieve the admin user's credentials from the overcloudrc file and connect to the web Dashboard.

You are now able to follow the *OpenStack Verification* section.

CHAPTER

THIRTEEN

OPENSTACK VERIFICATION

Once connected to the OPNFV Dashboard make sure the OPNFV target system is working correctly:

1. In the left pane, click Compute -> Images, click Create Image.
2. Insert a name “cirros”, Insert an Image Location `http://download.cirros-cloud.net/0.3.4/cirros-0.3.4-x8`
3. Select format “QCOW2”, select Public, then click Create Image.
4. Now click Project -> Network -> Networks, click Create Network.
5. Enter a name “internal”, click Next.
6. Enter a subnet name “internal_subnet”, and enter Network Address `172.16.1.0/24`, click Next.
7. Now go to Project -> Compute -> Instances, click Launch Instance.
8. Enter Instance Name “first_instance”, select Instance Boot Source “Boot from image”, and then select Image Name “cirros”.
9. Click Launch, status will cycle though a couple states before becoming “Active”.
10. Steps 7 though 9 can be repeated to launch more instances.
11. Once an instance becomes “Active” their IP addresses will display on the Instances page.
12. Click the name of an instance, then the “Console” tab and login as “cirros”/”cubswin:)”
13. To verify storage is working, click Project -> Compute -> Volumes, Create Volume
14. Give the volume a name and a size of 1 GB
15. Once the volume becomes “Available” click the dropdown arrow and attach it to an instance.

Congratulations you have successfully installed OPNFV!

CHAPTER

FOURTEEN

FREQUENTLY ASKED QUESTIONS

CHAPTER

FIFTEEN

LICENSE

All Apex and “common” entities are protected by the Apache 2.0 License.

CHAPTER

SIXTEEN

REFERENCES

16.1 OPNFV

OPNFV Home Page

OPNFV Genesis project page

OPNFV Apex project page

16.2 OpenStack

OpenStack Liberty Release artifacts

OpenStack documentation

16.3 OpenDaylight

Upstream OpenDaylight provides a number of packaging and deployment options meant for consumption by downstream projects like OPNFV.

Currently, OPNFV Apex uses OpenDaylight's Puppet module, which in turn depends on OpenDaylight's RPM.

16.4 RDO Manager

RDO Manager website

Authors Tim Rozet (trozet@redhat.com)

Authors Dan Radez (dradez@redhat.com)

Version 1.0

CHAPTER

SEVENTEEN

INDICES AND TABLES

- search