SDNVPN Colorado documentation

Release draft (ab41fa1)

OPNFV

August 08, 2016

CONTENTS

CHAPTER

ONE

SDNVPN PLATFORM COMPONENTS AND FEATURES

The SDN VPN feature enhances OPNFV's baseline OpenStack deployment with the possibility to configure BGP based VPNs according to the OpenStack Neutron Stadium project BGPVPN. The BGPVPN project consists of an API specification, framework implementation and a number of backend drivers (Bagpipe, OpenContrail, Nuage and OpenDaylight currently). In OPNFV Brahmaputra only the ODL backend is supported.

This document will give the user instructions on how to deploy the SDN VPN scenarios verfied for the Colorado release of the OPNFV platform, using the Fuel installer.

A sister document covers installation using the APEX installer.

CHAPTER TWO

INTRODUCTION

This document provides guidelines on how to install and configure the os-odl_l2_bgpvpn_ha and os-odl_l2_bgpvpn_ha scenarios of OPNFV including required software and hardware configurations.

Description of bgpvpn scenarios Internal transport tunnel mesh Install Neutron BGPVPN additions (networking-bgpvpn) Neutron odl additions (networking-odl) install and configure Quagga (incl. config on ODL side) configure OVS to connect to ODL and set up the right bridges (network architecture) set up iptables to allow connections between OVS and ODL set up HA proxy so that ODL can be reached

CHAPTER

THREE

HARDWARE REQUIREMENTS

3.1 Bare metal deployment on Pharos Lab

Hardware requirements for bare-metal deployments of the OPNFV infrastucture are specified by the Pharos project. The Pharos project provides an OPNFV hardware specification for configuring your hardware at: http://artifacts.opnfv.org/pharos/docs/pharos-spec.html.

3.2 Virtual deployment hardware requirements

To perform a virtual deployment of an OPNFV scenario on a single host, that host has to meet the hardware requirements outlined in the <missing spec>.

3.3 Additional Hardware requirements

Since Opendaylight is running on the controller it is recommended to give more resources to the controller virtual machines. Our recommendation is to have +2 virtual cores and

> +8 virtual memory. Together with the commonly used recommendation this sums up to: 4 virtual cores
> 16 GB

See in Installation section how to configure this.

CHAPTER

FOUR

PREPARING YOUR JUMPHOST TO INSTALL FUEL BY SCRIPT

Before starting the installation of the <scenario> scenario some preparation of the machine that will host the Fuel VM must be done.

4.1 Setting up the jumphost

4.2 Installation of required packages

To be able to run the installation of the basic opnfv fuel installation the Jumphost (or the host which serves the VMs for the virtual deployment) needs to install the following packages:

```
sudo apt-get install -y git make curl libvirt-bin libpq-dev qemu-kvm qemu-system tightvncserver virt-manager sshpass fuseiso genisoimage blackbox xterm python-pip python-git python-dev python-oslo.config python-pip python-dev libffi-dev libxml2-dev libxslt1-dev libffi-dev libxml2-dev libxslt1-dev expect curl python-netaddr p7zip-full sudo pip install GitPython pyyaml netaddr paramiko lxml scp python-novaclient python-neutronclient python-glanceclient python-keystoneclient debtcollector netifaces enum
```

4.3 Download the source code and artifact

To be able to install the scenario os-odl_l2-bgpvpn one can follow the way CI is deploying the scenario. First off all the opnfv-fuel repo needs to be cloned:

```
git clone ssh://<user>@gerrit.opnfv.org:29418/fuel
```

This command downloads the whole repo fuel. We need now to switch it to he stable Brahmaputra branch:

```
cd fuel git checkout stable/brahmaputra
```

Now download the appropriate OPNFV Fuel ISO into an appropriate folder: wget http://artifacts.opnfv.org/fuel/brahmaputra/opnfv-brahmaputra.3.0.iso

The ISO version may change. Check https://www.opnfv.org/opnfv-brahmaputra-fuel-users to get the latest ISO.

CHAPTER

FIVE

FUEL INSTALLATION AND SCENARIO DEPLOYMENT

This section describes the installation of the os-odl_l2-bgpvpn-ha or os-odl_l2-bgpvpn-noha OPNFV reference platform stack across a server cluster.

5.1 Scenario Preparation

dea.yaml and dha.yaml need to be copied and changed to the purpose of the lap where you deploy. Copy the full lap config from:

> <path-to-opnfv-fuel-repo>/deploy/config/labs/devel-pipeline/elx to <path-to-opnfv-fuel-repo>/deploy/config/labs/devel-pipeline/<your-lab-name>

Add at the bottom of dha.yaml:

> **disks:** fuel: 100G controller: 100G compute: 100G
>
> # Here the infrastructure VMs can be defined. # The entries are not mandatory! If it is left empty # the default defined in deploy/templates will # be used. define_vms:
>
> > **controller:**
> >
> > > **vcpu:** value: 4
> > >
> > > **memory:**
> > >
> > > > **attribute_equlas:** unit: KiB
> > > >
> > > > value: 16388608
> > >
> > > **currentMemory:**
> > >
> > > > **attribute_equlas:** unit: KiB
> > > >
> > > > value: 16388608

Check if dea.yaml contains all your needed changes.

5.2 Installation procedures

We describe several alternative procedures in the following. Go to

> cd <opnfv-fuel-repo>/ci

5.2.1 Full automatic virtual deployment High Availablity Mode

```
sudo bash ./deploy.sh -b file://<path-to-opnfv-fuel-repo>/config/ -l devel-pipeline -p <your-lab-name> -s
os-odl_l2-bgpvpn-ha -i file://<path-to-fuel-iso>
```

5.2.2 Full automatic virtual deployment NO High Availablity Mode

```
sudo bash ./deploy.sh -b file://<path-to-opnfv-fuel-repo>/config/ -l devel-pipeline -p <your-lab-name> -s
os-odl_l2-bgpvpn-noha -i file://<path-to-fuel-iso>
```

5.2.3 Automatic Fuel installation and manual scenario deployment

```
sudo bash ./deploy.sh -b file://<path-to-opnfv-fuel-repo>/config/ -l devel-pipeline -p <your-lab-name> -s
os-odl_l2-bgpvpn-ha -i file://<path-to-fuel-iso> -e
```

With -e option the installer does not launch environment deployment, so a user can do some modification before the scenario is realy deployed. Another interesting option is the -f option which deploys the scenario on existing Fuel.

CHAPTER

SIX

REFERENCES

6.1 OPNFV

1. OPNFV Home Page
2. OPNFV documentation- and software downloads

6.2 OpenStack

3. OpenStack Liberty Release artifacts
4. OpenStack documentation

6.3 OpenDaylight

5. OpenDaylight artifacts

6.4 Fuel

6. The Fuel OpenStack project
7. Fuel documentation overview
8. Fuel planning guide
9. Fuel quick start guide
10. Fuel operations guide
11. Fuel Plugin Developers Guide
12. Fuel OpenStack Hardware Compatibility List

6.5 Fuel in OPNFV

13. OPNFV Installation instruction for the Brahmaputra release of OPNFV when using Fuel as a deployment tool
14. OPNFV Build instruction for the Brahmaputra release of OPNFV when using Fuel as a deployment tool

15. OPNFV Release Note for the Brahmaputra release of OPNFV when using Fuel as a deployment tool

CHAPTER

SEVEN

CONFIGURING SDNVPN FEATURES

Fuel installer configuration

In order to install the BGPVPN feature, the corresponding checkbox in Fuel has to be selected. This will trigger installation of the OpenStack BGPVPN API extension for Neutron (set up for using the ODL driver).

In addition, ODL has to be installed, see the corresponding section in the respective installer documentation on how to install ODL. If the BGPVPN feature is installed, ODL will automatically be installed with VPN Service karaf feature activated.

No post-deploy configuration is necessary. The Fuel BGPVPN plugin and the ODL plugin should set up the cluster ready for BGPVPNs being created. This includes the set-up of internal VxLAN transport tunnels between compute nodes.

No post-configuration activities are required.

CHAPTER

EIGHT

SDN VPN CAPABILITIES AND USAGE

The BGPVPN feature enables creation of BGP VPNs according to the OpenStack BGPVPN blueprint at https://blueprints.launchpad.net/neutron/+spec/neutron-bgp-vpn. In a nutshell, the blueprint defines a BGPVPN object and a number of ways how to associate it with the existing Neutron object model, including a unique definition of the related semantics. The BGPVPN framework supports a backend driver model with currently available drivers for Bagpipe, OpenContrail, Nuage and OpenDaylight.

Currently, in OPNFV only ODL is supported as a backend for BGPVPN. API calls are mapped onto the ODL VPN Service REST API through the BGPVPN ODL driver and the ODL Neutron Northbound module.

8.1 Feature and API usage guidelines and example

For the details of using OpenStack BGPVPN API, please refer to the documentation at http://docs.openstack.org/developer/networking-bgpvpn/.

CHAPTER

NINE

INDICES

- search