INSTALLATION RUNBOOK FOR
Huawei Technologies Co., Ltd + Huawei Volume Driver

Product Name:	**Huawei Volume Driver**
Driver Version:	**Mitaka**
MOS Version:	**9.0**
OpenStack Version:	**Mitaka**
Product Type:	**Cinder Driver**

Document History

Version	Revision Date	Description
0.1	12-02-2016	Initial Version
0.2	3-20-2017	Modify the part of cinder configuration
0.3	07-24-2017	Add support for Huawei OceanStor Dorado V3 series storage.

1. Introduction

This document is to serve as a detailed Deployment Guide for Huawei Volume driver with Mirantis OpenStack 9.0. Huawei Technologies Co., Ltd offers Huawei Volume Driver to support iSCSI and Fibre Channel connections and enables Huawei Storages such as, OceanStor Dorado V3 series, OceanStor V3 series storage, OceanStor TV2 storage, and OceanStor 18000 high-end storage to provide block storage services for OpenStack cloud environment. This document describes the reference architecture, installation steps for certified MOS and Huawei Volume Driver, limitations and testing procedures.

1.1 Target Audience

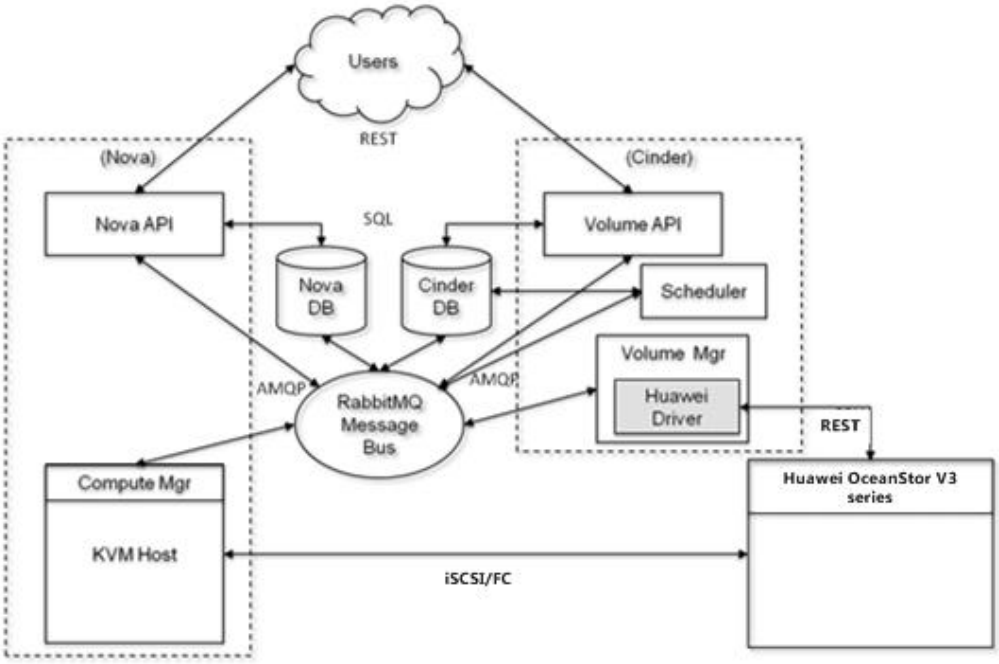
This documentation is intended for OpenStack administrators provided with the details of the setup, configuration and tests run on Mirantis OpenStack with Huawei Volume driver for the purpose of Unlocked OpenStack Driver Validation. It assumes that you have experience with network and cloud concepts.

2. Product Overview

Huawei Volume Driver enables integration between OpenStack Cloud and Huawei Storages such as, OceanStor Dorado V3 series, OceanStor V3 series storage, OceanStor TV2 storage, and OceanStor 18000 high-end storage to provide block storage services for OpenStack.
Huawei Volume Driver implemented OpenStack Cinder accesses Huawei storage service APIs, such as volume, snapshot, Qos etc and supports various operations.

3. Joint reference architecture

Huawei Volume Driver architecture. Prints in red modules are implemented by Huawei Volume Driver.

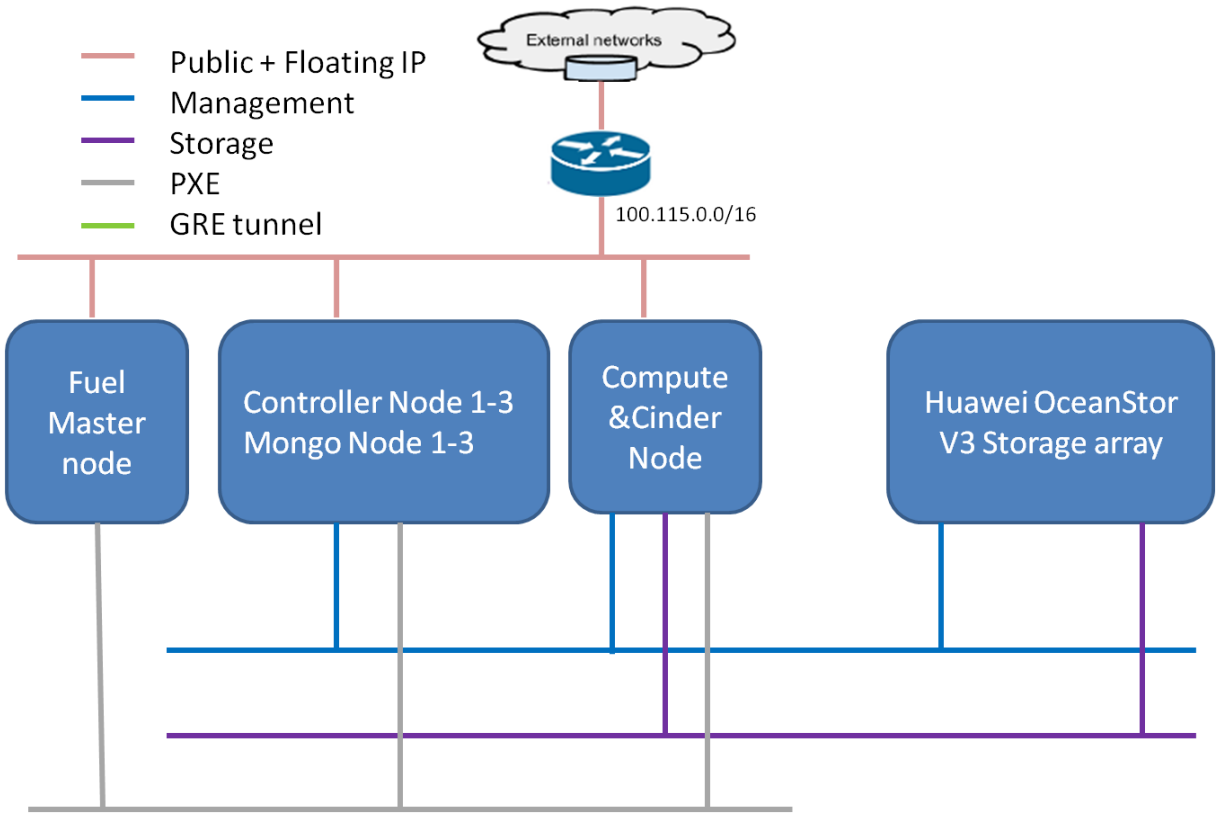


4. Physical and Logical Network Topology

It is highly recommended that the storage network is broken out as either a physical interface, or as a separate VLAN, to keep the traffic segregated from other traffic types. If you have enough physical interfaces, we strongly recommend using a separate physical interface for storage network.

Each Server at least 2 NICs are required.

- eth0 - untagged port for PXE network (Fuel network), Management (VLAN 101), Public/Floating (VLAN 102)
- eth1 - port for networks: Storage (untagged)



5. Installation and Configuration

5.1 Overview of MOS installation steps

- ◇ Setup OpenStack Network configuration in Switch device.
- ◇ Install Fuel Master Node on Physical Server. Customer can use a VM for fuel master by following the MOS user guide. In this demo, we use a physical server instead.
- ◇ Deploy MOS Slave nodes on Fuel UI
- ◇ Run Fuel Health Check, make sure all cases pass.
- ◇ Manually integrated Huawei OceanStor V3 storage and Huawei Volume driver with MOS.
- ◇ Run Fuel Heath Check test on Fuel UI
- ◇ Verify Huawei Volume Driver basic functions on MOS, plus additional manual/automated acceptance tests.

5.2 MOS Installation in details

5.2.1 Hardware and environment

The following hardware devices are used in this demo.

Name	Device	Count
Server	Huawei RH2288-v2	2
	Huawei RH2285	1
Network Switch	Huawei Quidway S3300	1
Storage	Huawei OceanStor 5800 V3	1

5.2.2 Switch configuration

Use the following configuration to deploy Mirantis OpenStack using Huawei Quidway S3300 network switch.

For one master node and four Openstack nodes of physical servers, each server have two NICs with following VLAN IDs are used:

- eth0 - untagged port for PXE network (Fuel network), Management (VLAN 101), Public/Floating (VLAN 102)
- eth1- port for networks: Storage (untagged)

The connection between switch port and server NIC as below: each switch port used for one sever NIC.

Switch Port	Server name	Server NIC
E0/0/1	Fuel Master	eth0
E0/0/2	Controller Node 1	eth0
E0/0/3	Compute&Cinder Node 1	eth0
E0/0/6	Fuel Master	eth1
E0/0/7	Controller Node 1	eth1
E0/0/8	Compute&Cinder Node 1	eth1

The details port configuration for Huawei Quidway S3300 network switch is below:

```
#
!Software Version V100R005C01SPC100
sysname openstack_sw
#
vlan batch 101 to 102 1000 to 1030
#
vlan 1
interface Vlanif1
ip address 100.115.114.254 255.255.0.0
#
interface Ethernet0/0/1
port link-type trunk
port trunk allow-pass vlan 101 to 102 1000 to 1030
ntdp enable
ndp enable
bpdu enable
#
interface Ethernet0/0/2
port link-type trunk
port trunk allow-pass vlan 101 to 102 1000 to 1030
ntdp enable
ndp enable
bpdu enable
dhcp snooping trusted
#
```

interface Ethernet0/0/3
port link-type trunk
port trunk allow-pass vlan 101 to 102 1000 to 1030
ntdp enable
ndp enable
bpdu enable
#
interface Ethernet0/0/6
ntdp enable
ndp enable
bpdu enable
#
interface Ethernet0/0/7
ntdp enable
ndp enable
bpdu enable
#
interface Ethernet0/0/8
ntdp enable
ndp enable
bpdu enable

5.2.3 Install Fuel Master Node

◇ Mount MirantisOpenStack-9.0.iso through IPMI on one Huawei RH2285 server and power the machine on. Set the boot order for the system with the Huawei DVD media.
Using the Fuel User guide, go through the installation process until Fuel Setup will appear. Enter into it.

◇ Navigate to Network Setup menu, adjust configuration to one provided in the table below and then click 'Check' button to ensure the configuration is valid.

Interface	IP address	Netmask	Default Gateway
eth0	100.115.114.14	255.255. 0.0	100.115.0.1
eth1(PXE)	192.168.10.10	255.255.255.0	-

◇ Navigate to PXE Setup menu, adjust configuration to one provided in the table below and then click 'Check' button to ensure the configuration is valid.

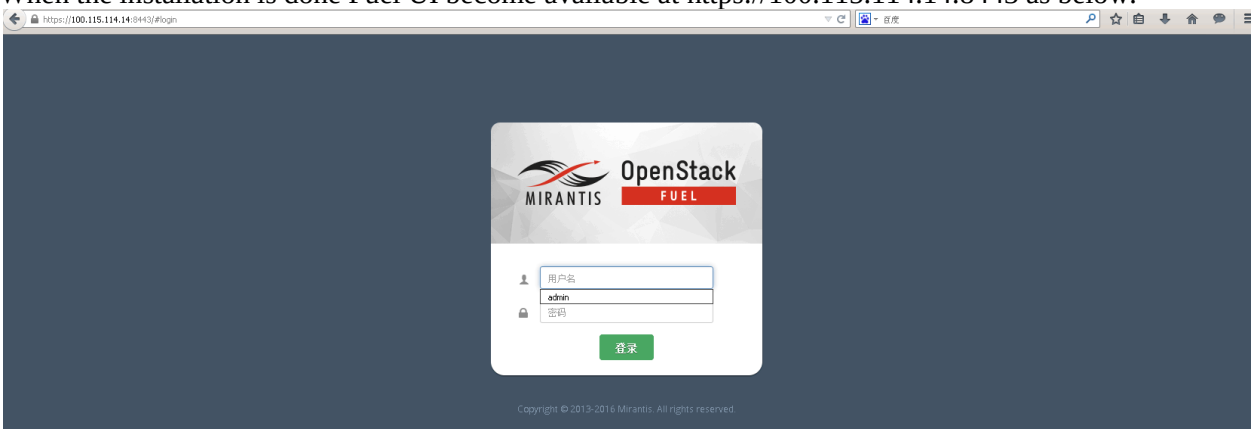
Interface	DHCP Pool Start	DHCP Pool End	DHCP Gateway
eth1(PXE)	192.168.10.110	192.168.10.254	192.168.10.10

◇ Navigate to DNS & Hostname menu, adjust configuration to one provided in the table below and then click 'Check' button to ensure the configuration is valid. It will also check that the Internet is reachable by the Fuel node because it will be required for the MOS installation later.

Hostname	Domain	Search Domain	External DNS	Hostname to test DNS

fuel	Domain.tld	Domain.tld	8.8.4.4, 8.8.8.8	baidu.com

- ◇ Keep Time Sync menu untouched with default settings entered. Click to 'Check' button to ensure NTP servers are reachable.
- ◇ Change default root password to make sure Fuel Health Check cases pass.
- ◇ Save and Quit Setup and boot the Fuel Master Node.
- ◇ When the installation is done Fuel UI become available at https://100.115.114.14:8443 as below.



5.3 Creation of OpenStack environment

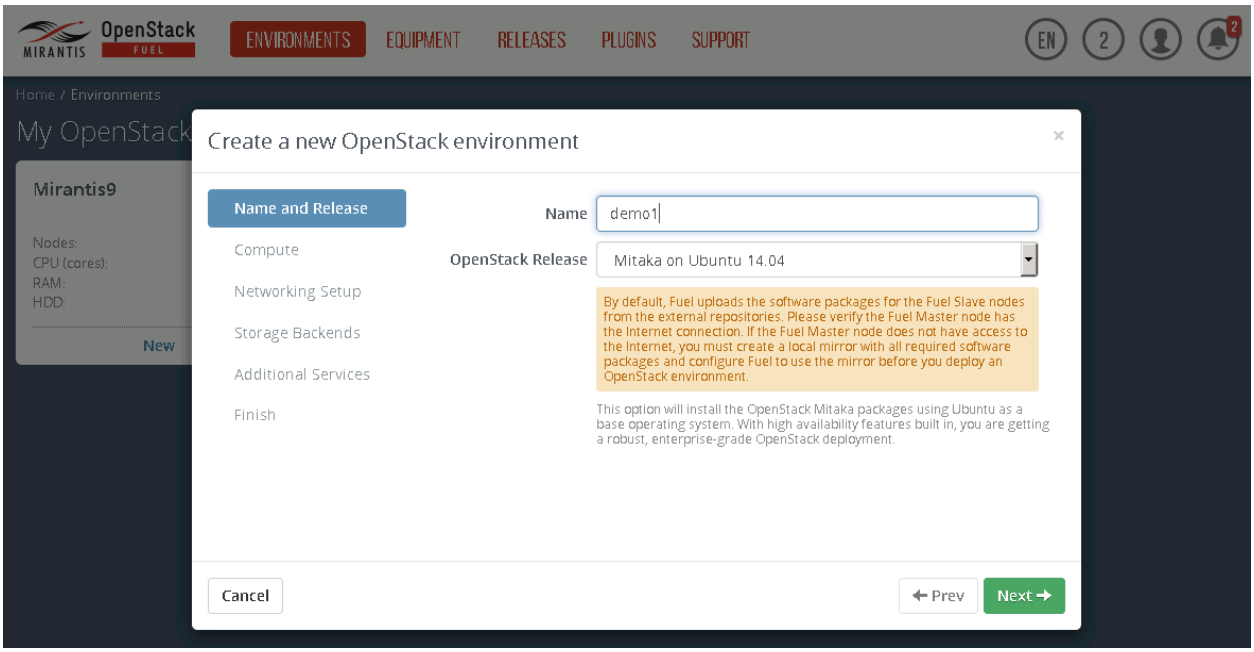
The settings for the OpenStack cloud in this demo are below:

- Name: Mirantis9.0
- OpenStack release: Mitaka on Ubuntu 14.04
- Deployment mode: Single (one controller)
- Compute: KVM
- Cinder backend: Cinder LVM over iSCSI for volumes (Use default providers)
- Glance backend: Swift (Use default providers)
- Network setup: Neutron with VLAN segmentation
- Ceilometer: Not Install, please install them if need
- Murano and Sahara: Not Install, please install them if need.

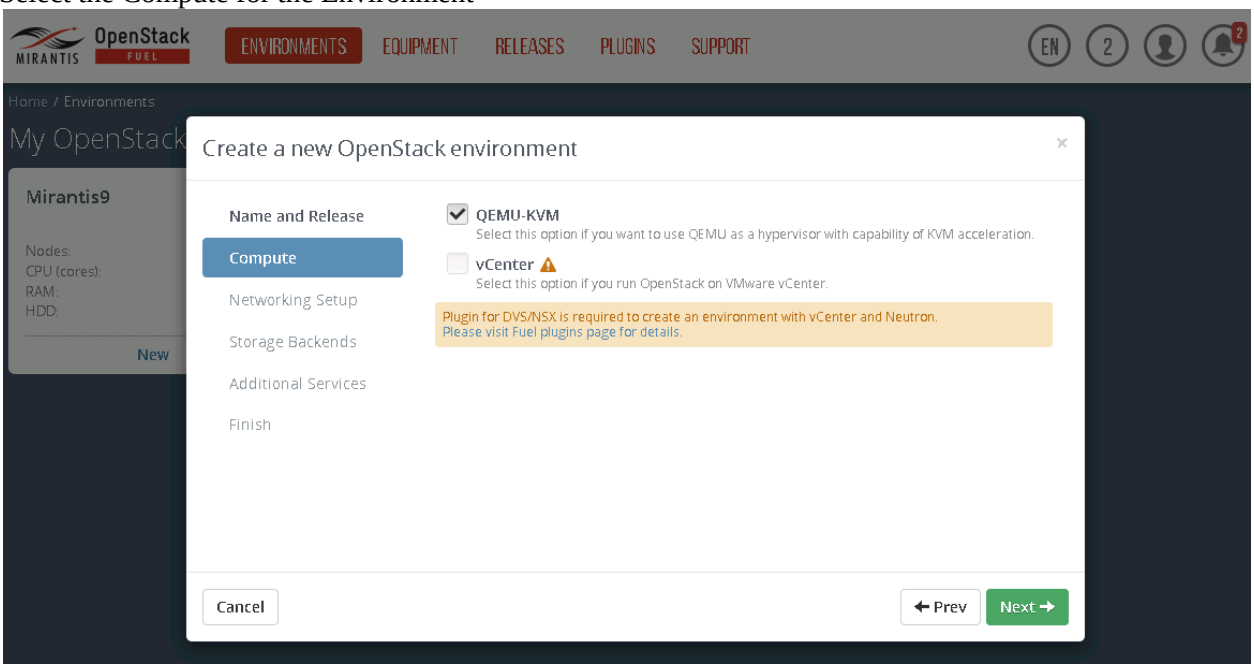
Note:

For creating OpenStack environment, please find the detail description in MOS quickstart guide: https://docs.mirantis.com/openstack/fuel/fuel-9.0/quickstart-guide.html#quickstart-guide

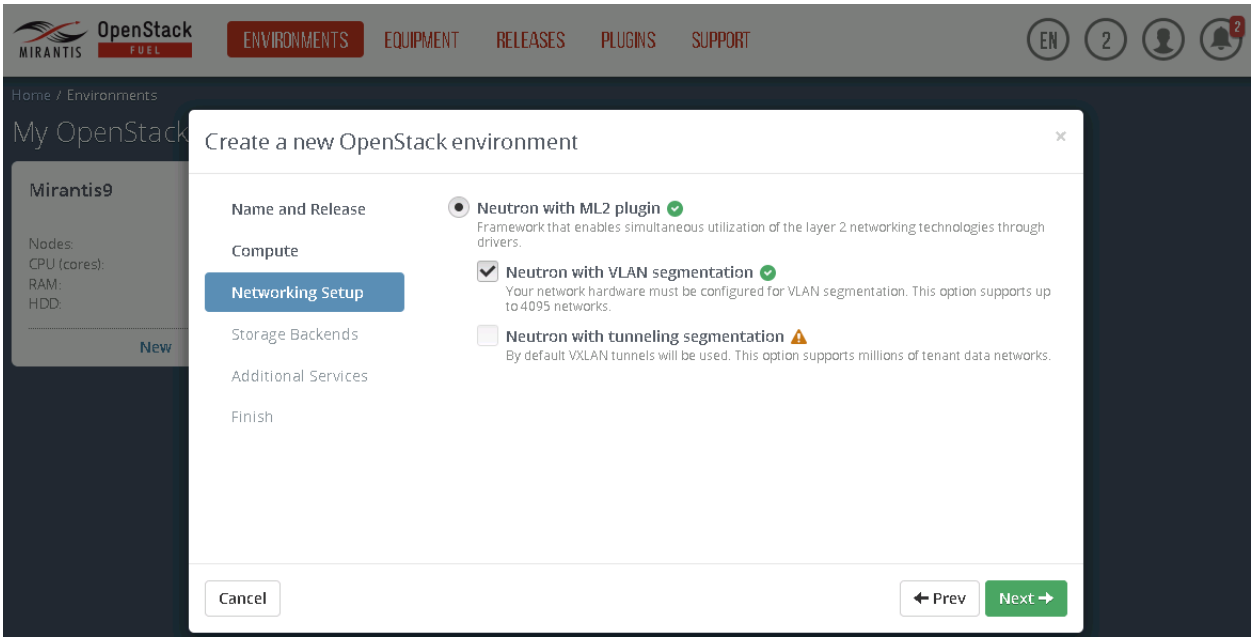
1) On the Fuel UI, click on "New OpenStack Environment".
2) When the wizard opens, enter the name and the desired OpenStack Release.



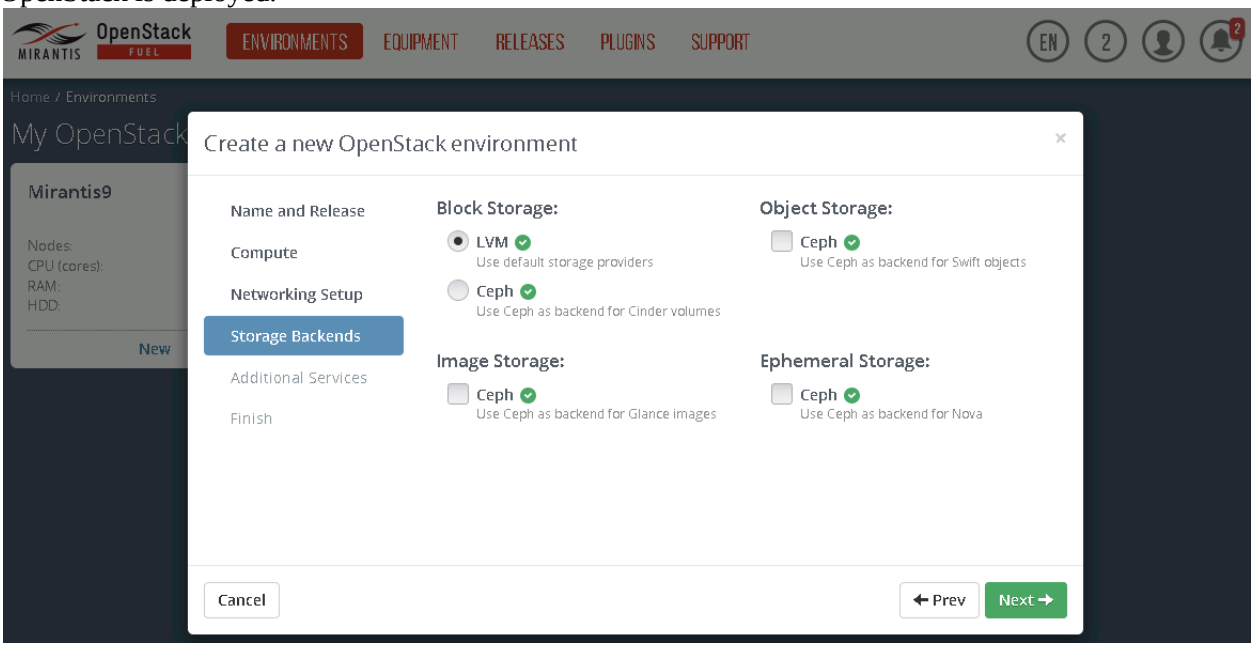
3) Select the Compute for the Environment



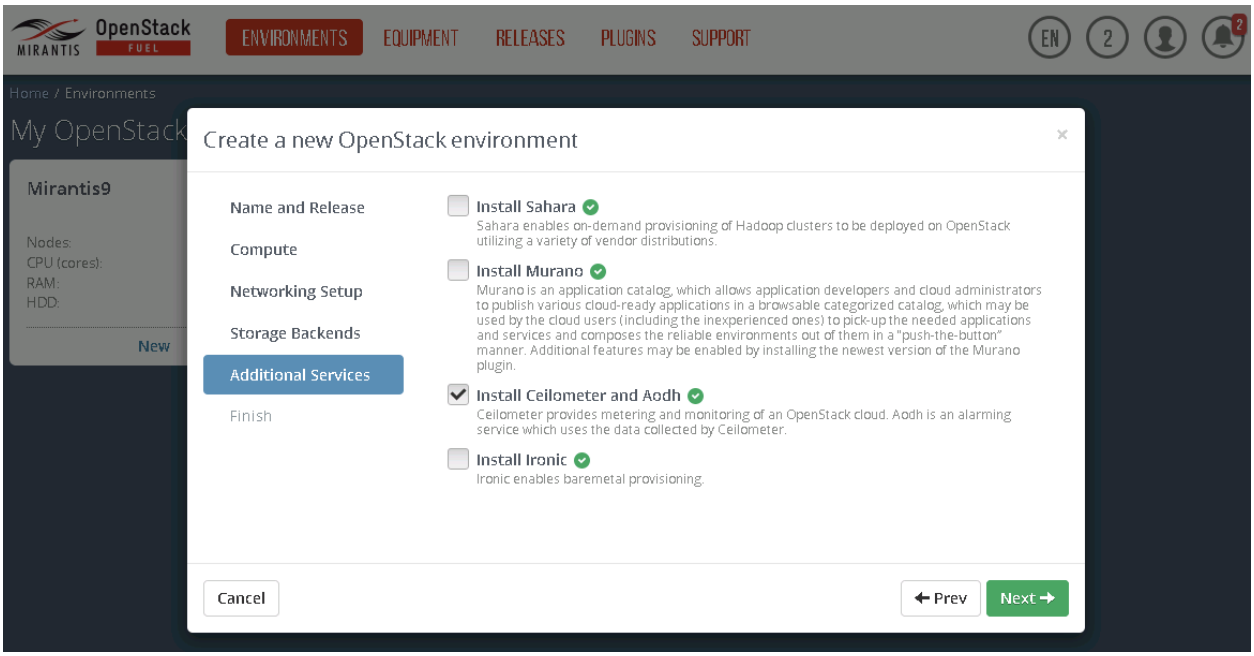
4) Select the required Neutron Setup.



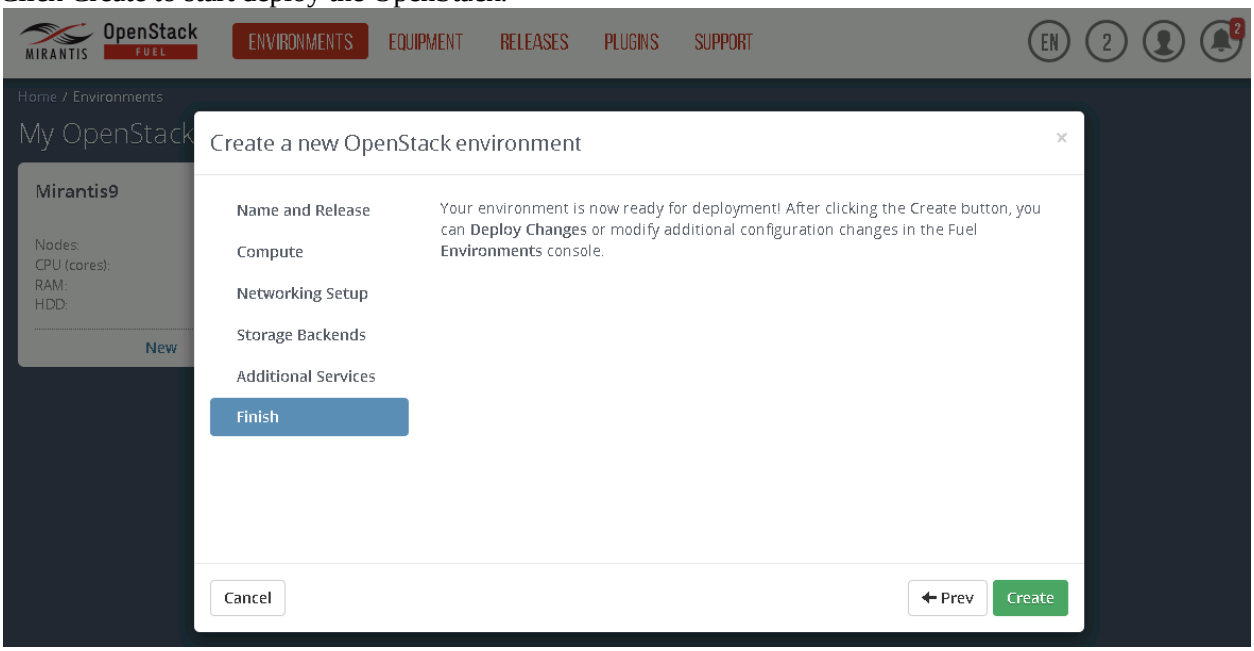
5) Under Storage Backends, leave the default option. Huawei Volume driver can be installed after the OpenStack is deployed.



6) Select the additional services and click on next.



7) Click Create to start deploy the OpenStack.



5.4 MOS Installation

5.4.1 Add the bootstrap source

root@node-6:~# fuel-bootstrap build –activate

5.4.2 Make the mirror source

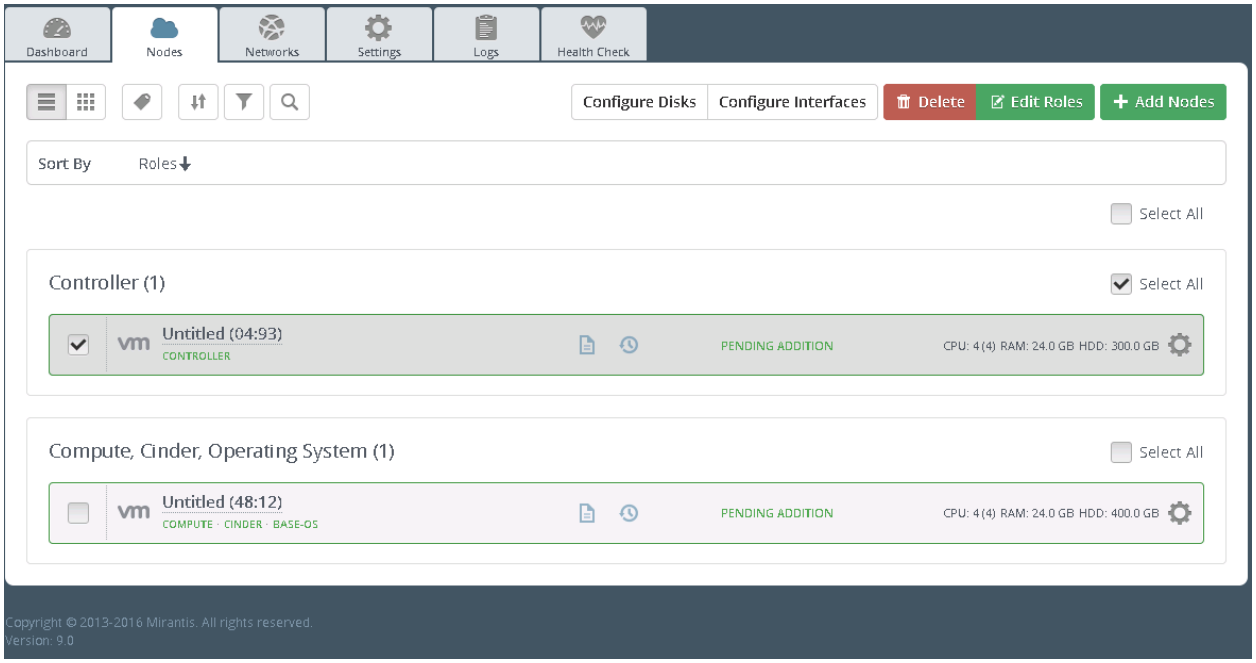
root@node-6:~# fuel-createmirror

5.4.3 Nodes

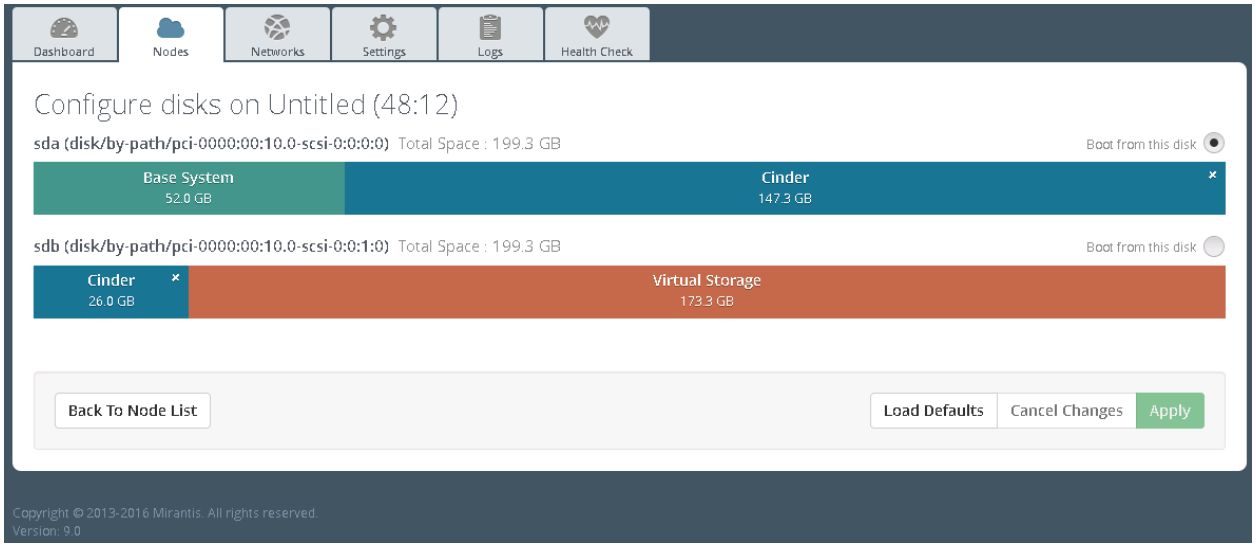
All nodes: 1 Controller nodes, and 1 Compute node combined with Storage-Cinder.
For this demo, hardware model information as follow:

Controller Node 1	Compute Node
RH2288-v2	RH2288-v2

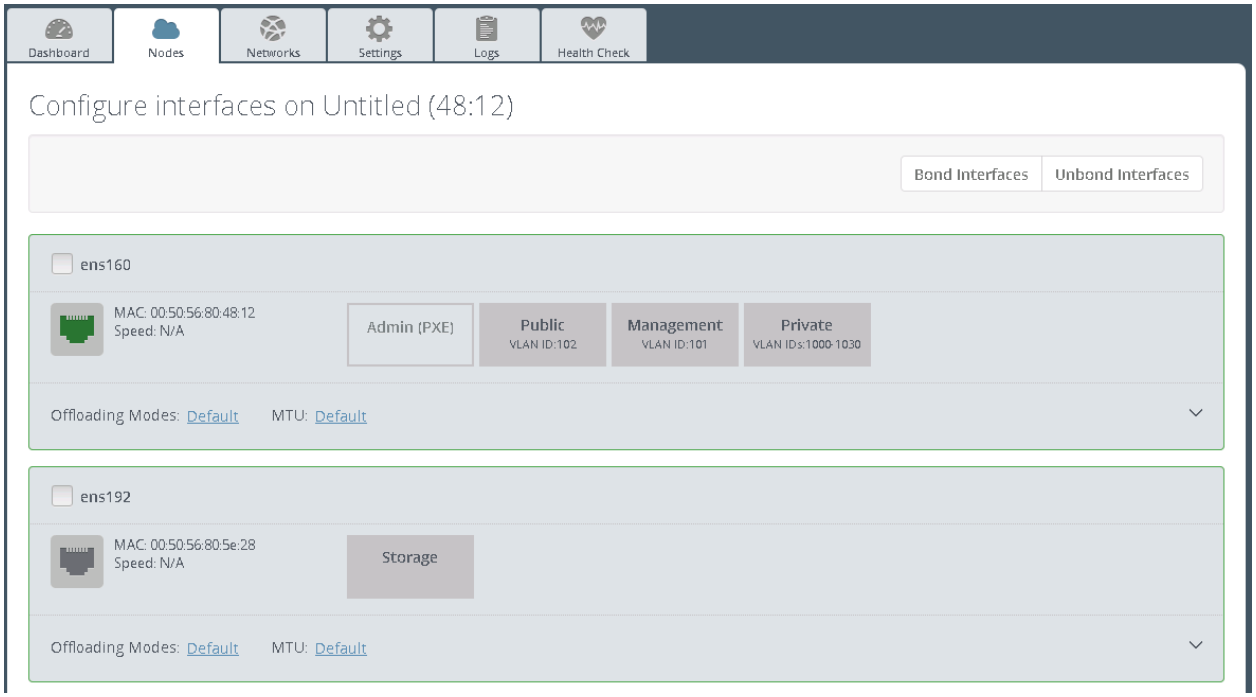
Once all the nodes have booted up through PXE, they appear on the Fuel UI. Configure controllers, cinder and compute.



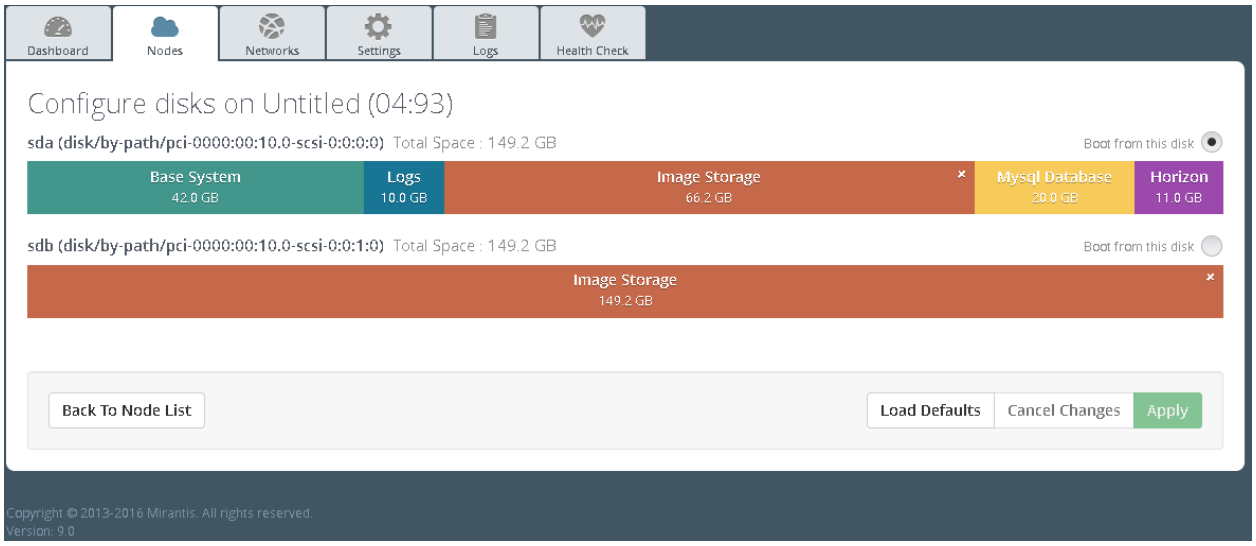
Compute Node (48:12) disks configuration:



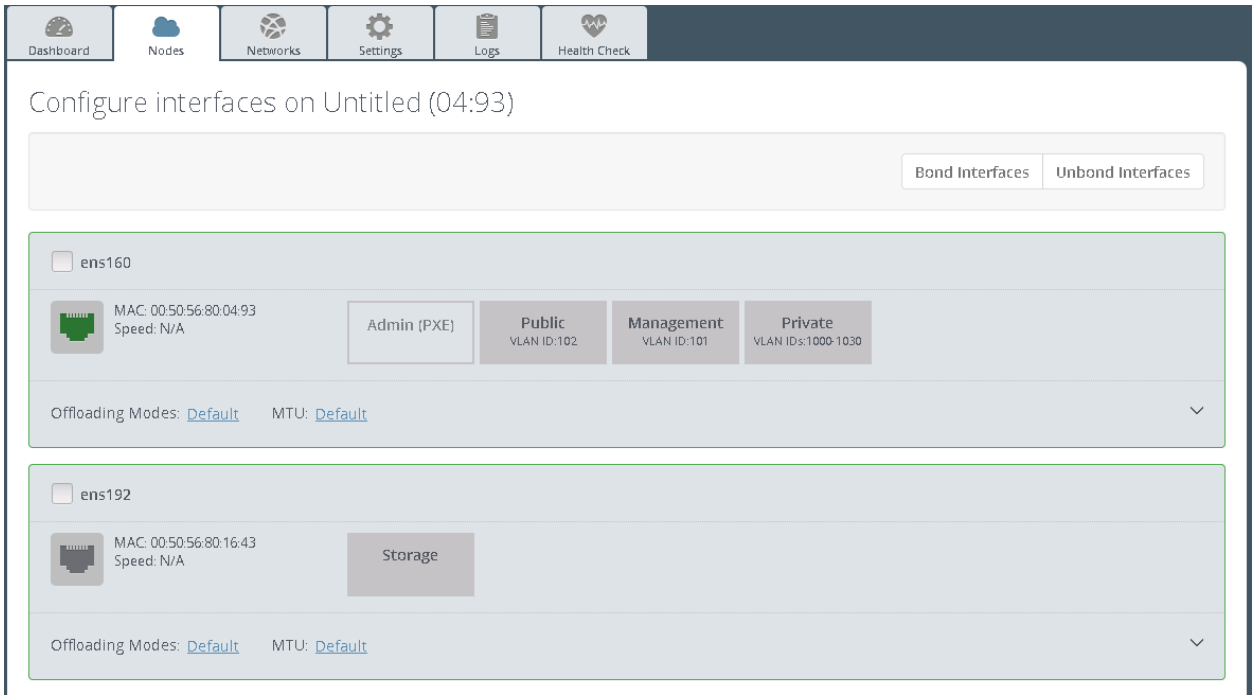
Compute Node (48:12) network interfaces configuration:



Controller node (04:93) disks configuration:

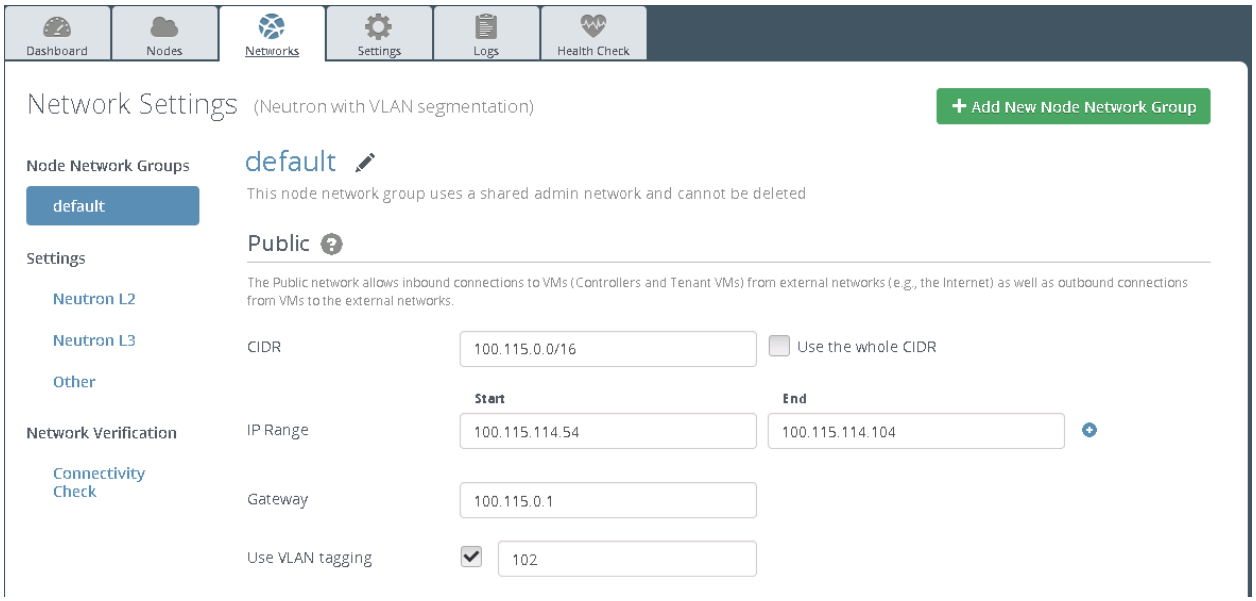


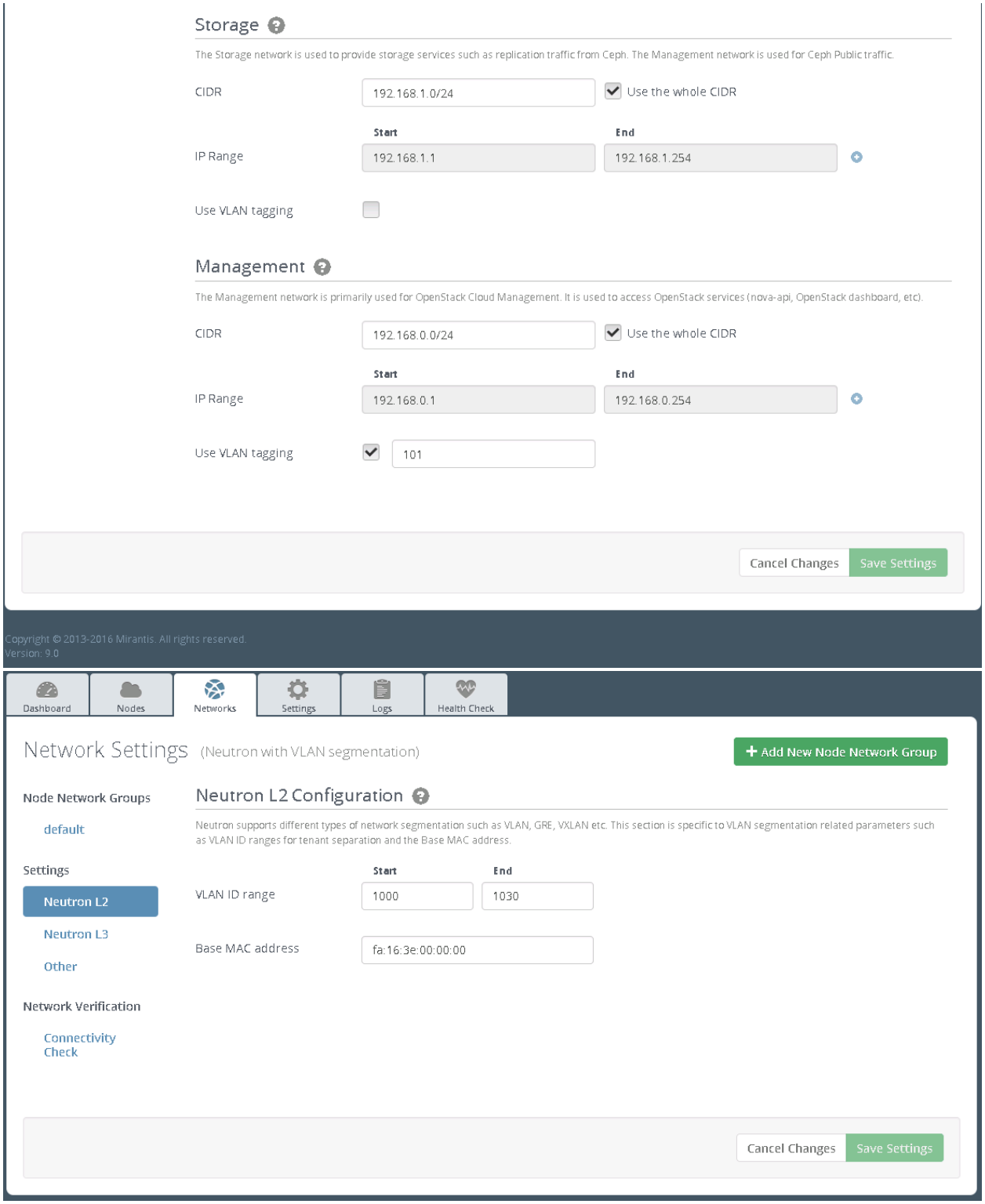
Controller node (04:93) network Interfaces configuration:

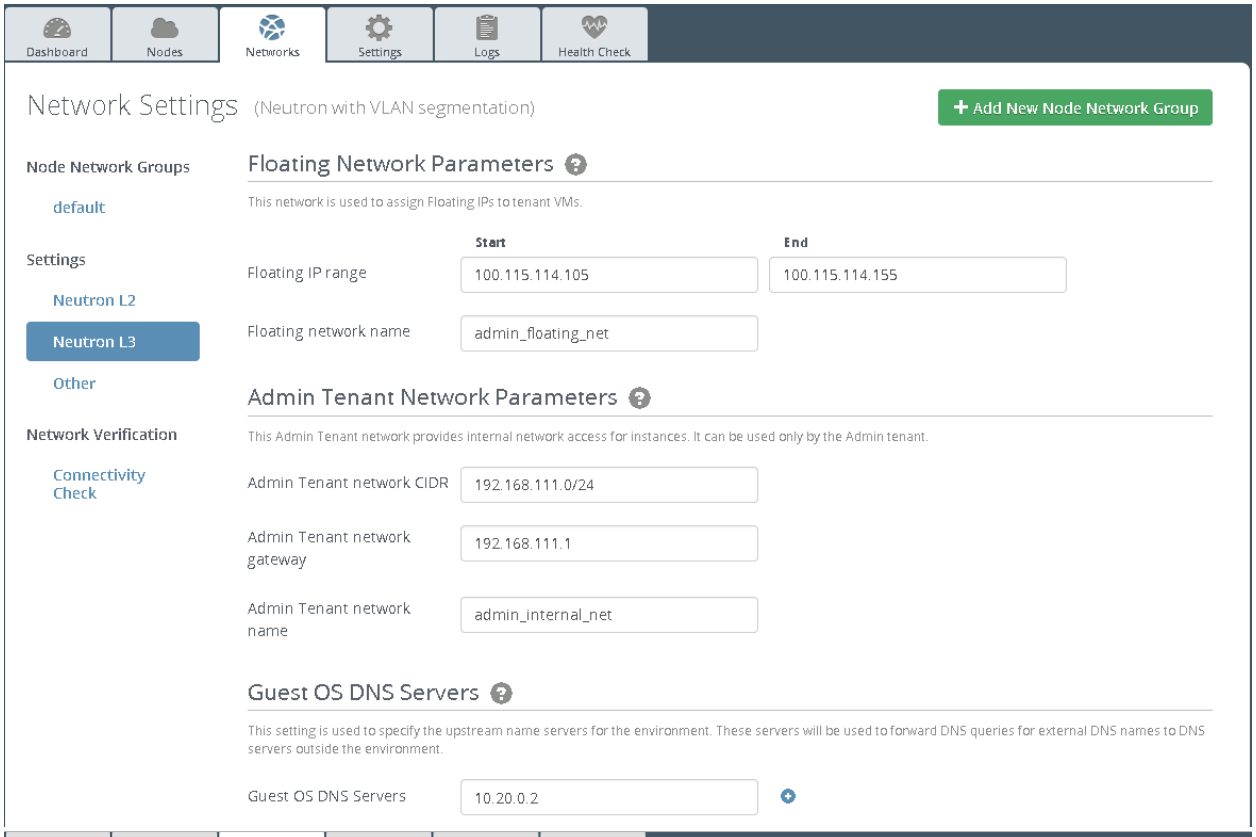


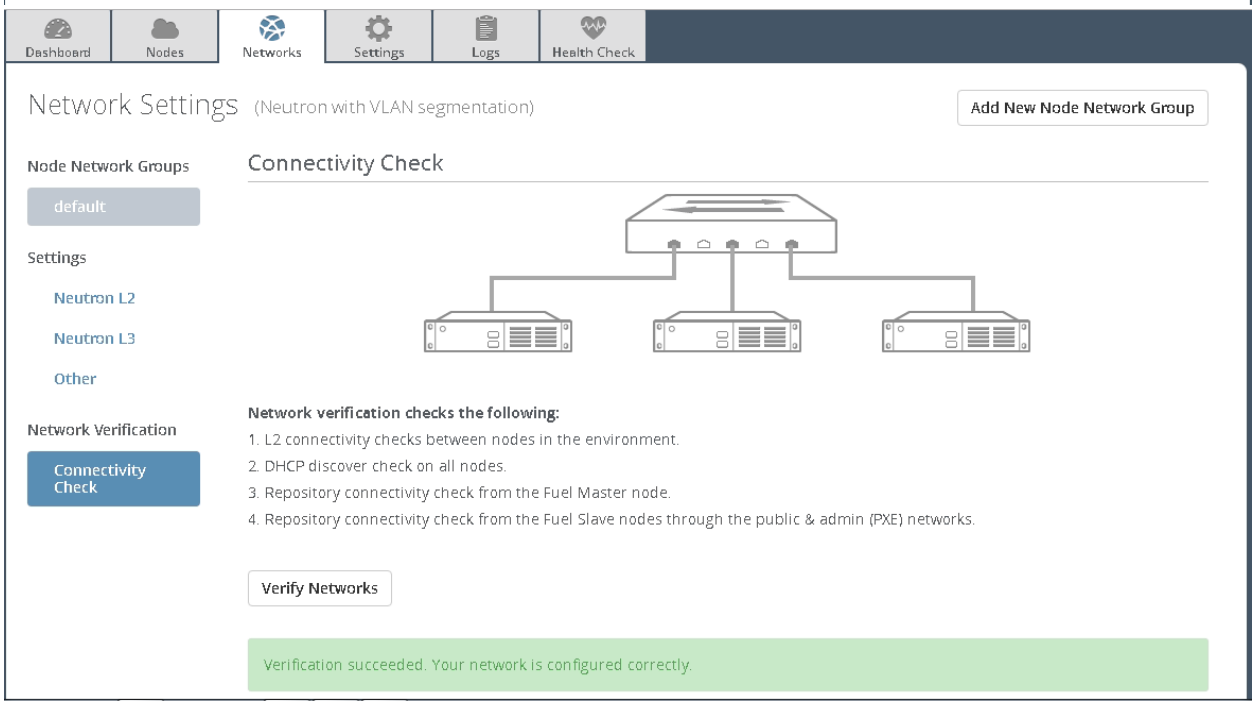
5.4.4 Network Setting

Change the Public, Storage, Management and Private network as your plan. After configuring the nodes, run “Verify Networks” under Networks tab. The test should pass.




Storage
The Storage network is used to provide storage services such as replication traffic from Ceph. The Management network is used for Ceph Public traffic.
CIDR
192.168.1.0/24
Use the whole CIDR
Start
End
IP Range
192.168.1.1
192.168.1.254
Use VLAN tagging
Management
The Management network is primarily used for OpenStack Cloud Management. It is used to access OpenStack services (nova-api, OpenStack dashboard, etc).
CIDR
192.168.0.0/24
Use the whole CIDR
Start
End
IP Range
192.168.0.1
192.168.0.254
Use VLAN tagging
101
Cancel Changes
Save Settings
Copyright © 2013-2016 Mirantis. All rights reserved.
Version: 9.0
Dashboard
Nodes
Networks
Settings
Logs
Health Check
Network Settings
(Neutron with VLAN segmentation)
+ Add New Node Network Group
Node Network Groups
default
Settings
Neutron L2
Neutron L3
Other
Network Verification
Connectivity Check
Neutron L2 Configuration
Neutron supports different types of network segmentation such as VLAN, GRE, VXLAN etc. This section is specific to VLAN segmentation related parameters such as VLAN ID ranges for tenant separation and the Base MAC address.
Start
End
VLAN ID range
1000
1030
Base MAC address
fa:16:3e:00:00:00
Cancel Changes
Save Settings



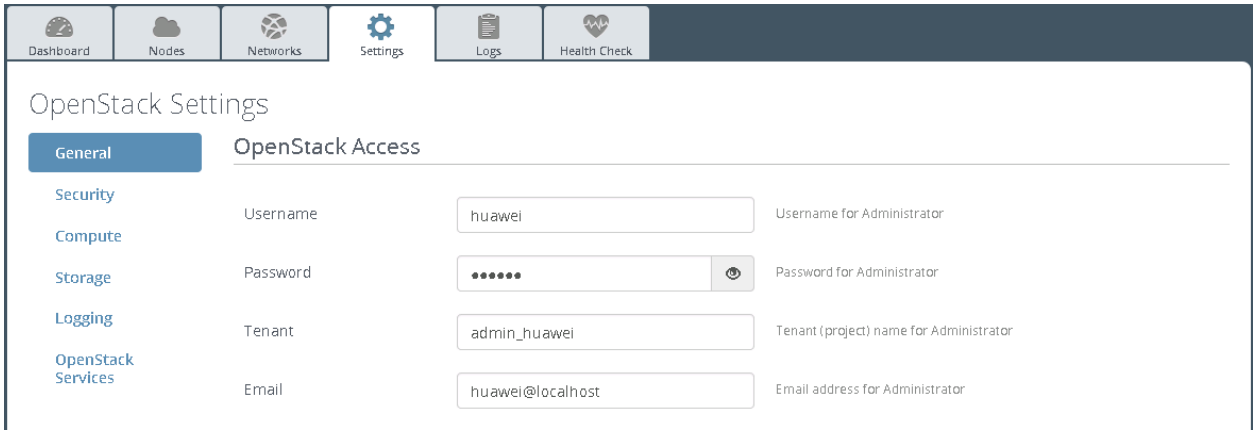


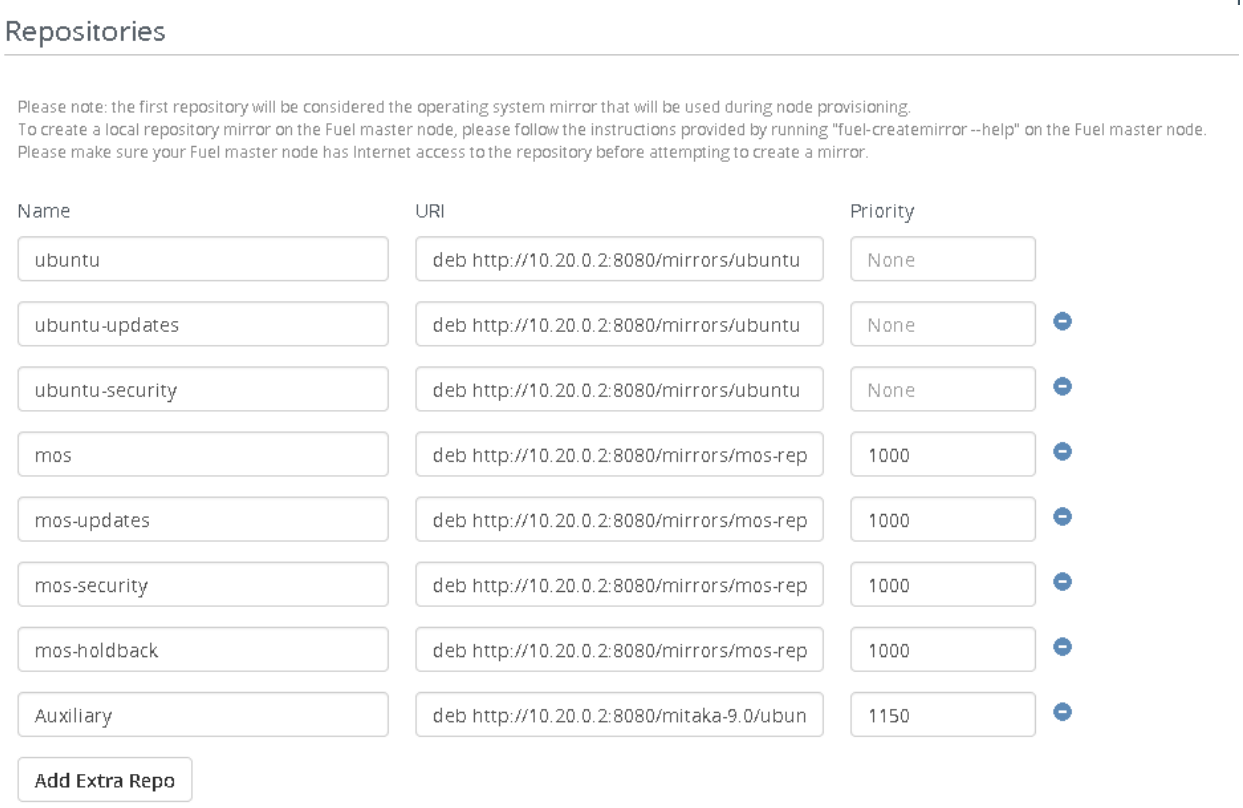
5.4.5 OpenStack Settings

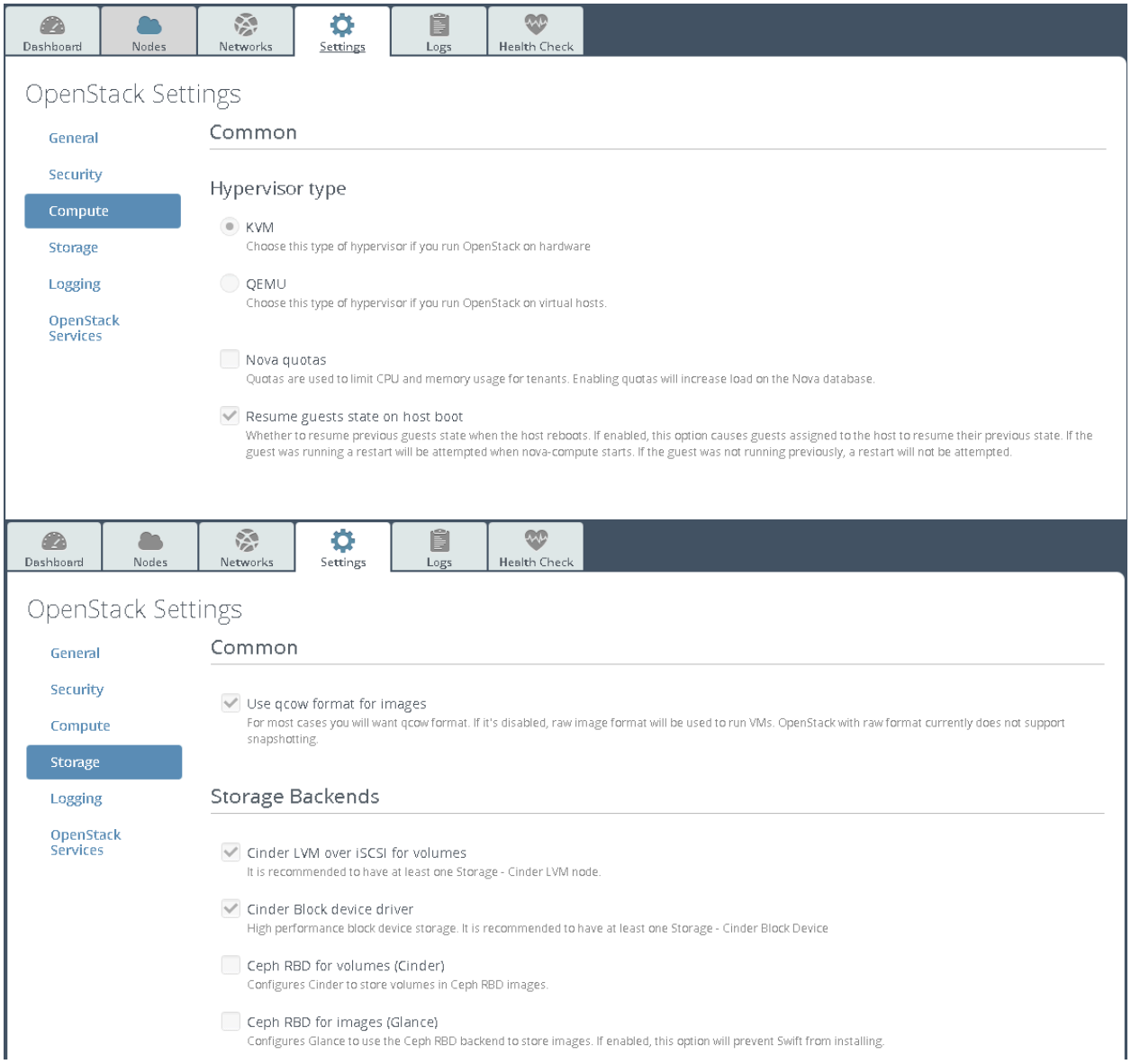
The "Settings" tab allows you to set or modify various values for the system. Many other values can be set by editing configuration files and running command-line tools on the nodes. You can find the details description in user guide:

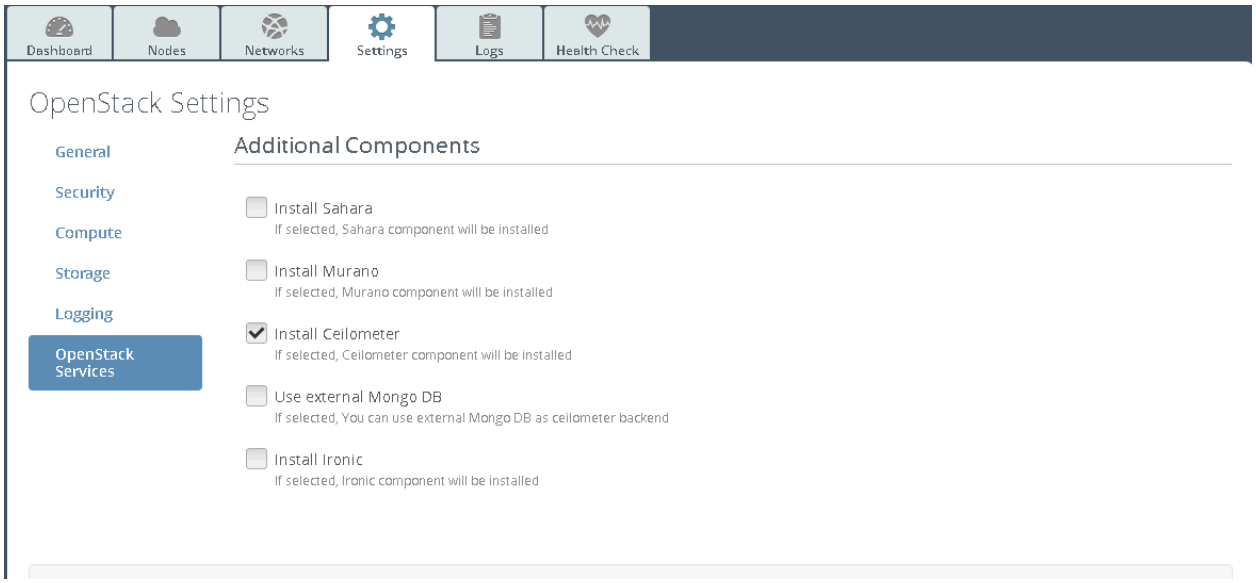
https://docs.mirantis.com/openstack/fuel/fuel-8.0/fuel-user-guide.html#configure-your-environment

In this demo, we change the default horizon setting, DNS and NTP setting as blew. The other settings are keep default.






Dashboard
Nodes
Networks
Settings
Logs
Health Check
OpenStack Settings
General
Security
Compute
Storage
Logging
OpenStack Services
Common
Hypervisor type
KVM
Choose this type of hypervisor if you run OpenStack on hardware
QEMU
Choose this type of hypervisor if you run OpenStack on virtual hosts.
Nova quotas
Quotas are used to limit CPU and memory usage for tenants. Enabling quotas will increase load on the Nova database.
Resume guests state on host boot
Whether to resume previous guests state when the host reboots. If enabled, this option causes guests assigned to the host to resume their previous state. If the guest was running a restart will be attempted when nova-compute starts. If the guest was not running previously, a restart will not be attempted.
Dashboard
Nodes
Networks
Settings
Logs
Health Check
OpenStack Settings
General
Security
Compute
Storage
Logging
OpenStack Services
Common
Use qcow format for images
For most cases you will want qcow format. If it's disabled, raw image format will be used to run VMs. OpenStack with raw format currently does not support snapshotting.
Storage Backends
Cinder LVM over iSCSI for volumes
It is recommended to have at least one Storage - Cinder LVM node.
Cinder Block device driver
High performance block device storage. It is recommended to have at least one Storage - Cinder Block Device
Ceph RBD for volumes (Cinder)
Configures Cinder to store volumes in Ceph RBD images.
Ceph RBD for images (Glance)
Configures Glance to use the Ceph RBD backend to store images. If enabled, this option will prevent Swift from installing.



5.4.6 The followings are bridge names and their IP addresses after the deployment for the reference:

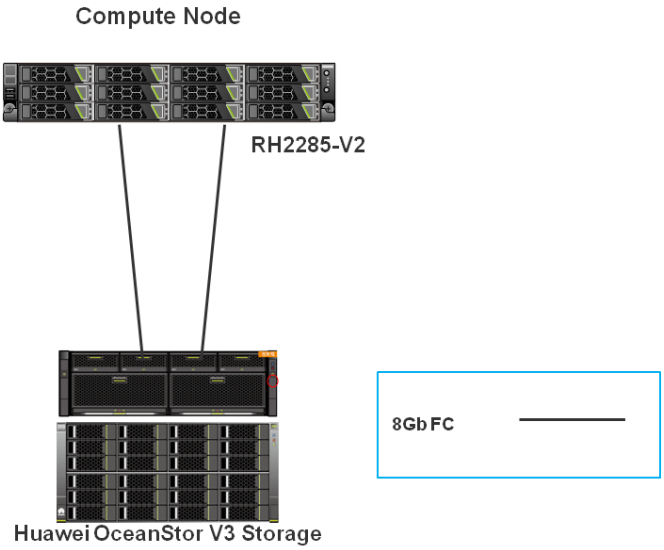
Network Interface	Controller (combine Mongo)Node 1	Compute (combine Cinder) Node
eth0 (br-fw-admin)	10.20.0.10	10.20.0.3
eth1(br-ex/public)	IP: 100.115.114.55	100.115.114.56
eth1 (br-mgmt)	192.169.0.3	192.169.0.4
eth1 (br-storage)	192.168.1.1	192.168.1.2

5.5 Huawei Volume Driver Installation Procedure

The following is the procedure that can be followed to install the Huawei Volume Driver. Make sure all cable connection between Compute node and Huawei OceanStor V3 storage are finished. In this demo, we use fibre to connect computer node and storage device directly.

Note:

In production environment, Huawei still recommends to using a dedicated SAN network for OpenStack environment.



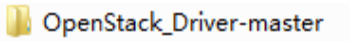
1) Hardware Preparation:
   - Huawei OceanStor V3 series storage (Huawei OceanStor 5800 V3 is used in this demo)

2) Software Preparation:
   - Huawei Volume Driver

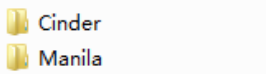
5.5.1 Get Huawei Volume Driver

You can get the Huawei Volume Driver for Huawei Storage from GitHub website in below link:
https://github.com/huaweistorage/OpenStack_Driver/tree/master

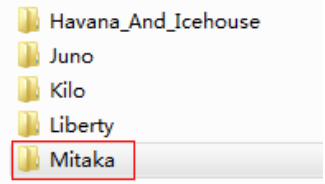
Unzip OpenStack_Driver-master.rar , it has the following 1 folder



Inside of the folder it has two folders



Inside of the Cinder, there are five versions of the OpenStack,



5.5.2 Cinder configuration on all cinder nodes

In this demo, Cinder component is installed in compute node. The following configuration is done on compute node.

1) Please create a folder names '*huawei*' in path */usr/lib/python2.7/dist-packages/cinder/volume/drivers/*. If it exists, backup this folder and remove all files that already in *huawei* folder.

2) using SCP to upload all files from above *Mitaka* folder to */usr/lib/python2.7/dist-packages/cinder/volume/drivers/huawei/*.
After you done, all files should show as below:

```
root@node-5:/usr/lib/python2.7/dist-packages/cinder/volume/drivers/huawei# ll
total 396
drwxr-xr-x  3 root root  4096 Dec  6 10:55 ./
drwxr-xr-x 23 root root  4096 Dec  6 10:56 ../
-rw-r--r--  1 root root  3434 Dec  6 10:55 constants.py
drwxr-xr-x  2 root root  4096 Dec  6 10:55 extend/
-rw-r--r--  1 root root 11163 Dec  6 10:55 fc_zone_helper.py
-rw-r--r--  1 root root 10577 Dec  6 10:55 huawei_conf.py
-rw-r--r--  1 root root 91780 Dec  6 10:55 huawei_driver.py
-rw-r--r--  1 root root 26672 Dec  6 10:55 huawei_t.py
-rw-r--r--  1 root root  3390 Dec  6 10:55 huawei_utils.py
-rw-r--r--  1 root root 14811 Dec  6 10:55 hypermetro.py
-rw-r--r--  1 root root    33 Dec  6 10:55 __init__.py
-rw-r--r--  1 root root 23397 Dec  6 10:55 replication.py
-rw-r--r--  1 root root 84774 Dec  6 10:55 rest_client.py
-rw-r--r--  1 root root  8867 Dec  6 10:55 smartx.py
-rw-r--r--  1 root root 94052 Dec  6 10:55 ssh_client.py
```

3) Configuring the volume driver for V3 (Fibre Channel)

This configuration file is an example for this certification scenario, for details please checks below description of Parameters in the Configuration File.

a. In */etc/cinder*, use *vi* command to create a Huawei-customized driver configuration file names "*/etc/cinder/cinder_huawei_conf.xml*".
b. Configure parameters in the driver configuration file. You must change the bold parameters in this configuration files according to your production environment, such as RestURL, account, LUNType , StoragePool and Cinder node HostIP etc.

```
<?xml version='1.0' encoding='UTF-8'?>
<config>
        <Storage>
            <Product>V3</Product>
            <Protocol>FC</Protocol>
            <RestURL>https://100.115.10.224:8088/deviceManager/rest/</RestURL>
            <UserName>admin</UserName>
            <UserPassword>Admin@storage1</UserPassword>
        </Storage>
        <LUN>
                <LUNType>Thick</LUNType>
                <WriteType>1</WriteType>
                <LUNcopyWaitInterval>5</LUNcopyWaitInterval>
                <Timeout>432000</Timeout>
                <StoragePool>StoragePoolName</StoragePool>
        </LUN>
        <Host HostIP="192.168.8.6"  HostType="Linux" />
</config>
```

Important notes:

- For more details about the parameters in the configuration file, please check below Parameters in the Configuration File. You can also find the details in section *4.2.7 Parameters in the*

Configuration File at below link:
https://github.com/huaweistorage/OpenStack_Driver/blob/master/ReleaseDoc/en/OpenStack%20Cinder%20Driver%20Configuration%20Guide.pdf

Table 1 Mandatory parameters

Parameter	Default Value	Description	Applicable To
Product	-	Type of a storage product. Such as **V3**.	All
Protocol	-	Type of a connection protocol. Possible value is **FC**.	All
RestURL	-	Access address of the REST interface, **https://x.x.x.x:8088/devicemanager/rest/**. **x.x.x.x** is the management IP address.	V3
UserName	-	Storage administrator username.	All
UserPassword	-	Storage Password of an administrator.	All
StoragePool	-	Name of a storage pool to be used.	All

Note:
For the configuration of other products like OceanStor Dorado V3 series, TV2 storage and OceanStor 18000 high-end storage or other protocol like iSCSI, please also reference:
https://github.com/huaweistorage/OpenStack_Driver/blob/master/ReleaseDoc/en/OpenStack%20Cinder%20Driver%20Configuration%20Guide.pdf

Table 2 Optional parameters

Parameter	Default Value	Description	Applicable To
LUNType	Thick	Type of the LUNs to be created. The value can be **Thick** or **Thin**.	V3
WriteType	1	Cache write type. Possible values are: **1** (write back), **2** (write through), and **3** (mandatory write back).	All
LUNcopyWaitInterval	5	After LUN copy is enabled, the plug-in frequently queries the copy progress. You can set a value to specify the query interval.	All
Timeout	432000	Timeout interval for waiting LUN copy of a storage device to complete. The unit is second.	V3
OSType	Linux	Operating system of the Nova compute node's host.	All
HostIP	-	IP address of the Nova compute node's host.	All

Notes:
You need to get Huawei's license first before creating thin lun.

c. Change *cinder_huawei_conf.xml* file owner to cinder user and cinder user group to allow cinder service to full access permission.

root@node-6:~# chown cinder:cinder /etc/cinder/cinder_huawei_conf.xml

d. Configure the cinder.conf file

Backup the*/etc/cinder/cinder.conf* file before any modify. In the *[default]* block of */etc/cinder/cinder.conf*, add the following contents. *enabled_backends* indicates the default backend will be used when creating volumes. *volume_driver* indicates the loaded driver file, and *cinder_huawei_conf_file* indicates the specified Huawei-customized configuration file.

enabled_backends=OceanStor_V3

[OceanStor_V3]
volume_driver = cinder.volume.drivers.huawei.huawei_driver.HuaweiFCDriver
cinder_huawei_conf_file = /etc/cinder/cinder_huawei_conf.xml
volume_backend_name = OceanStor_V3

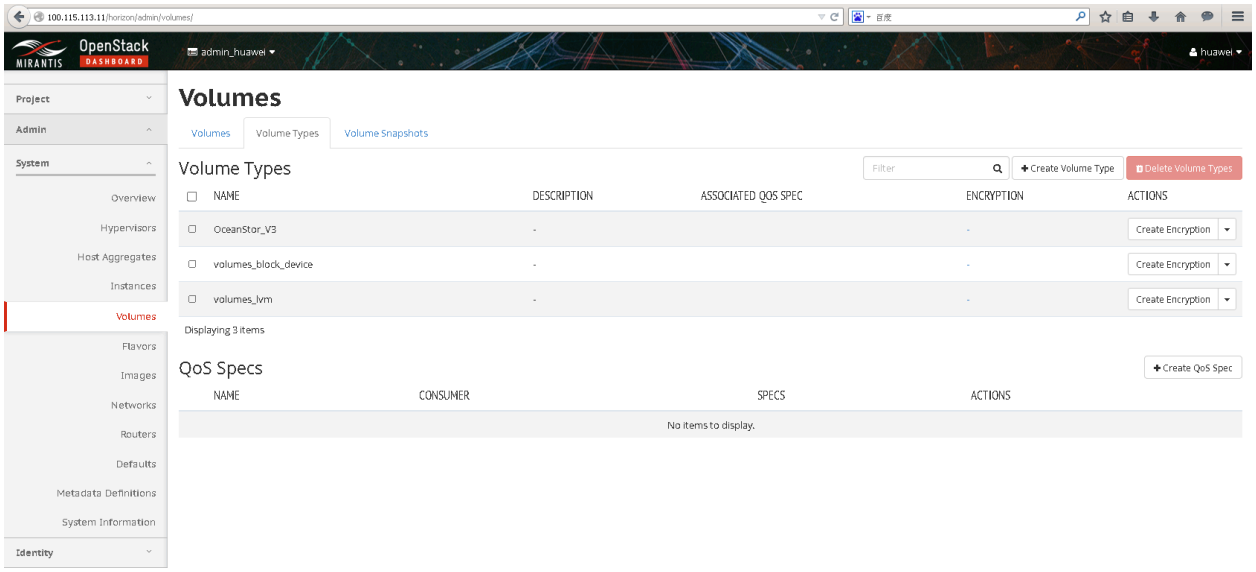
e. Run the service cinder-volume restart command to restart the Block Storage service.

root@node-6:~# service cinder-volume restart

f. Once the *cinder.conf* and Huawei customized driver configuration files have been created, cinder commands need to be issued in order to create and associate OpenStack volume types with the declared volume_backend_names:

root@node-6:~# cinder type-create OceanStor_V3
root@node-6:~# cinder type-key OceanStor_V3 set volume_backend_name=OceanStor_V3

g. Check cinder *OceanStor_V3* volume type on OpenStack Dashboard:



6. Testing

6.1 Test tools

Equipment	Purpose
Fuel health check scripts	Functional/acceptance testing
other	Test Huawei Volume Driver Function

6.2 Test cases

- Fuel Health Check test cases, see Fuel Health Check test result 6.3.1
- Other manually test cases:

Function	Description	Remark
Create Volume	Create OceanStor_V3 volume type empty volume on OpenStack Dashboard or CLI	pass
Show Volume	Query OceanStor_V3 volume type volume on OpenStack Dashboard or CLI	pass
Delete Volume	Delete OceanStor_V3 volume type volume on OpenStack Dashboard or CLI	pass
Create Snapshot	Create OceanStor_V3 volume type volume snapshot on OpenStack Dashboard or CLI	pass
Show Snapshot	Query OceanStor_V3 volume type volume snapshot on OpenStack Dashboard or CLI	pass
Delete Snapshot	Delete OceanStor_V3 volume type volume snapshot on OpenStack Dashboard or CLI	pass
Create Volume From Image	Create OceanStor_V3 volume type volume from image on OpenStack Dashboard or CLI	pass

Create Volume From Volume	Create OceanStor_V3 volume type volume from volume on OpenStack Dashboard or CLI	pass
Create Volume From Snapshot	Create OceanStor_V3 volume type volume from snapshot on OpenStack Dashboard or CLI	pass
Launch Instance From Volume	Launch Instance From Volume on OpenStack Dashboard or CLI	pass
Attach Volume to Instance	Attach Volume to Instance on OpenStack Dashboard or CLI	pass
Detach Volume to Instance	Detach Volume to Instance on OpenStack Dashboard or CLI	pass

6.2.1 Target Use case(s)

1) Large-scale cloud computing data centers
Huawei Volume Driver enables Huawei Storages integrate with MOS, providing converged functionality that maximizes performance and capacity utilization, Huawei storage products meet a wide range of needs: entry-level, mid-range, and high-end enterprise applications; Big Data; vertical industry applications; and cloud storage applications.

2) Critical enterprise IT infrastructure
MOS integrate with Huawei Storages, providing the best data services for enterprise, mission-critical businesses.

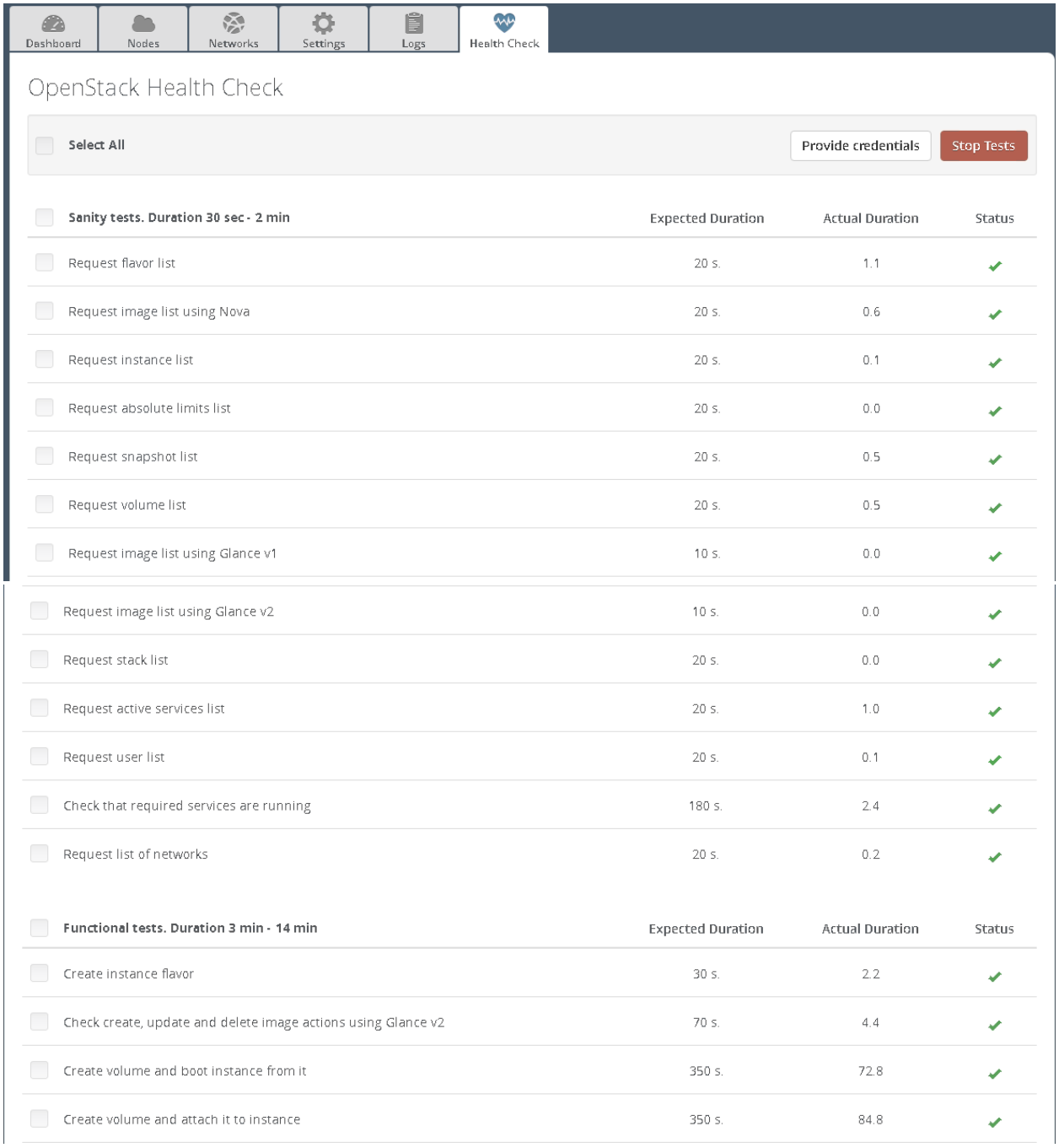
6.2.2 Deployment modes and configuration options

OS	Mode	HV	Network	Storage
			Neutron with VLAN	Cinder driver
Ubuntu	single	KVM	x	x

6.3 Test results

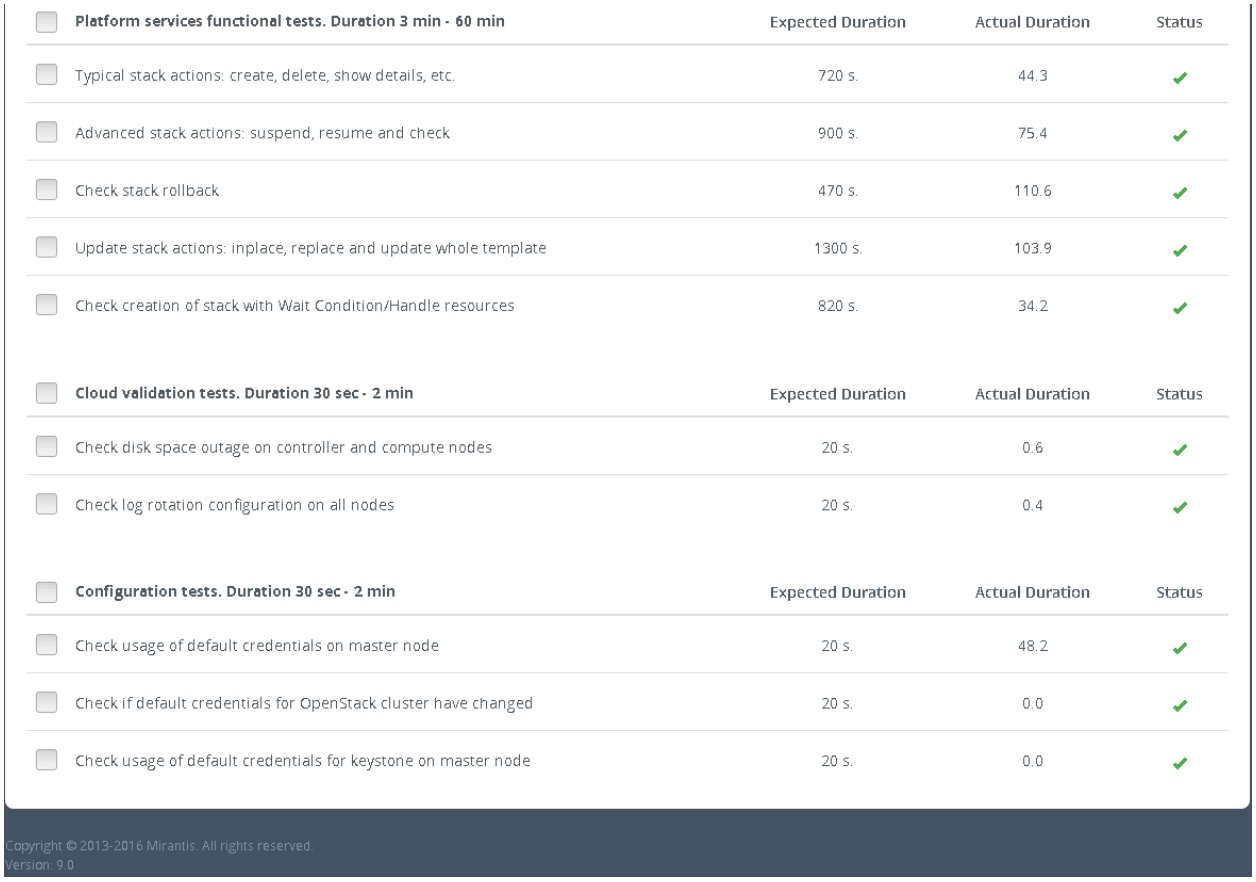
6.3.1 Fuel Health Check Result

The Fuel Health Check result is tested with Huawei Volume Driver and Huawei OceanStor V3 storage. Because this OpenStack environment is configured with just one controller, so the HA tests are skipped.


Dashboard
Nodes
Networks
Settings
Logs
Health Check
OpenStack Health Check
Select All
Provide credentials
Stop Tests
Sanity tests. Duration 30 sec - 2 min
Expected Duration
Actual Duration
Status
Request flavor list
20 s.
1.1
Request image list using Nova
20 s.
0.6
Request instance list
20 s.
0.1
Request absolute limits list
20 s.
0.0
Request snapshot list
20 s.
0.5
Request volume list
20 s.
0.5
Request image list using Glance v1
10 s.
0.0
Request image list using Glance v2
10 s.
0.0
Request stack list
20 s.
0.0
Request active services list
20 s.
1.0
Request user list
20 s.
0.1
Check that required services are running
180 s.
2.4
Request list of networks
20 s.
0.2
Functional tests. Duration 3 min - 14 min
Expected Duration
Actual Duration
Status
Create instance flavor
30 s.
2.2
Check create, update and delete image actions using Glance v2
70 s.
4.4
Create volume and boot instance from it
350 s.
72.8
Create volume and attach it to instance
350 s.
84.8

☐	Check network connectivity from instance via floating IP	300 s.	109.3	✔
☐	Create keypair	25 s.	1.2	✔
☐	Create security group	25 s.	1.1	✔
☐	Check network parameters	50 s.	0.2	✔
☐	Launch instance	200 s.	25.2	✔
☐	Launch instance with file injection	200 s.	29.6	✔
☐	Launch instance, create snapshot, launch instance from snapshot	300 s.	52.6	✔
☐	Create user and authenticate with it.	80 s.	5.5	✔

	HA tests. Duration 30 sec - 8 min	Expected Duration	Actual Duration	Status
☐	Check state of haproxy backends on controllers	10 s.	—	—
☐	Check data replication over mysql	10 s.	—	—
☐	Check if amount of tables in databases is the same on each node	10 s.	—	—
☐	Check galera environment state	10 s.	—	—
☐	Check pacemaker status	10 s.	—	—
☐	RabbitMQ availability	100 s.	—	—
☐	RabbitMQ replication	100 s.	—	—



Platform services functional tests. Duration 3 min - 60 min	Expected Duration	Actual Duration	Status
Typical stack actions: create, delete, show details, etc.	720 s.	44.3	✔
Advanced stack actions: suspend, resume and check	900 s.	75.4	✔
Check stack rollback	470 s.	110.6	✔
Update stack actions: inplace, replace and update whole template	1300 s.	103.9	✔
Check creation of stack with Wait Condition/Handle resources	820 s.	34.2	✔

Cloud validation tests. Duration 30 sec - 2 min	Expected Duration	Actual Duration	Status
Check disk space outage on controller and compute nodes	20 s.	0.6	✔
Check log rotation configuration on all nodes	20 s.	0.4	✔

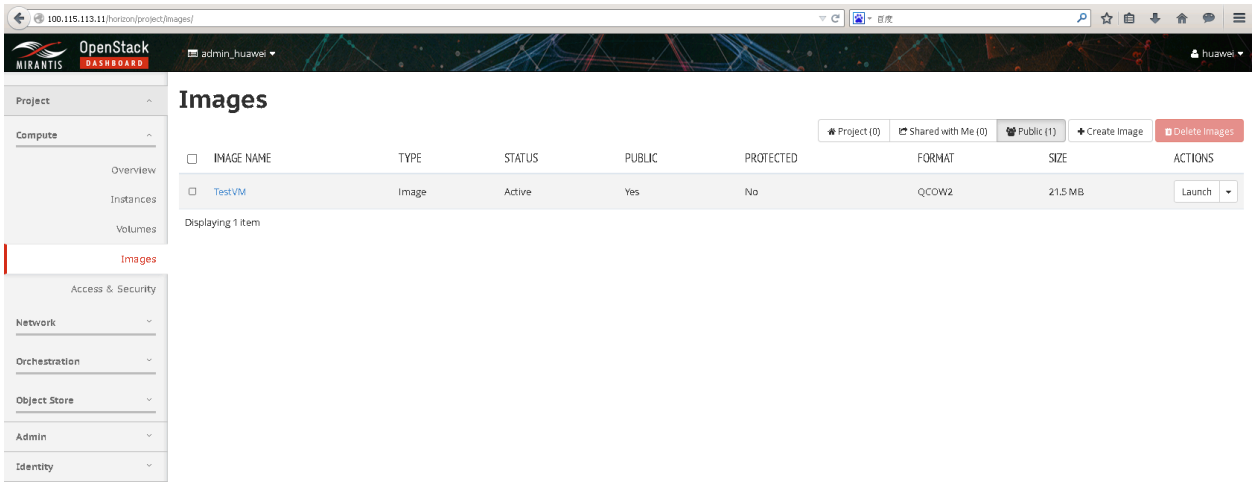
Configuration tests. Duration 30 sec - 2 min	Expected Duration	Actual Duration	Status
Check usage of default credentials on master node	20 s.	48.2	✔
Check if default credentials for OpenStack cluster have changed	20 s.	0.0	✔
Check usage of default credentials for keystone on master node	20 s.	0.0	✔

Copyright © 2013-2016 Mirantis. All rights reserved.
Version: 9.0

6.3.2 Other manually testing result screenshots.

We want to check that all the volume operations for Huawei OceanStor_V3 volume type are working normally. The volumes used in following test cases are all created in OceanStor_V3 volume type.

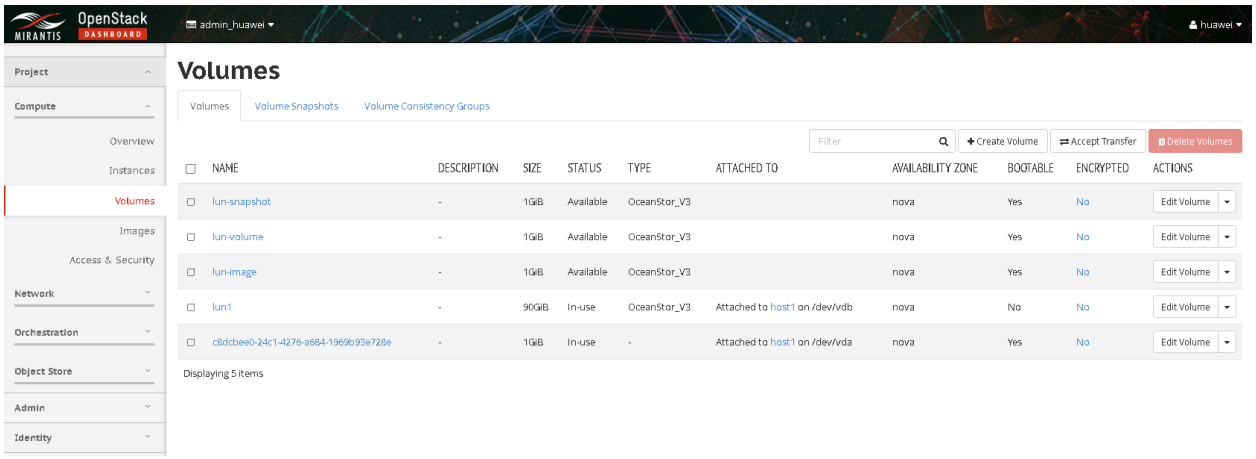
a. Created cirros image list on OpenStack Dashboard. We use the default cirros image which created by installing OpenStack cloud environment for creating volume from image test case.



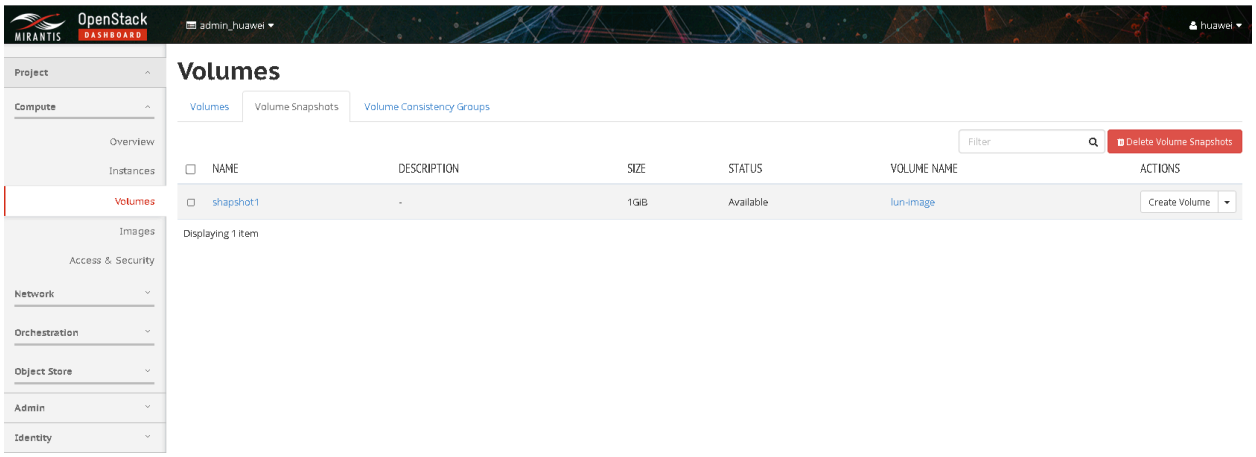
b. Volume operations: Created OceanStor_V3 type volume list on Openstack Dashboard.

- Create empty volume: create a empty volume from OceanStor_V3 volume type.
- Create volume from image: create a volume from cirros image.
- Create volume from snapshot: take a snapshot for a specific volume and create a volume from the snapshot.
- Create volume from volume: create a volume from an OceanStor_V3 volume.
- Delete: delete an OceanStor_V3 type volume.
- Modify: modify the size of a OceanStor_V3 type volume.

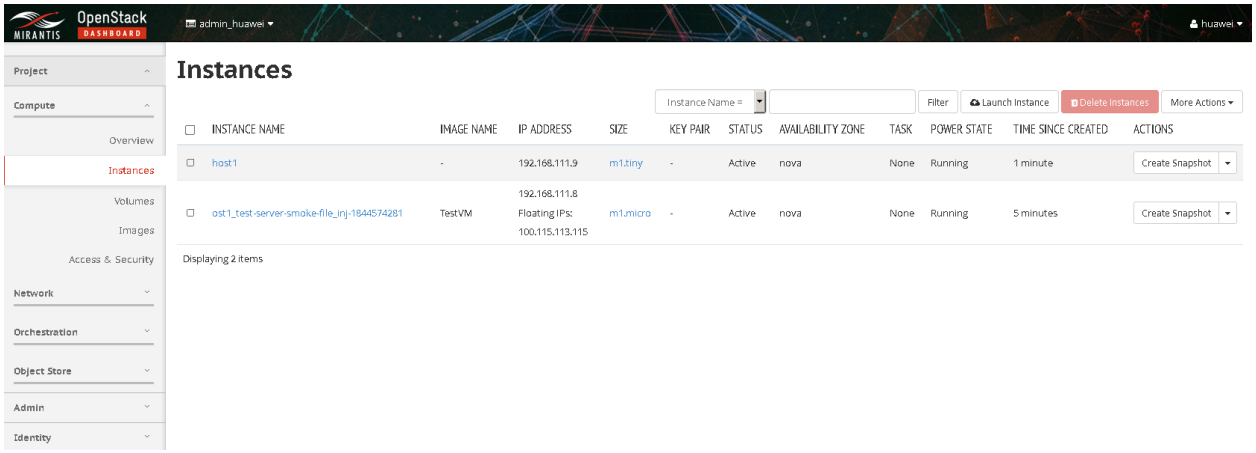
The volume list created in OceanStor_V3 volume type.



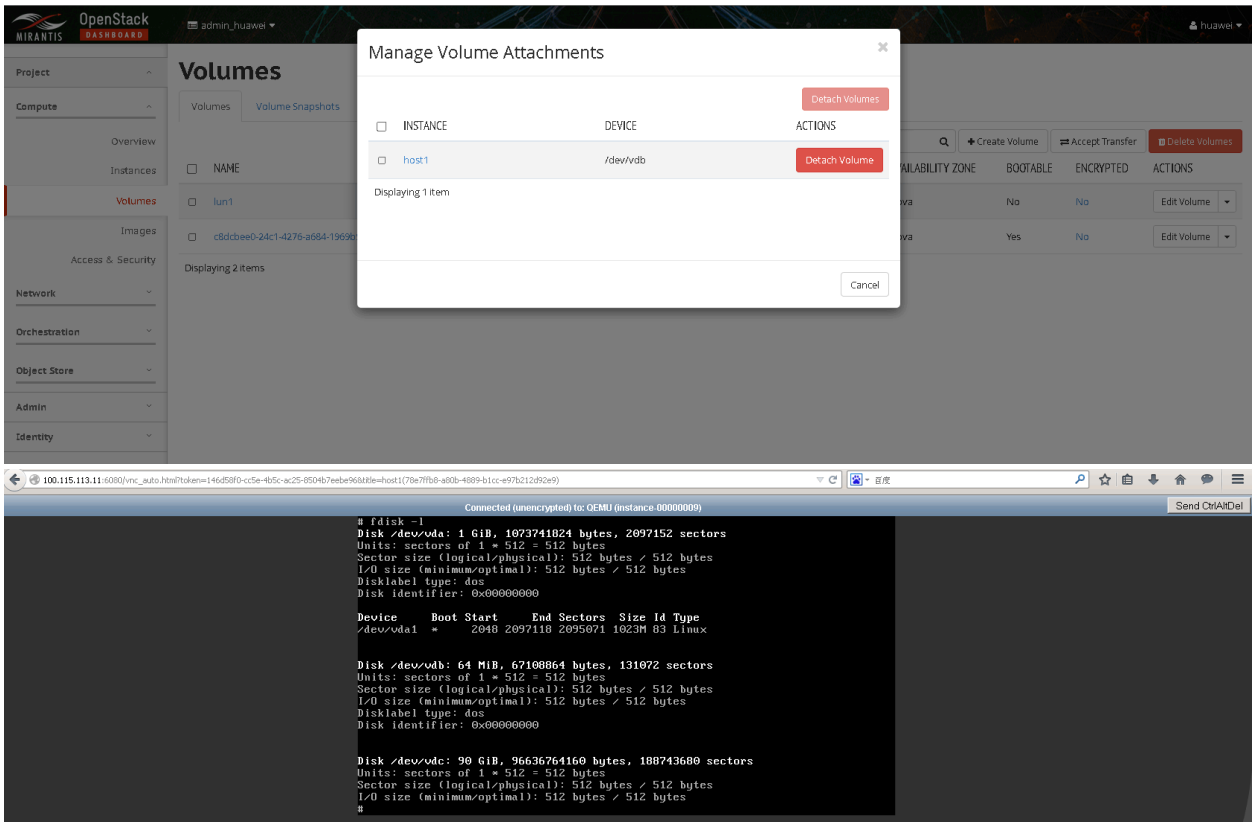
c. Created Volume Snapshots list on Openstack Dashboard.



d. Created Instances with OceanStor_V3 volumes on Openstack dashboard. (Also checked delete, modify, etc)



e. Attach/detach an OceanStor_V3 volume to instance on Openstack Dashboard



7. Appendix.

7.1 How to download Huawei Volume Driver software.
Visiting https://github.com/huaweistorage/OpenStack_Driver/tree/master

7.2 How to get the Huawei Volume Driver configuration guide
https://github.com/huaweistorage/OpenStack_Driver/blob/master/ReleaseDoc/en/OpenStack%20Cinder%20Driver%20Configuration%20Guide.pdf

7.3 How to get the Huawei Storage document
Visiting http://e.huawei.com/en/products/cloud-computing-dc/storage