INSTALLATION RUNBOOK FOR

Nimble Storage Cinder Drive

Product Name:	**[Nimble Cinder Driver]**
Product Version:	**[1.0]**
MOS Version:	**[7.0]**
OpenStack Version:	**[Kilo]**
Product Type:	**[Cinder Driver]**

Contents

Document History

Version	Revision Date	Description
1.0	03-28-2016	Initial Version

1. Introduction

Nimble Storage provides flash storage solutions, which are fully integrated with OpenStack using the Nimble Cinder driver. This document is a detailed Deployment Guide for the Nimble Cinder driver. It describes the reference architecture, limitations, and testing procedures, and provides steps to install and validate Mirantis OpenStack and the Nimble Cinder driver.

1.1 **Target Audience**

This document is for OpenStack administrators and developers who want to set up and configure Mirantis Fuel and the Nimble Storage Cinder driver. Refer to the Mirantis Fuel and OpenStack documentation if needed.

2. Product Overview

Nimble Storage solutions are built on its patented Cache Accelerated Sequential Layout (CASL™) architecture. CASL leverages the unique properties of flash and disk to deliver high performance and capacity – all within a dramatically small footprint. CASL and InfoSight™ form the foundation of the Adaptive Flash platform, which allows for the dynamic and intelligent deployment of storage resources to meet the growing demands of business-critical applications.

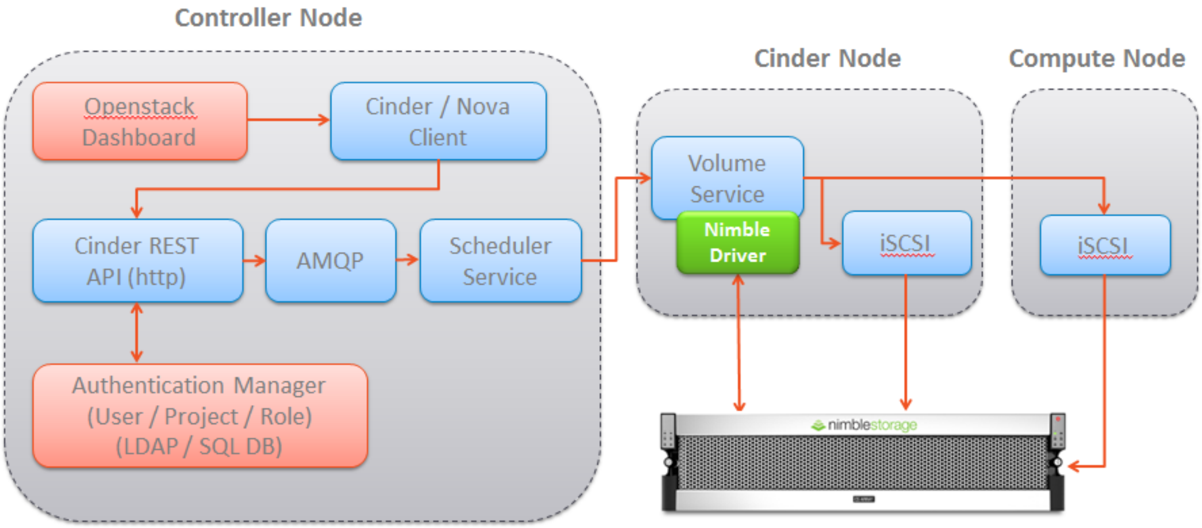
Nimble Storage enables enterprise IT organizations to implement a single architectural approach to dynamically cater to the needs of varying workloads. Within this single platform, customers can apply multiple service levels to address a wide range of enterprise applications. Assign an All Flash service level for demanding applications requiring deterministic low latency, while simultaneously applying an Auto Flash service level which more efficiently uses flash resources to cater to the majority of mainstream applications that require a balance of high performance and capacity. For applications with lower performance requirements, the No Flash service level optimizes for cost of capacity.

Nimble Storage is the only storage platform that optimizes across performance, capacity, data protection, and reliability within a dramatically smaller footprint.

3. Joint Reference Architecture

Nimble Storage fully integrates with the OpenStack platform through the OpenStack Block Storage Service (Cinder) driver, allowing a host to configure and manage Nimble Storage array features through the Cinder interface. The OpenStack Block Storage Service (Cinder) is composed of 4 basic elements:

- Cinder REST API - communicates with the authentication manager to validate a request (privilege and security)
- AMQP (for example, RabbitMQ) - queues a request into a messaging system of choice
- Scheduler - selects the proper backend to fulfill a volume management request
- Volume Service - communicates with the Nimble Storage array to complete a request



Nimble Storage integration supports the following operations for the OpenStack Block Storage Service (Cinder) driver. Refer to the OpenStack documentation for information on these operations:

- Volume Create/Delete
- Volume Attach/Detach
- Snapshot Create/Delete
- Create Volume from Snapshot
- Get Volume Stats
- Copy Image to Volume
- Copy Volume to Image
- Clone Volume
- Extend Volume

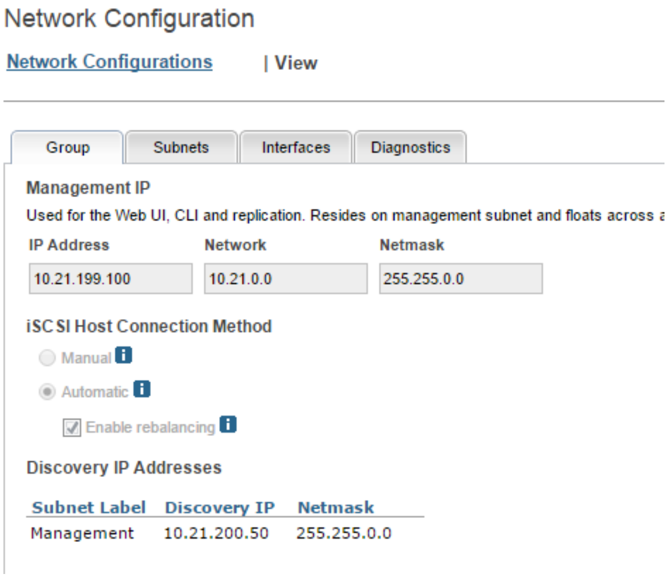
4. Installation and Configuration

4.1 Environment Preparation

Refer to the Mirantis OpenStack v7.0 User Guide for hardware requirements and instructions for using the Fuel Master node to deploy a Mirantis OpenStack environment. After the hardware is confirmed, use the following checklist to ensure Nimble Storage connectivity:

- Cinder node is able to access the Nimble Storage management subnet/network
- Cinder node is able to access the Nimble Storage data subnet/network
- Compute node is able to access the Nimble Storage data subnet/network

To acquire Nimble Storage subnet information, go to Administration -> Network Configuration -> Active Settings:



Management IP

Used for the Web UI, CLI and replication. Resides on management subnet and floats across a

IP Address	Network	Netmask
10.21.199.100	10.21.0.0	255.255.0.0

iSCSI Host Connection Method

○ Manual

● Automatic

☑ Enable rebalancing

Discovery IP Addresses

Subnet Label	Discovery IP	Netmask
Management	10.21.200.50	255.255.0.0

To check if Cinder node has access to management and data subnet, login to Cinder node and send pings to both Nimble Management IP address and Discovery IP address.

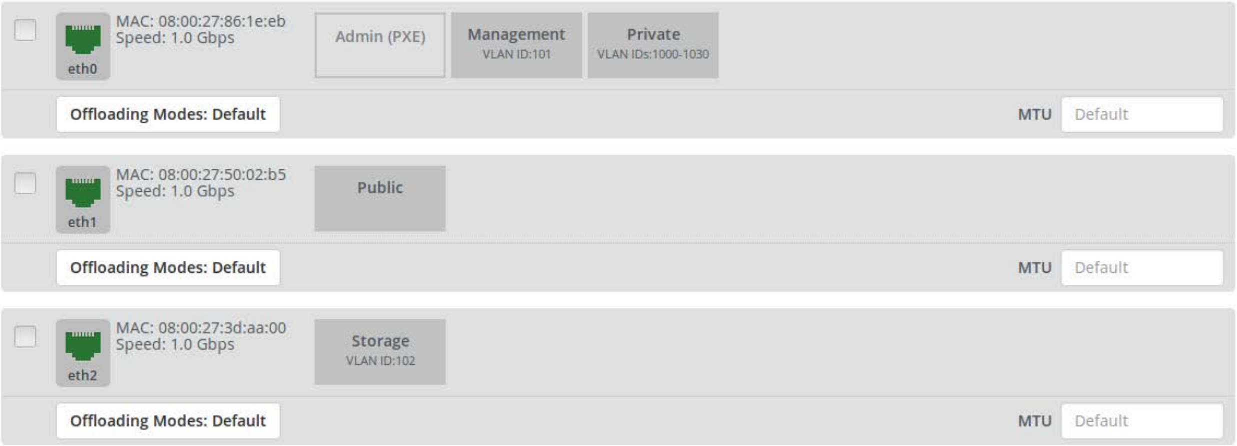
4.2 MOS Installation

From the Mirantis website, download Mirantis OpenStack v7.0. Follow the latest MOS installation guide on the Mirantis web site to complete the setup. Log into the Fuel UI and click on “New OpenStack Environment” to create a new OpenStack environment. Select compute, network, storage and additional services based on need. Click Finish and deploy.

Note that it’s required to deploy at least one storage node for Cinder services. It’s fine to deploy controller and storage on the same node. There is no need to enable Ceph storage. See section 4.3 for Cinder configuration.

4.2.1 Physical & Logical Network Topology

Each server requires two or more physical NICs. The network settings use OpenStack Neutron with VLAN segmentation. It is recommended to move storage interface to a separate physical interface as below.

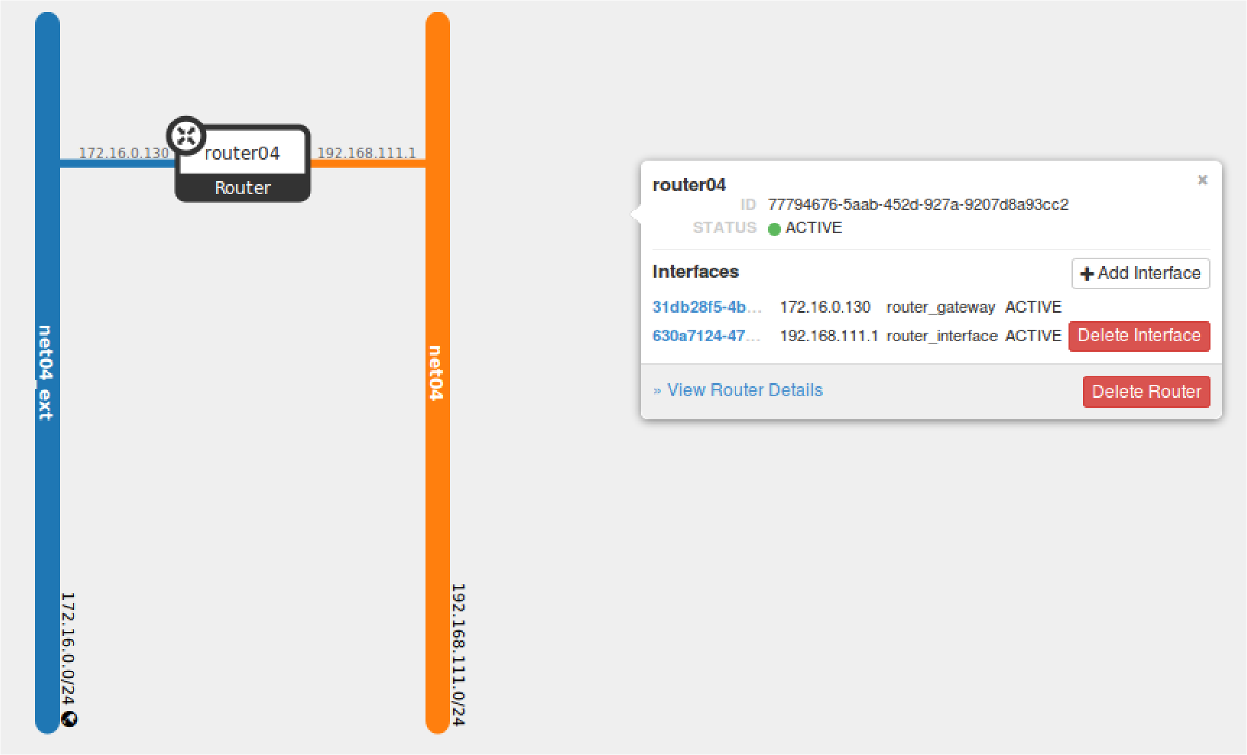


□ eth0 MAC: 08:00:27:86:1e:eb
Speed: 1.0 Gbps
Admin (PXE) | Management VLAN ID:101 | Private VLAN IDs:1000-1030
Offloading Modes: Default MTU Default

□ eth1 MAC: 08:00:27:50:02:b5
Speed: 1.0 Gbps
Public
Offloading Modes: Default MTU Default

□ eth2 MAC: 08:00:27:3d:aa:00
Speed: 1.0 Gbps
Storage VLAN ID:102
Offloading Modes: Default MTU Default

The public network 172.16.0.0/24 (shown in blue) and the private network 192.168.111.0/24 (shown in orange) below, interface with router04.

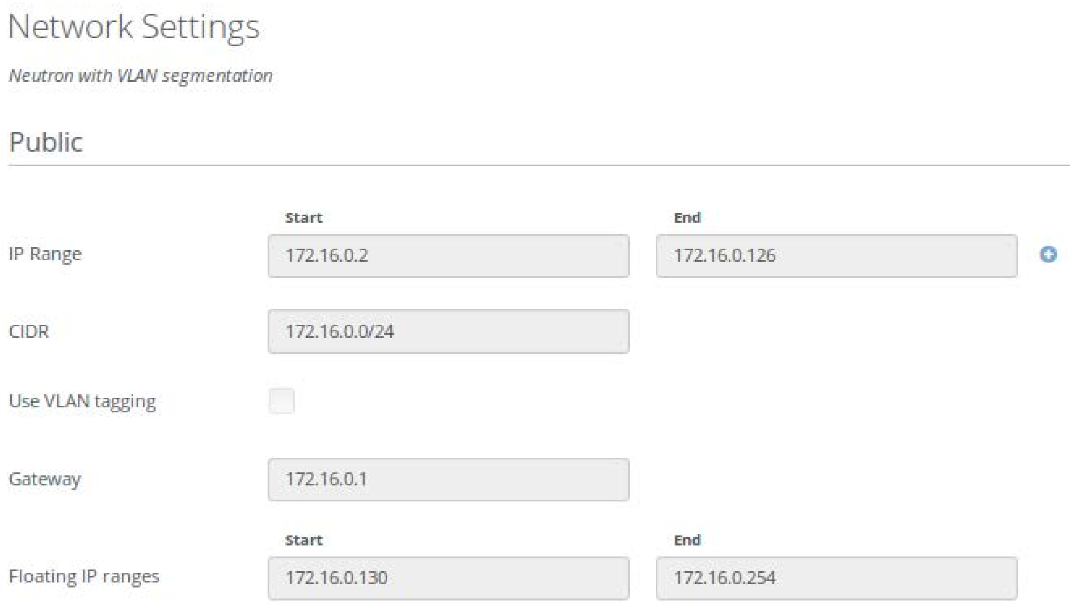


router04
ID 77794676-5aab-452d-927a-9207d8a93cc2
STATUS ● ACTIVE

Interfaces

31db28f5-4b...	172.16.0.130	router_gateway	ACTIVE
630a7124-47...	192.168.111.1	router_interface	ACTIVE

The network settings in Fuel are:

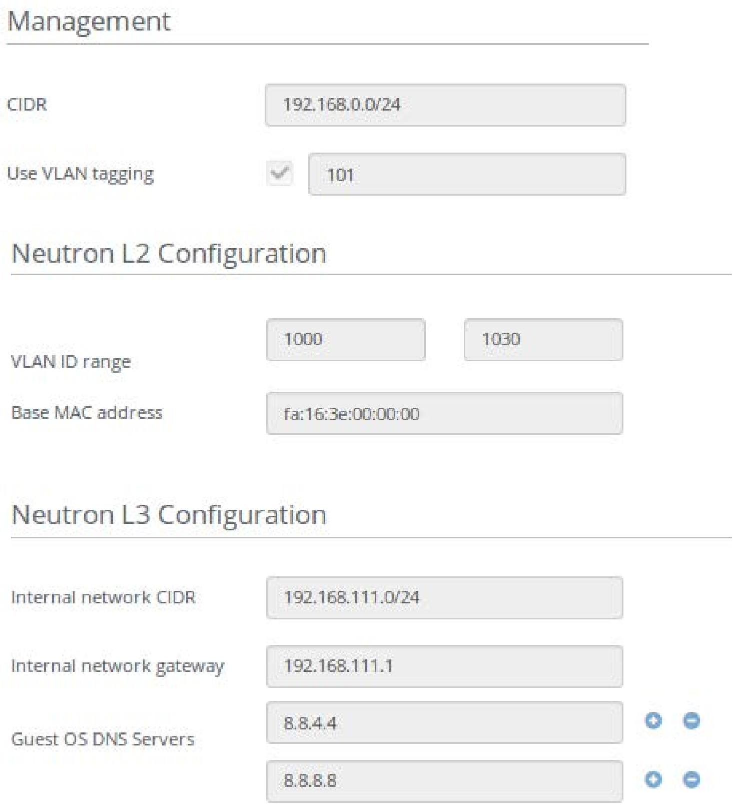


Network Settings

Neutron with VLAN segmentation

Public

	Start	End
IP Range	172.16.0.2	172.16.0.126
CIDR	172.16.0.0/24	
Use VLAN tagging	□	
Gateway	172.16.0.1	
Floating IP ranges	172.16.0.130	172.16.0.254



Management

CIDR: 192.168.0.0/24

Use VLAN tagging ☑ 101

Neutron L2 Configuration

VLAN ID range: 1000 1030

Base MAC address: fa:16:3e:00:00:00

Neutron L3 Configuration

Internal network CIDR: 192.168.111.0/24

Internal network gateway: 192.168.111.1

Guest OS DNS Servers: 8.8.4.4, 8.8.8.8

4.2.2 Health Check Results

After the deployment of the new OpenStack environment is done, go to the "Health Check" tab, click "Select All" and "Run Test". See section 6.2 for Fuel health check results with the Nimble Cinder driver installed.

4.3 Nimble Cinder Driver Installation Steps

On all Cinder storage nodes, update the file /etc/cinder/cinder.conf with the given configuration. For a single back-end configuration (a single Nimble Storage array/group), add the parameters within the [default] section as follows. The san_ip is the management IP address of Nimble Storage array/group.

```
[default]
san_ip = NIMBLE_MGMT_IP
san_login = NIMBLE_USER
san_password = NIMBLE_PASSWORD
volume_driver = cinder.volume.drivers.nimble.NimbleISCSIDriver
```

If there are more than one Nimble Storage array/group or several back-end storage solutions serving the same OpenStack compute configuration, a multi back-end Cinder configuration should be considered. For more information about Cinder multi back-end, see https://wiki.openstack.org/wiki/Cinder-multi-backend. For a multi back-end configuration, which supports multiple Nimble Storage arrays or a single Nimble Storage array with arrays from other vendors, use the following parameters.

```
[default]
enabled_backends = Nimble-Cinder

[Nimble-Cinder]
san_ip = NIMBLE_MGMT_IP
san_login = NIMBLE_USER
san_password = NIMBLE_PASSWORD
volume_driver = cinder.volume.drivers.nimble.NimbleISCSIDriver
volume_backend_name = NIMBLE_BACKEND_NAME
```

For a multi back-end configuration, the Nimble Storage volume-type is created and associated with a back-end name as follows. Single back-end configuration users do not need to create the volume-type.

```
$ cinder type-create NIMBLE_VOLUME_TYPE
$ cinder type-key NIMBLE_VOLUME_TYPE set volume_backend_name=NIMBLE_BACKEND_NAME
```

Restart all Cinder services including cinder-api, cinder-scheduler and cinder-volume services.

4.4 Limitations

It’s recommended to use the latest NimbleOS software on the Nimble controller to take advantages of the features. To update NimbleOS software, login to Nimble GUI and go to Administration -> Software. Or go to https://infosight.nimblestorage.com and select Resources -> Software Downloads -> NimbleOS for the current version.

5. Testing

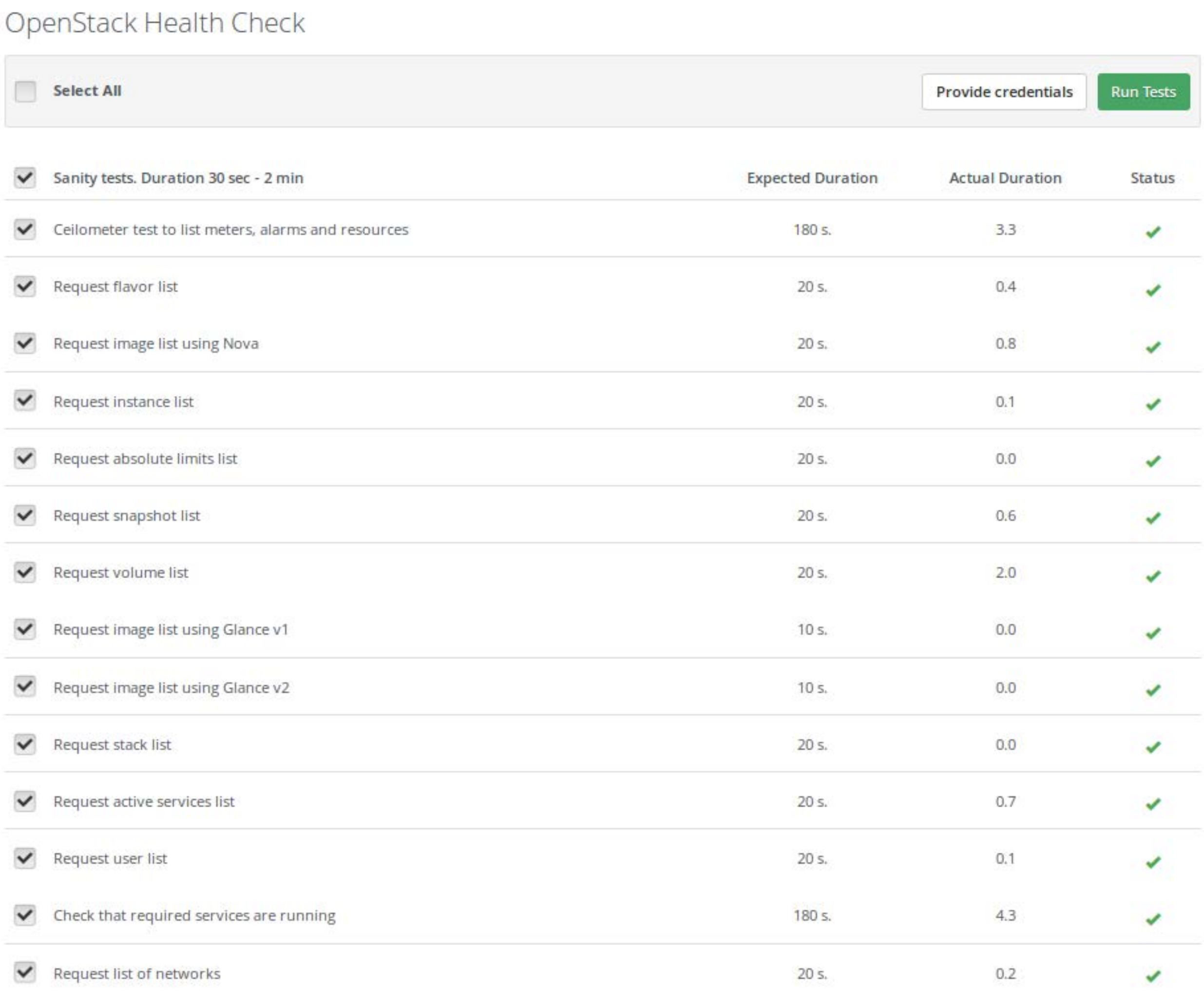
5.1 Test Cases

- Run and pass Fuel Health Check test cases
- Volume List/Create/Delete/Extend
- Volume Attach/Detach
- Snapshot List/Create/Delete
- Create Volume from Snapshot
- Get Volume Stats
- Copy Image to Volume
- Upload Volume to Image
- Clone Volume
- Create a boot instance from a new volume
- Fuel Health Check with Nimble Cinder driver

5.2 Test Results

Test case	Result
Create volume	PASS
Create volume with a specific volume type	PASS
Extend volume size	PASS
List volume	PASS
Delete volume	PASS
Attach a volume to an instance	PASS
Detach a volume from an instance	PASS
Create snapshot	PASS
List snapshot	PASS
Delete snapshot	PASS
Get volume stats	PASS
Copy image to volume	PASS
Clone volume from a snapshot	PASS
Create a boot instance from a new volume	PASS

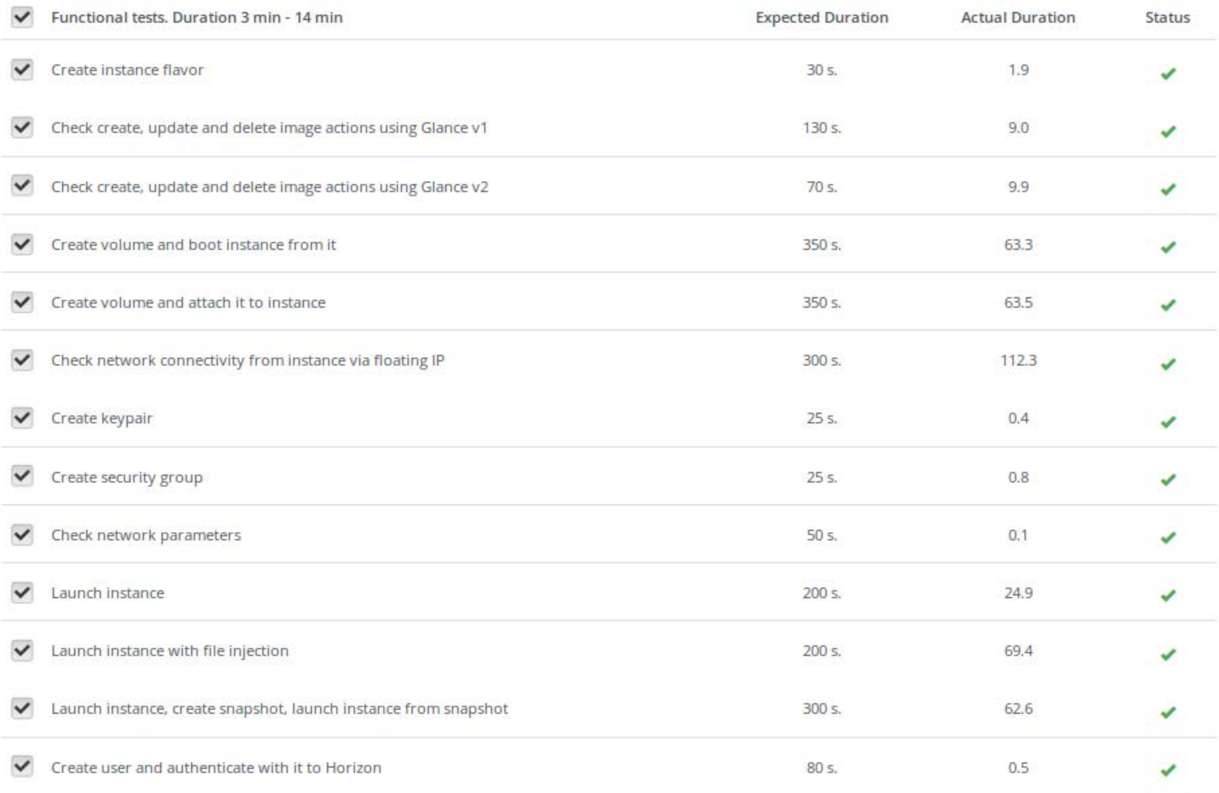
Fuel Health Check result with Nimble Cinder driver:



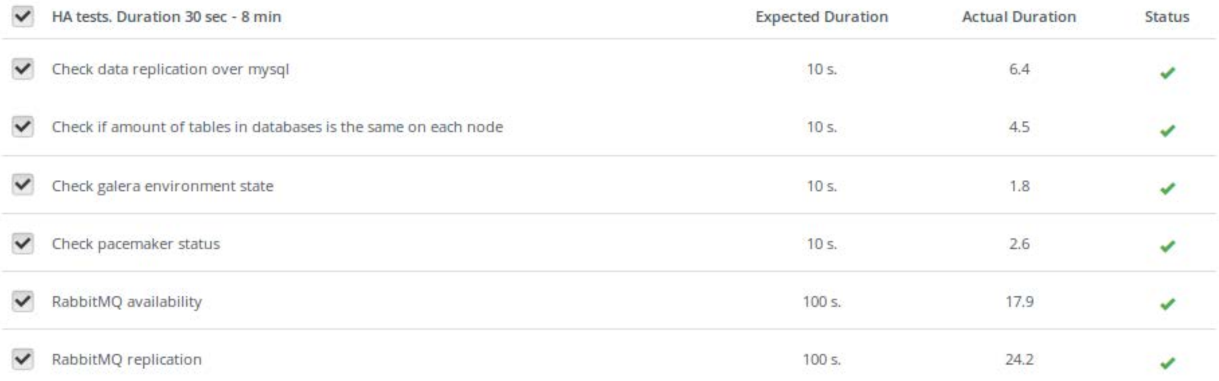
OpenStack Health Check

□ Select All

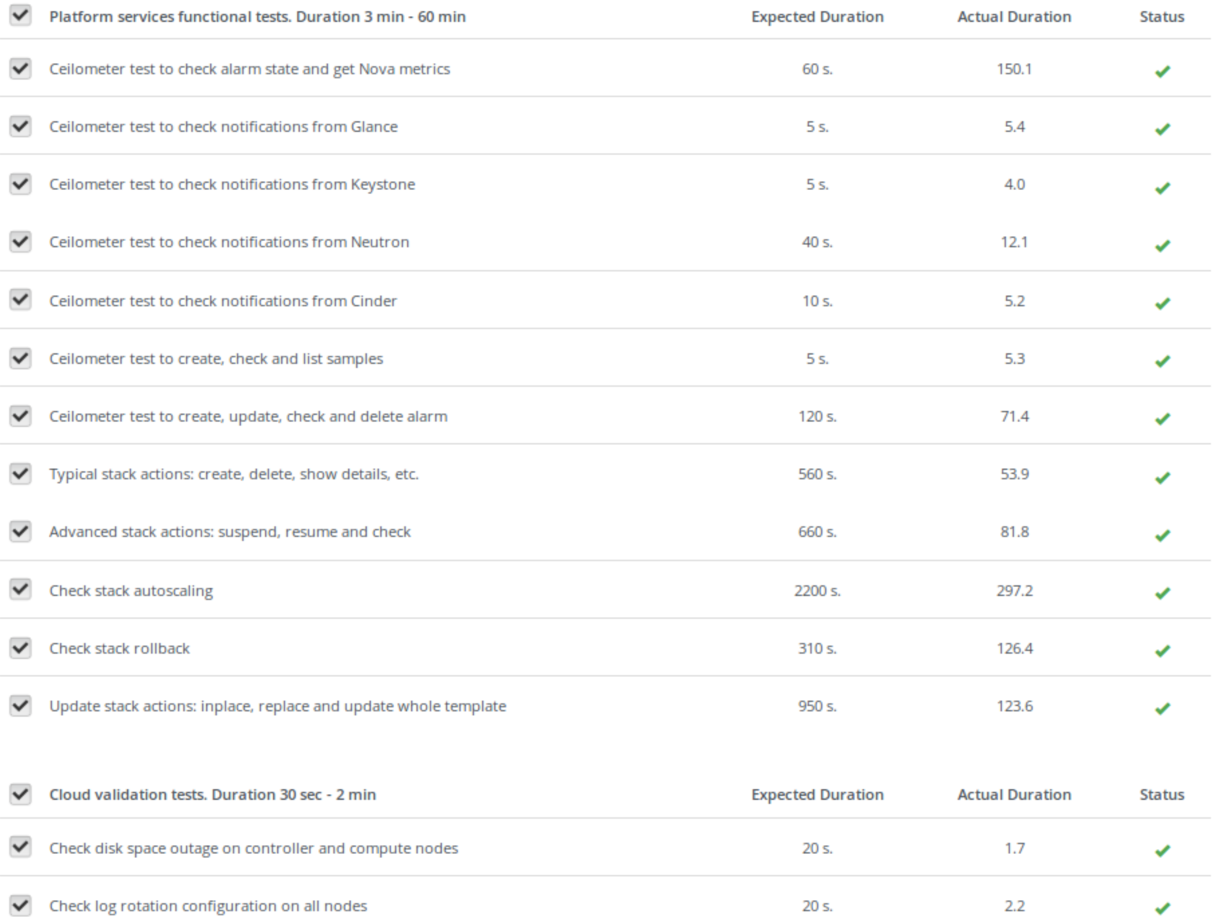
☑ Sanity tests. Duration 30 sec - 2 min	Expected Duration	Actual Duration	Status
☑ Ceilometer test to list meters, alarms and resources	180 s.	3.3	✓
☑ Request flavor list	20 s.	0.4	✓
☑ Request image list using Nova	20 s.	0.8	✓
☑ Request instance list	20 s.	0.1	✓
☑ Request absolute limits list	20 s.	0.0	✓
☑ Request snapshot list	20 s.	0.6	✓
☑ Request volume list	20 s.	2.0	✓
☑ 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.	0.7	✓
☑ Request user list	20 s.	0.1	✓
☑ Check that required services are running	180 s.	4.3	✓
☑ 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.	1.9	✓
☑ Check create, update and delete image actions using Glance v1	130 s.	9.0	✓
☑ Check create, update and delete image actions using Glance v2	70 s.	9.9	✓
☑ Create volume and boot instance from it	350 s.	63.3	✓
☑ Create volume and attach it to instance	350 s.	63.5	✓
☑ Check network connectivity from instance via floating IP	300 s.	112.3	✓
☑ Create keypair	25 s.	0.4	✓
☑ Create security group	25 s.	0.8	✓
☑ Check network parameters	50 s.	0.1	✓
☑ Launch instance	200 s.	24.9	✓
☑ Launch instance with file injection	200 s.	69.4	✓
☑ Launch instance, create snapshot, launch instance from snapshot	300 s.	62.6	✓
☑ Create user and authenticate with it to Horizon	80 s.	0.5	✓



☑ HA tests. Duration 30 sec - 8 min	Expected Duration	Actual Duration	Status
☑ Check data replication over mysql	10 s.	6.4	✓
☑ Check if amount of tables in databases is the same on each node	10 s.	4.5	✓
☑ Check galera environment state	10 s.	1.8	✓
☑ Check pacemaker status	10 s.	2.6	✓
☑ RabbitMQ availability	100 s.	17.9	✓
☑ RabbitMQ replication	100 s.	24.2	✓



☑	Platform services functional tests. Duration 3 min - 60 min	Expected Duration	Actual Duration	Status
☑	Ceilometer test to check alarm state and get Nova metrics	60 s.	150.1	✓
☑	Ceilometer test to check notifications from Glance	5 s.	5.4	✓
☑	Ceilometer test to check notifications from Keystone	5 s.	4.0	✓
☑	Ceilometer test to check notifications from Neutron	40 s.	12.1	✓
☑	Ceilometer test to check notifications from Cinder	10 s.	5.2	✓
☑	Ceilometer test to create, check and list samples	5 s.	5.3	✓
☑	Ceilometer test to create, update, check and delete alarm	120 s.	71.4	✓
☑	Typical stack actions: create, delete, show details, etc.	560 s.	53.9	✓
☑	Advanced stack actions: suspend, resume and check	660 s.	81.8	✓
☑	Check stack autoscaling	2200 s.	297.2	✓
☑	Check stack rollback	310 s.	126.4	✓
☑	Update stack actions: inplace, replace and update whole template	950 s.	123.6	✓

☑	Cloud validation tests. Duration 30 sec - 2 min	Expected Duration	Actual Duration	Status
☑	Check disk space outage on controller and compute nodes	20 s.	1.7	✓
☑	Check log rotation configuration on all nodes	20 s.	2.2	✓

5.3 Troubleshooting

OpenStack Cinder is used to provide access to block storage resources. When there is a block storage or Cinder issue in OpenStack, it’s suggested to review the following logs to further isolate the issue:

- cinder-api log (/var/log/cinder/api.log): It is helpful to track down endpoint or connectivity issues. For example, if a create volume operation fails, cinder-api.log will be helpful to identify if the actual API request made it to the backend storage or not.
- cinder-volume log (var/log/cinder/volume.log): Cinder driver will log operation information in the log. When API call made it to the backend storage device, it will record storage controller operations and results in the log.
- cinder-scheduler.log (/var/log/cinder/scheduler.log), cinder-manage.log and backup.log are usually not directly relevant for troubleshooting purposes.

Troubleshooting and other help is available from the Nimble Storage Support site at https://infosight.nimblestorage.com or you can email Nimble Storage Support at support@nimblestorage.com. All documentation and knowledge base articles are available on InfoSight. You can self-register on InfoSight using the Enroll Now link on the main page.