INSTALLATION RUNBOOK FOR Triliodata + TrilioVault

Application Type: **[**Backup and disaster recovery**]**

Application Version: **[2.1]**

MOS Version: **[7.0]**

OpenStack version: **[Kilo]**

Content

Document History

Version	Revision Date	Description
1.0	08-04-2016	Initial Version

1 Introduction

This document describes about TrilioVault Deployment.
TrilioVault, by Trilio Data, is a native OpenStack service that provides policy-based comprehensive backup and recovery within OpenStack environments. The solution captures point-in-time workloads (Application, OS, Compute, Network, Configurations, Data and Metadata of an environment) as full or incremental snapshots. These snapshots can be held in a variety of storage environments (NFS, 3rd party Arrays, etc.). With TrilioVault and its single click recovery, organizations improve Recovery Time Objectives (RTO) and Recovery Point Objectives. IT departments are enabled to fully deploy OpenStack solutions and provide business assurance through enhanced data retention, protection and integrity.

1.1 Target Audience

This application is targeted for all OpenStack users.

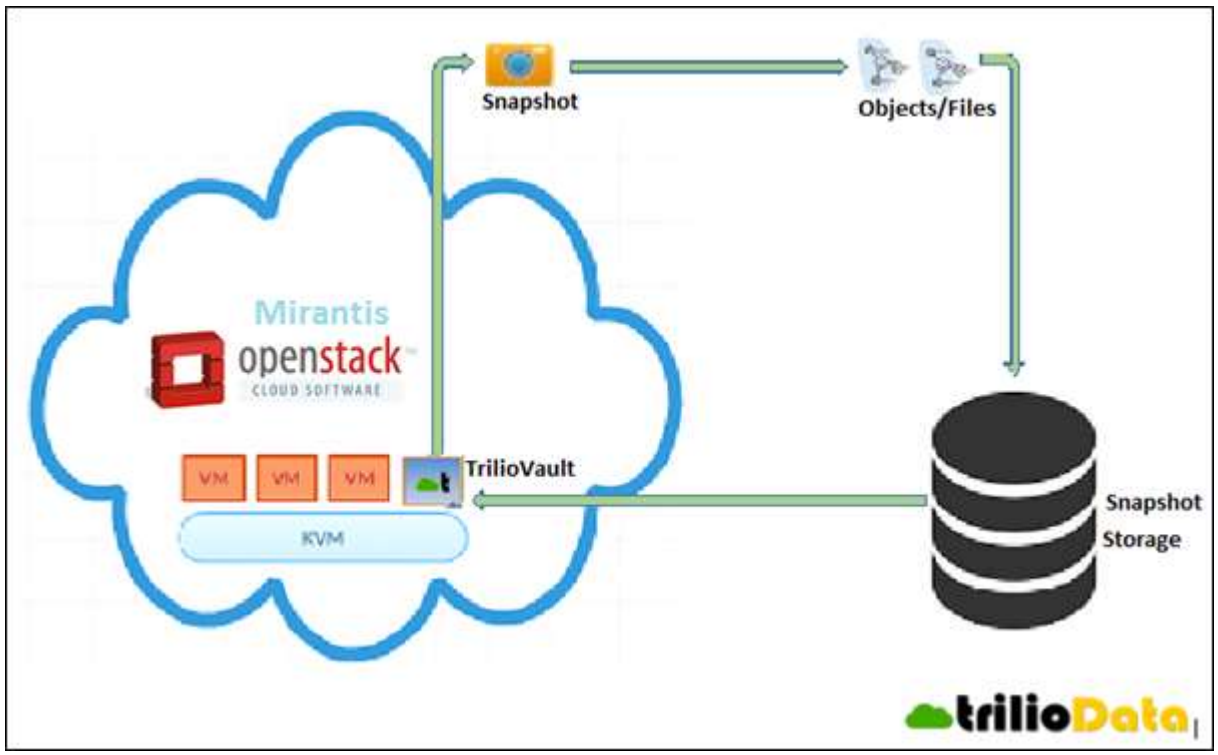
2 Application overview

With the use of TrilioVault's VAST (Virtual Snapshot Technology), Enterprise IT and Cloud Service Provider (Mirantis) can now use backup and disaster recovery as a solution to prevent data loss or data corruption through point-in-time snapshots and seamless one-click recovery.
TrilioVault takes point-in-time backup of the entire workload cluster consisting of compute resources, network configuration and storage data as a whole. TrilioVault also takes incremental backup thereby reducing the time to capture the snapshot of the changes that were made in the system as full snapshot is not required. The benefits of using VAST for backup and restore could be summarized as below

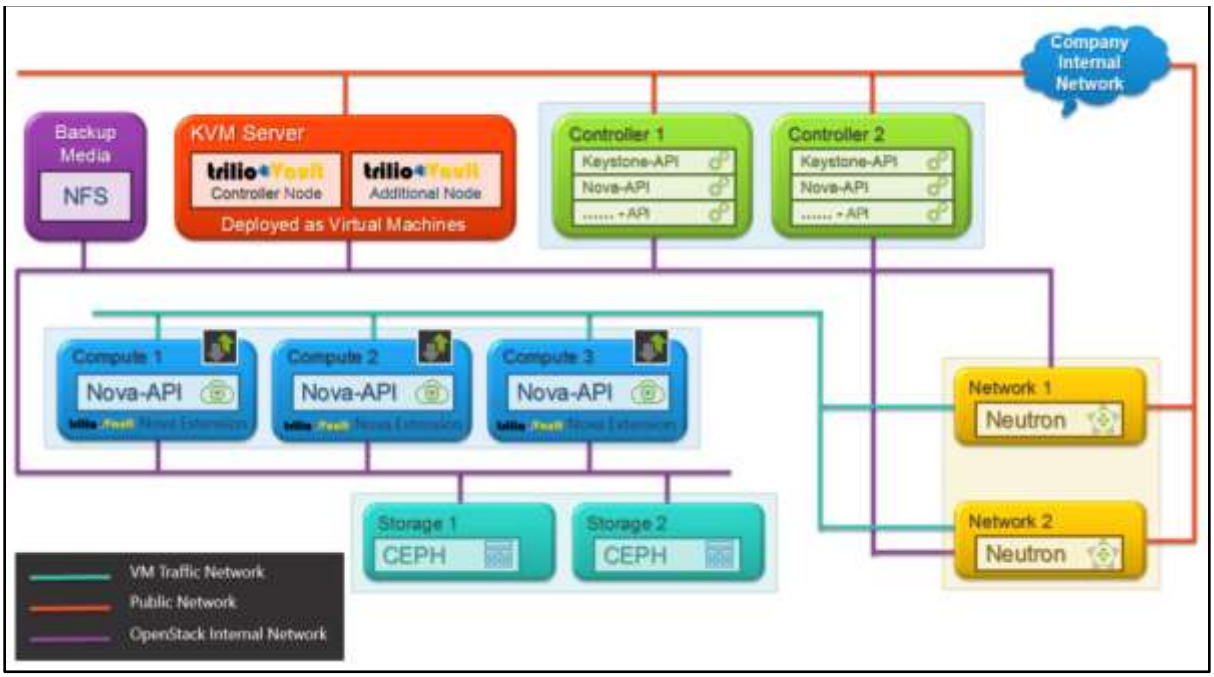
- Efficient capture and storage of snapshots.
- Faster and reliable recovery.
- Easy migration of the application between clouds.
- Through policy and automation lowered Total Cost of Ownership.

Prerequisite: NFS server needs to be configured where backup/snapshot will be stored.

3 Joint Reference Architecture



4 Physical & Logical Network Topology



5 Installation & Configuration

5.1 Environment preparation

TrilioVault can be deployed on standalone KVM box or OpenStack environment itself. OpenStack environment means you can import the appliance QCOW2 and then launch an instance from that image.

5.2 MOS installation

(https://docs.mirantis.com/openstack/fuel/fuel-7.0/user-guide.html)

Step 1) Deploy Fuel master.

To deploy fuel master we have downloaded Mirantis Fuel 7 iso image from following link : https://www.mirantis.com/software/mirantis-openstack/releases/

Once we have fuel master iso we have uploaded this image on data-store and created a vm to boot using this iso. Once we have the configure page option we can proceed with step2.

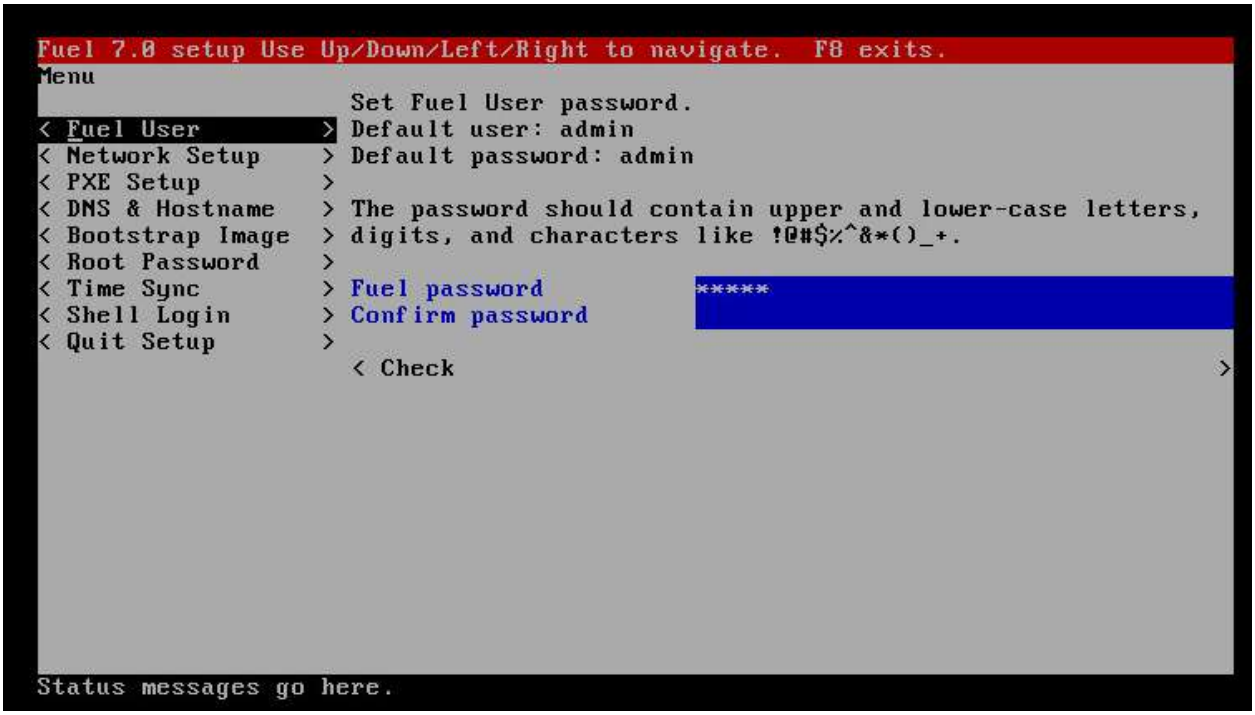
Fuel version 7 installation guide: https://docs.mirantis.com/openstack/fuel/fuel-7.0/user-guide.html

Step 2) After Fuel Server boots up we get a console for configuration.
The console-based Fuel Setup allows you to customize the Admin (PXE) logical network if you want to use a different network interface.

This tool provides a simple way to configure Fuel for your particular networking environment, while helping to detect errors early so you do not need to troubleshoot individual configuration files.

Within Fuel Setup, you can configure the following parameters:

DHCP/Static configuration for each network interface
Select interface for Fuel Admin network
Define DHCP pool (bootstrap) and static range (installed nodes)
Set NTP servers for Time settings
Root password
Fuel password
DNS options
Launch shell for optional pre-deployment tasks



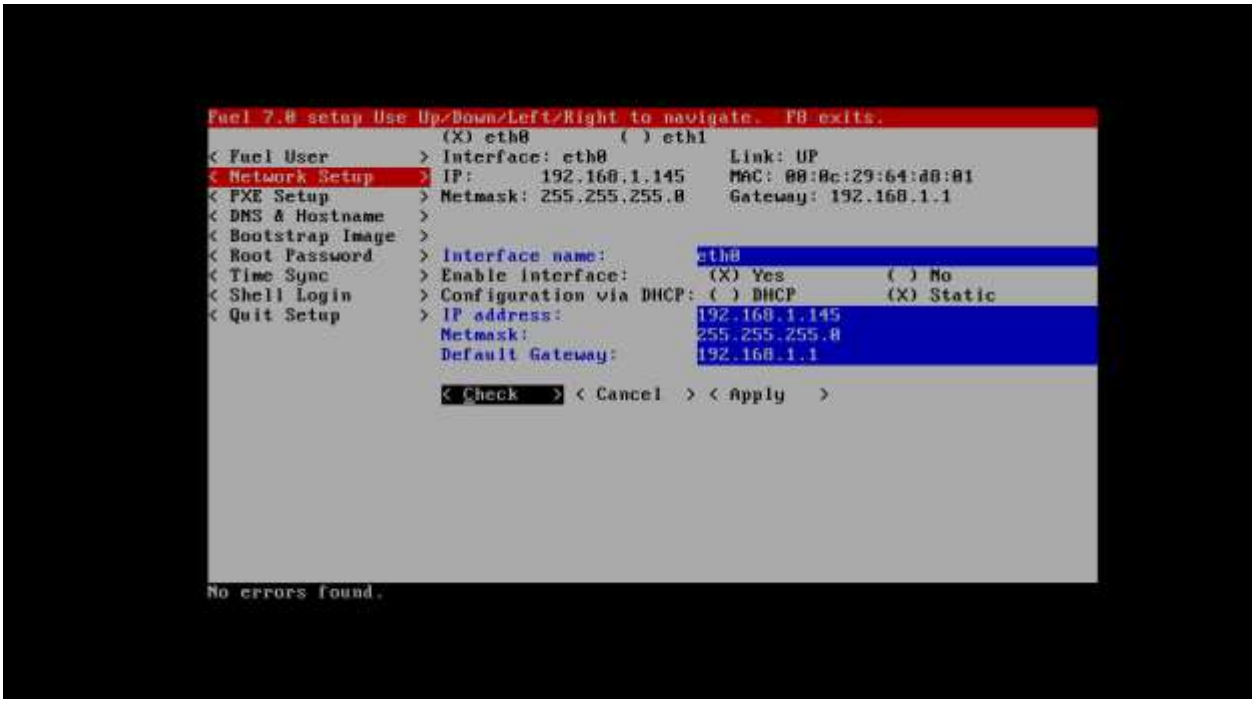
Step 3) Configuring Network settings

Network settings has 2 parts - editable Network settings and non-editable Network Interface current status. NIC current status area shows the current network interface status, including name, Link Status, current IP address, MAC address, Netmask and Gateway.

Network Settings from the editable Network Setup part become effective only after they are applied with the Apply button.

Network Setup includes the following configurable sections:

Network Interface Selector - Shows all available network interfaces, physical and virtual. Select the interface you want to configure with arrow keys and click Space or Enter to show its configuration.
Interface name - Here you may rename the selected network interface.
Enable interface - Here you may turn the selected network interface ON or OFF.
Configuration via DHCP - You may set interface to get settings from the existing external DHCP server. Do not set DHCP=Yes for the network interface you are going to use for Admin (PXE) network!
IP Address - allows to set static IP address for selected NIC.
Netmask - allows to set network mask for selected NIC.
Default gateway - allows to set the gateway for selected NIC.
Button Check - Validates the unsaved settings on the Network Setup section without applying.
Button Apply - Validates the unsaved settings on the Network Setup section and makes the new settings effective.



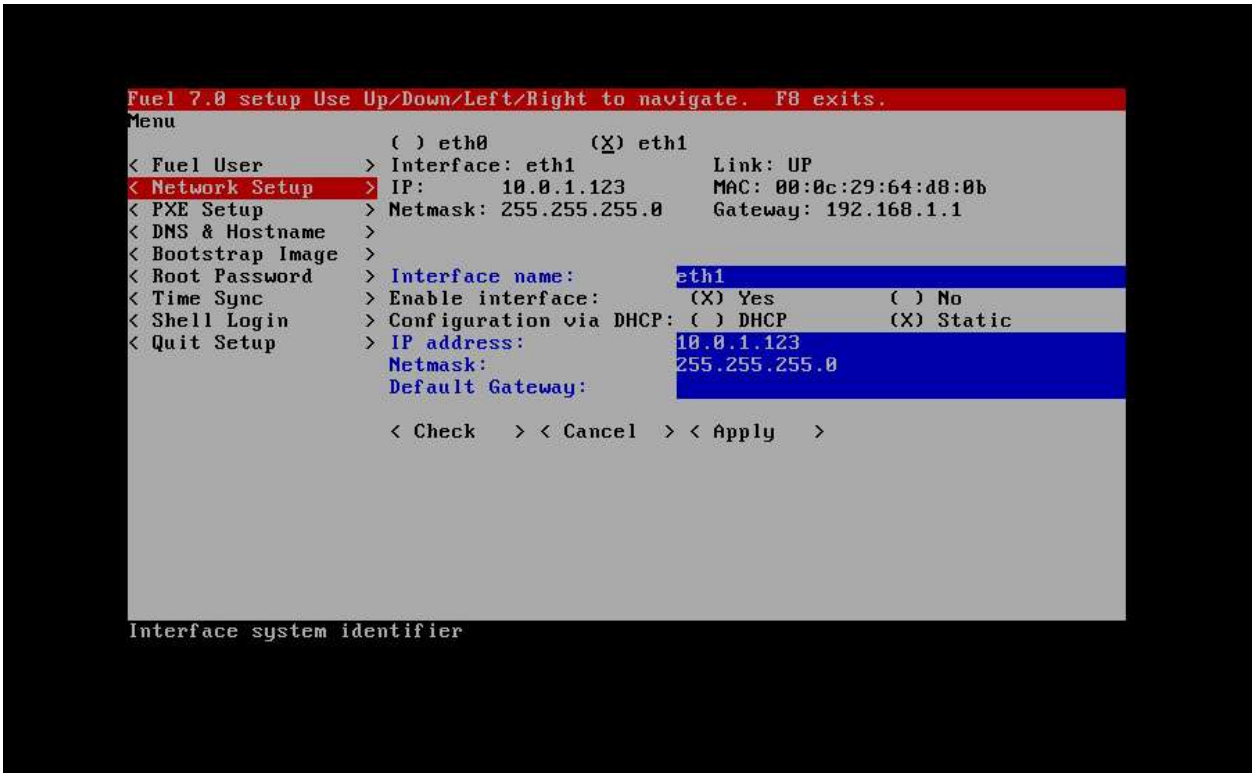
Step 4) We will be setting eth0 as public IP and using eth1 as DHCP sever.

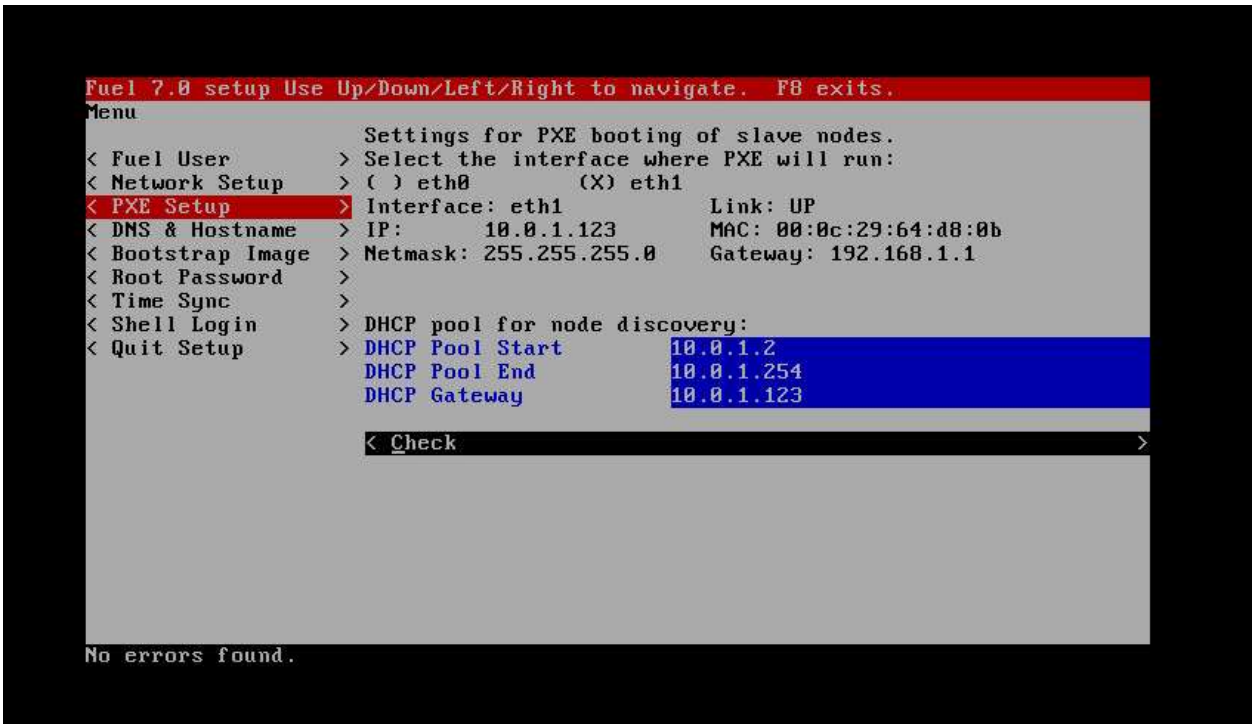
Here you may select the network interface you are going to use for PXE/Admin network and set DHCP pool ranges.

PXE Setup has 2 parts - editable PXE settings and current status information about the selected Network Interface that cannot be edited. NIC current status area shows the current network interface status, including name, Link Status, current IP address, MAC address, Netmask and Gateway. It also shows warnings, related to the currently selected NIC misconfiguration.

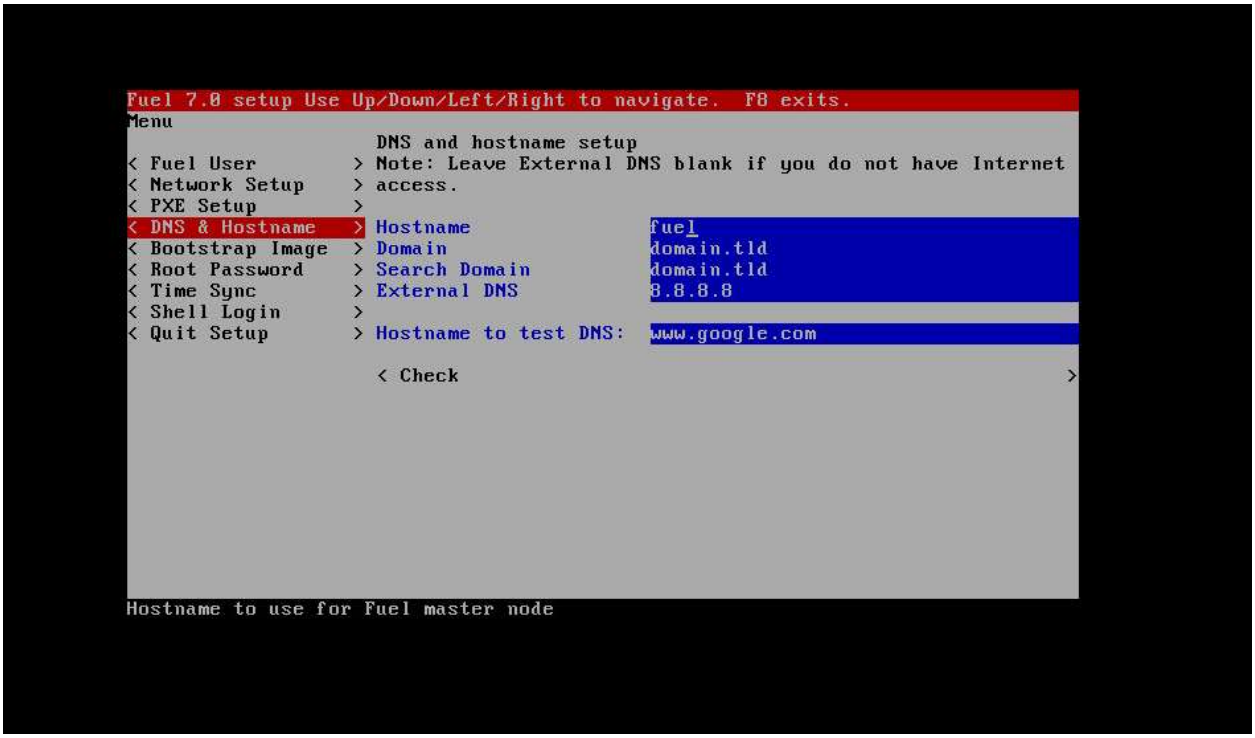
PXE setup includes the following options:

Network Interface Selector - Shows all available network interfaces, physical and virtual. Select the interface you want to configure with arrow keys and click Space or Enter to show its configuration.
DHCP Pool for node discovering - Here you may define DHCP Pool Start and End IP addresses. These addresses should be located inside the CIDR that is configured for the currently selected NIC.
Check button - verifies the current unsaved settings against the currently selected NIC without applying.

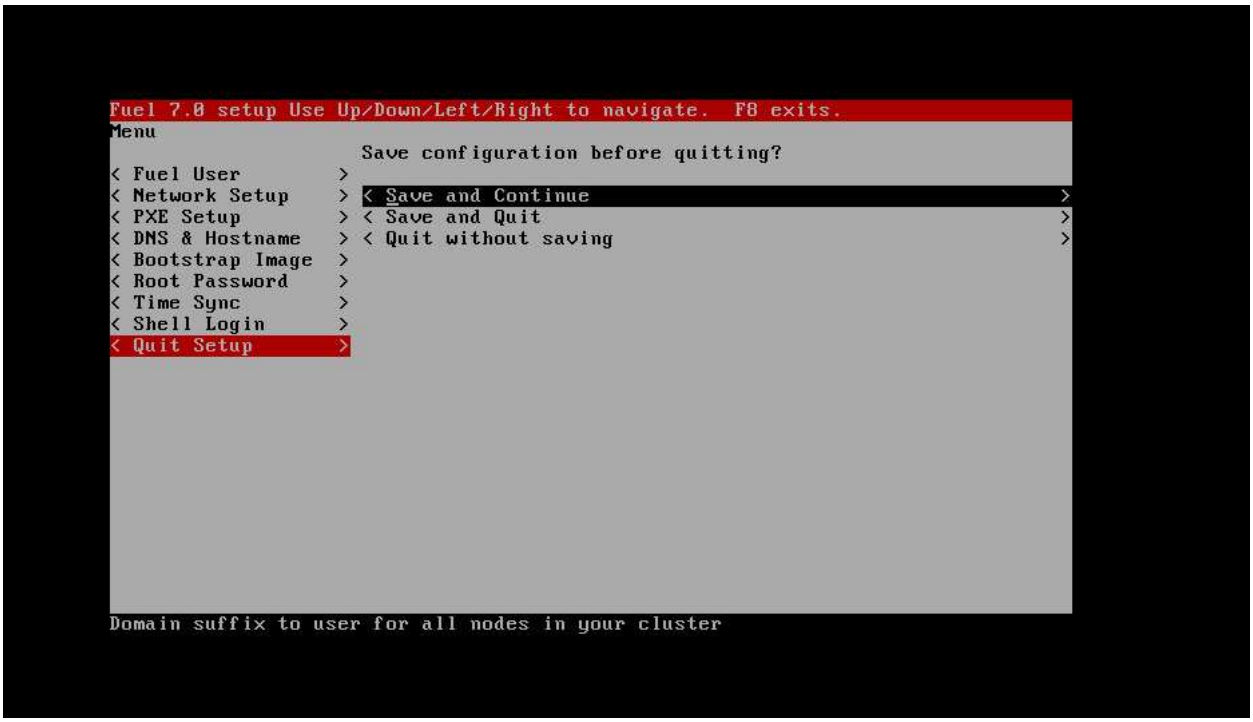




Step 5) Details on settings:
Hostname - master node host name (without domain)
Domain - master node domain name. If the master node has several network interfaces, you may connect non-PXE one to the existing corporate network and set the real domain name. Otherwise, use default or any valid stub name.
Search domain - in most cases, should match the Domain field, unless you know what you are doing.
External DNS - Point it to the corporate or Internet-based DNS server if your master node is connected to the corporate network by Non-PXE network interface. Otherwise - leave blank, since it may block Fuel Setup from network settings save due to failed DNS test.
Hostname to test DNS - any existing host name, which Fuel Setup may ping in order to check DNS settings.
Please do not hesitate to use Check button to verify your future network settings in advance.



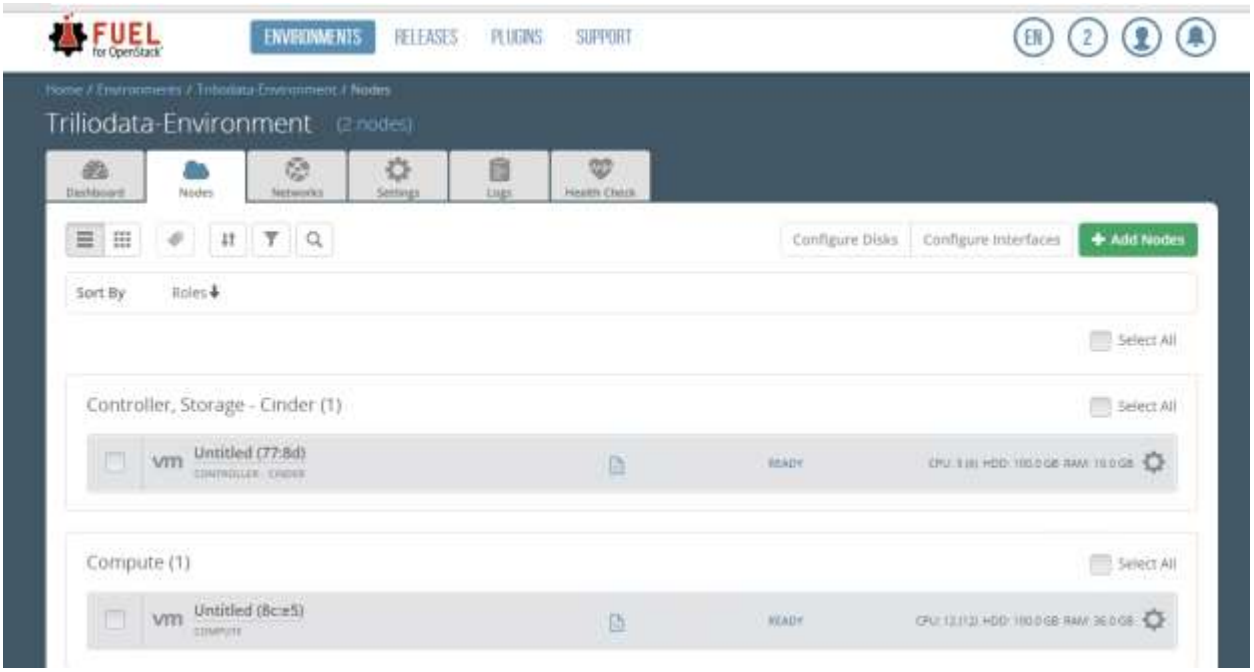
Step 6) Bootstrap image is Ubuntu by default. Also we are not changing root password.



Step 7) Bootstrap slave node's.

Once the slave nodes are up fuel master automatically provides them with a DHCP IP and bootstraps it. After bootstrap slave node is listed in Fuel GUI where we can proceed to assign role to the slave node in respective environment.
Note: These are the nodes that we are going to configure as controller, compute and cinder.

Step 4) Define roles for slave nodes.
Once we are logged in FUEL dashboard using our credentials. We can define roles to the nodes which are now visible in Fuel admin dashboard.
To test the functionality, we can define Controller, Compute and Cinder role's.



Step 5) Setup Network setting based on configuration and Verify network.
TrilioVault configuration we should have **access to keystone private and public endpoint**. So network should be setup in such a manner that we have access to private and public keystone endpoints. (Rest network configuration can be as user requirement)

Sample network example:

Public:
Here we are using range CIDR 192.168.1.0/25

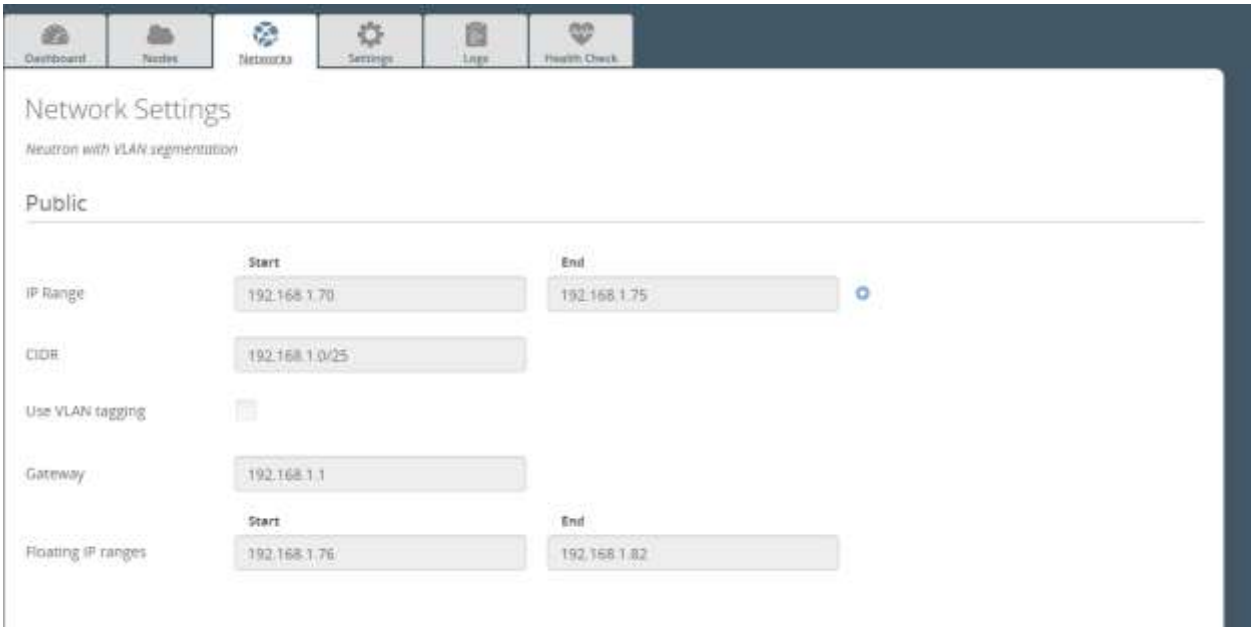
IP range start 192.168.1.70 and end 192.168.1.75
Gateway 192.168.1.1
Floating IP start 192.168.1.76 end 192.168.1.82

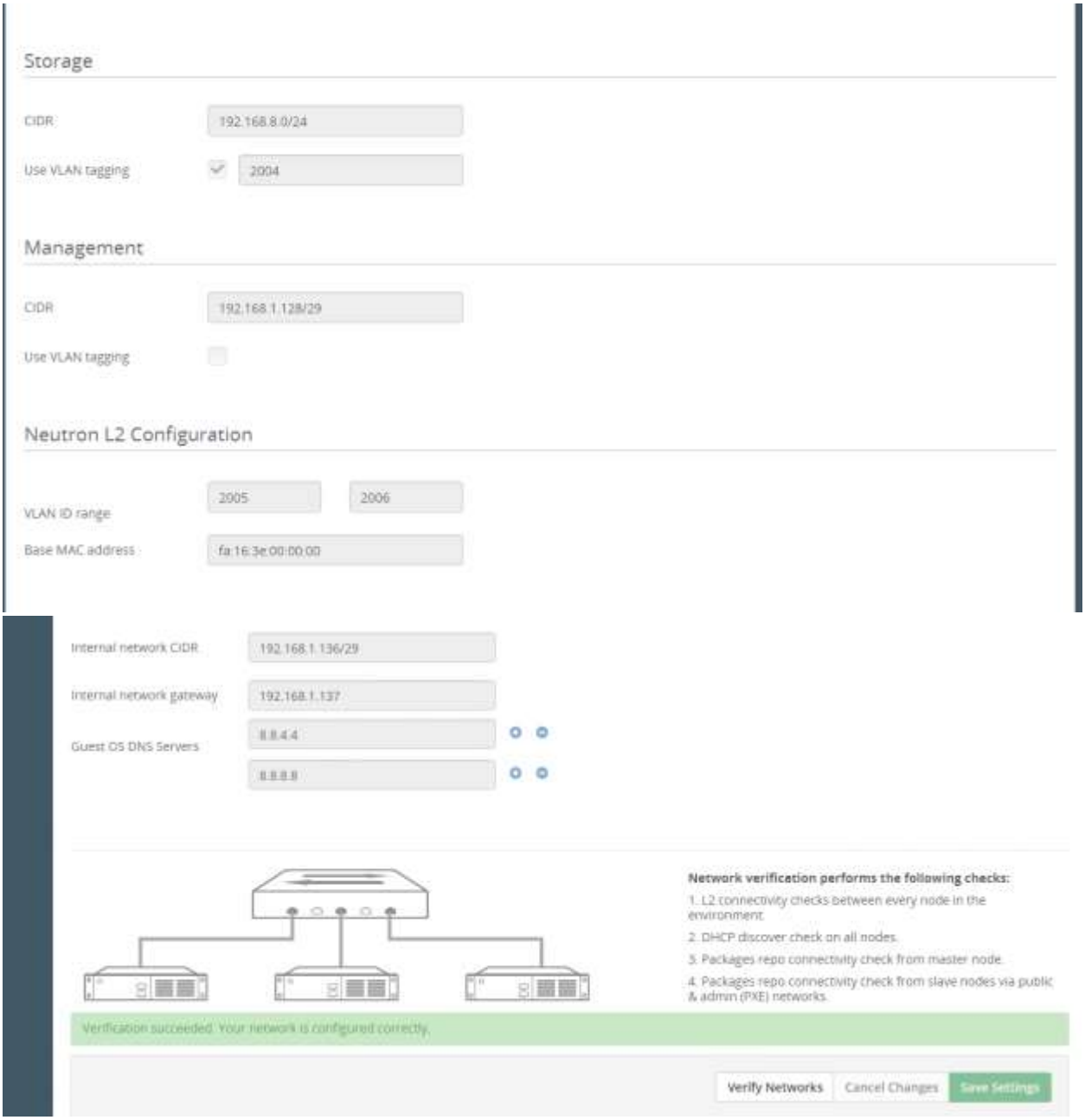
Storage network we can specify any network with or without VLAN tag.

Management network should be publically accessible which we should be used without VLAN tag.

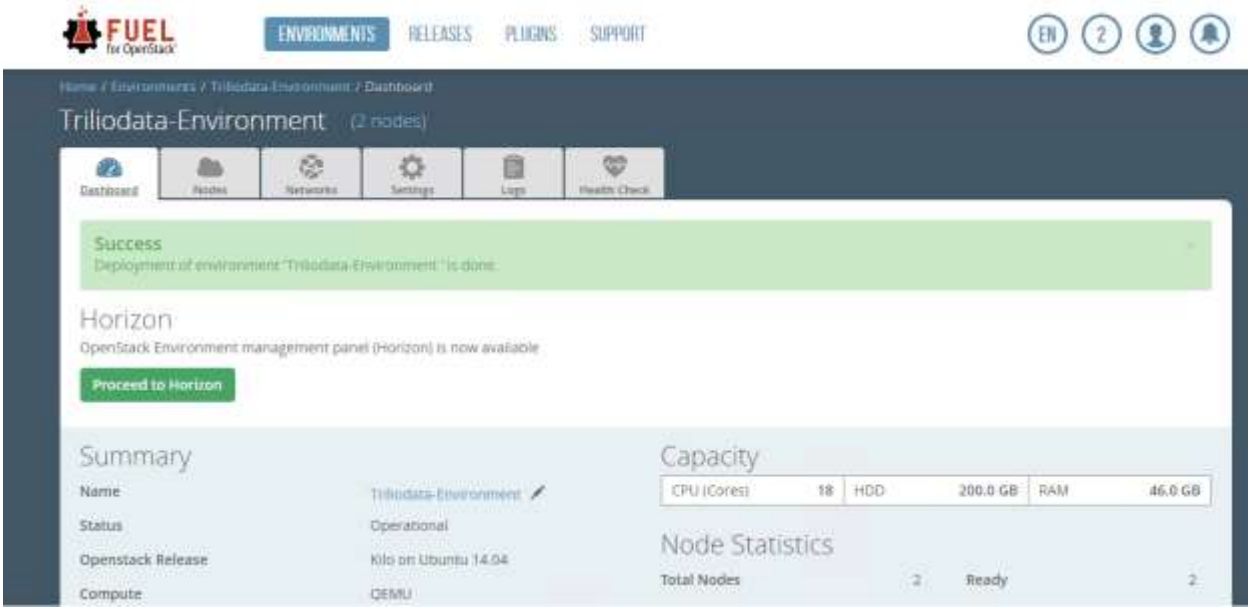
Neutron L2 configuration: we can specify VLAN ID range from 2005 to 2006

Neutron L3 configuration:
Internal network CIDR: 192.168.1.136/29
Internal network gateway 192.168.1.137
DNS: 8.8.8.8 and 8.8.4.4





Step 6) Once network Verification passed deploy OpenStack environment and start using it from the horizon link provided by FUEL.



5.2.1 Health Check Result:

Before Triliovault Integration:

Following test cases expected to fail:

- Check usage of default credentials on master node. Default credentials for ssh on master node were not changed. Please refer to OpenStack logs for more details.

- Check if default credentials for OpenStack cluster have changed. Default credentials values are used. We kindly recommend that you changed all defaults.

- Check usage of default credentials for keystone on master node. Default credentials for keystone on master node were not changed

5.3 TrilioVault installation steps

Installation Overview

TrilioVault is a product of Trilio Data for backup and disaster recovery. It is distributed to customers as a **QCOW2** image. The installation process is described in the below sections in detail.

To install TrilioVault, follow these steps.

1. Pre-installation tasks (KVM environment).
2. Configuring TrilioVault.

Install TrilioVault Nova Extension API on OpenStack controller node.
Install TrilioVault Nova Extension on all OpenStack compute nodes.
The above steps are described in sections below.

TrilioVault pre-installation tasks

You can install TrilioVault on directly on pure KVM.
To proceed with the TrilioVault installation, you need to setup a KVM box.

Install KVM (QEMU) on CentOS 7 / RHEL 7

KVM stands for Kernel Based Virtual Machine, is a virtualization software which provides ability to run a multiple guest operating systems with the help of hardware virtualization extensions. We will deploy Triliovault on standalone KVM box.

Steps to install KVM box:
Step 1) Issue the following command to install latest qemu package and also virt-manager which provides graphical interface to manage virtual machines.
Command: sudo yum install kvm virt-manager libvirt virt-install qemu-kvm xauth dejavu-lgc-sans-fonts

Step 2) For the networking part, our KVM-host will act as a router for its guests and we will need to create a bridge interface to allow the guest to communicate out of the host. Guests will use NAT on the host to connect to the real network. To allow such type of setup it's needed to allow ip forwarding in the kernel parameters.
Command: echo "net.ipv4.ip_forward = 1"|sudo tee /etc/sysctl.d/99-ipforward.conf
Command: sudo sysctl -p /etc/sysctl.d/99-ipforward.conf

Step 3) After allowing the host to do ip forwarding, we need to change the network configuration. Basically we will keep our original physical interface as it is but will assign its IP-address to the brige. In the example host-machine there is one real interface called

eno16777736 and the script in /etc/sysconfig/network-scripts/ifcfg-eno16777736 looks like this:

```
1 DEVICE="eno16777736"
2 ONBOOT=yes
3 IPADDR="192.168.202.111"
4 NETMASK="255.255.255.0"
5 GATEWAY="192.168.202.2"
6 HWADDR="00:0c:29:32:d0:4c"
7 DNS1="192.168.202.2"
```

The first thing to change here, is to comment out everything that is IP-related and tell the interface which interface will be the bridge. Resulting in /etc/sysconfig/network-scripts/ifcfg-eno16777736 to look like this:

```
  DEVICE="eno16777736"
2 ONBOOT=yes
3 #IPADDR="192.168.202.111"
4 #NETMASK="255.255.255.0"
5 #GATEWAY="192.168.202.2"
6 HWADDR="00:0c:29:32:d0:4c"
7 #DNS1="192.168.202.2"
8 BRIDGE=virbr0
```

Next, we can create the config-script for the bridge interface virbr0 in /etc/sysconfig/network-scripts/ifcfg-virbr0. Most details can be copied from the original script for eno16777736:

```
1 DEVICE="virbr0"
2 TYPE=BRIDGE
3 ONBOOT=yes
4 BOOTPROTO=static
5 IPADDR="192.168.202.111"
6 NETMASK="255.255.255.0"
7 GATEWAY="192.168.202.2"
8 DNS1="192.168.202.2"
```

Step 4) Finish and check the KVM installation
Basically all components are now ok but before KVM can be used it’s a good idea to perform a reboot in order to load the kvm-modules and to relaod the new network settings. After the reboot, we should check if the necessary kernel modules are loaded, which means that KVM successfully can handle the VM-extensions of our CPU:
Command: lsmod | grep kvm
Output expected:
kvm_intel 138567 0

kvm 441119 1 kvm_intel

Step 5) Check if the bridge is installed and in an up-state:
Command: ip a show virbr0
Should list your bridge information.

Step 6) Copy triliovault qcow2 image to the /opt folder and run following command:

Installing TrilioVault on KVM

KVM or Kernel Based Virtual Machine is a free virtualization tool on which TrilioVault can be installed. To install TrilioVault on KVM, follow these steps.

1. Download the image **tvault qcow2** and place it on a KVM appliance.
2. Make sure that the KVM server has following resources available:
    - RAM: 24 GB, VCPUS:4, Root Disk:40GB.
3. To launch **tvault node** execute the command mentioned below on the command prompt.

    virt-install -n TVault --memory 24576 --vcpus 4 --os-type linux --os-variant ubuntu14.04 --disk /opt/tvault-appliance-os-2.0.qcow2,device=disk,bus=virtio,size=40 -w bridge=virbr0,model=virtio --graphics none –import

4. Assign a IP to the TrilioVault/Tvault node by editing /etc/network/interfaces.d/eth0.cfg file.

Configuring TrilioVault

Once the Instance is successfully launched, point your web browser to the TrilioVault appliance. For this, open a new Browser with the url **https://floating-ip-address** (Floating IP Address was created in step 23). A landing page for TrilioVault is displayed as shown in Figure 1. Follow the instructions as given on the page to proceed with configuration.

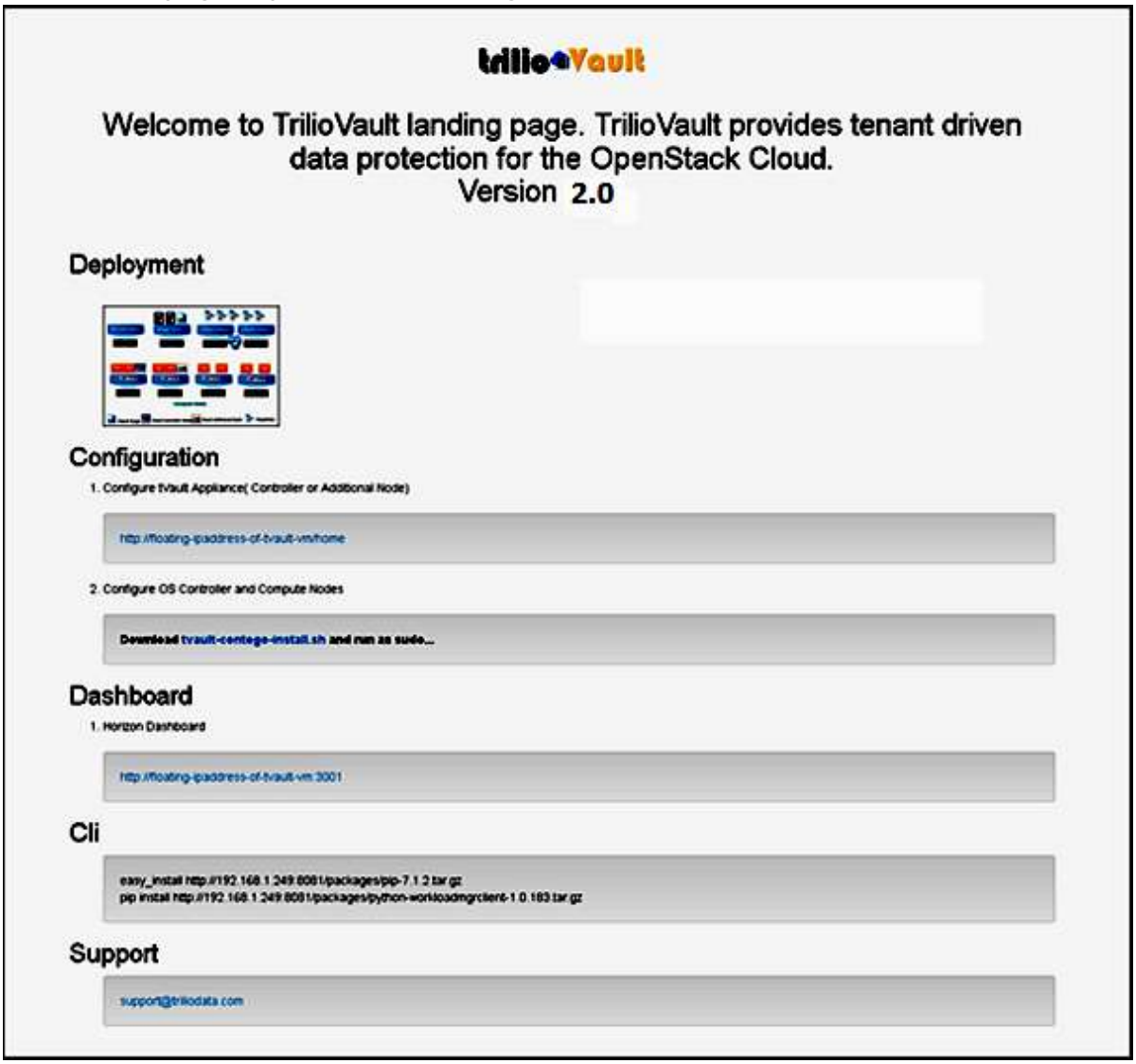


Figure 1: TrilioVault Configuration.

1. Right-click the hyperlink **https://floating-ip-address-of-tvm/home** under **Configure tVault Appliance** to open the link in new browser. The login screen of TVault is displayed as shown in Figure 2.

Figure 2: TrilioVault login.

2. Login as admin, Change Password screen is displayed as shown in Figure 3.

Figure 3: TrilioVault Change password.

3. Change the password and click **Submit**. TrilioVault home page is displayed as shown in Figure 4.

Figure 4: TrilioVault Home Page.

4. Click the Configuration link. **TrilioVault Appliance Configuration** screen is displayed as shown in Figure 5.

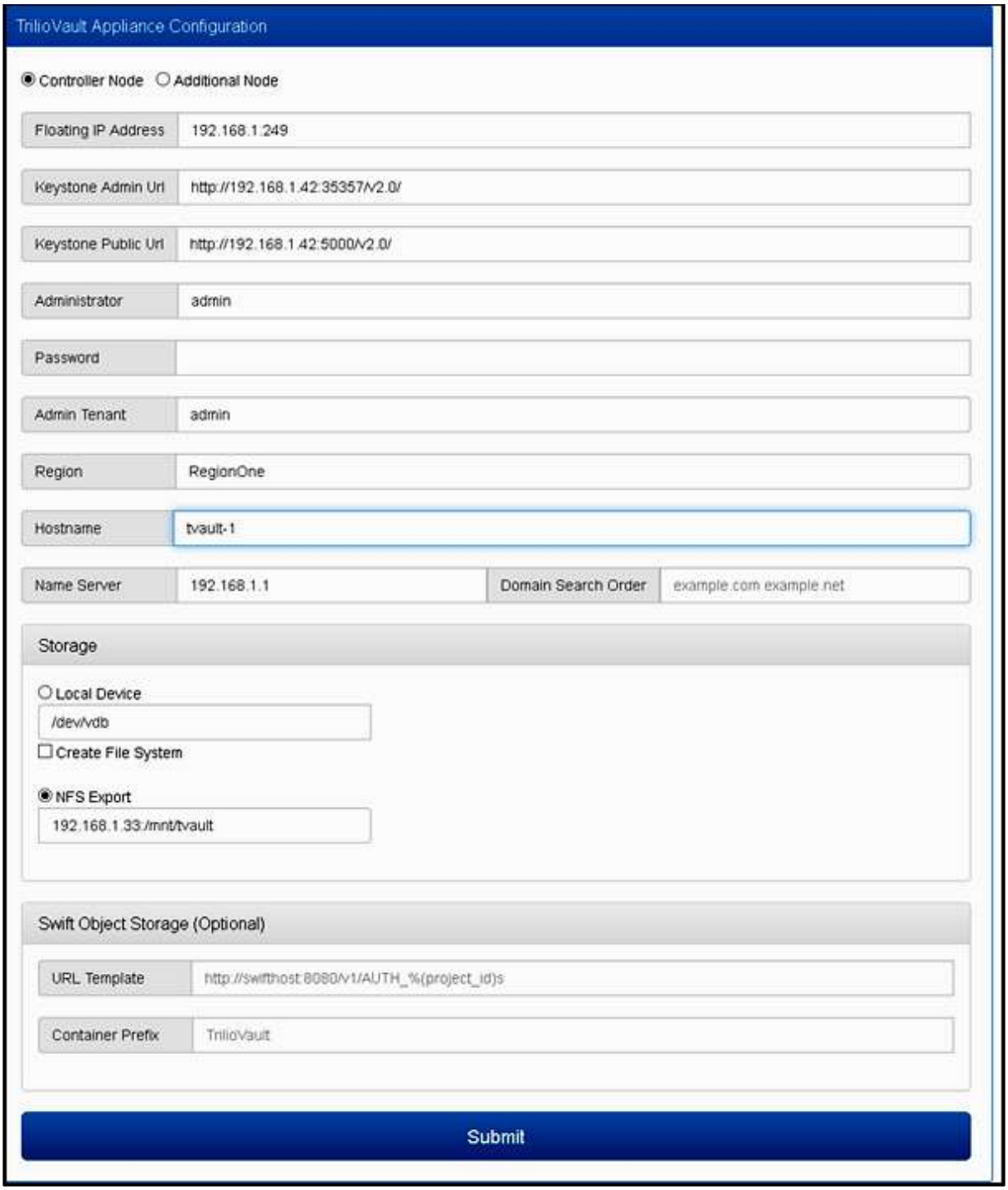


Figure 5: TrilioVault Configuration details.

Table 1 describes the detailed field description for Figure 5.

Table 1: TrilioVault Configuration details for Controller Node.

Field Name	Description
Floating IP Address	Provide the floating IP address of Tvault appliance In case of multi-node Tvault cluster, assume any one node as master node which is already configured and provide its IP for all other nodes.
Keystone Admin URL	Provide the Keystone Admin URL of users OpenStack setup. You can get this Admin URL from your **keystone endpoint-list** command on controller node of OpenStack.
Keystone Public URL	Provide the Keystone Public URL of users OpenStack setup. You can get this Admin URL from your **keystone endpoint-list** command on controller node of OpenStack.
Administrator	Provide the name of Administrator of OpenStack setup.
Password	Provide the password for the Administrator.
Admin Tenant	Provide the name of Admin Tenant of OpenStack setup.
Region	Provide the region where user wants to deploy this application. Default value is **RegionOne** for Kilo and **regionOne** for Icehouse.
Host Name	Provide the new name for the Host.
Name Server	Provide the IP address of Name Server.
Domain Search Order	Provide the name of the domain in which workload VMs should be searched.
Storage	Provide the storage space path where the user wants to store snapshots.

5. Click **Ok** when asked to continue. After configuration is over, the following screen is displayed as shown in Figure 6.

Figure 6: TrilioVault Configuration complete.

Click the link **Horizon Dashboard**. TrilioVault web page is displayed as shown in Figure 7.

Note Install **tvault-contego** on all the Controller and Computer nodes before accessing the Horizon Dashboard

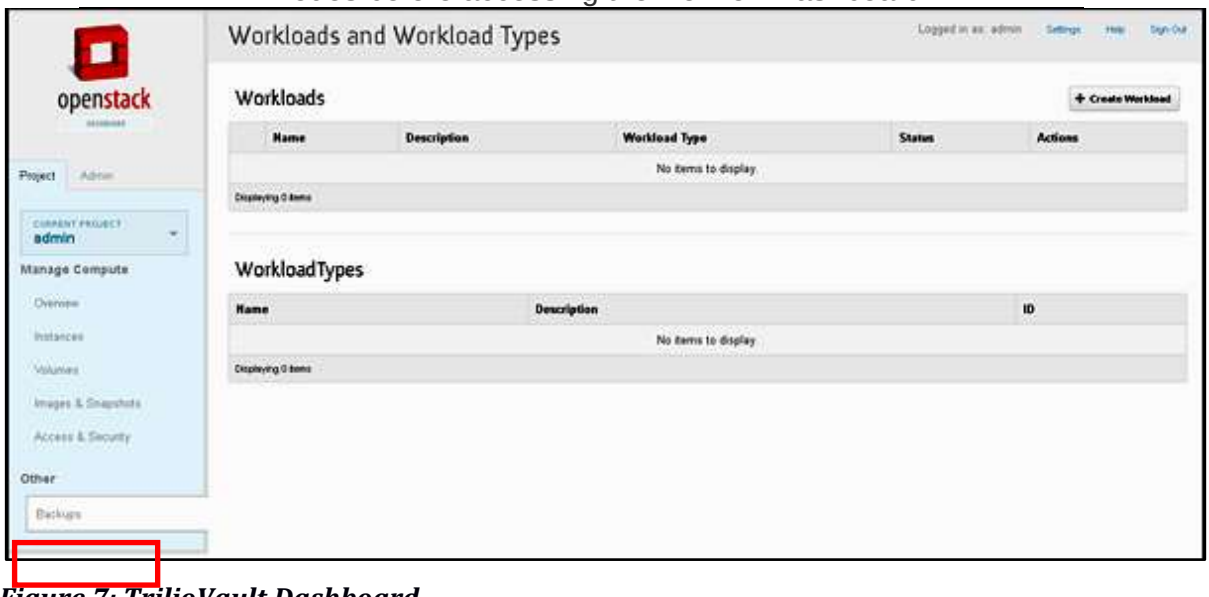


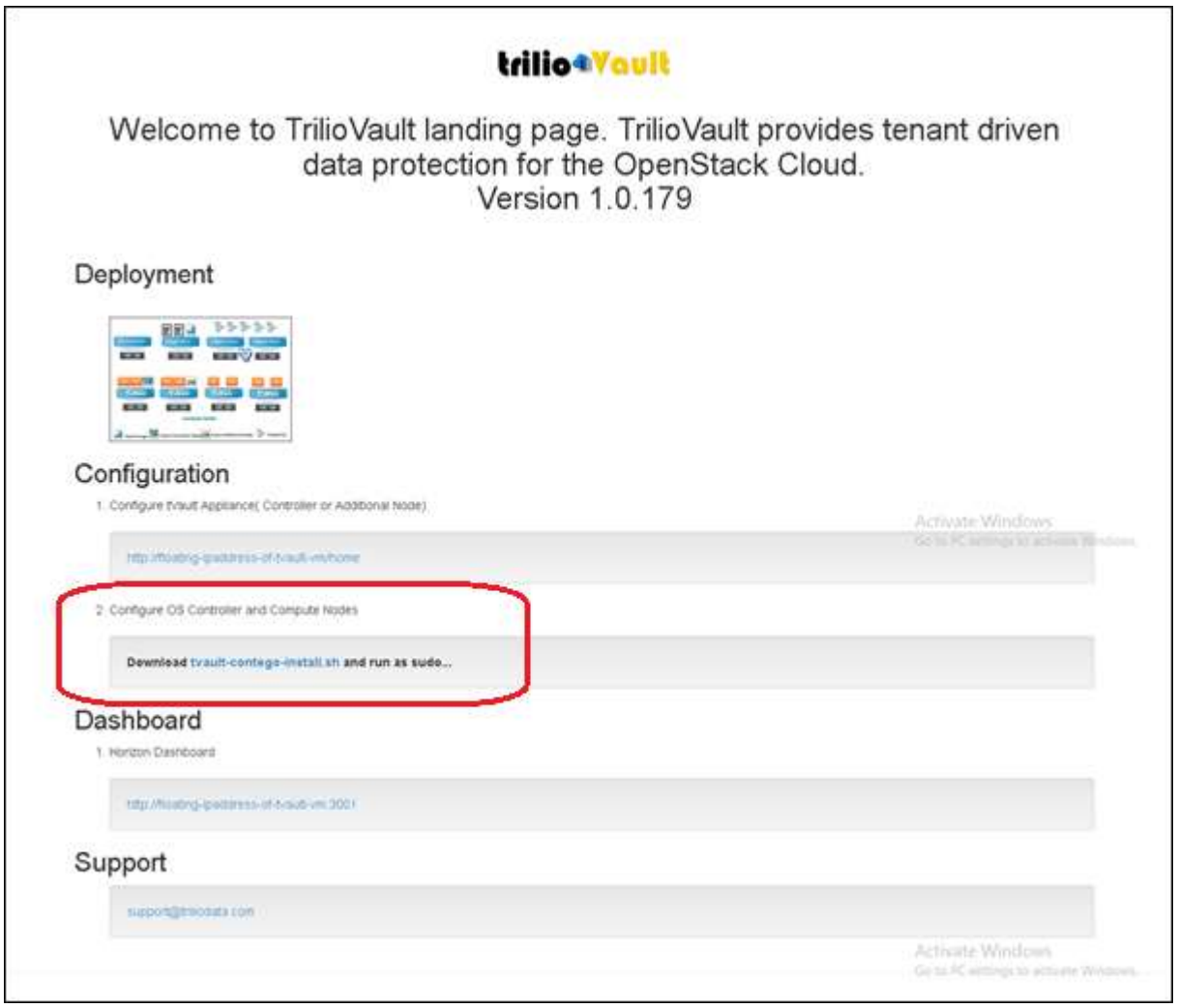
Figure 7: TrilioVault Dashboard.

Installing TrilioVault Nova Extension on Controller Node

It is necessary to install TrilioVault Nova Extension on the Controller node to proceed with the administrative tasks. To install **TrilioVault Nova Extension** on **Controller node**, follow these steps.

Note	The values in RED are user-entered values.

1. Navigate to **https://floating-ip-address-of-tvm** screen.
2. Copy download link of tvault-contego-install.sh from **Download tvault-contego-install.sh** (To download right click and do copy link address on landing VM page)



3. Navigate to the command prompt of the Controller node and use wget command to download tvault-contego-install script as shown below:
   wget https://<floating-ipaddress-of-triliovault-controller>/tvault-contego-install.sh --no-check-certificate
   Note: wget and paste link which we copied along with –no-check-certificate

4. Assign executable permissions to script.
   chmod +x tvault-contego-install.sh
5. Execute script to install tvault-contego API.
   ./tvault-contego-install.sh

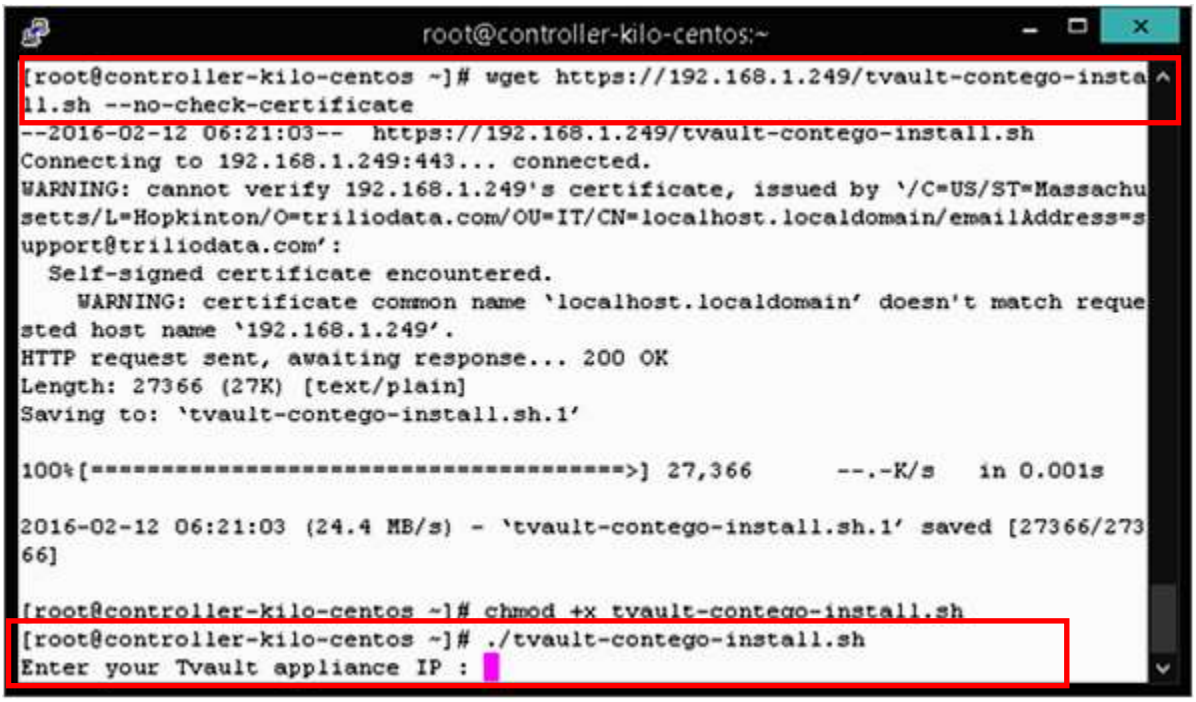


Figure 8: Installation at Controller node.

a. A script will prompt you to enter **Tvault appliance IP** shown in Figure 8.
b. Enter the Floating IP Address and press **Enter**. The screen as displayed in Figure 9 is displayed.
c. You will see a message for selecting the Option as **Controller** or **Compute**.

Figure 9: Enter IP on Controller node.

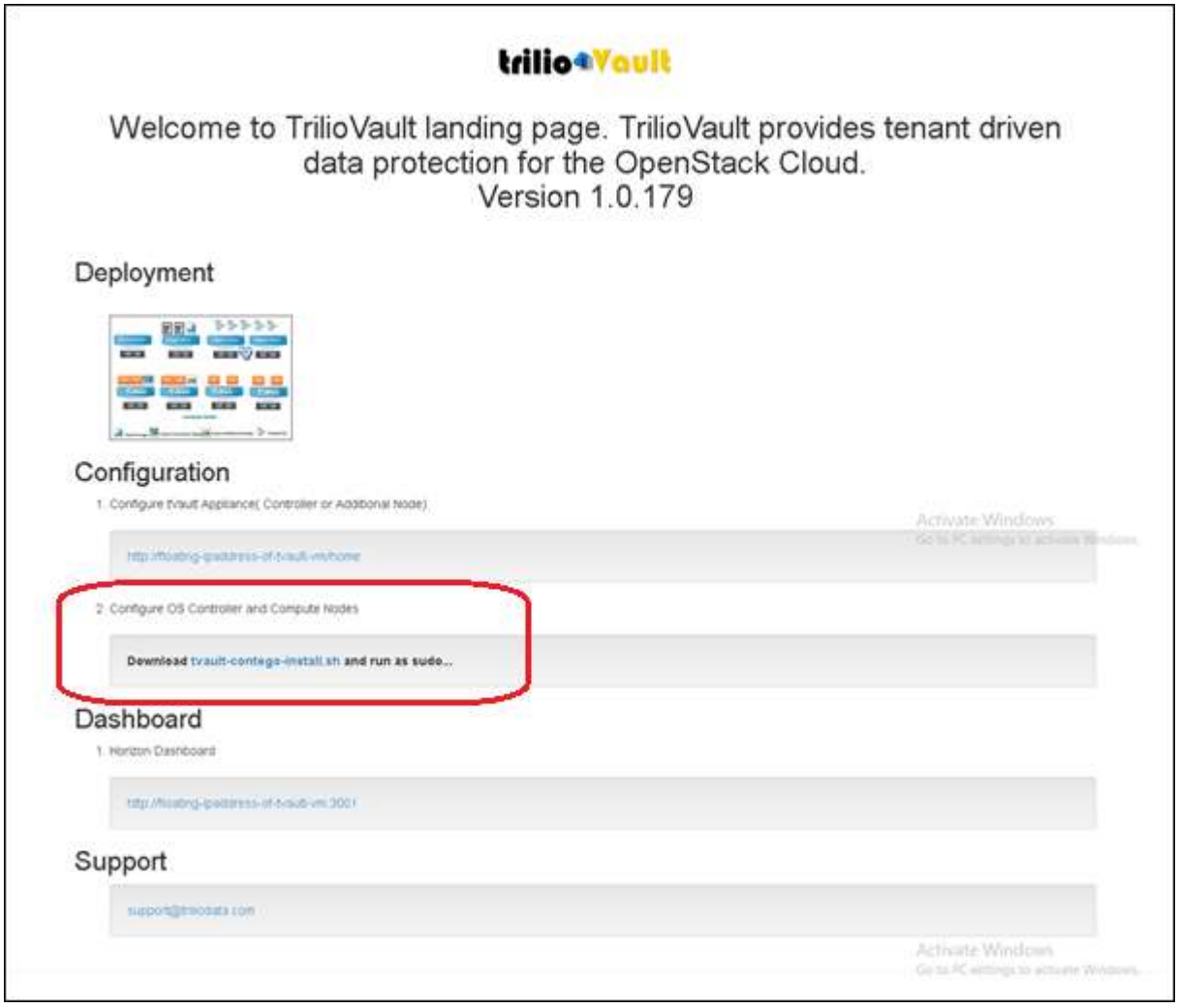
d. We want to use **Controller**. Type the number 1 and press Enter.
The tvault Nova Extension will be installed on the Controller node.

Installing TrilioVault Nova Extension on Computer Node

To install TrilioVault Nova Extension on **Computer** node, follow these steps.

Note The values in RED are user entered values.

1. Navigate **https://floating-ip-address** screen.
2. Copy download link of tvault-contego-install.sh from **Download tvault-contego-install.sh**. (To download right click and do copy link address on landing VM page)



3. Navigate to the command prompt of the controller node and use wget command to download tvault-contego-install script as shown below

wget https://<floating-ipaddress-of-triliovault-controller> /tvault-contego-install.sh --no-check-certificate

4. Assign executable permissions to script as shown in Figure 10.

chmod +x tvault-contego-install.sh

5. Execute script to install tvault-contego as shown in Figure 10.

./tvault-contego-install.sh

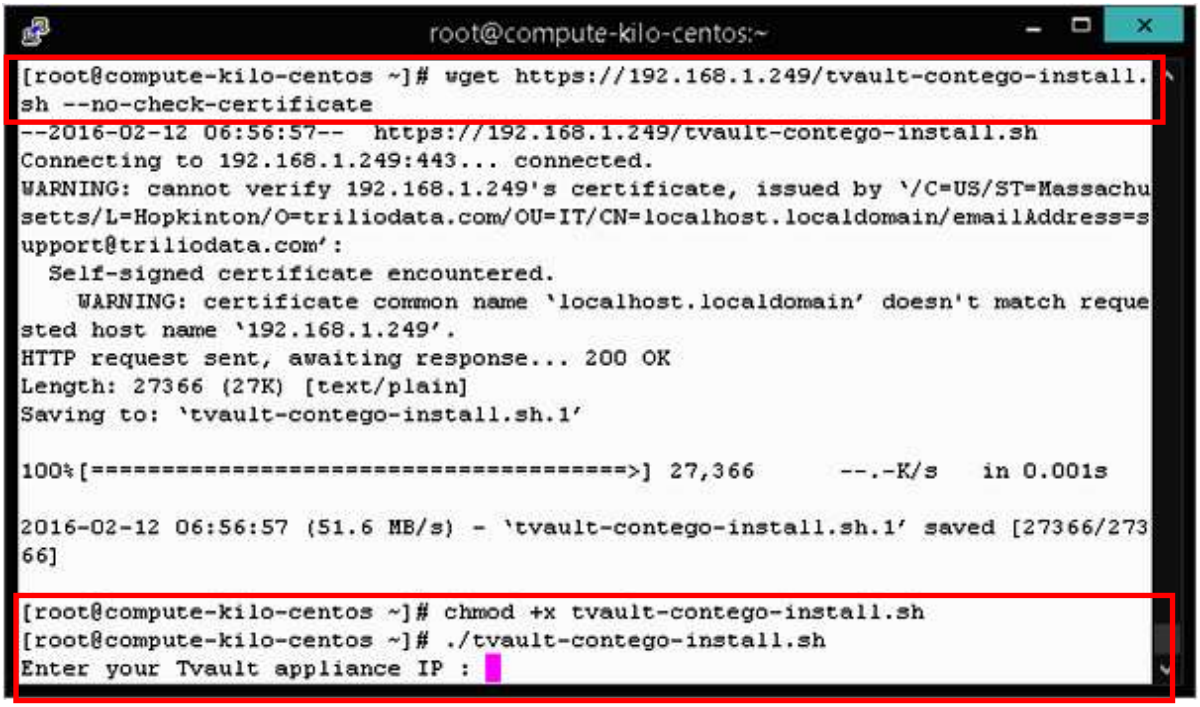


Figure 10: Installation at Computer node.

a. You will see a message for entering the Tvault appliance IP.
b. Enter the Floating IP Address and press Enter.
c. You will see a message for selecting the Option as **Controller** or **Compute** as shown in Figure 11.

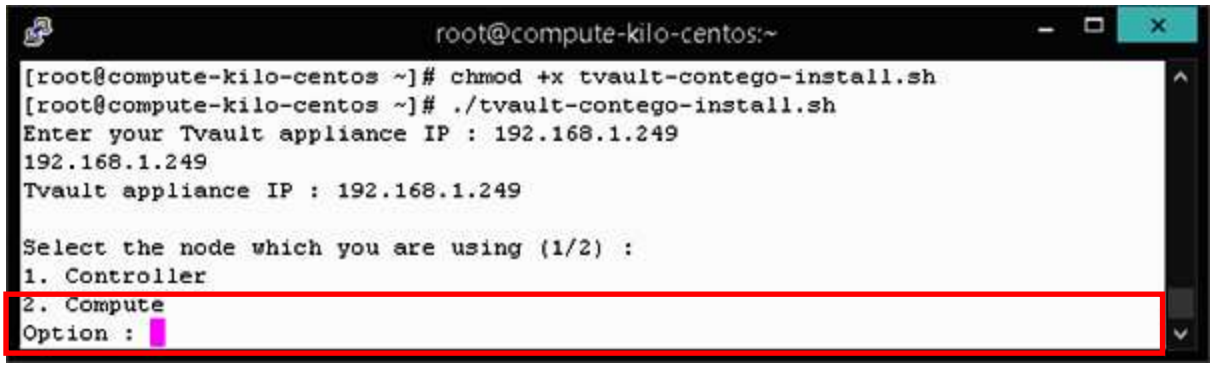


Figure 11: Entering IP on Computer node.

d. We want to use **Computer**. Type the number 2 and press **Enter**.

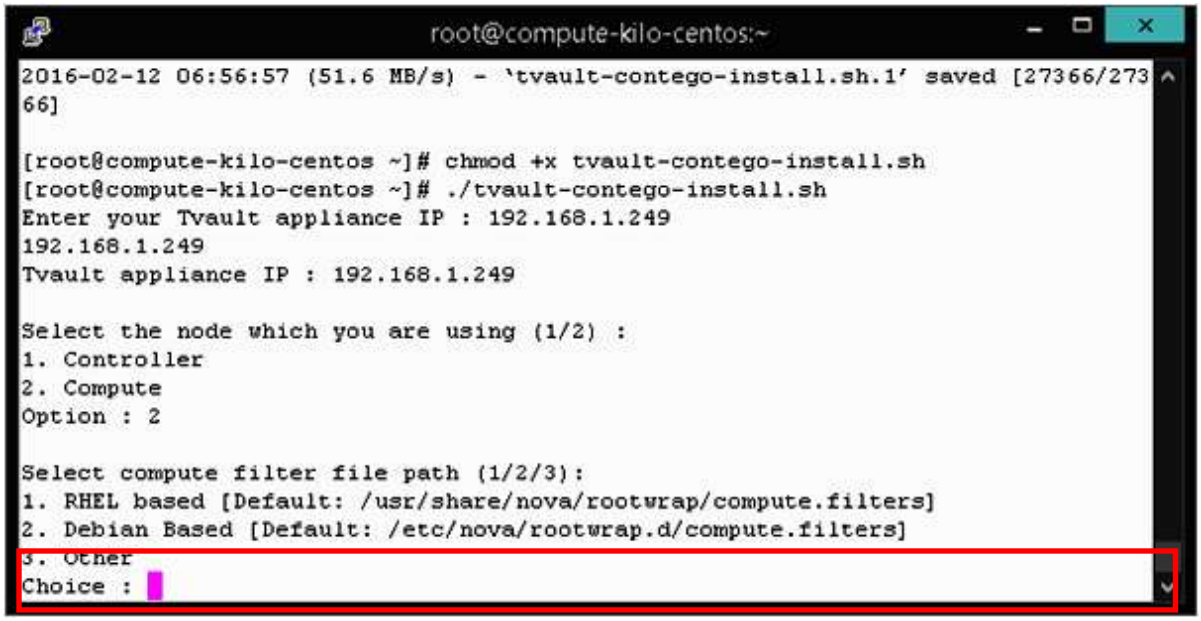


Figure 12: Compute file path option.

e. We enter 1. (Selection is based on your OpenStack based operating system) Please select 2 when you are using Fuel verison 7 which supports Ubuntu.

Figure 13: Selection of OpenStack backup media.

f. Our backup media is NFS so we enter 1.

Figure 14: Compute node NFS share path.

g. Enter the NFS share path as entered in Figure 5 of **Error! Reference source not found.** on page **Error! Bookmark not defined.**.

The **tvault Nova Extension** will be installed on the **Compute** node.

h. Use the URL **https://floating-ip-address-of-tvm/home** to launch TrilioVault.

TrilioVault Configuration File

The TrilioVault Nova Extension nodes can be installed directly by running the configuration file on the Controller or Compute nodes. Configuration file contains all the data that is required while the command is being executed. To run the configuration file in Controller node, follow these steps.

Controller Node:

To run configuration file in **Controller** node, follow these steps.

1. On the command prompt of the Compute node execute the command **./tvault-contego-install.sh –auto**

Note	Make the following changes in your auto configuration file.

Enter your Tvault appliance IP : < floating-ipaddress-of-tvault-node >
Option : 1 (when asked to enter the option of controller or compute node)

Computer Node:

To run configuration file in **Computer** node, follow these steps.

2. On the command prompt of the Compute node execute the command **./tvault-contego-install.sh –auto**

Note	Make the following changes in your auto configuration file.

Enter your Tvault appliance IP : < floating-ipaddress-of-tvault-node >

Option : 2 (when asked to enter the option of controller or compute node)

Choice : 1 (when asked to enter the option for file path)

Select the type of backup media (1/2) :
1. NFS
2. Swift
Option : 1

Enter NFS share path [Format: IP:/path/to/nfs_share] : <NFS storage path entered in configuration details screen>

Configuring Additional Node

Additional Node is a footprint of OpenStack scalability and availability. However, there must be at least one controller node. Additional Node helps to process tvault operations faster and parallel thereby distributing its load across nodes. It has same interface that of

Controller node, with the exception of adding an extra compute engine. To configure Additional Node, follow these steps.

1. Launch Instance and **Associate floating IP address** as mentioned in steps 16 through 24.
2. Launch tvault with this new floating ip by login as Admin.
3. Right-click the hyperlink **https://floating-ip-address-of-tvm/home** under **Configure tVault Appliance** to open the link in new browser. The login screen of TVault is displayed as shown in Figure 2.
4. Click the Configuration link. **TrilioVault Appliance Configuration** screen is displayed as shown in Figure 15.
5. Select the checkbox **Additional Node**.

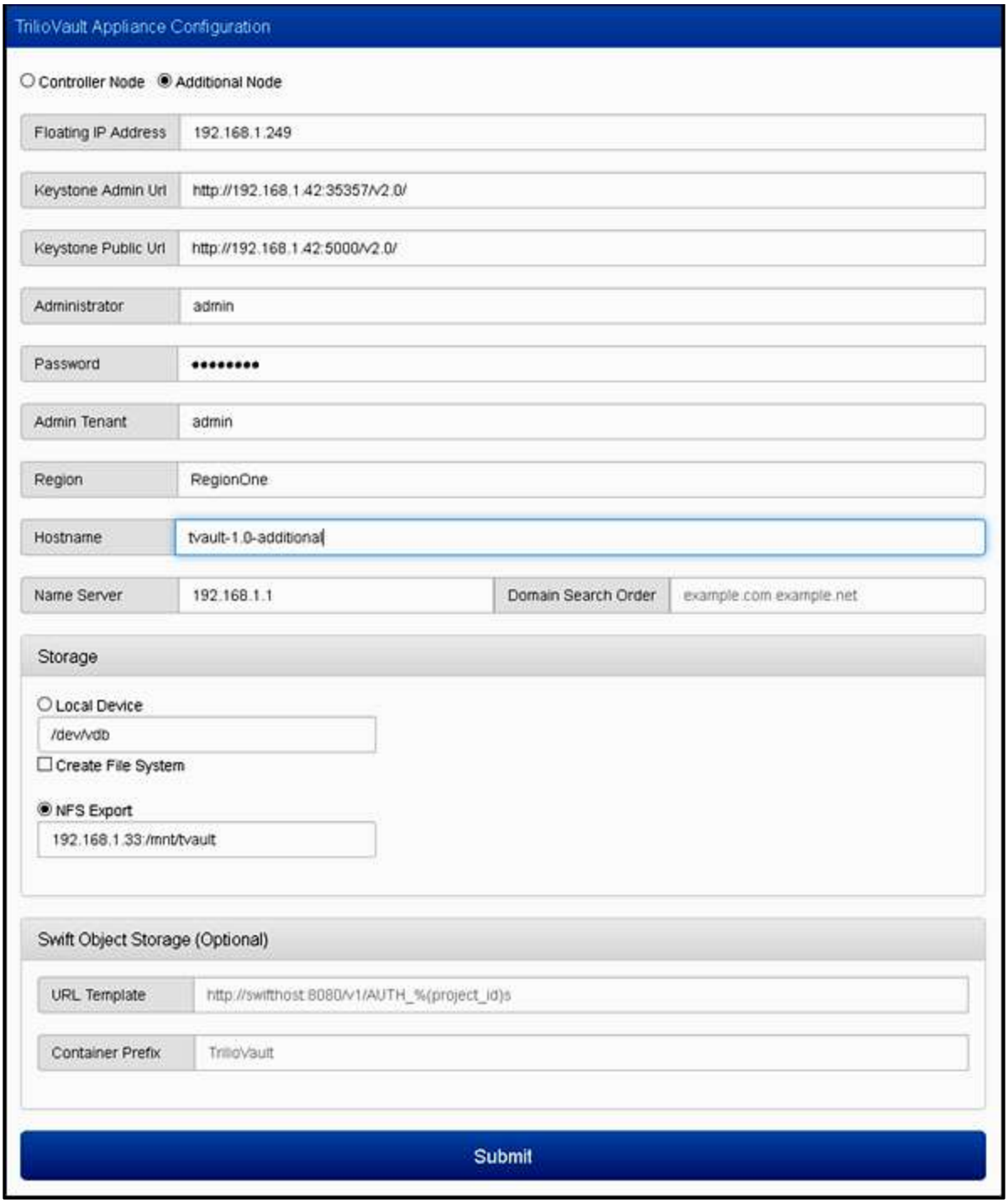


Figure 15: Configuring Additional TrilioVault Node.

Table 2: TrilioVault Configuration details for Additional Node.

Field Name	Description
Floating IP Address	Provide the floating IP address of tvault controller node. In case of multi-node tvault cluster, assume any one node as master node which is already configured and provide *its* IP for all other nodes.
Keystone Admin	Provide the Keystone Admin URL of users OpenStack setup.
Keystone Public URL	Provide the Keystone Public URL of users OpenStack setup.
Administrator	Provide the name of Administrator of OpenStack setup.
Password	Provide the password of the Administrator.
Admin Tenant	Provide the name of Admin Tenant of OpenStack setup.
Region	Provide the region where user wants to deploy this application. Default value is **RegionOne** for Kilo and **regionOne** for Icehouse.
Host Name	Provide the new name for the Host.
Name Server	Provide the IP address of Name Server.
Domain Search Order	Provide the name of the Domain in which workload VMs should be searched.
Storage	Provide storage space path where a user wants to store snapshots.

6. Click **Ok** when asked to continue. After configuration is over, the following screen is displayed as shown in Figure 6.

Post Installation Checklist

To ensure that the Tvault is installed successfully

1. Run the following command on Compute node command prompt.
   service tvault-contego status
   If it is installed successfully it will display active status in green color. If there is a problem, it will show failed status in red color.
2. Login to TrilioVault Dashboard. If you are unable to login, refer **Error! Reference source not found.** section on page **Error! Bookmark not defined.**.

5.4 Limitations

TrilioVault is a backup and recovery solution from Trilio Data. To run on OpenStack, the following build is required.

- **QCOW2** – can be deployed in the same or different OpenStack environment.

Hardware Requirements

Hardware requirements of TrilioVault are mentioned below.

Table 3: Hardware Requirements.

TrilioVault Appliance - QCOW2	
Flavor of TrilioVault Appliance	Storage: 40 GB Memory: 24 GB vCPUS: 4 Ephemeral Disk: 100 GB Swap Disk: 1024 MB

Software Requirements

Software requirements of TrilioVault are mentioned below.

Table 4: Software Requirements.

TrilioVault Appliance - QCOW2	
OpenStack Release	Icehouse or Kilo
OpenStack Distribution	Mirantis Fuel 7
	Ubuntu 14.04.4 LTS (Trusty Tahr)

After Triliovault Integration:

Following test case expected to fail:

- Check usage of default credentials on master node. Default credentials for ssh on master node were not changed. Please refer to OpenStack logs for more details.

- Check if default credentials for OpenStack cluster have changed. Default credentials values are used. We kindly recommend that you changed all defaults.

- Check usage of default credentials for keystone on master node. Default credentials for keystone on master node were not changed

5.5 Testing

5.5.1 Test cases

TVAUT-1: Test backup of image booted instance.
TVAUT-2: Test restore of image booted instance backup.
TVAUT-3: Test backup of volume booted instance.
TVAUT-4: Test restore of volume booted instance backup.
TVAUT-5: Test data integrity after restore for image booted instance.
TVAUT-6: test data integrity after restore for volume booted instance.

TVAUT-7: Test selective restore to different internal network.
TVAUT-8: Test selective restore to different volume type.
TVAUT-9: Test backup of multi instance.
TVAUT-10: test selective restore to restore single instance from multiple backed up.
TVAUT-11: Test restore to same key-pair.
TVAUT-12: Test selective restore to different flavor.
TVAUT-13: Test installation of Horizon Plugin in OpenStack.
TVAUT-14: Test edit/Modify Workload after creation.
TVAUT-15: Verify scheduler for different timestamp.
TVAUT-16: Verify retention policy.
TVAUT-17: Verify full backup interval policy.
TVAUT-18: Test file-manager deployment.
TVAUT-19: Test mount/unmount snapshot for file level access.
TVAUT-20: Test Email notification's.
TVAUT-21: Test Serial/Parallel workload snapshot/restore.
TVAUT-22: Verify admin panel functionality.
TVAUT-23: Test tVault API Service/ tVault Scheduler Service/ tVault Workload Service Stop/Start from Trilioault Service on configuration.
TVAUT-24: Test tenant level backup/restore.
TVAUT-25: Test tvault-contego(agent) installation on controller node.
TVAUT-26: Test tvault-contego(agent) installation on compute node.
TVAUT-27: Test 20 instance snapshot in single workload.
TVAUT-28: Test 20 instance restore from single workload.
TVAUT-29: Test 200GB Volume filled backup/restore.
TVAUT-30: Test 200GB Volume empty backup/restore.
TVAUT-31: Test 20 Workload backup simultaneously.
TVAUT-32: Test 50 Workload backup simultaneously.
TVAUT-33: Test backup qemu-agent enabled in backed up VM.
TVAUT-34: Test incremental snapshot/backup.
TVAUT-35: Test restore from incremental snapshot/restore.
TVAUT-36: Test CLI for Workload creation.
TVAUT-37: Test CLI for snapshot/backup specific workload.
TVAUT-38: Test CLI for restore specific workload.
TVAUT-39: Test import workload for selective workload.
TVAUT-40: Test import workload for all workload.
TVAUT-41: Test Triliovault upgrade.
TVAUT-42: Test reconfigure with new configuration.
TVAUT-43: Test NTP after TrilioVault configuration.
TVAUT-44: Verify logs are accessible through configuration page itself.
TVAUT-45: Test TrilioVault additional node configuration.
TVAUT-46: Test OpenStack network is unaffected via TrilioVault.
TVAUT-47: Test Instance creation works fine in OpenStack after snapshot/backup.
TVAUT-48: Test Instance creation works fine in OpenStack after restore.
TVAUT-49: Test tvault-contego uninstall on controller and compute node.

TVAUT-50: Test workload/snapshot/restore delete.

5.5.2 Test Results

TVAUT-1: Passed
TVAUT-2: Passed
TVAUT-3: Passed
TVAUT-4: Passed
TVAUT-5: Passed
TVAUT-6: Passed
TVAUT-7: Passed
TVAUT-8: Passed
TVAUT-9: Passed
TVAUT-10: Passed
TVAUT-11: Passed
TVAUT-12: Passed
TVAUT-13: Passed
TVAUT-14: Passed
TVAUT-15: Passed
TVAUT-16: Passed
TVAUT-17: Passed
TVAUT-18: Passed
TVAUT-19: Passed
TVAUT-20: Passed
TVAUT-21: Passed
TVAUT-22: Passed
TVAUT-23: Passed
TVAUT-24: Passed
TVAUT-25: Passed
TVAUT-26: Passed
TVAUT-27: Passed
TVAUT-28: Passed
TVAUT-29: Passed
TVAUT-30: Passed
TVAUT-31: Passed
TVAUT-32: Passed
TVAUT-33: Passed
TVAUT-34: Passed
TVAUT-35: Passed
TVAUT-36: Passed
TVAUT-37: Passed
TVAUT-38: Passed

TVAUT-39: Passed
TVAUT-40: Passed
TVAUT-41: Passed
TVAUT-42: Passed
TVAUT-43: Passed
TVAUT-44: Passed
TVAUT-45: Passed
TVAUT-46: Passed
TVAUT-47: Passed
TVAUT-48: Passed
TVAUT-49: Passed
TVAUT-50: Passed