

Guide to the ScaleIO Cinder plugin ver. 1.0.0 for Fuel

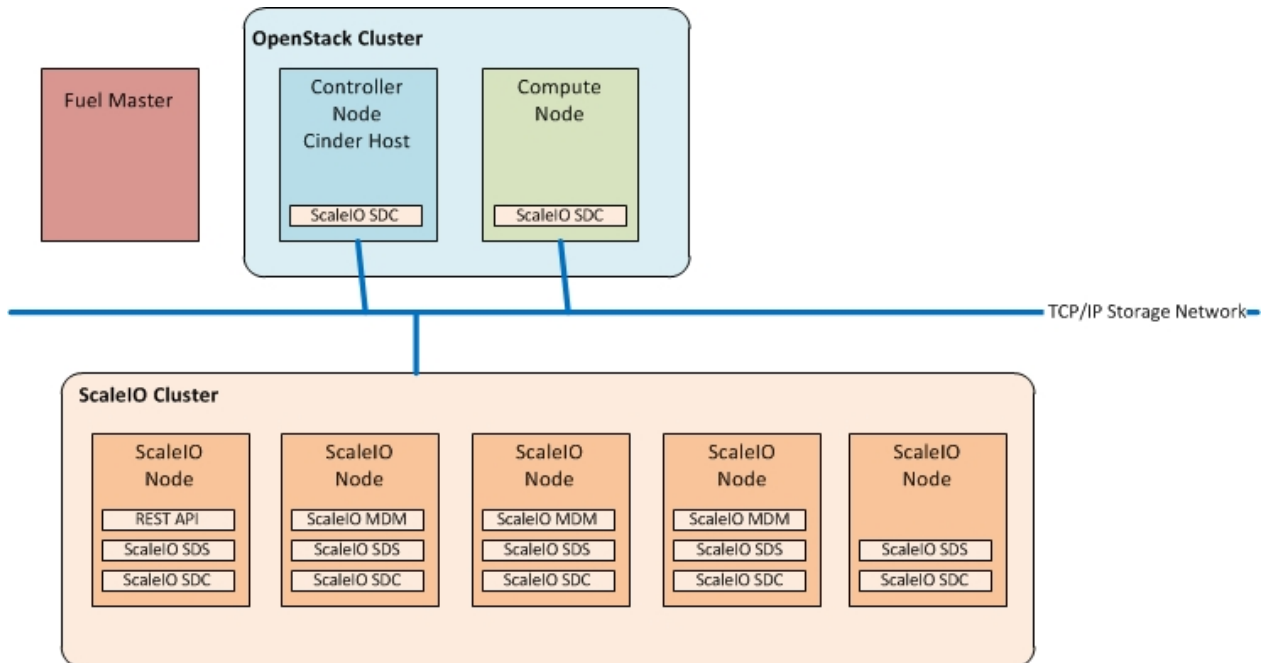
Introduction

This guide provides instructions on installing, configuring and using ScaleIO Cinder plugin for Fuel. The ScaleIO plugin allows OpenStack to work with an **External** ScaleIO deployment. This ScaleIO plugin for Fuel extends Mirantis OpenStack functionality by adding support for ScaleIO block storage.

ScaleIO is a software-only solution that uses existing servers local disks and LAN to create a virtual SAN that has all the benefits of external storage, but at a fraction of cost and complexity. ScaleIO utilizes the existing local internal storage and turns it into internal shared block storage.

Overview

The following diagram shows the plugin's high level architecture:



From the figure we can see that we need the following OpenStack roles and services:

OpenStack roles and services

Service Role/Name	Description	Installed in
Controller Node + Cinder Host	A node that runs network, volume, API, scheduler, and image services. Each service may be broken out into separate nodes for scalability or availability. In addition this node is a Cinder Host, that contains the Cinder Volume Manager	OpenStack Cluster

Compute Node	A node that runs the nova-compute daemon that manages Virtual Machine (VM) instances that provide a wide range of services, such as web applications and analytics	OpenStack Cluster
--------------	--	-------------------

In the **external ScaleIO cluster** we have installed the following roles and services:

ScaleIO cluster roles and services

Service Role	Description	Installed in
ScaleIO Gateway (REST API)	The ScaleIO Gateway Service, includes the REST API to communicate storage commands to the ScaleIO Cluster, in addition this service is used for authentication and certificate management.	ScaleIO Cluster
Meta-data Manager (MDM)	Configures and monitors the ScaleIO system. The MDM can be configured in redundant Cluster Mode, with three members on three servers, or in Single Mode on a single server.	ScaleIO Cluster
Tie Breaker (TB)	'Tie Breaker service helps determining what service runs as a master vs. a slave.'	ScaleIO Cluster
Storage Data Server (SDS)	Manages the capacity of a single server and acts as a back-end for data access. The SDS is installed on all servers contributing storage devices to the ScaleIO system. These devices are accessed through the SDS.	ScaleIO Cluster
Storage Data Client (SDC)	A lightweight device driver that exposes ScaleIO volumes as block devices to the application that resides on the same server on which the SDC is installed.	Openstack Cluster

Note: for more information in how to deploy a ScaleIO Cluster, please refer to the ScaleIO manuals located in the [download packages](#) for your platform and [watch the demo](#).

Requirements

These are the plugin requirements:

Requirement	Version/Comment
Mirantis OpenStack compatibility	6.1
ScaleIO Version	>= 1.32
Controller and Compute Nodes' Operative System	CentOS/RHEL 6.5

OpenStack Cluster (Controller/cinder-volume node) can access ScaleIO Cluster	via a TCP/IP Network
OpenStack Cluster (Compute nodes) can access ScaleIO Cluster	via a TCP/IP Network
Install ScaleIO Storage Data Client (SDC) in Controller and Compute Nodes	Plugin takes care of install

Limitations

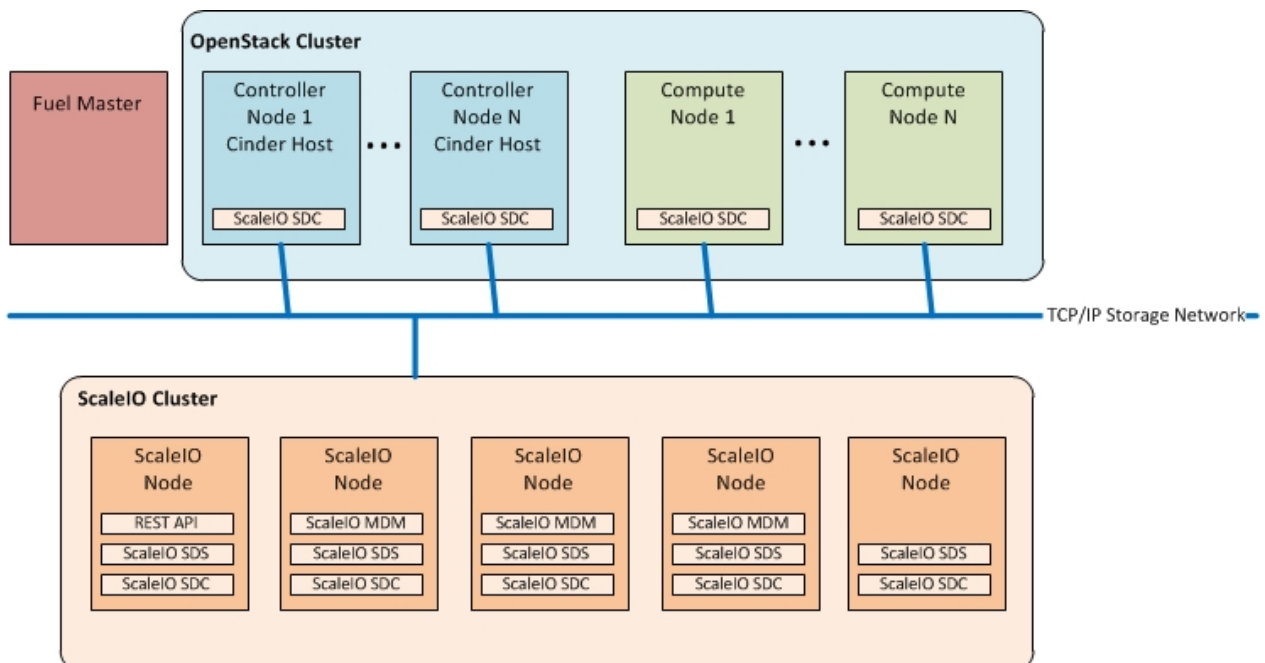
Currently Fuel doesn't support multi-backend storage.

Configuration

Plugin files and directories:

File/Directory	Description
Deployment_scripts	Folder that includes the bash/puppet manifests for deploying the services and roles required by the plugin
Deployment_scripts/puppet	
environment_config.yaml	Contains the ScaleIO plugin parameters/fields for the Fuel web UI
metadata.yaml	Contains the name, version and compatibility information for the ScaleIO plugin
pre_build_hook	Mandatory file - blank for the ScaleIO plugin
repositories/centos	Empty Directory, the plugin scripts will download the required CentOS packages
repositories/Ubuntu	Empty Directory, not used
taks.yaml	Contains the information about what scripts to run and how to run them

This Fuel plugin will install the ScaleIO Storage Data Client (SDC) service on each Controller node and Compute node in the cluster. This is necessary in order for the VMs in each compute node to utilize ScaleIO Storage:



Before starting a deployment there are some things that you should verify:

1. Your ScaleIO Cluster can route 10G Storage Network to all Compute nodes as well as the Cinder Control/Manager node.
2. An account on the ScaleIO cluster is created to use as the OpenStack Administrator account (use the login/password for this account as san_login/password settings).
3. The IP address from the ScaleIO cluster is obtained.

Install ScaleIO Cinder Plugin

To install the ScaleIO-Cinder Fuel plugin:

1. Download it from the [Fuel Plugins Catalog](#).
2. Copy the *rpm* file to the Fuel Master node:

```
[root@home ~]# scp scaleio-cinder-1.0-1.0.0-1.noarch.rpm root@fuel:/tmp
```

3. Log into Fuel Master node and install the plugin using the [Fuel CLI](#):

```
[root@fuel ~]# fuel plugins --install scaleio-cinder-1.0-1.0.0-1.noarch.rpm
```

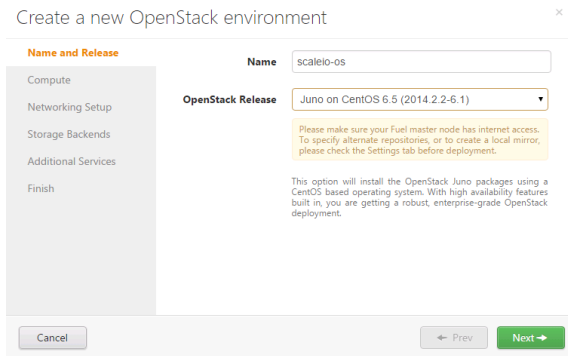
4. Verify that the plugin is installed correctly:

```
[root@fuel-master ~]# fuel plugins
id | name           | version | package_version
---|-----|-----|-----
1  | scaleio-cinder | 1.0.0   | 1.0.0
```

Configure ScaleIO plugin

Once the plugin has been copied and installed at the Fuel Master node, you can configure the nodes and set the parameters for the plugin:

1. Start by creating a new OpenStack environment following the [Mirantis OpenStack User Guide](#).
2. Configure your environment.



3. Open the **Settings** tab of the Fuel web UI and scroll down the page. Select the Fuel plugin checkbox to enable ScaleIO Cinder plugin for Fuel:

Deploy ScaleIO on a cluster

ScaleIO repo URL	<input type="text"/>	Repo URL for ScaleIO software
UserName	<input type="text"/>	ScaleIO Admin User Name
Password	<input type="text"/>	ScaleIO Admin Password
ScaleIO GW IP	<input type="text"/>	ScaleIO Gateway IP
ScaleIO Primary IP	<input type="text"/>	ScaleIO Primary MDM IP
ScaleIO Secondary IP	<input type="text"/>	ScaleIO Secondary MDM IP
ScaleIO protection domain	<input type="text"/>	Protection domain for ScaleIO
ScaleIO storage pool 1	<input type="text"/>	First storage pool for ScaleIO
Fault sets list	<input type="text"/>	Comma separated list of fault sets

Parameter name	Parameter description
userName	The ScaleIO User name
Password	The ScaleIO password for the selected user name
ScaleIO GW IP	The IP address of the the ScaleIO Gateway service
ScaleIO Primary IP	The ScaleIO cluster's primary IP address
ScaleIO Secondary IP	The ScaleIO cluster's secondary IP address
ScaleIO protection domain	Name of the ScaleIO's protection domain
ScaleIO storage pool 1	Name of the first storage pool

Note

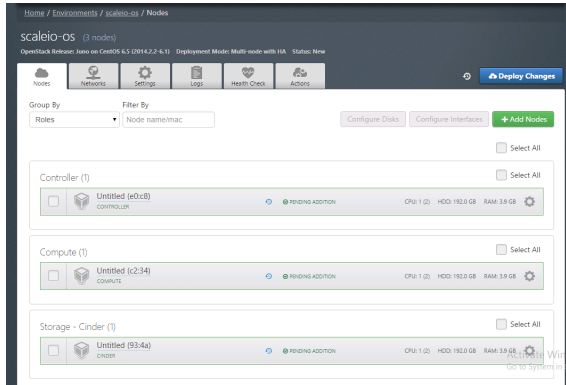
Please refer to the ScaleIO documentation for more information on these parameters.

This is an example of the ScaleIO configuration parameters populated:

Deploy ScaleIO on a cluster

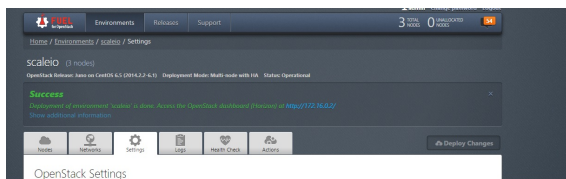
ScaleIO repo URL	<input type="text" value="http://10.247.52.47"/>	Repo URL for ScaleIO software
UserName	<input type="text" value="admin"/>	ScaleIO Admin User Name
Password	<input type="text" value="Password123"/>	ScaleIO Admin Password
ScaleIO GW IP	<input type="text" value="192.168.33.103"/>	ScaleIO Gateway IP
ScaleIO Primary IP	<input type="text" value="192.168.33.103"/>	ScaleIO Primary MDM IP
ScaleIO Secondary IP	<input type="text" value="192.168.33.101"/>	ScaleIO Secondary MDM IP
ScaleIO protection domain	<input type="text" value="default"/>	Protection domain for ScaleIO
ScaleIO storage pool 1	<input type="text" value="default"/>	First storage pool for ScaleIO
Fault sets list	<input type="text" value="fault"/>	Comma separated list of fault sets

4. After the configuration is done, you can add the nodes to the Openstack Deployment.



5. You can run the network verification check and **deploy changes** then.

6. After deployment is completed, you should see a success message:

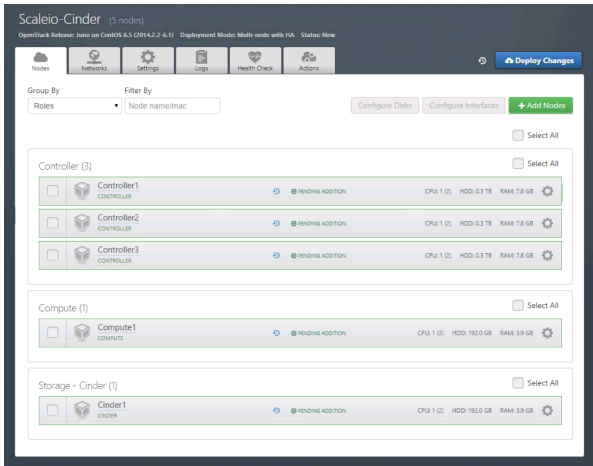


Note

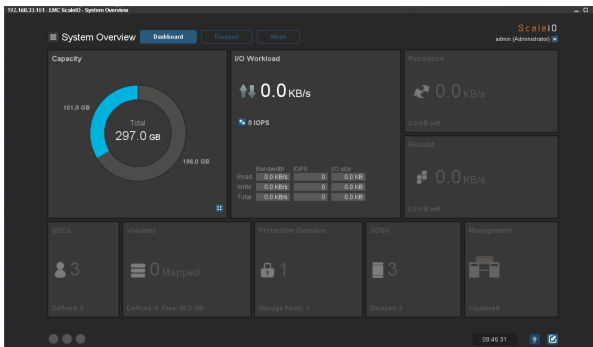
It may take an hour or more for the OpenStack deployment to complete, depending on your hardware configuration.

User Guide

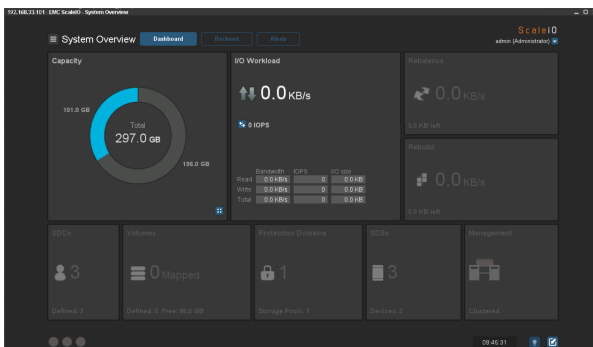
1. Install ScaleIO-Cinder plugin using the [Installation Guide](#).
2. Create environment with enabled plugin in fuel ui, lunch the fuel site and check setting section to make sure the Scaleio-Cinder section exists
3. Add 3 nodes with Controller role and 1 node with Compute and another role:



4. Picture of the External ScaleIO Cluster Running:



5. Retrieve the external ScaleIO Cluster information. For our example these are the configuration settings:



6. Use the ScaleIO Cluster information to update the ScaleIO Plugin information:

Fuel plugin for ScaleIO Cinder

UserName <input type="text" value="admin"/>	ScaleIO Admin User Name
Password <input type="text" value="Password123"/>	ScaleIO Admin Password
ScaleIO GW IP <input type="text" value="192.168.33.103"/>	ScaleIO Gateway IP
ScaleIO Primary IP <input type="text" value="192.168.33.101"/>	ScaleIO Primary MDM IP
ScaleIO Secondary IP <input type="text" value="192.168.33.103"/>	ScaleIO Secondary MDM IP
ScaleIO protection domain <input type="text" value="default"/>	Protection domain for ScaleIO
ScaleIO storage pool 1 <input type="text" value="default"/>	First storage pool for ScaleIO
Fault sets list <input type="text" value="11,72,53"/>	Comma separated list of fault sets

7. Apply network settings

8. Use the networking settings that are appropriate for your environment. For our example we used the default settings provided by Fuel:

The screenshot shows the Fuel web interface with the following configuration sections:

- Management:** CIDR: 192.168.0.0/24, Use VLAN tagging: 101
- Neutron L2 Configuration:** Start: 1000, End: 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: 10.247.186.39, 8.8.8.8

Below the configuration is a network diagram showing a central switch connected to three servers. A text box titled "Network verification performs the following checks:" lists:

1. L2 connectivity checks between every node in the environment.
2. DHCP discover check on all nodes.
3. Packages repo connectivity check from master node.
4. Packages repo connectivity check from slave nodes via public & admin (PXE) networks.

Buttons at the bottom: Verify Networks, Cancel Changes, Save Settings.

9. Run network verification check:

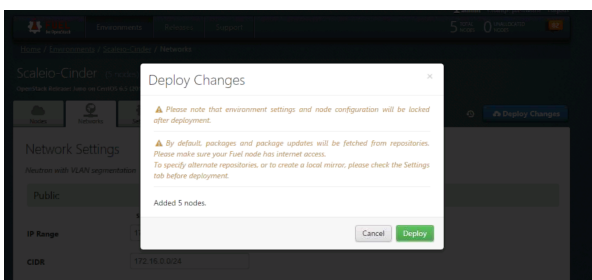
The screenshot shows the Fuel web interface with the network verification results:

- Neutron L3 Configuration:** Internal network CIDR: 192.168.111.0/24, Internal network gateway: 192.168.111.1, Guest OS DNS Servers: 10.247.186.39, 8.8.8.8

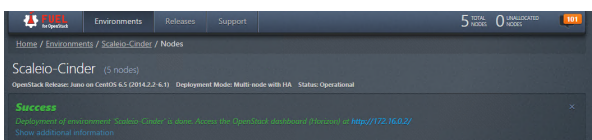
The network diagram is the same as in the previous screenshot. A green banner at the bottom states: "Verification succeeded. Your network is configured correctly."

Buttons at the bottom: Verify Networks, Cancel Changes, Save Settings.

10. Deploy the cluster:



11. Once the deployment finished successfully, open OpenStack Dashboard (Horizon):



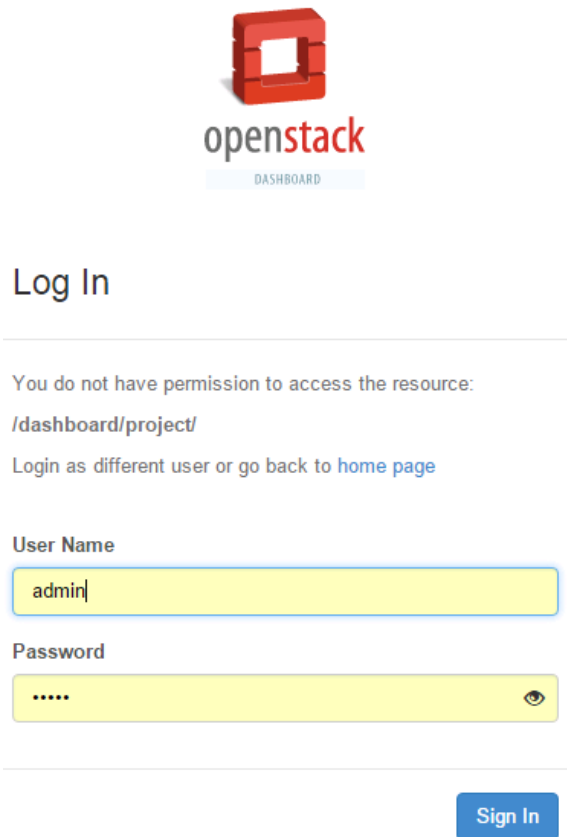
12. Check Storage tab under system information and make sure ScaleIO service is up and running:

Name	Host	Zone	Status	State
cinder-scheduler	node-33.domain.tld	nova	Enabled	Up
cinder-scheduler	node-34.domain.tld	nova	Enabled	Up
cinder-scheduler	node-35.domain.tld	nova	Enabled	Down
cinder-volume	node-31.domain.tld	nova	Enabled	Up
cinder-volume	node-36.domain.tld@ScaleIO	nova	Enabled	Up
cinder-scheduler	node-30.domain.tld	nova	Enabled	Up
cinder-volume	node-30.domain.tld@ScaleIO	nova	Enabled	Up

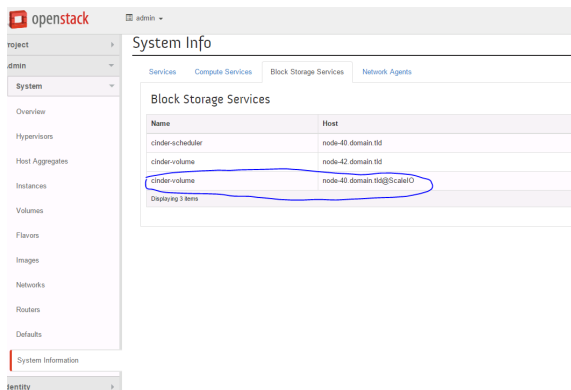
ScaleIO Cinder plugin OpenStack operations

Once the OpenStack Cluster is setup, we can setup ScaleIO Volumes. This is an example in how to attach a Volume to a running VM:

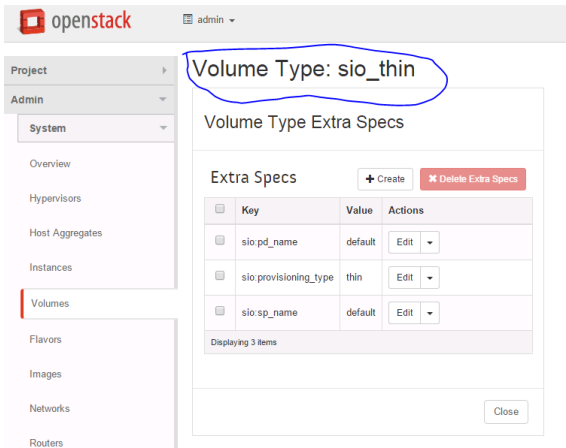
1. Login into the OpenStack Cluster:



2. Review the Block storage services by navigating: Admin -> System -> System Information section. You should see the ScaleIO Cinder Volume.



3. Review the System Volumes by navigating to: Admin -> System -> Volumes. You should see the ScaleIO Volume Type:



4. Create a new OpenStack Volume:

Create Volume

Volume Name *
vol_tg_test

Description
Description

Description: Volumes are block devices that can be attached to instances.

Volume Limits
Total Gigabytes (0 GB) 1,000 GB Available
Number of Volumes (0) 10 Available

Volume Source
No source, empty volume

Type
sio_thin

Size (GB) *
1

Availability Zone
Any Availability Zone

Cancel Create Volume

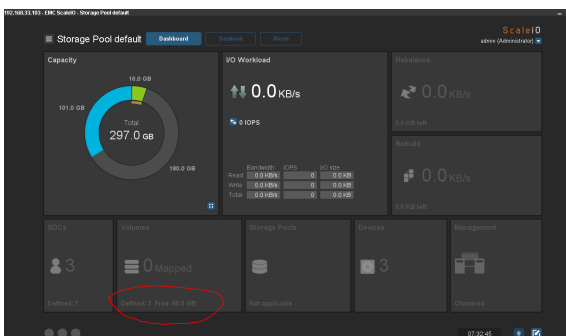
5. View the newly created Volume:

Volumes

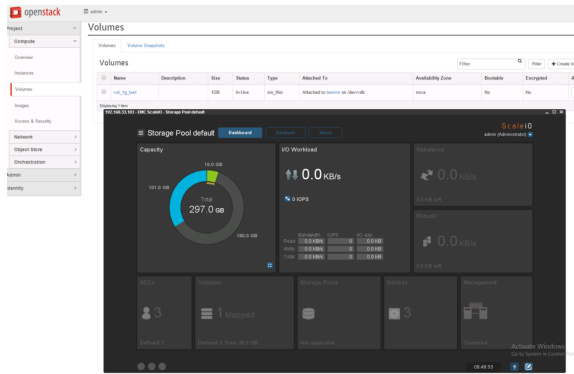
Name	Description	Size	Status	Type	Attached To	Availability Zone
vol_tg_test		1GB	Available	sio_thin		nova

Displaying 1 item

6. In the ScaleIO Control Panel, you will see that no Volumes have been mapped yet:



7. Once the Volume is attached to a VM, the ScaleIO UI will reflect the mapping:



Appendix

- [ScaleIO Web Site](#)
- [ScaleIO Documentation](#)
- [ScaleIO Download](#)
- [Fuel Enable Experimental Features](#)
- [Fuel Plugins Catalog](#)