

EMC STORAGE SOLUTION WITH OPENSTACK



Table of Contents

Introduction.....	4
Introduction	5
Business Challenge.....	6
EMC Storage for OpenStack.....	8
Key Components.....	10
OpenStack	11
EMC VNX	17
EMC XtremIO	18
EMC ScaleIO.....	19
Lab Overview	22
Lab Environment	23
Lab Credentials	28
Lab Scenario	29
Labs.....	30
Lab 1 - Using EMC VNX with Cinder Block Storage Service	33
Lab 1 - Using EMC VNX with OpenStack Cinder Block Storage	34
Lab 2 - Using EMC XtremIO with Cinder Block Storage Service	79
Lab 2 - Using EMC XtremIO with OpenStack Cinder Block Storage.....	80
Lab 3 - Using EMC ScaleIO with Cinder Block Storage Service	92
Lab 3 - Using EMC ScaleIO with OpenStack Cinder Block Storage	93
Optional Lab 4 - Live Migration of OpenStack Instance & Volume.....	133
Lab 4 - Live Migration	134

Optional Lab 5 - Using EMC ViPR with Cinder Block Storage Service	159
Lab 5 - Using EMC ViPR with Cinder	160
Optional Lab 6 - OpenStack Monitoring and Reporting.....	184
Lab 6 - OpenStack Monitoring and Reporting.....	185
Troubleshooting	196
Troubleshooting & Tips	197
Conclusion	205
Conclusion.....	206

Introduction

Introduction

Welcome to the EMC Storage Solutions for OpenStack Lab.

The Lab demonstrates how to prepare EMC Storage devices (VNX, ScaleIO, XIO) and install, configure and manage block storage in an OpenStack cloud environment. You will get to understand the various storage configuration touchpoints, how to manage the storage lifecycle from volume creations, snapshots, cloning etc and will further touch upon how to troubleshoot and isolate problems.

In this Lab you will review how a Cloud Provider called Code Nebulous is building and managing its cloud infrastructure with OpenStack and EMC storage.

This lab is based on recently announced **EMC Storage Solutions Reference Architecture for Managing EMC Storage Arrays with OpenStack Juno**.

This published **Reference Architecture** supports three major OpenStack distributions:

- Mirantis OpenStack
- Canonical Ubuntu Cloud Server
- RedHat Enterprise OpenStack Platform

Additionally, there are optional labs included:

- Introduction to OpenStack Monitoring & Reporting Solution Pack for EMC ViPR-SRM
- Using EMC ViPR with OpenStack Cinder

Business Challenge

1. Code Nebulous

Code Nebulous provides infrastructure as a service (IaaS) cloud-hosting solutions to 10 of the top 50 Internet sales portal sites in the world. They want to grow the business and penetrate into different geographical locations and verticals to compete with other providers.

Code Nebulous realizes other Cloud Providers today are increasingly driving a new model for the delivery of IT services. As IaaS and PaaS service use expands, cloud providers are faced with an urgency to enable cloud services by reduce complexity, drive automation, deliver secure multi-tenancy, improve TCO, and simplify end to end supply chain so they can get to the market as quickly as possible.

Enterprises are looking to Code Nebulous to provide rich cloud services via a self service portal or via rich APIs in a cost efficient way so they can drive their business transformation.

In advanced cloud delivery models, application owners are quickly changing their thinking from infrastructure-based deployments to service level-based deployments. Less consideration is given to the mechanics of the underlying infrastructure and more to performance levels that can be guaranteed by their business critical applications, regardless of the underlying components.

Code Nebulous need to be able to deliver the service levels Enterprise customers are requesting in the most automated and efficient manner possible. Offering an Infrastructure-as-a-Service (IaaS) platform with pre-engineered service-levels, and providing metering and chargeback reporting in an efficient and automated way, is an extremely important step toward realizing IT-as-a-Service (ITaaS) efficiency.



2. Cloud Challenges

Code Nebulous requires a scalable, tiered, and highly available infrastructure on which to deploy application workloads and provide services in an efficient way. There are several new technologies available to assist in creating a solution but in this new world of OpenSource where both community and vendors eco-system support is growing it opens the door for Code Nebulous to build cloud services in an efficient way. The customer needs to know how to best use these technologies to support service-level agreements, integrate proprietary systems, and minimize cost.

Many Enterprise organizations are under pressure to provide enterprise-quality service levels on demand without paying enterprise prices. As a result, Code Nebulous departments need to create cost-effective alternatives to proprietary cloud services; alternatives that do not compromise enterprise features such as data protection, disaster recovery (DR), and guaranteed service levels.

As IT organizations implement an OpenStack cloud, they must consider the following factors:

- The infrastructure must be deployed quickly so that business value can be recognized quickly.
- The OpenStack cloud infrastructure and operations must be designed to reduce licensing costs.
- Risk of downtime must be controlled through disciplined change control and careful management of component compatibility.
- Support agreements must be established for all elements of the solution.

EMC Storage for OpenStack

The purpose of this solution is to demonstrate integration of OpenStack software with EMC storage systems and validate the environment for performance, scalability, and functionality. Customers will realize:

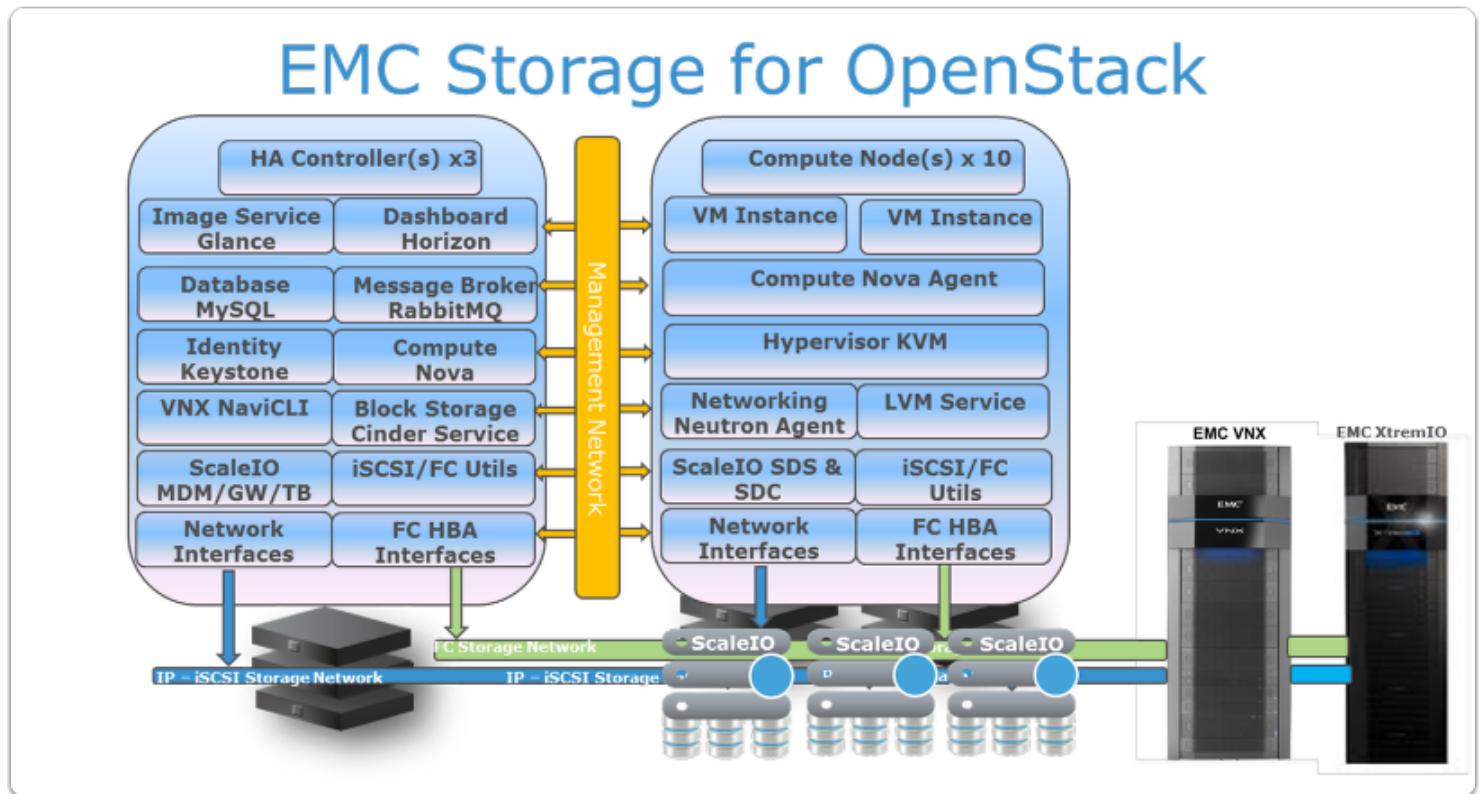
- Decreased costs associated with environment scalability.
- More choices in supported hardware.
- Greater flexibility in cloud migration.
- Lower operational and maintenance costs.

Solution

Code Nebulous team have selected EMC VNX, XtremIO and ScaleIO as their storage technologies to integrate with OpenStack powered cloud.

This solution demonstrates how to use EMC storage arrays and OpenStack Cinder drivers to provide the storage resources for a robust OpenStack environment. This solution incorporates the following components:

- OpenStack Juno release
- EMC VNX - The EMC VNX family delivers a choice of systems ranging from affordable entry-level solutions to high-performance, petabyte-capacity configurations servicing the most demanding application requirements
- EMC ScaleIO - EMC ScaleIO is a software-only server-based storage area network (SAN) that converges storage and compute resources to form a single-layer, enterprise-grade storage product.
- EMC XtremIO - XtremIO is a scale-out clustered design that grows capacity and performance linearly to meet any requirement
- Cinder drivers for EMC VNX and XtremIO (part of Juno release)
- Cinder drivers for EMC ScaleIO (available from EMC Support)



Key Components

OpenStack

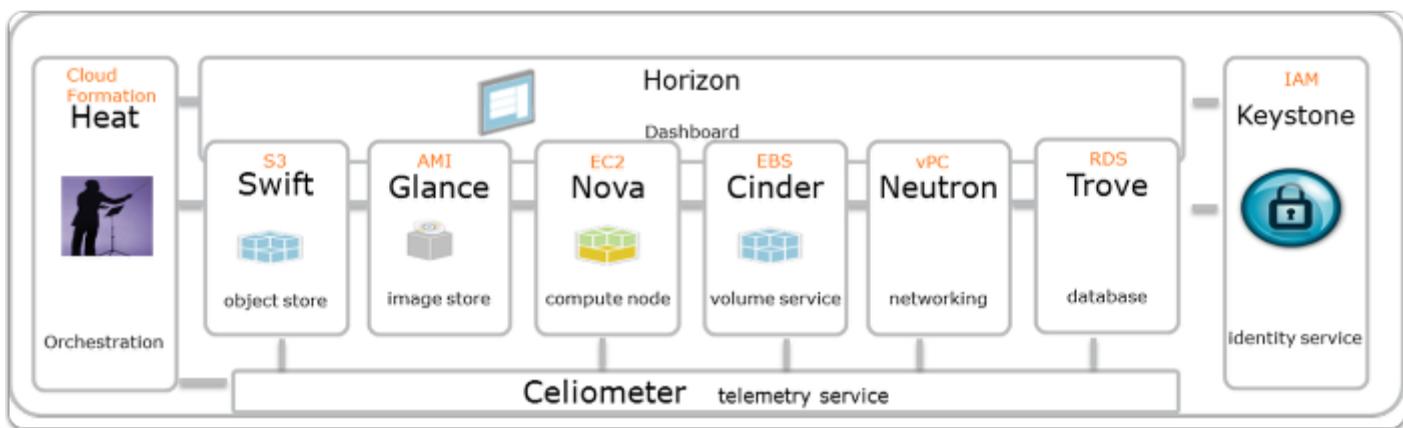
OpenStack is the open source software for creating private and public clouds.

OpenStack software controls large pools of compute, storage, and networking resources throughout a datacenter, managed through a dashboard or via the OpenStack API. OpenStack works with popular enterprise and open source technologies making it ideal for heterogeneous infrastructure.

1. OpenStack Architecture

OpenStack is a cloud operating system that controls large pools of compute, storage, and networking resources throughout a datacenter, all managed through a dashboard that gives administrators control while empowering their users to provision resources through a web interface.

In addition to the original code from the OpenStack Foundation (<https://www.openstack.org/software/>) , there is a number of different OpenStack distributions available. For our lab we have used the Mirantis OpenStack: <https://software.mirantis.com/>



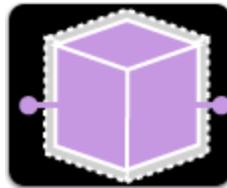
2. OpenStack Compute

OpenStack Compute enables enterprises and service providers to offer on-demand computing resources, by provisioning and managing large networks of virtual machines. Compute resources are accessible via APIs for developers building cloud applications and via web interfaces for administrators and users. The compute architecture is designed to scale horizontally on standard hardware, enabling the cloud economics companies have come to expect.



3. OpenStack Networking

OpenStack Networking is a pluggable, scalable and API-driven system for managing networks and IP addresses. Like other aspects of the cloud operating system, it can be used by administrators and users to increase the value of existing datacenter assets. OpenStack Networking ensures the network will not be the bottleneck or limiting factor in a cloud deployment and gives users real self service, even over their network configurations.



4. OpenStack Storage

In addition to traditional enterprise-class storage technology, many organizations now have a variety of storage needs with varying performance and price requirements. OpenStack has support for both Object Storage and Block Storage, with many deployment options for each depending on the use case. It provides a fully distributed, API-accessible storage platform that can be integrated directly into applications

Object Storage is ideal for cost effective, scale-out storage. Block Storage allows block devices to be exposed and connected to compute instances for expanded storage, better performance and integration with enterprise storage platforms.

In this lab your primary focus will be on the OpenStack Block Storage backends provided by EMC VNX, XtremIO and ScaleIO. An optional lab will focus on EMC ViPR integration with Cinder.

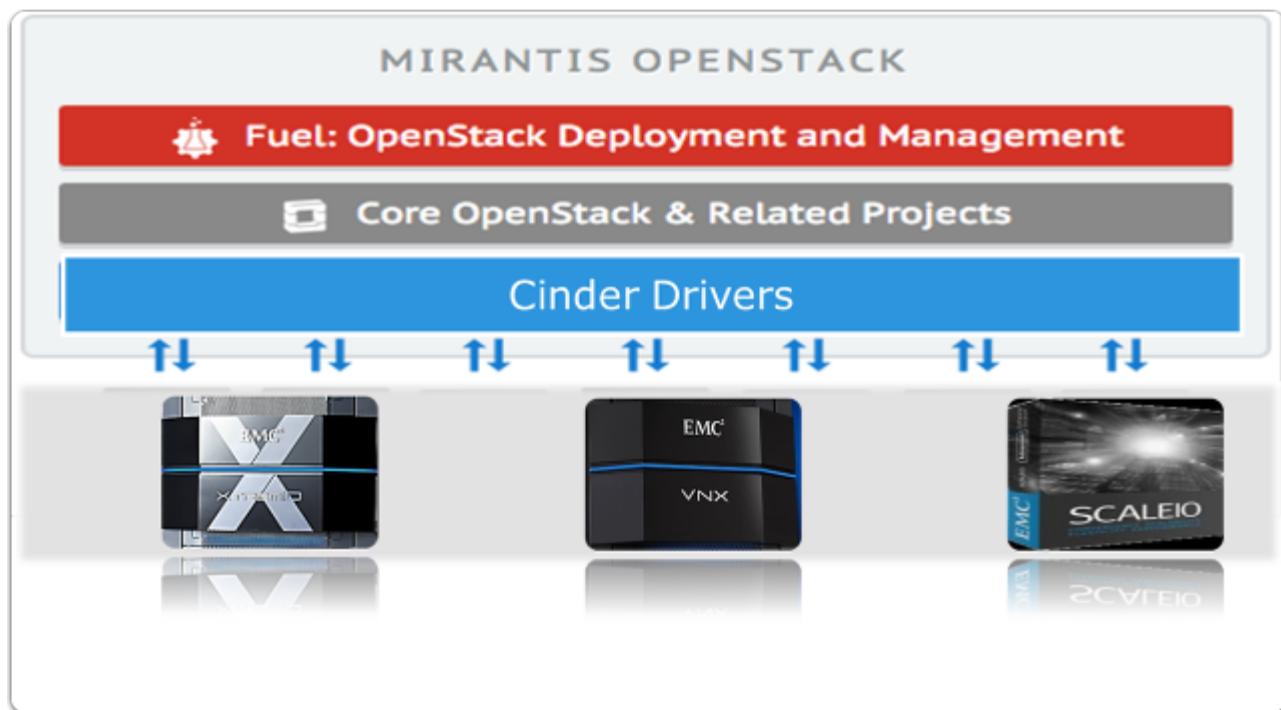


5. OpenStack Cinder Block Storage Service

OpenStack Cinder provides persistent block level storage devices for use with OpenStack compute instances.

The block storage system manages the creation, attaching and detaching of the block devices to servers. Block storage volumes are fully integrated into OpenStack Compute and the Dashboard allowing for cloud users to manage their own storage needs.

In this lab you will learn how EMC storage systems integrate with OpenStack Cinder Block Storage service.

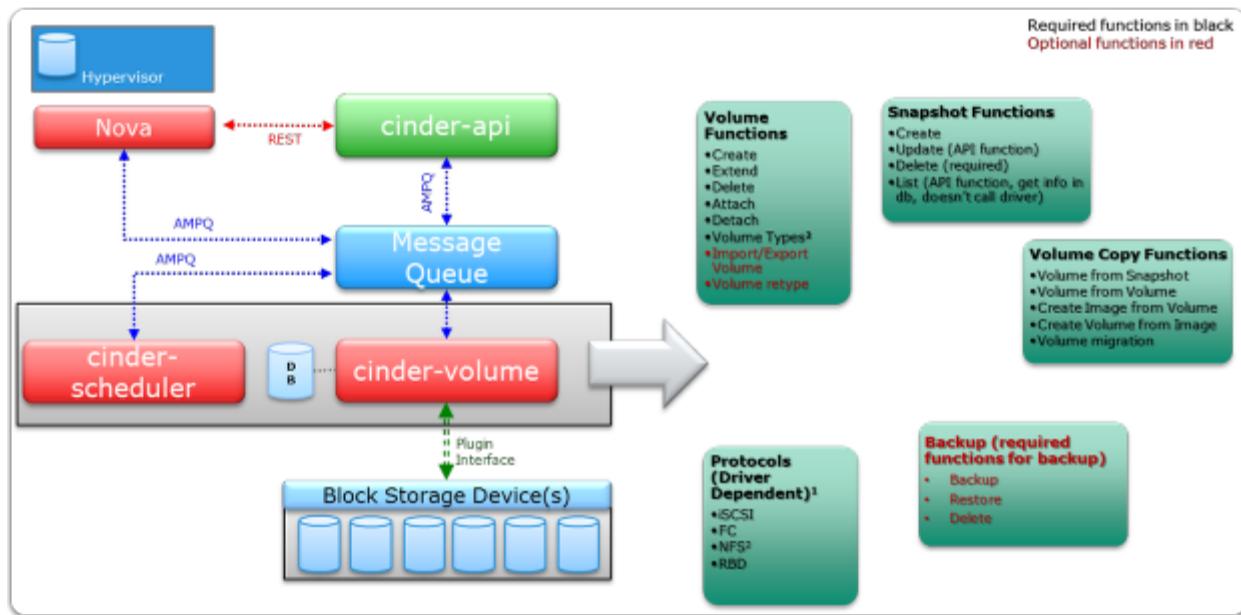


6. Cinder Service Plugins (Drivers)

In addition to using simple Linux server storage, OpenStack Cinder has unified storage support for numerous storage platforms including EMC VNX, XtremIO and ScaleIO.

EMC developed plugins for Cinder enable block storage for performance sensitive scenarios, such as database storage, expandable file systems, or providing a server with access to raw block level storage.

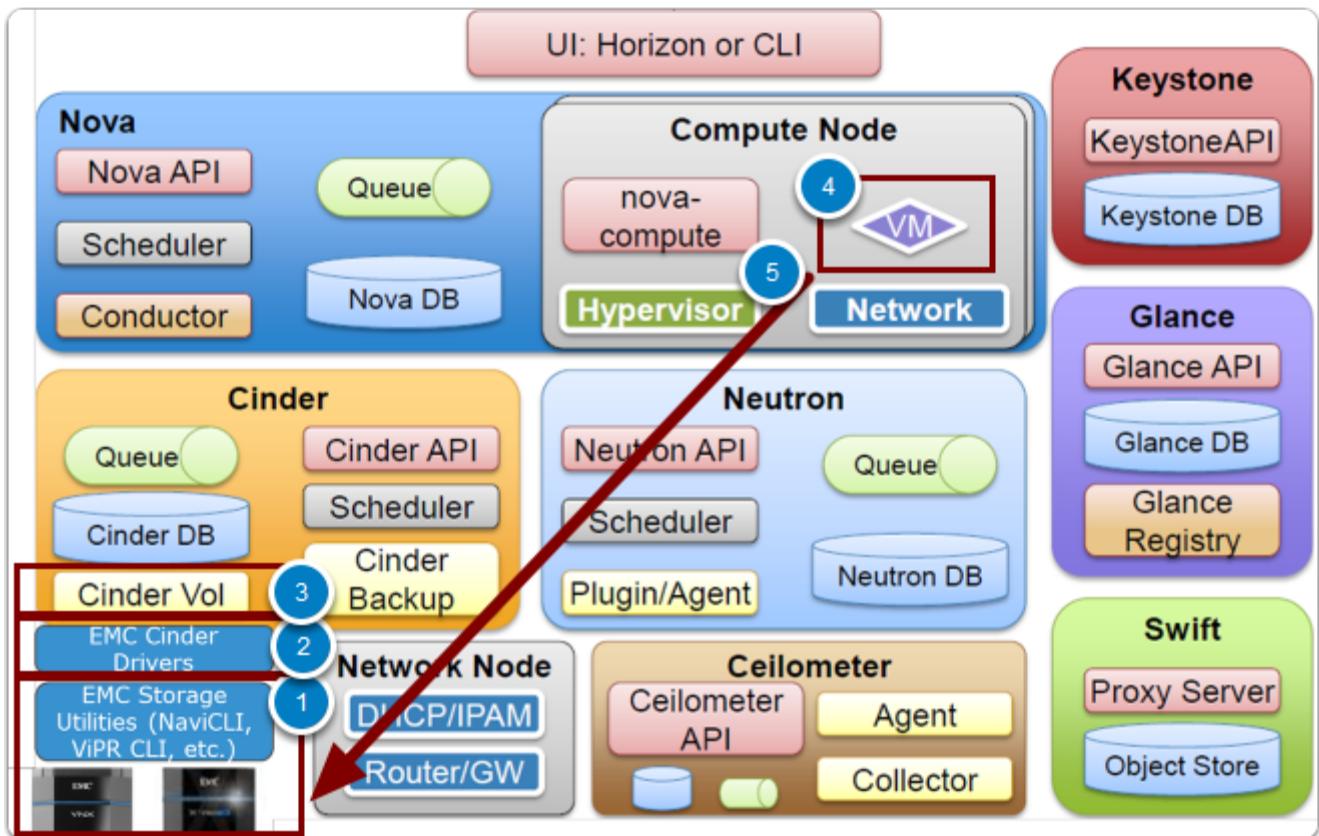
These plugins (drivers) support the entire volume lifecycle, and snapshot management, providing powerful functionality for storing and backing up data.



7. OpenStack Volumes

While the OpenStack process for launching a VM instance and attaching the storage volume to it is fairly complex, in your labs you will focus specifically on the following:

1. Reviewing the EMC Storage settings required for OpenStack integration
2. Installing and configuring the EMC Storage API's and utilities, such as NaviCLI for VNX, ViPR CLI, etc.
3. Installing and configuring the Cinder drivers for EMC Storage systems (VNX, XtremIO, ScaleIO and ViPR)
4. Creating OpenStack Volume Types and Volumes
5. Attaching the newly created Volumes to the VM Instance



EMC VNX

EMC VNX is a high-performing unified storage with unsurpassed simplicity and efficiency, optimized for virtual applications. With the VNX Series, you'll achieve new levels of performance, protection, compliance, and ease of management.

1. VNX Automation and Data Protection

EMC VNX fully automated storage tiering (FAST) simultaneously delivers both the highest performance and the lowest total cost of ownership (TCO).

VNX delivers the highest levels of application availability and data protection with continuous operations and any-point-in-time recovery.



2. VNX Driver for Cinder

EMC VNX direct driver for OpenStack Cinder supports both iSCSI and FC protocol. VNX iSCSI direct driver and VNX FC direct driver are separately based on the ISCSIDriver and FCDriver defined in Block Storage.

VNX Direct driver was made available in the OpenStack Juno release and perform the volume operations by executing Navisphere CLI (NaviSecCLI) which is a command line interface used for management, diagnostics and reporting functions for VNX.

EMC XtremIO

EMC XtremIO is an all-flash purpose-built platform for production, dev/test, BI/analytics, training, rampant cloning, user self-service, predictable performance with scale, lower TCO.

XtremIO in the Cloud

EMC XtremIO provides continuous availability and access for mission-critical, performance-sensitive apps in an app landscape. It consistently delivers reliable, predictable, performance that meets all SLAs for advanced applications while reducing storage and hardware/processor requirements to lower your app storage TCO.



XtremIO Cinder Driver

High performance XtremIO All Flash Array (AFA) offers Block Storage services to OpenStack.

In this lab you will learn how using this driver OpenStack Cinder Block Storage service can connect to an XtremIO Storage cluster.

EMC ScaleIO

EMC ScaleIO: Software-only server-based storage area network (SAN) that converges storage and compute resources to form a single-layer, enterprise-grade storage product. ScaleIO storage is elastic and delivers linearly scalable performance. Its scale-out server SAN architecture can grow from a few to thousands of servers.

1. ScaleIO in the Cloud

Convergence

ScaleIO uses servers' direct-attached storage (DAS) and aggregates all disks into a global, shared, block storage. ScaleIO features single-layer compute and storage architecture without requiring additional hardware or cooling/power/space

Scalability and Performance

Breaking traditional barriers of storage scalability, ScaleIO scales out to hundreds and thousands of nodes and multiple petabytes of storage. The parallel architecture and distributed volume layout delivers a massively parallel system that deliver I/O operations through a distributed system. As a result, performance can scale linearly with the number of application servers and disks, leveraging fast parallel rebuild and rebalance without interruption to I/O. ScaleIO has been carefully designed and implemented with ScaleIO software components so as to consume minimal computing resources.

Elasticity and Resilience

With ScaleIO, any administrator can add, move, or remove servers and capacity on demand during I/O operations. The software responds automatically to any infrastructure change and rebalances data accordingly across the grid nondisruptively. ScaleIO can add capacity on demand, without capacity planning or data migration and grow in small or large increments and pay as you grow, running on any server and with any storage media

Platform Agnosticism

ScaleIO natively supports all leading Linux distributions and hypervisors. It works agnostically with any solid-state drive (SSD) or hard disk drive (HDD) regardless of type, model, or speed.



2. ScaleIO Components

ScaleIO Data Client (SDC)

- Acts as Block Device Driver
- Exposes volumes to applications
- Service must run to provide access to volumes
- Over TCP/IP

ScaleIO Data Service (SDS)

- Abstracts storage media
- Contributes to storage pools
- Performs I/O operations

ScaleIO Metadata Manager (MDM)

- Not located in the data path
- Provides Monitoring and Configuration management
- Holds cluster-wide component mapping

3. ScaleIO Cinder Driver

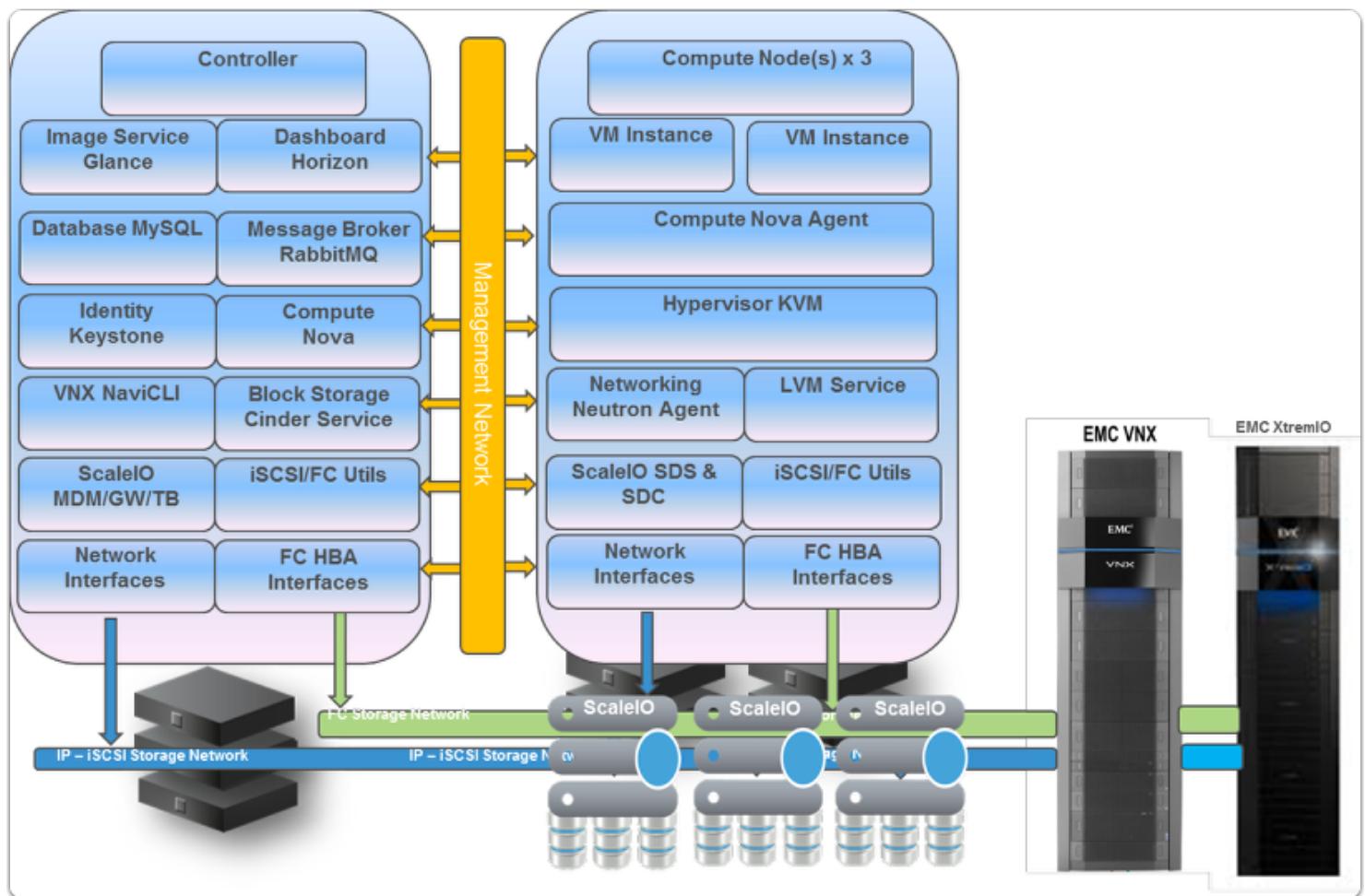
The ScaleIO elastic storage solution includes a Cinder driver, which interfaces between ScaleIO and OpenStack, and presents volumes to OpenStack as block devices which are available for block storage. It also includes an OpenStack Nova driver, for handling compute and instance volume related operations. The ScaleIO driver executes the volume operations by communicating with the backend ScaleIO MDM through the ScaleIO REST Gateway.

Lab Overview

Lab Environment

1. Lab Architecture

This lab's environment has been designed to approximate a typical OpenStack Cloud deployment with a Controller node and several Compute nodes.



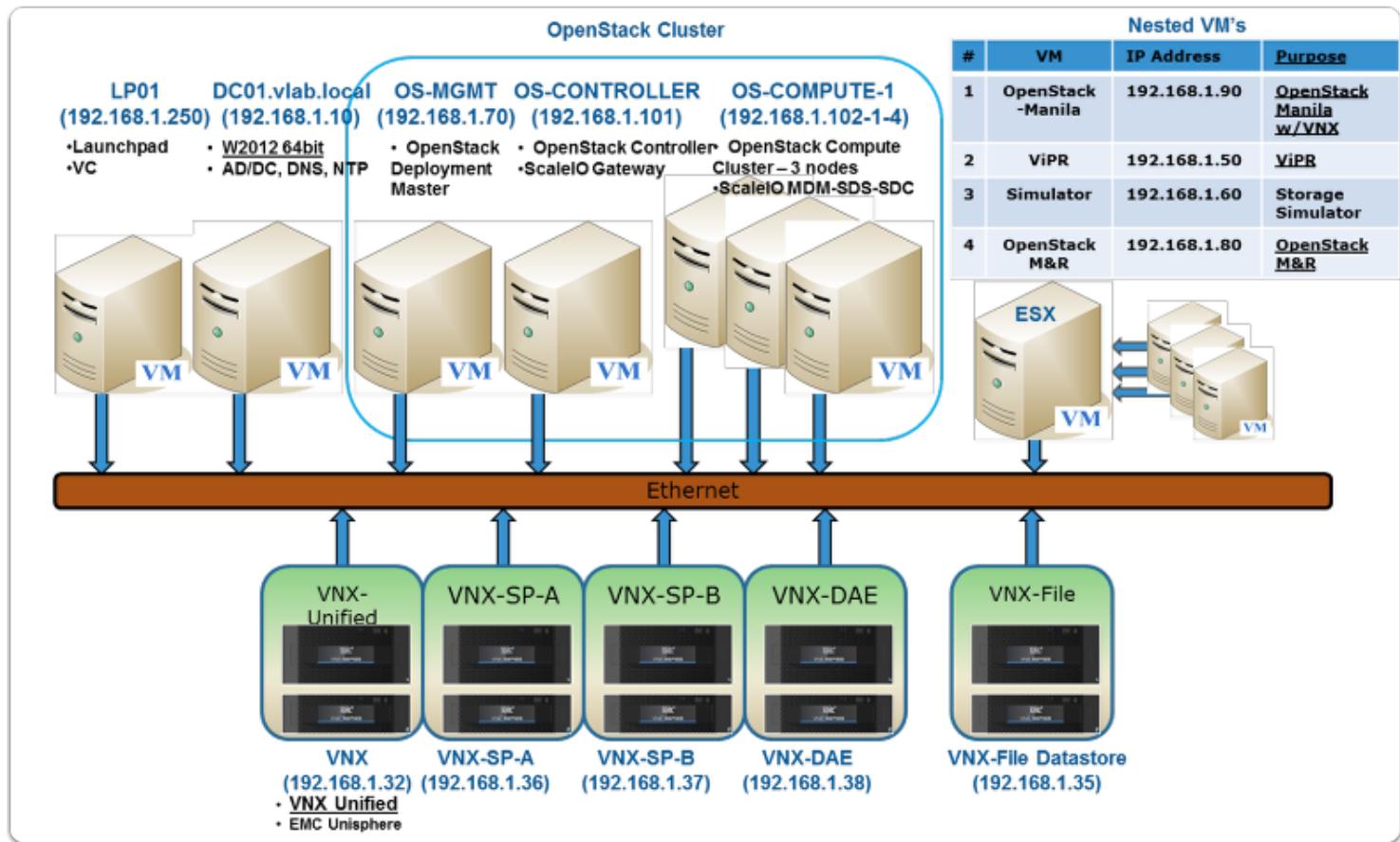
2. Virtualized Lab Environment

The environment for this lab includes the following:

- Domain Controller
- Launchpad VM that also includes VMware vCenter
- OpenStack Cluster with Management Node, Controller Node and three Compute Nodes

- VMware ESX server running "nested" VM's
- Standalone VNX File simulator acting as vSphere Datastore
- VNX Unified simulator (block & file storage)

Note: While this lab does not include XtremIO simulator, the XtremIO Cinder driver configuration is covered in Lab 2.



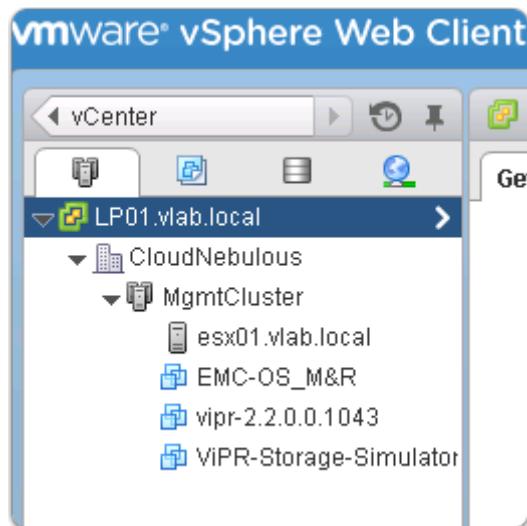
3. Nested Virtual Machines

This lab uses nested VMs for some of the lab scenarios.

Additionally, some storage simulators and other infrastructure-related VM's are running on the same ESX host.

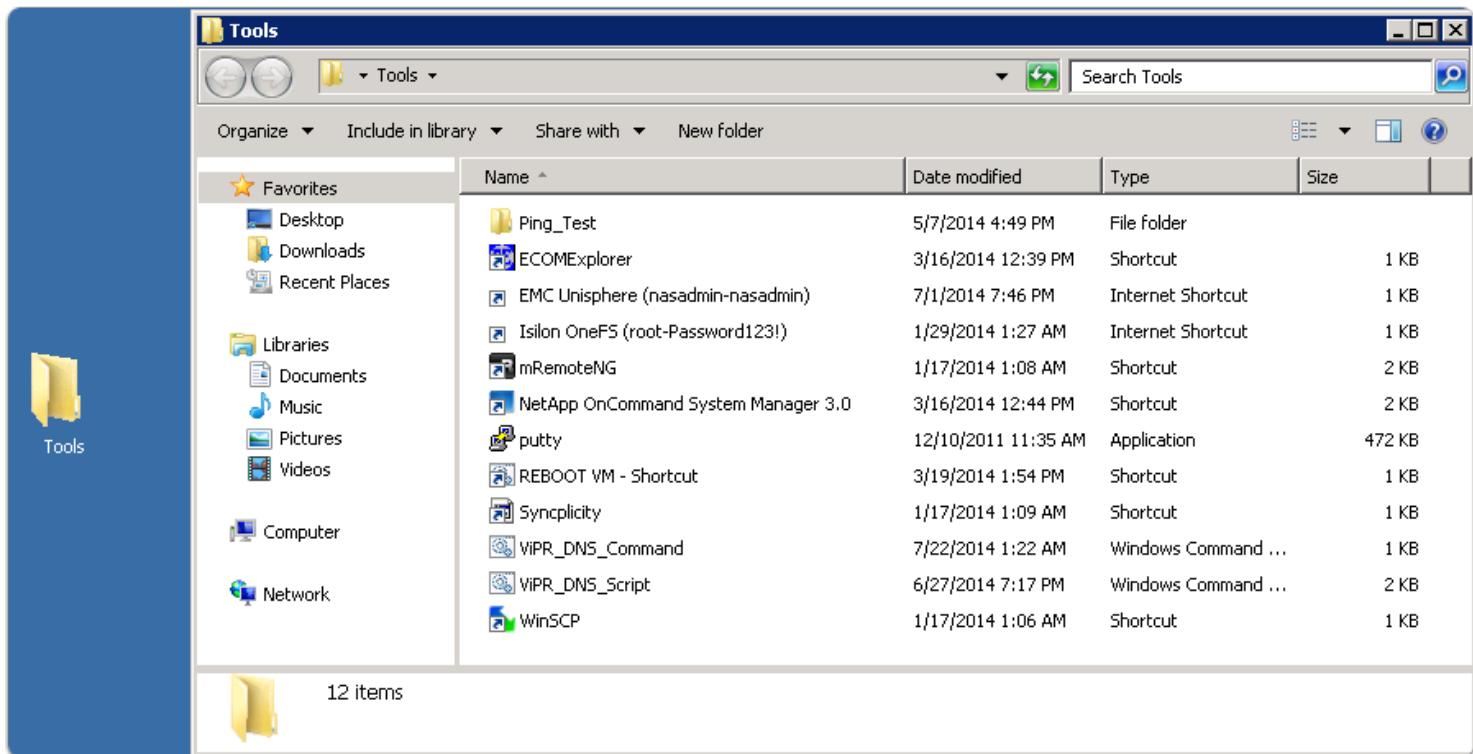
You will need to power on some of these nested VM's using automated scripts as instructed during the lab.

!!! Note: Please DO NOT power up or down any of the VM's until instructed!



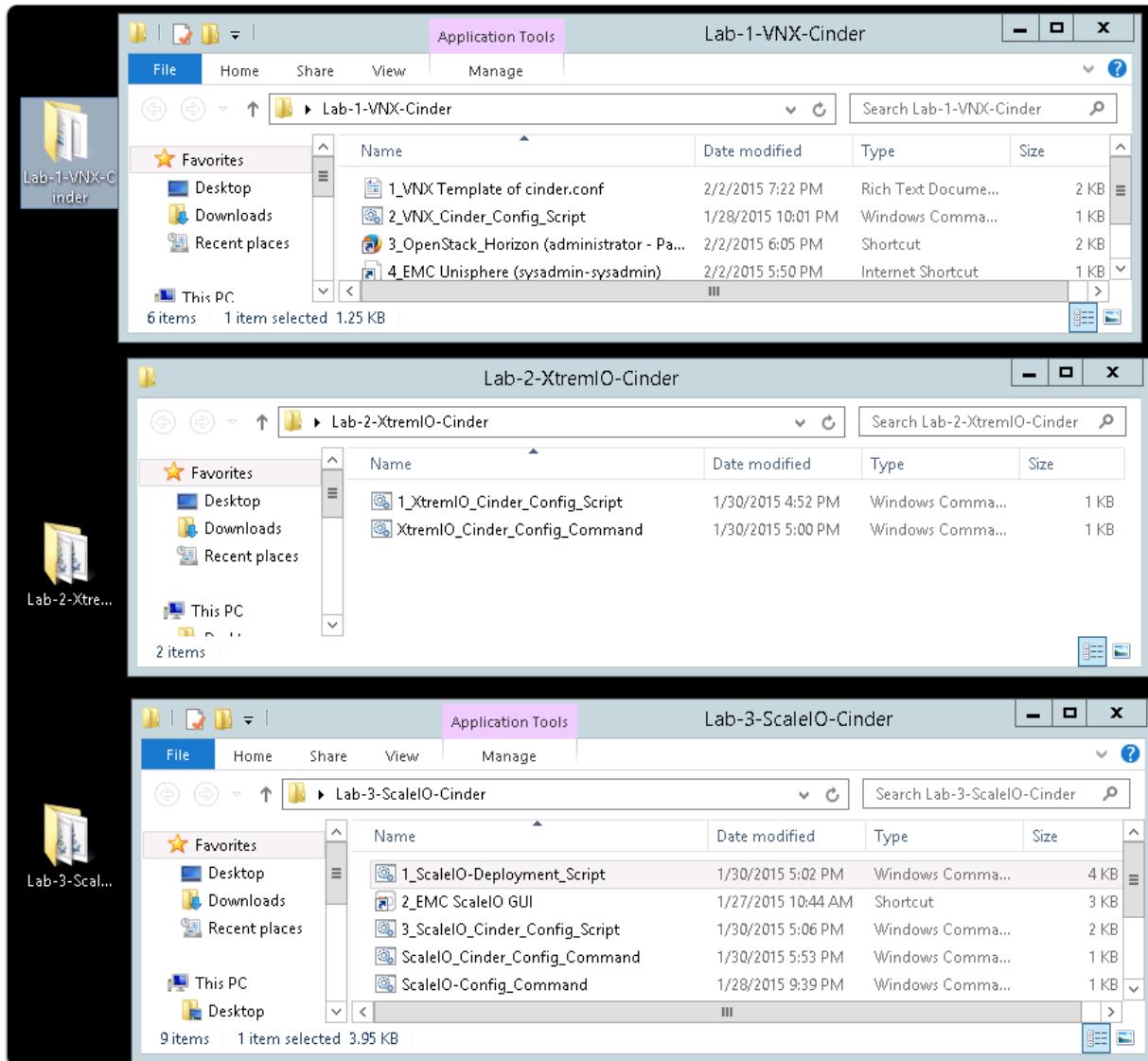
4. Lab Tools

To assist you in executing this lab, a miscellaneous number of tools are located in the Tools folder on the Launchpad Desktop, i.e PutTY terminal client, WinSCP file transfer utility, etc.:



5. Accessing Labs

Separate folders have been created on the Launchpad Desktop with all relevant scripts and shortcuts for accessing Lab scenarios.



Lab Credentials

- Windows credentials: VLAB\Administrator - Password123!
- Linux SSH / CLI credentials: root - Password123!
- OpenStack Horizon login: Administrator - Password123!
- Unisphere login: sysadmin - sysadmin
- ScaleIO UI login: admin - Password123!
- ViPR Controller login: root - Password123!

Lab Scenario

Stephan is the Cloud Administrator for CodeNebulous Cloud Systems.

One of Stephan's tasks is to sit down with his Tenants and design the storage layout based on the IO profile provided to him. This is time consuming in planning and design. Stephan is also tasked in provisioning storage for all his Tenants which is time consuming.

Stephan has recently overseen the deployment of the OpenStack Cloud platform in one of Code Nebulous Data Centers. Stephan as the IT administrator has been tasked with the following tasks so that he can hand over the environment to his own Tenants so that they can now provision their own storage in a secure and multi tenancy environment:

1. Review the Cinder driver capabilities for several different EMC Storage systems, including extra-specs
2. Review and become familiar the different Cinder drivers configuration steps
3. Follow the configuration steps and implement Cinder integration with EMC Storage backends
4. Identify the Storage Pools pre-created on the EMC Storage systems by Storage Admin
5. Create storage volume types for OpenStack Volumes and associate the Volume Types with Storage Pools
6. Practice creating new volumes for different Volume Types
7. Ensure that new Volumes are created using EMC Storage Element Managers, i.e. Unisphere, ScaleIO and ViPR UI's
8. Verify that new Volumes can be attached to an OpenStack VM instance
9. Ensure that new Storage Volume can be mounted and formatted by logging into VM instance using SSH
10. Practice performing volume life-cycle management tasks (extend, dedup, compress, etc.)

The following Labs will guide Stephan through each step.

Labs

The following is a high level overview of the labs. Each of these labs can be executed independently. You are not required to complete a previous lab before starting another.

Labs and Use Cases

This Hand On Lab covers the following Use Cases:

!!! Note: Each of these labs can be completed independently of one another, unless specifically noted!

Lab 1: Using EMC VNX with OpenStack Cinder Block Storage

This lab takes approximately 15 min to complete. In this lab you, as a Cloud Administrator, will learn how to:

- Install required pre-requisite utilities for VNX Cinder driver
- Configure Cinder to use VNX direct driver
- Create different VNX Block Volume Types and Volumes
- Attach Volumes to OpenStack instances and make them available for data storage
- Manipulate the Volumes through their life-cycle



Lab 2: Using EMC XtremIO with OpenStack Cinder Block Storage

This lab takes approximately 10 min to complete. In this lab you, as a Cloud Administrator, will learn how to:

- Configure Cinder to use XtremIO direct driver
- Create different XtremIO Block Volume Types and review them in OpenStack Horizon UI.



Lab 3: Using EMC ScaleIO with OpenStack Cinder Block Storage

This lab takes approximately 15 min to complete. In this lab you, as a Cloud Administrator, will learn how to:

- Install ScaleIO on Controller and Compute nodes to enable sharing of local disks
- Configure Cinder to use ScaleIO driver
- Create different ScaleIO Block Volume Types and Volumes
- Attach Volumes to OpenStack instances and make them available for data storage
- Manipulate the Volumes through their life-cycle



Lab 4: Optional - Live Migration of OpenStack VM instances & Storage Volumes

This lab is optional. It takes approximately 10 min to complete and requires that you have previously completed the following labs:

- Lab 1 - Using VNX with Cinder Block Storage
- Lab 3 - Using ScaleIO with Cinder Block Storage

In this lab you as a Cloud Administrator will learn how to migrate a "live" active OpenStack virtual machine instance and its associated storage volume. In the course of the Agile development process, Cloud Admins will encounter a situation when the workload and its associated data needs to be moved from a low-cost distributed storage to a high-performance high-availability external storage array.



Lab 5: Optional - Using EMC ViPR with OpenStack Cinder Block Storage

This lab takes approximately 10 min to complete. In this lab you, as a Cloud Administrator, will learn how to:

- Configure Cinder to use ViPR Controller driver

- Create different ViPR Block Volume Types and Volumes
- Attach Volumes to OpenStack instances and make them available for data storage
- Manipulate the Volumes through their life-cycle



Lab 6: Optional - OpenStack Monitoring & Reporting

This lab is optional! It takes approximately 10 min to complete. In this lab you, as a Cloud Administrator, will learn how to:

- Review the OpenStack Capacity and Chargeback reports for Tenants and Projects
- Customize the reports and share them with the Tenants



Lab 1 - Using EMC VNX with Cinder Block Storage Service

Lab 1 - Using EMC VNX with OpenStack Cinder Block Storage

In this lab you, as a Cloud Administrator will become familiar the VNX Cinder drivers configuration steps and implement Cinder integration with EMC VNX.

You will also create Cinder Volume Types associate with the VNX storage pools, create new volumes for different Volume Types and attach them to an OpenStack VM instance.

1. VNX Components

While EMC VNX supports both Block and File storage, this lab focuses on VNX Block storage backend for OpenStack Cinder Block storage service. You will also us EMC Unisphere to verify the block volumes created by OpenStack in the VNX Block storage pool.

2. VNX Cinder Driver Supported operations

As of Juno release of OpenStack cloud software (November 2014), VNX Cinder Driver is included in the trunk code and is distributed with Juno software. VNX Cinder driver supported operations are available at: <http://docs.openstack.org/trunk/config-reference/content/emc-vnx-direct-driver.html>

- Create, delete, attach, and detach volumes.
- Create, list, and delete volume snapshots.
- Create a volume from a snapshot.
- Copy an image to a volume.
- Clone a volume.
- Extend a volume.
- Migrate a volume.
- Retype a volume.
- Get volume statistics.
- Create and delete consistency groups.
- Create, list, and delete consistency group snapshots.

3. Lab Scenario

In this lab you as a Cloud Admin will:

- Review the Cinder driver capabilities for EMC VNX, including extra-specs
- Review and become familiar with the VNX direct Cinder drivers configuration steps
- Follow the configuration steps and implement Cinder integration with EMC VNX Storage backend
- Identify the Storage Pools pre-created on the EMC VNX using Unisphere created by Storage Admin

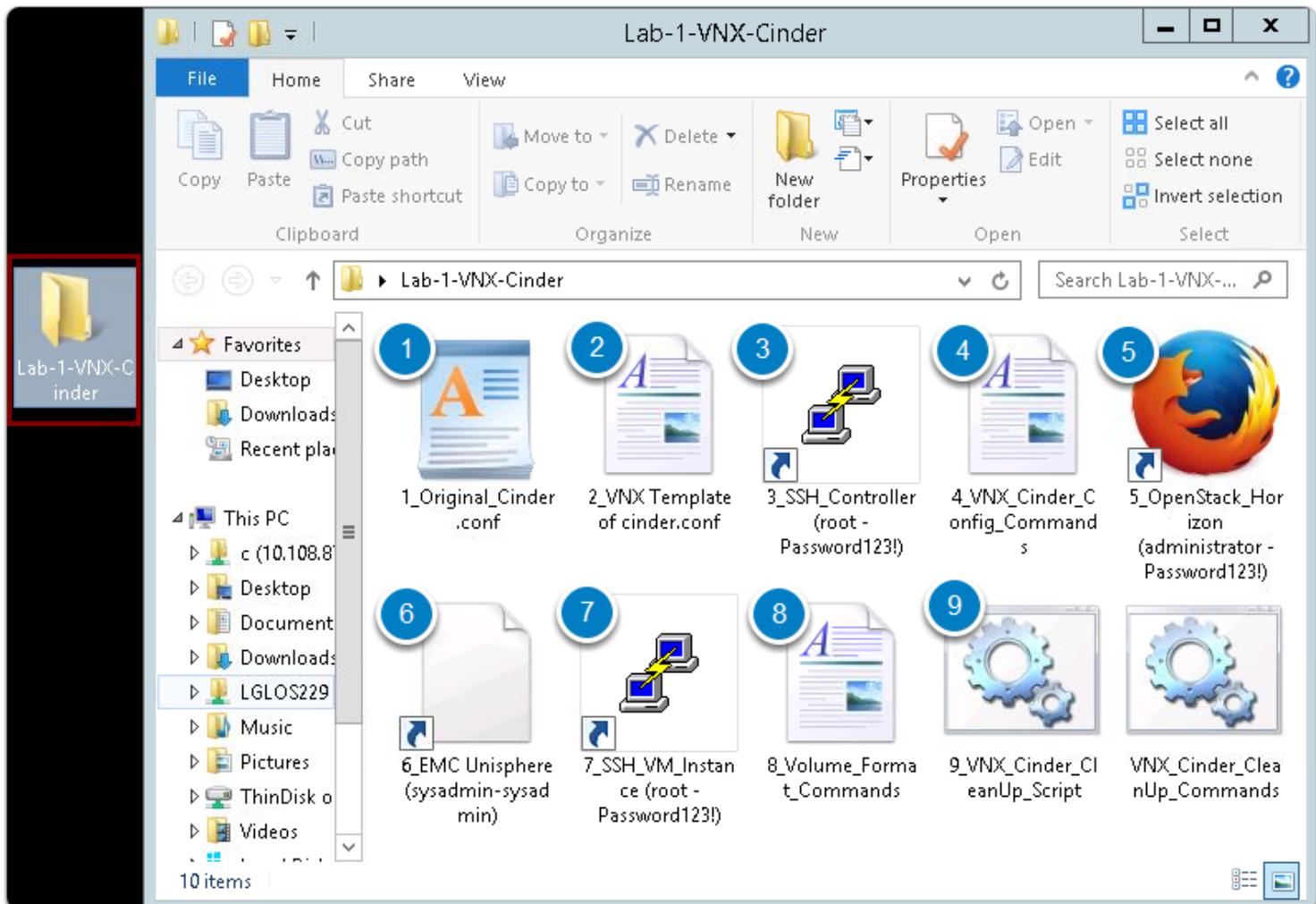
- Create storage volume types for OpenStack Volumes and associate the Volume Types with Storage Pools
- Practice creating new volumes for different Volume Types
- Ensure that new Volumes are created using EMC Unisphere
- Verify that new Volumes can be attached to a OpenStack VM instance
- Ensure that new Storage Volume can be mounted and formatted by logging into VM instance using SSH
- Practice performing volume life-cycle management tasks (extend, dedup, compress, etc.)

4. Lab-1 Folder

Please open and review **Lab-1-VNX-Cinder** folder

Pay attention to the Numbers in the name of the shortcut - for your convenience you will be using these shortcuts in order:

1. Original Cinder Config File
2. VNX Template of Cinder.Conf
3. SSH Utility to access the OpenStack Controller terminal
4. VNX Cinder Configuration Commands, etc.
5. OpenStack Horizon UI
6. EMC Unisphere to view the VNX Storage Pool and created volumes
7. SSH Utility to access OpenStack VM Instance terminal
8. Volume Format Commands
9. Cleanup script to complete this lab and prepare for the next one.



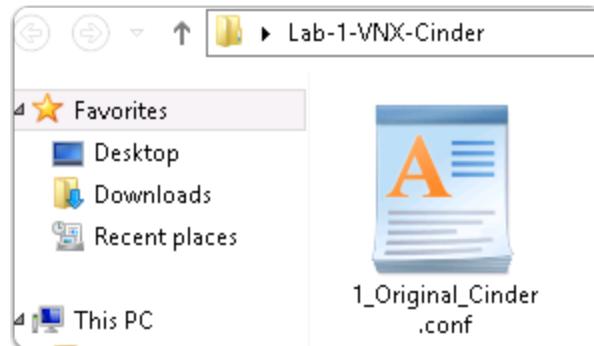
4.1 VNX Cinder Driver Configuration Steps

To implement the VNX backend for Cinder block service, you will now

- Install VNX Command line API - NaviCLI
- Modify Cinder configuration file /etc/cinder/cinder.conf
- Restart Cinder API and Cinder Volume services
- Create VNX Volume types in Cinder

4.2 Cinder.conf file

For your reference, the Cinder.conf configuration file has been provided which contains all necessary information for VNX storage backend. Please open the 1-Original_Cinder.conf and review it:

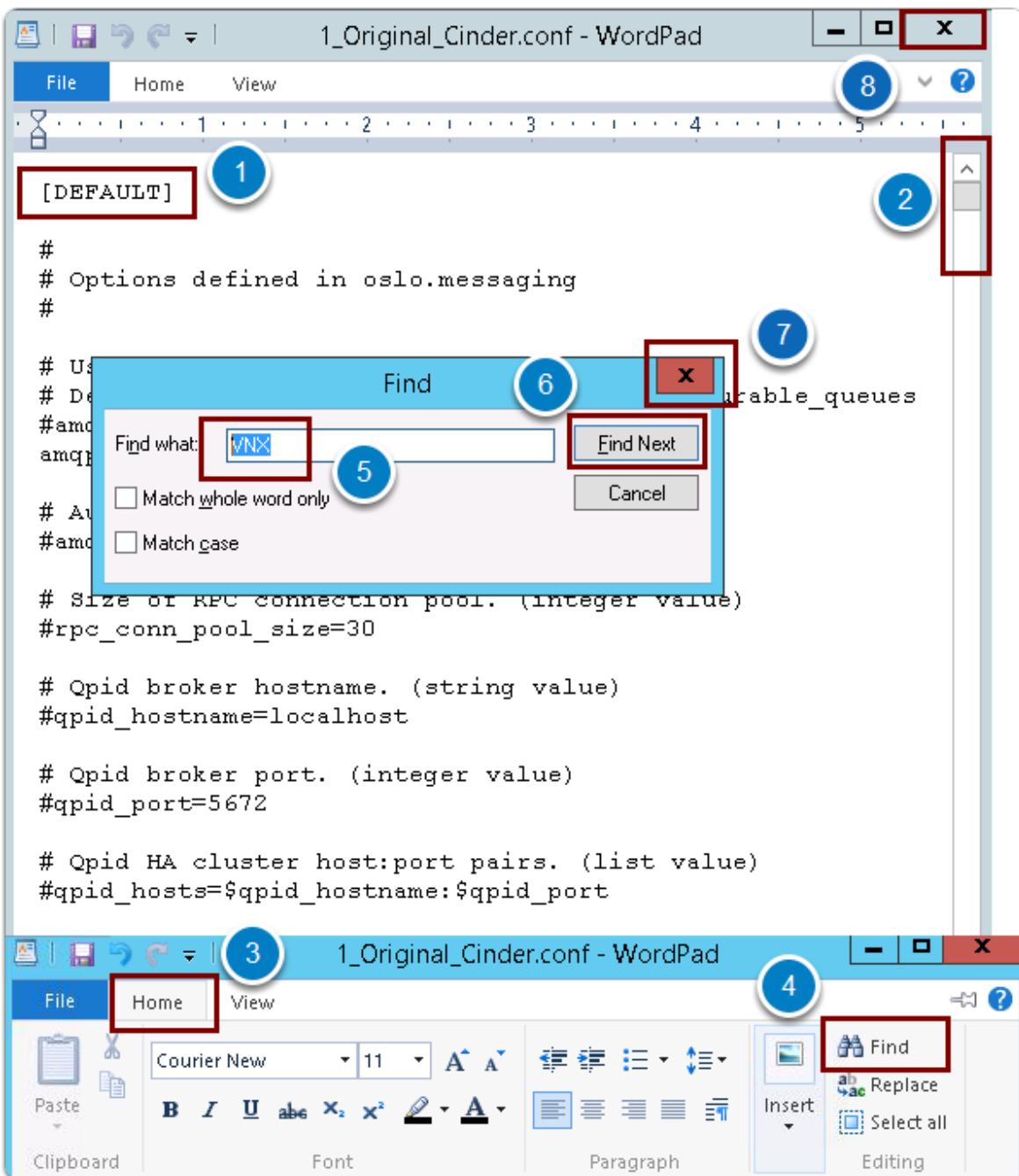


4.3 Cinder.conf Default Section

1. Please review the content of the Cinder.conf file and its [DEFAULT] section
2. Scroll down as necessary to find the configuration settings for VNX Storage
3. Click **Home**
4. Then click **Find** menu option
5. Input **VNX**
6. Click **Find Next** as necessary
7. Then close search window
8. Please close **1_Original_Cinder.conf** file after reviewing

As you can see the necessary configuration entries for Cinder integration with VNX are available out of the box.

All necessary configuration settings have been summarized in a single section of a separate sample file.



4.4 Sample Cinder.conf file

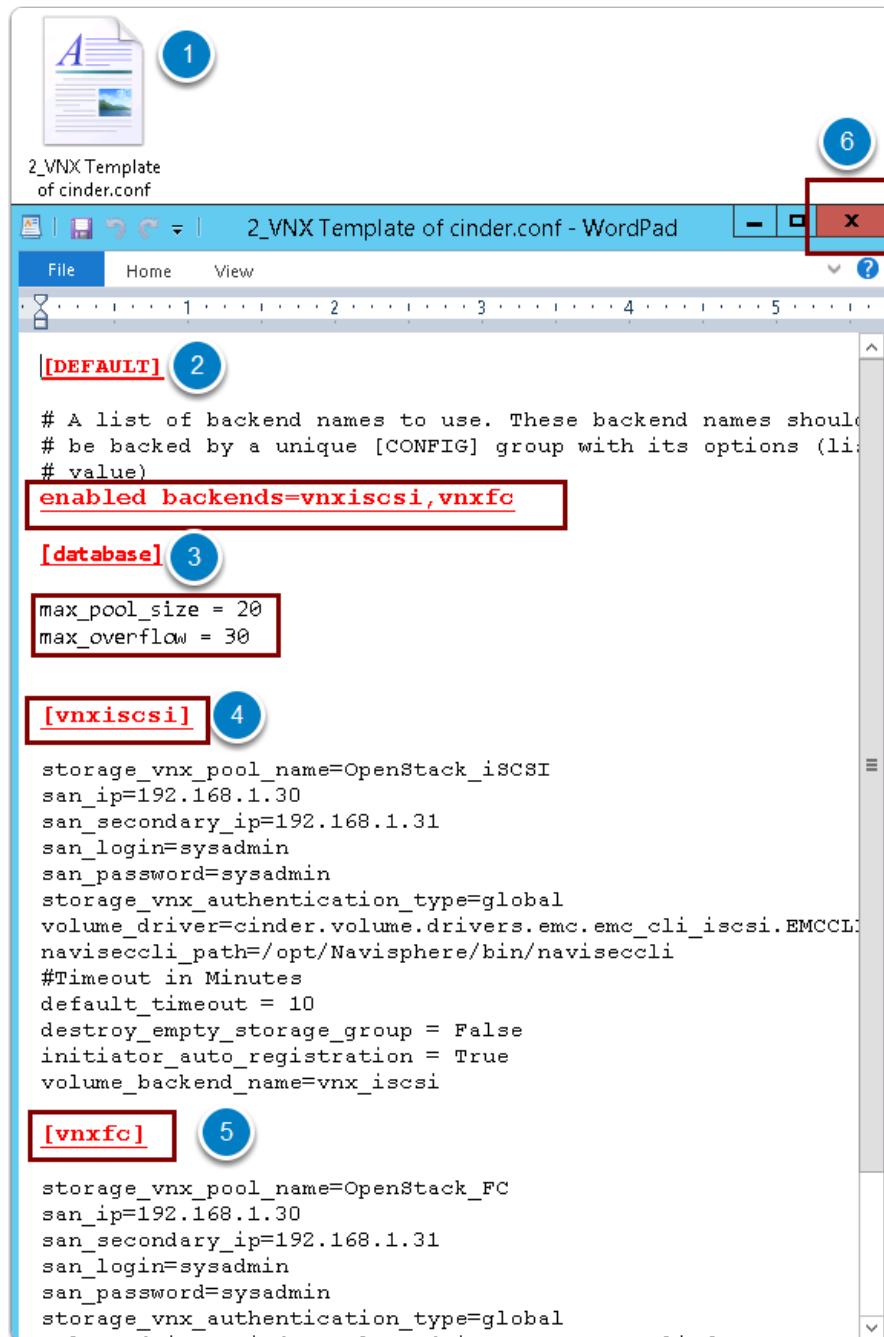
For your convenience all the changes required for configuration of the VNX backend for Cinder block service have been summarized in a sample Cinder.conf file.

The relevant sections are highlighted in Red.

1. Please open a file called **2_VNX Template of Cinder.conf**
2. Review the settings in the **[DEFAULT]** section for enabled backends.
3. Review the settings in the **[database]** section for Max pool size and overflow
4. Review the section for VNX iSCSI block storage, i.e. VNX system IP address, access credentials, etc.
5. Review the section for VNX FibreChannel block storage, i.e. VNX system IP address, access credentials, etc.

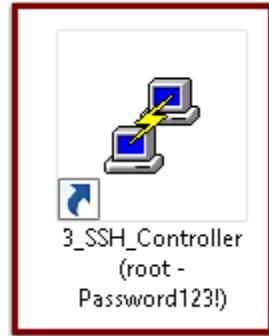
To save you time a cinder.conf file with all the highlighted settings has been precreated. As part of the lab you will simply copy over the modified file.

6. Please close the **2_VNX Template of Cinder.conf**



4.5 PuTTY SSH Client

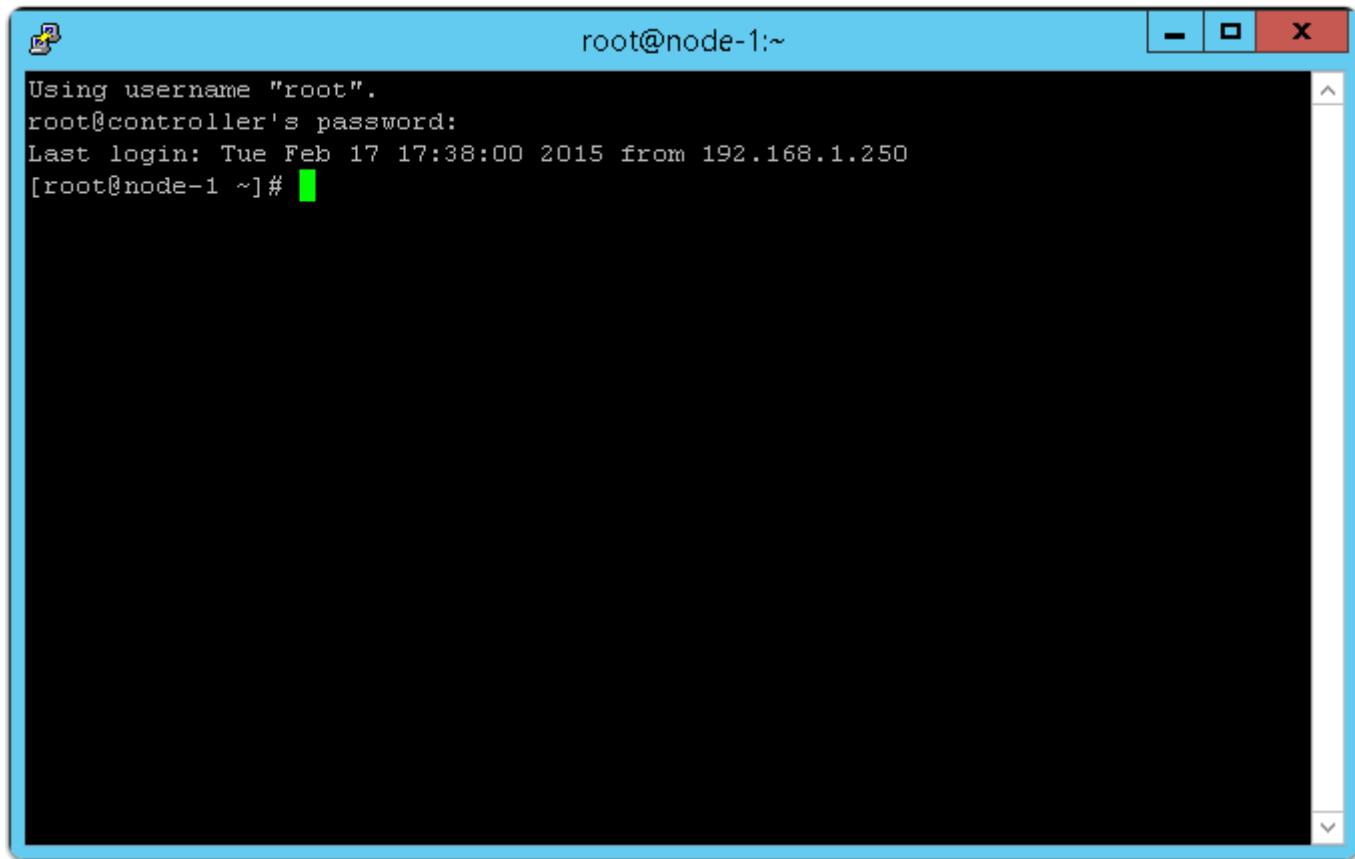
Open 3_SSH_Controller shortcut - it launches OpenStack Controller remote terminal using PuTTY SSH client.



4.6 Logging into SSH Terminal

The shortcut will automatically log you in. Just in case - the SSH login credentials are *root - Password123!*

DO NOT CLOSE the SSH Terminal window and return back to the Lab-1 folder



A screenshot of an SSH terminal window titled "root@node-1:~". The window shows the following text:

```
Using username "root".
root@controller's password:
Last login: Tue Feb 17 17:38:00 2015 from 192.168.1.250
[root@node-1 ~]#
```

The terminal has a blue border and standard window controls (minimize, maximize, close) at the top right.

4.7 VNX Driver Configuration Commands File

Please open the **4_VNX_Cinder_Config_Commands** File by double-clicking on it.

This file contains the commands you will be using to configure VNX Cinder driver. You can copy and paste the commands in this file to your PuTTY SSH terminal window by using right click, Copy and Paste.



4.8 Configuration Commands

1. Please arrange the SSH Terminal window and the 4_VNX_Cinder_Config_Commands file side-by-side. Please copy and paste the commands in this file to your PuTTY SSH terminal window one command block at the time!
2. Install the NaviCLI and observe the command execution in the SSH Terminal
3. Copy over of pre-configured *cinder.conf* and *policy.json* files
4. Restart Cinder services. Note: The openstack-cinder-volume service may not be running before the restart command. This is normal and will not affect you lab execution, as long as it starts successfully.
5. Create Cinder Volume Types and assign them to the backends.
6. Review the created Volume Types and associated Extra Specs.

```

root@node-1:~#
Last login: Tue Mar  3 04:12:00 2015 from 192.168.1.250
[root@node-1 ~]# rpm -ivh /tmp/vnx/NaviCLI-Linux-64-x86-en_US-7.33.3.0.72-1.x86_64.rpm
Preparing... ################################# [100%]
  package NaviCLI-Linux-64-x86-en_US-7.33.3.0.72-1.x86_64 is already installed
[root@node-1 ~]# /opt/Navisphere/bin/navisecccli security -certificate -setLevel low
[root@node-1 ~]# cp /tmp/vnx/cinder.conf.vnx /etc/cinder/cinder.conf
[root@node-1 ~]# chown cinder:cinder /etc/cinder/cinder.conf
[root@node-1 ~]# cp /tmp/vnx/policy.json.vnx /etc/cinder/policy.json
[root@node-1 ~]# chown cinder:cinder /etc/cinder/policy.json
[root@node-1 ~]# ls -l /etc/cinder
total 184
-rw----- 1 cinder cinder 2281 Feb 21 05:29 api-paste.ini
-rw----- 1 cinder cinder 78165 Mar  3 04:18 cinder.conf
-rw----- 1 cinder cinder 77330 Feb 21 07:38 cinder.conf.orig
-rw-r--r-- 1 cinder cinder 224 Feb 23 21:22 cinder_scaleio.config
-rw-r--r-- 1 cinder cinder 3152 Mar  3 04:18 policy.json
-rw-r--r-- 1 cinder cinder 3200 Feb 21 07:39 policy.json.orig
-rw-r--r-- 1 root  cinder  942 Dec 16 14:51 rootwrap.conf
drwxr-xr-x 2 root  root  4096 Feb 22 04:55 rootwrap.d
drwxr-xr-x 2 cinder root  4096 Dec 16 14:52 volumes
[root@node-1 ~]# service openstack-cinder-api restart
Stopping openstack-cinder-api:                                         [ OK ]
Starting openstack-cinder-api:                                         [ OK ]
[root@node-1 ~]# service openstack-cinder-volume restart
Stopping openstack-cinder-volume:                                       [ OK ]
Starting openstack-cinder-volume:                                       [ OK ]
[root@node-1 ~]# source /root/adminrc
[root@node-1 ~]#
[root@node-1 ~]# cinder type-create "VNX-ThickVolume"
cinder type-key VNX-ThickVolume set volume_backend_name=vnx_iscsi
cinder type-create "VNX-ThinVolume"
cinder type-key VNX-ThinVolume set volume_backend_name=vnx_iscsi
cinder type-create "VNX-DeduplicatedVolume"
+-----+

```

Please copy and paste commands into SSH Controller
one block at the time!

1

2

3

4

5

6

4.9 Created VNX Volume Types

1. Please review the created Cinder Volume Types
 2. Please review the created Cinder Volume Extra Specs.
 3. When finished, please minimize the SSH Terminal window.

You have new mail in /var/spool/mail/root

[root@node-1 ~]# cinder type-list

ID	Name
5e05bc40-4259-42dc-b49a-6d0959595f62	VNX-CompressedVolume
765fc54d-ba2a-4a6f-a25e-ad4941577c63	VNX-ThinVolume
7ce4f825-5e49-441a-99c9-125d8cdc0f11	VNX-DeduplicatedVolume
bbf9fbb9-afbc-4f5a-8237-1b79342da114	VNX-ThickVolume

[root@node-1 ~]# cinder extra-specs-list

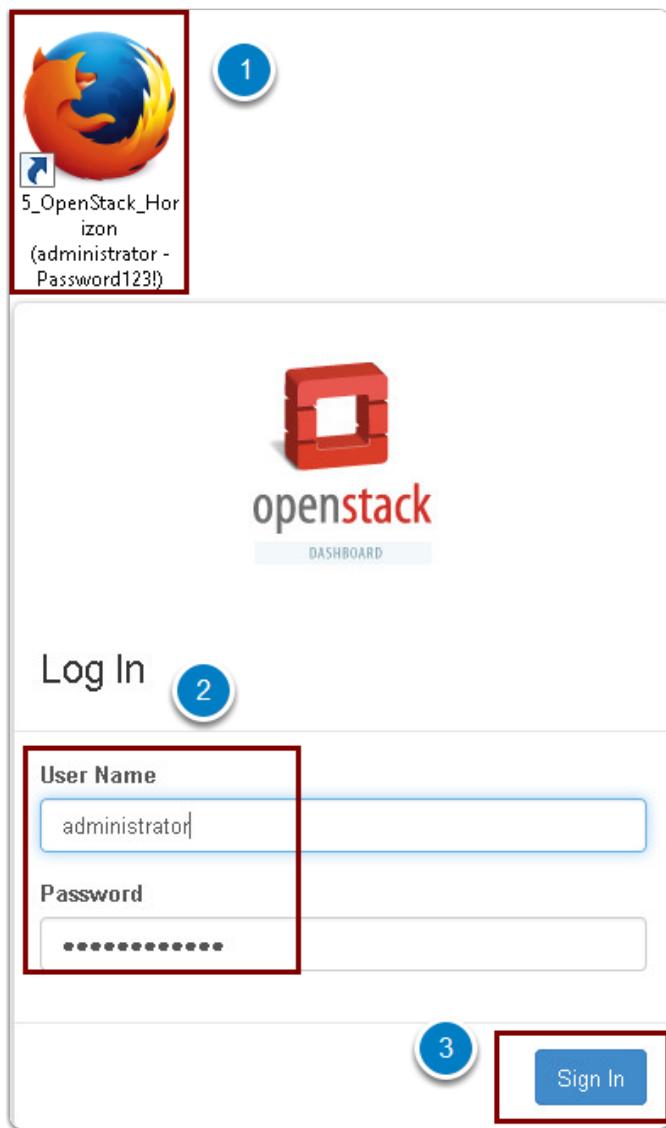
ID	Name
	extra_specs

ID	Name	extra_specs
5e05bc40-4259-42dc-b49a-6d0959595f62	VNX-CompressedVolume	(u'storage_type:provisioning': u'compressed', u'volume_backend_name': u'venx_iscsi', u'compression_support': u'True')
765fc54d-ba2a-4a6f-a25e-ad4941577c63	VNX-ThinVolume	(u'storage_type:provisioning': u'thin', u'volume_backend_name': u'venx_iscsi')
7ce4f825-5e49-441a-99c9-125d8cdc0f11	VNX-DeduplicatedVolume	(u'storage_type:provisioning': u'deduplicated', u'volume_backend_name': u'venx_iscsi', u'deduplication_support': u'True')
bbf9fbb9-afbc-4f5a-8237-1b79342da114	VNX-ThickVolume	(u'storage_type:provisioning': u'thick', u'volume_backend_name': u'venx_iscsi')

[root@node-1 ~]#

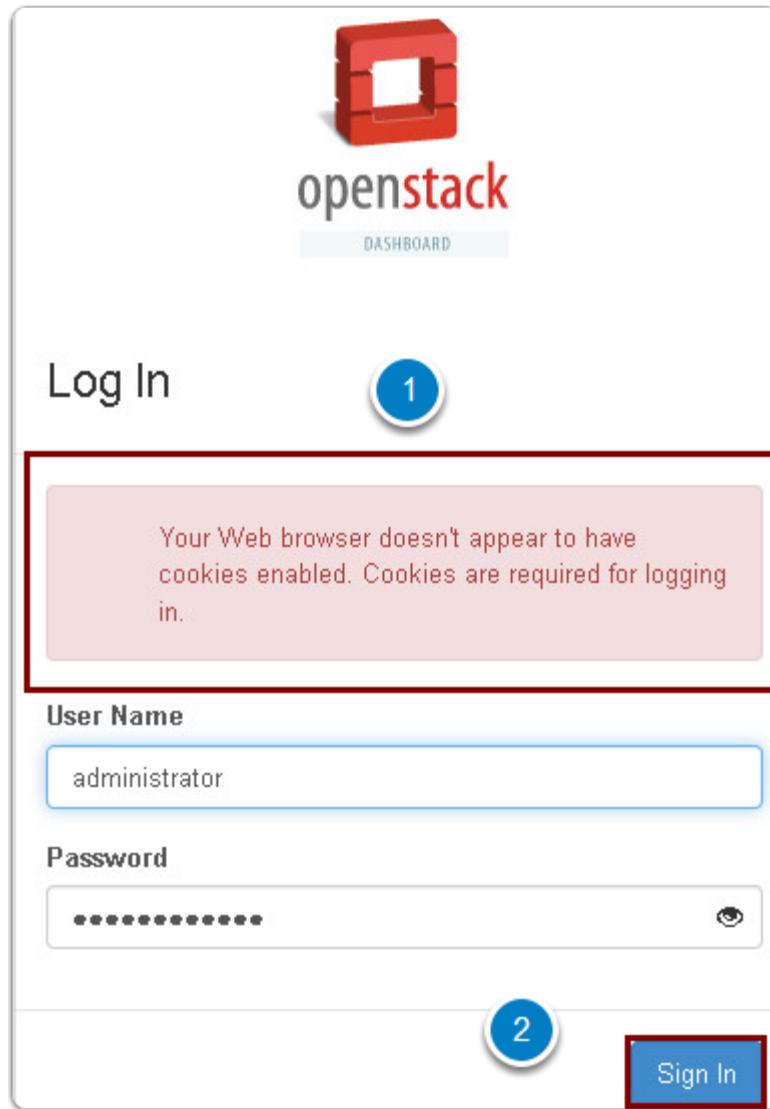
5. Using OpenStack Horizon UI

1. Return to the Lab-1-VNX-Cinder folder and launch the 5_OpenStack_Horizon(administrator - Password123!) shortcut
2. The user credentials should be cached in the browser window. If needed please type the credentials:
 - User Name: administrator
 - Password: Password123!
3. Click Sign In to continue



5.1 Browser Cookie Warning

1. Occasionally you may see a browser cookie warning
2. Please disregard it and simply click on Sign In again to continue



5.2 OpenStack Cloud Overview

1. Please review the Admin dashboard
2. Next select the Volumes Tab

The screenshot shows the OpenStack Admin dashboard with the following details:

- Header:** openstack, cloud dropdown, administrator, Sign Out.
- Left Sidebar:** Project dropdown, Admin (selected), System, Overview (highlighted with a red box), Hypervisors, Host Aggregates, Instances, Volumes (highlighted with a red box and circled with a blue '2'), Flavors.
- Main Content:**
 - Overview:** Usage Summary, Select a period of time to query its usage: From 2015-02-01 To 2015-02-02, Submit button, note: The date should be in YYYY-mm-dd format.
 - Active Instances:** 0 Active RAM: 0 bytes This Period's VCPU-Hours: 0 This Period's GB-Hours: 0
 - Usage:** Download CSV Summary button, table headers: Project Name, VCPUs, Disk, RAM, VCPU Hours, Disk GB Hours.
 - Table Data:** No items to display, Displaying 0 items.

5.3 OpenStack Volume Types

1. Please review the **Volumes** tab
2. Next select **Volume Types**
3. Review newly created **Volume Types**
4. Next, review **Extra Specs** for each VNX volume types - remember the settings in the Cinder.conf you have reviewed earlier? Please click on the **Extra Specs**.

The screenshot shows the OpenStack Horizon dashboard. The left sidebar is collapsed, showing options like Project, Admin, and System. Under System, the 'Volumes' option is highlighted with a red box and a blue circle containing the number 1. The main content area is titled 'Volumes' (circled with a blue 2). Below it is a sub-section titled 'Volume Types'. A table lists four volume types: 'VNX-ThickVolume', 'VNX-ThinVolume', 'VNX-CompressedVolume', and 'VNX-DuplicatedVolume'. The 'Name' column header is circled with a blue 3. To the right of the table are two buttons: '+ Create Volume Type' and 'Delete VolumeTypes'. The 'Actions' column contains four dropdown menus, each with 'View Extra Specs' and other options. The first dropdown is circled with a blue 4. At the bottom of the 'Volume Types' section, it says 'Displaying 4 items'. Below this is another section titled 'QOS Specs' with a table and a '+ Create QOS Spec' button.

Name	Associated QOS Spec	Actions
VNX-ThickVolume		View Extra Specs Delete
VNX-ThinVolume		View Extra Specs Delete
VNX-CompressedVolume		View Extra Specs Delete
VNX-DuplicatedVolume		View Extra Specs Delete

5.4 Volume Type Extra Specs

1. Please review the VNX Volume Type extra specs - these were configuration parameters you have provided while creating this Volume Type. These parameters correspond to the VNX backend configuration.

Volume Type: VNX-ThickVolume

Volume Type Extra Specs

Extra Specs [+ Create](#) [✖ Delete Extra Specs](#)

<input type="checkbox"/>	Key	Value	Actions
<input type="checkbox"/>	storagetype:provisioning	thick	Edit ▼
<input type="checkbox"/>	volume_backend_name	vnx_iscsi	Edit ▼

6. OpenStack Project

1. Please switch to the Project Tab
2. Expand the Compute Tab if necessary
3. Select Instances Tab
4. Click on the drop-down menu list
5. Select Resume Instance
6. Note this Instance external IP Address - we'll be using it to SSH into this instance later in this lab.

The screenshot shows the OpenStack Instances page. The left sidebar has a 'Project' tab (1) selected, followed by 'Compute' (2), 'Instances' (3), 'Volumes', 'Images', 'Access & Security', 'Network', 'Orchestration', 'Admin', and 'Identity'. The main area is titled 'Instances' and shows a table with one item. The table columns are: Instance Name, Image Name, IP Address, Size, Key Pair, Status, Availability Zone, Task, Power State, Time since created, and Actions. The single row shows 'Cloud-Workload' as the Instance Name, 'TestVM' as the Image Name, '192.168.111.4' and '192.168.1.151' as IP Addresses (with the second one highlighted in red), 'micro' as the Size, '-' as the Key Pair, 'Suspended' as the Status, 'nova' as the Availability Zone, 'None' as the Task, 'Shut Down' as the Power State, and '21 hours, 6 minutes' as the Time since created. The 'Actions' column contains a dropdown menu (4) with options: 'Create Snapshot' (highlighted in red), 'Associate Floating IP', 'Edit Instance', 'Resume Instance' (highlighted in red), and 'Terminate Instance'. Step 6 points to the IP address '192.168.1.151'.

Instance Name	Image Name	IP Address	Size	Key Pair	Status	Availability Zone	Task	Power State	Time since created	Actions
Cloud-Workload	TestVM	192.168.111.4 192.168.1.151	micro	-	Suspended	nova	None	Shut Down	21 hours, 6 minutes	<div style="border: 1px solid #ccc; padding: 2px;">Create Snapshot</div> <div style="border: 1px solid #ccc; padding: 2px;">Associate Floating IP</div> <div style="border: 1px solid #ccc; padding: 2px;">Edit Instance</div> <div style="border: 1px solid #ccc; padding: 2px;">Resume Instance</div> <div style="border: 1px solid #ccc; padding: 2px;">Terminate Instance</div>

6.1 Project Volumes

1. Please select the **Volumes** tab next - you will now create a new Volume using one of the VNX Volume Types created in the earlier steps
2. Select **Create Volume**

The screenshot shows the OpenStack dashboard for the 'Compute' project. The left sidebar has 'Compute' selected under 'Project'. The 'Volumes' tab is highlighted with a red box and a blue circle containing the number '1'. On the right, the 'Volumes' page is displayed with a table header:

Name	Description	Size	Status	Type	Attached To	Availability Zone	Bootable	Encrypted	Actions
------	-------------	------	--------	------	-------------	-------------------	----------	-----------	---------

The table body shows 'No items to display.' and 'Displaying 0 items'. At the top right of the 'Volumes' page, there is a 'Create Volume' button with a plus sign, which is also highlighted with a red box and a blue circle containing the number '2'.

6.2 Creating New Volume

OpenStack volumes are persistent block-storage devices that may be attached and detached from instances, but they can be attached to only one instance at a time. Similar to an external hard drive, they do not provide shared storage in the way a network file system or object store does. It is left to the operating system in the instance to put a file system on the block device and mount it.

You will now create a new Volume using earlier created Volume Types:

1. Please enter the Volume name: **VNX-Thin-Volume-1**
2. Provide Volume Description
3. Select **Volume Type** from the drop-down list - **VNX-ThinVolume**
4. Enter the **Volume Size**: **10GB**
5. Please select the **Availability Zone**: **nova**
6. Click **Create Volume**

Note: Please leave all other fields with default values.

Create Volume

Volume Name *

VNX-Thin-Volume-1

1

Description

New VNX Volume using Thin provisioning

2

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

VNX-ThinVolume

3

▼

Size (GB) *

10

4

Availability Zone

nova

5

6

Cancel

Create Volume

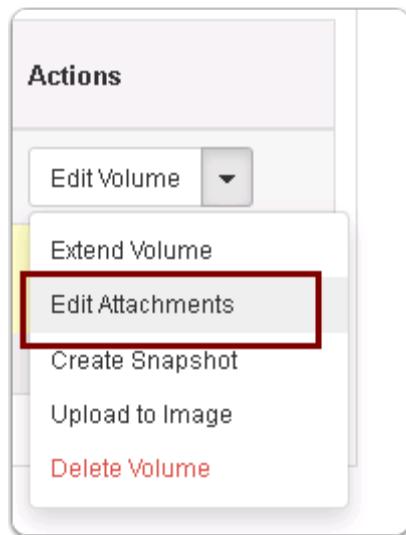
6.3 Newly Created Volume

1. Review the newly created Volume
2. Verify the Volume Status: Available
3. Check the Volume Type - VNX-ThinVolume
4. Observe that the Volume is not attached to any Instance.

<input type="checkbox"/>	Name 1	Description	Size	Status 2	Type 3	Attached To 4	Availability Zone	Bootable	Encrypted	Actions
<input type="checkbox"/>	VNX-ThinVolume-1		10GB	Available	VNX-ThinVolume		nova	No	No	Edit Volume ▾

6.4 Edit Volume Attachments

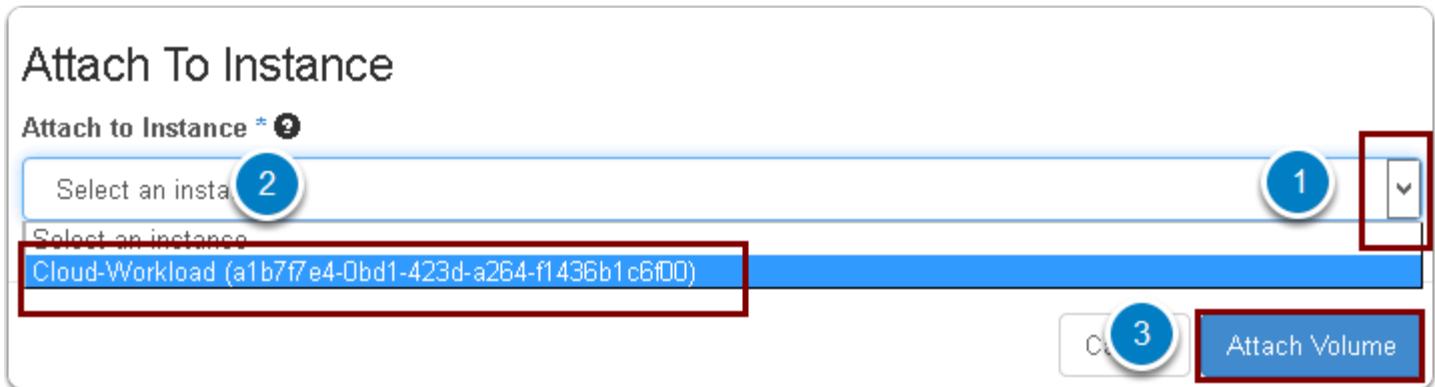
Select Action from the drop-down menu - **Edit Attachments**. We will now edit Volume Attachments



6.5 Attach Volume to Instance

1. Please click on the drop-down menu

2. Select the the Cloud-Workload instance from the list
3. Click Attach Volume



6.6 Attached Volume

Please review the attached volume:

1. Review the changed Volume Status: in-Use
2. Review the Attached To field and the device name - you will need to use it shortly

<input type="checkbox"/>	Name	Description	Size	Status 1	Type	Attached To 2
<input type="checkbox"/>	VNX-ThinVolume-1		10GB	In-Use	VNX-ThinVolume	Attached to Cloud-Workload on /dev/vdb

7. Launching EMC Unisphere

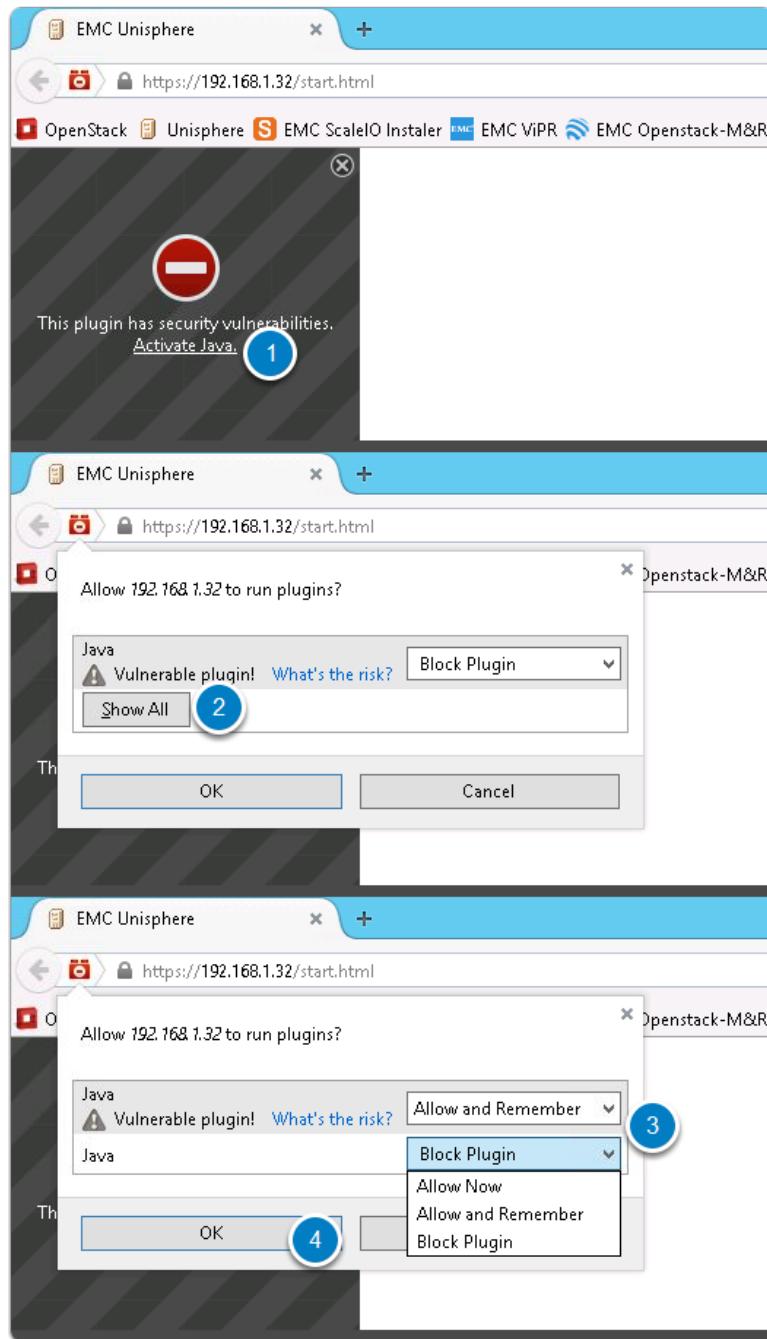
Now that the Volume has been created, you can verify the new Volume on the VNX storage system using EMC Unisphere. Please double-click on the 6_EMC_Unisphere (sysadmin-sysadmin) shortcut in the Lab-1 folder.



7.1 Optional Step: Java Security Warning

If you see Java Security warning accept plugin by doing the following:

1. Click **Activate Java** link
2. Click **Show All** button
3. Select **Allow and Remember** for both plugins
4. Click **OK**

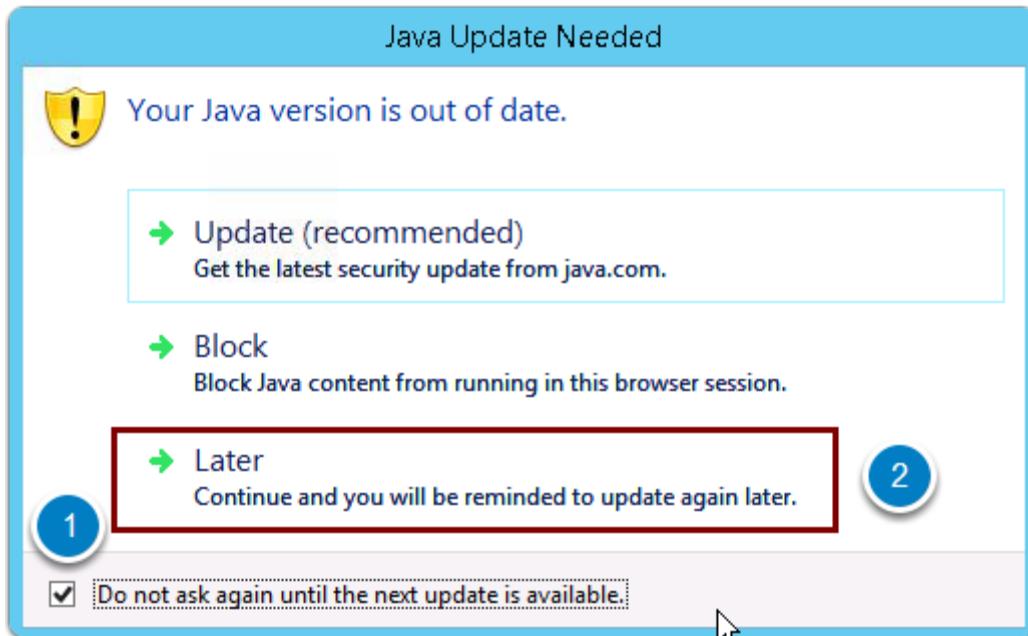


7.2 Optional Step: Java Update

You may see a Java Update Needed window.

1. Click **Do not ask again until the next update is available** checkbox
2. Click **Later**

Note: You may need to repeat these steps twice!

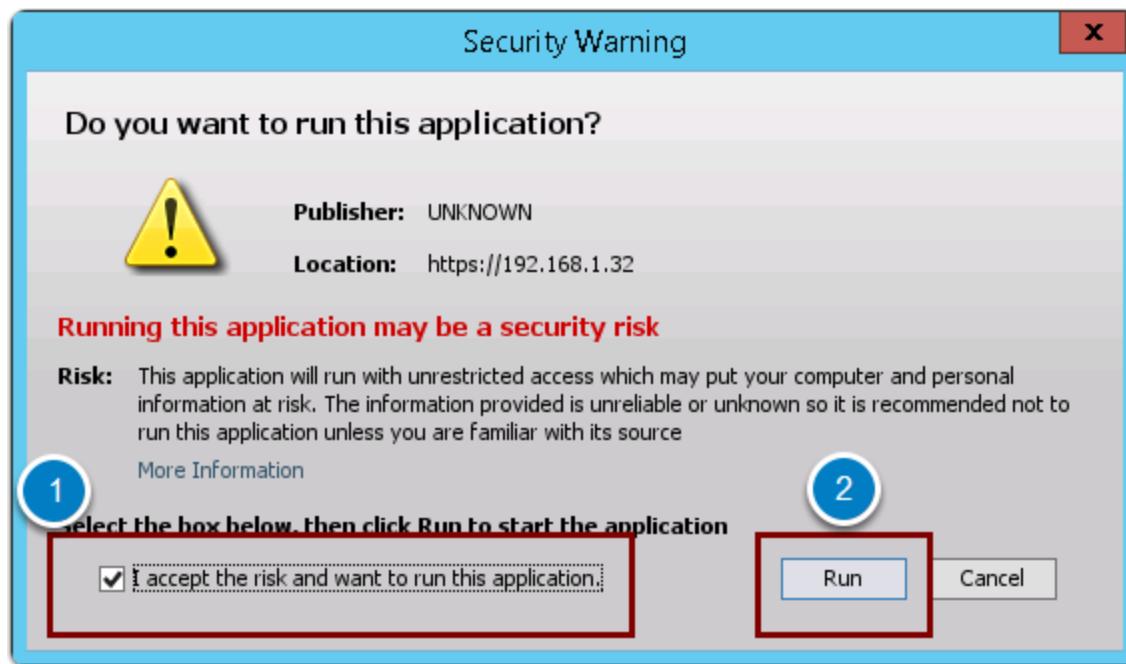


7.3 Accepting Java Security Warning

Please pay attention to the security warning for Java applet:

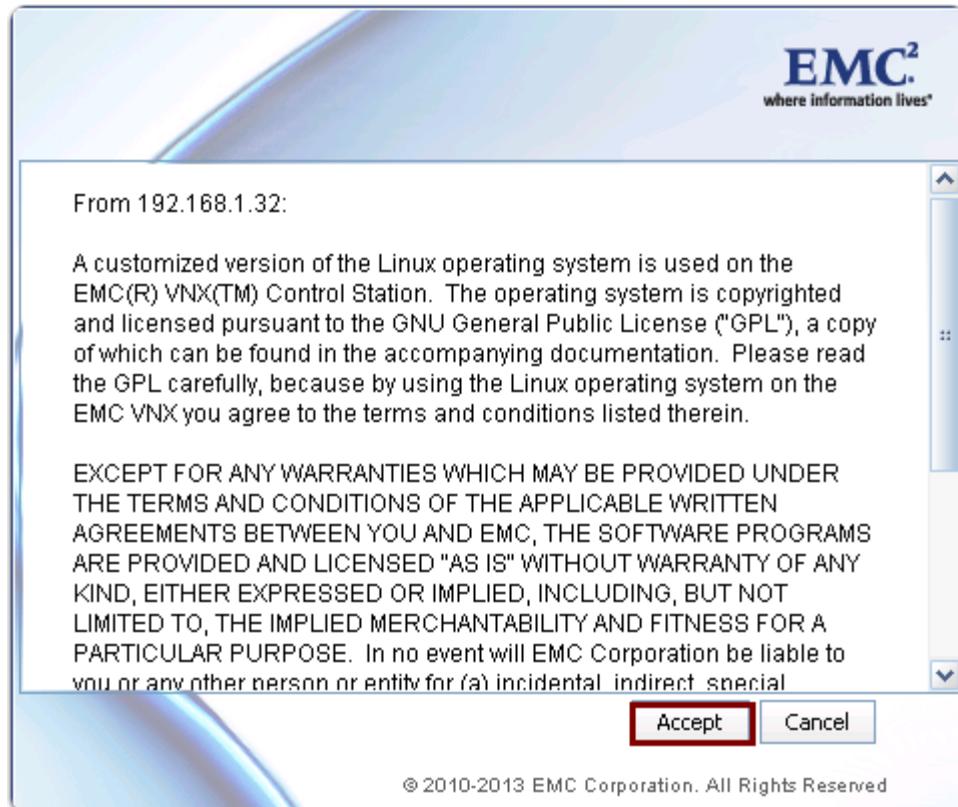
1. Check the box to accept the risk
2. Click Run to continue

Note: You may need to repeat these steps twice!



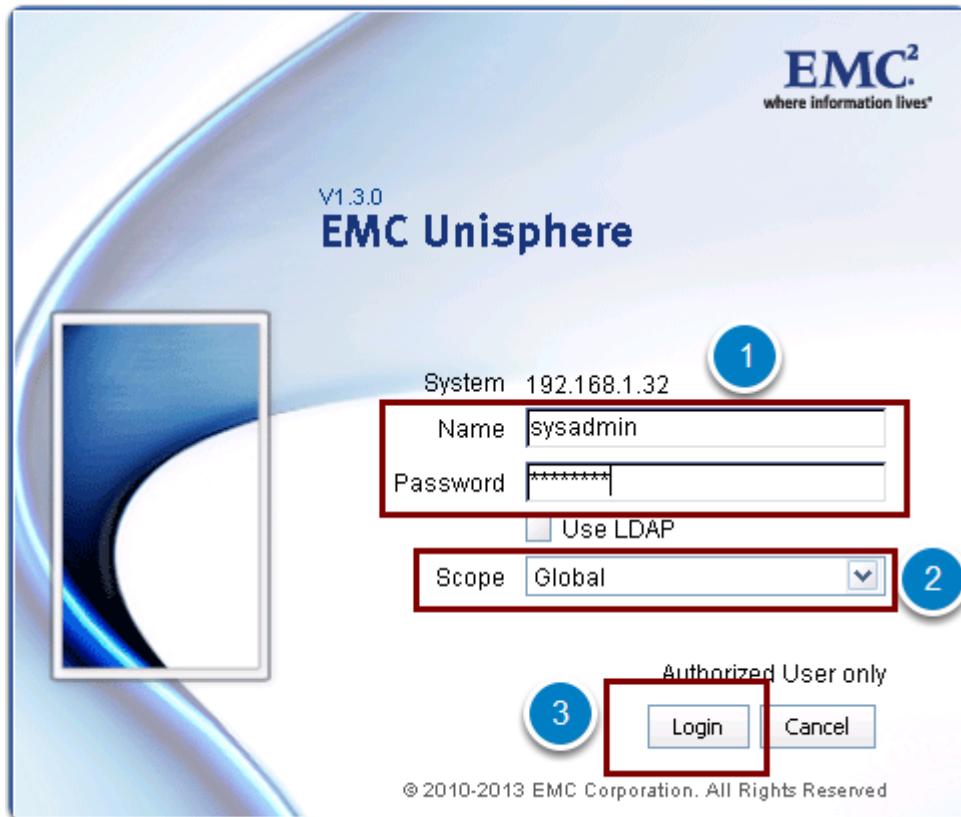
7.4 Accepting EMC License Warning

Please accept EMC Corporation License agreement:



7.5 Logging into Unisphere

1. Please login into Unisphere with credentials: **sysadmin - sysadmin**
2. Make sure your **Scope** is selected as **Global**
3. Click **Login** to continue



7.6 Unisphere Dashboard

1. In the Unisphere Dashboard please select the Storage System called VSA00084500833

The screenshot shows the EMC Unisphere Dashboard interface in Mozilla Firefox. The URL in the address bar is <https://192.168.1.32/start.html>. The dashboard has a blue header with the Unisphere logo and navigation tabs: All Systems, Dashboard (which is selected), System List, Domains, and Alerts.

Systems by Severity

System	Domain	Status	Model
VSA00084500833	Local	Error (1)	VNX5400 (U...)

A red box highlights the row for VSA00084500833, and a blue circle with the number 1 is positioned above the first column of the table.

Alerts by Severity

Severity	System	Message	Created
Error	VSA00084500833	DPE (Bus 0 Enclo...	Aug 20, 2014 6:1...

Overall Capacity - Most Free Space (1 of 1)

This chart shows the total capacity of 430.49 GB. The legend indicates "Free Raw Disk" (blue) and "Free Storage Pool" (orange). The chart is labeled "VSA00084500833".

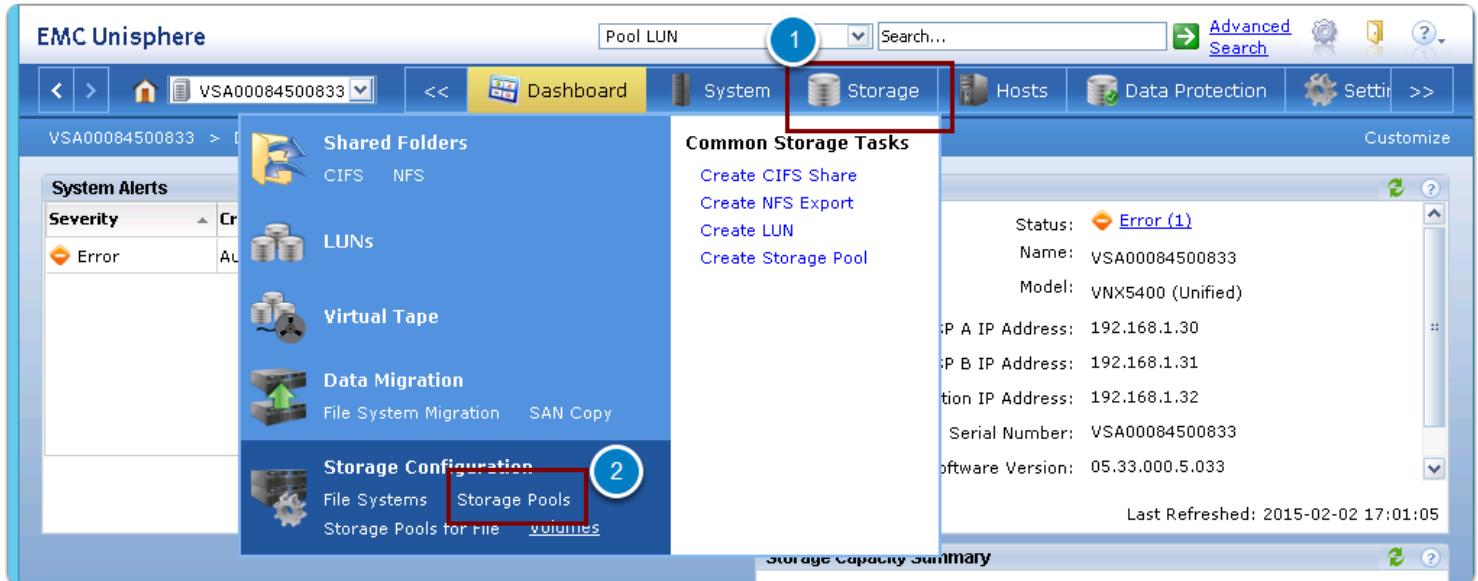
Capacity for File - Most Free Space (1 of 1)

This chart shows the capacity for files, with a value of 10 GB. The legend indicates "Free" (green). The chart is labeled "VSA00084500833".

At the bottom, there are status indicators: Alerts: 1 Error, Certificates: 1, and User: sysadmin. The last refresh times for both charts are listed as 2015-02-02 16:58:32 and 2015-02-02 16:58:34 respectively.

7.7 Block Storage Volumes

1. Next please hover your cursor over the Storage menu from the top menu bar
2. Navigate to the Storage Configuration options and select Storage Pools



7.8 Newly Created Volume(s)

1. Please select the OpenStack Storage Pool - this is the same pool you specified in when creating the Cinder Volume Type earlier in this lab
2. Review the LUNs - note the volume ID, its size, state and thin provisioning type
3. Next select Properties of this newly created LUN

The screenshot shows the EMC Unisphere interface. The top navigation bar includes 'Pool LUN' and a search bar. Below the navigation is a breadcrumb path: VSA00084500833 > Storage > Storage Configuration > Storage Pools. The 'Storage' tab is selected. A sub-navigation bar shows 'Pools' (selected) and 'RAID Groups'. The main area displays a table of storage pools:

Name	FAST Cache	State	RAID Type	Drive Type	Total Capa...	Free Capa...	Allocated ...	%Consum...	Subscribed...	%
OpenStack	On	Ready	RAID5	Mixed	112.720	107.459	5.260	<div style="width: 5%;">5.26</div>	5.26	5.26
Pool 1 - File...	On	Ready	RAID5	SAS	37.573	5.260	32.313	<div style="width: 32.313%;">32.313</div>	32.313	32.313

At the bottom of this table are buttons for 'Selected' (1 Selected), 'Create', 'Delete', 'Properties', and 'Expand'. The status bar indicates 'Last Refreshed: 2015-02-28 17:51:47'.

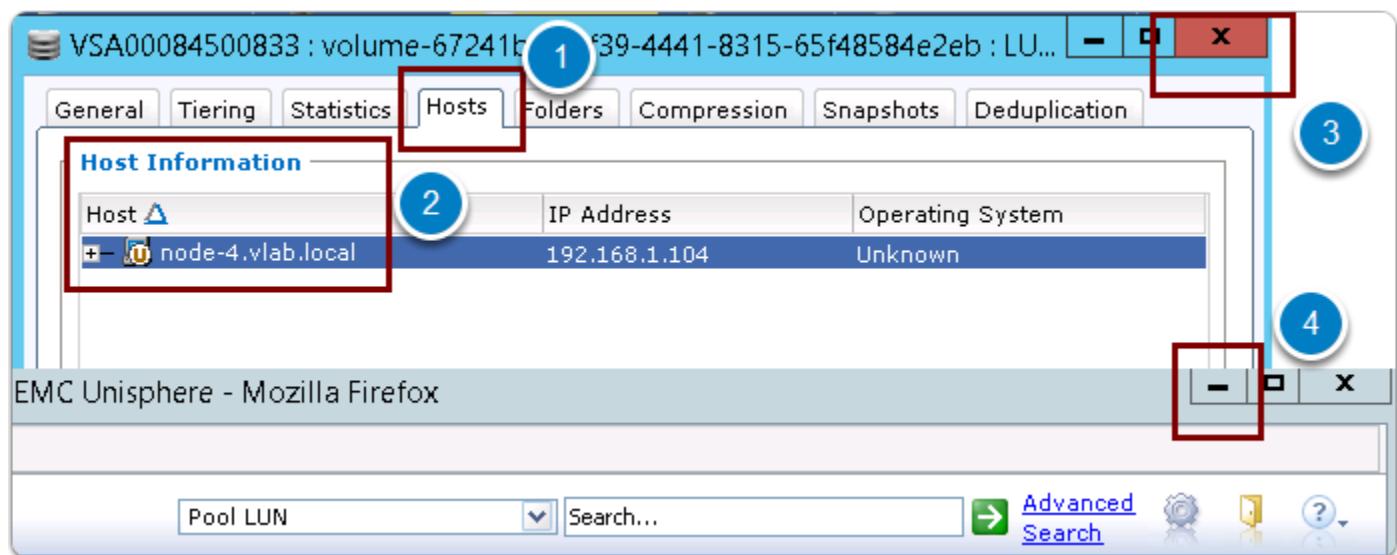
The second part of the screenshot shows the 'Details' section for LUNs. The 'LUNs' tab is selected. A filter bar shows 'Usage ALL User LUN'. The table lists LUN details:

Name	ID	State	Thin	Compr...	Snapsh...	User C...	Curren...	Host
volume-67241bca-1f39-4441-8315-65f48584e2eb	0	Ready	On	Off	On	10.000	SP B	node

At the bottom of this table are buttons for 'Selected' (1 Selected), 'Delete', 'Properties' (3), and 'Storage Group'. The status bar indicates 'Filtered: 1 of 1'.

7.9 Volume Properties

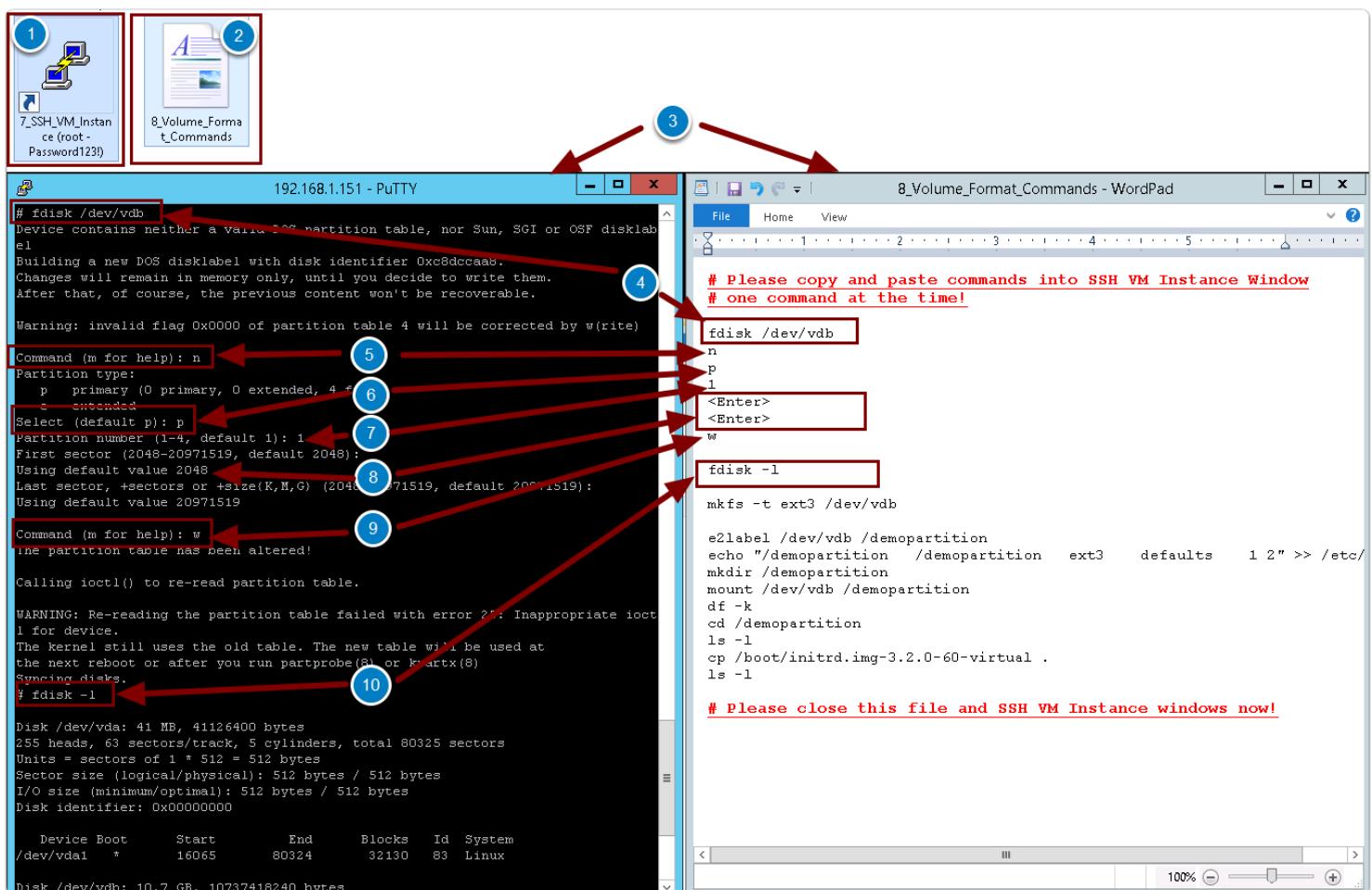
1. Select Hosts Tab in the LUN's Properties dialog
2. Review the Hosts in this LUN's export group: node-4.vlab.local - this is one of the OpenStack Compute nodes in our cluster
3. Please close the LUN Properties dialog
4. Then minimize EMC Unisphere Browser window.



8. Volume Life-Cycle Management

8.1 Creating New Partition

1. Open minimized Lab-1 Folder and select **7_SSH_VM_Instance (root-Password123!)** - This will open an SSH Terminal connection to the running OpenStack instance using its Floating IP address 192.168.1.151
2. Open shortcut called **8_Volume_Format_Commands**. You can copy the commands from the file into SSH terminal - one command at the time using right click, Copy and Paste.
3. Arrange the opened windows side-by-side as shown
4. Partition the newly created and attached volume: **fdisk /dev/vdb**
5. Enter command **n** (New partition) and press **Enter**.
6. Select default: **p** (for primary partition)
7. Enter partition number: **1**
8. Accept default initial and end blocks to use the entire volume - Use **Enter**
9. Enter command: **w** (write the information and quit)
10. Check the new partition using command: **fdisk -l**



8.2 Formatting New Partition

- Format the new partition as an ext3 file system type: `mkfs -t ext3 /dev/vdb`

The screenshot shows two windows side-by-side. On the left is a Putty terminal window titled "192.168.1.151 - PUTTY". It displays the output of the command `# mkfs -t ext3 /dev/vdb`. The output includes details about the filesystem creation, such as block sizes, inodes, and reserved blocks. A red box highlights the command itself, and a blue circle highlights the number "1" at the top right of the Putty window. On the right is a WordPad window titled "8_Volume_Format_Commands - WordPad". It contains a series of Linux commands for partition management and file system creation. A red box highlights the command `mkfs -t ext3 /dev/vdb`, which corresponds to the one in the Putty window. Red text at the top of the WordPad window says "# Please copy and paste commands into SSH VM Instance Window # one command at the time!". Red text at the bottom says "# Please close this file and SSH VM Instance windows now!". A red arrow points from the Putty window to the WordPad window, indicating where to copy the command.

```

# mkfs -t ext3 /dev/vdb
mke2fs 1.42.2 (27-may-2012)
Filesystem label=
OS type: Linux
Block size=4096 (log=2)
Fragment size=4096 (log=2)
Stride=0 blocks, Stripe width=0 blocks
655360 inodes, 2621440 blocks
131072 blocks (5.00%) reserved for the super user
First data block=0
Maximum filesystem blocks=2684354560
80 block groups
32768 blocks per group, 32768 fragments per group
8192 inodes per group
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632

Allocating group tables: done
Writing inode tables: done
Creating journal (32768 blocks): done
Writing superblocks and filesystem accounting information: done

```

```

File   Home   View
1 2 3 4 5

# Please copy and paste commands into SSH VM Instance Window
# one command at the time!

fdisk /dev/vdb
n
p
1
<Enter>
<Enter>
w

fdisk -l

mkfs -t ext3 /dev/vdb

e2label /dev/vdb /demopartition
echo "/demopartition /demopartition ext3 defaults 1 2" >> /etc/fstab
mkdir /demopartition
mount /dev/vdb /demopartition
df -k
cd /demopartition
ls -l
cp /boot/initrd.img-3.2.0-60-virtual .
ls -l

# Please close this file and SSH VM Instance windows now!

```

8.3 Mounting New Volume

You will now prepare the new volume and write new data to it. You can copy and paste the commands below directly into the SSH Terminal window using right click, Copy and Paste.

- Assign a Label with e2label: `e2label /dev/vdb /demopartition`
- Add the new partition to /etc/fstab: `echo "/demopartition /demopartition ext3 defaults 1 2" >> /etc/fstab`
- Verify the new fstab entry: `cat /etc/fstab`
- First create the base directory: `mkdir /demopartition`
- Mount the new file system: `mount /dev/vdb /demopartition`
- And check it: `df -k`
- Change directory to the new volume: `cd /demopartition`
- List the content: `ls -l`
- Copy data to a new volume: `cp /boot/initrd.img-3.2.0-60-virtual .` Note: Please do not forget the dot at the end of this command line
- List its content again: `ls -l` - Please note the new data file placed in the new volume!

11. When finished, please close the SSH Terminal to the VM instance and the 8_Volume_Format_Commands file. You will no longer need them.

The image shows two terminal windows side-by-side. The left terminal window (PuTTY) displays a series of Linux commands being run:

```

# e2label /dev/vdb /demopartition
# echo "/demopartition /demopartition ext3 defaults 1 2" >> /etc/fstab
# cat /etc/fstab
# /etc/fstab: stat file system information.
#
# <file system> <mount pt> <type> <options> <dump> <pass>
/dev/root   auto    rw,noexec  0 1
proc        proc    defaults    0 0
devpts      /dev/pts devpts  defaults,gid=5,mode=620 0 0
tmpfs       /dev/shm tmpfs   mode=0777    0 0
sysfs      /sys     sysfs  defaults    0 0
tmpfs       /run     tmpfs   rw,nosuid,relatime,size=200k,mode=0 0 0

/demopartition /demopartition ext3 defaults 1 2
# mkdir /demopartition
# mount /dev/vdb /demopartition
# df -k
Filesystem      1K-blocks   Used  Available  Mounted on
/dev             21872      0  21872  /dev
/dev/vda1        23797  18511  4058  82%  /
tmpfs           25448      0  25448  /dev/shm
tmpfs            200      68  132  34%  /run
/dev/vdb        1032206  150  9642688  2%  /demopartition
# cd /demopartition/
# ls -l
total 16
drwxr-xr-x  2 root  root  16384 Feb 25 11:59 lost+found
# cp /boot/initrd.img-3.2.0-60-virtual .
# ls -l
total 3452
-rw-r--r--  1 root  root  3513903 Feb 25 11:59 initrd.img-3.2.0-60-virtual
total 16
drwxr-xr-x  2 root  root  16384 Feb 25 11:59 lost+found

```

The right terminal window (8_Volume_Format_Commands) shows the commands being copied from the left window:

```

# Please copy and paste commands into SSH VM Instance Window
# one command at the time!
fdisk /dev/vdb
n
p
1
<Enter>
<Enter>
w

fdisk -l
fs -t ext3 /dev/vdb
e2label /dev/vdb /demopartition
echo "/demopartition /demopartition ext3 defaults 1 2" >> /etc/fstab
cat /etc/fstab
mkdir /demopartition
mount /dev/vdb /demopartition
df -k
cd /demopartition
ls -l
cp /boot/initrd.img-3.2.0-60-virtual .
ls -l

# Please close this file and SSH VM Instance windows now!

```

Blue numbered circles (1 through 11) point to specific lines of code in both windows, indicating the sequence of operations:

- 1: e2label /dev/vdb /demopartition
- 2: # Please copy and paste commands into SSH VM Instance Window
- 3: # one command at the time!
- 4: fdisk /dev/vdb
- 5: n
- 6: p
- 7: 1
- 8: <Enter>
- 9: <Enter>
- 10: w
- 11: # Please close this file and SSH VM Instance windows now!

8.4 Creating Volume Snapshot

As part of the data protection requirements, OpenStack Block Storage also allows for creating snapshots of volumes. Block-level volume snapshot is crash-consistent, so it is best if the volume is not connected to an instance when the snapshot is taken. Since you have only written a small amount of static data to the new Volume, you will force the Cinder to create a snapshot of the attached volume.

1. Please return to the OpenStack Horizon in your Firefox browser and select the Volumes tab
2. Click on the drop-down menu next to our VNX-ThinVolume-1
3. Select Create Snapshot option

The screenshot shows the 'Volumes' tab in the OpenStack Horizon interface. A single volume, 'VNX-ThinVolume-1', is listed with the following details:

	Name	Description	Size	Status	Type	Attached To	Availability Zone	Bootable	Encrypted	Actions
<input type="checkbox"/>	VNX-ThinVolume-1		10GB	In-Use	VNX-ThinVolume	Attached to Cloud-Workload on /dev/vdb	nova	No	No	Edit Volume ▼ Edit Attachments Create Snapshot Upload to Image

Annotations with circled numbers indicate the steps:

- 1: Points to the 'Volumes' tab in the top navigation bar.
- 2: Points to the dropdown menu icon next to the volume name.
- 3: Points to the 'Create Snapshot' option in the dropdown menu.

8.5 Snapshot Details

Please provide Snapshot details as needed:

1. **Snapshot Name:** VNX-VolumeSnapshot-1
2. **Description:** VNX Volume Snapshot
3. Click **Create Volume Snapshot (Force)** to continue

Create Volume Snapshot

This volume is currently attached to an instance. In some cases, creating a snapshot from an attached volume can result in a corrupted snapshot.

Description:
From here you can create a snapshot of a volume.

Snapshot Limits

Total Gigabytes (88 GB)	1,000 GB Available
Number of Snapshots (1)	10 Available

1 **Snapshot Name ***
VNX-VolumeSnapshot-1

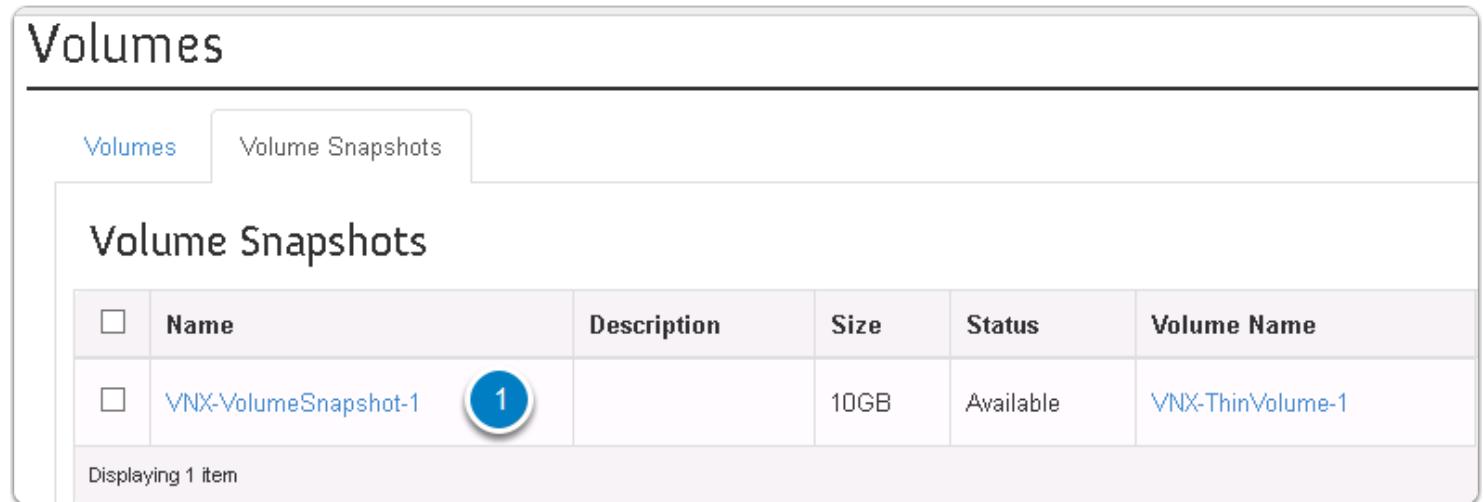
2 **Description**
VNX Volume Snapshot

3 **Create Volume Snapshot (Force)**

8.6 Volume Snapshots

1. Review the details of the newly created Volume Snapshot

Next please switch to the EMC Unisphere browser window - you will review the volume snapshot on the VNX storage array.



The screenshot shows the 'Volume Snapshots' section of the EMC Unisphere interface. At the top, there are two tabs: 'Volumes' (which is selected) and 'Volume Snapshots'. Below the tabs, the title 'Volume Snapshots' is displayed. A table lists one item:

<input type="checkbox"/>	Name	Description	Size	Status	Volume Name
<input type="checkbox"/>	VNX-VolumeSnapshot-1	(1)	10GB	Available	VNX-ThinVolume-1

At the bottom of the table, it says 'Displaying 1 item'.

8.7 Volume Snapshot on VNX

1. Please select our OpenStack storage pool
2. Then select the Volume you created earlier - if you had the volume properties already open, please close this window and then click on the OpenStack storage pool again
3. Select Properties

The screenshot shows the EMC Unisphere interface with the following details:

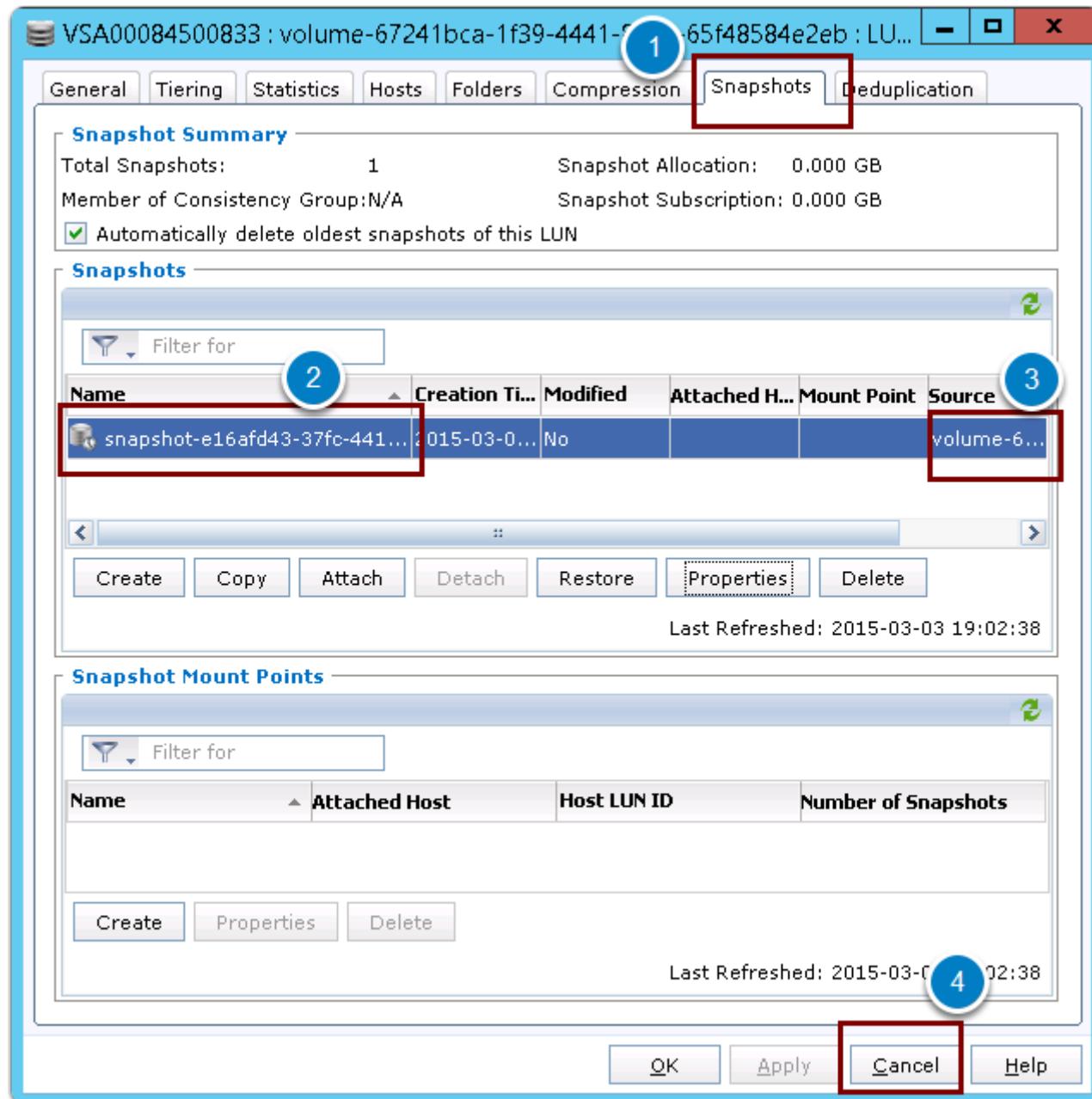
- Storage Pools Panel (Top):**
 - Header: EMC Unisphere, Pool LUN.
 - Navigation: Back, Home, VSA00084500833, Dashboard, System, Storage (highlighted), Hosts, Devices.
 - Breadcrumb: VSA00084500833 > Storage > Storage Configuration > Storage Pools.
 - Buttons: Pools (selected), RAID Groups.
 - Table Headers: Name, FAST Cache, State, RAID Type, Drive Type, Total Capa..., Free Capa..., Allocated ..., %Consum...
 - Table Data:

Name	FAST Cache	State	RAID Type	Drive Type	Total Capa...	Free Capa...	Allocated ...	%Consum...
OpenStack	On	Ready	RAID5	Mixed	112.720	107.459	5.260	46%
Pool 1 - File...	On	Ready	RAID5	SAS	37.573	5.260	32.313	86%
 - Buttons: Create, Delete, Properties, Expand.
 - Text: Last Refreshed: 2015.
- Volume Details Panel (Bottom):**
 - Header: Details.
 - Buttons: LUNs (selected), Snapshot Mount Points, Disks.
 - Table Headers: Name, ID, State, Thin, Compr..., Snapsh..., User C...
 - Table Data:

Name	ID	State	Thin	Compr...	Snapsh...	User C...
volume-67241bca-1f39-4441-8315-65f48584e2eb	0	Ready	On	Off	On	10.000
 - Buttons: Delete, Properties (highlighted with a red box).
 - Text: Go Storage Group.

8.8 Snapshot Properties

1. Select the Snapshot tab in the the Volume Properties dialog
2. Review the newly created Snapshot
3. Review the Source - our VNX Volume
4. Click Cancel and then minimize the Unisphere browser window.



9. Conclusion

In this lab you, wearing the hat of Code Nebulous Cloud Administrator, have successfully accomplished the following:

- Deployed and configured EMC VNX Cinder driver to use the VNX arrays block storage as a backend
- Created Cinder volume types associated with the VNX storage pools
- Create and attached a new block volume to an active OpenStack instance
- Logged into Cloud-Workload instance to format and mount the newly created volume
- Written applications data into the new volume and take a snapshot.

Now the Code Nebulous Cloud Users can begin building and using the applications running on the Cloud-Workload instance and store its data on the persistent block volume.

This lab is now complete. Please close the following windows as you will not be using them any longer:

- Lab-1-VNX-Cinder folder
- EMC Unisphere browser window
- 4_VNX_Cinder_Config_Commands file
- Close SSH Terminal Window

Please proceed to the next lab, or select other labs from the list.

Lab 2 - Using EMC XtremIO with Cinder Block Storage Service

Lab 2 - Using EMC XtremIO with OpenStack Cinder Block Storage

EMC XtremIO is an all-flash purpose-built platform for production and dev/test with predictable scalable performance.

While this lab does not include XtremIO Storage simulator, you will still be able to:

- Deploy an XtremIO Cinder driver
- Perform driver configuration steps
- Create XtremIO Volume types

1. Supported Operations

XtremIO Cinder driver supported operations are available at: [http://docs.openstack.org/trunk/config-reference/
content/emc-xtremio-driver.html](http://docs.openstack.org/trunk/config-reference/content/emc-xtremio-driver.html)

- Create, delete, clone, attach, and detach volumes
- Create and delete volume snapshots
- Create a volume from a snapshot
- Copy an image to a volume
- Copy a volume to an image
- Extend a volume



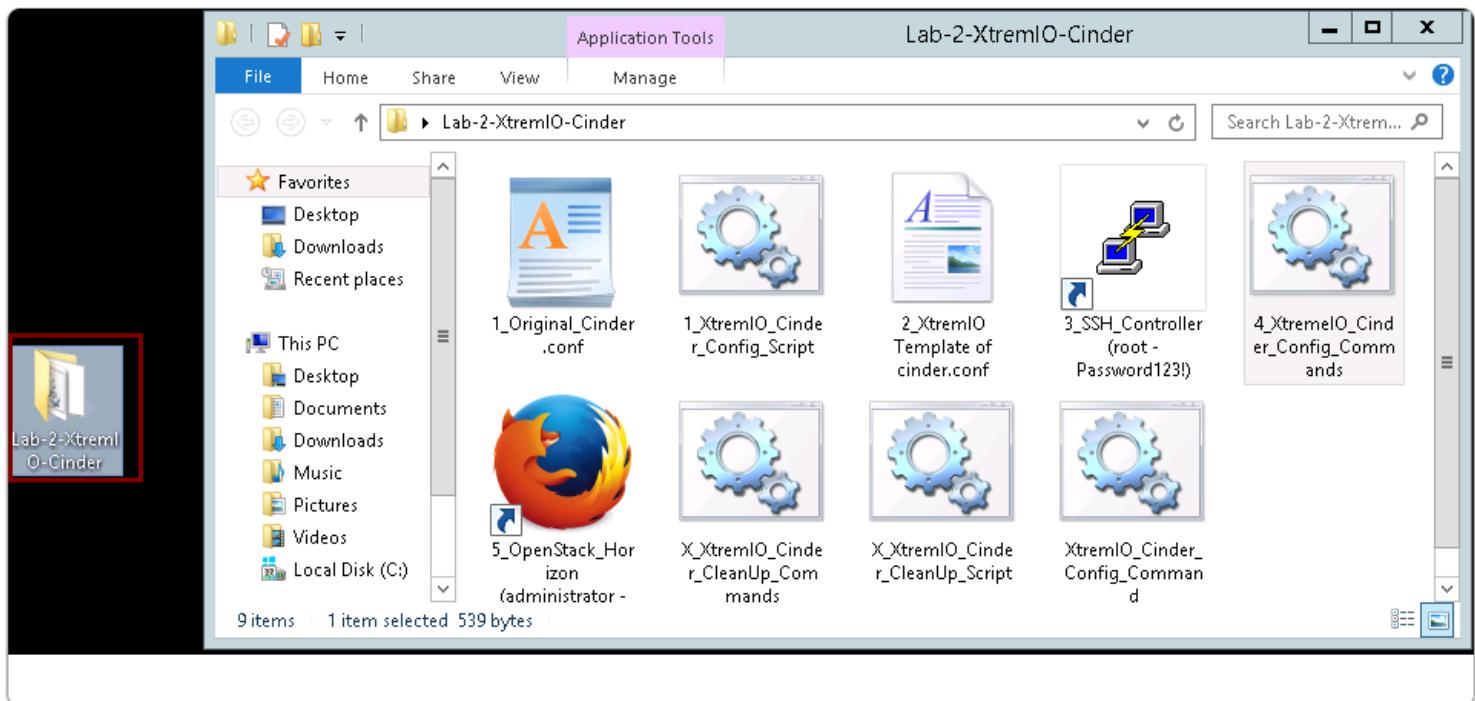
2. Lab Scenario

In this lab as a Cloud Admin will:

- Review the Cinder driver capabilities for EMC XtremIO, including extra-specs
- Review and become familiar with the XtremIO direct Cinder drivers configuration steps
- Follow the configuration steps and implement Cinder integration with EMC XtremIO Storage backend
- Create storage volume types for OpenStack Volumes and associate the Volume Types with Storage Pools
- Review newly available volume types in Horizon

3. Lab-2: XtremIO with Cinder Folder

Review Lab-2 folder



4. XtremIO Cinder Driver

In this section of the lab you will perform configuration steps that are required to enable XtremIO storage backend for OpenStack Cinder service.

Here are the installation steps - please review these steps ONLY, do NOT perform them, as the lab environment has already been prepared for you.

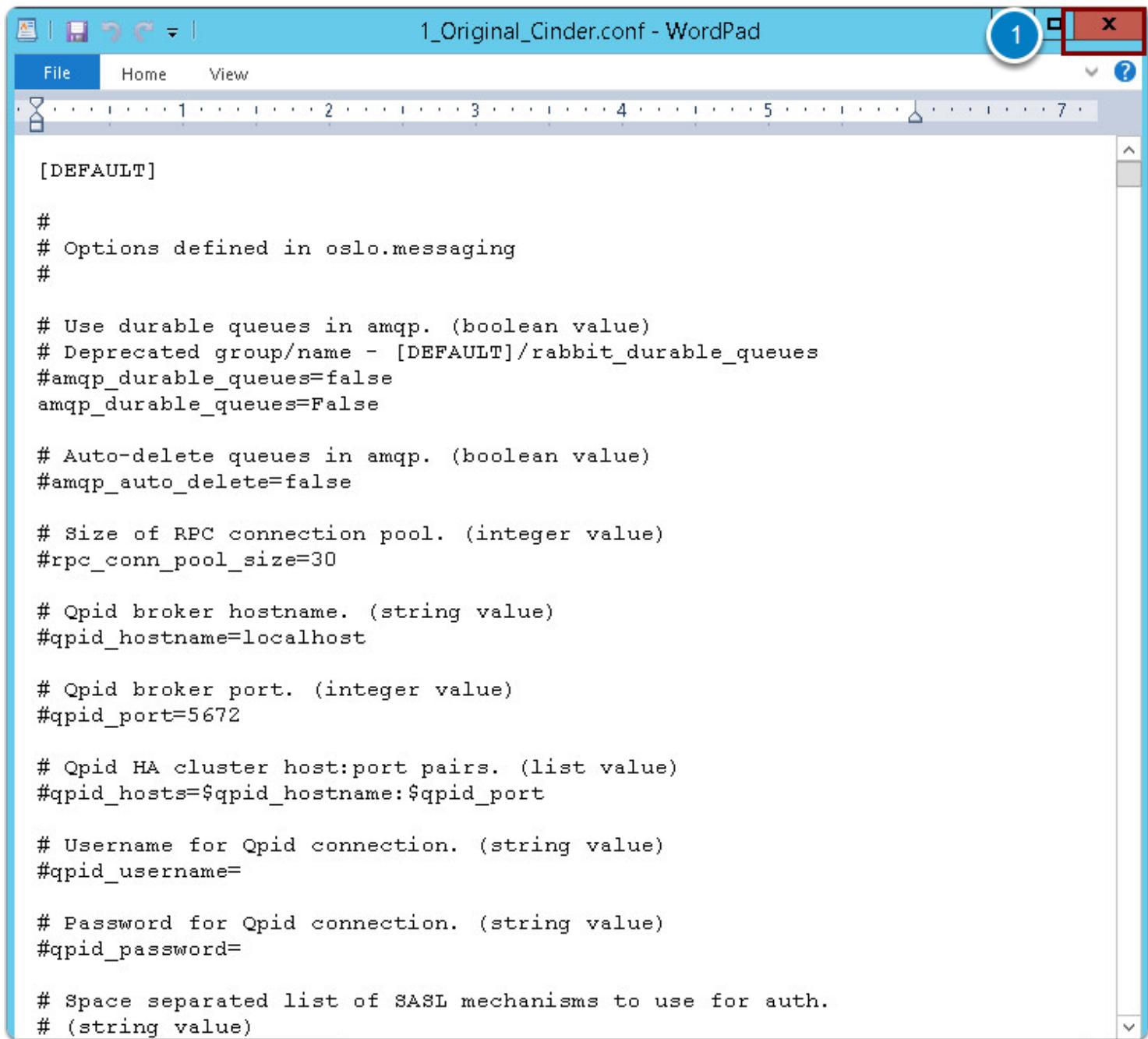
- XtremIO Cinder driver is available in the trunk of the OpenStack Juno release and is a part of every OpenStack software distribution
- XtremIO driver configuration involved making changes to the file called `/etc/cinder/cinder.conf`, containing parameters for the REST API endpoint and Storage Pool.

Please expand the Lab-2-XtremIO-Cinder folder and double-click on `1_Original_Cinder.conf` file shortcut:



4.1 Cinder Configuration File

- As you can see, the original Cinder configuration file is fairly large and complex. Please review the the file and then close it.



```
[DEFAULT]

#
# Options defined in oslo.messaging
#

# Use durable queues in amqp. (boolean value)
# Deprecated group/name - [DEFAULT]/rabbit_durable_queues
#amqp_durable_queues=false
amqp_durable_queues=False

# Auto-delete queues in amqp. (boolean value)
#amqp_auto_delete=false

# Size of RPC connection pool. (integer value)
#rpc_conn_pool_size=30

# Qpid broker hostname. (string value)
#qpid_hostname=localhost

# Qpid broker port. (integer value)
#qpid_port=5672

# Qpid HA cluster host:port pairs. (list value)
#qpid_hosts=$qpid_hostname:$qpid_port

# Username for Qpid connection. (string value)
#qpid_username=

# Password for Qpid connection. (string value)
#qpid_password=

# Space separated list of SASL mechanisms to use for auth.
# (string value)
```

4.2 XtremIO Cinder Configuration

A template has been provided with all necessary settings for enabling XtremIO backend for Cinder.

1. Please open the file called 2_XtremIO_Template_of_Cinder.conf in the Lab-2-XtremIO-Cinder folder
2. Please review the configuration for enabling XtremIO iSCSI and FibreChannel (FC) backends in the DEFAULT section.
3. Review the REST API parameters for iSCSI configuration
4. Review the REST API parameters for FC configuration
5. Please close the file when finished.

1

2

3

4

5

```
[DEFAULT]
# A list of backend names to use. These backend names should
# be backed by a unique [CONFIG] group with its options (list
# value)
enabled_backends = xioiscsi,xiofc

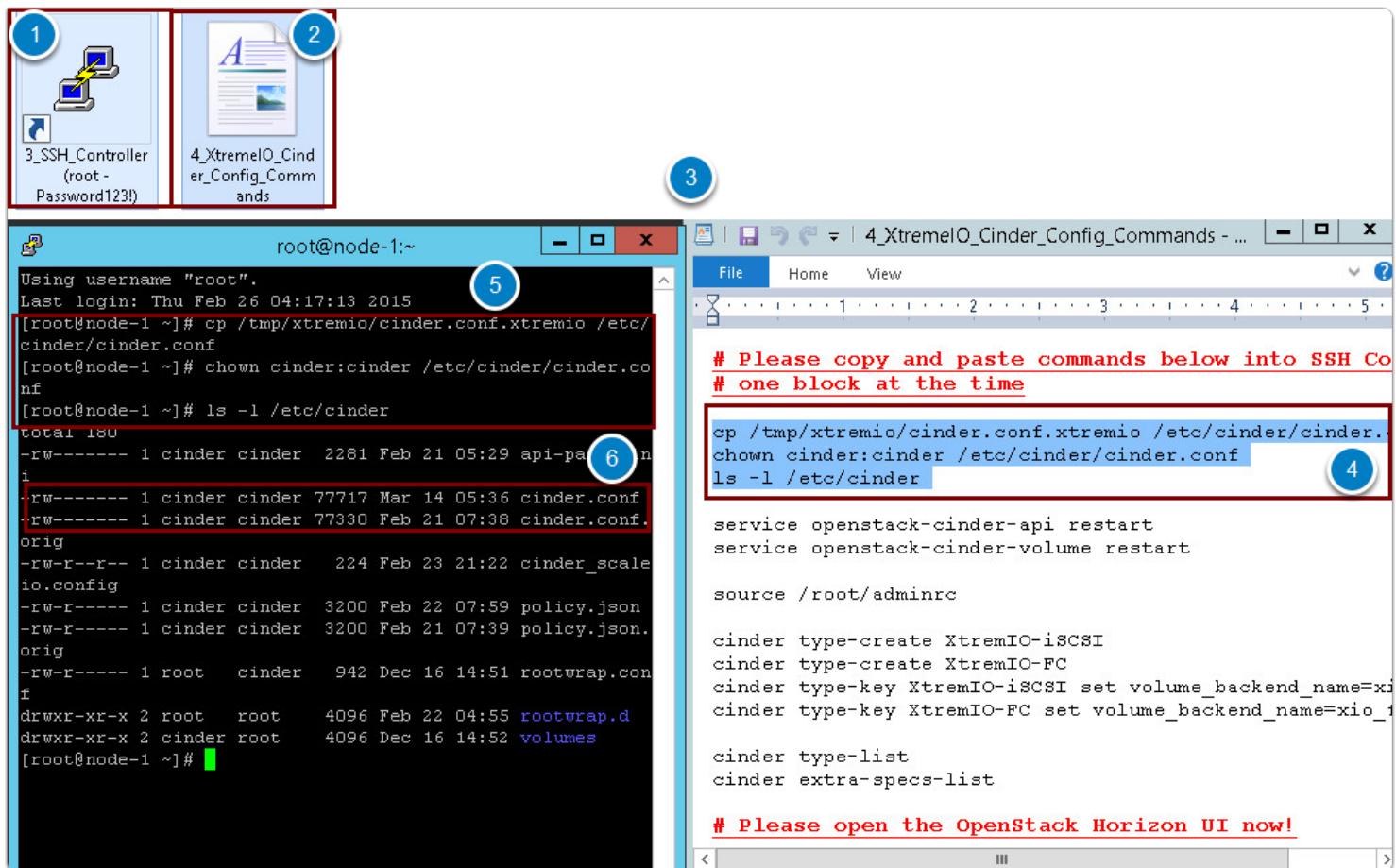
[xioiscsi]
san_ip=192.168.50.50
san_login=openstack
san_password=Password123!
volume_driver=cinder.volume.drivers.emc.xtremio.XtremIOISCSIDriver
volume_backend_name=xio_iscsi

[xiofc]
san_ip=192.168.50.50
san_login=openstack
san_password=Password123!
volume_driver=cinder.volume.drivers.emc.xtremio.XtremIOFibreChannelDriver
volume_backend_name=xio_fc
```

4.3 Using PuTTY SSH Client

To save you some time in running this lab, a file has been created with the commands to be executed on the OpenStack Controller Node to install and configure XtremIO Cinder driver.

1. Please open PuTTY shortcut to OpenStack Controller Node using shortcut called 3_SSH_Controller
2. Then open 4_XtremIO_Config_Commands file
3. Please arrange the open SSH Terminal window and the command file side by side as shown - you will be copying the commands from the file into terminal window using right click, Copy and Paste.
4. It is recommended to copy and paste commands into SSH Terminal window as a block. Blocks are separated by blank lines
5. Please copy the first command block and paste into SSH Terminal window using right-click. A copy of the cinder.conf file has been created in the /tmp directory with all necessary settings you have reviewed earlier - this file is copied over the original /etc/cinder/cinder.conf file. It is also necessary to change the ownership of this file to cinder user
6. Please review the updated cinder.conf file



4.4 Running Configuration Commands on the Controller

1. It is necessary to restart the Cinder API and Volume services to read the new configuration files.
2. Next you will create XtremIO Volume Types - please copy and paste the command blocks into SSH Terminal window.

The screenshot shows a dual-terminal session. The left terminal window is a black box representing an SSH connection to 'root@node-1'. It displays a series of commands run on the OpenStack node:

```

root@node-1 ~]# service openstack-cinder-api restart
Stopping openstack-cinder-api:
[ OK ]
Starting openstack-cinder-api:
[ OK ]
[root@node-1 ~]# service openstack-cinder-volume restart
Stopping openstack-cinder-volume:
[ OK ]
Starting openstack-cinder-volume:
[ OK ]
[root@node-1 ~]# source /root/adminrc
[root@node-1 ~]#
[root@node-1 ~]# cinder type-create XtremIO-iSCSI
+-----+
| ID      | Name   |
+-----+
| be56fa6e-112a-4469-a132-98b433087d3d | XtremIO-iSCSI |
+-----+
[root@node-1 ~]# cinder type-create XtremIO-FC
+-----+
| ID      | Name   |
+-----+
| 80428ee8-f435-493a-b0d3-92d85539b02e | XtremIO-FC |
+-----+
[root@node-1 ~]# cinder type-key XtremIO-iSCSI set volume_backend_name=xio_iscsi
[root@node-1 ~]# cinder type-key XtremIO-FC set volume_backend_name=xio_fc
[root@node-1 ~]#

```

The right terminal window is a white box representing a file browser titled '4_XtremIO_Cinder_Config_Commands - ...'. It contains a single file named 'cinder.conf.xtremio' with the following content:

```

# Please copy and paste commands below into SSH Co
# one block at the time
cp /tmp/xtremio/cinder.conf.xtremio /etc/cinder/cinder.c
chown cinder:cinder /etc/cinder/cinder.conf
ls -l /etc/cinder

service openstack-cinder-api restart
service openstack-cinder-volume restart
1

source /root/adminrc
2
cinder type-create XtremIO-iSCSI
cinder type-create XtremIO-FC
cinder type-key XtremIO-iSCSI set volume_backend_name=xio_
cinder type-key XtremIO-FC set volume_backend_name=xio_

cinder type-list
cinder extra-specs-list

# Please open the OpenStack Horizon UI now!

```

The terminal window has a status bar at the bottom showing '100%' and zoom controls.

4.5 Reviewing XtremIO Volume Types

1. Please copy and paste the last command block into SSH Terminal window
2. Review the newly created XtremIO Volume Types. Note: you may see other volume types created earlier in the previous labs
3. Review the extra specs assigned to these new Volume Types.
4. That's it - the XtremIO Cinder driver configuration is now complete! Please close the SSH Terminal window (confirm exit as necessary)
5. Please close the Config commands file.

root@node-1:~# cinder type-list

ID	Name
80428ee8-f435-493a-b0d3-92d85539b02e	XtremIO-FC
be56fa6e-112a-4469-a132-98b433087d3d	XtremIO-iSCSI

[root@node-1:~# cinder extra-specs-list

ID	Name
80428ee8-f435-493a-b0d3-92d85539b02e	XtremIO-FC
be56fa6e-112a-4469-a132-98b433087d3d	XtremIO-iSCSI

Please copy and paste commands below into SSH Co
one block at the time

```

cp /tmp/xtremio/cinder.conf.xtremio /etc/cinder/cinder.conf
chown cinder:cinder /etc/cinder/cinder.conf
ls -l /etc/cinder

service openstack-cinder-api restart
service openstack-cinder-volume restart

source /root/adminrc

cinder type-create XtremIO-iSCSI
cinder type-create XtremIO-FC
cinder type-key XtremIO-iSCSI set volume_backend_name=xio_i
cinder type-key XtremIO-FC set volume_backend_name=xio_f

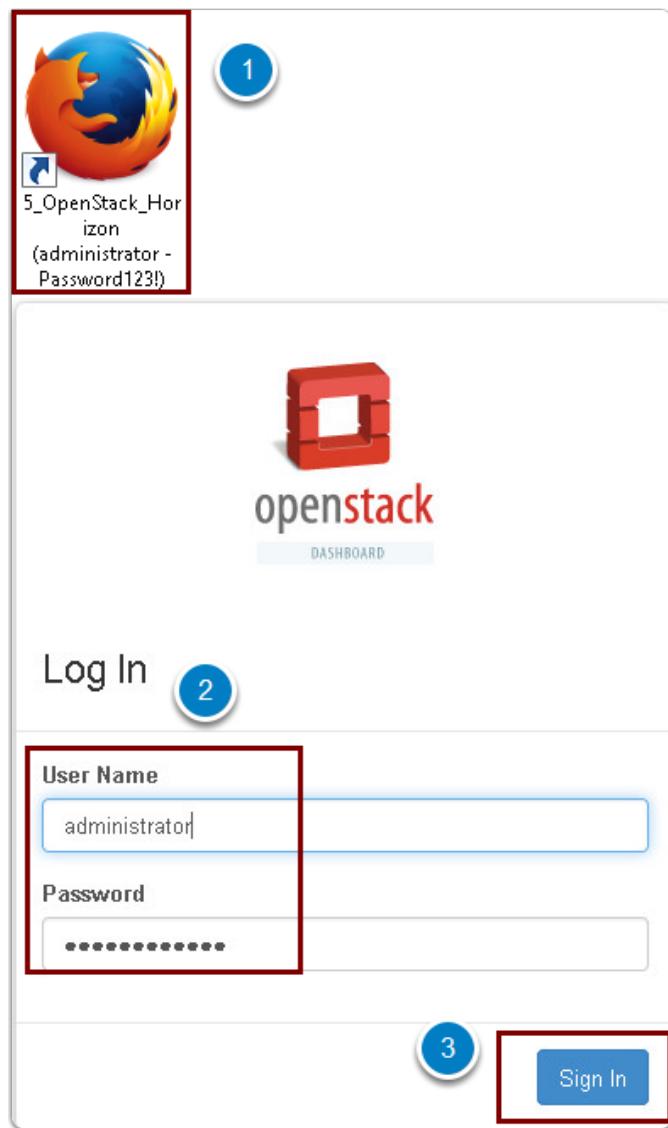
```

cinder type-list
cinder extra-specs-list

Please open the OpenStack Horizon UI now!

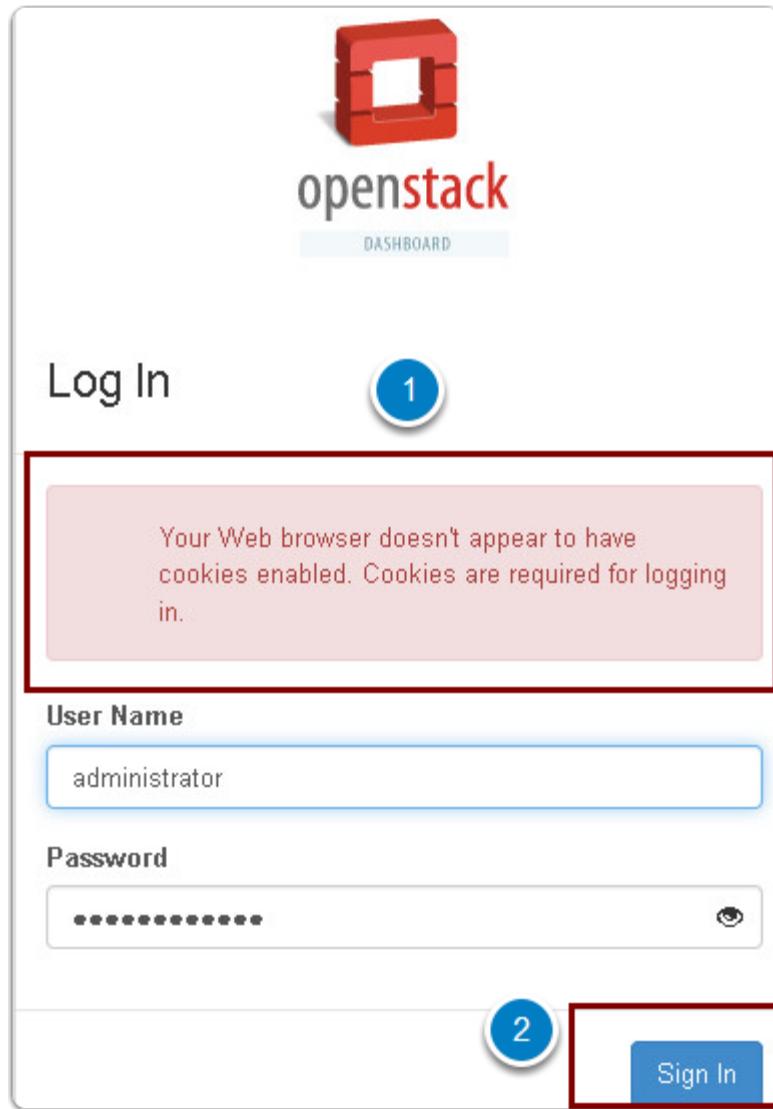
5. Using OpenStack Horizon UI

1. Return to the Lab-2-XtremIO-Cinder folder and launch the 5_OpenStack_Horizon(administrator - Password123!) shortcut
2. The user credentials should be cached in the browser window. If needed please type the credentials:
 - User Name: administrator
 - Password: Password123!
3. Click Sign In to continue



5.1 Browser Cookie Warning

1. Occasionally you may see a browser cookie warning
2. Please disregard it and simply click on **Sign In** again to continue



5.2 Reviewing XtremIO Volume Types

1. Please review the Dashboard for OpenStack project - **cloud**
2. Please select **Volumes** in OpenStack Horizon Admin menu on the left hand side
3. Select **Volume Types** tab

4. Review your created XtremIO volume types. Note: you may see other volume types created earlier in the previous labs
5. Please click on the Extra Specs menu to see the volume type details.

The screenshot shows the OpenStack dashboard with the 'System' project selected in the sidebar. The main area displays usage statistics and a list of volume types.

Usage Summary:

- From: 2015-03-01
- To: 2015-03-14
- Submit
- The date should be in YYYY-mm-dd format.
- Active Instances: 1 Active RAM: 64MB This Period's VCPU-Hours: 318.03 This Period's GB-Hours: 0.00

Usage:

Project Name	VCPUs	Disk	RAM	VCPU Hours	Disk GB Hours
cloud	1	0Bytes	64MB	318.03	0.00

Volumes:

Volume Types (4) (highlighted with a red box)

Name	Associated QoS Spec	Actions
XtremIO-iSCSI		+ Create Volume Type View Extra Specs (highlighted with a red box)
XtremIO-FC		View Extra Specs (highlighted with a red box)

Displaying 2 items

5.3 Volume Type Extra Specs

1. Please review the XtremlO Volume Type extra specs - these were configuration parameters you have provided while creating this Volume Type. These parameters correspond to the XtremlO backend configuration.

Volume Type: XtremlO-iSCSI

Volume Type Extra Specs

	Key	Value	Actions
<input type="checkbox"/>	volume_backend_name	xio_iscsi	<input type="button" value="Edit"/> <input type="button" value="Delete"/>

Displaying 1 item

A screenshot of a web-based interface titled "Volume Type: XtremlO-iSCSI". Below it is a section titled "Volume Type Extra Specs". A table lists one extra spec: "volume_backend_name" with the value "xio_iscsi". The "Key" column has a red box around it, and the "Value" column has a blue circle with the number "1" in it. There are "Create" and "Delete Extra Specs" buttons at the top right of the table area.

6. Conclusion

Congratulations! In this lab you, wearing the hat of Code Nebulous Cloud Administrator, have successfully accomplished the following:

- Deployed and configured EMC XtremlO Cinder driver to use the XtremlO array block storage as a backend
- Created Cinder volume types associated with the XtremlO storage pools

Now Code Nebulous Cloud Users can begin building and using the applications running on the Cloud-Workload instance and store its data on the persistent block volume.

This lab is now complete. Please close the following windows as you will not be using them any longer:

- Lab-2-XtremlO-Cinder folder

Please proceed to the next lab, or select other labs from the list.

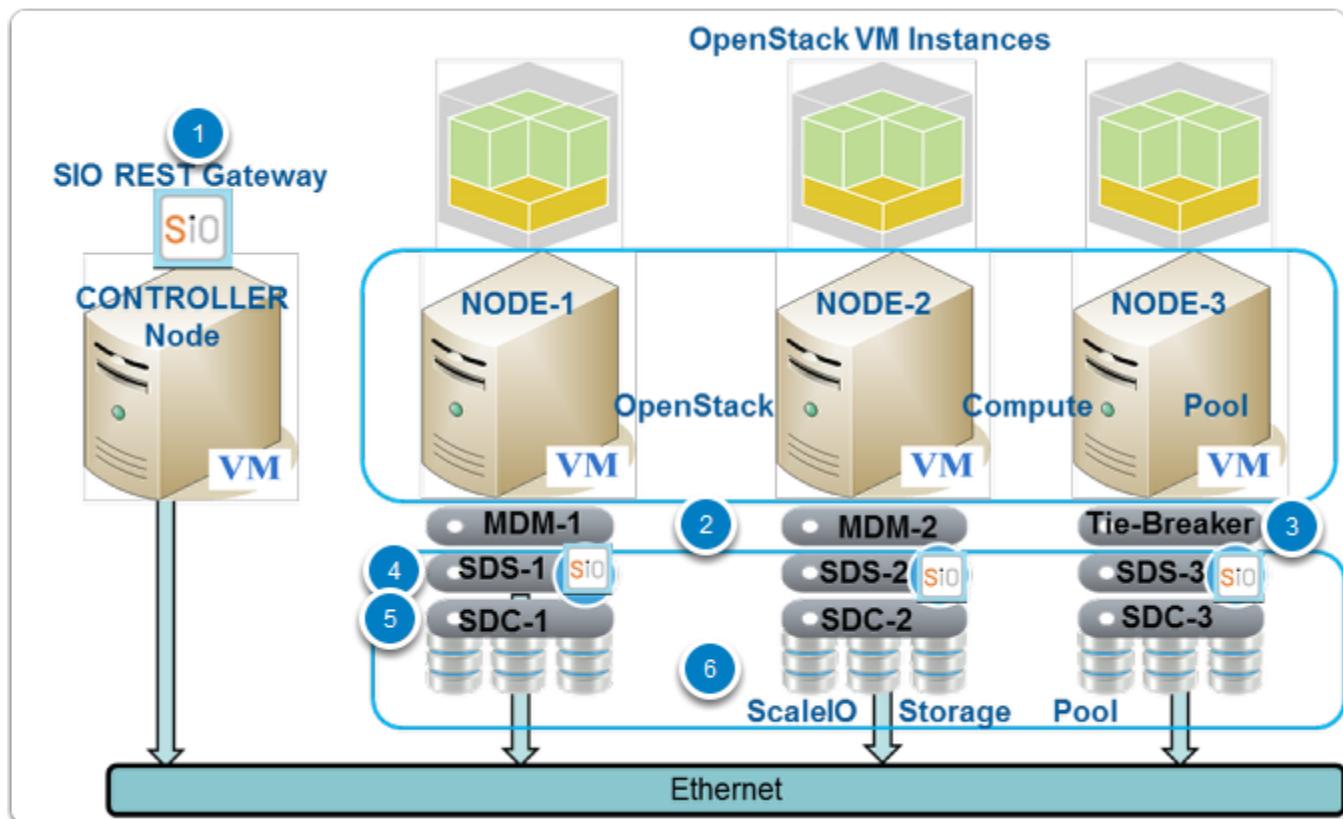
Lab 3 - Using EMC ScaleIO with Cinder Block Storage Service

Lab 3 - Using EMC ScaleIO with OpenStack Cinder Block Storage

1. ScaleIO Components

In this lab environment, a typical ScaleIO deployment architecture has been implemented that includes:

1. REST API Gateway deployed on the OpenStack Controller node. This gateway also hosts the ScaleIO Installation Manager web application - you will be using it to deploy the ScaleIO cluster components.
2. Metadata Managers (MDM) deployed on two of the OpenStack Compute nodes (The MDM's can also be deployed on the Controller nodes in the Highly-Available OpenStack environment)
3. Tie-Breaker (TB) deployed on the third OpenStack Compute node
4. Storage Data Server (SDS) deployed on each OpenStack Compute node
5. Storage Data Client (SDC) deployed on each OpenStack Compute node
6. SDC's present the local disks on Compute nodes as part of the global ScaleIO Storage Pool



2. Lab Scenario

In this lab as a Cloud Admin you will:

- Review the Cinder driver capabilities for EMC ScaleIO, including extra-specs
- Review and become familiar with the ScaleIO Cinder drivers configuration steps
- Follow the configuration steps and implement Cinder integration with EMC ScaleIO Storage backend
- Create storage volume types for OpenStack Volumes and associate the Volume Types with Storage Pools
- Practice creating new volumes for different Volume Types
- Ensure that new Volumes are created using EMC ScaleIO User Interface (UI)
- Verify that new Volumes can be attached to an OpenStack VM instance
- Ensure that new Storage Volume can be mounted and formatted by logging into VM instance using SSH
- Practice performing volume life-cycle management tasks (extend, dedup, compress, etc.)

3. Integration with OpenStack

The ScaleIO elastic storage solution includes a Cinder driver, which interfaces between ScaleIO and OpenStack, and presents volumes to OpenStack as block devices which then become available to Cinder block storage.

It also includes an OpenStack Nova driver, for handling compute and instance volume related operations.

In an OpenStack Cloud, the ScaleIO components SDS and SDC can be installed on the Compute Nodes to present their commodity disc as a single (or multiple) elastic storage pool.

The ScaleIO driver executes the volume operations by communicating with the backend ScaleIO MDM through the ScaleIO REST Gateway. This REST API Gateway can be installed on one of the OpenStack Controller nodes, while MDM's and Tie-Breaker can be installed on other Controller nodes.

3.1 Supported Operations

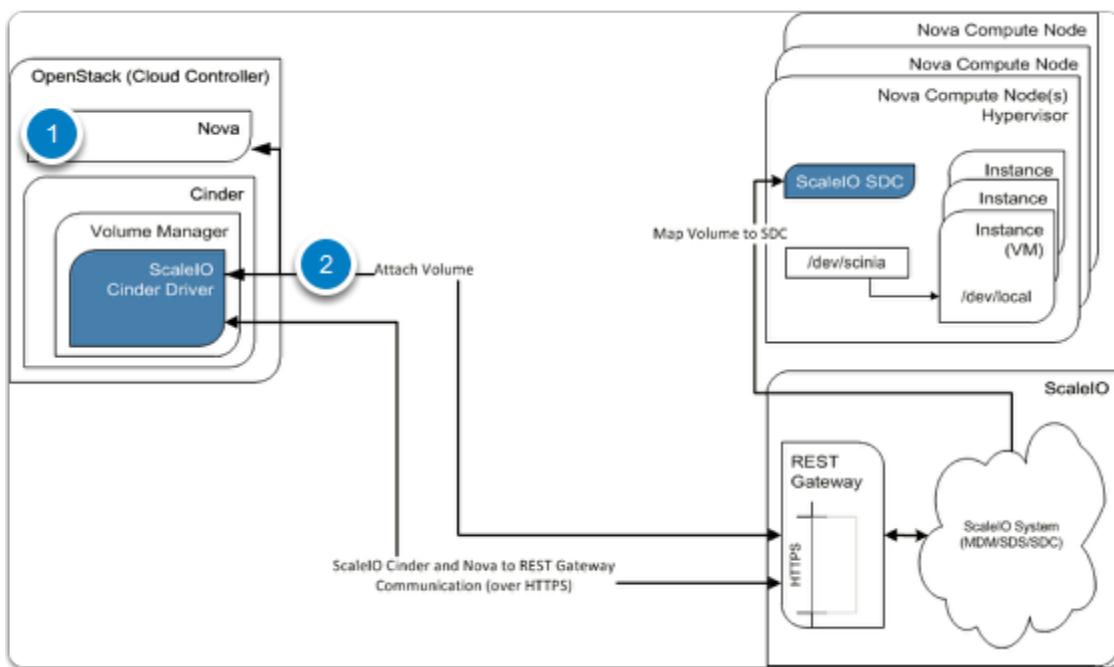
The ScaleIO Cinder driver supports the following operations:

- Create volume
- Delete volume
- Attach volume
- Detach volume
- Create snapshot
- Delete snapshot
- Create volume from snapshot
- Create cloned volume
- Copy image to volume
- Copy volume to image
- Extend volume

3.2 Volume Mapping Flow

The following figure shows a typical flow for mapping a volume. The OpenStack flow is as follows:

1. Create volume (Cinder operation)—calls the ScaleIO add_volume command
2. Attach volume to instance—initiated by Nova compute manager by calling attach_volume
3. Triggers the Cinder driver to perform the volume attach operation at the storage side (initialize_connection)—not needed in the ScaleIO scenario.
4. Nova LibvirtDriver attach_volume is called, which calls the connect_volume of the specific LibvirtVolumeDriver (determined by the volume_driver_type that the Cinder driver returns in the connection information) to finish the attach operation on the Hypervisor side. This calls the ScaleIO map_volume_to_SDC command and then verifies that the volume is exposed to the hypervisor.
5. After the volume is attached to the compute node, it is passed through to the VM by OpenStack.



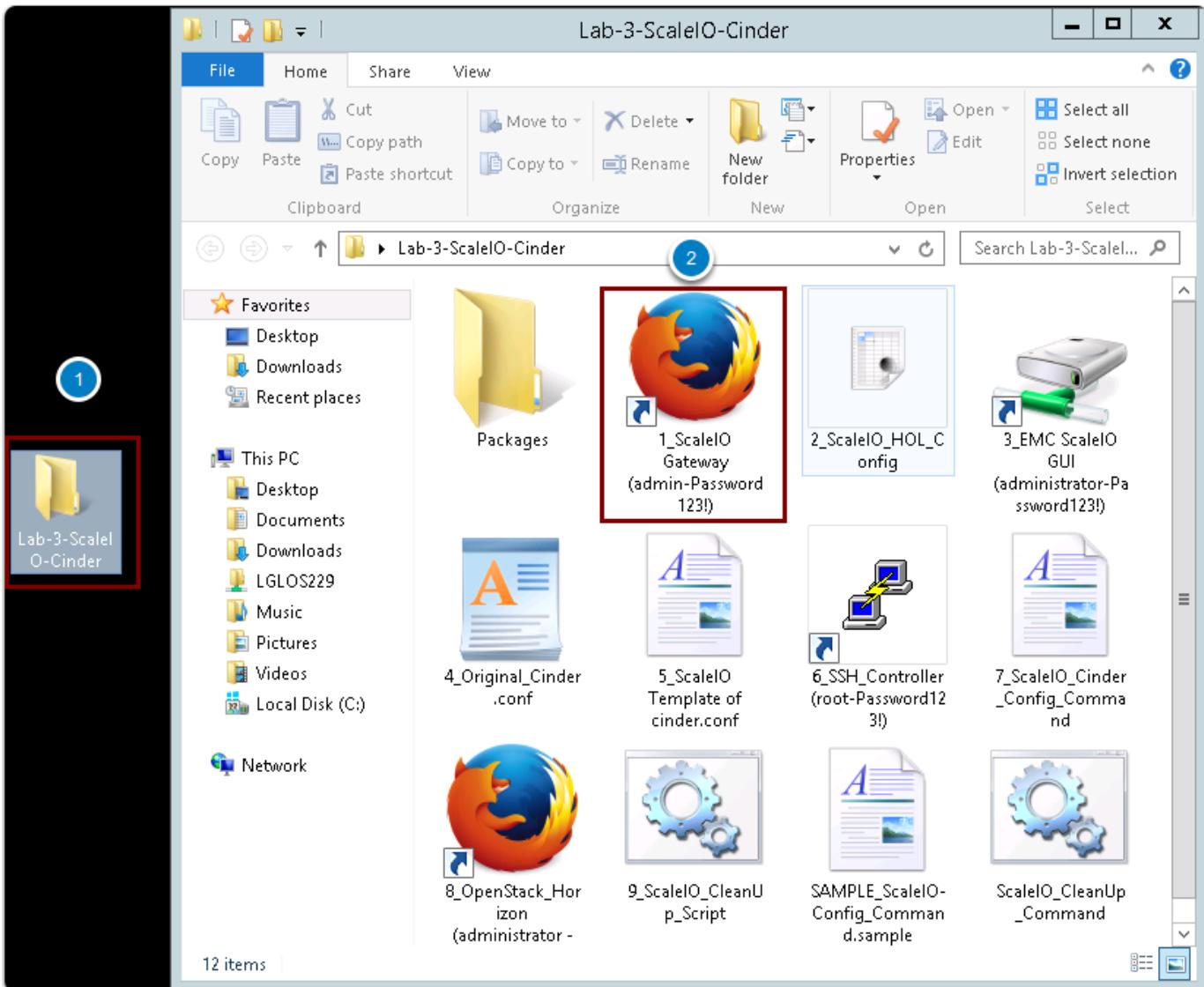
4. Installing ScaleIO on Compute Nodes

For your convenience and in the interests of time, a pre-installed ScaleIO REST API Gateway has been installed on the OpenStack Controller node.

As a Cloud Admin you will now install other ScaleIO Software Defined Storage components on the Compute nodes of the OpenStack Cloud.

1. Please open the Lab-3-ScaleIO-Cinder folder on your Desktop

2. Select the shortcut in the Lab-3-ScaleIO-Cinder folder called 1_ScaleIO Gateway

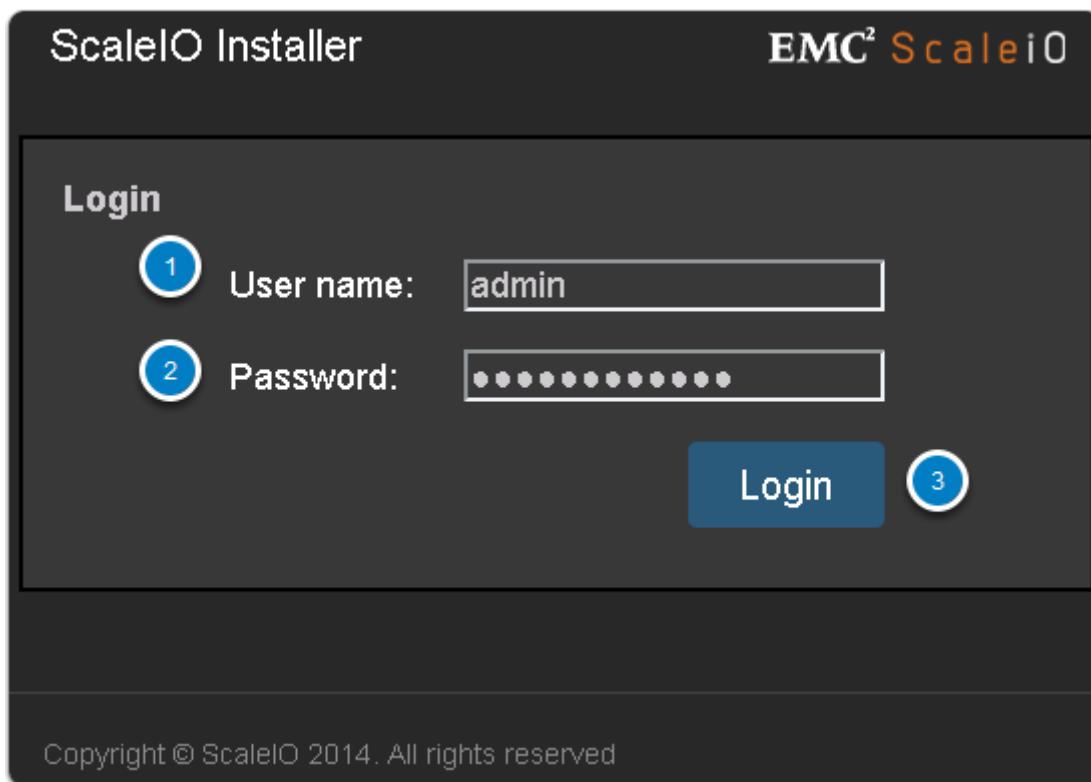


4.1 Logging into ScaleIO Installation Manager (IM)

Please login into ScaleIO Installation Manager. The user credentials are cached in your browser.

However, just in case you need to type the credentials in, please use:

1. User name: *admin*
2. Password: *Password123!*
3. Click Login to continue



4.2 Getting Started

While the ScaleIO installation can be performed manually using CLI, or automated or scripted for deployment in the production cloud environment, for the purpose of this lab you will begin the installation using the ScaleIO Installation Manager.

1. Please click on "Get Started"

The screenshot shows the EMC ScaleIO Installation Manager interface. At the top, there are five navigation buttons: Home, Packages, Install, Monitor, and Maintain. To the right of these is the EMC² ScaleIO logo. Below the navigation bar, the title "Welcome to ScaleIO" is displayed. A message indicates that users can choose to install a new ScaleIO system using the Installation Manager (version 1.31.258.2). Two main installation paths are presented: "Install using this web interface:" and "Install using the command line:". The "Install using this web interface:" section contains three numbered steps: 1. Upload the [installation packages](#) to the Installation Manager. 2. [Upload](#) a CSV topology file, review it, and initiate the installation. 3. [Monitor and approve](#) the installation progress. Step 3 is circled with a blue circle containing the number "1". Below these steps is a large "Get Started >" button, which is also circled with a red box. The "Install using the command line:" section contains two numbered steps: 1. [Download the CLI](#). 2. Run:
`java -jar install-CLI.jar`
Below this section is a "Download CLI" button with a download icon.

4.3 Installation Packages

1. Please review all pre-loaded packages - these are the ScaleIO components to be installed on the OpenStack Compute nodes
2. Then click Proceed to Install.

Packages uploaded to Installation Manager

Type	OS	Linux flavor	Version	Latest	Size	F
LIA	Linux	RHEL6	1.31-1277.3	✓	1.32 MB	EI
Callhome	Linux	RHEL6	1.31-1277.3	✓	21.16 kB	EI
SDC	Linux	RHEL6	1.31-1277.3	✓	2.97 MB	EI
SDS	Linux	RHEL6	1.31-1277.3	✓	1.64 MB	EI
TB	Linux	RHEL6	1.31-1277.3	✓	1.61 MB	EI
MDM	Linux	RHEL6	1.31-1277.3	✓	4.27 MB	EI

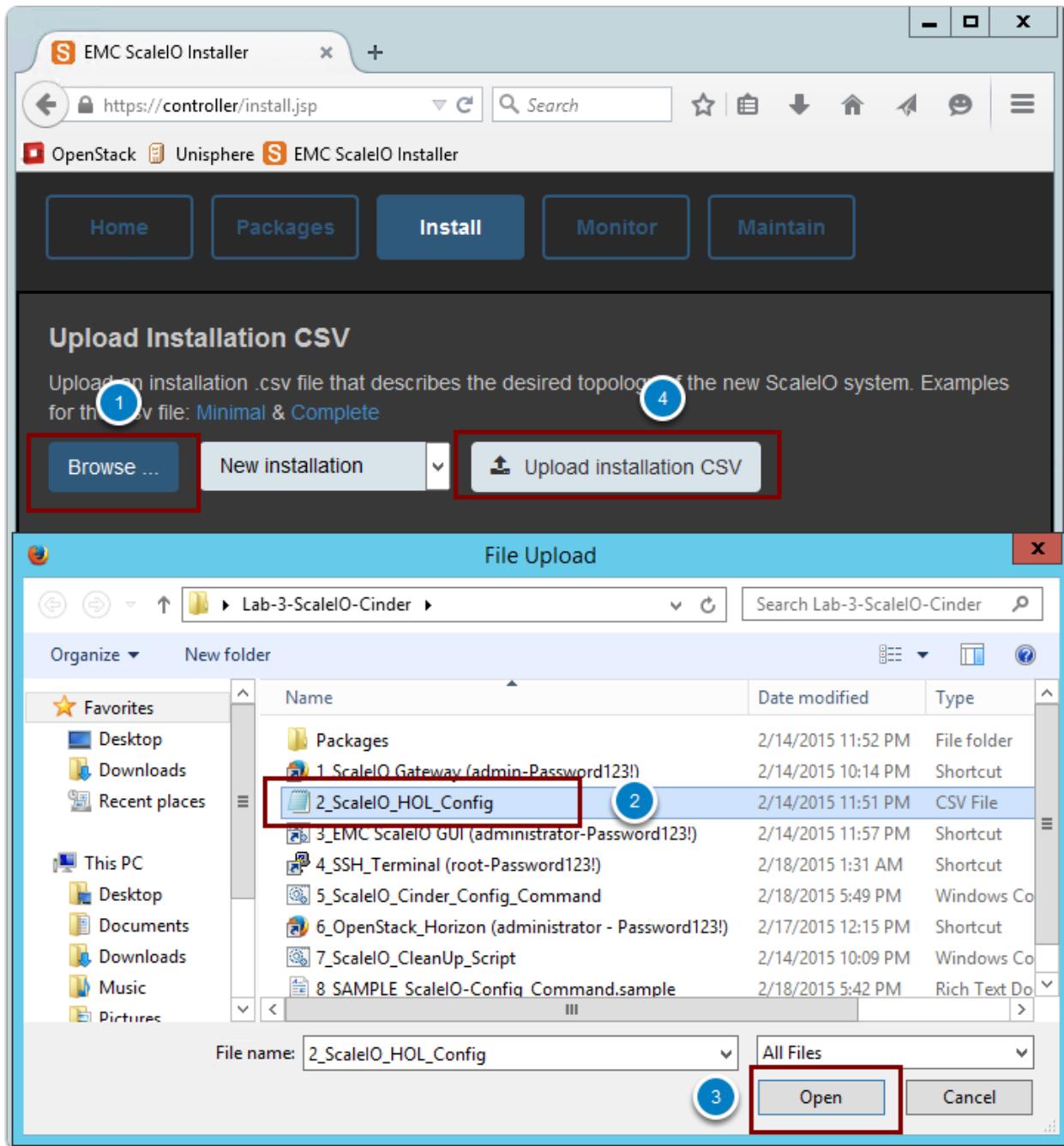
Show only latest packages

Proceed to Install >

4.4 Using Configuration Template

While you can provide all necessary information about Compute nodes, such as the IP addresses and login credentials, the ScaleIO Installation Manager can also use a CSV file that contains the information about the intended MDM, SDS and SDC deployment.

1. Click on **Browse** button to locate the installation.csv file
2. Select **2_ScaleIO_HOL_Config.csv**
3. Click **Open**
4. Then click on **Upload installation CSV** to continue;



4.5 Review Configuration

1. Please type in MDM and LIA password as shown in the boxes: *Password123!* - Please make sure that you type the password twice for each credential.
2. Make sure to uncheck the Call Home checkbox - in the interest of time you will not be installing the Call Home package.

3. Please review and verify the installation topology of MDM's and Tie-Breaker on the provided Compute nodes
4. Review and verify the installation topology of the SDS modules
5. Review and verify the installation topology of the SDC modules
6. Click on Start Installation to continue

Review Installation Configuration

The CSV file was successfully processed

Review the configuration, enter passwords and other options, then start the installation.

Credentials Configuration (1)

MDM Password	*****	*****	?
LIA Password	*****	*****	?

Advanced Set advanced options

Syslog Configure the sending of syslog events (optional)

Call Home (2) Configure call home (optional)

Topology

MDMs and Tie-Breaker (3)

Role	Management IP(s)	MDM IP(s)	OS	User name	Password
Primary	10.0.0.3	10.0.0.3	Linux	root	show
Secondary	10.0.0.4	10.0.0.4	Linux	root	show
Tie-Breaker	N/A	10.0.0.5	Linux	root	show

SDS List (4)

IP	All IPs	OS	PD	SP	Devices	Names	User name	Password
10.0.0.5	10.0.0.5	Linux	cloud	pool1	/dev/sdb	sdbDevice	root	show
10.0.0.4	10.0.0.4	Linux	cloud	pool1	/dev/sdb	sdbDevice	root	show
10.0.0.3	10.0.0.3	Linux	cloud	pool1	/dev/sdb	sdbDevice	root	show

Page 1 of 1 View 1 - 3 of 3

SDC List (5)

IP	OS	User name	Password
10.0.0.5	Linux	root	show
10.0.0.4	Linux	root	show
10.0.0.3	Linux	root	show

Page 1 of 1 View 1 - 3 of 3

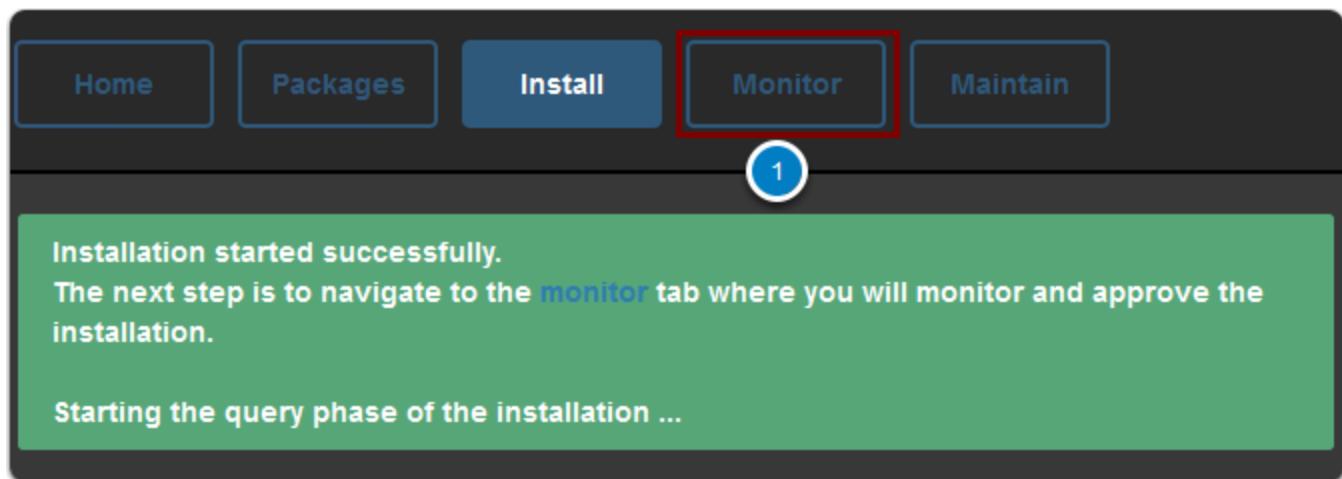
(6) **Start Installation >** - OR - *** Upload a different CSV**

5. ScaleIO Installation Process

The installation process will take you through verification of the compute nodes, uploading the installation packages and configuring the ScaleIO cluster.

5.1 Monitoring the Installation

Click on **Monitor** tab to observe the installation process and approve the installation phases..



5.2 Query Phase

1. Please review the **Query Phase** - verify that all tasks have completed successfully.
2. Confirm the completion of the **Query phase** by closing the notification window

The screenshot shows the EMC ScaleIO interface. At the top, there are two tabs: "Packages" and "Install". The "Install" tab is selected. A prominent notification box is displayed in the center, stating "Query phase completed" with a blue information icon. Below this, it says "All 4 commands completed successfully." To the right of the notification are two circular icons: one with a red border and a white "X", and another with a blue border and the number "2". In the background, under the heading "Operation Progress", it says "Server: install - query pha". Below this, a table titled "Progress of this phase" lists four completed tasks. At the bottom of the interface, there are checkboxes for "Auto Refresh", "Hide Completed Successfully", and "Show All Phases".

Phase	IP	Command	Status	Start time	Details
1: Query	Installation Ma...	validate and orchestrate new c...	✓ completed	less than a minut	Details
1: Query	10.0.0.5	validate node	✓ completed	less than a minut	Details
1: Query	10.0.0.4	validate node	✓ completed	less than a minut	Details
1: Query	10.0.0.3	validate node	✓ completed	less than a minut	Details

Progress of this phase: completed: 4, pending: 0, aborted: 0, failed: 0 (3:48:57 AM)

Phase IP Command Status Start time Details

1: Query Installation Ma... validate and orchestrate new c... ✓ completed less than a minut [Details](#)

1: Query 10.0.0.5 validate node ✓ completed less than a minut [Details](#)

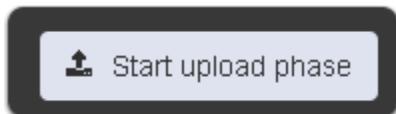
1: Query 10.0.0.4 validate node ✓ completed less than a minut [Details](#)

1: Query 10.0.0.3 validate node ✓ completed less than a minut [Details](#)

Auto Refresh Hide Completed Successfully Show All Phases

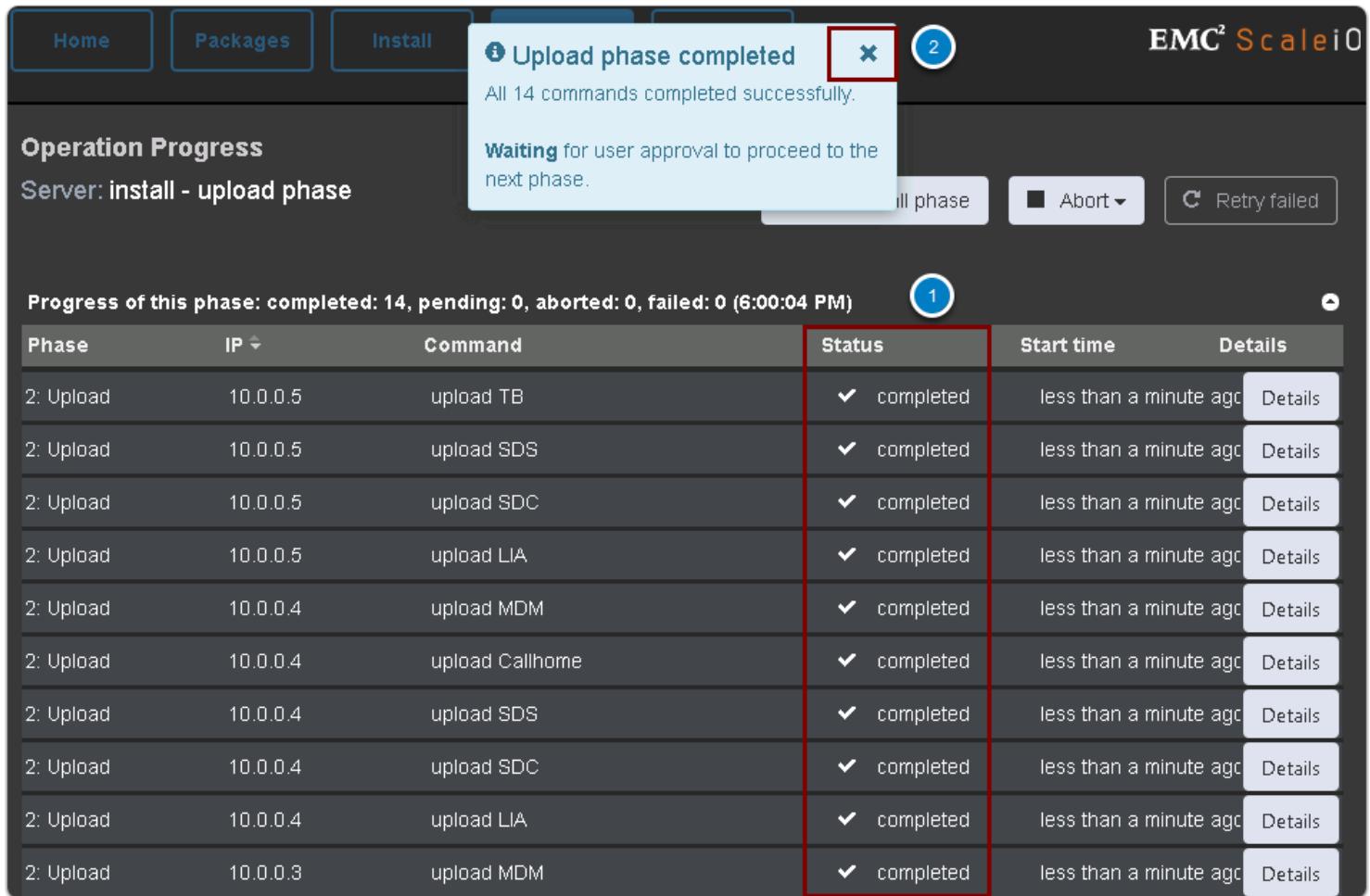
5.3 Starting Upload Phase

Please start the Upload phase by selecting the



5.4 Upload Phase

1. Please review and verify successful completion of the **Upload** phase
2. Close the notification popup window



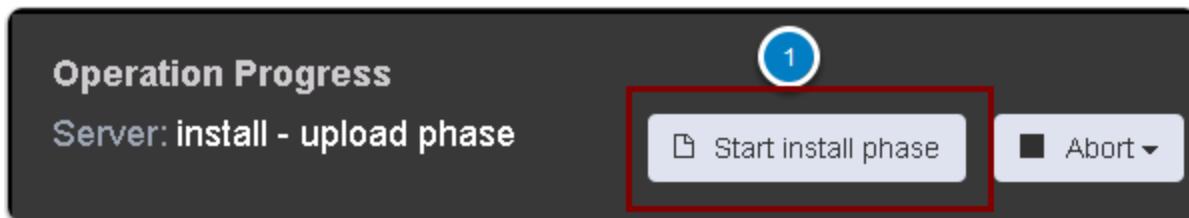
The screenshot shows the EMC ScaleIO interface with the following details:

- Top Navigation:** Home, Packages, Install, EMC² ScaleIO
- Notification Pop-up:**
 - Message:** Upload phase completed (with an info icon)
 - Text:** All 14 commands completed successfully.
 - Buttons:** X (close), 2 (dismiss)
- Operation Progress Section:**
 - Title:** Operation Progress
 - Text:** Server: install - upload phase
 - Status:** Waiting for user approval to proceed to the next phase.
 - Buttons:** Abort, Retry failed
- Table:** Progress of this phase

Phase	IP	Command	Status	Start time	Details
2: Upload	10.0.0.5	upload TB	✓ completed	less than a minute ago	Details
2: Upload	10.0.0.5	upload SDS	✓ completed	less than a minute ago	Details
2: Upload	10.0.0.5	upload SDC	✓ completed	less than a minute ago	Details
2: Upload	10.0.0.5	upload LIA	✓ completed	less than a minute ago	Details
2: Upload	10.0.0.4	upload MDM	✓ completed	less than a minute ago	Details
2: Upload	10.0.0.4	upload Callhome	✓ completed	less than a minute ago	Details
2: Upload	10.0.0.4	upload SDS	✓ completed	less than a minute ago	Details
2: Upload	10.0.0.4	upload SDC	✓ completed	less than a minute ago	Details
2: Upload	10.0.0.4	upload LIA	✓ completed	less than a minute ago	Details
2: Upload	10.0.0.3	upload MDM	✓ completed	less than a minute ago	Details

5.5 Starting Install Phase

1. Please start the **Install** phase by clicking on the button.



The screenshot shows the EMC ScaleIO interface with the following details:

- Operation Progress Section:**
 - Title:** Operation Progress
 - Text:** Server: install - upload phase
 - Buttons:** Start install phase (highlighted with a red box), Abort

5.6 Install Phase

1. Please review and verify successful completion of the **Install** phase
2. Close the notification popup window

The screenshot shows the EMC ScaleIO interface. At the top, there are three tabs: Home, Packages, and Install. The Install tab is selected. In the center, a notification box displays the message: "Install phase completed" with an info icon, and "All 14 commands completed successfully." A red box highlights the close button of this notification box. To the right of the notification, a blue circle with the number "2" indicates pending actions. Below the notification, the "Operation Progress" section shows the status of the "install - install phase" for a server at IP 10.0.0.5. A tooltip "Waiting for user approval to proceed to the next phase." is shown over the progress bar. A red box highlights the progress bar. On the right side of the progress bar, there are buttons for "Abort" and "Retry failed". Below the progress bar, a table lists the completed commands:

Phase	IP	Command	Status	Start time	Details
3: Install	10.0.0.5	install TB	✓ completed	less than a minute ago	Details
3: Install	10.0.0.5	install SDS	✓ completed	less than a minute ago	Details
3: Install	10.0.0.5	install SDC	✓ completed	less than a minute ago	Details
3: Install	10.0.0.5	install LIA	✓ completed	less than a minute ago	Details
3: Install	10.0.0.4	install MDM	✓ completed	less than a minute ago	Details
3: Install	10.0.0.4	install Callhome	✓ completed	less than a minute ago	Details
3: Install	10.0.0.4	install SDS	✓ completed	less than a minute ago	Details
3: Install	10.0.0.4	install SDC	✓ completed	less than a minute ago	Details
3: Install	10.0.0.4	install LIA	✓ completed	less than a minute ago	Details
3: Install	10.0.0.3	install MDM	✓ completed	less than a minute ago	Details

5.7 Starting Configure Phase

1. Please start the Configure phase by clicking on the button

The screenshot shows the "Operation Progress" section for the "install - install phase" on a server at IP 10.0.0.5. A red box highlights the "Start configure phase" button, which has a gear icon and the text "Start configure phase". A blue circle with the number "1" is positioned above the button.

5.8 Completing Configuration Phase

1. Please review the completion status for each configuration task
2. Confirm the completion of the Configure Phase and close the popup notification window
3. Mark Install Operation Complete to proceed to the maintenance of the ScaleIO Cluster

The screenshot shows the EMC vLab interface. At the top, a blue header bar contains the EMC² vLAB logo. Below the header, there is a green progress bar with the text "Progress for all phases: completed: 45, pending: 0, aborted: 0, failed: 0 (10:20:31 PM)". A blue circular button labeled "1" is positioned next to the progress bar.

A modal dialog box is open, titled "Configure phase completed" with a red-bordered "X" icon. It displays the message "All 13 commands completed successfully." and a green checkmark icon followed by the text "Completed Install Operation". Below this, it lists statistics: "Completed: 45", "Aborted: 0", and "Failed: 0". A blue circular button labeled "2" is positioned next to the dialog. A blue circular button labeled "3" is positioned below the dialog, pointing to a blue button labeled "Mark operation completed >". This "Mark operation completed" button is highlighted with a red rectangle.

The main content area is a table showing the progress of various tasks. The table has columns: Phase, IP, Command, Status, Start time, and Details. The rows are:

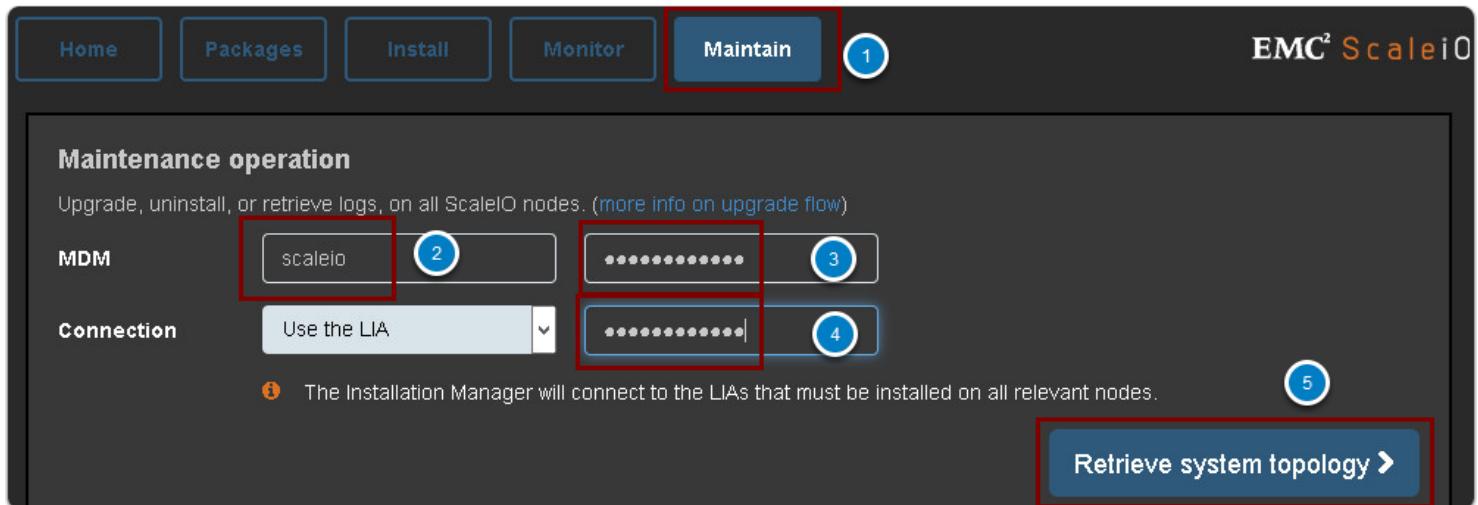
Phase	IP	Command	Status	Start time	Details
1: Query	Installation Manager	validate and orchestrate new commands f...	✓ completed	2 minutes ago	Details
1: Query	10.0.0.5	validate node	✓ completed	2 minutes ago	Details
2: Upload	10.0.0.5	upload TB	✓ completed	2 minutes ago	Details
2: Upload	10.0.0.5	upload SDS	✓ completed	2 minutes ago	Details
2: Upload	10.0.0.5	upload SDC	✓ completed	2 minutes ago	Details
2: Upload	10.0.0.5	upload LIA	✓ completed	2 minutes ago	Details
3: Install	10.0.0.5	install TB	✓ completed	2 minutes ago	Details
3: Install	10.0.0.5	install SDS	✓ completed	2 minutes ago	Details

5.9 Maintain Installed ScaleIO Cluster

You will now review the topology of your newly installed ScaleIO cluster.

1. Please click the **Maintain** button
2. Provide the name or IP Address of the MDM - you have created a DNS name for the ScaleIO MDM: **scaleio**
3. Please enter password for MDM - it's the same password you provided while importing the ScaleIO CSV configuration file: **Password123!**
4. Please enter password for LIA connection - same password as above: **Password123!**

5. Click Retrieve system topology to continue



5.10 ScaleIO Cluster Configuration

Please review the ScaleIO cluster topology:

1. The Compute nodes where ScaleIO components are installed (You are seeing the IP addresses assigned to the Compute nodes interfaces connected to the separate Storage Network)
2. Roles and status of the ScaleIO services, i.e. MDM/TB, SDS and SDC.

This is it! You have completed the installation of the **ScaleIO component services** on the **OpenStack Compute nodes**. You will now use the ScaleIO User Interface to review the storage configuration and capacity available in our new ScaleIO Cluster.

Please minimize your browser window.

The screenshot shows a web-based interface for managing a ScaleIO cluster. At the top, a green header bar displays the message "The system topology was successfully retrieved." Below this, there are two numbered circular callouts: "1" points to a link for upgrading, uninstalling, or retrieving logs for all ScaleIO nodes; "2" points to a link for more info on the upgrade flow. The main content area is titled "Hosts List" and contains a table with three columns: IP, MDM/TB, SDS, and SDC. The table lists three hosts with their respective service statuses. The entire "Hosts List" section and the "Services" column are highlighted with a red border.

IP	MDM/TB	SDS	SDC
10.0.0.5	✓	✓	✓
10.0.0.4	✓	✓	✓
10.0.0.3	✓	✓	✓

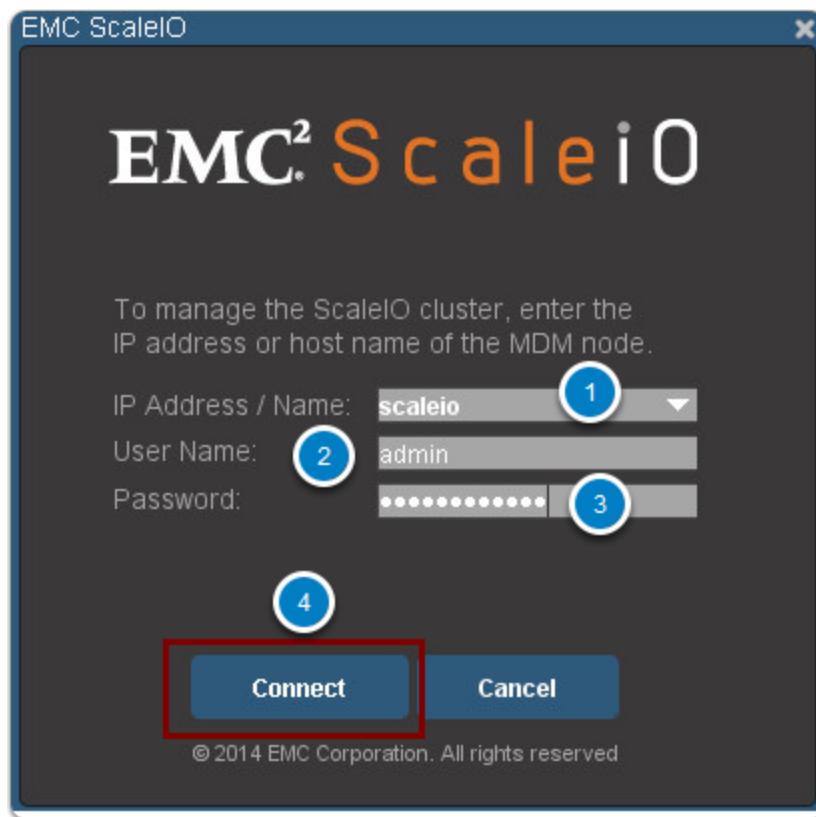
5.11 Starting ScaleIO GUI

Please return to the Lab-3-ScaleIO-Cinder folder and double-click on the shortcut called "3_EMC ScaleIO GUI"



5.12 Using ScaleIO GUI

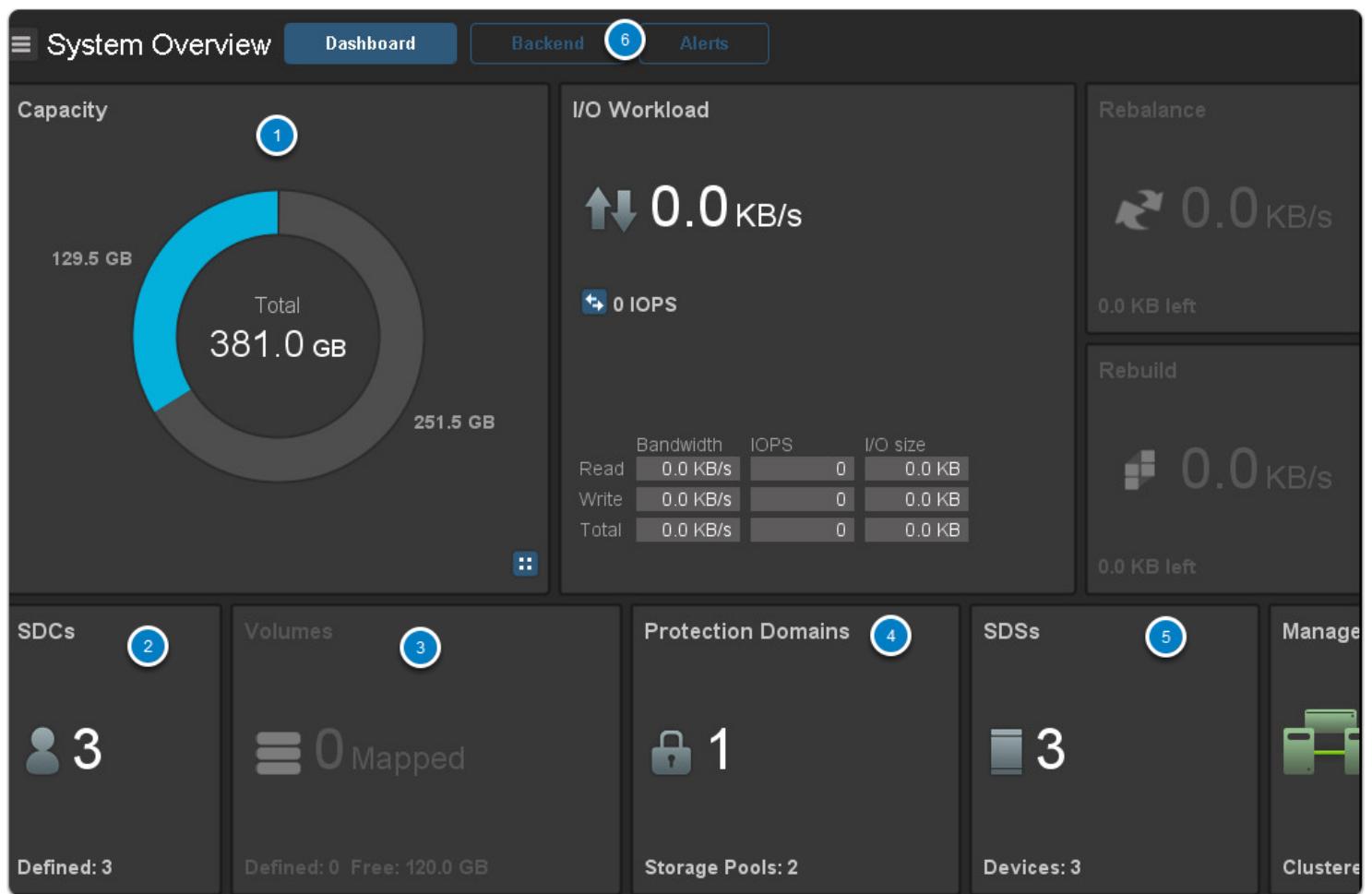
1. Please enter the MDM host name: **scaleio**
2. Enter User Name: **admin**
3. Enter Password: **Password123!**
4. Click **Connect** to login into ScaleIO GUI



5.13 Reviewing ScaleIO Cluster

Please review the details of the new ScaleIO Cluster. The ScaleIO dashboard displays:

1. Total storage capacity - 381 GB
2. Number of SDS's - 3
3. Number of Volumes created - 0. Note: you will use the ScaleIO UI to review the volumes created in the later steps.
4. ScaleIO Protection domain - 1
5. Number of SDC's - 3.
6. Next please click on the Backend button to review the details of the cluster.



5.14 ScaleIO Cluster Backends

The Backend view displays capacity overview, usage, health, rebuild/rebalance, Application I/O, overall I/O, etc.

1. Please expand the ScaleIO protection domain **cloud** by clicking on the plus sign
2. Expand each SDS and review the associated global storage pool
3. Review individual discs and available physical storage - these are spare physical discs (`/dev/sdb`) in each compute node
4. When finished, please return to the **Dashboard** view in preparation to our next steps and minimize the ScaleIO GUI.

Item	Total Capacity	Capacity In-Use	I/O Bandwidth	IOPS	Rebuild	Rebalance
System 1	381.0 GB	0.0 KB (0.0 %)	0.0 KB/s	0	0.0 KB/s	0.0 KB/s
cloud 2	381.0 GB	0.0 KB (0.0 %)	0.0 KB/s	0	0.0 KB/s	0.0 KB/s
SDS-Compute-1	127.0 GB	0.0 KB (0.0 %)	0.0 KB/s	0	0.0 KB/s	0.0 KB/s
pool1						
sdbDevice 3	127.0 GB	0.0 KB (0.0 %)	0.0 KB/s	0	0.0 KB/s	0.0 KB/s
SDS-Compute-2	127.0 GB	0.0 KB (0.0 %)	0.0 KB/s	0	0.0 KB/s	0.0 KB/s
pool1						
sdbDevice	127.0 GB	0.0 KB (0.0 %)	0.0 KB/s	0	0.0 KB/s	0.0 KB/s
SDS-Compute-3	127.0 GB	0.0 KB (0.0 %)	0.0 KB/s	0	0.0 KB/s	0.0 KB/s
pool1						
sdbDevice	127.0 GB	0.0 KB (0.0 %)	0.0 KB/s	0	0.0 KB/s	0.0 KB/s

6. ScaleIO Cinder Driver Installation

In this section of the lab you will perform installation and configuration steps that are required to enable ScaleIO storage backend for OpenStack Cinder service.

Here are the installation steps - please review these steps ONLY, do NOT perform them, as the lab environment has already been prepared for you.

- Install ScaleIO Cinder driver - available as a part of the ScaleIO software distribution from <https://support.emc.com>
- The driver installation Python script copies the volume driver and inserts volume_drivers key in the Nova configuration file `/etc/nova/nova.conf` and `/etc/cinder/cinder.conf`

- Creates a ScaleIO driver configuration file called `/etc/cinder/cinder_scaleio.config`, containing parameters for the REST Gateway, Protection Domain and Storage Pool.

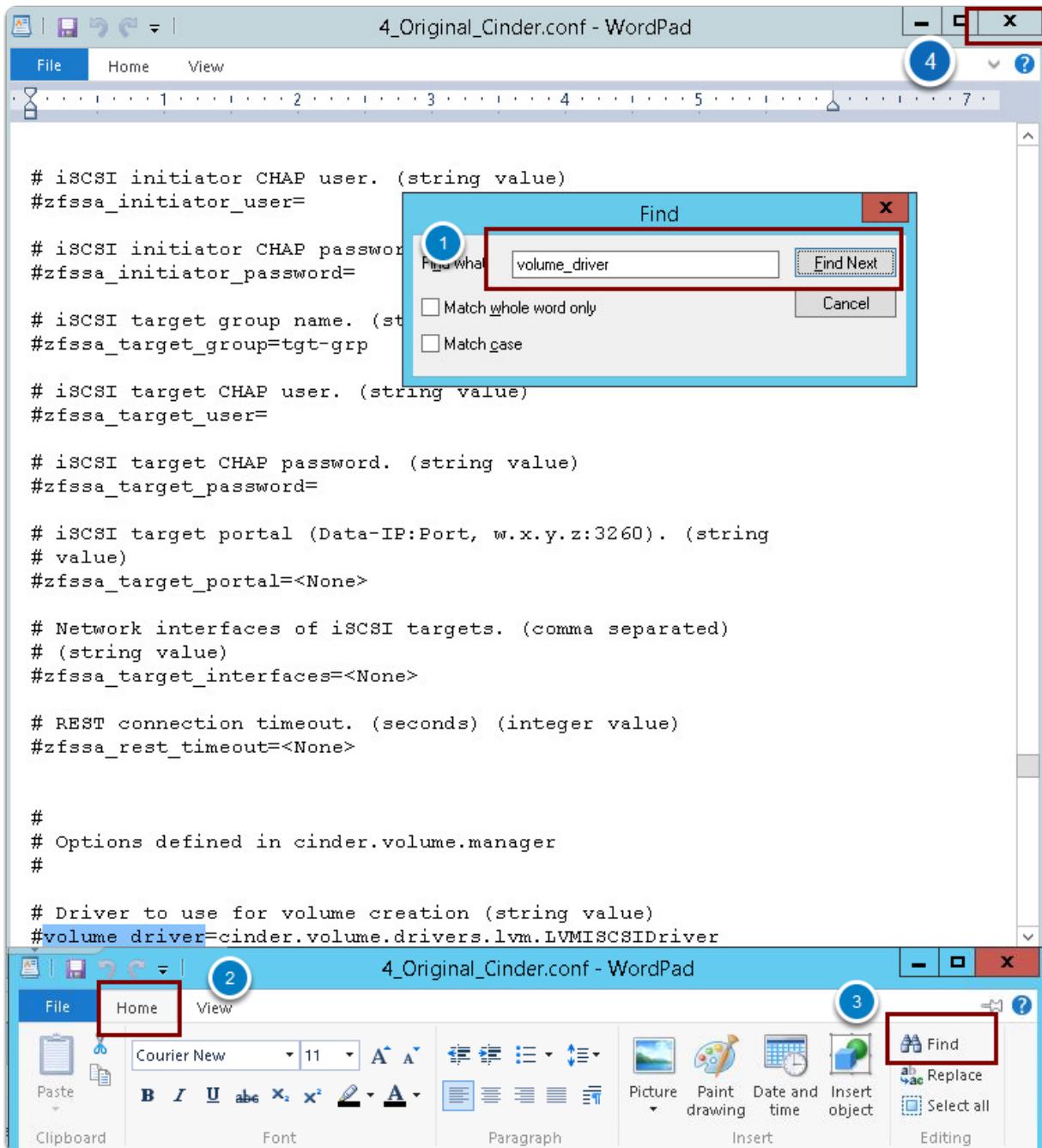
Please expand the Lab-3-ScaleIO-Cinder folder and double-click on `4_Original_Cinder.conf` file shortcut:



6.1 ScaleIO Cinder Config File

You will now review the Cinder service configuration file:

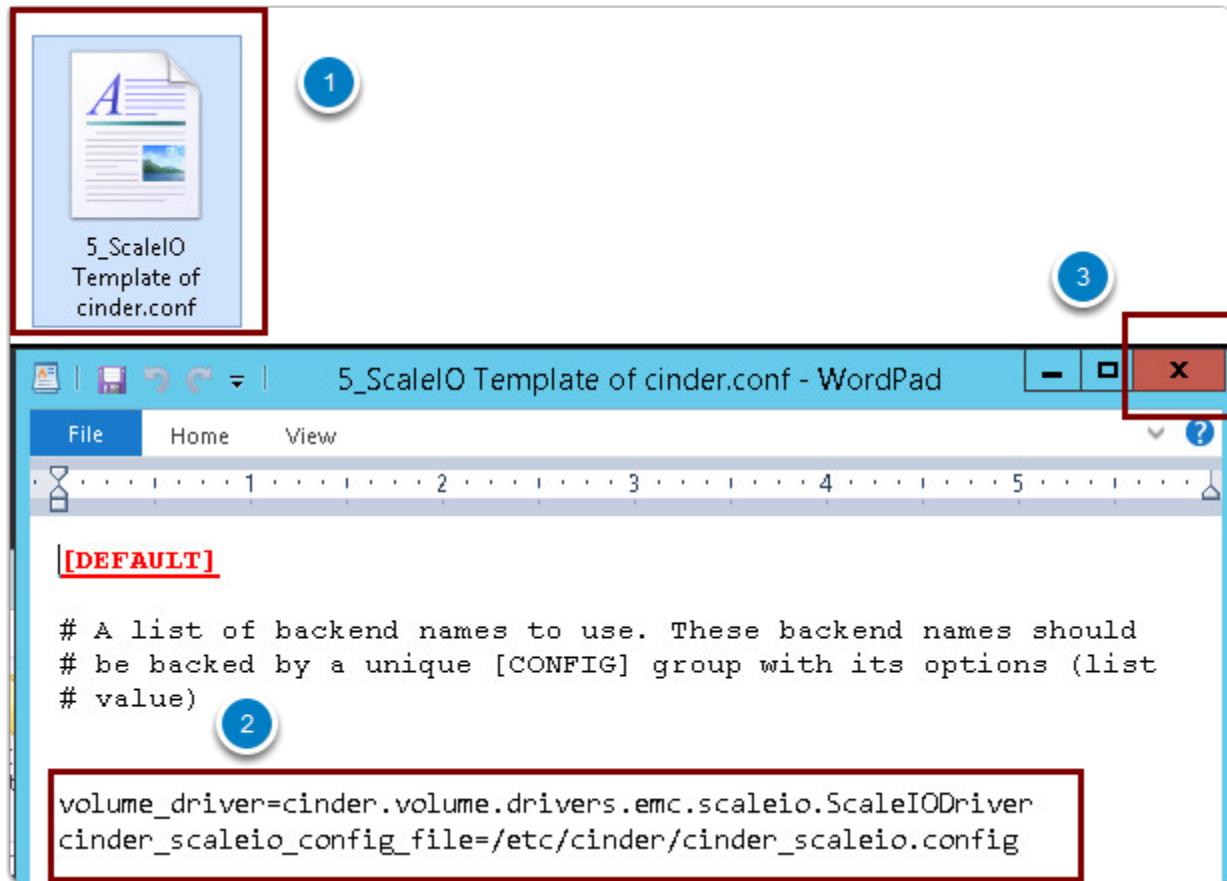
1. Please press Ctrl+F to start search and type in the search pattern **volume_driver** and click **Next** - that's where the ScaleIO Cinder driver will need to be configured.
2. If Ctrl+F doesn't work please click **Home** menu tab.
3. Then select **Find** menu option.
4. Please close the Cinder configuration file at this point - in the interests of time we have created a pre-configured file that you will later copy over this original one.



6.2 Pre-Configured Cinder Conf File

1. Please return to the Lab-3-ScaleIO-Cinder folder and open a template file we created for your review called **5_ScaleIO_Template_of_Cinder.conf**
2. Review the content of the pre-configured template of **cinder.conf** file with the **volume_driver** settings required for ScaleIO storage backend.

3. Please close the file when finished your review and return to the Lab-3-ScaleIO-Cinder folder.

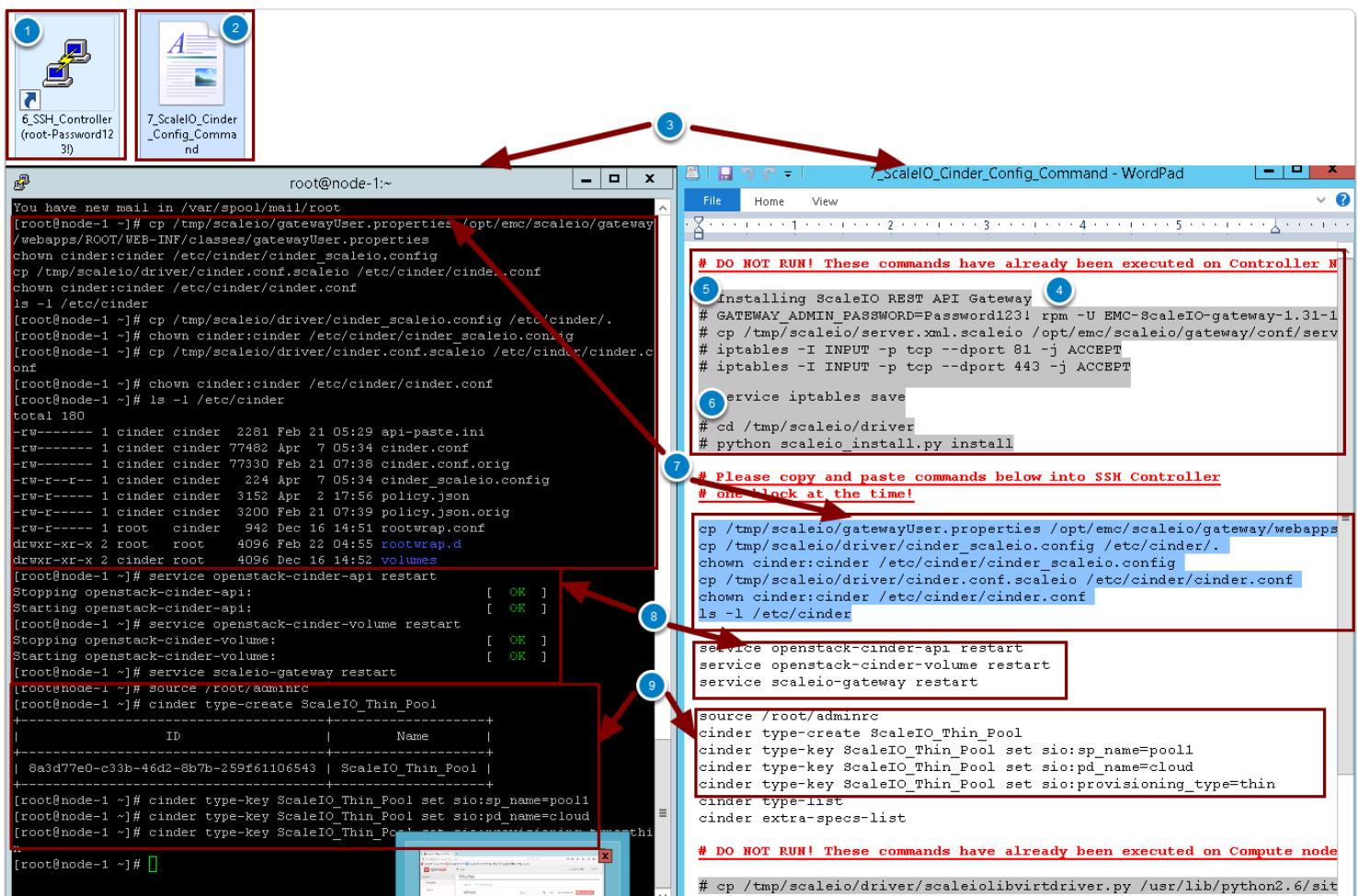


6.3 Using PuTTY SSH Utility

To save you some time in running this lab, a file has been created with the commands to be executed on the OpenStack Controller Node to install and configure ScaleIO Cinder driver.

1. Please open PuTTY shortcut to OpenStack Controller Node using shortcut called 6_SSH_Controller
2. Then open 7_ScaleIO_Config_Commands file
3. Please arrange the open SSH Terminal window and the command file side by side as shown - you will be copying the commands from the file into terminal window using right click, Copy and Paste.
4. The grey-shaded areas contain commands that are necessary for installation, but take some time to complete. Please DO NOT run these commands, we have already executed them.
5. First greyed-out block installed the ScaleIO Gateway service on the OpenStack Controller node and opened the required ports in the Linux firewall.
6. Second greyed-out block installed the ScaleIO Cinder driver on the Controller node and Nova driver on the Compute nodes

7. It is recommended to copy and paste commands into SSH Terminal window as a block. Blocks are separated by blank lines. Please copy the command block and paste into SSH Terminal window using right-click. We have created a copy of the ScaleIO gatewayUser.properties file in the /tmp directory with all necessary settings you have reviewed earlier - this file is copied over the original file. We have also created a copy of the cinder.conf file in the /tmp directory with all necessary settings you have reviewed earlier - this file is copied over the original /etc/cinder/cinder.conf file. It is also necessary to change the ownership of this file to cinder user
8. It is necessary to **restart** the ScaleIO and Cinder API and Volume services to read the new configuration files - Please copy the next command block and paste into SSH Terminal window using right-click.
9. You will now create ScaleIO Volume types and assign them to Cinder backends. Please copy the next block in the 7_ScaleIO_Cinder_Config.Commands file and paste it into SSH Terminal window using **right-click**



6.4 Creating ScaleIO Volume Types

Now that ScaleIO Cinder driver is installed and configured, you will create ScaleIO Volume Types in OpenStack Cinder service.

1. Please run the `cinder type-list` command and review the created `ScaleIO_Thin_Pool` volume type
2. Please run the `cinder extra-specs-list` command and review the extra specifications for this volume type:
 - ScaleIO Storage Pool name: `pool1`
 - ScaleIO Protections Domain name: `cloud`
 - Volume provisioning type: `thin`
3. Please minimize the **SSH terminal** window - we will now begin using the OpenStack Horizon UI.

```

source /root/adminrc
cinder type-create ScaleIO_Thin_Pool
cinder type-key ScaleIO_Thin_Pool set sio:sp_name=pool1
cinder type-key ScaleIO_Thin_Pool set sio:pd_name=cloud
cinder type-key ScaleIO_Thin_Pool set sio:provisioning_type=thin
cinder type-list
cinder extra-specs-list

```

1. A red arrow points from the 'cinder type-list' command to the output table.

```

root@node-1:~#
[root@node-1 ~]# source /root/adminrc
[root@node-1 ~]# cinder type-key ScaleIO_Thin_Pool set sio:sp_name=pool1
[root@node-1 ~]# cinder type-create ScaleIO_Thin_Pool
cinder type-key ScaleIO_Thin_Pool set sio:pd_name=cloud
cinder type-key ScaleIO_Thin_Pool set sio:provisioning_type=thin
cinder type-list
cinder extra-specs-list
+-----+-----+
| ID   |      Name      |
+-----+-----+
| 17bfccad-3ecd-42a0-8d33-0ae3d14ca560 | ScaleIO_Thin_Pool |
+-----+-----+

```

2. A red arrow points from the 'cinder extra-specs-list' command to the output table.

```

[root@node-1 ~]# cinder extra-specs-list
+-----+-----+
| ID       |      Name      |
+-----+-----+
| extra_specs |           |
+-----+-----+
| 17bfccad-3ecd-42a0-8d33-0ae3d14ca560 | ScaleIO_Thin_Pool | {u'sio:pd_name': u'cloud', u'sio:provisioning_type': u'thin', u'sio:sp_name': u'pool1'} |
+-----+-----+

```

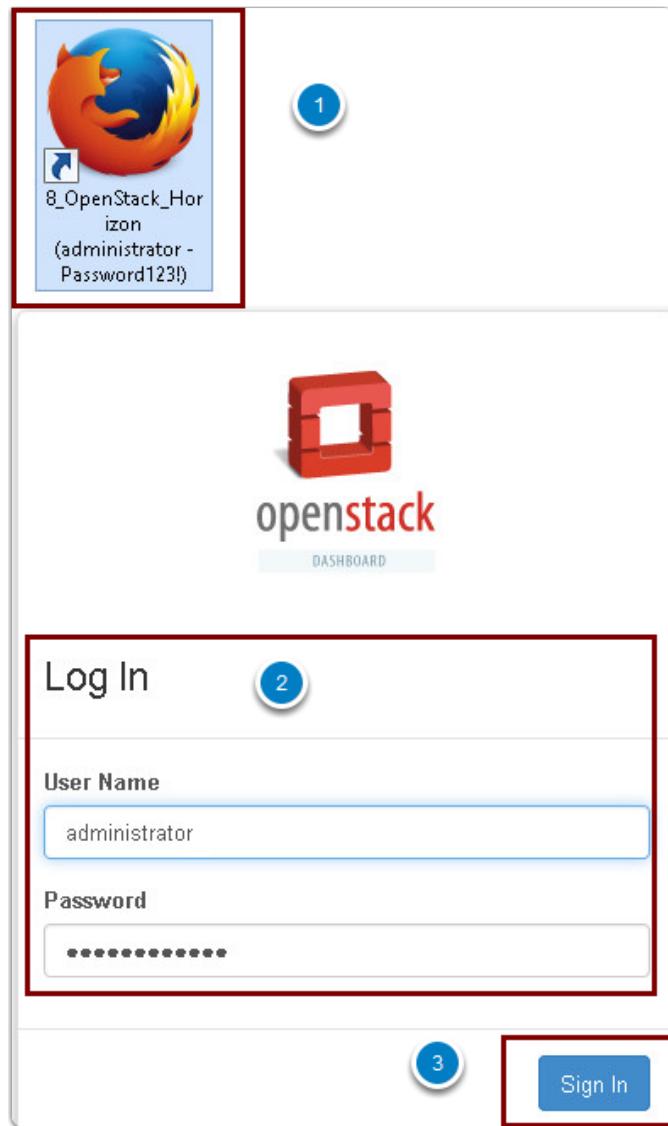
3. A red box highlights the close button in the terminal window title bar.

7. Using OpenStack Horizon UI

1. Return to the Lab-3-ScaleIO-Cinder folder and launch the 8_OpenStack_Horizon(administrator - Password123!) shortcut
2. The user credentials should be cached in the browser window. If needed please type the credentials:

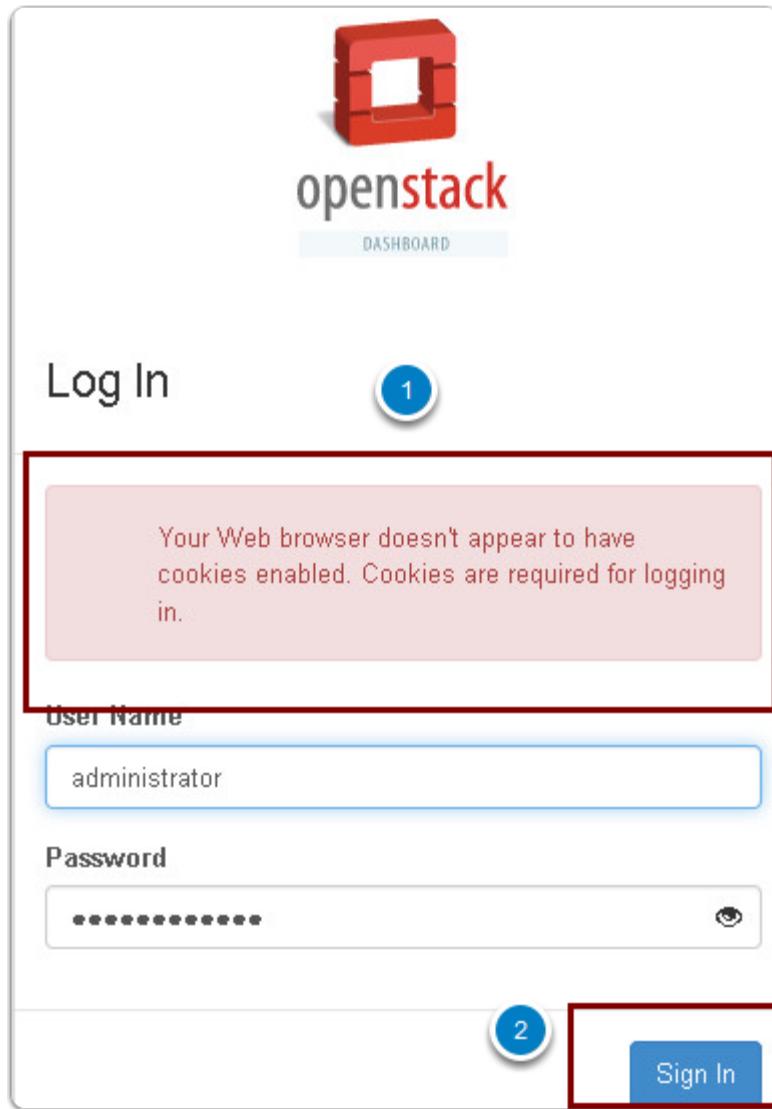
- User Name: *administrator*
- Password: *Password123!*

3. Click **Sign In** to continue



7.1 Browser Cookie Warning

1. Occasionally you may see a browser cookie warning
2. Please disregard it and simply click on **Sign In** again to continue



7.2 Reviewing ScaleIO Volume Types

1. Please review the Dashboard for OpenStack project - **cloud**
2. Please select **Volumes** in OpenStack Horizon Admin menu on the left hand side
3. Click on the **Volume Types** tab

4. Review your created ScaleIO volume type - **ScaleIO_Thin_Pool**. Note: You may see other volume types in this table created earlier in the previous labs.
5. Please click on the **Extra Specs** menu to see the volume type details.

The screenshot shows the OpenStack dashboard with the 'System' project selected in the sidebar. The main content area displays the 'Usage' section, which includes a table showing resource usage over a specified period. The 'Volume Types' section is also visible, showing a list of existing volume types, including 'ScaleIO_Thin_Pool'. A red box highlights the 'Volume Types' tab, and a blue circle labeled '1' is placed on the 'Project Name' column header. A red box highlights the 'ScaleIO_Thin_Pool' row, and a blue circle labeled '2' is placed on the 'Name' column header. A red box highlights the 'View Extra Specs' button in the 'Actions' column for the 'ScaleIO_Thin_Pool' row, and a blue circle labeled '3' is placed on the 'Volume Types' tab. A red box highlights the 'View Extra Specs' button, and a blue circle labeled '4' is placed on the 'Name' column header. A blue circle labeled '5' is placed on the 'Actions' column header for the 'ScaleIO_Thin_Pool' row.

Project Name	VCPUs	Disk	RAM	VCPU Hours	Disk GB Hours
cloud	1	0Bytes	64MB	312.17	0.00

Displaying 1 item

Name	Associated QOS Spec	Actions
ScaleIO_Thin_Pool		View Extra Specs

Displaying 1 item

Name	Consumer	Specs	Actions
------	----------	-------	---------

7.3 Volume Type Extra Specs

1. Please review the ScaleIO Volume Type extra specs - these were configuration parameters you have provided while creating this Volume Type. These parameters correspond to the ScaleIO Cluster configuration.

2. Please click on the Project Tab - you will now create a new ScaleIO volume

The screenshot shows the OpenStack dashboard with the 'Project' tab selected (highlighted with a red box). The main content area displays the 'Volume Type: ScaleIO_Thin_Pool' configuration. Under 'Volume Type Extra Specs', there is a table titled 'Extra Specs' with three rows. The 'Value' column for all three rows is highlighted with a red box. A blue circle with the number '1' is positioned above the 'Create' button, and a red circle with the number '2' is positioned above the 'cloud' value in the 'Value' column.

Key	Value	Actions
sio:pd_name	cloud	Edit ▾
sio:provisioning_type	thin	Edit ▾
sio:sp_name	pool1	Edit ▾

7.4 Optional Step: Activating Instance

If you have not previously activated a suspended Cloud-Workload VM instance in the earlier labs, please do so now:

1. Select Instances tab in the Project menu
2. Click on the drop-down menu next to the Cloud-Workload instance
3. Select Resume Instance option, then return to the Volumes tab.

The screenshot shows the OpenStack Instances interface. The 'Instances' tab is selected, indicated by a red box and a blue numbered circle '1'. The main table displays one item: 'Cloud-Workload' (TestVM), which is currently 'Suspended'. A context menu is open over this instance, with several options visible: 'Create Snapshot' (disabled), 'Associate Floating IP', 'Edit Instance', 'Resume Instance' (highlighted with a red box and blue circle '2'), and 'Terminate Instance'.

<input type="checkbox"/>	Instance Name	Image Name	IP Address	Size	Key Pair	Status	Availability Zone	Task	Power State	Time since created	Actions
<input type="checkbox"/>	Cloud-Workload	TestVM	192.168.111.4 192.168.1.151	m1.micro	-	Suspended	nova	None	Shut Down	2 weeks, 3 days	Create Snapshot Associate Floating IP Edit Instance Resume Instance Terminate Instance

Displaying 1 item

7.5 OpenStack Project Menu

1. Select Volumes tab
2. You will now create a new Volume using ScaleIO Volume type - please click on Create Volume menu

The screenshot shows the OpenStack Project menu interface. On the left, there is a sidebar with the following tabs:

- Project (highlighted with a red box and a blue numbered circle 1)
- Compute
- Overview
- Instances
- Volumes (highlighted with a red box and a blue numbered circle 2)
- Images

The main content area is titled "Volumes". It has two tabs at the top: "Volumes" (selected) and "Volume Snapshots". Below the tabs is a search bar with filters and a "Create Volume" button (highlighted with a red box and a blue numbered circle 3). A message "No items to display." is shown above a table. The table has the following columns:

Name	Description	Size	Status	Type	Attached To	Availability Zone	Bootable	Encrypted	Actions
Displaying 0 items									

7.6 Creating New Volume

Please provide new ScaleIO Volume details:

1. Volume Name - **ScaleIO-Thin-Volume**
2. Provide a Description as you see fit
3. Leave Volume Source field with default value
4. Select a Volume Type from the drop-down menu - **ScaleIO_Thin_Pool**
5. Specify the Volume size in the increments of 8GB - 16GB
6. Leave Availability Zone field with default value - **Any Availability Zone**
7. Click on Create Volume to continue

Create Volume

Volume Name *

ScaleIO-Thin-Volume

1

Description

ScaleIO Thin Volume

2

Description:

Volumes are block devices that can be attached to instances.

Volume Limits**Total Gigabytes (0 GB)**

1,000 GB A

Number of Volumes (0)

10 A

Volume Source

No source, empty volume

3

▼

Type

ScaleIO_Thin_Pool

4

▼

Size (GB) *

16

5

7

Cancel

Create Volume

7.7 Created Volume

1. Please verify that the new ScaleIO Volume is available. **Note:** You may see other volume types in this table created earlier in the previous labs - please ignore them.
2. Next you will attach this new Volume to our cloud-workload VM instance - please click on the **Edit Volume** drop-down menu
3. Select **Edit Attachments** from the list

Volumes

Volumes Volume Snapshots

Filter Filter + Create Volume ✖ Delete Volumes

<input type="checkbox"/>	Name	Description	Size	Status	Type	Attached To	Availability Zone	Bootable	Encrypted	Actions
1	ScaleIO-Thin-Volume	ScaleIO Thin Volume	16GB	Available	ScaleIO_Thin_Pool		nova	No	No	<div style="border: 1px solid red; padding: 2px;">Edit Volume</div> <div style="margin-top: 2px;">▼</div> <div style="border: 1px solid red; padding: 2px;">Extend Volume</div> <div style="margin-top: 2px;">Edit Attachments</div> <div style="margin-top: 2px;">Create Snapshot</div> <div style="margin-top: 2px;">Upload to Image</div> <div style="margin-top: 2px;">Delete Volume</div>

Displaying 1 item

1 2 3

7.8 Attaching Volume

1. Please click on the drop-down menu
2. Select an active instance from the - Cloud-Workload
3. Click **Attach Volume** to continue

Manage Volume Attachments

Attachments

Instance	Device	Actions
No items to display.		

Displaying 0 items

Attach To Instance

A 2 1 to Instance * ②

Cloud-Workload (a1b7f7e4-0bd1-423d-a264-f1436b1c6f00) 1

Select an instance

Cloud-Workload (a1b7f7e4-0bd1-423d-a264-f1436b1c6f00)

Cancel 3 Attach Volume

7.9 Volume In-Use

1. Please review the status of the newly attached volume - In-Use

Now please return to the previously minimized ScaleIO UI Dashboard

	Name	Description	Size	Status	Type	Attached To	Availability Zone	Bootable	Encrypted	Actions
<input type="checkbox"/>	ScaleIO-Thin-Volume	ScaleIO Thin Volume	16GB	In-Use	ScaleIO_Thin_Pool	Attached to Cloud-Workload on /dev/vdb	nova	No	No	<button>Edit Volume</button>

7.10 Created Volume in ScaleIO

Please review the newly created volume in the ScaleIO Dashboard and notice the Volume status - Mapped

Note: It may take several minutes for ScaleIO to update the volume information.

SDCs 3 Defined: 3	Volumes 1 Mapped Defined: 2 Free: 120.0 GB	Protection Domains 1 Storage Pools: 2	SDSs 3 Devices: 3	Management Clustered
---------------------------------	--	---	---------------------------------	-----------------------------

8. Conclusion

In this lab you, wearing the hat of Code Nebulous Cloud Administrator, have successfully accomplished the following:

- Deployed and configured the EMC ScaleIO cluster on the OpenStack Compute nodes to take advantage of the commodity spare physical drives available in the Compute nodes
- Deployed and configured ScaleIO Cinder driver to use the ScaleIO block storage cluster as a backend
- Created Cinder volume types associated with the ScaleIO storage pools
- Created and attached a new block volume to an active OpenStack instance.

Now the Code Nebulous Cloud Users can begin building and using the applications running on the Cloud-Workload instance and store its data on the persistent block volume.

Note: In the interests of time you will not be logging into Cloud-Workload instance to format and mount the newly created volume.

This lab is now complete. Please close the following windows as you will not be using them any longer:

- Lab-3-ScaleIO_Cinder folder
- 7_ScaleIO_Cinder_Config_Commands file.
- ScaleIO GUI
- SSH Terminal Window

Please proceed to the next lab, or select other labs from the list.

Optional Lab 4 - Live Migration of OpenStack Instance & Volume

Lab 4 - Live Migration

In this lab you as a Cloud Administrator will learn how to migrate a "live" active OpenStack virtual machine instance and its associated storage volume. In the course of the Agile development process, Cloud Admins will encounter a situation when the workload and its associated data needs to be moved from a low-cost distributed storage to a high-performance high-availability external storage array.

Upon completion of this lab you will become familiar with the Live Migration configuration steps and use OpenStack CLI commands to migrate an active VM from one compute node to another and its attached volume to a different storage backend (from ScaleIO to VNX).

This Lab 4 requires that you have previously completed the following labs:

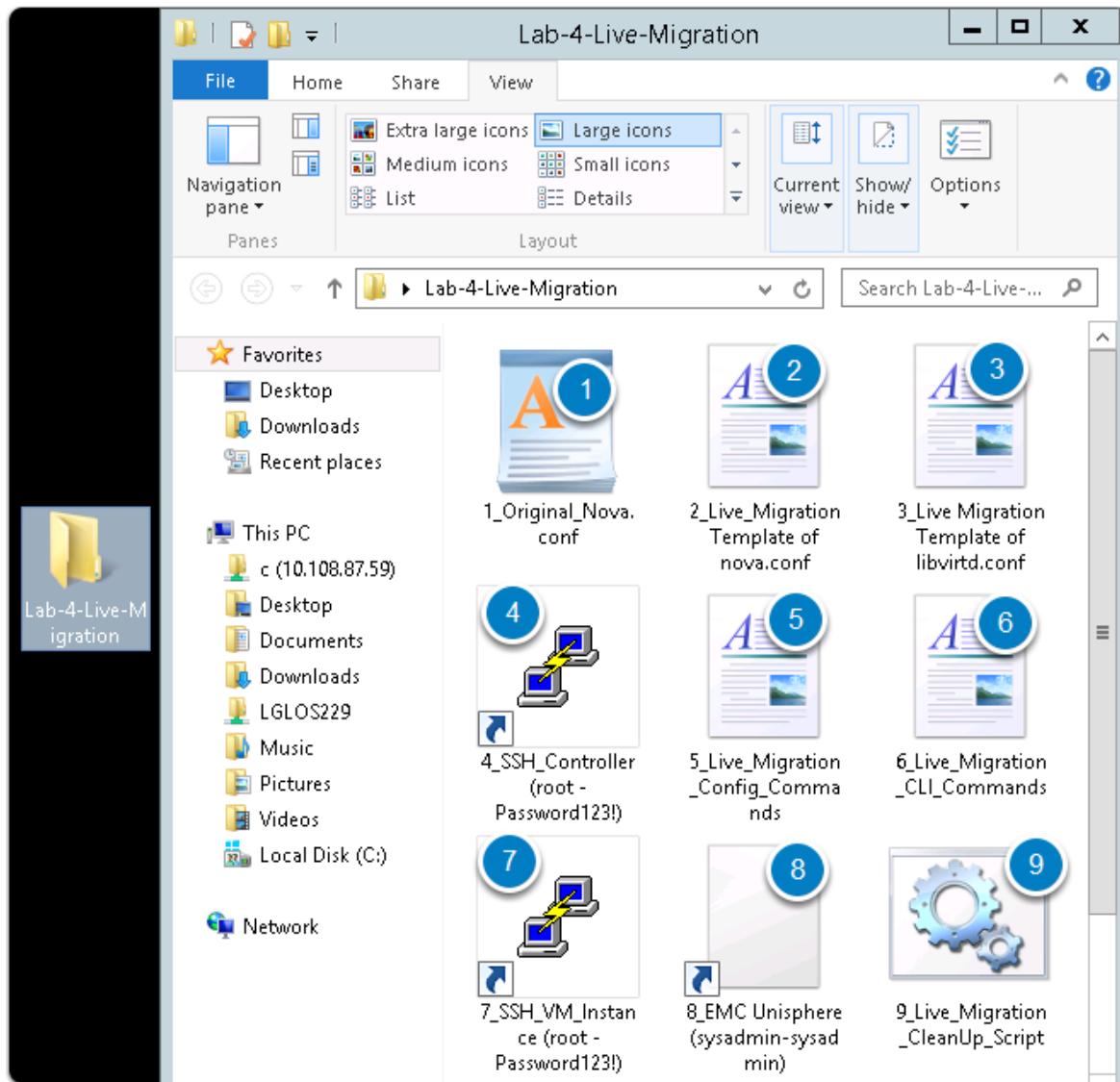
- Lab 1 - Using VNX with Cinder Block Storage
- Lab 3 - Using ScaleIO with Cinder Block Storage

1. Lab-4 Folder

Please open and review **Lab-4-Live-Migration** folder

Pay attention to the Numbers in the name of the shortcut - for your convenience you will be using these shortcuts in order

1. Original Nova Config File
2. Live Migration Template of Nova.Conf
3. Live Migration Template of libvirt.conf file
4. SSH Utility to access the OpenStack Controller terminal
5. Live Migration Configuration Commands
6. Live Migration CLI Commands
7. SSH Utility to access OpenStack VM Instance terminal
8. EMC Unisphere to view the Storage Pool and migrated volumes
9. Cleanup script to complete this lab and prepare for the next one



1.1 Live Migration Configuration Steps

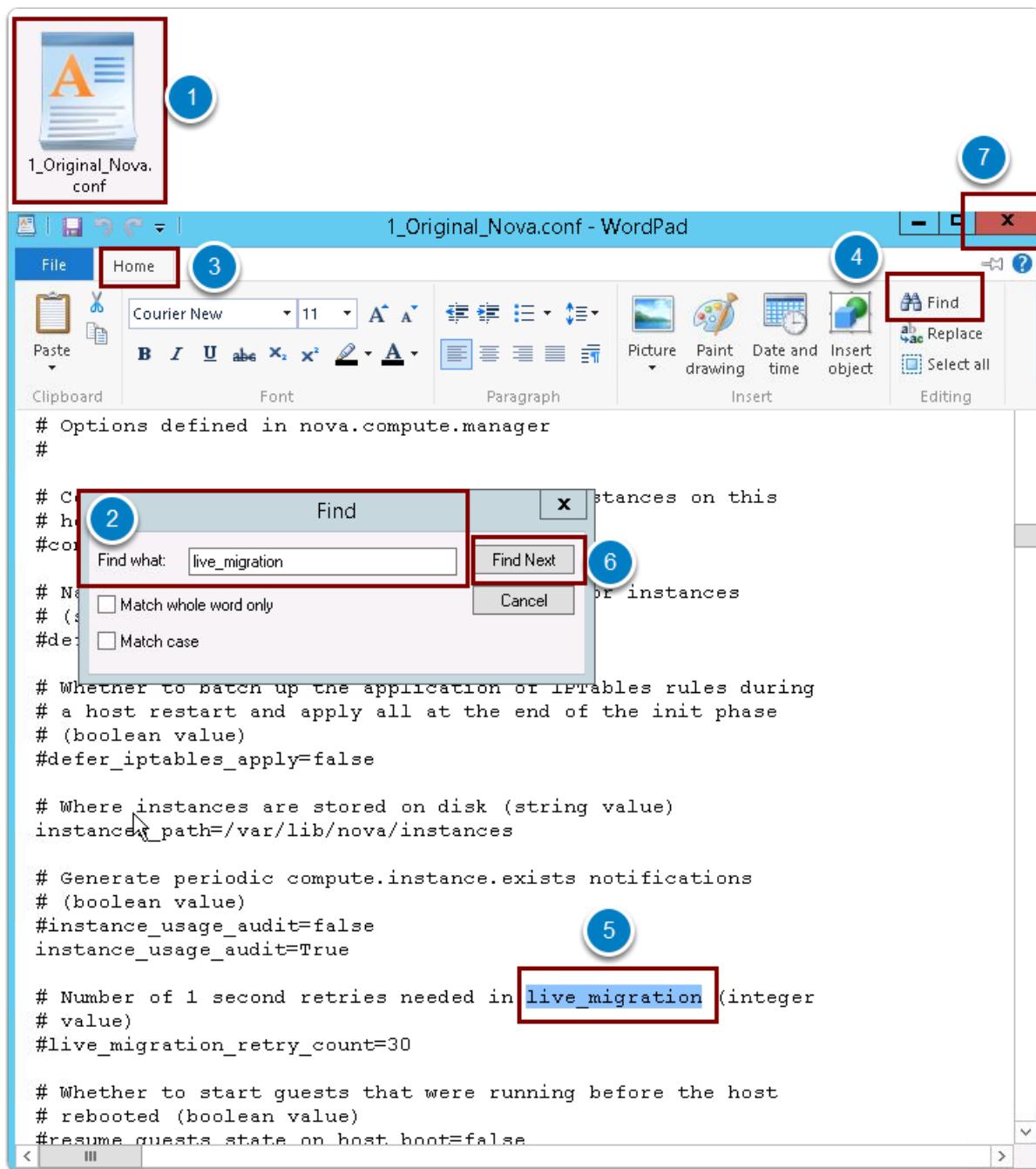
Live Migration capabilities for a VM instance and its attached volume, you will now

- Review the **Nova Compute** service configuration on **Compute nodes**
- Review the **libvirt** service configuration on **Compute nodes**
- Configure **Cinder** service by modifying the **Cinder configuration file /etc/cinder/cinder.conf** on the **Controller node**
- Restart **Cinder API** and **Cinder Volume** services
- Initiate a live migration of a volume attached to an active VM instance.

1.2 Original Nova. Conf File

For your reference you have been provided with the Nova.conf configuration file where all necessary configuration settings for enabling Live Migration on the Compute Nodes need to be made.

1. Please open the **1-Original_Nova.conf** and review it
2. You will use the search pattern: **live_migration**
3. In WordPad, select **Home**
4. Then select **Find** menu option
5. Review all settings for enabling Live Migration
6. Click **Find Next** as necessary
7. Then close the file to continue



1.3 Sample Nova.conf file

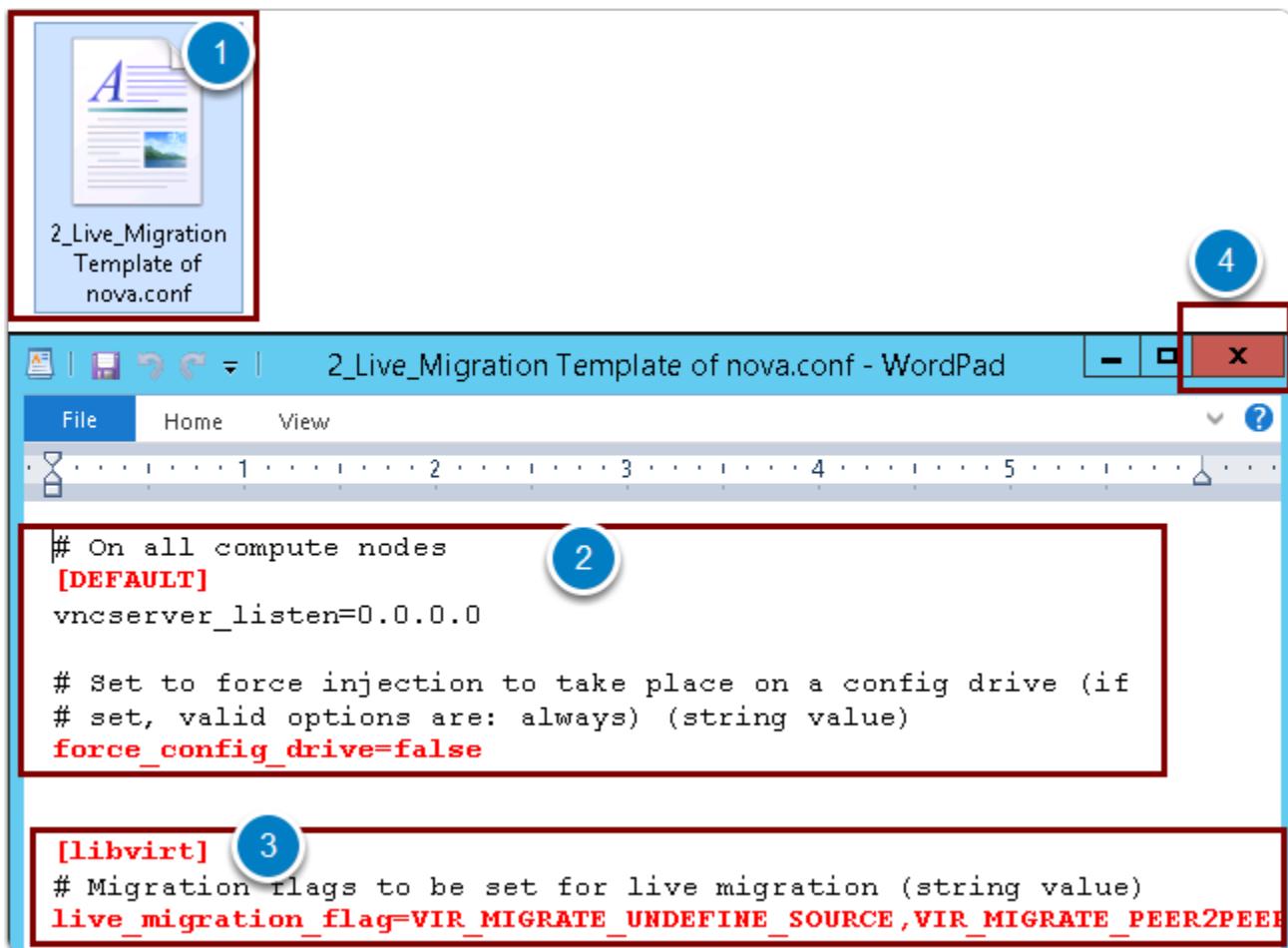
For your convenience, all the changes required for configuration of the Nova Compute service have been summarized in a sample Nova.conf file.

The relevant sections are highlighted in Red.

1. Please open a file called 2_Live_Migration Template of Nova.conf - these settings should be applied to ALL OpenStack Compute Nodes.
2. Review the settings in the [DEFAULT] section for vncserver and config drive.
3. Review the settings in the [libvirt] section for live migration flags

To save your time a nova.conf file with all the highlighted settings has already been created. As part of the lab you will simply copy over the modified file. This has already been completed for you to save you time in completing this lab.

4. Please close the 2_Live_Migration Template of Nova.conf



1.4 Sample Libvirt Configuration File

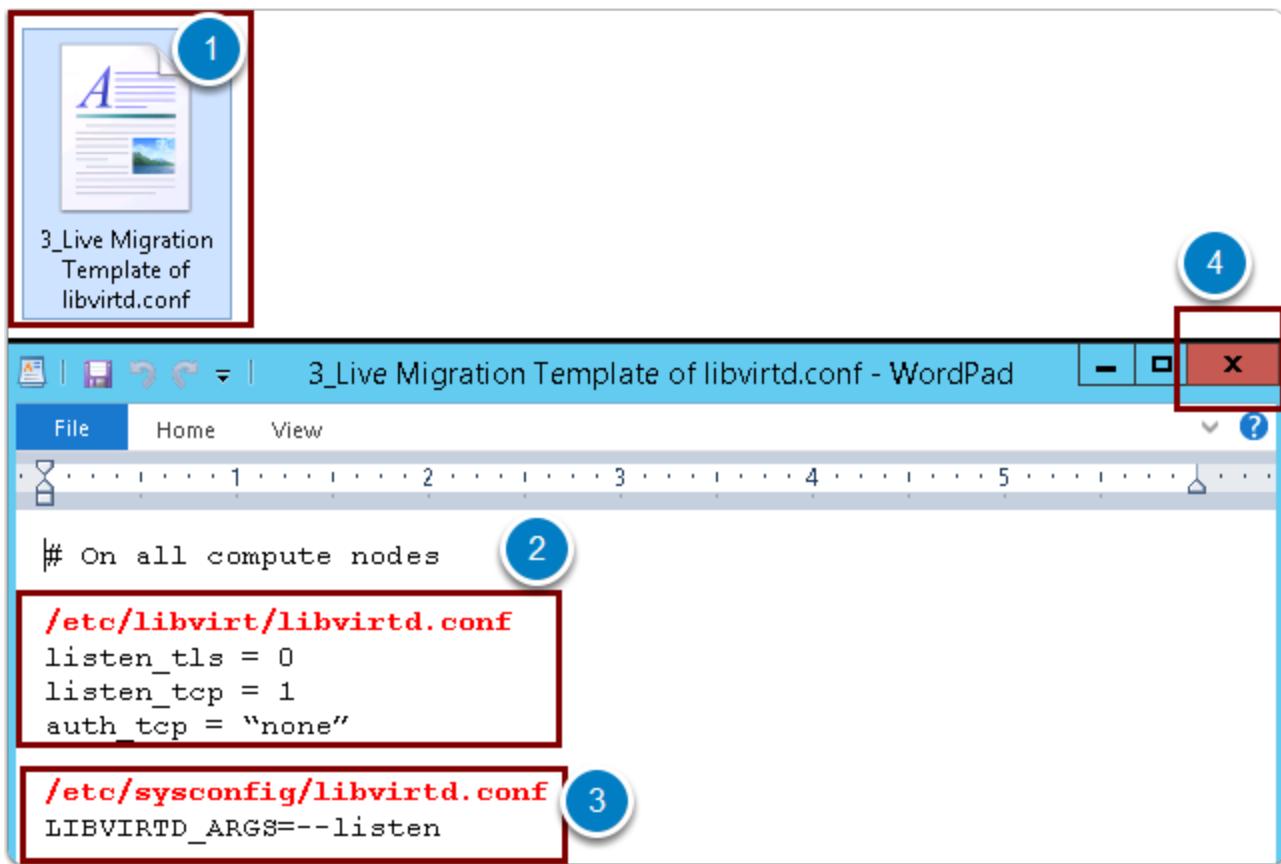
For your convenience, all the changes required for configuration of the Libvirt service have been summarized in a sample Libvirt.conf file.

The relevant sections are highlighted in Red.

1. Please open a file called **3_Live_Migration Template of libvirt.conf** - these settings should be applied to ALL OpenStack Compute Nodes.
2. Review the settings in the **/etc/libvirt/libvirtd.conf** file for TCP Authentication.
3. Review the settings in the **/etc/sysconfig/libvirtd.conf** file

To save time, refer to the pre-created **libvirtd.conf** files with all the highlighted settings. As part of the lab you will simply copy over the modified files. This has already been completed for you to save you time in completing this lab.

4. Please close the **3_Live_Migration Template of libvirt.conf**

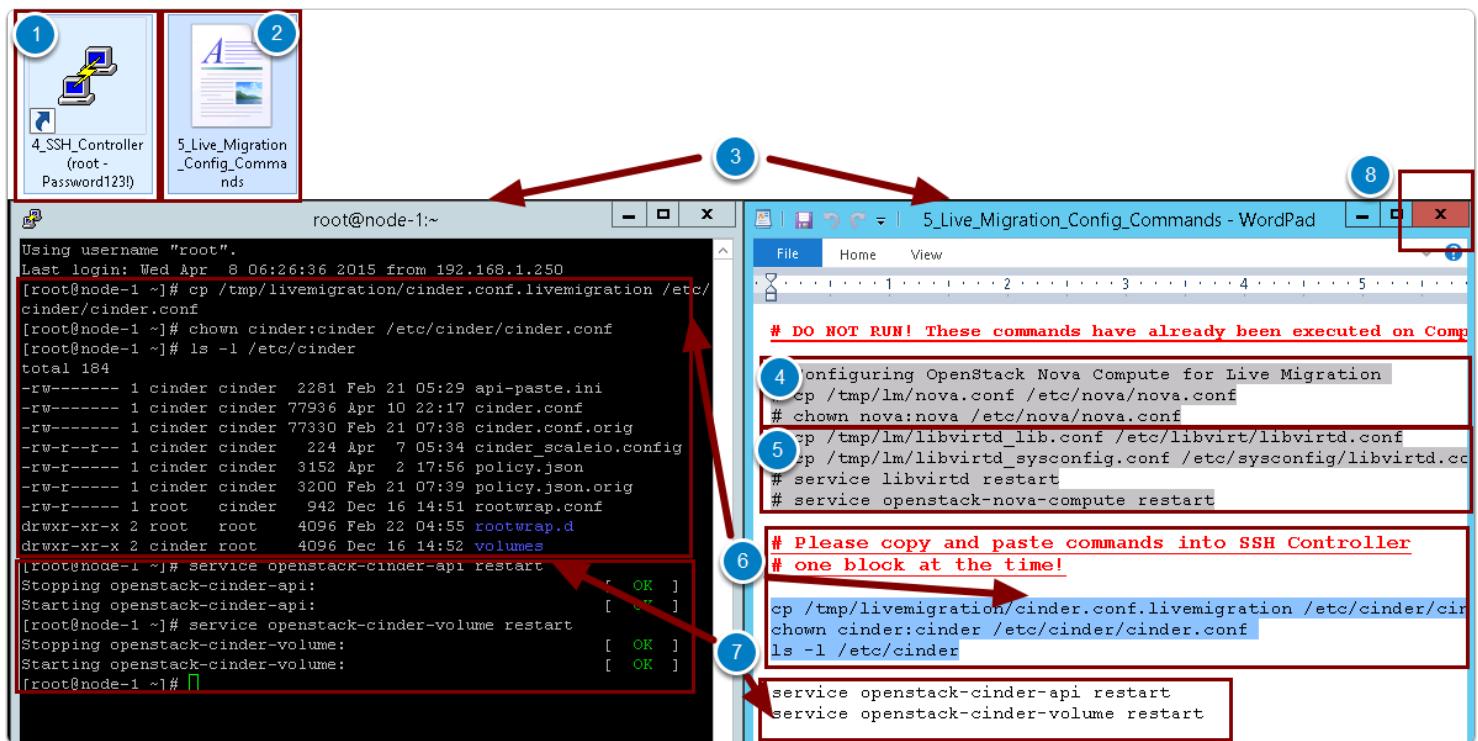


1.5 Using PuTTY SSH Utility

To save you some time in running this lab, a file with the commands to be executed on the OpenStack Compute and Controller Nodes has been created to configure Live Migration:

1. Please open PuTTY shortcut to OpenStack Controller Node using shortcut called **4_SSH_Controller**
2. Then open **5_Live_Migration_Config_Commands** file

3. Please arrange the open SSH Terminal window and the command file side by side as shown - you will be copying the commands from the file into terminal window using right click, Copy and Paste. The grey-shaded areas contain commands that are necessary for installation, but take some time to complete. Please DO NOT run these commands, we have already executed them.
4. First greyed-out block was used to configure the Nova Compute service settings on the Compute Nodes
5. Second greyed-out block was used to configure the libvirt service settings on the Compute Nodes
6. It is also suggested to copy and paste commands into **SSH Terminal** window as a block. Blocks are separated by blank lines. Please copy the third command block and paste into **SSH Terminal** window using **right-click**. These commands will configure Cinder Block storage service on the Controller node for Live Migration
7. Then continue with the fourth command block - these commands will restart the Cinder services so the Live Migration settings take effect.
8. When finished, please close the **5_Live_Migration_Config_Commands** file. **!!! Note: Please keep the SSH terminal session for Controller open!**



2. Live Migration

During the Live Migration you, as a Cloud Admin, will:

- Review the sequence of steps and associated commands to migrate a VM instance **Cloud-Workload** from a DevOps Cloud Host (represented by node-2.vlab.local) to a Production Cloud (represented by node-3.vlab.local)

- Migrate Block Volume attached to the above instance from DevOps Cloud storage (represented by ScaleIO backend) to the high-performance Production Cloud Storage (represented by VNX backend).

2.1 Migration Commands

To demonstrate the versatility of the OpenStack integration with EMC Storage, you will be performing the Live Migration using OpenStack Command Line (CLI). For your convenience, a text file with all required commands has been prepared in the Lab-4-Live-Migration folder

- Please open the shortcut called **6_Live_Migration_CLI_Commands** by double-clicking on it.



2.2 VM Instance Migration Commands

Please review the VM Instance migration commands in the opened **6_Live_Migration_CLI_Commands** file.

Note: You will NOT be executing the **VM instance Live Migration** as it takes some time to complete due to the size of the instance.

1. The greyed-out commands provide you with the overview and details of the Cloud-Workload instance and its current and future hosts and commands for the live migration of the Cloud-Workload instance.
2. Please note the VM Instance ID - it will be used in the Volume Migration

Note: Please DO NOT execute the live migration of the VM instance as this will take long time!

```

# Please copy and paste commands into SSH Controller
# one at the time! Do NOT copy the greyed-out lines!
1 Instance Migration. !Important: DO NOT RUN these commands!
# nova list
# Select an instance ID and review instance details.
# nova show a1b7f7e4-0bd1-423d-a264-f1436b1c6f00
# Select the compute node the instance will be migrated to (node-3.vlab.local).
# nova service-list
# Verify that compute node has sufficient resources for migration:
# nova host-describe node-4.vlab.local
# Migrate the active instance:
# nova live-migration --block-migrate a1b7f7e4-0bd1-423d-a264-f1436b1c6f00 node-
# Verify that instance have migrated successfully:
#nova list
# nova show a1b7f7e4-0bd1-423d-a264-f1436b1c6f00 2

# Volume Migration. !Important: Volume must be attached!
# List available backends:
source /root/adminrc
cinder service-list

# Create new volume to be migrated: ScaleIO-VNX-Migration 32GB
cinder create --display-name ScaleIO-VNX-Migration --volume-type ScaleIO_Thin_Po

# Attache new volume to VM Instance - insert the volume_id from the previous command:
nova volume-attach a1b7f7e4-0bd1-423d-a264-f1436b1c6f00 <volume_id>

# Migrate this volume to another backend (from ScaleIO to VNX iSCSI):
# Insert the volume_id from the previous command:
cinder migrate --force-host-copy True <volume_id> node-1.vlab.local@vnxiscsi#vnx

# Verify that volume migrated successfully:
# Insert the volume_id from the previous command:
cinder show <volume_id>
# Please launch EMC Unisphere in your browser window!

```

2.3 Creating Volume for Migration

1. Return to previously opened 6_Live_Migration_CLI_Commands file arranged side by side with the opened 4_SSH_Controller terminal window
2. Next command block provides you with the details of the storage backends and service. Copy and paste the command lines from the text file into SSH terminal window using right-click - please only execute commands

that are not commented out nor greyed-out! Also ensure that you select the **ENTIRE** command line! (use Shift+End)!

3. You will now create a new volume to be migrated from ScaleIO to VNX. Please run the command:

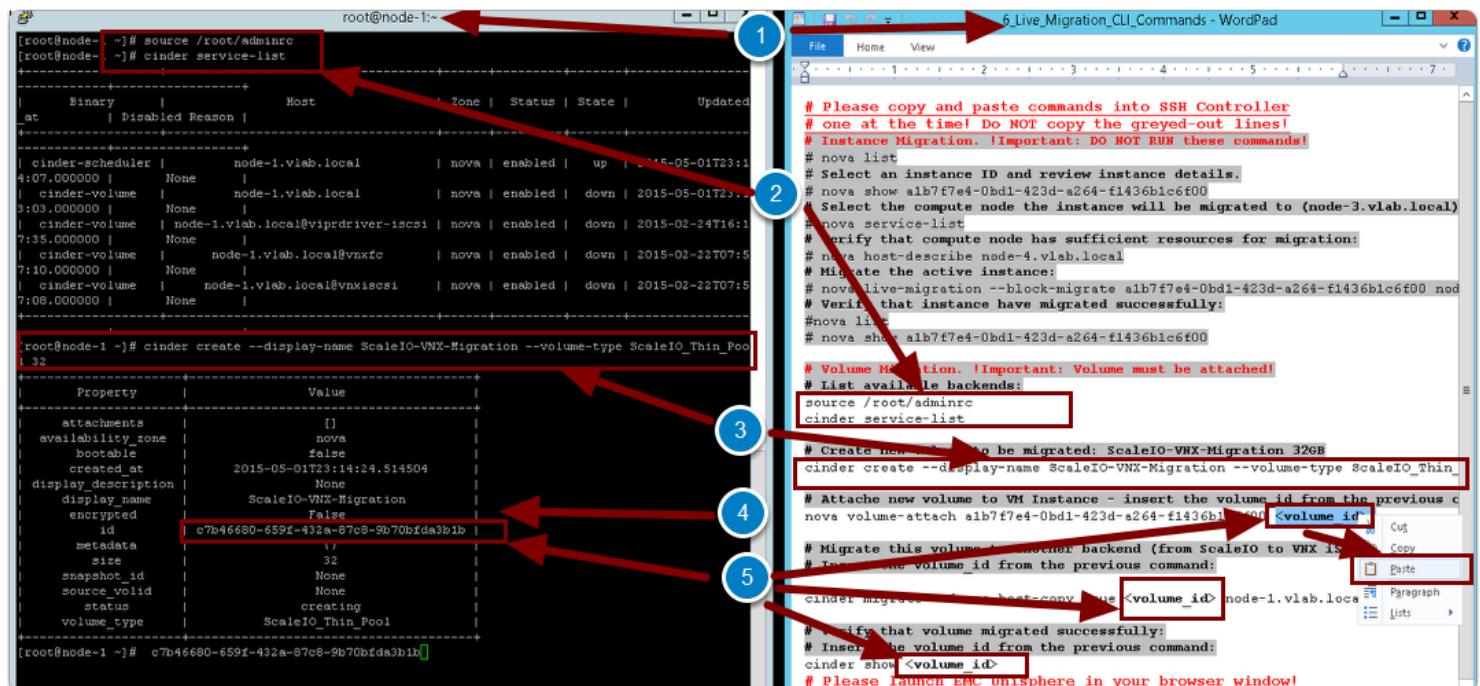
cinder create --display-name ScaleIO-VNX-Migration --volume-type ScaleIO_Thin_Pool 32 - You can copy this command from the file and paste into SSH Terminal window using Right-Click

4. Highlight the Volume ID in the **cinder** command output and copy it into Clipboard using Right-Click
5. Highlight the **<volume_id>** in the sample commands below and use the Right-Click menu Paste option to replace the **<volume_id>** placeholder with the actual volume ID from the previous command:

nova volume-attach a1b7f7e4-0bd1-423d-a264-f1436b1c6f00 <volume_id>

cinder migrate --force-host-copy True <volume_id> node-1.vlab.local@vnxiscsi#vnx_iscsi

cinder show <volume_id>



2.4 Volume Migration

You will now attach the volume to the active VM instance and migrate it:

1. Please select the entire command with <Volume_ID> replaced with your actual volume ID from the previous command using Shift-End, copy it and paste it in the SSH Terminal window using Right-Click. Run the command:

```
nova volume-attach a1b7f7e4-0bd1-423d-a264-f1436b1c6f00 <volume_id>
```

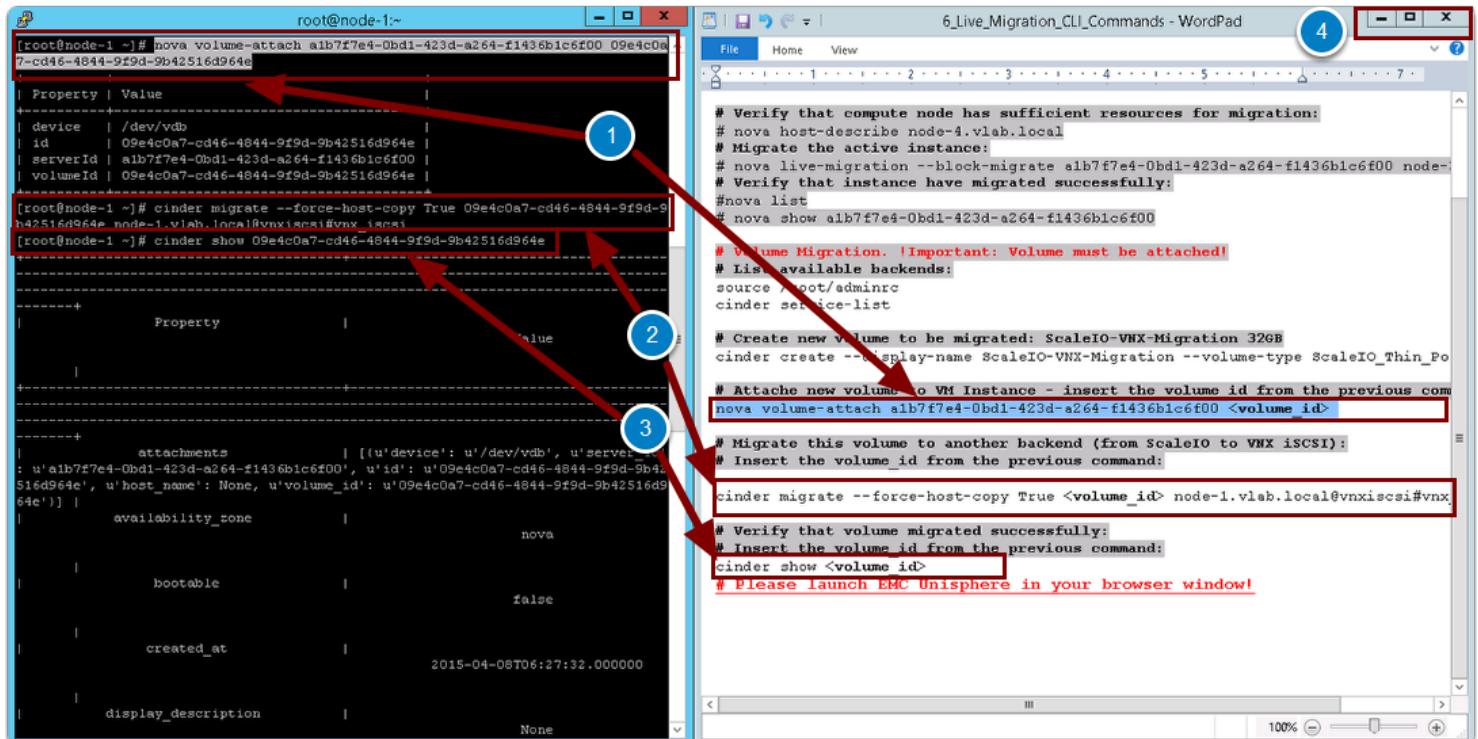
2. Migrate the volume attached to Cloud-Workload instance from ScaleIO backend to VNX iSCSI backend. Select the entire command with <Volume_ID> replaced with your actual volume ID using Shift-End, copy it and paste it in the SSH Terminal window using Right-Click and run the command:

```
cinder migrate --force-host-copy True <volume_id> node-1.vlab.local@vnxiscsi#vnx_iscsi
```

3. Verify the migration by selecting the entire command with <Volume_ID> replaced with your actual volume ID. Select the entire command with <Volume_ID> replaced with your actual volume ID using Shift-End, copy it and paste it in the SSH Terminal window using Right-Click and run the command:

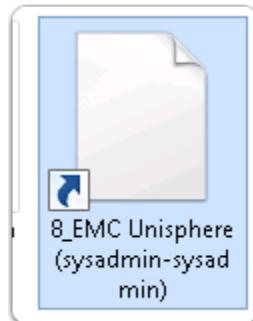
cinder show <volume_id> and review the volume details

4. When finished, please close the 6_Live_Migration_CLI_Commands file, you will not need it again. Click Don't Save to confirm closing the file.



3. Launching EMC Unisphere

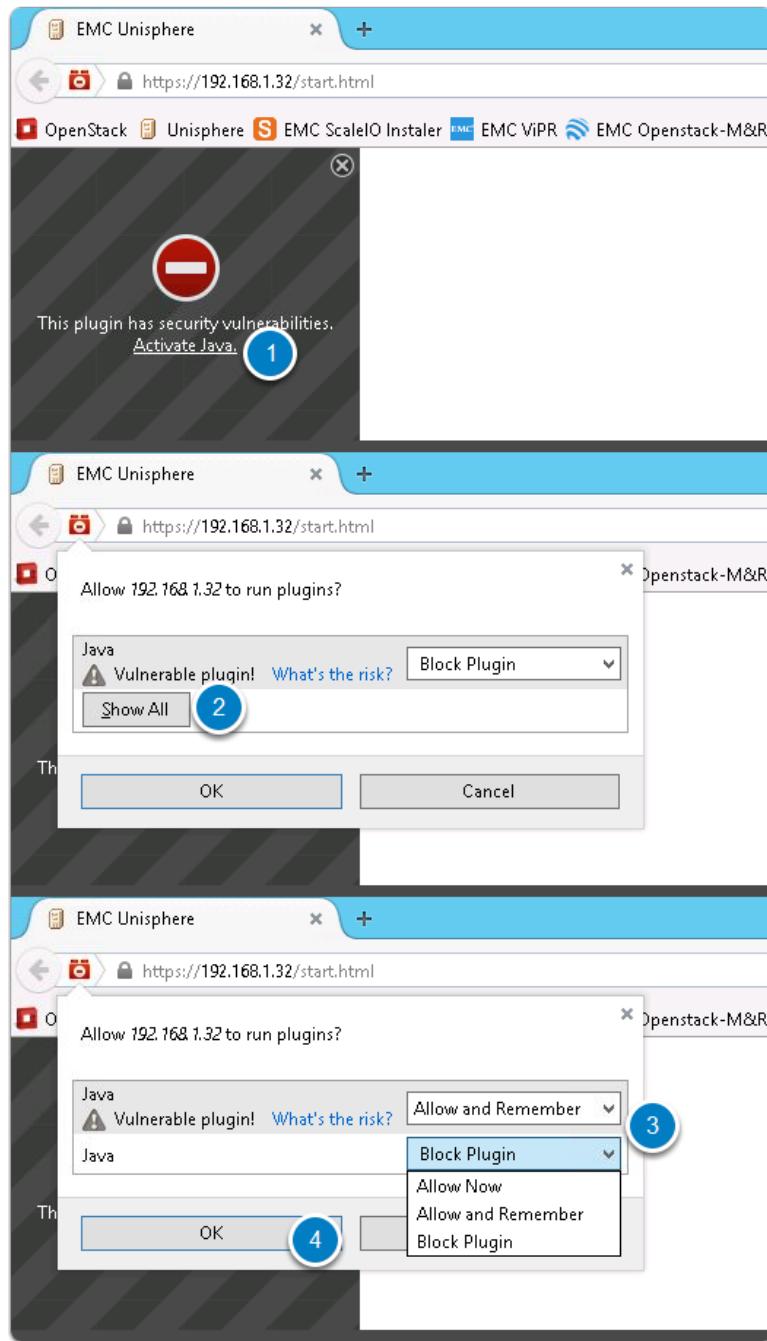
Now that the Volume has been migrated, you can verify the new Volume on the VNX storage system using **EMC Unisphere**. Please double-click on the **8_EMC_Unisphere (sysadmin-sysadmin)** shortcut in the Lab-4-Live-Migration folder.



3.1 Optional Step: Java Security Warning

If you see Java Security warning accept plugin by doing the following:

1. Click **Activate Java** link
2. Click **Show All** button
3. Select **Allow and Remember** for both plugins
4. Click **OK**

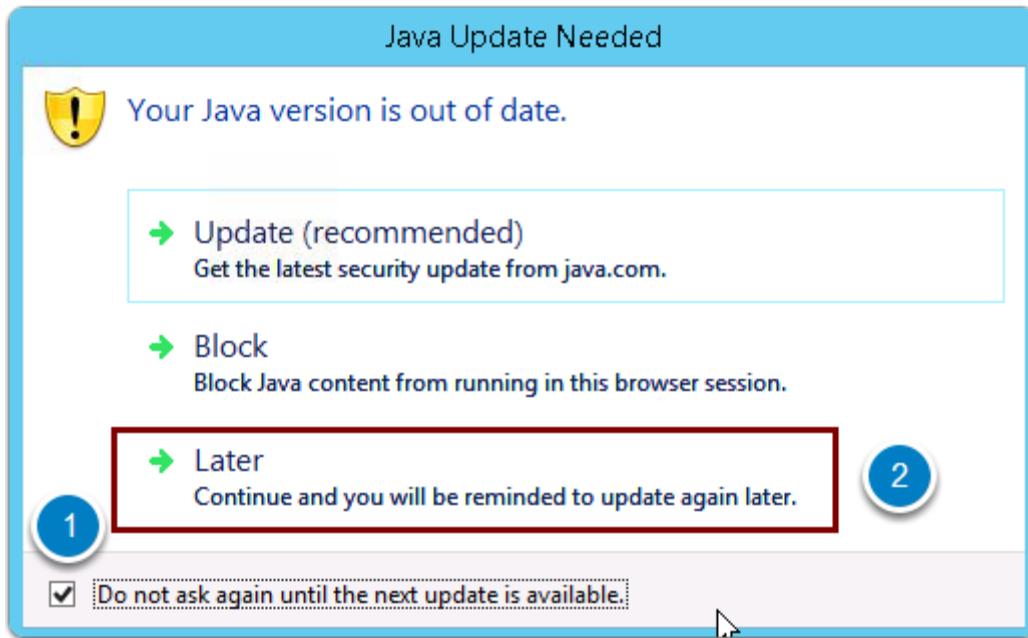


3.2 Optional Step: Java Update

You may see a Java Update Needed window.

1. Click "Do not ask again until the next update is available" checkbox
2. Click Later

Note: You may need to repeat these steps twice!

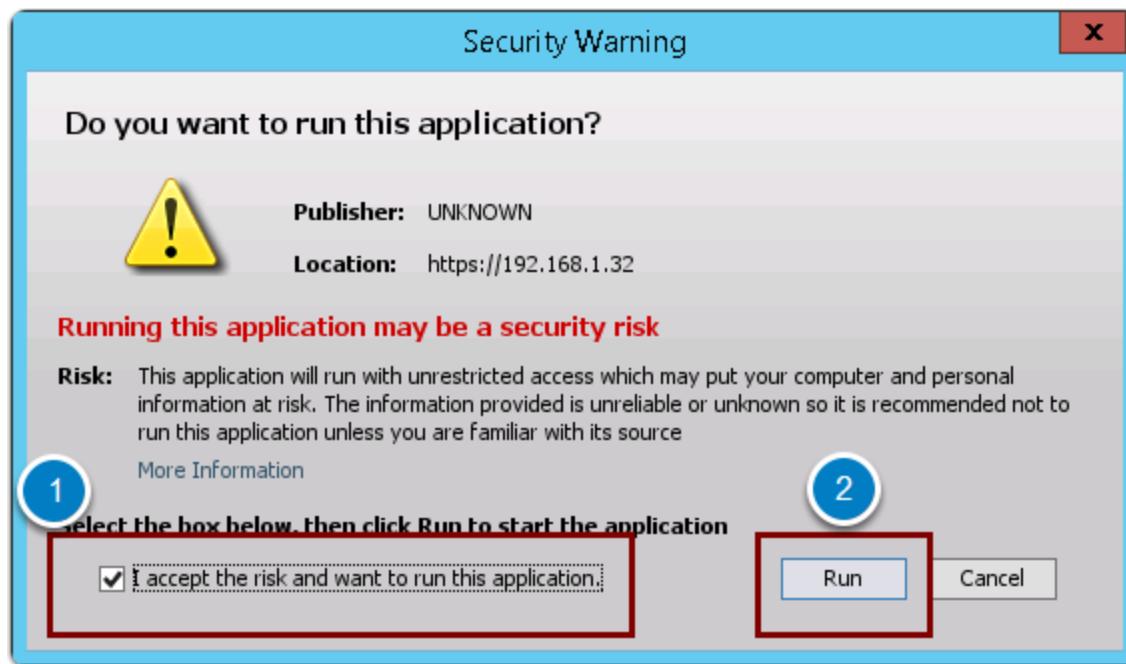


3.3 Accepting Java Security Warning

Please pay attention to the security warning for Java applet:

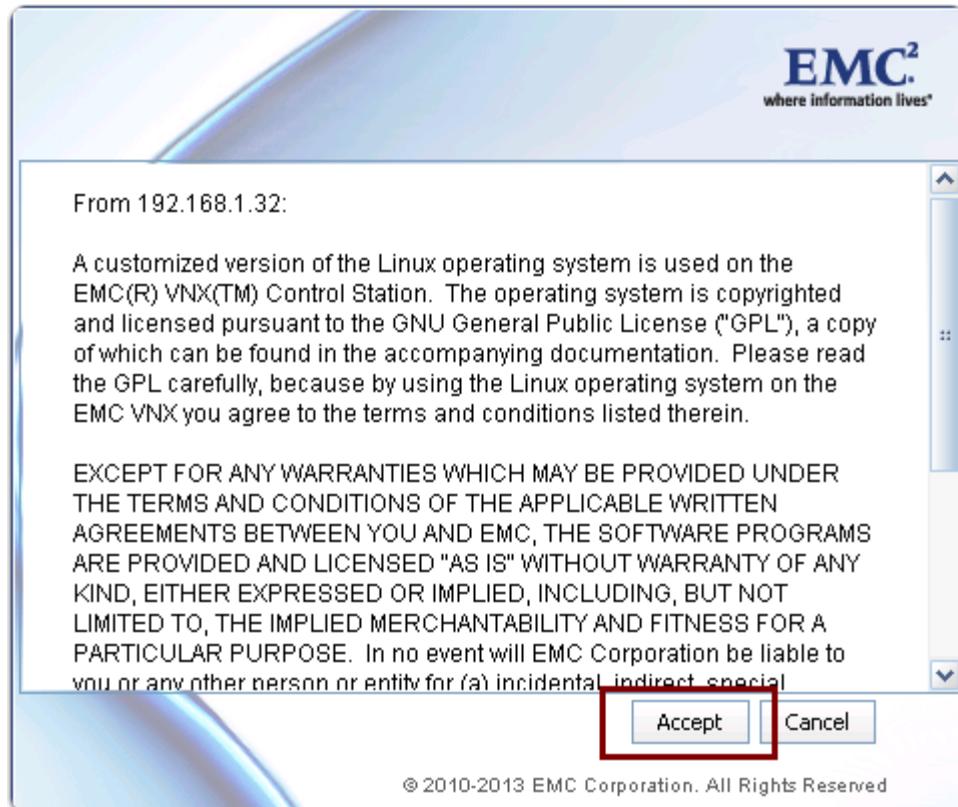
1. Check the box to accept the risk
2. Click Run to continue

Note: You may need to repeat these steps twice!



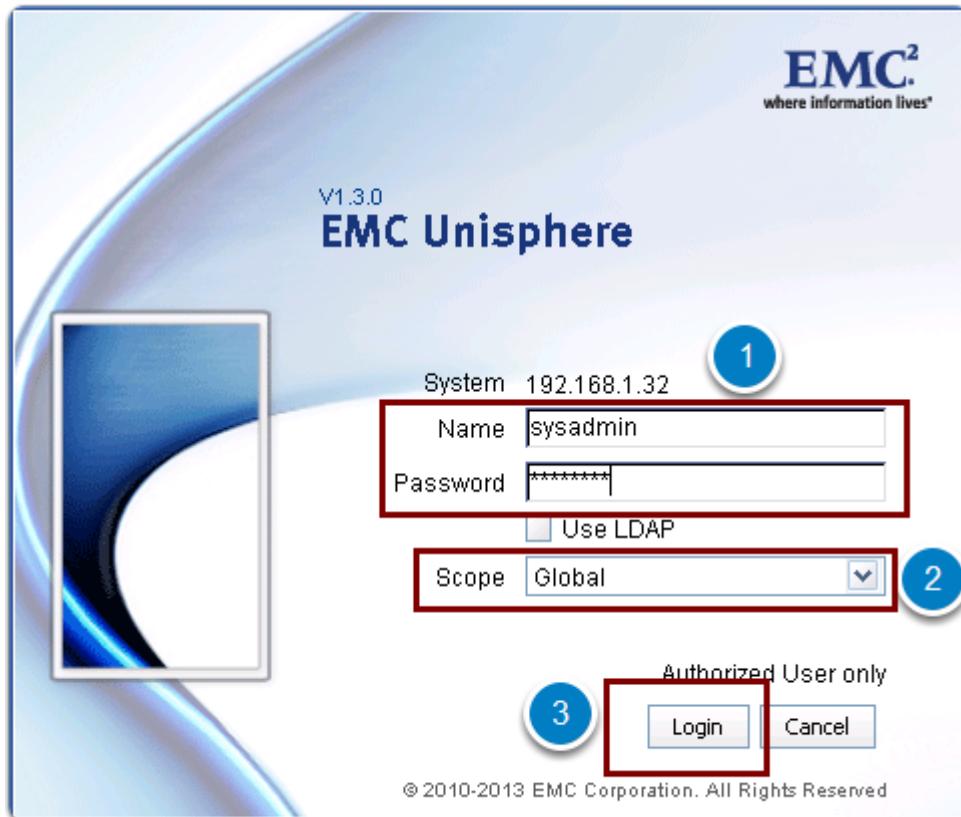
3.4 Accepting EMC License Warning

Please accept EMC Corporation License agreement:



3.5 Logging into Unisphere

1. Please login into Unisphere with credentials: *sysadmin - sysadmin*
2. Make sure your Scope is selected as Global
3. Click Login to continue



3.6 Unisphere Dashboard

1. In the Unisphere Dashboard please select the Storage System called VSA00084500833

The screenshot shows the EMC Unisphere Dashboard interface in Mozilla Firefox. The URL in the address bar is <https://192.168.1.32/start.html>. The dashboard has a blue header with the Unisphere logo and navigation tabs: All Systems, Dashboard (which is selected), System List, Domains, and Alerts.

Systems by Severity

System	Domain	Status	Model
VSA00084500833	Local	Error (1)	VNX5400 (U...)

A red box highlights the row for VSA00084500833, and a blue circle with the number 1 is positioned above the table.

Alerts by Severity

Severity	System	Message	Created
Error	VSA00084500833	DPE (Bus 0 Enclo...	Aug 20, 2014 6:1...

Overall Capacity - Most Free Space (1 of 1)

This chart displays capacity in GB for the system VSA00084500833. The Y-axis ranges from 0 to 500 GB, and the X-axis ranges from 0 to 500 GB. A blue bar represents Free Raw Disk space, reaching approximately 430.49 GB. An orange bar represents Free Storage Pool space, reaching 450 GB.

Capacity (GB)

Free Raw Disk
Free Storage Pool

VSA00084500833

430.49

Last Refreshed: 2015-02-02 16:58:32

Capacity for File - Most Free Space (1 of 1)

This chart displays capacity in GB for the system VSA00084500833. The Y-axis ranges from 0 to 18 GB, and the X-axis ranges from 0 to 18 GB. A green bar represents Free space, reaching 10 GB.

Capacity (GB)

Free

VSA00084500833

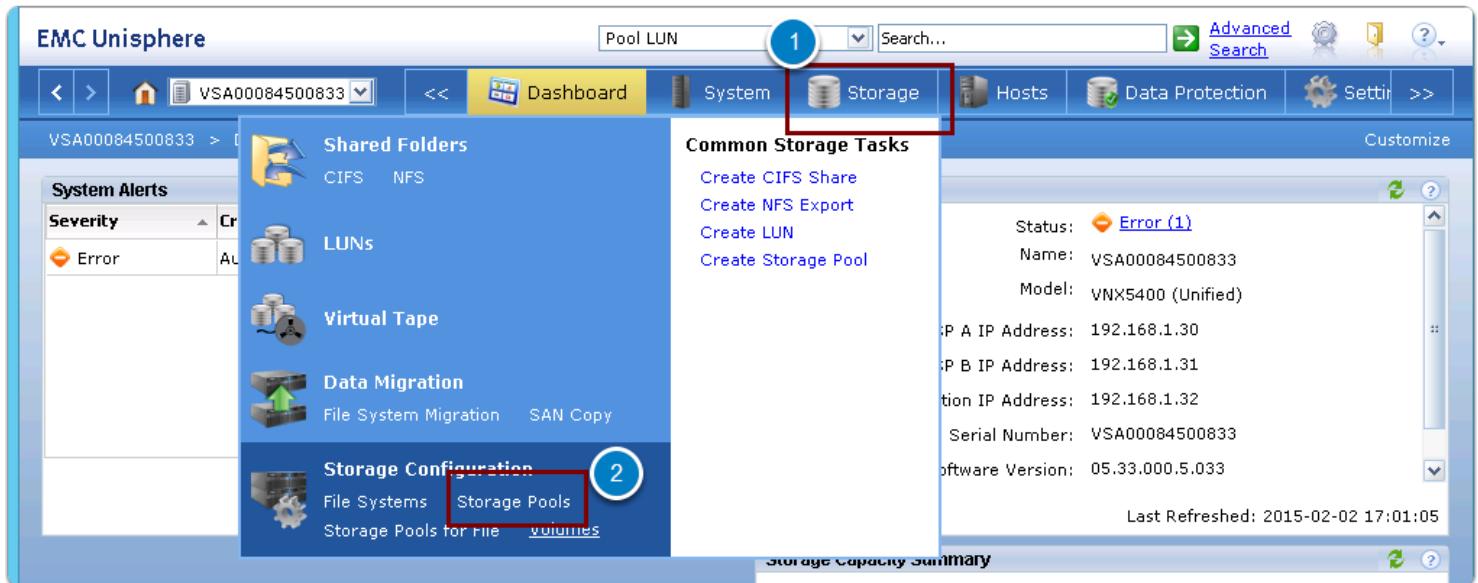
10

Last Refreshed: 2015-02-02 16:58:34

Alerts: 1 Error Certificates: 1 User: sysadmin

3.7 Block Storage Volumes

1. Next please hover over the **Storage** menu from the top menu bar
2. Navigate to the **Storage Configuration** options and select **Storage Pools**



3.8 Newly Migrated Volume

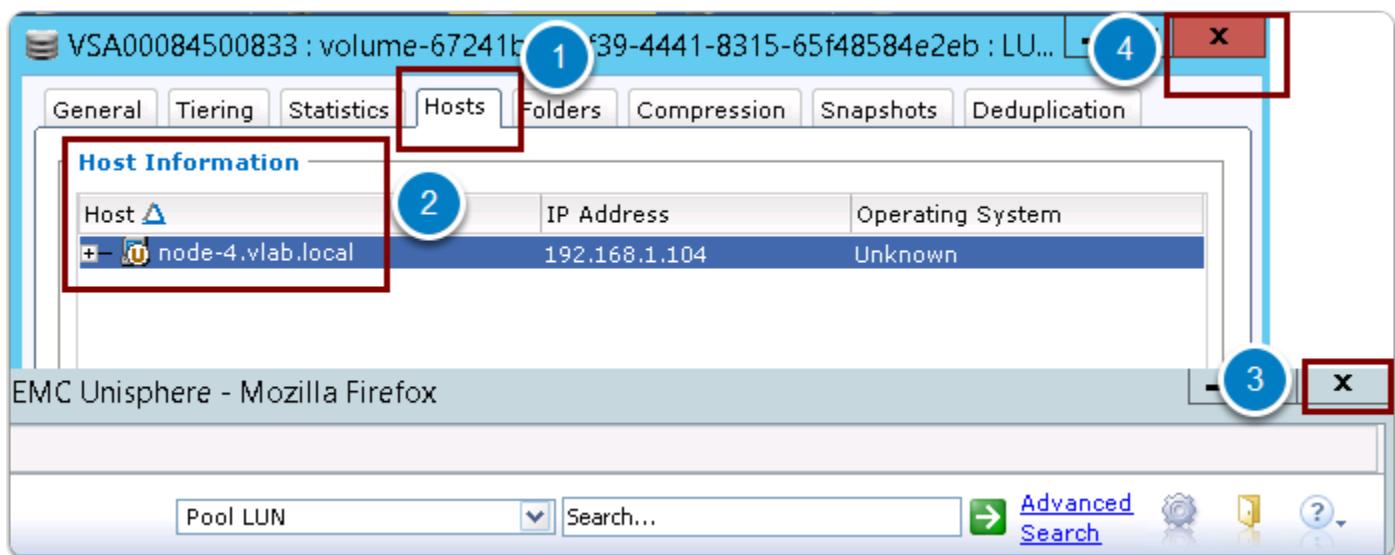
1. Please select the OpenStack Storage Pool
2. Click Refresh icon to see your newly migrated volume
3. Click on the LUN Tab
4. Highlight the newly migrated LUN (volume) - note the Volume size 32GB - this is the same volume you have created earlier.
5. Click on Properties to review the details of this newly created LUN

The screenshot shows two main windows from the EMC vLab interface:

- Pools Window (Top):** Shows a table of storage pools. The first pool, "OpenS... On" (labeled 1), has a yellow bar under %Consumed. The second pool, "Pool 1 ... On" (labeled 2), has a brown bar under %Consumed. A red box highlights the Refresh icon in the top right corner.
- Details Window (Bottom):** Shows a table of LUNs. The first LUN, "volume-97b7fe0b-d35e-49b6-f4746b508d09" (labeled 3), has a yellow bar under %Consumed. A red box highlights the "LUNs" tab. The "Properties" button in the bottom left (labeled 5) is also highlighted with a red box. A red box highlights the "Properties" button in the bottom left of the LUN table (labeled 4).

3.9 Volume Attachment

1. Select Hosts Tab in the LUN's Properties dialog
2. Review the Hosts in this LUN's export group: node-4.vlab.local - this is the new OpenStack Compute nodes to which you have migrated the Cloud-Workload instance earlier in this lab
3. Please close the LUN Properties dialog
4. Then close EMC Unisphere Browser window.



4. Note!

Note: when executing the subsequent labs you may see two migration volumes in the OpenStack Horizon UI Volumes tab - the original one and the migrated one. This is normal at this time, as Horizon currently does not support volume migration. Please ignore the second volume and its status."

5. Conclusion

In this lab you, wearing the hat of Code Nebulous Cloud Administrator, have successfully accomplished the following:

- Configured OpenStack Controller and Compute nodes for live migration of a VM instance and its associated block storage volume
- Reviewed the migration process for VM instance from one Compute node to another based on its available CPU and RAM resources
- Migrated an attached block storage volume from ScaleIO storage pool to a VNX array storage pool
- Verified the availability of the migrated storage volume and its attached instances

Now the Code Nebulous Cloud Users can begin using the newly built applications running on the Cloud-Workload instance in a high-performance Production environment!

This lab is now complete. Please close the following windows as you will not be using them any longer:

- **Lab-4-Live-Migration** folder
- **SSH Terminal Window**
- **EMC Unisphere** browser window

Please proceed to the next lab, or select other labs from the list.

Optional Lab 5 - Using EMC ViPR with Cinder Block Storage Service

Lab 5 - Using EMC ViPR with Cinder

ViPR Controller includes a Cinder driver, which interfaces between ViPR and OpenStack, and presents Software Defined Storage volumes to OpenStack as block devices which then become available to Cinder block storage.

While ViPR Controller can provision both block and file storage, only the Cinder driver is available for OpenStack Juno release. This driver is available from both EMC Support site as well as the GitHub: <https://github.com/emcvipr/controller-openstack-cinder>

1. ViPR Components

In this lab the ViPR Controller runs as a vSphere Virtual Machine. It also uses several storage simulators, configured to run on a separate vSphere VM.



2. Lab Scenario

In this lab as a Cloud Admin you will:

- Review the Cinder driver capabilities for EMC ViPR, including extra-specs
- Review and become familiar with the ViPR Cinder drivers configuration steps
- Follow the configuration steps and implement Cinder integration with EMC ViPR Storage backend
- Create storage volume types for OpenStack Volumes and associate the Cinder Volume Types with ViPR Virtual Storage Pools
- Practice creating new volumes for different Volume Types
- Ensure that new Volumes are created using EMC ViPR User Interface (UI).

3. ViPR Cinder Driver Supported operations

ViPR Cinder driver supported operations are available at: <https://github.com/emcvipr/controller-openstack-cinder>

The EMC ViPR Cinder driver contains both an ISCSIDriver as well as a FibreChannelDriver, with the ability to create/delete and attach/detach volumes and create/delete snapshots, etc.

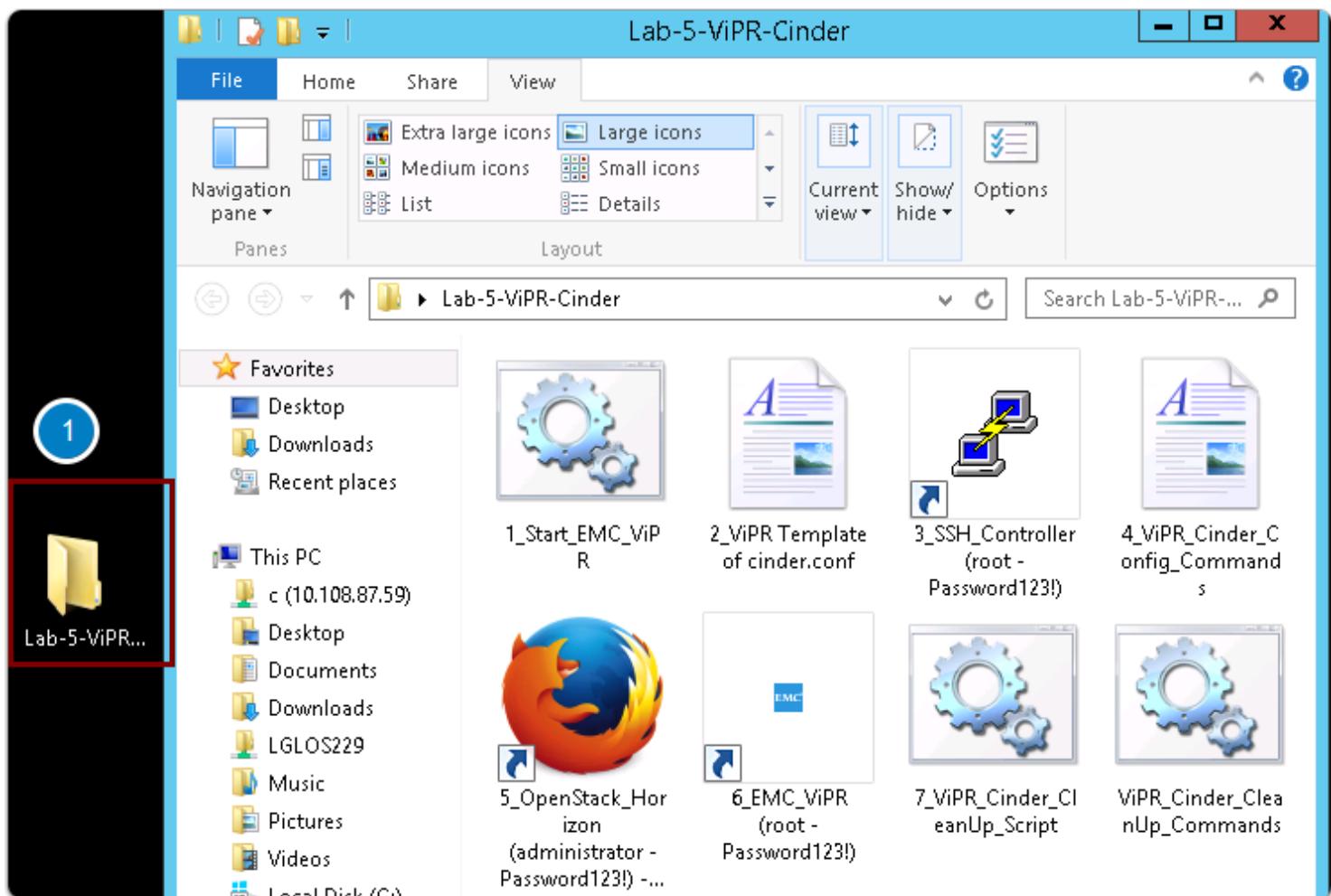
The following operations are supported:

- Create volume.
- Delete volume.
- Attach volume.
- Detach volume.
- Create snapshot.
- Delete snapshot.
- Get Volume Stats.
- Copy image to volume.
- Copy volume to image.
- Clone volume.
- Extend volume.
- Create volume from snapshot.

4. Lab-5 Folder

Please open and review Lab-5-ViPR-Cinder folder

Pay attention to the Numbers in the name of the shortcut - for your convenience you will be using these shortcuts in order.



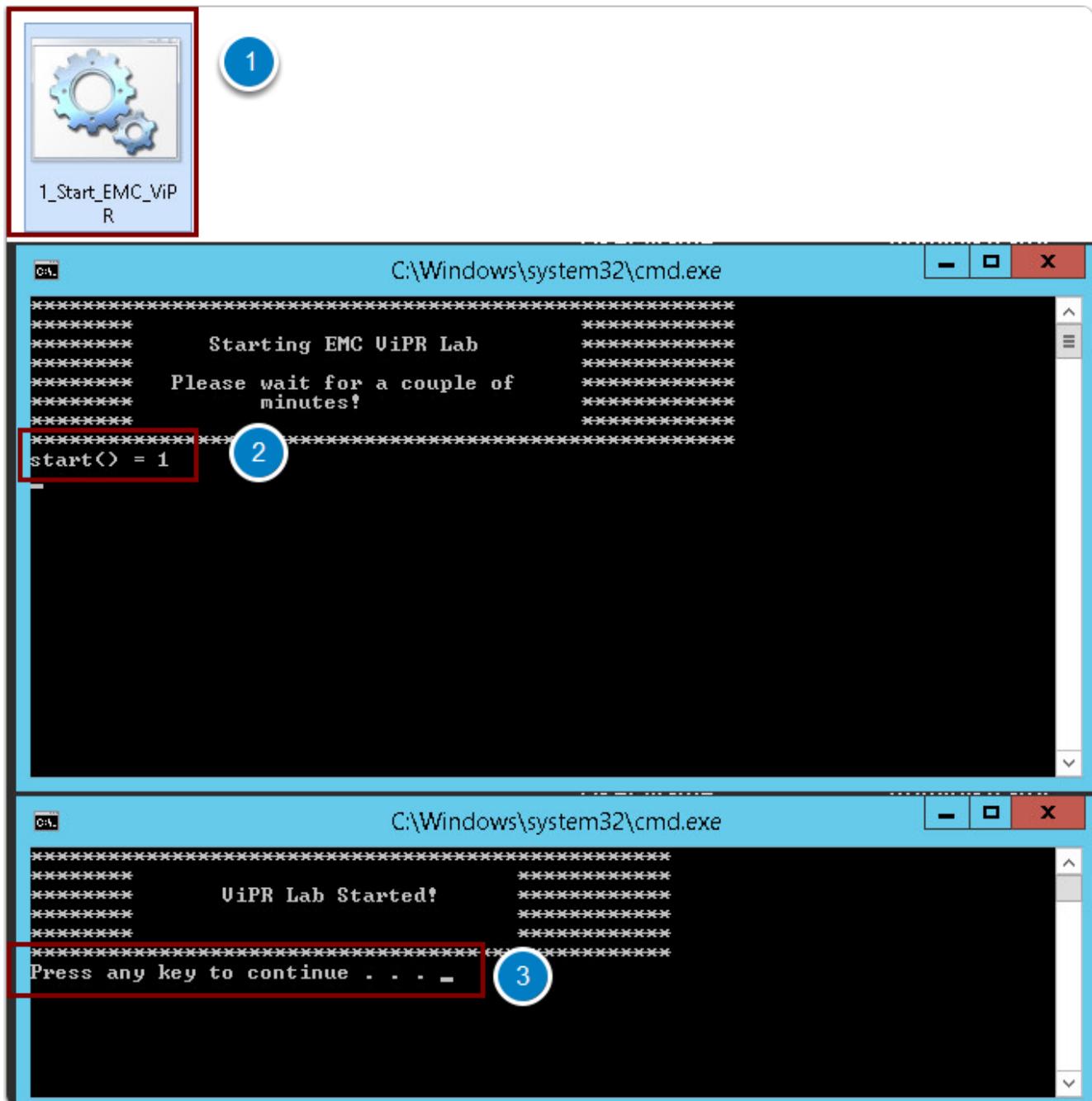
4.1 Starting ViPR

You will first start ViPR Controller and the associated storage simulators:

1. Please launch the 1_Start_EMCC_ViPR shortcut by double-clicking on it
2. Verify that the ViPR Controller has started - the script output should show: start() = 1

3. Please verify that the lab has started and press any key to close the script window and return to the Lab-4-ViPR-Cinder folder

Note: Please wait 5 minutes for starting ViPR services.

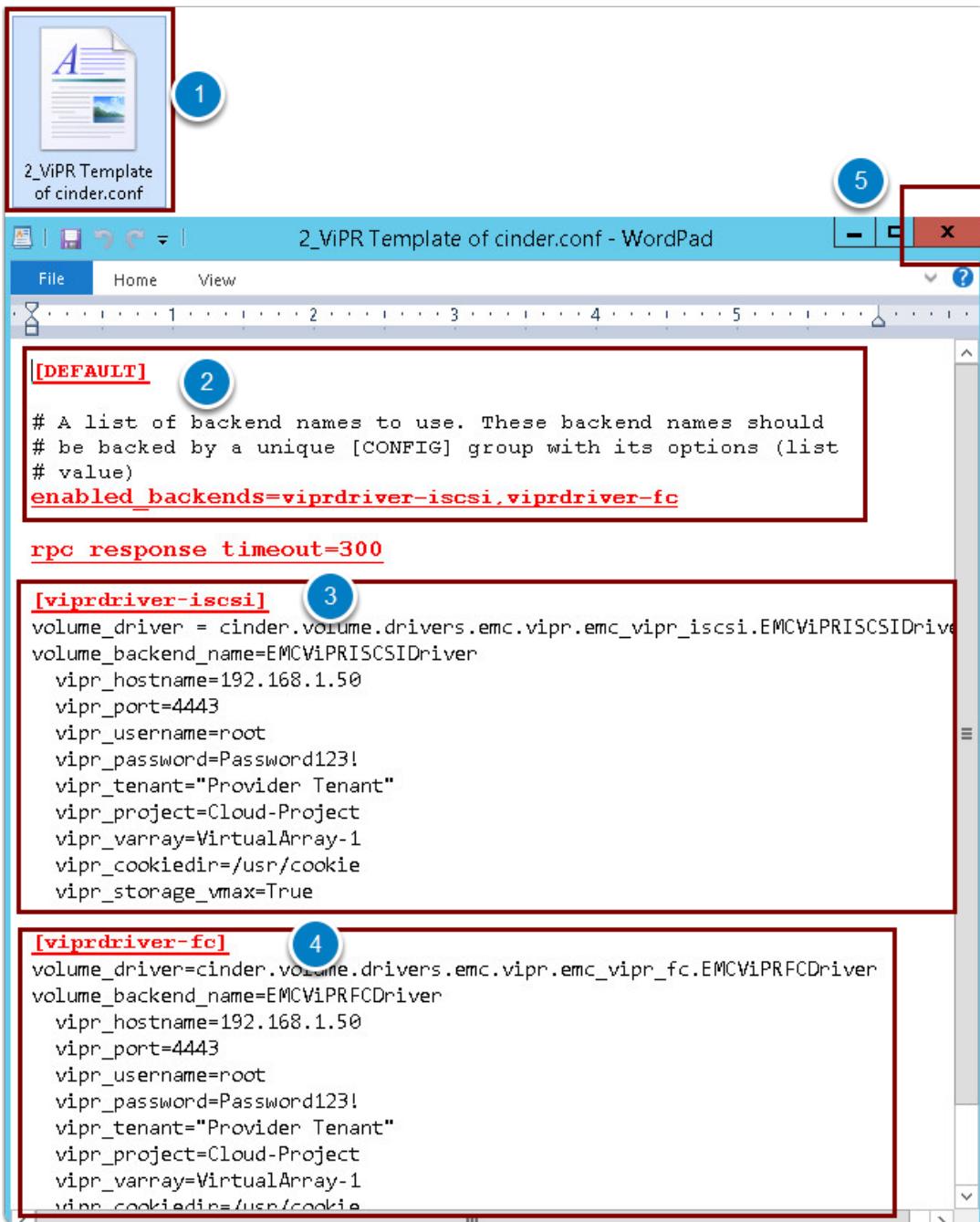


4.2 ViPR Cinder Driver Configuration Steps

ViPR Cinder driver uses ViPR CLI API to communicate with ViPR Controller. Additionally, the OpenStack Cinder service needs to be configured to recognize ViPR as a valid storage backend.

A template has been provided with all necessary configuration changes required by ViPR in the cinder.conf file for your review.

1. Please open the 2_ViPR_Template_of_Cinder.conf file in the Lab-5-ViPR-Cinder folder
2. Review the **enabled_backends** key in the **DEFAULT** section. ViPR Cinder drivers supports both **iSCSI** and **FibreChannel (FC)** storage protocols.
3. Review the **iSCSI** driver configuration section
4. Review the **FC** driver configuration section
5. Please **close** the file to continue



4.3 Using PuTTY SSH Utility

To save you some time in running this lab, a file has been created with the commands to be executed on the OpenStack Controller Node to install and configure ViPR Cinder driver.

1. Please open PuTTY shortcut to OpenStack Controller Node using shortcut called 3_SSH_Controller

2. Then open **4_ViPR_Config_Commands** file
3. Please arrange the open SSH Terminal window and the command file side by side as shown - you will be copying the commands from the file into terminal window using right click, Copy and Paste.
4. The grey-shaded areas contain commands that are necessary for installation, but take some time to complete. Please **DO NOT** run these commands, we have already executed them. The greyed-out block of commands installed the ViPR CLI API on the OpenStack Controller node and installed the ViPR Cinder driver
5. It is also recommended to copy and paste commands into **SSH Terminal** window as a **block**. Blocks are separated by blank lines. Please copy the first command block and paste into **SSH Terminal** window using right-click. We have created a copy of the **cinder.conf** file in the **/tmp/vipr** directory with all necessary settings you have reviewed earlier - this file is copied over the original **/etc/cinder/cinder.conf** file. It is also necessary to change the ownership of this file to **cinder** user
6. Please highlight, copy and paste the second command block into **SSH Terminal** window - It is necessary to restart the **Cinder API and Volume** services to read the new configuration files.
7. Please review the executed commands in the **SSH Terminal** window.

1

2

3

4

5

6

7

4_ViPR_Cinder_Config_Commands

root@node-1:~# cp /tmp/vipr/cinder.conf.vipr /etc/cinder/cinder.conf
[root@node-1 ~]# chown cinder:cinder /etc/cinder/cinder.conf
[root@node-1 ~]# ls -l /etc/cinder
total 184
-rw----- 1 cinder cinder 2281 Feb 21 05:29 api-paste.ini
-rw----- 1 cinder cinder 78046 Mar 14 07:07 cinder.conf
-rw----- 1 cinder cinder 77330 Feb 21 07:38 cinder.conf.orig
-rw-r--r-- 1 cinder cinder 224 Feb 23 21:22 cinder_sca_leio.config
-rw-r----- 1 cinder cinder 3200 Feb 22 07:59 policy.json
-rw-r----- 1 cinder cinder 3200 Feb 21 07:39 policy.json.orig
-rw-r----- 1 root cinder 942 Dec 16 14:51 rootwrap.cnf
drwxr-xr-x 2 root root 4096 Feb 22 04:55 rootwrap.d
drwxr-xr-x 2 cinder root 4096 Dec 16 14:52 volumes
[root@node-1 ~]# service openstack-cinder-api restart
Stopping openstack-cinder-api: [OK]
Starting openstack-cinder-api: [OK]
[root@node-1 ~]# service openstack-cinder-volume restart
Stopping openstack-cinder-volume: [OK]
Starting openstack-cinder-volume: [OK]
[root@node-1 ~]#

DO NOT RUN! These commands have already been executed
Installing ViPR CLI API
mkdir /tmp/vipr/cli
cd /tmp/vipr/cli
wget https://vipr.vlab.local:4443/cli --no-check-cert
tar -xvf ./viprcli.tar
python setup.py install
python
>>> import viprcli
>>> exit
mkdir /usr/lib/python2.6/site-packages/cinder/volume
cp /tmp/vipr/driver /usr/lib/python2.6/site-packages

Please copy and paste commands below into SSH
One block at the time
cp /tmp/vipr/cinder.conf.vipr /etc/cinder/cinder.conf
chown cinder:cinder /etc/cinder/cinder.conf
ls -l /etc/cinder

service openstack-cinder-api restart
service openstack-cinder-volume restart

source /root/adminrc

cinder type-create "ViPR Base Pool"
cinder type-key "ViPR Base Pool" set ViPR:VPOOL="ViPR-cinder"
cinder type-key "ViPR Base Pool" set volume_backend_name

cinder type-create "ViPR High Availability"
cinder type-key "ViPR High Availability" set ViPR:VPOOL="ViPR-HA"
cinder type-key "ViPR High Availability" set volume_backend_name="ViPR-HA"

4.4 Creating ViPR Volume Types

Now that ViPR Cinder driver is installed and configured, we will create ViPR Volume Types in OpenStack Cinder service.

1. Please copy the next block in the **4_ViPR_Cinder_Config.Commands** file and paste it into SSH Terminal window using right-click
2. Please review the **cinder type-create** command and created ViPR volume types
3. Please close the SSH terminal window and the ViPR Config commands file- we will now begin using the **OpenStack Horizon UI**.

The screenshot displays two windows side-by-side. On the left is a terminal window titled 'root@node-1:~' showing the execution of several 'cinder type-create' commands to establish different volume types using the ViPR driver. The output includes the creation of 'ViPR Base Pool', 'ViPR High Availability', and 'ViPR High Performance FC'. On the right is a Microsoft Word document titled '4_ViPR_Cinder_Config_Commands - Word...' which contains the same commands. A red box highlights the terminal output, and a blue circle labeled '2' is placed over the terminal window. Another blue circle labeled '3' is placed over the Microsoft Word document. A third blue circle labeled '1' is placed over the first line of the command block in the Word document. A large red box at the bottom right of the Word document encloses the final two lines of the command block, with the text '# Please open the OpenStack Horizon UI now!' written below it.

```

root@node-1:~# source /root/adminrc
root@node-1:~#
[root@node-1 ~]# cinder type-create "ViPR Base Pool"
:cinder type-key "ViPR Base Pool" set ViPR:VPOOL="ViPR-Block-Pool"
:cinder type-key "ViPR Base Pool" set volume_backend_name=EMCViPRISCSIDriver

:cinder type-create "ViPR High Availability"
:cinder type-key "ViPR High Availability" set ViPR:VPOOL="ViPR-VPLEX-HA-Pool"
:cinder type-key "ViPR High Availability" set volume_backend_name=EMCViPRISCSIDriver

:cinder type-create "ViPR High Performance FC"
:cinder type-key "ViPR High Performance FC" set ViPR:VPOOL="ViPR-High-Performance-FC"
:cinder type-key "ViPR High Performance FC" set volume_backend_name=EMCViPRFCDriver

+-----+
| ID | Name |
+-----+
| aaeef531a-a7cb-41ed-922b-c3e00b2df799 | ViPR Base Pool |
+-----+
[root@node-1 ~]# cinder type-key "ViPR Base Pool" set ViPR:VPOOL="ViPR-Block-Pool"
[root@node-1 ~]# cinder type-key "ViPR Base Pool" set volume_backend_name=EMCViPRISCSIDriver
[root@node-1 ~]# cinder type-create "ViPR High Availability"
:cinder type-key "ViPR High Availability" set ViPR:VPOOL="ViPR-VPLEX-HA-Pool"
:cinder type-key "ViPR High Availability" set volume_backend_name=EMCViPRISCSIDriver

:cinder type-create "ViPR High Performance FC"
:cinder type-key "ViPR High Performance FC" set ViPR:VPOOL="ViPR-High-Performance-FC"
:cinder type-key "ViPR High Performance FC" set volume_backend_name=EMCViPRFCDriver

+-----+
| ID | Name |
+-----+

```

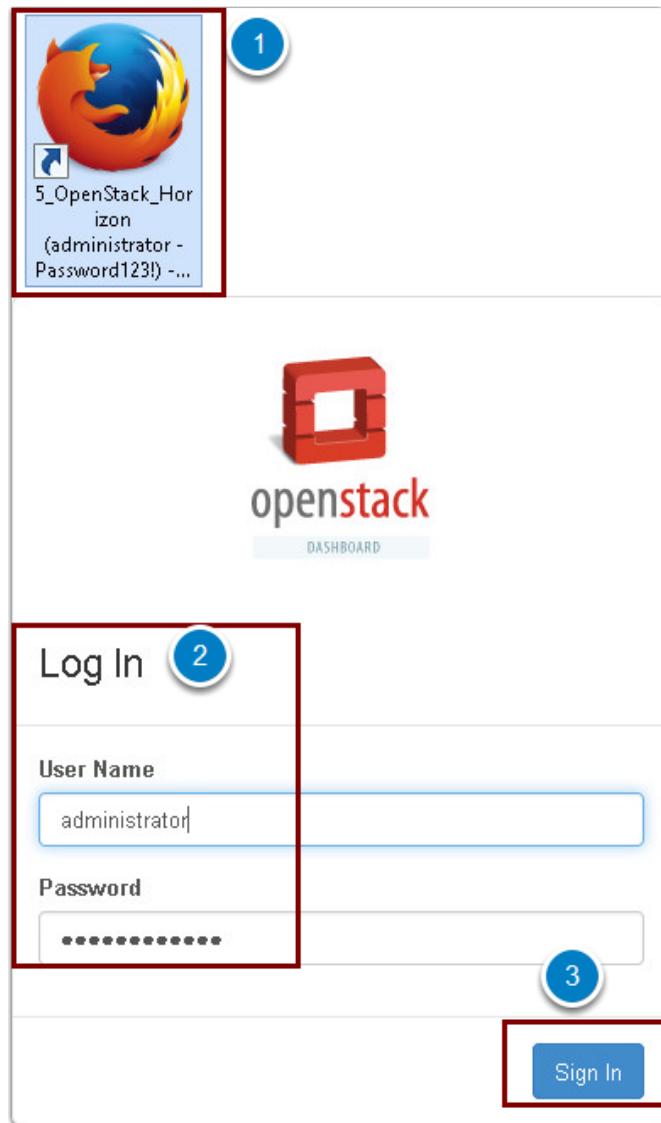
5. Using OpenStack Horizon

1. Return to the Lab-5-ViPR-Cinder folder and launch the **5_OpenStack_Horizon(administrator - Password123!)** shortcut

2. The user credentials should be cached in the browser window. If needed please type the credentials:

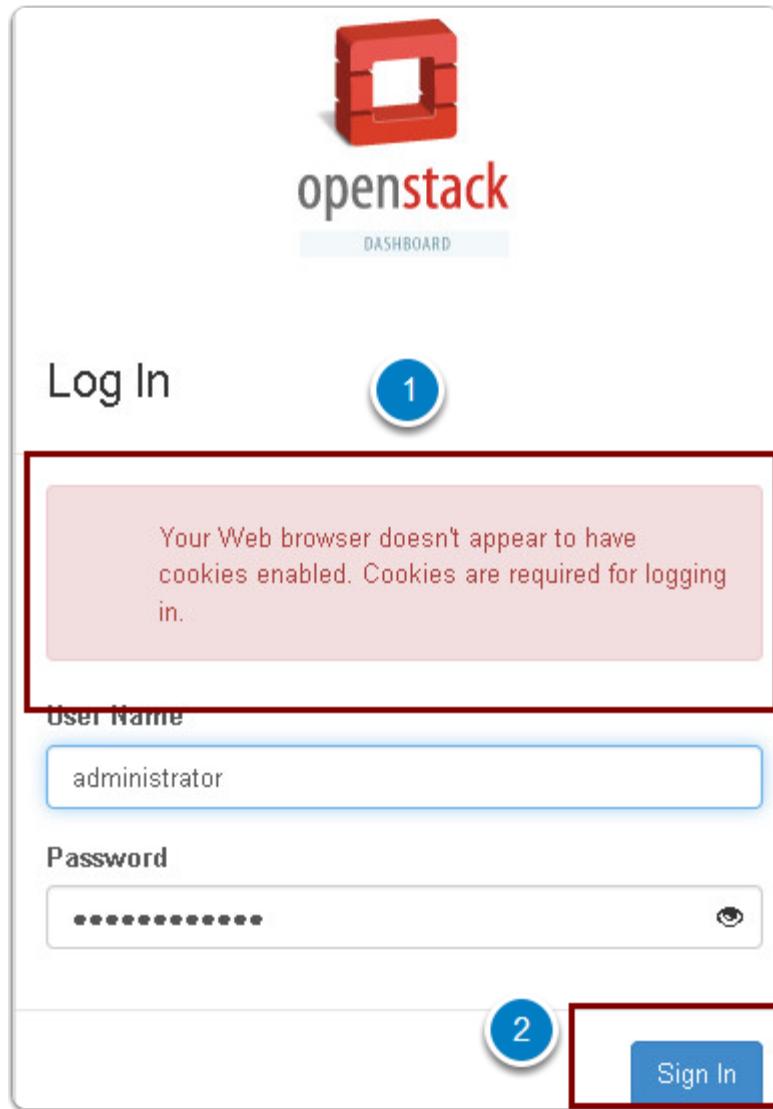
- User Name: *administrator*
- Password: *Password123!*

3. Click **Sign In** to continue



5.1 Browser Cookie Warning

1. Occasionally you may see a browser cookie warning
2. Please disregard it and simply click on **Sign In** again to continue



5.2 Reviewing ViPR Volume Types

1. Please review the Dashboard for OpenStack project - **cloud**
2. Please select **Volumes** In OpenStack Horizon Admin menu on the left hand side
3. Click on the **Volume Types** tab

4. Review your created ViPR volume types. Note: You may see other volume type created earlier in the previous labs - please ignore them.
5. Please click on the Extra Specs menu to see the volume type details.

The screenshot shows the OpenStack interface with the following details:

- Left Sidebar:** Shows 'Project' and 'Admin' dropdown menus, and 'System' under 'Admin'. The 'Instances' and 'Volumes' links are highlighted with red boxes and blue circles numbered 2 and 3 respectively.
- Overview Page:** Displays 'Usage Summary' and a table for 'Active Instances'. The table includes columns: Project Name, VCPUs, Disk, RAM, VCPU Hours, and Disk GB Hours. One entry for 'cloud' is shown.
- Volumes Page:** Shows tabs for 'Volumes' (selected), 'Volume Types', and 'Volume Snapshots'. A table lists 'Volume Types' with columns: Name, Associated QoS Spec, and Actions (containing 'View Extra Specs'). Three items are listed: 'ViPR Base Pool', 'ViPR High Availability', and 'ViPR High Performance FC'. The 'ViPR Base Pool' row is highlighted with a red box and blue circle 5.

5.3 Volume Type Extra Specs

1. Please review the Extra Specs for a newly created Volume Type: ViPR Base Pool
 - ViPR Virtual Storage Pool (VSP)
 - ViPR Cinder driver - iSCSI

2. Please select the Project tab - you will now create new ViPR volume.

Volume Type: ViPR Base Pool

Volume Type Extra Specs

Key	Value	Actions
ViPR:VPOOL	ViPR-Block-Pool	Edit
volume_backend_name	EMCViPRISCSIDriver	Edit

Displaying 2 items

5.4 Project View

1. Please select Volumes tab
2. You will now create a new Volume using one of the ViPR Volume types - please click on Create Volume menu

Volumes

Name	Description	Size	Status	Type	Attached To	Availability Zone	Bootable	Encrypted	Actions
No items to display.									

+ Create Volume

5.5 Creating ViPR Volume

Please provide new ViPR Volume details:

1. Volume Name - ViPR-iSCSI
2. Provide a Description as you see fit
3. Leave Volume Source field with default value
4. Select a Volume Type from the drop-down menu - ViPR Base Pool
5. Specify the Volume size - 10GB
6. Leave Availability Zone field with default value - Any Availability Zone
7. Click on Create Volume to continue

Create Volume

Volume Name * 1 ViPR-iSCSI

Description 2 ViPR iSCSI Volume

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 3

Type

- No volume type
- No volume type
- 4 ViPR Base Pool
- ViPR High Availability
- ViPR High Performance FC

Size (GB) * 5 10 7

Create Volume

5.6 New ViPR Volume

1. Please verify that the new ViPR Volume is available
2. Next you will review this new Volume in ViPR Controller UI.

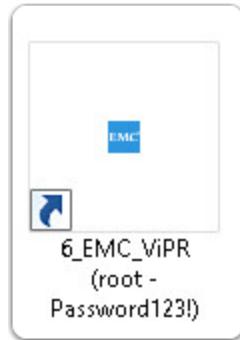
Volumes

Volumes										
		Filter		Filter		Create Volume		Delete Volumes		
<input type="checkbox"/>	Name	Description	Size	Status	Type	Attached To	Availability Zone	Bootable	Encrypted	Actions
<input type="checkbox"/>	ViPR-iSCSI	ViPR iSCSI Volume	10GB	Available	ViPR Base Pool		nova	No	No	Edit Volume ▼

Displaying 1 item

6. EMC ViPR

Now review your newly created volume in ViPR - please double-click on the 6_EMCA_ViPR (root - Password123!) icon in the Lab-5-ViPR-Cinder folder.

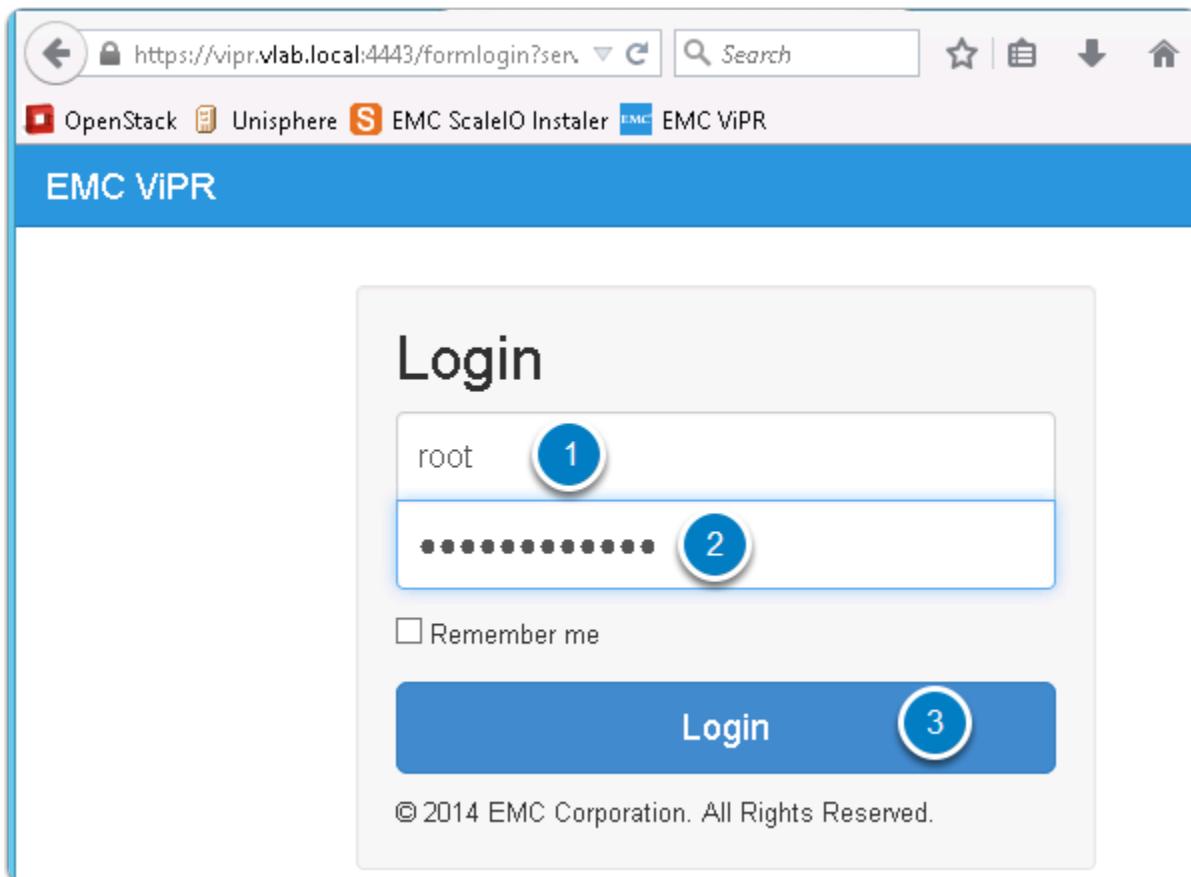


6.1 Logging into ViPR

Please login into EMC ViPR:

1. User name: *root*

2. Password: *Password123!*
3. Click Login to continue



6.2 ViPR Dashboard

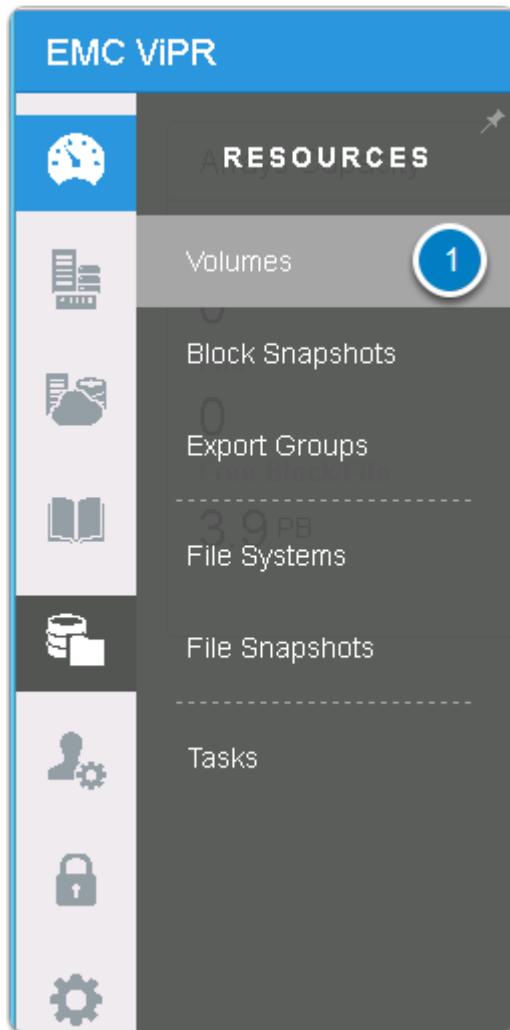
Please review the ViPR Controller administrator Dashboard

1. Select **Resources** icon from the left-hand menu - you will now review the newly created ViPR Volume.

The screenshot shows the EMC ViPR Dashboard interface. On the left, there is a vertical sidebar with icons for different management categories: Arrays Capacity, Physical Assets, Version and License, System Health, Controller Status, and a gear icon for Settings. The 'Resources' icon, which is a cylinder with a bar chart, is highlighted with a red box. The main content area is divided into three columns: 'Arrays Capacity' (showing 0 Block and 0 File), 'Physical Assets' (listing Storage Systems (14), Storage Providers (3), Data Protection Systems (1), Fabric Managers (1), Networks (6), Compute Images, Vblock Compute Systems, Hosts (7), Clusters, and VMware vCenters), and 'Version and License' (showing Controller version vipr-2.2.0.0.1043, Licenses status as Licensed, and System Health with a Stable status). At the bottom, it says 'Last Updated 2015-02-27 17:32'.

6.3 Block Volumes

Select **Volumes** from the fly-out menu



6.4 Tenant Project Volumes

1. Select Cloud-Tenant from the drop-down menu list
2. Review the Cloud-Project
3. Review the newly created volume - ViPR-iSCSI
4. Review ViPR Virtual Storage Array - VirtualArray-1
5. Review the ViPR Virtual Storage Pool associated with the OpenStack Cinder volume type
6. Please click on the volume to view additional details

The screenshot shows the EMC ViPR interface for managing storage volumes. The top navigation bar includes 'EMC ViPR', a user icon with '0' notifications, 'Help', and a 'root' user indicator. The main search bar contains 'Search...'. On the left, there's a sidebar with icons for Tenant, Volume, and Virtual Array.

The main area displays a table of volumes. The first row shows the configuration for creating a new volume:

Tenant:	Cloud-Tenant	Project:	Cloud-Project
Volume:	Cloud-Tenant		
Volume Name:	ViPR-iSCSI	Virtual Array:	VirtualArray-1
Virtual Pool:	ViPR-Block-Pool	Protocols:	iSCSI,FC

Numbered callouts point to specific fields:

- 1: Tenant dropdown (highlighted)
- 2: Project dropdown (highlighted)
- 3: Volume Name input field (highlighted)
- 4: Virtual Array dropdown (highlighted)
- 5: Virtual Pool dropdown (highlighted)
- 6: Volume Name input field (highlighted)

Below the table, pagination controls show 'First', '←', '1', '→', and 'Last', with '0 entries selected' and 'Showing 1 to 1 of 1 entries'.

6.5 Volume Details

Please review the ViPR Volume details - click **More Details** to continue

 **ViPR-iSCSI**
Volume

Actions  Delete Volume ▾

WWN:	60000000000000000000000000000002
Size:	10.00 GB Provisioned / 10.00 GB Requested
Virtual Array:	 VirtualArray-1
Virtual Pool:	 ViPR-Block-Pool

[More Details](#)

► [Exports](#)
► [Snapshots](#)
► [Full Copies](#)
► [Continuous Copies](#)

Tasks

Name	Progress	State	Start	Elapsed
CREATE VOLUME	<div style="width: 100%;">100%</div>	✓ Complete	15 hours ago	a few seconds

First ← 1 → Last Showing 1 to 1 of 1 entries

6.6 Additional Volume details

Please review the additional volume details

 **ViPR-iSCSI**
Volume

WWN: 60000000000000000000000000000000
Size: 10.00 GB Provisioned / 10.00 GB Requested
Virtual Array:  VirtualArray-1
Virtual Pool:  ViPR-Block-Pool
Storage System: SYMMETRIX+999595867618
Device Label: ViPR-iSCSI
Native ID: 00001
ID: urn:storageos:Volume:88898f1e-3e63-46bc-988c-1bc2ba07fd15:vdc1
Created: Mar 14th 2015, 3:47:13 am
Tags: OpenStack:updated_at:2015-03-14 07:46:59
OpenStack:attach_status:detached
OpenStack:project_id:ddfb7c98096241168c86d0298a99b414
OpenStack:volume_type_id:aaef531a-a7cb-41ed-922b-c3e00b2df799
OpenStack:created_at:2015-03-14 07:46:59
OpenStack:user_id:97eefa7c07d1410d912715b0e642573
OpenStack:availability_zone:nova
OpenStack:display_description:ViPR iSCSI Volume
OpenStack:replication_status:disabled
OpenStack:volume_type:<cinder.db.sqlalchemy.models.VolumeTypes object at 0x52bf3d0>
OpenStack:display_name:ViPR-iSCSI
OpenStack:_sa_instance_state:<sqlalchemy.orm.stateInstanceState object at 0x52235d0>
OpenStack:host:node-1.vlab.local@viprdriver-iscsi#EMCVIPRISCSIDriver
OpenStack:size:10
OpenStack:id:ed5429a6-89d2-4653-9ce1-3d99dd10825
OpenStack:scheduled_at:2015-03-14 07:46:59
[Less Details](#)

6.7 Cleaning Up ViPR Lab

This lab is now complete. Please close the ViPR browser window and return to the Lab-5-ViPR-Cinder folder.

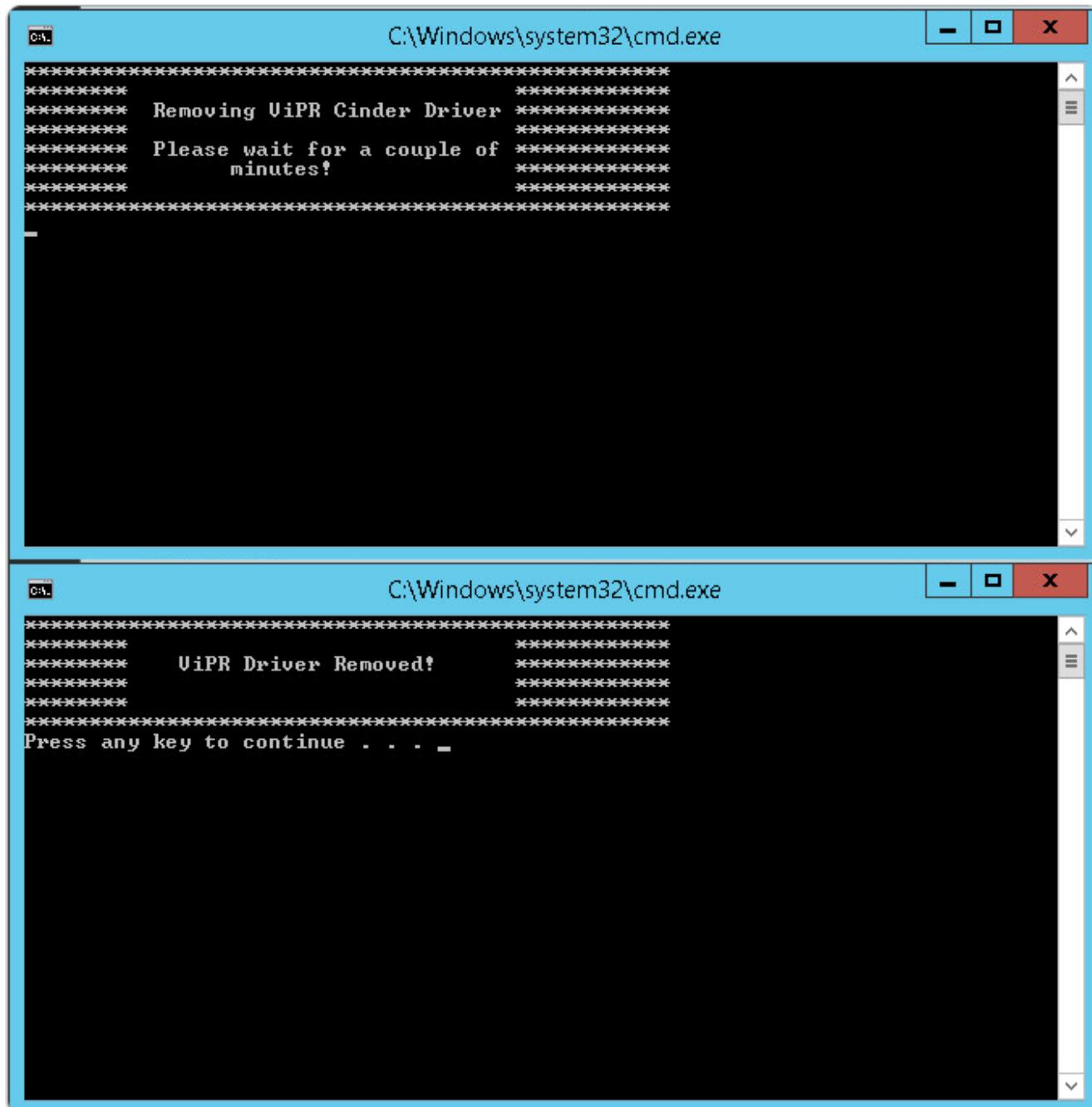
Please launch the shortcut called **7_ViPR_Cinder_CleanUp_Script** by double-clicking on it. This script will stop ViPR and Storage simulator VM's and revert OpenStack Cinder configuration to default in preparation to the next lab.



6.8 Completing the ViPR Lab

Once the ViPR Controller VM has been stop and ViPR Cinder driver removed, press any key to continue - this lab is now complete!

Please close the command window as needed.



7. Conclusion

Congratulations! In this lab you, wearing the hat of Code Nebulous Cloud Administrator, have successfully accomplished the following:

- Deployed and configured ViPR Cinder driver to use the ViPR Software-defined block storage as a Cinder backend
- Created Cinder volume types associated with the ViPR virtual storage pools
- Create a new ViPR block volume.

Now the Code Nebulous Cloud Users can begin building and using the applications running on the Cloud-Workload instance and store its data on the persistent block volume.

Note: In the interests of time you will not be logging into Cloud-Workload instance to format and mount the newly created volume.

This lab is now complete. Please close the following windows as you will not be using them any longer:

- Lab-5-ViPR-Cinder folder

Please proceed to the next lab, or select other labs from the list.

Optional Lab 6 - OpenStack Monitoring and Reporting

Lab 6 - OpenStack Monitoring and Reporting

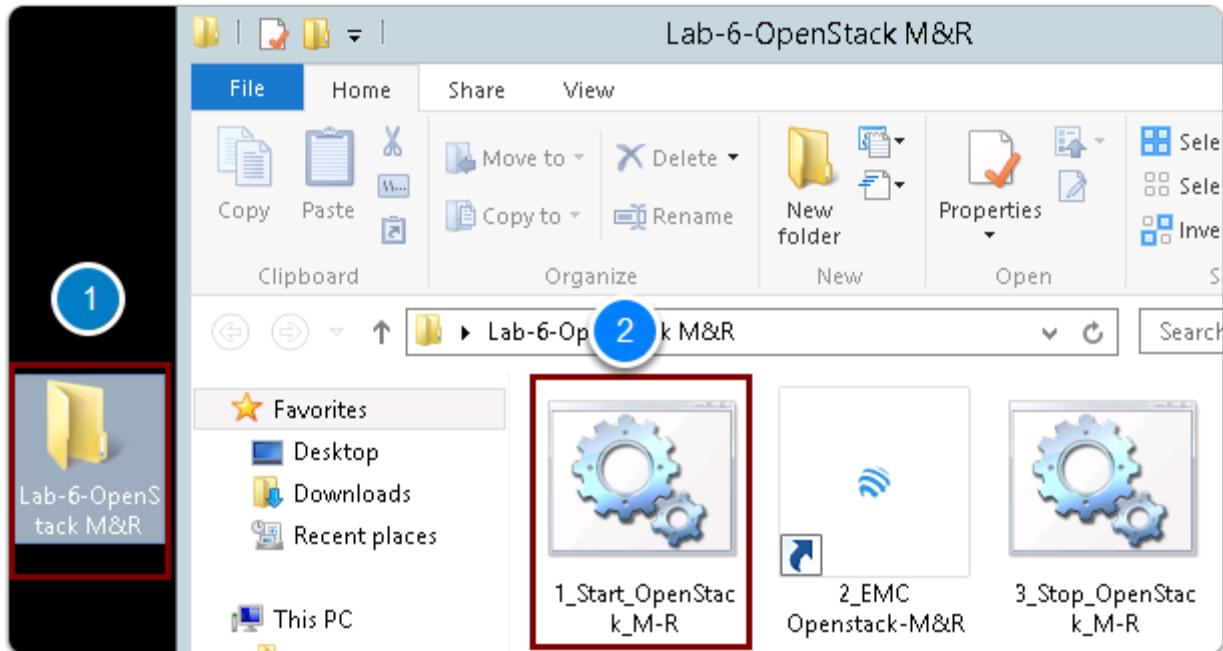
ViPR SRM provides actionable insight into end-to-end data path relationships and topology to understand service dependencies. It displays detailed performance trends across the data path to identify the impact storage has on applications. With addition of the OpenStack Monitoring & Reporting (M&R) Solution Pack, ViPR SRM can now enable Cloud Admin to gather Capacity, Performance and Chargeback data and provide comprehensive reports to the tenants.

In this lab, an historical data sample has been collected in the demo environment. While similar, this sample data may not match your specific lab environment.

1. Lab-6: OpenStack M&R Folder

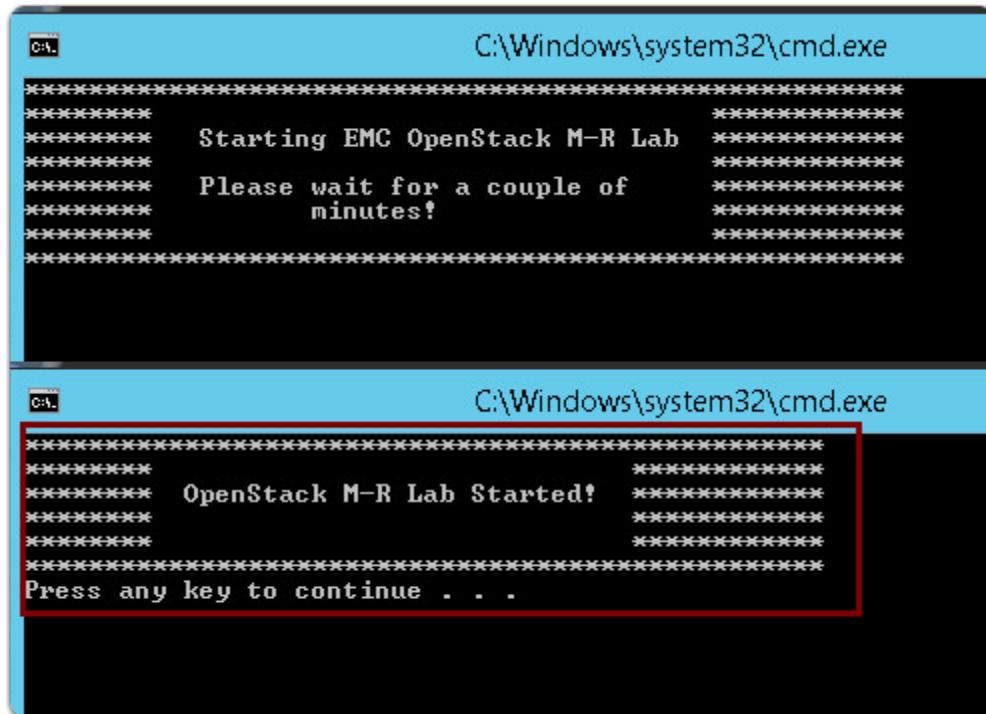
For your convenience, all necessary shortcuts used in this lab have been provided in a Desktop folder called "Lab-6-OpenStack_M&R"

1. Please open the Lab-6-OpenStack_M&R folder
2. Launch the lab by double-clicking on the shortcut 1_Start_OpenStack_M-R - this script will power on the OpenStack_M&R nested Virtual Machine.



2. Powering Up OpenStack_M&R VM

Please verify that the OpenStack_M&R VM has started and press any key to continue.



3. Using OpenStack M&R UI

1. Please launch the shortcut to OpenStack M&R from the Lab-6-OpenStack_M&R folder. Note: It may take a couple of minutes for the OpenStack M&R to start. If your browser times out, simply refresh the browser window:
2. Login with credentials:
 - User Name: *admin*
 - Password: *changeme*
3. Click **Sign In** to continue



4. OpenStack M&R Dashboard

1. Please navigate to the Report Library

2. Then expand the OpenStack Reports

Welcome (logged as admin) Saturday, March 14, 2015

All

All

March 2015, Friday 13 » Saturday 14, 8:13 PM EDT | **Last 1 Day**

EDIT MODE

Dashboards View high-level summaries for storage and networks.

Explore 

Operations Monitor and validate alerts, compliance, service level agreements, and situations to watch.

Planning 

Report Library Browse reports from installed SolutionPacks.

Report Library 

1

2

Scheduled Reports
Stored Reports
Favorite Reports
Dashboards
Explore
Operations
Planning
Report Library
EMC M&R Health
OpenStack
Inventory
Capacity Dashboard
Chargeback Dashboard
Project Report

5. Capacity Dashboard

1. Please select Capacity Dashboard

All >> Report Library >> OpenStack

Report Library / OpenStack

March 2015, Friday 13 » Saturday 14, 8:15 PM EDT | Last 1 Day

OpenStack

4 elements found, displaying all elements.

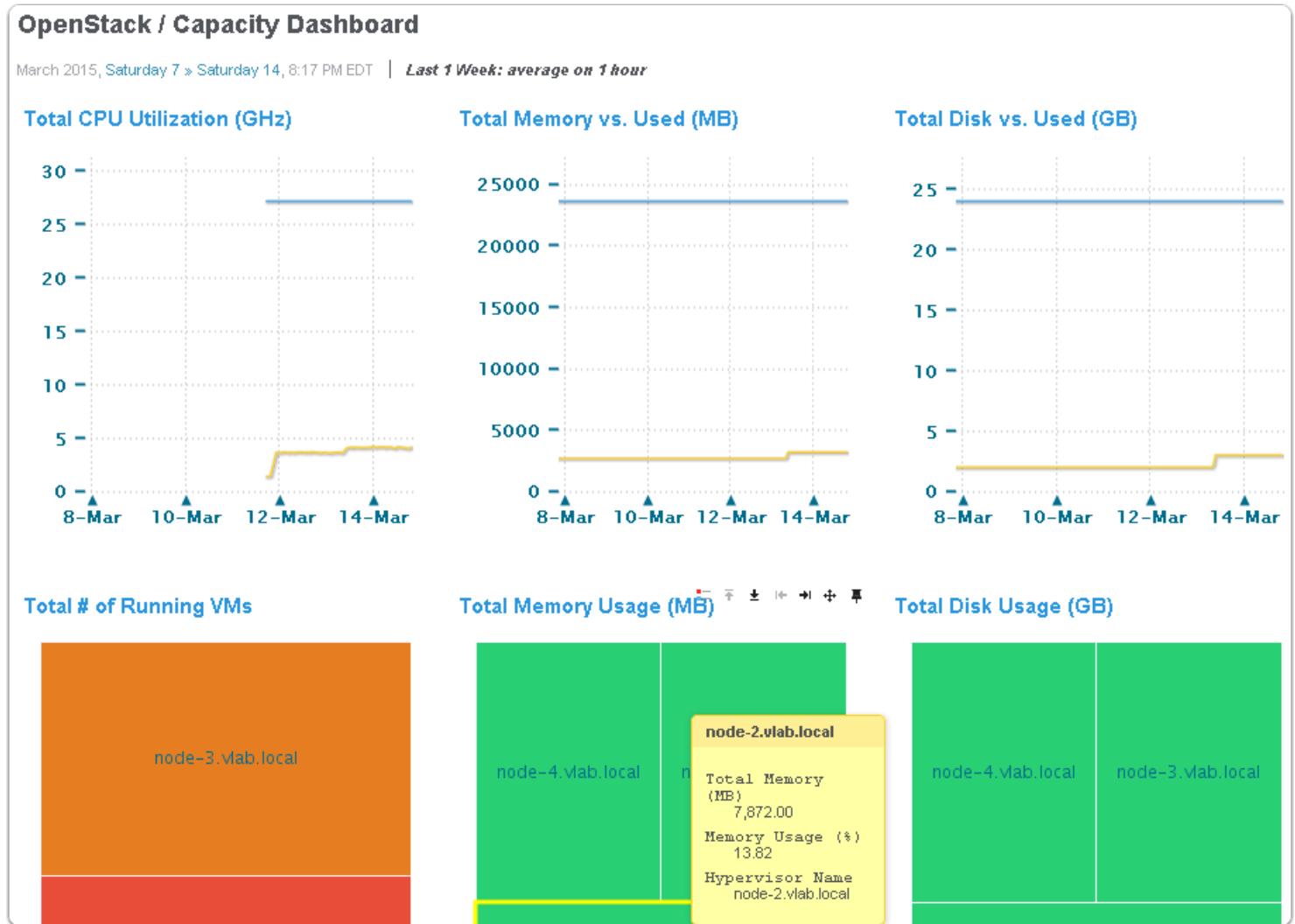
Available reports
Capacity Dashboard
Chargeback Dashboard
Inventory
Project Report

Report generated in 0.159s.

1

5.1 OpenStack Capacity Reports

Please review the Capacity reports



6. Chargeback Dashboard

Please expand and select the Chargeback Dashboard

The screenshot shows the OpenStack Chargeback Dashboard interface. On the left, a sidebar menu includes options like Scheduled Reports, Stored Reports, Favorite Reports, Dashboards, Explore, Operations, Planning, Report Library (with sub-options for EMC M&R Health, OpenStack, Inventory, Capacity Dashboard), and Chargeback Dashboard, which is highlighted with a red box. The main dashboard area displays two pie charts: 'Project Cost Distribution (%)' and 'Resource Cost Distribution (%)'. Below the charts are two tables: 'Project Costs Report' and 'Resource Costs Report'.

Project Cost Distribution (%)

Category	Percentage
marketing	14.28%
qa	85.72%

Resource Cost Distribution (%)

Resource	Percentage
CPU	> 0%
STORAGE	4.51%
NETWORK	8.89%
MEMORY	86.6%

Project Costs Report

5 elements found, displaying all elements.

Project Name	Costs (\$ Current 1 Month)	Costs (\$ YTD)	Costs (\$ Last 1 Year)
engineering			
qa			
devops	110.3	110.3	110.3
marketing	46.8	46.8	46.8

Resource Costs Report

4 elements found, displaying all elements.

Resource Name	Cost (\$ Current Month)	Cost (\$ YTD)	Cost (\$ Last 1 Year)
CPU			
MEMORY			
NETWORK			
STORAGE	99.00	1,101.00	1,101.00

7. Project Report

1. Please review the Project Report
2. Select the "cloud" project form the table

The screenshot shows the EMC vLab interface with the following details:

- Left Sidebar:** Includes links for Scheduled Reports, Stored Reports, Favorite Reports, Dashboards, Explore, Operations, Planning, Report Library (selected), OpenStack (selected), Inventory, Capacity Dashboard, Chargeback Dashboard, and Project Report (selected).
- Header:** Shows the path All > Report Library > OpenStack > Project Report, the date range March 2015, Friday 13 > Saturday 14, 8:20 PM EDT, and Last 1 Day.
- Table:** Displays a list of projects with the following columns: Project Name, Groups, Domain, and Project ID. The 'cloud' project is highlighted with a red box and a blue circle containing the number '2'. The table data is as follows:

Project Name	Groups	Domain	Project ID
cloud	All VM Guests	Default	ddfb7c98096241168c66d0298a99b414
devops	All VM Guests	Default	235dc6f5263493d997f347ceb51af24
marketing	All VM Guests	Default	8ef24175c3374158a59731dade07124e
cloud	All VM Guests	Default	2e638e2c841e43989a38058fa8b7c363
qa	All VM Guests	Default	9cbefcbde9cf41cdb9fd494cbf23fe56
engineering	All VM Guests	Default	52081ee039b642cb91e70c749f9ddf6e

Report generated in 0.227s.

7.1 Project Details

Please review the Project details for cloud project

The screenshot shows the EMC vLab interface with the following details:

- Left Sidebar:** Includes sections for Scheduled Reports, Stored Reports, Favorite Reports, Dashboards, Explore, Operations, Planning, Report Library (with sub-options like EMC M&R Health, OpenStack, Inventory, Capacity Dashboard, Chargeback Dashboard), and Project Report (with a selected item: "cloud, 2e638e2c841e43989a38058fa8b7c363").
- Header:** Shows "EDIT MODE" with a search bar, "DISPLAY", "EXPORTS", and "TOOLS".
- Project Report Title:** "Project Report / cloud, 2e638e2c841e43989a38058fa8b7c363".
- Time Range:** March 2015, Friday 13 > Saturday 14, 8:22 PM EDT | Last 1 Day: average on real-time.
- VM Attributes:** One element found. Table:

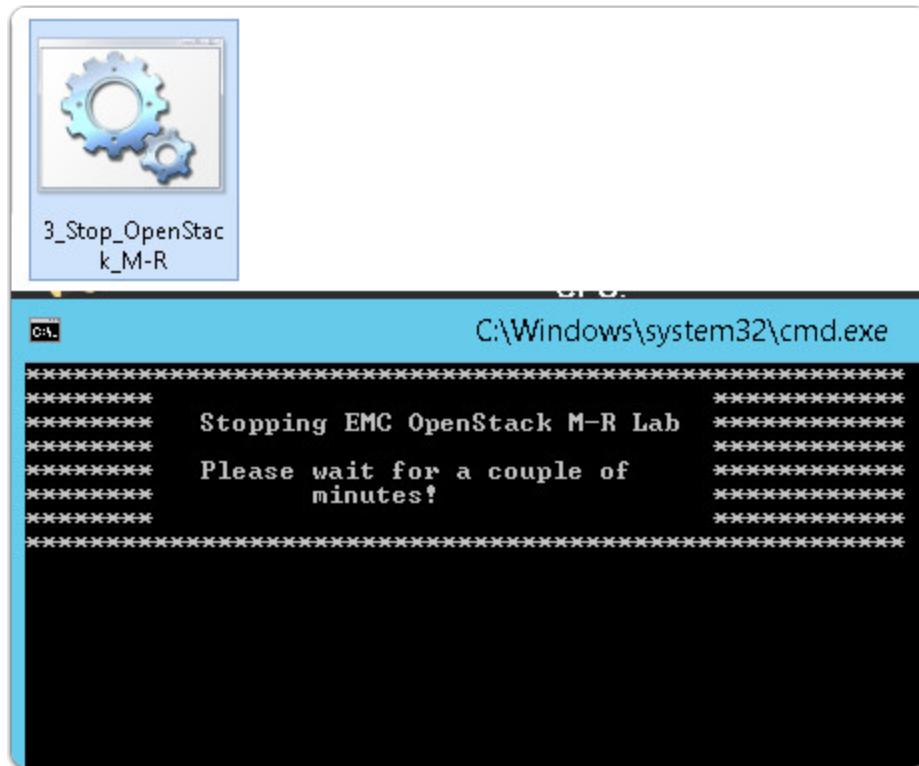
VM Name	IP	Flavor	Network	Owner	Created	Total Memory (MB)	Used Memory (MB)	RSS Memory (MB)	VCPU	Availability (%)	Project
Cloud-Workload	192.168.1.111.2	m1.micro	net04	administrator	2015-03-07T00:41:45Z				1	100	cloud
- Volume Attributes:** Two elements found, displaying all elements. Table:

Volume Name	Type	Description	Created	Availability Zone	Is Bootable?	Attached VM ID	Size (GB)	Status	Project
ScaleIO-ThinVolume	ScaleIO_Thin_Pool	ScaleIO-ThinVolume, ScaleIO Thin Volume	2015-03-07T01:43:57, 2015-03-07T02:46:43	nova	false		8	available, error	cloud
VNX-ThinVolume	VNX-ThinVolume	VNX-ThinVolume, VNX Thin Volume	2015-03-07T02:23:04, 2015-03-07T02:39:30, 2015-03-07T01:42:46	nova	false	d660a909-8c47-42d2-87ac-2ae0cdcd005	2	error, in-use	cloud
- Network Attributes:** Two elements found, displaying all elements. Table:

Network Name	Is Shared?	Availability (%)	Project	OpenStack	Data Grp
net04	true	100	cloud	192.168.1.101	OPENSTACK-NETWORK

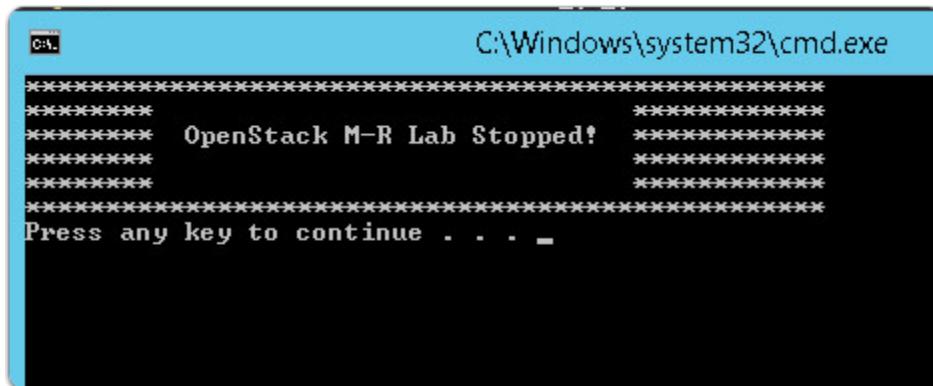
8. Completing the OpenStack M&R Lab

1. Please launch the 3_Stop_OpenStack_M-R shortcut from the Lab-6-OpenStack_M&R folder
2. Verify that the lab has stopped (OpenStack_M&R VM has been powered off)



8.1 Lab 6 Completed

Press any key to continue



9. Conclusion

Congratulations! In this lab you, wearing the hat of Code Nebulous Cloud Administrator, have successfully accomplished the following:

- Collected and reviewed EMC OpenStack Monitoring & Reporting data for Tenant Chargeback and Cloud Capacity planning.

This lab is now complete. Please close the following windows as you will not be using them any longer:

- Lab-6-OpenStack_M&R folder
- OpenStack M&R Browser Window

Please proceed to the next lab, or select other labs from the list.

Troubleshooting

Troubleshooting & Tips

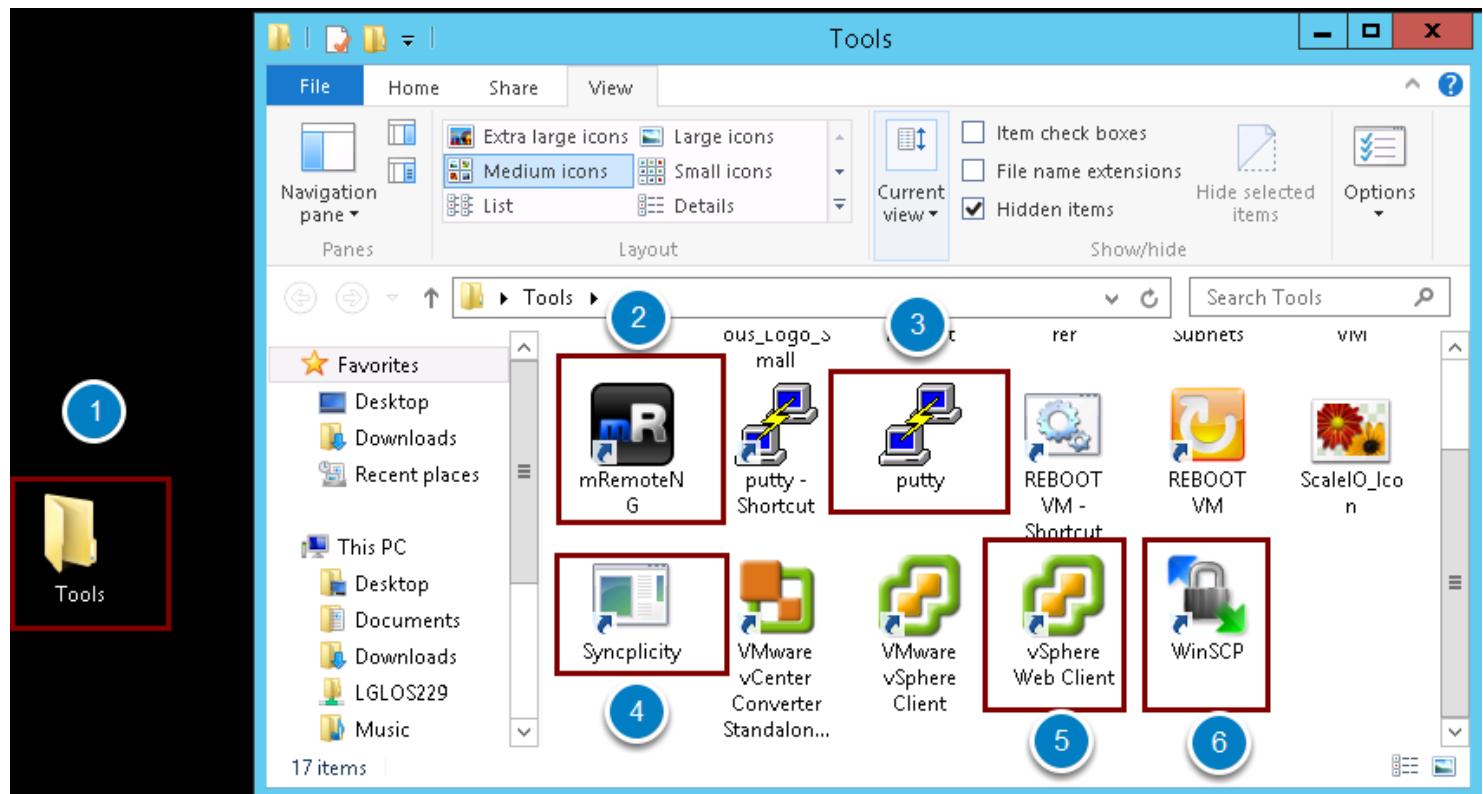
This section provides general suggestions and resolutions to potential issues that may arise in this lab.

This area is for general problems that may occur at any point during the lab.

1. Tools Folder

For your convenience, various tools have been provided that may be helpful in your troubleshooting.

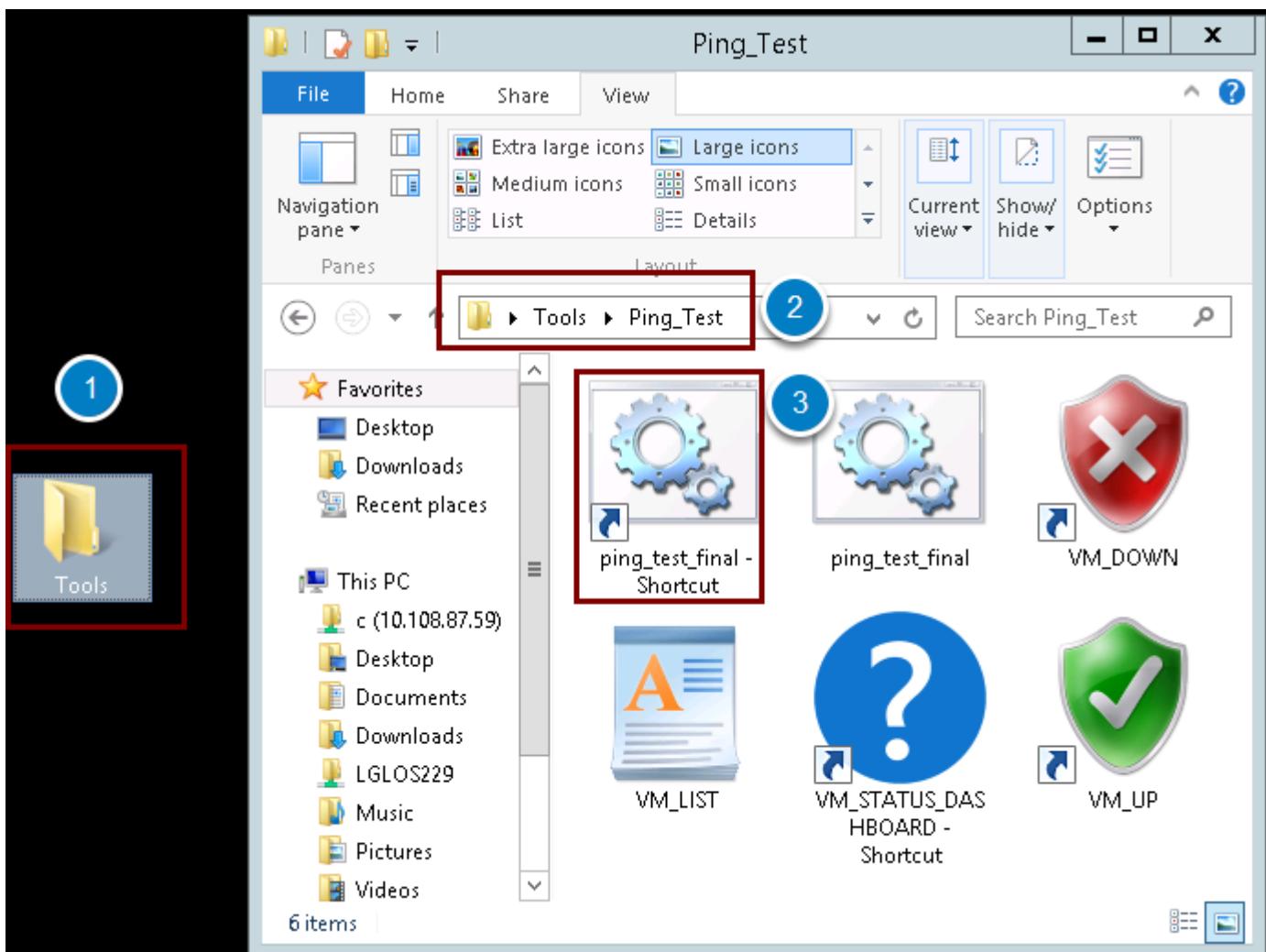
1. These tools are located in the **Tools** folder on the Launchpad Desktop
2. You can use mRemoteNG to access the Windows VM's in this lab
3. You can use Putty SSH terminal utility to access Linux VM's in this lab
4. You can use Syncplicity to upload custom content to and from this lab
5. You can use vSphere Web Client to check status and power on / off of the "nested" VM's
6. You can use WinSCP to copy the files to and from the Linux VM's in this lab



2. VM's Availability

This lab contains several Virtual Machines used at different points during this lab. You can initiate a script that verifies VM availability by executing ping command.

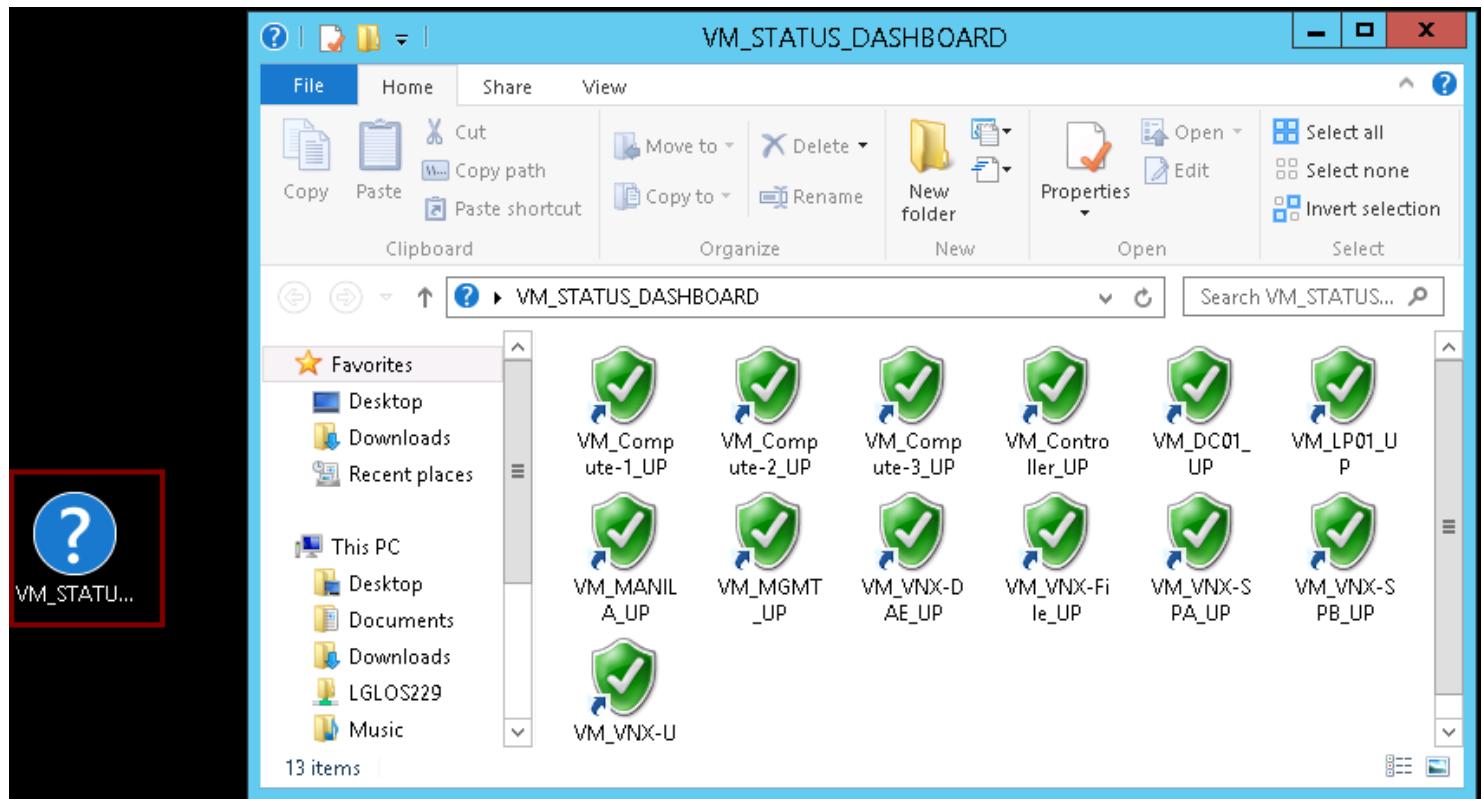
1. Open the Tools folder located on the Launchpad Desktop
2. Navigate to the Ping_Test folder
3. Launch the ping_test_final script



2.1 VM Status

You can verify the status of these VM's by checking the "VM Status Dashboard" Folder on the Launchpad Desktop.

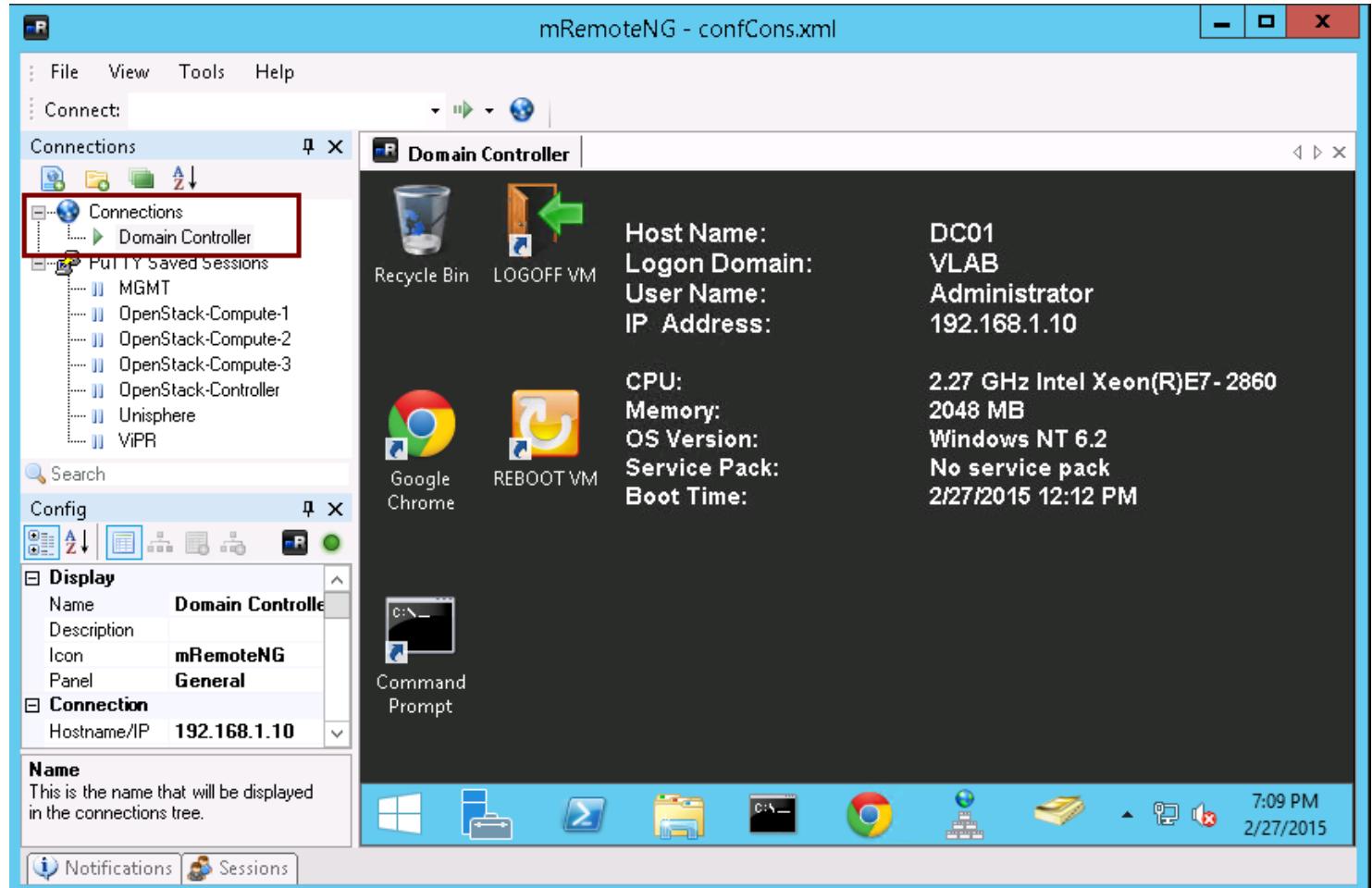
All shield icons should be **green**



3. Using mRemoteNG

You can use mRemoteNG to access the other Windows VM's in this lab.

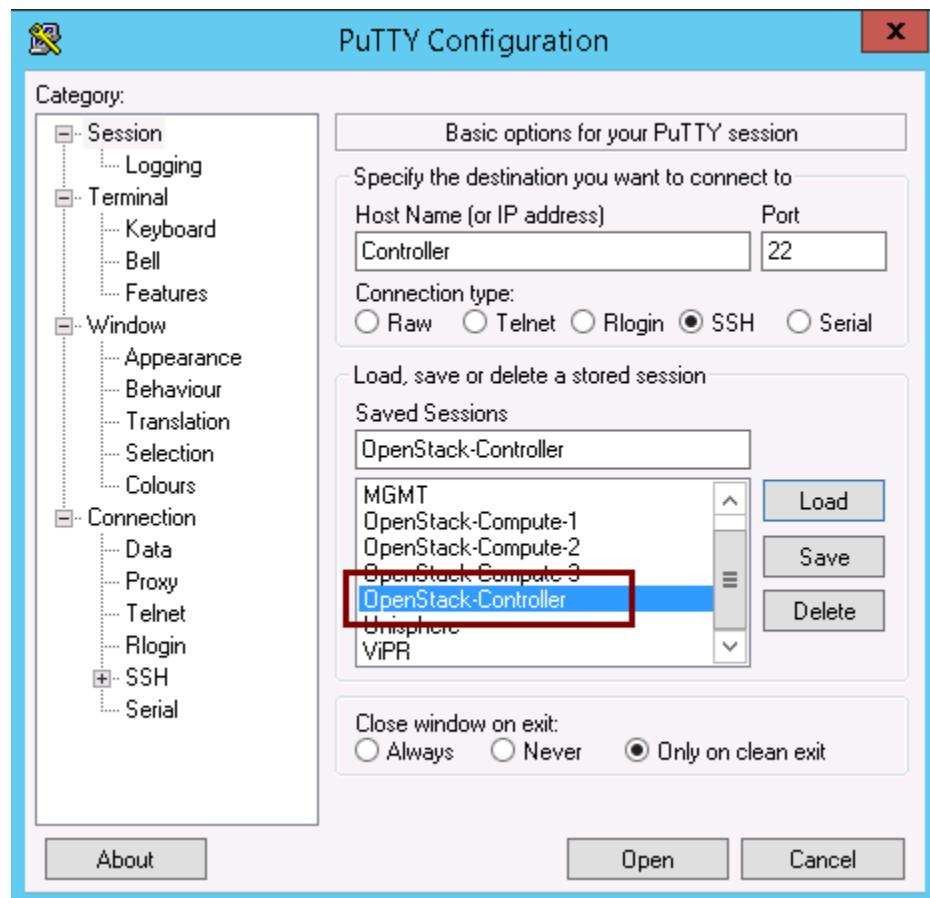
Pre-defined sessions have been created that allow you to access the Windows VM's without typing their IP address.



4. Using PuTTY SSH Terminal

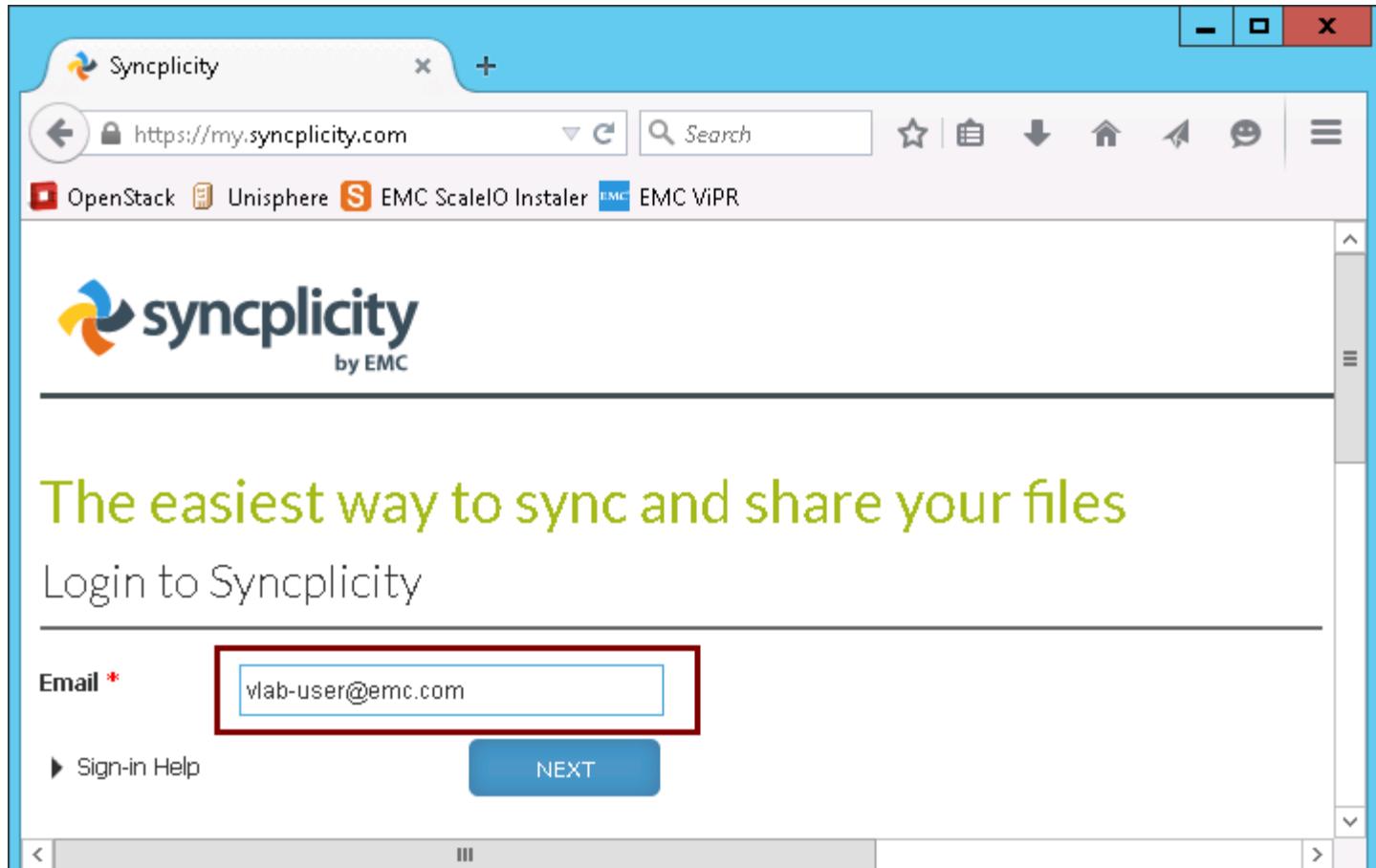
You can use PuTTY SSH Terminal utility to access other Linux VM's in this lab. Pre-defined sessions have been created that allow you access to the VM without typing it's IP address

All SSH credentials are the same: *root - Password123!*



5. Using Syncplicity

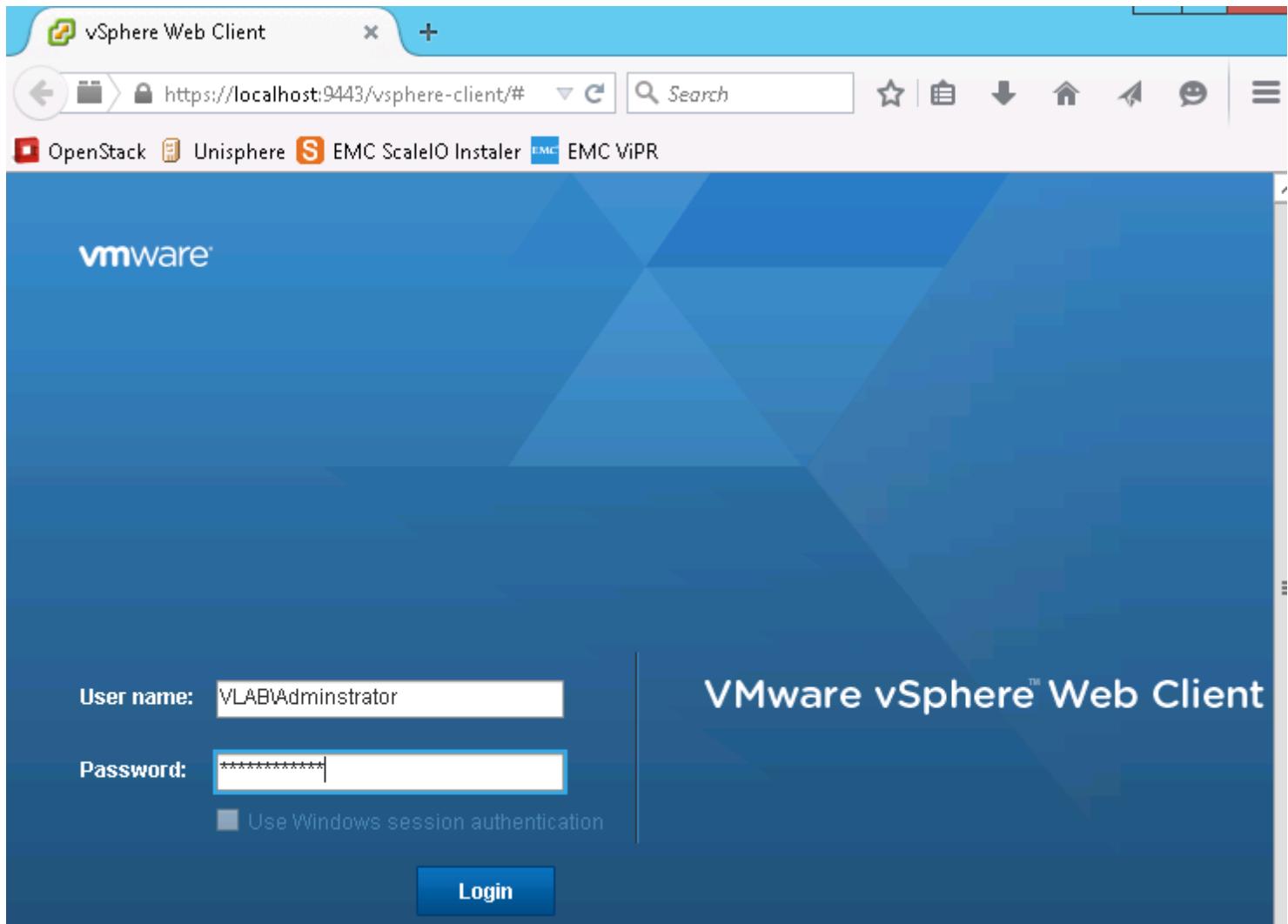
You can use Syncplicity to upload your custom content to and from this lab using your Syncplicity account:



6. Using vSphere Web Client

You can use vSphere Web Client to check status and power on / off of the "nested" VM's

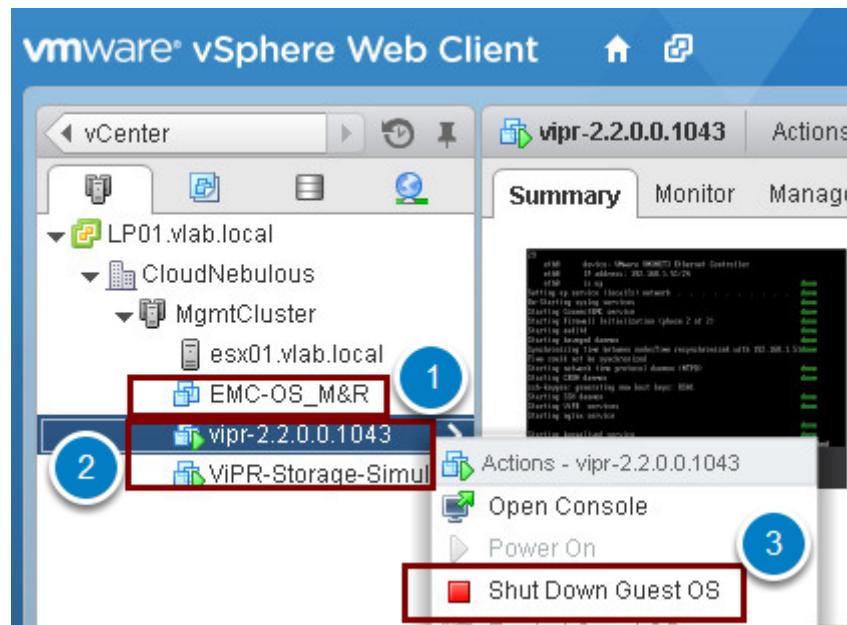
Login with credentials: **VLAB\Administrator - Password123!**



6.1 Nested VM's

While troubleshooting Lab 5: ViPR with Cinder, or Lab 6: OpenStack Monitoring & Reporting (M&R) labs, please make sure to check the status of the nested VM's. Having the unnecessary VM's powered on during other labs will lead to oversubscription of the ESX resources (CPU and RAM) and can cause timeouts in lab activities:

1. EMC-OS_M&R VM should be powered on only during Lab 6: OpenStack Monitoring & Reporting (M&R)
2. ViPR-2.2.0.0.1043 and ViPR-Storage-Simulator VM's should be powered on only during Lab 5: ViPR with Cinder
3. You can power off unnecessary VM's using right-click menu "Shut Down Guest OS" command



Conclusion

Conclusion

Congratulations!

Bart, Stephan's manager has received fantastic feedback from Code Nebulous' Cloud Operations Team after partaking in Stephan's knowledge transfer session. The feedback highlighted the ease of bringing up an OpenStack cloud with support for building and managing multiple tenants in the heterogeneous cloud infrastructure throughout its entire lifecycle.

Additionally, Stephan had conducted a knowledge transfer session for Code Nebulous Tenants. The feedback highlighted the ease of use and quick and easy way to provisioning and manage Infrastructure services and all around good user experience.