



Global Leader in Software Defined Storage.

NexentaEdge

Installation Guide 1.1 FP3

Date: June, 2016

Subject: NexentaEdge Installation Guide

Software: NexentaEdge

Software Version: 1.1.0 FP3

Part Number: 2000-nedge-1.1-FP3-000010-A

Copyright © 2016 Nexenta Systems™, ALL RIGHTS RESERVED

Notice: No part of this publication may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying and recording, or stored in a database or retrieval system for any purpose, without the express written permission of Nexenta Systems (hereinafter referred to as “Nexenta”).

Nexenta reserves the right to make changes to this document at any time without notice and assumes no responsibility for its use. Nexenta products and services only can be ordered under the terms and conditions of Nexenta Systems’ applicable agreements. All of the features described in this document may not be available currently. Refer to the latest product announcement or contact your local Nexenta Systems sales office for information on feature and product availability. This document includes the latest information available at the time of publication.

Nexenta, NexentaStor, NexentaEdge, NexentaFusion, and NexentaConnect are registered trademarks of Nexenta Systems in the United States and other countries. All other trademarks, service marks, and company names in this document are properties of their respective owners.

This document applies to the following product versions:

Product	Versions supported
NexentaEdge™	1.1 FP3
Ubuntu Linux	14.04.3 LTS (3.19 Linux kernel)
CentOS	7.2, 7.3
Red Hat Enterprise Linux	7.2, 7.3
Chef	11.0
OpenStack Swift OpenStack Cinder	Icehouse, Juno, Kilo, Liberty, Mitaka
Docker	1.9
Flocker	1.8

Contents

1	Installation Overview	1
	About This Installation Guide	1
	Components of a NexentaEdge Cluster	1
	NexentaEdge Deployment Options	4
	Docker/Baremetal Deployment	4
	Deployment Using Packages in an ISO File	4
	Juju Charm Deployment	4
	NexentaEdge Installation Procedure	5
2	Prerequisites	6
	Networking Requirements	6
	Switch Hardware Requirements	6
	Replicast Network Requirements	6
	Data and Gateway Node Requirements	6
	Hardware Requirements	7
	Software Requirements	7
	Deployment Workstation Requirements	8
	Requirements for the Chef Environment	8
3	Installing NexentaEdge	9
	Before You Begin	9
	Planning Server Node Deployment	9
	Network Switch Configuration	10
	Downloading the NexentaEdge Deployment and Administration Tools	10
	Extracting the NEDEPLOY and NEADM Archives	11
	Preparing Storage Devices for NexentaEdge Deployment	11
	Deploying NexentaEdge	12
	RedHat/CentOS Considerations	12
	Using the Command Line	13
	Using the NEDEPLOY Wizard	16
	Initializing the NexentaEdge Cluster	19

Installing the NexentaEdge License	19
Online License Installation	20
Offline License Installation	20
Creating a Logical Cluster	21
Creating a Tenant	21
Creating a Bucket	22
4 Configuring NexentaEdge Storage Service Groups	23
About NexentaEdge Storage Service Groups	23
Before You Begin	23
iSCSI Deployment Example	24
Additional iSCSI Management Commands	26
Integration with OpenStack Cinder	27
Amazon S3 Deployment Example	27
Additional Configuration	28
OpenStack Swift Deployment Example	28
Additional Configuration	30
Configuring Zoning for Data Nodes	30
5 Integrating NexentaEdge with OpenStack Cinder	32
Overview	32
Setting Up an iSCSI Storage Service for Cinder	32
Enabling the NexentaEdge Cinder Plugin	32
6 Integrating NexentaEdge with OpenStack Horizon	35
Overview	35
Prerequisites	35
Installing the Horizon Dashboard Plugin	35
Registering the NexentaEdge Cluster as a Swift Storage System	37
7 Deploying NexentaEdge from an ISO File	41
Overview	41
Dependencies	41
Deployment Procedure	42

Preface

This documentation presents information specific to Nexenta products. The information is for reference purposes and is subject to change.

Intended Audience

This documentation is intended for Object Storage Administrators and Network Administrators and assumes that you have a working knowledge of UNIX. The document also assumes that you have experience with data storage concepts, such as object storage, ZFS, iSCSI, NFS, CIFS, and so on.

Documentation History

The following table lists the released revisions of this documentation.

Table 1: Documentation Revision History

Revision	Date	Description
2000-nedge-1.0-000010-A	July, 2015	1.0 GA release
2000-nedge-1.0-000010-B	September, 2015	1.0.1 release
2000-nedge-1.1-000010-A	February, 2016	1.1.0 GA release
2000-nedge-1.1-FP1-000010-B	May, 2016	1.1.0 FP1 release
2000-nedge-1.1-FP2-000010-A	May, 2016	1.1.0 FP2 release
2000-nedge-1.1-FP3-000010-A	June, 2016	1.1.0 FP3 release

Comments

Your comments and suggestions to improve this documentation are greatly appreciated. Send any feedback to doc.comments@nexenta.com and include the documentation title, number, and revision. Refer to specific pages, sections, and paragraphs whenever possible.

Installation Overview

This chapter includes the following topics:

- [NexentaEdge Installation Procedure](#)
- [Components of a NexentaEdge Cluster](#)
- [NexentaEdge Installation Procedure](#)

About This Installation Guide

This guide contains the procedures for installing the basic components of a NexentaEdge cluster. It details the requirements for the servers and switches, and contains procedures for configuring the servers to be part of a NexentaEdge cluster.

After performing the tasks in this guide, you will have a NexentaEdge cluster that is capable of accepting requests from clients running storage services that NexentaEdge supports, including iSCSI block storage, OpenStack Cinder, and Amazon S3.

See the *NexentaEdge User Guide* for information about adding servers and storage devices to your cluster and other administrative tasks.

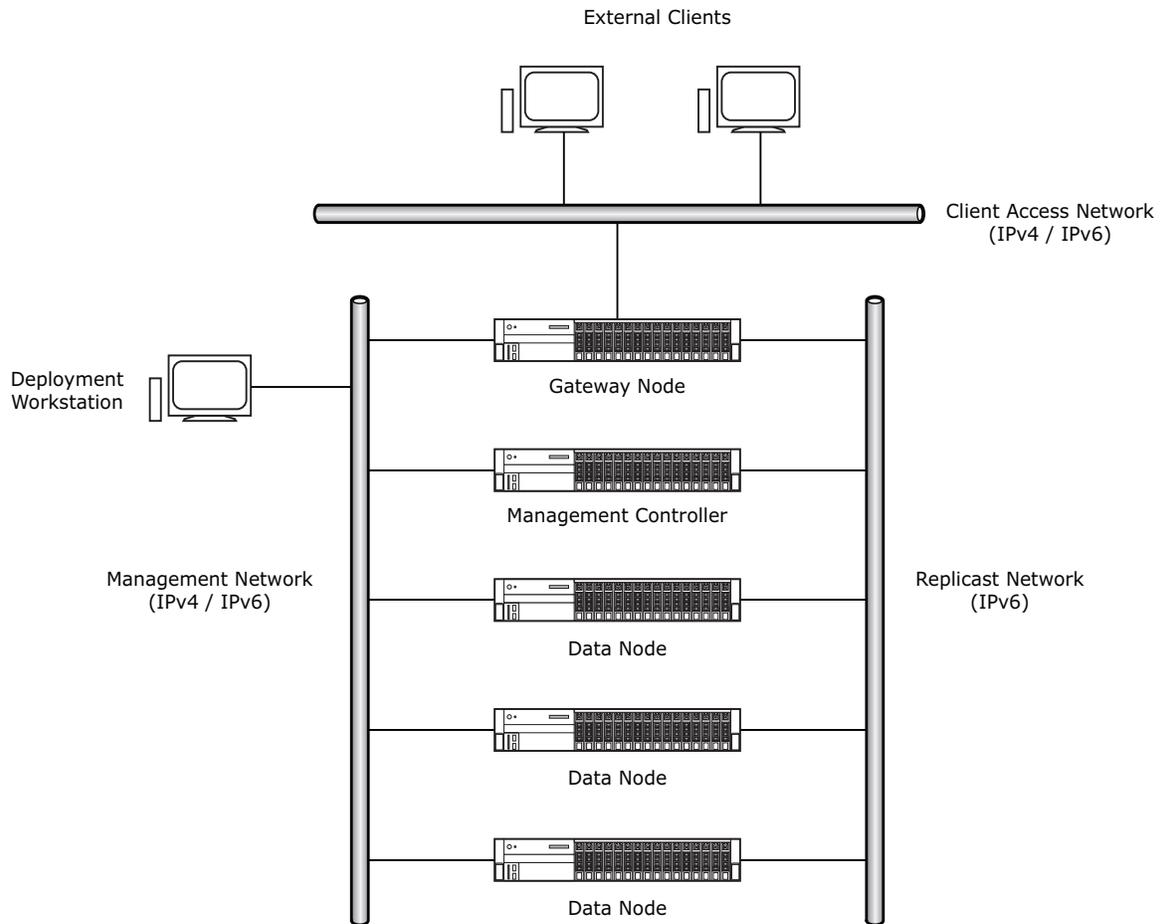
Components of a NexentaEdge Cluster

From a physical perspective, a NexentaEdge cluster is a collection of server devices connected via a high-performance 10 Gigabit switch. From a logical perspective, a NexentaEdge cluster consists of data nodes and gateway nodes that communicate over a Replicast network. The cluster provides storage services over an external network using the protocols that NexentaEdge supports, including OpenStack Swift, OpenStack Cinder, Amazon S3, and iSCSI. Note that the term “NexentaEdge cluster” in this manual refers to a logical NexentaEdge cluster.

A NexentaEdge deployment consists of a single physical cluster and one or more logical clusters. Each logical cluster may have multiple namespaces configured for different tenants.

[Figure 1-1](#) shows the components of a NexentaEdge cluster.

Figure 1-1: NexentaEdge Cluster Components



A NexentaEdge cluster consists of the following components. A given device may have multiple roles assigned to it; for example, a server may be configured as a data node, gateway node and management controller.

- Data nodes

The data nodes collectively provide the storage for the NexentaEdge cluster. Objects are broken into chunks and distributed across the data nodes using the Replicast protocol. The set of data nodes where the chunks are stored or retrieved is determined based on server load and capacity information.

Data nodes may be configured with interfaces to an IPv6 Replicast network for data distribution and storage and to an IPv4 network (either an external network or dedicated management network) for initial configuration with the NEDEPLOY tool and subsequent administration with the NEADM tool.

After initial configuration, data nodes require only a connection to the Replicast network, since administration of the data nodes is done by the deployment workstation via the management controller node, which has connectivity to both the management network and the Replicast network.

- Gateway nodes

Gateway nodes provide the connection between external clients and the data stored in the NexentaEdge cluster. Gateway nodes accept and respond to client requests, translating them into actions performed in the NexentaEdge cluster. Gateway nodes are provisioned with interfaces to the external network, Replicast network, and management network (if different from the external network)

When you configure a NexentaEdge cluster, you indicate which storage service(s) you want to provide for a given tenant, then specify which should be the gateway nodes for that cluster/tenant/service combination.

- Replicast network

The Replicast network is an isolated IPv6 VLAN used for communication and data transfer among the data nodes and gateway nodes in the NexentaEdge cluster. The Replicast protocol provides the means for efficient storage and retrieval of data in the cluster.

- Deployment workstation

The deployment workstation is the system from which you deploy and configure the NexentaEdge software to the other nodes. NexentaEdge uses the Chef environment for installation. You deploy NexentaEdge using a Chef Solo instance packaged with the NexentaEdge software.

To deploy NexentaEdge to the nodes in the cluster using the NEDEPLOY tool, the deployment workstation must have IPv4 network connectivity to the nodes, either through a management network or an external network.

- Management controller node

A management controller is a node that translates external cluster-wide behavior into internal component-specific configuration, and may provide the connection between the deployment workstation and the data nodes in the Replicast network. At least one of the nodes in the cluster must be a management controller. Management controllers need to have network connectivity to both the deployment workstation and to the other nodes in the cluster.

- External network

External clients store and retrieve data in the NexentaEdge cluster by communicating with gateway nodes on the external network. The external network may use IPv4 or IPv6 and is likely to carry traffic unrelated to NexentaEdge.

- Management network

To aid in deploying and administering the NexentaEdge cluster, you may elect to place the deployment workstation and data nodes in a dedicated IPv4 management network. The NEDEPLOY and NEADM tools, running on the deployment workstation, send configuration information to and receive status information from the data nodes over this network.

- External clients

External clients are end-user machines that access data stored in the NexentaEdge cluster via gateway nodes. External clients access data in the cluster using APIs of the storage services NexentaEdge supports: OpenStack Swift and Amazon S3 via HTTP/REST, or iSCSI block storage.

From a client perspective, the NexentaEdge cluster acts as an OpenStack Swift or Amazon S3 object storage system. To accommodate applications that expect block storage, the NexentaEdge cluster can act as an iSCSI or OpenStack Cinder target.

NexentaEdge Deployment Options

NexentaEdge supports a variety of deployment options, including deployment using Docker containers, installing from an ISO file, and deployment/configuration using Juju charms.

Docker/Baremetal Deployment

NexentaEdge data and gateway nodes can optionally be deployed within Docker containers. Since multiple Docker containers can reside on a single physical server, a given server can have multiple nodes deployed to it, which allows the size of the cluster to scale to the full storage capacity of the server.

For applications running in Docker containers that use NexentaEdge cluster storage, NexentaEdge now features a plugin for Flocker that allows the NexentaEdge data nodes to be moved to a new server when the application's Docker container and associated disks are moved. See [this link](#) for more information.

By default, NexentaEdge is deployed without using Docker containers; that is, a *baremetal* deployment. All of the available storage on the server is allocated by NexentaEdge for use as a single data node.

Deployment Using Packages in an ISO File

During the normal installation and deployment process for NexentaEdge, packages are downloaded from the Internet and installed on the nodes. However, in some installations, downloading packages is not possible, since the nodes may be blocked from the Internet due to security requirements.

To accommodate this kind of installation, Nexenta provides a method to install the required packages from a repository contained in an ISO file instead of downloading them over the Internet. See [Deploying NexentaEdge from an ISO File](#) for installation details.

Juju Charm Deployment

You can deploy NexentaEdge using Canonical's Juju management tool. The functionality of the NexentaEdge deployment tool (NEDEPLOY) and administration tool (NEADM) is contained in a collection of Juju charms. Individual charms allow you to deploy a NexentaEdge cluster with a specified number of nodes, add nodes to the cluster, and configure OpenStack Cinder and Swift storage services. See [this link](#) for more information.

NexentaEdge Installation Procedure

The following table lists the tasks you perform to install the NexentaEdge software and initialize the cluster. For more information about each task, click the link in the right column.

Table 1-1: NexentaEdge Installation Tasks

Task	Instructions
1. Verify that your servers and switches meet the requirements for NexentaEdge.	<ul style="list-style-type: none"> For server hardware requirements and supported operating systems, see Data and Gateway Node Requirements. For switch requirements, see Networking Requirements. See the document “Configuration Guidelines for NexentaEdge” for specific hardware recommendations for various types of NexentaEdge deployments.
2. Determine how many gateway and data nodes your NexentaEdge cluster will have.	Planning Server Node Deployment
3. Configure the switches that support the NexentaEdge servers.	Network Switch Configuration
4. Download and extract the NexentaEdge Cluster Deployment (NEDEPLOY) and Administration (NEADM) tools to the deployment workstation.	Downloading the NexentaEdge Deployment and Administration Tools Extracting the NEDEPLOY and NEADM Archives
5. Run the NEDEPLOY tool to add nodes to the NexentaEdge cluster.	Deploying NexentaEdge
6. Install and activate your product license.	Installing the NexentaEdge License
7. Use the NEADM tool to create one or more logical clusters.	Creating a Logical Cluster
8. Add one or more tenants to the logical clusters.	Creating a Tenant
9. Configure NexentaEdge to work with storage services.	Configuring NexentaEdge Storage Service Groups iSCSI Deployment Example Amazon S3 Deployment Example OpenStack Swift Deployment Example Integrating NexentaEdge with OpenStack Cinder
10. Configure zones (failure domains) for the data nodes in the NexentaEdge cluster.	Configuring Zoning for Data Nodes
11. Set up OpenStack Horizon to display cluster status and/or create containers and objects directly from Horizon dashboard screens (optional).	Integrating NexentaEdge with OpenStack Horizon

Prerequisites

This chapter includes the following topics:

- [Networking Requirements](#)
- [Data and Gateway Node Requirements](#)
- [Deployment Workstation Requirements](#)

Networking Requirements

This section lists the requirements for the Replicast network, which connects the data and gateway nodes in the NexentaEdge cluster.

Switch Hardware Requirements

In general, the network switches that you use for the Replicast network must meet the following requirements:

- 10 Gigabit Ethernet non-blocking enterprise-class switch.
- 9000 MTU jumbo frame support.
- 802.3 Flow Control must be enabled for all Replicast ports (switch and host).
- Support for Multicast Listener Discovery (MLD) snooping for IPv6 multicast traffic (must be enabled on the switch used for the Replicast VLAN).

Replicast Network Requirements

- The Replicast network must be on a separate untagged VLAN, or be a physically separate network
- The Replicast network must be a single IPv6 subnet that has no ingress or egress routes
- The Replicast network must support jumbo frames (9000 MTU)

Data and Gateway Node Requirements

The data and gateway nodes must meet the hardware and software requirements described in this section.

Hardware Requirements

- CPU requirements
 - x86-64bit architecture with a CPU that supports SSE4.2 (full SIMD instruction set extension)
- Network interface requirements

Each node requires multiple network interfaces:

- Management network interface – IPv4, static or DHCP-assigned address
- Replicast network interface – IPv6 unconfigured; jumbo frame support (can be configured by setting up IPv4 with 9000 MTU)
- Client access network interface – Each gateway node must have network connectivity to the clients that will use it for data storage (for example, as an iSCSI target).
- 802.3 Flow Control must be enabled for all Replicast ports (switch and host)

During NexentaEdge deployment, at least one interface on each node must have Internet access, so that required components can be downloaded.

- Memory per node:
 - 512MB RAM per 1TB of hard disk or any rotational device
 - 2GB RAM per 1TB of SSD or any flash device
 - 4GB RAM per 10 Gigabit Ethernet port

For example, if a node has four 4TB hard drives, one 512GB SSD, and one 10 Gigabit network port, it requires a minimum of 14GB RAM. Nexenta recommends at least 16GB RAM per data node, and 48GB per gateway node. If a node will function both as a data node and gateway node, Nexenta recommends at least 64GB RAM.

- To store the NexentaEdge core files, the size of the root partition for each node should be twice the amount of installed RAM, or you can mount `/opt/nedge/var/cores` to a partition or NFS share at least this size.
- All attached SATA/SAS disks or SSDs (other than the Linux operating system disk) must be fully allocated for the CCOW storage system.
- Each gateway node must have a dedicated 10 Gigabit Ethernet port for gateway services, in addition to a dedicated interface port for the CCOW storage system.
- The minimum number of nodes for a NexentaEdge deployment are three data nodes and one gateway node. Each data node must have minimum of four storage devices.

Software Requirements

Before being deployed in a NexentaEdge cluster, the data and gateway nodes must be pre-provisioned with an operating system. The following Linux distributions have been tested to work with NexentaEdge:

- Ubuntu Linux LTS 14.04.3 (3.19 Linux kernel)
- RedHat Enterprise Linux 7.2 or 7.3

- Community Enterprise Linux (CentOS) 7.2 or 7.3

Note that for RedHat/CentOS 7.2 installations, the [ELRepo repository](#) is required. It is not required for RedHat/CentOS 7.3 installations.

For RedHat/CentOS 7.2 installations, Nexenta recommends provisioning servers with the full installation ISO. Avoid using the “minimal” profile.

In addition, all data and gateway nodes must be configured with a management IP address and SSH access. The `/etc/sudoers` on each node must allow all root privileges to the user ID that will be deploying NexentaEdge.

Deployment Workstation Requirements

The deployment workstation is the system from which you deploy the components of NexentaEdge to the data and gateway nodes. It must have one of the following operating systems installed:

- Ubuntu Linux LTS 14.04
- MacOSX 10.x
- CentOS 7.x
- Red Hat Enterprise Linux 7.x

The user ID with which you deploy NexentaEdge must have password-less SSH access to all of the nodes to be added to the cluster, as well as all root privileges set in `/etc/sudoers` on each node.

Requirements for the Chef Environment

The deployment workstation makes use of the Chef environment (www.getchef.com) to install the NexentaEdge software. You deploy NexentaEdge using a Chef Solo instance packaged with the NexentaEdge software. A Chef Solo deployment does not require uploading cookbooks to a Chef server.

Installing NexentaEdge

This chapter includes the following topics:

- [Before You Begin](#)
- [Planning Server Node Deployment](#)
- [Network Switch Configuration](#)
- [Downloading the NexentaEdge Deployment and Administration Tools](#)
- [Extracting the NEDEPLOY and NEADM Archives](#)
- [Preparing Storage Devices for NexentaEdge Deployment](#)
- [Deploying NexentaEdge](#)
- [Installing the NexentaEdge License](#)
- [Creating a Logical Cluster](#)
- [Creating a Tenant](#)
- [Creating a Bucket](#)

Before You Begin

Before you start to deploy NexentaEdge, verify that your environment meets the hardware and software requirements described in the [Prerequisites](#) chapter.

Planning Server Node Deployment

Use the following guidelines when planning deployment of the server nodes in the NexentaEdge cluster:

- Determine the number of gateway and data nodes to be deployed.

Plan to create at least three data nodes and one gateway node. You can increase the number of nodes as needed, up to a maximum of 2,000. In general, each physical server is configured as a single data or gateway node, although a given server can be configured with both data and gateway roles.

It is possible to configure virtual machines as data or gateway nodes, but this is not recommended for a production deployment.
- Determine the roles of the nodes: data node, gateway node, or management controller.

You must have at least one gateway node and one management controller, and at least three server nodes in total. A given server can combine all of these roles.

- Verify that each node in the cluster has access to the Internet.

A server node may require Internet access for cluster provisioning using Chef. You may need to use a proxy if some or all nodes do not have Internet access, or you can grant the nodes access to specific URLs that are required for the deployment.

In the current release, these URLs include <http://www.chef.io> and <https://prodpkg.nexenta.com>.

- For each node, allocate storage devices for the NexentaEdge cluster.

By default, during server node deployment, NexentaEdge allocates all raw (unpartitioned) devices visible on the server. Note that if any data exists on these devices, it will be lost during the deployment process.

Network Switch Configuration

Prior to configuring a NexentaEdge cluster, you need to configure the network switches that support the NexentaEdge servers. A node can be deployed after its switch port has been configured. Nexenta recommends that you add the nodes to the cluster only after all the switch port ranges have been configured.

The Replicast network, which connects the nodes in the cluster, must be a separate VLAN with no ingress or egress routes. It is also possible to set up multiple IPv6 subnets that only route IPv6 traffic between themselves. Consult with Nexenta if your deployment requires the use of IPv6 routers.

Downloading the NexentaEdge Deployment and Administration Tools

To install and configure NexentaEdge, you download two software tools: NEDEPLOY and NEADM. The NexentaEdge Cluster Deployment (NEDEPLOY) tool deploys NexentaEdge to the nodes in the cluster. The NexentaEdge Administration (NEADM) tool allows you to administer the NexentaEdge cluster and obtain information about its performance.

You download and run the NEDEPLOY and NEADM tools on the system you designated as the NexentaEdge deployment workstation.

The user ID with which you deploy NexentaEdge must have password-less SSH access to all of the nodes to be added to the cluster, as well as all root privileges set in `/etc/sudoers` on each node.

Nexenta provides the NEDEPLOY and NEADM tools for the following operating systems (x86 (32 bit) / x64 bit only):

- Ubuntu Linux LTS 14.04.3
- MacOSX 10.x
- Red Hat Enterprise Linux 7.2 or 7.3
- CentOS 7.2 or 7.3

Note: The example commands in this guide use Ubuntu Linux LTS 14.04.3.

Download the NEDEPLOY and NEADM archives appropriate for your deployment workstation:

- nedeploy-linux_1.1_x86.tar.gz and neadm-linux_1.1_x86.tar.gz (Any compatible Linux, 32bit)
- nedeploy-linux_1.1_x64.tar.gz and neadm-linux_1.1_x64.tar.gz (Any compatible Linux, 64bit)
- nedeploy-darwin_1.1_x86.tar.gz and neadm-darwin_1.1_x86.tar.gz (Any compatible MacOSX, 32bit)
- nedeploy-darwin_1.1_x64.tar.gz and neadm-darwin_1.1_x64.tar.gz (Any compatible MacOSX, 64bit)

Extracting the NEDEPLOY and NEADM Archives

After you download the NEDEPLOY and NEADM tools to your deployment workstation, extract the archives to any directory.

❖ *To extract the NEDEPLOY and NEADM archives (using Ubuntu LTS 14.04.3):*

1. Log in to the deployment workstation.
2. Extract the NEDEPLOY and NEADM archive files to the home directory:

Example:

```
$ cd ~
$ tar -xzf <download-directory>/nedeploy-linux_1.1_x86.tar.gz
$ tar -xzf <download-directory>/neadm-linux_1.1_x86.tar.gz
```

3. Set the PATH environment variable to use the directories where you extracted the archives.

Example:

```
$ PATH=$PATH:~/nedeploy
$ PATH=$PATH:~/neadm
```

4. Verify that the NEDEPLOY and NEADM tools are now operational.

Example:

```
$ nedeploy
NexentaEdge Deployment Tool
(additional help text follows)

$ neadm
NexentaEdge Cluster Administration Tool
(additional help text follows)
```

Preparing Storage Devices for NexentaEdge Deployment

During deployment, NexentaEdge allocates all raw (unpartitioned) devices visible on the node to NexentaEdge cluster storage. Note that if any data exists on these devices, it will be lost during the deployment process.

It is possible that the node to which you are deploying NexentaEdge already has disks that are formatted and/or mounted. In order for NexentaEdge to allocate these disks to cluster storage, the partition tables for the disks need to be erased.

❖ *To erase the partition table for devices to be used for NexentaEdge cluster storage:*

1. For each device to be used as NexentaEdge cluster storage, enter the following commands:

```
$ dd if=/dev/zero of=<device> bs=100 count=1
$ hdparm -z <device>
```

Example:

```
$ dd if=/dev/zero of=/dev/sdat bs=100 count=1
$ hdparm -z /dev/sdat
```

On CentOS systems, the `hdparm` utility may not be installed by default. In this case, use the following command to install it:

```
$ yum install hdparm -y
```

Note:

You can optionally exclude specific disks from usage in the NexentaEdge cluster. See the descriptions of the `-x` and `-X` parameters for NEDEPLOY.

Deploying NexentaEdge

The NEDEPLOY tool performs the following configuration tasks:

- Ensuring the nodes meet the system requirements for NexentaEdge
- Adding NexentaEdge server nodes to the NexentaEdge cluster
- Allocating devices on the nodes for use as cluster storage
- Initializing the physical NexentaEdge cluster

You run NEDEPLOY from the NexentaEdge deployment workstation. NEDEPLOY can be run either directly from the command line or using an interactive wizard. Procedures for both types of deployments are included in the following sections.

If errors occur during the deployment process, you can view information about them in the `~/nedeploy/nedeploy_logs/nedge-deploy.log<timestamp>` file.

Before running NEDEPLOY, verify that the network switch topology to be used in the NexentaEdge cluster is set up correctly.

RedHat/CentOS Considerations

For RedHat/CentOS installations, Nexenta recommends the following prior to running NEDEPLOY:

- Pre-configure the networking interfaces used for the Replicast network with IPv4 addresses.
- Disable the Network Manager controller.
- Reboot the node to ensure the interface name is not automatically changed.
- Make sure that the `ifup` and `ifdown` commands work correctly for specified interface names.

- The built-in firewall must be configured to allow IPv6 multicast traffic on the Replicast interface(s). This can be achieved by adding the interface(s) to the list of trusted interfaces for the firewall service, or by stopping (`systemctl stop firewalld` command) and disabling the firewall service (`systemctl disable firewalld` command).

Note that for RedHat/CentOS 7.2 installations, the [ELRepo repository](#) is required. It is not required for RedHat/CentOS 7.3 installations.

For nodes running RedHat/CentOS 7.2, enter the following commands to upgrade to the ELRepo kernel:

```
$ rpm --import https://www.elrepo.org/RPM-GPG-KEY-elrepo.org
$ rpm -Uvh http://www.elrepo.org/elrepo-release-7.0-2.el7.elrepo.noarch.rpm
$ yum --enablerepo=elrepo-kernel install kernel-lt kernel-lt-headers
$ reboot
```

Following the reboot, if the new kernel version is listed first, then use the following command to set `grub` to use 0, which configures the system to boot off the newer kernel. You may want to leave some of the older 3.10.x kernels on the system in case you decide to revert to the stock RedHat/CentOS 7 kernel. You can always modify the default kernel to use later on.

```
$ grub2-set-default 0
```

Using the Command Line

Use the following procedures to run NEDEPLOY from the command line. These procedures check that the nodes meet the system requirements and deploy the NexentaEdge software.

Running the Pre-Check Utility

❖ *To verify that a node meets system requirements for NexentaEdge:*

1. Log in to the NexentaEdge deployment workstation (that is, the workstation where the NEDEPLOY tool is installed).
2. Run the pre-check utility to ensure that the node meets the requirements for being added to the cluster.

```
$ nedeploy precheck <ip-address> <username:password> -i <interface>
[-t <profile>] [-x <disks-to-exclude>] [-X <disks-to-reserve>]
```

Where:

<code><ip-address></code>	Is the management IP address of the node.
<code>username</code>	Is a user account on the node that has administrative privileges; for example, root. If you specify a different account than root, it must have password-less SSH access to the node, as well as all root privileges set in <code>/etc/sudoers</code> on the node.
<code>password</code>	Is the password for this user account.

- `-i <interface>` Is the Ethernet interface to be used for communication between nodes within the NexentaEdge cluster (that is, the Replicast network).
- `-t <profile>` Has to do with how metadata is distributed among the hard disk drives (HDDs) and solid state drives (SSDs) on the node and whether journaling operations are enabled. You can specify one of the following profiles:
- `capacity` NexentaEdge uses all of the available HDDs and SSDs as one large storage pool.
 - `performance` NexentaEdge offloads the majority of metadata to SSD. This is the default profile.
 - `gateway` NexentaEdge configures the node as a gateway with no disks allocated for cluster storage.
- `-x <disks-to-exclude>` Specifies a comma-separated list of one or more devices that NexentaEdge will not use as cluster storage. NexentaEdge adds these disks to a list of devices that it will never allocate to cluster storage. In this list, you can specify disks that you are using for other applications besides NexentaEdge.
- `-X <disks-to-reserve>` Specifies a comma-separated list of one or more devices that NexentaEdge will not allocate to cluster storage for this deployment. The difference between this list and the `<disks-to-exclude>` list is that the disks in the `<disks-to-exclude>` list are permanently excluded from use in the NexentaEdge cluster, while disks in the `<disks-to-reserve>` can be added to the cluster at a later time.

Example:

```
$ nedeploy precheck 10.3.30.34 root:password -i eth1 -t capacity
```

System response:

```
10.3.30.34 - Connecting
10.3.30.34 - Operating System:      "Ubuntu 14.04.3 LTS"
10.3.30.34 - Network Interface Speed: 10000
10.3.30.34 - Total Memory:          8176792kB
10.3.30.34 - No raw un-partitioned disks available
10.3.30.34 - CONFIGURATION CHECK SUCCESS
10.3.30.34 - PROFILE CHECK SUCCESS
```

If the pre-check utility indicates that the node does not meet the requirements for NexentaEdge, correct the issue if possible and re-run the utility.

Deploying NexentaEdge to the Nodes

❖ *To add nodes using NEDEPLOY from the command line:*

1. From the NexentaEdge deployment workstation, use the following command to deploy the NexentaEdge software to the nodes:

```
$ nedeploy deploy solo <ip-address> <nodename> <username:password>
-i <interface> [-t <profile>] [-x <disks-to-exclude>]
[-X <disks-to-reserve>] [-z <zone>] [-F <filesystem-type>] [-m] [--docker]
[--upgrade]
```

Where:

<code><ip-address></code>	Is the management IP address of the node.
<code><nodename></code>	Is the name of the node to add to the cluster. This will be the name recorded in the Chef database for the node. You can assign any name to the node, although Nexenta recommends that you use the host name of the node.
<code>username</code>	Is a user account on the node that has administrative privileges; for example, root. If you specify a different account than root, it must have password-less SSH access to the node, as well as all root privileges set in <code>/etc/sudoers</code> on the node.
<code>password</code>	Is the password for this user account.
<code>-i <interface></code>	Is the Ethernet interface to be used for communication between nodes within the NexentaEdge cluster (that is, the Replicast network).
<code>-t <profile></code>	Has to do with how metadata is distributed among the hard disk drives (HDDs) and solid state drives (SSDs) on the node and whether journaling operations are enabled. You can specify one of the following profiles: <code>capacity</code> NexentaEdge uses all of the available HDDs and SSDs as one large storage pool. <code>performance</code> NexentaEdge offloads the majority of metadata to SSD. This is the default profile. <code>gateway</code> NexentaEdge configures the node as a gateway with no disks allocated for cluster storage.
<code>-x <disks-to-exclude></code>	Specifies a comma-separated list of one or more devices that NexentaEdge will not use as cluster storage. NexentaEdge adds these disks to a list of devices that it will never allocate to cluster storage. In this list, you can specify disks that you are using for other applications besides NexentaEdge.
<code>-X <disks-to-reserve></code>	Specifies a comma-separated list of one or more devices that NexentaEdge will not allocate to cluster storage for this deployment. The difference between this list and the <code><disks-to-exclude></code> list is that the disks in the <code><disks-to-exclude></code> list are permanently excluded from use in the NexentaEdge cluster, while disks in the <code><disks-to-reserve></code> can be added to the cluster at a later time.

<code>-z <zone></code>	Specifies the zone to which the node belongs. All of the nodes in a given zone are considered to be part of the same failure domain; for example, a group of nodes that receive power from the same source can be placed in the same zone. NexentaEdge ensures that data in the cluster is replicated across multiple zones, so that failure of the nodes in a zone does not result in lost access to the data. See Configuring Zoning for Data Nodes for more information.
<code>-m</code>	Is an option to configure this node as a management controller.
<code>--docker</code>	Deploys NexentaEdge to the node using Docker containers. If the node has 80 TB or more of disk space, you must specify this option.
<code>-F <filesystem-type></code>	Specifies the filesystem type to use for LFS drivers. You can specify <code>ext4</code> , <code>zfs</code> , or <code>none</code> . The default of <code>none</code> causes the RD (raw disk) driver to be automatically enabled.
<code>--upgrade</code>	Upgrades the NexentaEdge core files on the node.

Example:

```
$ nedeploy deploy solo 10.3.30.32 root:password -i eth1 -t capacity
-x /dev/sdau,/dev/sdaw
```

2. Repeat the previous step for each node you want to add to the cluster.
3. After deploying NexentaEdge to all of the nodes, continue with [Initializing the NexentaEdge Cluster](#).

Using the NEDEPLOY Wizard

The NEDEPLOY wizard provides an interactive, input-driven method for deploying NexentaEdge. You can use the NEDEPLOY wizard as an alternative to deploying NEDEPLOY from the command line.

❖ *To run the NEDEPLOY wizard:*

1. From the NexentaEdge deployment workstation, use the following command to start the NEDEPLOY wizard:

```
$ nedeploy wizard [--skip-precheck] [--docker] [-t <profile>]
```

Where:

<code>--skip-precheck</code>	Disables the pre-check procedure, which is run as part of the deployment wizard. Note that running the wizard without the pre-check procedure may result in problems with the deployment.
<code>--docker</code>	Deploys NexentaEdge to the node using Docker containers. If the node has 80 TB or more of disk space, you must specify this option.

`-t <profile>` Has to do with how metadata is distributed among the hard disk drives (HDDs) and solid state drives (SSDs) on the node and whether journaling operations are enabled. You can specify one of the following profiles:

`capacity` NexentaEdge uses all of the available HDDs and SSDs as one large storage pool.

`performance` NexentaEdge offloads the majority of metadata to SSD. This is the default profile.

`gateway` NexentaEdge configures the node as a gateway with no disks allocated for cluster storage.

2. Read and accept the terms of License Agreement.
3. On the Add nodes to the cluster screen, highlight Add Node and press Enter.
4. Enter details for the new node; for example:

```

----- Add nodes to the cluster -----
Management Network IP      10.3.30.32
Address of node visible to the workstation's network

Chef Name                   node32
Name of node to register in Chef's database

Host Username               root
Username used to access the node via SSH

User Password               *****
Above user's password

Replicast Interface        eth1
Node's interface name used for internal replicast network

Zone                        1
Optional node zone, defaults to 0

Management Node            [x]
Node to manage and monitor cluster          + Done      - Cancel
Press Space to check/uncheck

```

Where:

- Management Network IP** Is the IP address of the server node that you want to add to the NexentaEdge cluster. The deployment workstation must be able to connect to the node using this address.
- Chef Name** Is the name of the server node to be added to the Chef database. You can assign any name to the server node as its Chef name. However, Nexenta recommends that you use the host name of the node.

- Host Username** Is a server node user account with administrative privileges; for example, root.
If you specify a different account than root, it must have password-less SSH access to the node, as well as all root privileges set in `/etc/sudoers` on the node.
- User Password** Is the password for the account specified as the Host Username.
- Replicast Interface** Is the Ethernet interface that is used for communication between nodes within the NexentaEdge cluster (that is, the IPv6 Replicast network). This would normally be different from the network used for managing the cluster. See [Figure 1-1](#).
- Zone** Specifies the zone to which the node belongs. All of the nodes in a given zone are considered to be part of the same failure domain; for example, a group of nodes that receive power from the same source can be placed in the same zone. NexentaEdge ensures that data in the cluster is replicated across multiple zones, so that failure of the nodes in a zone does not result in lost access to the data.

Note that if you are configuring multiple zones, they must be set up before the system is initialized and the license is installed.

See [Configuring Zoning for Data Nodes](#) for more information.
- Management Node** Specifies whether this node is a management controller node or not. You must assign one management controller per cluster.

5. After entering information for the new node, highlight `Done` and press Enter.
6. Repeat steps 3 - 5 for each node to be added to the cluster. The minimum number of nodes per cluster is three.
7. When you have finished adding node information, highlight `Deploy` and press Enter

```

----- Add nodes to the cluster -----
Chef Name      Management IP      Interface  Zone  Mgmt
node32         10.3.30.32        eth1       1     *
node33         10.3.30.33        eth1       1
node34         10.3.30.34        eth1       1

                                - Del Node  + Add Node  Deploy >>
    
```

8. At the `Are you ready to deploy?` prompt, press `y` to start the software deployment on the nodes.

The pre-check procedure determines whether the nodes meet the system requirements for NexentaEdge. If they do, the software is installed on the nodes; if they don't, the deployment wizard halts and displays the errors it encountered. You can view these errors in the `~/nedeploy/nedeploy_logs/nedge-deploy.log<timestamp>` file.

- When the deployment finishes, press Enter to exit the deployment wizard.

Initializing the NexentaEdge Cluster

❖ *To initialize the NexentaEdge cluster:*

- On the NexentaEdge deployment workstation, enter the `neadm system status` command. The first time you enter this command after deployment, NexentaEdge prompts you for the URL for the NexentaEdge management controller node.

Example:

```
$ neadm system status
Error: connect ECONNREFUSED
Unable to reach management node: http://0.0.0.0:8080
Please enter management node IP with NEDGE port (default 8080)
Example: http://1.2.3.4:8080
? Remote API URL: http://10.3.30.32:8080

Remote API URL successfully updated. Re-run system status again ...
```

- Verify the system status:

```
$ neadm system status
```

System response:

ZONE:HOST	SID	UTIL	CAP	CPU	MEM	DEVs	STATE
0:node32 [MGMT]	8C2C6400B2C5D2...	0%	64G	4/0.58@2.34Ghz	3.17G/7.8G	4/4	ONLINE
0:node33	AC4F3E7923F260...	0%	64G	4/0.42@2.34Ghz	3.18G/7.8G	4/4	ONLINE
0:node34	B2665616B2CFED...	0%	64G	4/1.06@2.34Ghz	2.99G/7.8G	4/4	ONLINE

Ensure that the nodes you added to the cluster have a status of `ONLINE`. Upon initial deployment, it may take up to two minutes for NexentaEdge to get the status for the nodes. If you see an error message when you enter the `neadm system status` command, wait two minutes and enter the `neadm system status` command again.

- If the servers have a status of `ONLINE`, initialize the cluster:

```
$ neadm system init
```

System response:

```
NexentaEdge cluster initialized successfully
System GUID: DD3EE6D7-1234-5678-9012-C1B41C7EABC8
```

Note the System GUID; you may need to use it when you configure your NexentaEdge license key.

Installing the NexentaEdge License

To activate NexentaEdge, you must install a license for the product. NexentaEdge supports either online or offline license installation. Both of these methods use the activation key you received when you downloaded the software from Nexenta Systems.

NexentaEdge has the following types of licenses:

- Enterprise license – Enables all NexentaEdge features. Depending on the type of Enterprise license you purchase from Nexenta Systems, there may be limitations on the amount of time NexentaEdge can be used or the amount of storage capacity that can exist in the cluster.

The following options are available for Enterprise licenses:

- Trial – Allows full functionality for 45 days after activation. After the trial period expires, you must install a perpetual license in order to keep using the product. A trial license does not have any limitation on the amount of storage in the cluster.
- Perpetual – Allows full functionality and does not expire. For a perpetual license, you can specify the amount of storage capacity that can be added to the cluster.

If you move from a trial license to a perpetual license, make sure the perpetual license allows at least as much storage capacity in the cluster as you have already deployed with your trial license.

Online License Installation

Use online license installation if the deployment workstation has access to the Internet. You enter the activation key with an NEADM command. NEADM contacts the Nexenta license server, which uses the activation key to generate your license. NEADM downloads your license and installs it in the system.

❖ *To install a NexentaEdge license using online license installation:*

1. Log in to the NexentaEdge deployment workstation.
2. Use the following command to generate and install the NexentaEdge license:

```
$ neadm system license set online <activation_key>
```

System response:

```
Online Activation Completed.
```

Offline License Installation

Use offline license installation if the deployment workstation does not have access to the Internet. You submit the activation key and System GUID to Nexenta, either by using the Support Portal, or contacting your Nexenta representative. Nexenta sends you a license, which you save as a text file, load onto the deployment workstation, and install in the system using the NEADM tool.

❖ *To install a NexentaEdge license using offline license installation:*

1. Log in to the NexentaEdge deployment workstation.
2. Display the System GUID.

```
$ neadm system license show
```

System response:

```
Installation GUID : DD3EE6D7-1234-5678-9012-C1B41C7EABC8
License           : not activated
```

3. Copy the value displayed for the `Installation GUID`. The GUID is also displayed when you initialize the cluster with the `neadm system init` command.
4. Submit the GUID, along with the activation key, to Nexenta. Use the Support Portal or contact your Nexenta representative. Nexenta will use the GUID and activation key to generate your license and send it to you.
5. Save the license as a text file in a location where it is accessible from the deployment workstation.
6. Use the following command to install the NexentaEdge license:

```
$ neadm system license set file <license_file>
```

System response:

```
License Set Successfully
```

Creating a Logical Cluster

A NexentaEdge deployment consists of a single physical cluster and one or more logical clusters. The physical cluster is simply the server devices and network switches used by NexentaEdge. A logical cluster is the collection of data and gateway nodes that make up the storage system. A logical cluster has one or more tenants. NexentaEdge provides services such as iSCSI block storage, OpenStack Swift, and Amazon S3 to specific tenants within a logical cluster.

While NexentaEdge supports an unlimited number of logical clusters, a typical deployment has one logical cluster.

❖ *To create a NexentaEdge cluster:*

1. Log in to the deployment workstation.
2. Create a cluster:

```
$ neadm cluster create <cluster_name>
```

Example:

```
$ neadm cluster create clu1
```

System response:

```
Cluster clu1 created successfully.
```

3. Proceed to [Creating a Tenant](#)

Creating a Tenant

A tenant is a group of users, or an account, that shares resources such as containers or virtual machines. Typically, a tenant is also the entity that is billed for storage services.

You must have at least one cluster created before you can create a tenant. After creating the tenant, you can create buckets and specify the storage services that NexentaEdge provides for this tenant.

❖ *To create a tenant:*

1. Log in to the deployment workstation.
2. Create a tenant:

```
$ neadm tenant create <cluster-name>/<tenant-name>
```

Example:

```
$ neadm tenant create clu1/ten1
```

System response:

```
Tenant ten1 created successfully.
```

3. Proceed to [Creating a Bucket](#)

Creating a Bucket

A bucket is a container for objects. For example, if you deploy an iSCSI storage service group, you can create a bucket and add LUNs to it.

Buckets are created on a per-tenant basis; to create a bucket, you must have first created a cluster and the tenant to which you want to assign the bucket.

❖ *To create a bucket:*

1. Log in to the deployment workstation.
2. Create a bucket:

```
$ neadm bucket create <cluster-name>/<tenant-name>/<bucket-name>
```

Example:

```
$ neadm bucket create clu1/ten1/buk1
```

System response:

```
Bucket buk1 created successfully.
```

3. Proceed to [Configuring NexentaEdge Storage Service Groups](#)

Configuring NexentaEdge Storage Service Groups

This chapter includes the following topics:

- [About NexentaEdge Storage Service Groups](#)
- [Before You Begin](#)
- [iSCSI Deployment Example](#)
- [Amazon S3 Deployment Example](#)
- [OpenStack Swift Deployment Example](#)
- [Configuring Zoning for Data Nodes](#)

About NexentaEdge Storage Service Groups

NexentaEdge enables you to easily provision one or multiple enterprise-class object and block storage services. External clients access data stored in the NexentaEdge cluster via gateway nodes using protocols of the storage services NexentaEdge supports.

NexentaEdge includes support for the following:

- OpenStack Swift object storage
- OpenStack Cinder block storage
- Amazon S3 object storage
- iSCSI block storage

To configure a NexentaEdge cluster, you indicate which storage service(s) you want to provide for a given tenant, then specify the servers that should serve as the gateway nodes for that storage service. The data and gateway nodes that provide the storage service are known as a *storage service group*.

To complete the installation, you may need to perform some additional configuration tasks that are specific to the storage services to which you are connecting the NexentaEdge cluster. For example, if you are configuring an iSCSI storage service group, you can create LUNs that are made available to iSCSI initiators via the iSCSI target enabled by the iSCSI storage service group. These tasks are covered in the *NexentaEdge User Guide*.

Before You Begin

Before attempting the procedures in this chapter, make sure you have already deployed a NexentaEdge cluster with at least one tenant.

iSCSI Deployment Example

The following example describes how to configure an iSCSI storage service group with one iSCSI LUN.

For the example, the following naming conventions are used:

Cluster Name	clu1
Tenant	ten1
Bucket	buk1
iSCSI storage service group name	isc01
LUN	LUN01
Gateway node	D586BF84009C4230BFEE31CAC961197D

❖ *To configure an iSCSI storage service group:*

1. Add an iSCSI storage service group to the cluster:

```
$ neadm service create iscsi isc01
```

System response:

```
Service isc01 created
```

2. Verify the server ID that you want to add to the cluster:

```
$ neadm system status
```

System response:

ZONE:HOST	SID	UTIL	CAP	CPU	MEM	DEVs	STATE
0:node32 [MGMT]	8C2C6400B2C5D2...	0%	64G	4/0.58@2.34Ghz	3.17G/7.8G	4/4	ONLINE
0:node33	AC4F3E7923F260...	0%	64G	4/0.42@2.34Ghz	3.18G/7.8G	4/4	ONLINE
0:node34	B2665616B2CFED...	0%	64G	4/1.06@2.34Ghz	2.99G/7.8G	4/4	ONLINE

3. Copy the server ID.
4. Associate the server ID with the iSCSI service group:

```
$ neadm service add isc01 newnode101
```

System response:

```
Service isc01 added to 8C2C6400B2C5D2C69CE0A
```

If you are deploying the iSCSI storage service group in a high-availability (HA) configuration, you can specify two server nodes to be gateway nodes by configuring a virtual IP address (VIP) for the storage service group; otherwise only one node can be a gateway node. See the *NexentaEdge User's Guide* for more information.

5. Apply the iSCSI storage service group to the cluster/tenant:

```
$ neadm service serve isc01 clu1/ten1
```

System response:

```
Service isc01 now serving path clu1/ten1
```

6. Enable the iSCSI storage service group:

```
$ neadm service enable isc01
```

System response:

```
Service isc01 enabled
```

7. Verify that the iSCSI storage service group is enabled.

```
$ neadm service list
```

System response:

```
TYPE  NAME  SERVERID  STATUS
iscsi isc01  D586BF84009C4230BFEE31CAC961197D  enabled
```

8. Add an iSCSI LUN:

```
$ neadm iscsi create <service-group> <lun-path> <size> [-b <block-size>]
[-s <chunk-size>] [-n <lun-number>] [-v <vip-address>/<netmask>]
[-r <replication-count>] [-t <IOPS-limit>]
```

Where:

<service-group>	Is the iSCSI storage service group for which the LUN is being created.
<lun-path>	Associates the LUN with a specific cluster/tenant/bucket combination for this iSCSI storage service group.
<size>	Is the size of the new LUN. To specify a size in gigabytes, use the GB suffix (for example, 2GB to specify 2 gigabytes). If you do not specify a suffix, a size in bytes is assumed.
-b <block-size>	Is the block size for the new LUN. The default is 4096 bytes. You can change this in increments of 512 bytes, with a minimum block size of 512 bytes. For ESX, this should be set to 512 bytes.
-s <chunk-size>	Sets the size of the chunks in the object that backs the LUN; NexentaEdge breaks objects stored for this LUN into chunks of this size. The default is 131,072 bytes. You can specify increments of 512 bytes, with a minimum chunk size of 512 bytes and a maximum of 131,072 bytes.
-n <lun-number>	Is the LUN number for the new LUN. You can specify a number from 1–255.
-v <vip-address>/<netmask>	If you are configuring multiple virtual IP addresses (VIPs) for the iSCSI storage service group, when you create new LUNs, you must specify which VIP the new LUN will be associated with. See the <i>NexentaEdge User Guide</i> .
-r <replication-count>	Sets the number of times the LUN object is replicated across the cluster. This can be set to 2, 3, or 4. The default replication count for a LUN object is 3.

`-t <IOPS-limit>` Sets a limit in 4K normalized IOPS for the LUN object.

For the `<size>`, `<block-size>`, and `<chunk-size>` parameters, you can optionally specify a suffix to indicate the units (B, KB, MB, GB, or TB). For example, to specify a size in gigabytes, use the GB suffix. If you do not specify a suffix, a size in bytes is assumed.

Example:

```
$ neadm iscsi create isc01 clu1/ten1/buk1/LUN01 1G
```

System response:

```
iscsi LUN clu1/ten1/buk1/LUN01 created.
```

9. Log in to your iSCSI initiator and discover the iSCSI target that you added to the service group.

Additional iSCSI Management Commands

- ❖ *To view the list of iSCSI LUNs:*

```
$ neadm iscsi list isc01
```

System response:

LUN	HOST	SIZE	BLOCK	CHUNK	REPCOUNT	PATH
1	newnode101	1G	4K	32K	3	clu1/ten1/buk1/LUN01

- ❖ *To create a snapshot of a LUN:*

```
$ neadm iscsi snapshot create isc01 clu1/ten1/buk1/LUN01@snap01
```

System response:

```
Snapshot snap01 created.
```

- ❖ *To view the list of LUN snapshots:*

```
$ neadm iscsi snapshot list isc01 clu1/ten1/buk1/LUN01
```

System response:

```
SNAPSHOTS:
snap01
```

- ❖ *To create a clone of the snapshot:*

```
$ neadm iscsi snapshot clone isc01 clu1/ten1/buk1/LUN01@snap01 clu1/ten1/buk1/LUN02
```

System response:

```
Snapshot snap01 cloned into LU clu1/ten1/buk1/LUN02.
```

Note: Specify a new LUN to clone the snapshot. If you specify an existing LUN, the procedure fails.

- ❖ *To roll back a LUN to a snapshot version:*

```
$ neadm iscsi snapshot rollback isc01 clu1/ten1/buk1/LUN01@snap01
```

System response:

```
iSCSI LU clu1/ten1/buk1/LUN01 rolled back to Snapshot: snap01
```

❖ *To delete a snapshot:*

```
$ neadm iscsi snapshot delete isc01 clu1/ten1/buk1/LUN01@snap01
```

System response:

```
Snapshot snap01 deleted.
```

See “Configuring an iSCSI Storage Service” in the *NexentaEdge User Guide* for more information.

Integration with OpenStack Cinder

A NexentaEdge cluster can provide block storage functions to OpenStack Cinder. OpenStack Cinder includes an API that can manage a block storage backend. You can configure this storage backend to be a NexentaEdge iSCSI storage service group.

To do this, you set up an iSCSI storage service group, then install and enable the NexentaEdge Cinder plugin, which configures OpenStack Cinder to use the iSCSI service group for storage. See [Integrating NexentaEdge with OpenStack Cinder](#) for configuration details.

Amazon S3 Deployment Example

The following example describes how to configure an Amazon S3 storage service group.

For the example, the following naming conventions are used:

Cluster Name	clu1
Tenant	ten1
Amazon S3 storage service group name	s301
Server ID	75D4F08B788914DD3A6AB91CAEB8958F

❖ *To configure an Amazon S3 storage service group:*

1. Add an Amazon S3 storage service group to the cluster:

```
$ neadm service create s3 s301
```

System response:

```
Service s301 created
```

2. Verify the server ID that you want to add to the cluster:

```
$ neadm system status
```

System response:

```
ZONE:HOST      SID              UTIL CAP CPU      MEM      DEVs STATE
```

```
0:node32 [MGMT] 8C2C6400B2C5D2... 0% 64G 4/0.58@2.34Ghz 3.17G/7.8G 4/4 ONLINE
0:node33      AC4F3E7923F260... 0% 64G 4/0.42@2.34Ghz 3.18G/7.8G 4/4 ONLINE
0:node34      B2665616B2CFED... 0% 64G 4/1.06@2.34Ghz 2.99G/7.8G 4/4 ONLINE
```

3. Copy the server ID.
4. Associate the server ID with the Amazon S3 storage service group:

Example:

```
$ neadm service add s301 node202
```

System response:

```
Service s301 added to 75D4F08B788914DD3A6AB91CAEB8958F
```

5. Apply the Amazon S3 storage service group to the cluster/tenant:

```
$ neadm service serve s301 clu1/ten1
```

System response:

```
Service s301 now serving path clu1/ten1
```

The combination of cluster and tenant is known as a logical path.

6. Enable the Amazon S3 storage service group:

```
$ neadm service enable s301
```

System response:

```
Service s301 enabled
```

After the Amazon S3 storage service group is enabled, you can use the IP address of the gateway node (server ID 8C2C6400B2C5D2C69CE0A) for Amazon S3 API communication.

Additional Configuration

After configuring an Amazon S3 storage service group, you can perform the following optional configuration tasks:

- Configuring authentication for the Amazon S3 storage service group
- Changing the ports for Amazon S3 API calls

These tasks are covered in the “Configuring an Amazon S3 Storage Service Group” chapter of the *NexentaEdge User Guide*.

OpenStack Swift Deployment Example

The following example describes how to configure an OpenStack Swift storage service group.

For the example, the following naming conventions are used:

```
Cluster Name          clu1
```

```
Tenant                AUTH_1234
OpenStack Swift storage service group name  sw01
Server ID              D586BF84009C4230BFEE31CAC961197D
```

❖ *To configure an OpenStack Swift storage service group:*

1. Add an OpenStack Swift storage service group to the cluster:

```
$ neadm service create swift sw01
```

System response:

```
Service sw01 created
```

2. Verify the server ID that you want to add to the cluster:

```
$ neadm system status
```

System response:

ZONE:HOST	SID	UTIL	CAP	CPU	MEM	DEVs	STATE
0:node32 [MGMT]	8C2C6400B2C5D2...	0%	64G	4/0.58@2.34Ghz	3.17G/7.8G	4/4	ONLINE
0:node33	AC4F3E7923F260...	0%	64G	4/0.42@2.34Ghz	3.18G/7.8G	4/4	ONLINE
0:node34	B2665616B2CFED...	0%	64G	4/1.06@2.34Ghz	2.99G/7.8G	4/4	ONLINE

3. Copy the server ID.

4. Associate the server ID with the OpenStack Swift storage service group:

```
$ neadm service add sw01 newnode101
```

System response:

```
Service sw01 added to 8C2C6400B2C5D2C69CE0A
```

5. Apply the OpenStack Swift storage service group to the cluster:

```
$ neadm service serve sw01 clu1
```

System response:

```
Service sw01 now serving path clu1
```

6. Enable the OpenStack Swift storage service group:

```
$ neadm service enable sw01
```

System response:

```
Service sw01 enabled
```

7. Verify that the OpenStack Swift storage service group is enabled.

```
$ neadm service list
```

System response:

TYPE	NAME	SERVERID	STATUS
swift	sw01	8C2C6400B2C5D2C69CE0A	enabled

After creating an OpenStack Swift storage service group, use the gateway IP with the port number for OpenStack Swift API.

After creating an OpenStack Swift storage service group, you can use the IP address of the gateway node (server ID `8C2C6400B2C5D2C69CE0A`) for OpenStack Swift API communication.

Additional Configuration

After configuring an OpenStack Swift storage service group, you can perform the following optional configuration tasks:

- Configuring authentication for the OpenStack Swift storage service group
- Changing the ports for OpenStack Swift API calls

These tasks are covered in the “Configuring an OpenStack Swift Storage Service Group” chapter of the *NexentaEdge User Guide*.

In addition, you can integrate the NexentaEdge cluster with OpenStack Horizon, the OpenStack browser-based dashboard, so that you can display cluster status and/or create containers and objects using the Horizon dashboard interface. See [Integrating NexentaEdge with OpenStack Horizon](#).

Configuring Zoning for Data Nodes

In the context of NexentaEdge, a *zone* refers to a group of servers that for failure considerations can be treated as a single domain; that is, as a failure domain. For example, a zone can be made up of a group of servers in a rack that receive power from a single source. If that power source should fail, all of the servers in the zone will lose power and fail. In this example, the rack is the failure domain. In NexentaEdge, a failure domain is represented as a zone.

When NexentaEdge replicates data across the nodes in the cluster, it makes three copies of each chunk and stores each copy in a different location. To ensure constant availability of the data, each replicated chunk should be stored on a node in a different zone. This ensures that if all the nodes in one zone fail, the data is still accessible from a node in one of the remaining zones.

By default, if you do not assign the data nodes to zones, all of the nodes are considered to be part of a single zone, *zone 0*. NexentaEdge distributes the chunks across the nodes in the cluster without regard to zone assignment.

To configure zoning, you assign the data nodes to zones other than zone 0. When you do this, NexentaEdge takes a node’s zone assignment into account when selecting where it stores each replicated chunk. NexentaEdge distributes copies of a chunk across the zones, so that if the chunk is not available from a given zone (for example, if all of the nodes in the zone fail), it may be available from one of the other zones.

A zoning configuration requires a minimum of three zones, each consisting of one or more data nodes. You can assign nodes to zones either during the NexentaEdge deployment process or using NEADM. The procedure below shows how to configure zoning using NEADM. For information about how to configure zoning during NexentaEdge deployment, see [Deploying NexentaEdge](#).

If you configure new zones with additional nodes after the initial NexentaEdge deployment, the number of nodes in the new zone must be at least 75 percent of the number of nodes in the other zones. For example, if your existing NexentaEdge configuration has 3 zones of 7 nodes each, and you want to add a new 4th zone, the new zone must have at least 5 (75 percent of 7) nodes. This is required to ensure that the new zone has sufficient resources (storage, memory, CPU capacity) to cover for a failed zone in case of a zone failure.

When data nodes are deployed in multiple Docker containers on a single physical server, all of the data nodes on that server are part of the same zone to which the server is assigned.

❖ *To configure zoning:*

1. From the NexentaEdge deployment workstation, use the following command to assign a node to a zone.

```
$ neadm system zone <nodename> <zone>
```

Example:

```
$ neadm system zone newnode101 1
```

2. Repeat the previous step for all of the nodes to be included in the zoning configuration.
3. On each node that you have assigned to a zone, use the following command to restart NexentaEdge services.

```
$ service nedge restart
```

4. On the NexentaEdge deployment workstation, use the following command to verify the zone assignments for the nodes.

```
$ neadm system status
```

System response:

ZONE:HOST	SID	UTIL	CAP	CPU	MEM	DEVs	STATE
1:node32 [MGMT]	8C2C6400B2C5D2...	0%	64G	4/0.58@2.34Ghz	3.17G/7.8G	4/4	ONLINE
1:node33	AC4F3E7923F260...	0%	64G	4/0.42@2.34Ghz	3.18G/7.8G	4/4	ONLINE
1:node34	B2665616B2CFED...	0%	64G	4/1.06@2.34Ghz	2.99G/7.8G	4/4	ONLINE

Integrating NexentaEdge with OpenStack Cinder

This chapter includes the following topics:

- [Overview](#)
- [Setting Up an iSCSI Storage Service for Cinder](#)
- [Enabling the NexentaEdge Cinder Plugin](#)

Overview

This chapter describes how to set up a NexentaEdge cluster to provide block storage functions to OpenStack Cinder. OpenStack Cinder includes an API that can manage a block storage backend. You can configure this storage backend to be a NexentaEdge iSCSI storage service.

To do this, you set up an iSCSI storage service, then install and enable the NexentaEdge Cinder plugin, which configures OpenStack Cinder to use the iSCSI service for storage.

If you deploy the iSCSI storage service in a high-availability (HA) configuration, where two gateway nodes provide a single target for initiator requests using a virtual IP (VIP) address, the NexentaEdge Cinder plugin automatically determines the VIP configured for the service.

For more information on the NexentaEdge Cinder plugin, see [this link](#).

Setting Up an iSCSI Storage Service for Cinder

To configure an iSCSI storage service to be used as backend storage for Cinder, follow the procedure for setting up an iSCSI storage service, as demonstrated in [iSCSI Deployment Example](#). Create an iSCSI storage service, but do not create any LUNs.

Enabling the NexentaEdge Cinder Plugin

The NexentaEdge Cinder plugin allows the iSCSI storage service to operate as a block storage backend for OpenStack. It manages the interaction between the Cinder API and the iSCSI storage service.

To obtain the NexentaEdge Cinder plugin, see [this link](#).

❖ *To enable the NexentaEdge Cinder plugin:*

1. Edit the file `/etc/cinder/cinder.conf`.
2. In the `cinder.conf` file, enable the backend and set it to default by configuring the following parameters under the `[DEFAULT]` section:

```
default_volume_type=nedge
enabled_backends = nedge
```

3. Add the following to the end of the `cinder.conf` file:

```
[nedge]
iscsi_helper = tgtadm
volume_group = nedge-volumes
volume_backend_name = nedge
volume_driver = cinder.volume.drivers.nexenta.nexentaedge.iscsi.NexentaEdgeISCSIDriver
nexenta_rest_address = <management-controller-address>
nexenta_rest_port = <management-REST-port>
nexenta_rest_protocol = <protocol>
nexenta_iscsi_target_portal_port = <iSCSI-target-port>
nexenta_rest_user = <management-username>
nexenta_rest_password = <management-password>
nexenta_lun_container = <lun-container-path>
nexenta_iscsi_service = <service-group>
nexenta_client_address = <target-gateway-address>
```

Where:

<management-controller-address>	Is the IP address of the Management Controller node for the NexentaEdge cluster.
<management-REST-port>	Is the port used for REST communication with the Management Controller node; default is 8080.
<protocol>	Is the protocol used for REST communication with the Management Controller node. This can be set to <code>http</code> , <code>https</code> , or <code>auto</code> ; default is <code>auto</code> .
<iSCSI-target-port>	Is the port for the iSCSI target; default is 3260.
<management-username>	Is the username of the NexentaEdge REST user; default is <code>admin</code> .
<management-password>	Is the password for the NexentaEdge REST user; default is <code>nexenta</code> .
<lun-container-path>	Is the cluster/tenant/bucket combination that contains the LUNs (this can be any existing logical bucket); for example, <code>clu1/ten1/buk1</code> .
<service-group>	Is the name of the NexentaEdge iSCSI storage service group used by Cinder; for example, <code>isc01</code> .
<target-gateway-address>	Is the IP address of the iSCSI target gateway. If a VIP is configured for the iSCSI storage service, this setting can be omitted.

4. Save and close the `cinder.conf` file.

5. Extract the NexentaEdge Cinder plugin to the `stack/cinder/cinder/volume/drivers/nexenta` directory.

6. Restart the OpenStack Cinder processes:

```
$ service cinder-volume restart
```

7. To display the `nedge` volume type in OpenStack Horizon, enter the following Cinder console commands on the OpenStack Cinder host:

```
$ cinder --os-username admin --os-tenant-name admin type-create nedge
$ cinder --os-username admin --os-tenant-name admin type-key nedge set
volume_backend_name=nedge
```

Integrating NexentaEdge with OpenStack Horizon

This chapter includes the following topics:

- [Overview](#)
- [Prerequisites](#)
- [Installing the Horizon Dashboard Plugin](#)
- [Registering the NexentaEdge Cluster as a Swift Storage System](#)

Overview

OpenStack Horizon is a browser-based dashboard interface for managing OpenStack services. If you configure an OpenStack Swift storage service for your NexentaEdge cluster (see [OpenStack Swift Deployment Example](#)), it can be integrated with OpenStack Horizon so that the cluster appears in Horizon dashboard displays.

Depending on how you want to use OpenStack Horizon with your NexentaEdge cluster, you can do one of the following:

- Install the Horizon dashboard plugin. If you only want to display information about the NexentaEdge cluster from OpenStack Horizon, install the Horizon dashboard plugin, which lets you display a screen with statistics and usage information about the cluster, but you cannot create containers or add objects from this screen.
- Register the cluster as a Swift storage system. To create containers and manage objects in the NexentaEdge cluster from OpenStack Horizon, register the NexentaEdge cluster as an OpenStack Swift storage system. When you do this, the NexentaEdge cluster is shown as an Object Store in Horizon.

Prerequisites

The procedures in this chapter assume you have already installed OpenStack Horizon and OpenStack Keystone. If not, go to docs.openstack.org for installation procedures or see [this link](#). The commands shown in the following examples are for Ubuntu Linux.

Installing the Horizon Dashboard Plugin

Use the following procedure to install a dashboard in OpenStack Horizon that displays statistics and status for your NexentaEdge cluster, including usage and health information for each node.

❖ *To install the Horizon dashboard plugin for NexentaEdge:*

1. Download the NexentaEdge dashboard package and extract it.

```
$ sudo sh -c "apt-get install tar wget -y && \
cd /usr/share/openstack-dashboard/openstack_dashboard/dashboards && \
wget https://prodpkg.nexenta.com/nedge/plugins/horizon/horizon-plugin-nexenta-edge_1.1.0-1.0.0_noarch.tar.gz && \
tar -xvf master.tar.gz"
```

2. Open the `/usr/share/openstack-dashboard/openstack_dashboard/settings.py` file for editing.

3. Register the new plugin as an installed app. Modify the `INSTALLED_APPS` list as follows:

```
INSTALLED_APPS = [ ... ,
'openstack_dashboard.dashboards.horizon_nedge_dashboard', ]
```

4. Specify the NexentaEdge statistics endpoint by adding the `NEDGE_URL` configuration parameter; for example:

```
NEDGE_URL="http://<management-node-ip>:8080/"
```

where `<management-node-ip>` is the IP address of the management controller node.

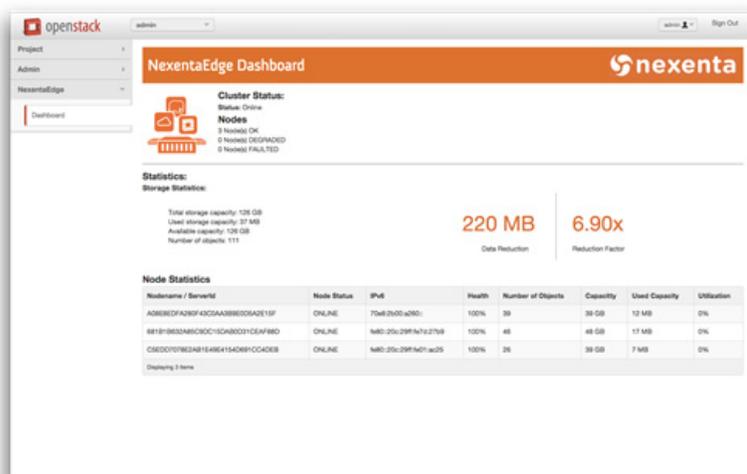
5. Save and close the `/usr/share/openstack-dashboard/openstack_dashboard/settings.py` file.

6. Restart the `apache` and `memcached` processes.

```
$ sudo sh -c "service apache2 restart && service memcached restart"
```

After you follow this procedure, the NexentaEdge Dashboard appears in the side menu of your Horizon Dashboard; for example:

Figure 6-1: NexentaEdge Dashboard in OpenStack Horizon



Registering the NexentaEdge Cluster as a Swift Storage System

To register the NexentaEdge cluster as an OpenStack Swift storage system, you create an object-store service for the NexentaEdge cluster, then create endpoints for this object-store service. After you restart the OpenStack Horizon HTTP server, the NexentaEdge cluster will be registered as a Swift storage system in Horizon. You will be able to create containers and add objects to the storage system from Horizon.

❖ *To register the NexentaEdge cluster as an OpenStack Swift storage system:*

1. Log in to the workstation where the OpenStack Keystone service is running.
2. Verify that you have proper credentials to issue Keystone commands.

Example:

```
$ keystone tenant-list
```

This command should produce output similar to the following:

id	name	enabled
b6999498cec44565a10b7175fa88f332	admin	True
2518f2e956874a5e96468e81a697981c	alt_demo	True
49a1d4d8a4a14beb8fd7a546e0c796bc	basic_tenant	True
8cb9382159a243069992eb791498da80	demo	True
5a8cf0239edd458c9f1af8839d39ba91	invisible_to_admin	True
23c584f44fd9486caf55436b04b10408	service	True
018dcd2252bd490d8b3d03143e1b2018	swifttenanttest1	True
287e400117b84e0fac1fba4ea97c15b	swifttenanttest2	True

3. Verify that an object-store service does not already exist. If one does, delete it.

Example:

```
$ keystone service-list
```

id	name	type	description
afc5fc5b8408413bb485d8b8db80e9fd	ec2	ec2	EC2 Compatibility Layer
6ad4f6db704f44fab51ae0d312ccf61d	glance	image	Glance Image Service
4b7d301f6b524339851109c3f5d88e9a	keystone	identity	Keystone Identity Service
f5db9a143b494f86aa522ed5f0c308e3	nova	compute	Nova Compute Service
f6ed0da15d224a3abaddf71b67db84e7	novav3	computev3	Nova Compute Service V3
6a4efe7a1194400f87a0284758c621f8	s3	s3	S3
b9d118ffd27e40b4b5b8fe9a47aea439	swift	object-store	Swift Service

```
$ keystone service-delete b9d118ffd27e40b4b5b8fe9a47aea439
```

4. If an OpenStack Swift storage service for the NexentaEdge cluster does not exist, create one. See [OpenStack Swift Deployment Example](#) for the relevant NEADM commands.
5. Create the NexentaEdge object-store service.

Example:

```
$ keystone service-create --name=nedge --type=object-store
--description="NexentaEdge Object Storage"
```

This command produces output similar to the following:

```
+-----+-----+
| Property | Value |
+-----+-----+
| description | NexentaEdge Object Storage |
| enabled | True |
| id | 75ef509da2c340499d454ae96a2c5c34 |
| name | nedge |
| type | object-store |
+-----+-----+
```

6. Copy the service ID from the output (shown in bold above).
7. Use the following command to create the service endpoint for the NexentaEdge cluster.

```
$ keystone endpoint-create \
  --service-id <service-id> \
  --publicurl 'http://<gateway-node-ip>:9981/v1/AUTH_%(tenant_id)s' \
  --internalurl 'http://<gateway-node-ip>:9981/v1/AUTH_%(tenant_id)s' \
  --adminurl http://<gateway-node-ip>:9981 \
  --region regionOne
```

Example:

```
$ keystone endpoint-create \
  --service-id 75ef509da2c340499d454ae96a2c5c34 \
  --publicurl 'http://10.3.30.20:9981/v1/AUTH_%(tenant_id)s' \
  --internalurl 'http://10.3.30.20:9981/v1/AUTH_%(tenant_id)s' \
  --adminurl http://10.3.30.20:8080 \
  --region regionOne
```

Where:

<code><service-id></code>	Is the service ID copied from the output of the <code>keystone service-create</code> command.
<code><gateway-node-ip></code>	Is the IP address of a gateway node for the storage service. The default ports for OpenStack Swift API calls are 9981 for HTTP and 443 for HTTPS.

`AUTH_%(tenant_id)s`

If deploying for a production environment, leave `AUTH_%(tenant_id)s` in the URL, but note that for each OpenStack project tenant, there must be a corresponding tenant in the NexentaEdge cluster with the name `AUTH_<project-tenant-id>`, where `<project-tenant-id>` is the ID of the OpenStack project's tenant in Keystone. Objects will be created under the relevant tenant for each project's object-storage interaction.

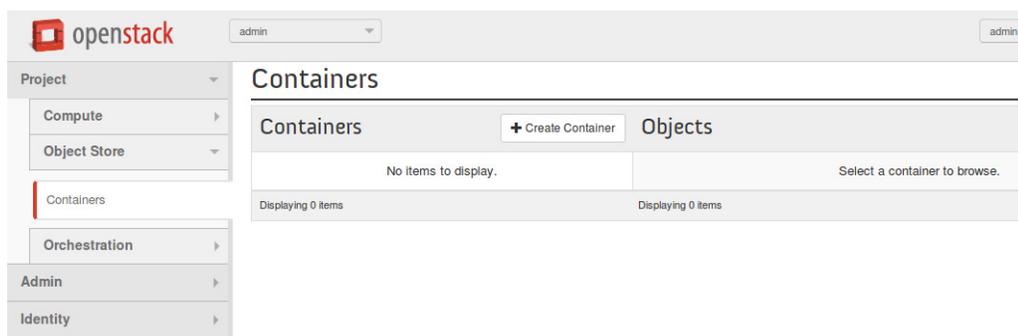
If deploying for a non-production environment, you can replace `AUTH_%(tenant_id)s` with any tenant configured in the NexentaEdge cluster. Objects will be created under this tenant regardless of the name of the OpenStack project's tenant.

- Refresh the OpenStack Horizon HTTP server to register the new component.

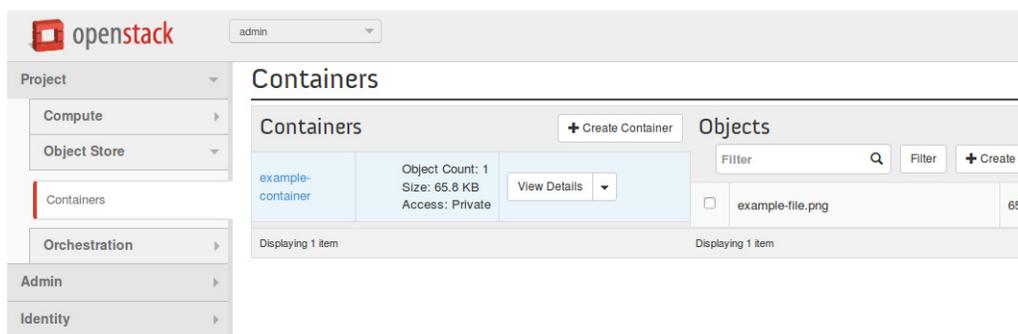
```
$ sudo service apache2 restart
```

Once the HTTP server restarts, the NexentaEdge cluster is registered as an OpenStack Swift storage system in OpenStack Horizon.

- Open a browser and go to your OpenStack page. Expand the list under Project to display an Object Store tab for the NexentaEdge object-store service. Click Containers to display the list of containers that exist within the service.



- From this page, you can click Create Containers to add a container, then add objects to the container.



Deploying NexentaEdge from an ISO File

This chapter includes the following topics:

- [Overview](#)
- [Dependencies](#)
- [Deployment Procedure](#)

Overview

During the normal installation and deployment process for NexentaEdge, packages are downloaded from the Internet and installed on the nodes. However, in some installations, downloading packages is not possible, since the nodes may be blocked from the Internet due to security requirements.

To accommodate this kind of installation, Nexenta provides a method to install the required packages from a repository contained in an ISO file instead of downloading them over the Internet. To deploy NexentaEdge using this method, you mount the ISO file on the deployment workstation, then run a script that causes the NEDEPLOY tool to get the packages from the mounted directory instead of the Internet.

Dependencies

All of the software and hardware requirements listed in the [Prerequisites](#) chapter apply when deploying NexentaEdge from an ISO file. In addition, the following core dependency packages must be available within the network where NexentaEdge is deployed:

- Netscape Portable Runtime (NSPR) for SSL support
- Net SNMP service and client
- Resource control group (cgroup) client
- Python interpreter and standard libraries

There should be a repo server containing these packages, and each node's APT/YUM configuration should point to that repo server.

Deployment Procedure

❖ *To deploy NexentaEdge using the ISO file:*

1. Obtain the NexentaEdge ISO file from Nexenta. The file will have a name similar to `nedge-darksite-deb_1.1.0FP1-1478.iso`. Copy the ISO file to the NexentaEdge deployment workstation.

2. Mount the ISO file on the NexentaEdge deployment workstation.

Example:

```
$ cd ~
$ mkdir nedgemnt
$ mount -o loop ~/nedge-darksite-deb_1.1.0FP1-1478.iso ~/nedgemnt
```

3. Run the `install.sh` script.

Example:

```
$ cd ~/nedgemnt
$ install.sh
```

4. At the prompt, enter the IP address of the deployment workstation.

Example:

```
Please enter this workstation's address
Hostname or IP: 10.3.30.35
```

5. The `install.sh` script starts the NEDEPLOY wizard. At this point, you can follow the standard NexentaEdge deployment procedure. See [Using the NEDEPLOY Wizard](#).

Global Headquarters

451 El Camino Real, Suite 201
Santa Clara, CA 95050
USA

2000-nedge-1.1-FP3-000010-A