

UniVirStore Manager User Manual

(Applicable for UVS Manager 3.16.1)

Change Logs

Document Version	Released Date	Description	Remarks
V-3.16.1	2022 December 27	Initial release Ceph version: Ceph pecific	Applicable for UVS 3.16.1

Table of Contents

Change Logs	2
Table of Contents	3
About Ceph	7
Plan and Configure your network before creating the Ceph cluster	7
Quick preview of the Ambedded UniVirStor (UVS) Manager	7
UVS manager License	9
Request & Upload the Permanent License	10
Cascading through each Menu options	11
UVS Manager Dashboard	11
Ceph Information	11
Ceph IO Status	12
Ceph Capacity Utilization, Placement Group Status & Recovery Speed	12
Main Manual	13
Host Management	16
Add host & Initial a Cluster	16
Open Web Terminal Command Line Interface	20
Remove hosts	20
NTP Server & System Time Zone	21
NTP - Time Sync for Ceph Cluster	21
Option 1: Use one of the Monitor nodes as the NTP server.	21
Option 2: Use an existing or external one as the NTP server.	22
Set the Time Zone	23
Push Time Zone & NTP configuration to all nodes	24
Deploy More Ceph Monitor Nodes	25
Deploy Ceph OSD services	25
Add OSD on Mars500	25
Add OSD on Mars400	27
OSD Encryption	31
Locate Storage Devices	33
Remove and Recycle the OSD	33
Node Rolling Reboot	36
Manipulating CRUSH – Crown of the Ceph Storage Cluster	38
Viewing the CRUSH Map Tree – Graphic Representation	38
Creating CRUSH Buckets – segregate the leaf nodes	40
Moving CRUSH bucket	42
Managing Rulesets – Create & List CRUSH Ruleset	43
Rename Bucket Type	45

Managing Ceph Pools – Your pool of resources to use	47
PG Number	47
Creating Ceph Pool	48
Create Replicated Pool	48
Erasure Code Profile	50
Create the Erasure Code Profiles	50
Create Erasure Coded pool	51
Delete Ceph Pools	53
Cache Tiering	53
Create RBD Images: Very own Block devices to map and use	55
Creating a Ceph RBD Image	55
Resizing RBD Images	56
Delete RBD Images	57
RBD Snapshots	58
RBD Snapshots Management	58
Create a Snapshot Manually	59
Create a Snapshot Schedule	59
Edit and Delete a Schedule	60
Delete Snapshot	61
Purge Snapshot	62
Rollback	62
Snapshot Clone	63
Create a Clone – copy-on-write child	63
Flatten a cloned image – Detaching child from a parent	64
RBD Mirroring - Asynchronous replicate your block device to another Ceph cluster for disaster protection	66
Create RBD Mirroring	67
Add Mirror Schedule	70
Show Mirror Status Details	73
Add Additional Images for Mirroring	73
Add Mirroring to Mirror Images from the Other Site	74
Remove a Mirror	75
Failover & Failback - Promotion, Demotion	76
Add More Sites for RBD Mirroring	77
Destroy Mirroring	79
CephX Authentication	80
Create CephX User	80
Edit CephX User	81
Ceph Object Storage	83
RADOS Gateway Management	83
Prepare Software on External Server Nodes	83
Create RADOS Gateway	84

Creating RADOS Gateways on external servers	85
Configure RGW Pools	85
SSL Encryption Options	87
Multiple RGW instances	87
Delete RADOS Gateway	90
Ceph Object Storage User Management	91
Create New Users	91
Edit User Quota	91
Delete a User	92
Ceph Object Storage Pool management – RGW own pools	93
Editing RADOS Gateway Pools	94
Setting Up Multi-Site RADOS Gateways	96
Steps to Build Multi-Site RADOS Gateways	96
Creating Master RADOS Gateway	97
Creating Secondary RADOS Gateway	97
Promoting Secondary to Master	101
iSCSI	103
iSCSI Gateway	103
Run iSCSI gateways on Mars monitor nodes	104
Create External iSCSI Gateway	105
Delete External iSCSI Gateway	107
Create LUN	107
Delete LUN	109
Configuring The iSCSI Initiators	110
iSCSI Initiator for Microsoft Windows	110
iSCSI Initiator for VMware ESX	110
iSCSI Initiator for Linux	110
Ceph Filesystem	112
Creating CephFS	112
Auto Mode	113
Manual Mode	114
Create More Metadata Servers	116
MDS Failover and Failback	116
Integrating Ceph with OpenStack – Reliable Storage Backend	118
Creating OpenStack Backend pools & Auth Keys	119
Download and save Keyring and ceph.conf file	119
Creating OpenStack pools and users manually – Doing Ceph way	120
Audit Logs – tracking the actions on WebUI	124
Audit Logs	124
Output Logs – the details to the UVS operations	125
Prometheus	126
Exporter services	126

Prometheus service	127
Change the MTU of Mars Nodes	130
Notification – Alerts on email	131
Dashboard User – Edit and View rights	133
Firmware Update	135
UPD Update	135
Package Update	136
Ceph-MGR Zabbix Module	137
Enable the Zabbix Module	137
The Zabbix Template	138
NFS & SAMBA protocol - using Ceph RBD Image	139
Steps to create RBD Image and map to Linux client	139
Making RBD Image mount persistent	140
Steps to create NFS Server	140
Steps to create SAMBA Server	141
RBD-NBD – MAP RBD IMAGES TO NBD DEVICE	142
Install RBD-NBD Package & Module	142
Copy Ceph configurations & keyrings	142
Map the RBD Image	142
Appendixes	144
Appendix 1. Configure the network before deploying a Ceph cluster	144
Appendix 2. Prepare the External RGW Hosts for Ceph Octopus	144

About Ceph

Ceph uniquely delivers object, block, and file storage in one unified system. Ceph is highly reliable, easy to manage, and free. Ceph delivers extraordinary scalability—thousands of clients accessing petabytes to exabytes of data. A Ceph Node leverages commodity hardware and intelligent daemons, and a Ceph Storage Cluster accommodates large numbers of nodes, which communicate with each other to replicate and redistribute data dynamically.

At the heart of every Ceph deployment is the Ceph Storage Cluster. It consists of two types of daemons:

- Ceph OSD Daemon: Ceph OSDs store data on behalf of Ceph clients. Additionally, Ceph OSDs utilize the CPU and memory of Ceph nodes to perform data replication, rebalancing, recovery, monitoring, and reporting functions.
- Ceph Monitor: A Ceph Monitor maintains a master copy of the Ceph cluster map with the current state.

From the end-user perspective, a Ceph storage cluster looks like a simple pool where it stores data. The storage cluster performs many complex operations that are transparent to the clients.

These operations include making replicas of data, using the selected CRUSH algorithm for placement of PG's across the cluster, etc.

To enable end-users to create and manage the Ceph cluster effectively, the UniVirStor Manager comes in handy.

Let us have a quick walk-through of Ambedded UniVirStor Manager WebUI to show available options and features; however, before accessing Ambedded UniVirStor Manager WebUI, we need to perform administrative tasks to make the nodes accessible on the local network.

Plan and Configure your network before creating the Ceph cluster

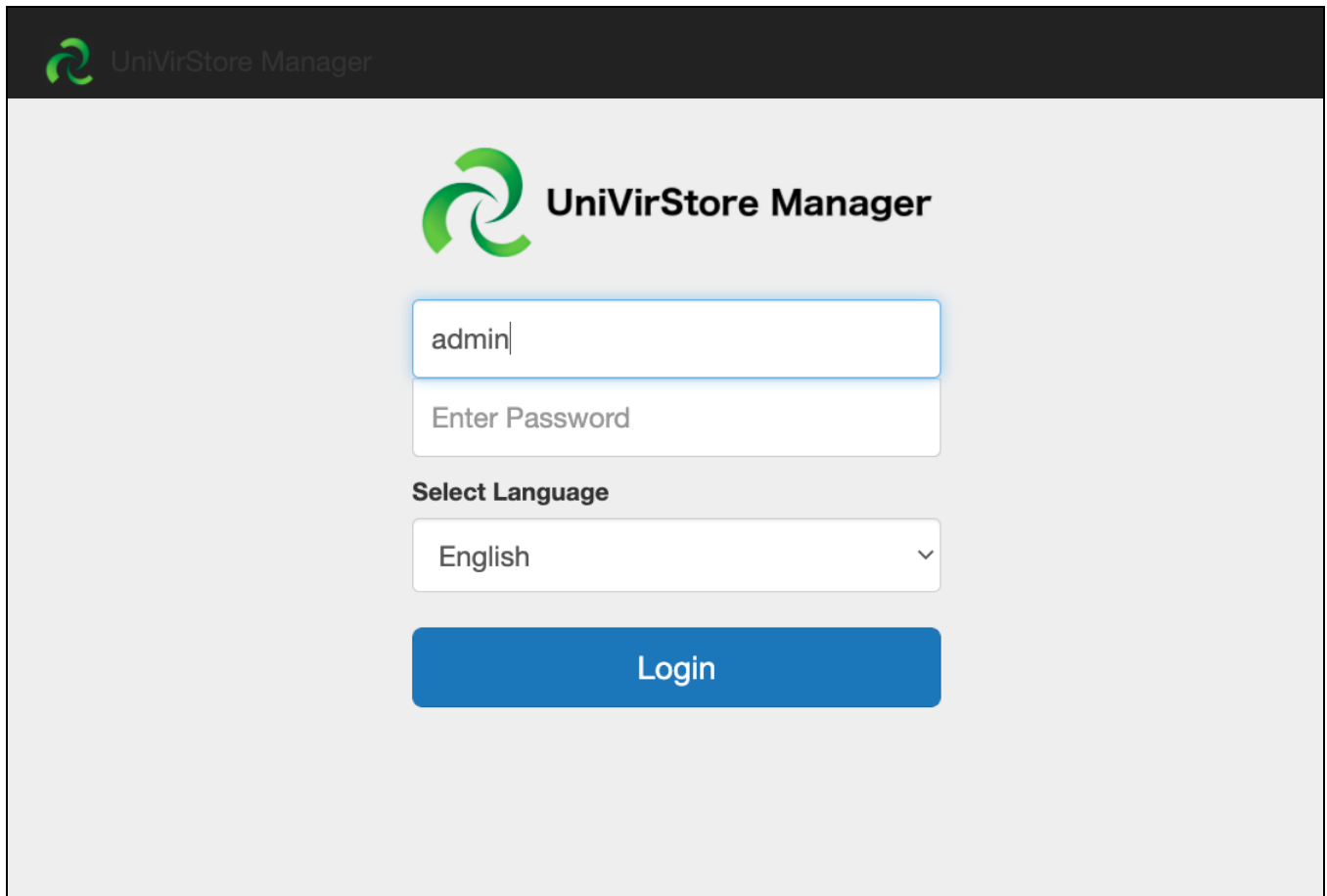
Please refer to the Mars 500 deployment guide for details of planning your network and deploy the cluster procedure.

Quick preview of the Ambedded UniVirStor (UVS) Manager

The Ambedded Mars 500 UVS Manager is available on all nodes running the ceph monitors. Please sing the URL:

https://IP_ADDR_OF_MON to open the UVS manager web UI.

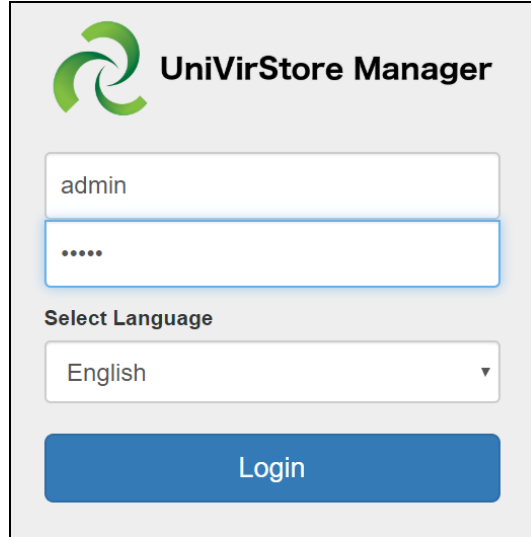
You get to the homepage, as shown below.




The image shows the UniVirStore Manager login page. At the top, there is a dark header bar with the UniVirStore Manager logo and name. Below this, the main content area is light gray. In the center, there is a green circular logo with a stylized 'U' shape. To the right of the logo, the text 'UniVirStore Manager' is displayed. Below the logo, there are two input fields: the first contains the text 'admin|' and the second is labeled 'Enter Password'. Below these fields, there is a 'Select Language' dropdown menu with 'English' selected. At the bottom, there is a blue 'Login' button.

Enter the provided Login name and respective password. The default username and password are “admin/admin”. You can change the password after entering the UVS manager.

Here comes the first feature of this Ceph management WebUI, multilingual support. You can select your preferred language from the drop-down menu namely; English, Spanish and Chinese.

The image shows a login interface for UniVirStore Manager. It features the Ambedded logo and the product name at the top. Below this are two input fields: the first contains the text 'admin' and the second contains five dots, indicating a password field. A 'Select Language' dropdown menu is positioned below the password field, with 'English' selected. At the bottom of the form is a blue 'Login' button.

 **UniVirStore Manager**

admin

.....

Select Language

English ▼


Login

At the bottom of the login page, you can find the copyright & UVS software version.

© 2020 [Ambedded Technology Co. Ltd.](#); Taiwan. All Rights Reserved | UniVirStore Manager Version 2.15.18 for aarch64

UVS manager License

The UVS manager needs a valid license to work correctly. There is a temporary license is activated automatically when you deploy the new cluster. You can check the last valid date from the Settings → About page. You have to get a permanent license based on the maximum number of OSD you can deploy in your cluster. The demo license is only valid for six months. Please check the section [Upload License](#) for how to request and upload a permanent license.

 **UniVirStore Manager** UniVirStore Manager Version 3.16

Licensed To:

Activation Code:

Valid Till:

Licensed Maximum OSDs:

Current Usage: 47 OSDs

(47 Nodes)

[New License Request](#) [Upload License](#)

- * Ceph is a registered trademark of Red Hat, Inc.
- * Some code for the dashboard is used from ceph-dash project (<https://github.com/Crapworks/ceph-dash>) which is a copyright of Christian Eichele
- * Other code and/or content of this project is a copyright of Ambedded Technology Co. Ltd., Taiwan.

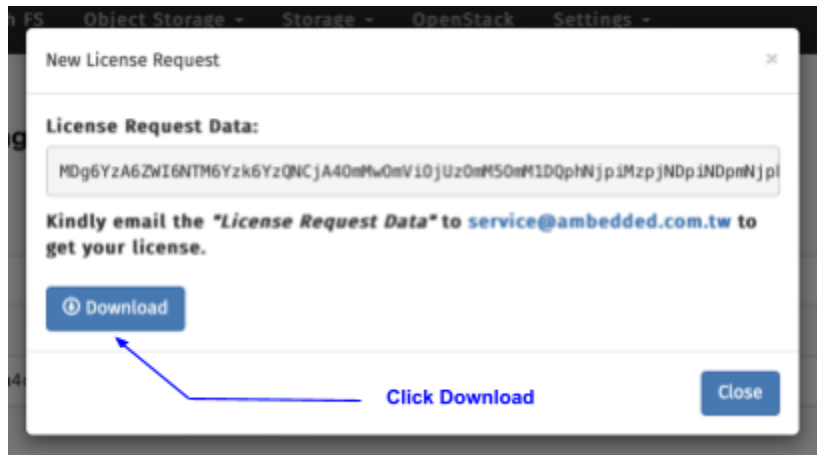
Request & Upload the Permanent License

After you have deploy you cluster you shall upload your permanent license before the temporary license becomes expired.

Here is the process for how to request and upload the license.

Note: When you add more Mars nodes in the future for scaling out the cluster, you have to request a new license that activates all OSD nodes.

1. Click the [New License Request](#) to download the request file. You will get a file named “requestfile.lic”.



2. Email the request file to service@ambedded.com.tw. We will send the license to you.
3. Go back to the **About** page and Click **Upload License** to upload the license.

IMPORTANT → Ensure that you have deployed all MON nodes before requesting for a license.

Cascading through each Menu options

UVS Manager Dashboard

There are three sections in the dashboard

- Ceph Information
- Ceph IO status
- Capacity & Utilization, Placement Group Status and Recovery Speed.

Ceph Information





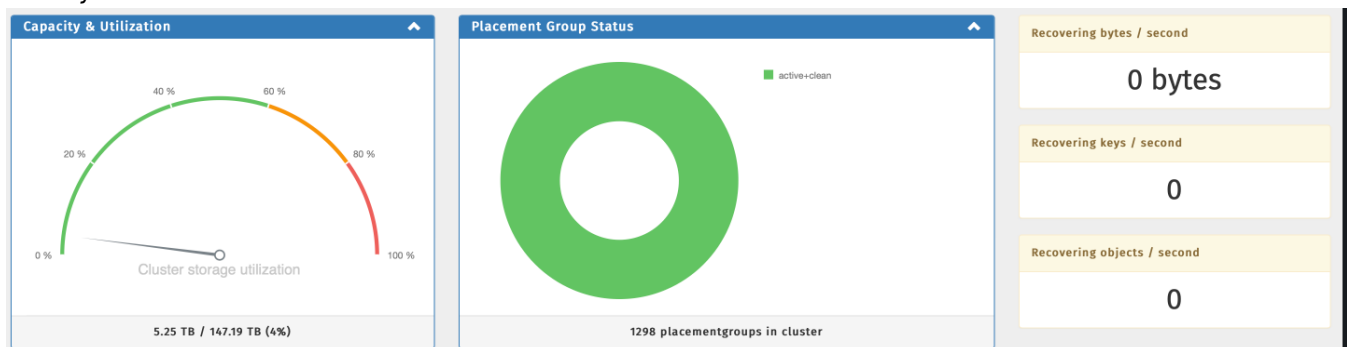
Ceph IO Status

You can click the “Write” or “Read” under the graph to switch the display of read and write or both.

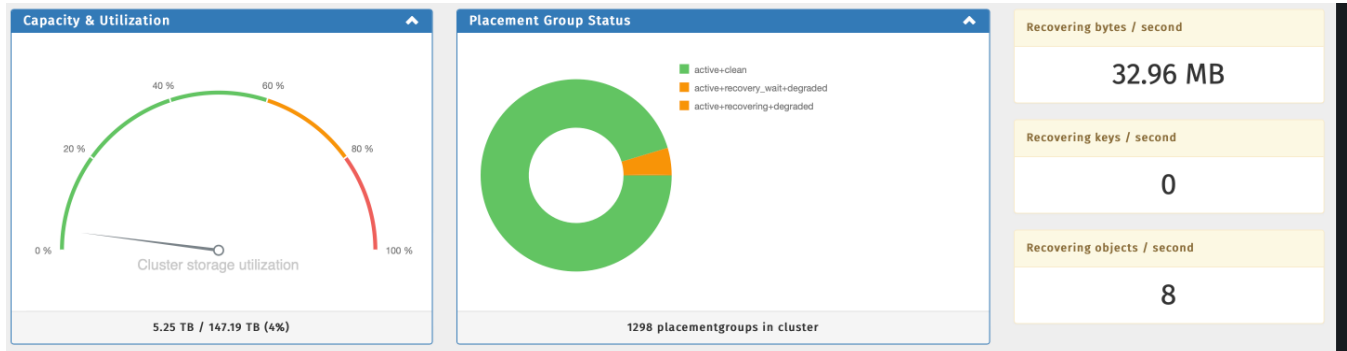


Ceph Capacity Utilization, Placement Group Status & Recovery Speed

Healthy Status



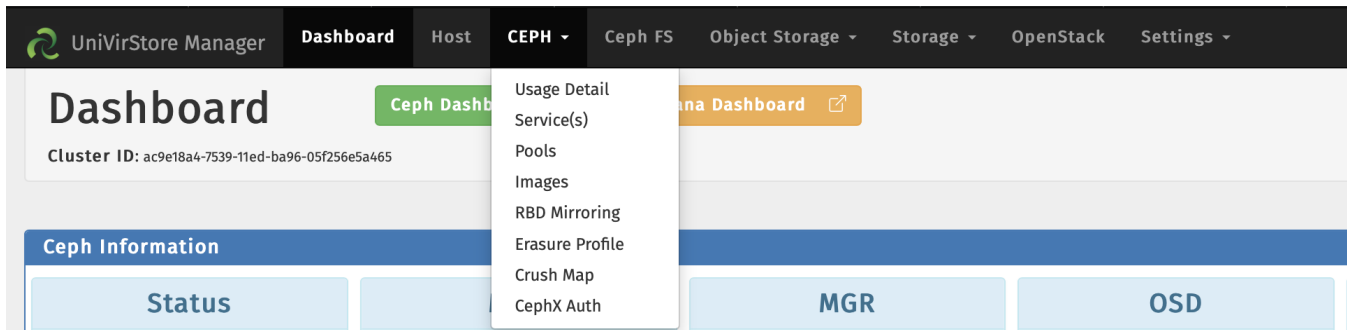
Unhealthy Status



Main Manual

You get main options Ceph, Object Storage, Storage, OpenStack, Settings, and Logout on the top bar.

The Ceph menu leads you to a sub-menu like the picture below.



Usage Detail: This option displays the usage details of all root, host, chassis, and rack defined in CRUSH Map and pools. The usage details of root, host, chassis, and rack include weight, size, used, available, and %Used. The usage details of the pool include used, used%, maximum capacity available, and objects.

Services: Selecting this option gives a list of Ceph MON, MGR, OSD and MDS services, which are part of the Ceph cluster. Using this, you can do MON & OSD service restart, reboot a Ceph node, and remove OSD from the Ceph cluster.

Pools: This option allows you to create, remove pools based on replicated or Erasure profiles. It also allows you to add a pool as a Cache tier for the erasure-code-based cold storage. A cache tier provides Ceph Clients with better I/O performance for a subset of the data stored in a backing storage tier.

Images: A Ceph Block Device is a block device image over multiple objects in the Ceph Storage Cluster, where each object gets mapped to a placement group and distributed, and the placement groups are spread across separate Ceph OSD daemons throughout the cluster. Ceph users can use Ceph Block Device kernel objects to provide a block device to a client.

RBD Mirroring: This feature helps you set the RBD mirroring service. The RBD mirroring service mirrors RBD images asynchronously between two Ceph clusters. This capability uses the RBD journaling image feature to

ensure crash-consistent replication between clusters. You can replicate all images in specific pools or the particular images to another Ceph cluster.

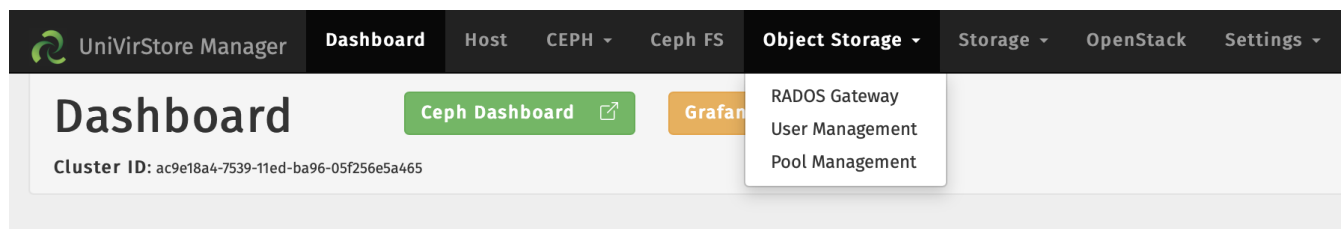
Erasure Profile: You can use the Erasure Profile function to create Erasure profiles. An erasure-coded pool stores each object as K+M chunks. Erasure coding divides an object into K data chunks and M coding chunks. The erasure pool has a size of K+M, so Ceph stores each chunk in the OSDs that belong to a placement group.

CRUSH Map: It contains a list of storage devices, the failure domain hierarchy (e.g., disk, host, rack, row, room, etc.), and rules for traversing the hierarchy when storing data. CRUSH map allows you to create and manage this hierarchy while having a complete graphical view.

CephX Auth: You can use this option to create Ceph users and their associated keyring to access the Ceph pools. You can also edit the capability of a user to limit his accessibility to a specific pool.

CephFS: This manual allows users to create and manage the Ceph file system.

Next in line is the **Object Storage** sub-menu. You can use it to create and manage RADOS gateways, object storage users, and storage pools.

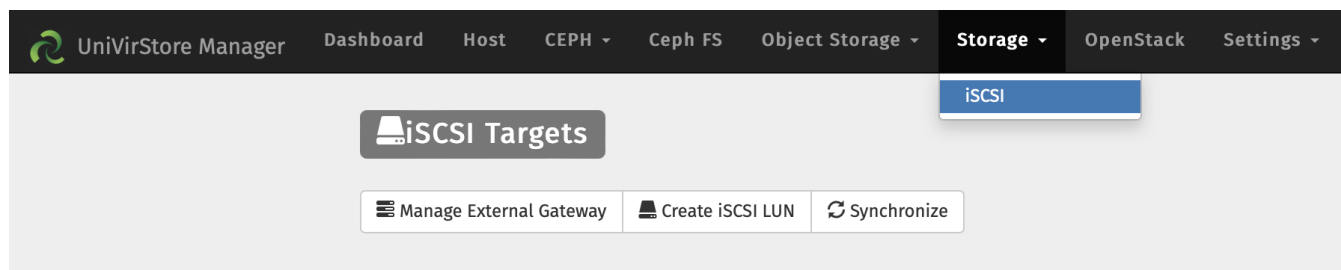


Rados Gateway: This allows Ceph users to create an RGW host for using Swift / S3 API and use the Ceph cluster as the Object Store service. Users can create a standalone gateway as well as gateways for multisite operation.

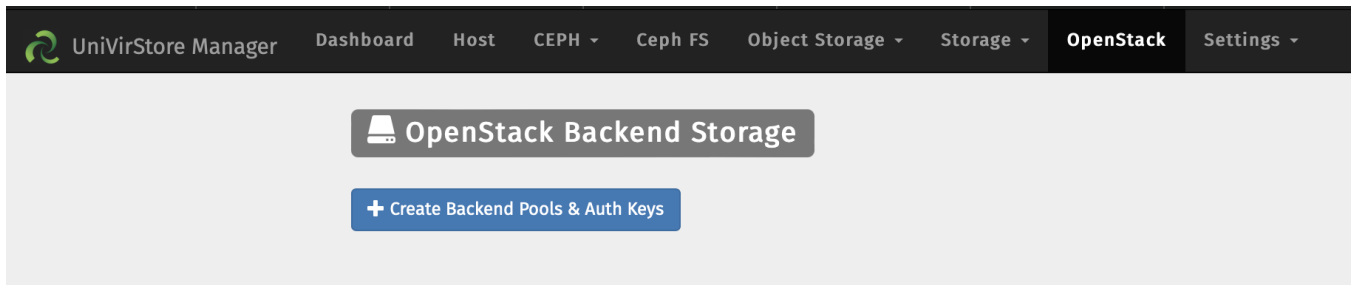
User Management: This sub-menu is for S3/Swift User Management. You can create and delete RGW users to use with Swift and S3. It also allows setting Quota for users.

Pool Management: This sub-menu allows users to list the RGW pools and also edit them as per requirement like changing CRUSH ruleset, changing PG numbers, replica size, etc.

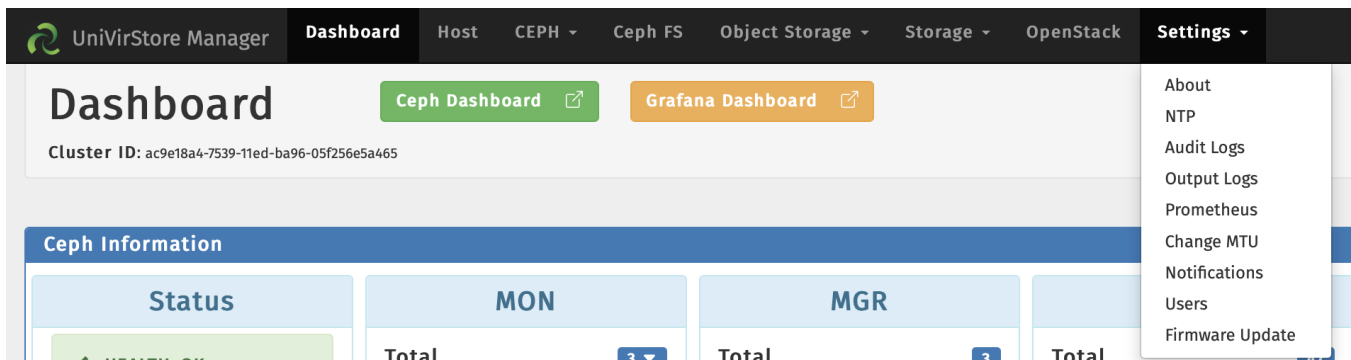
The third menu item is for the **Storage** option, which allows you to create an **iSCSI** target using a pool/image, to be used as a RAW device for iSCSI LUN.



The fourth option **OpenStack** helps create users for Cinder, Glance, and Nova to connect existing OpenStack setup and use the Ceph cluster as a backend to store Glance Images, Nova instances, and give Cinder blocks.



The fifth menu item, Settings, has several sub-menu options, which help administer a Ceph cluster and diagnose the state of the same. It also has a feature to set up and configure an NTP server, time zone, and MTU for all nodes. It also allows you to review UVS operation logs and set up a remote Syslog server for storing the audit logs. The Prometheus helps you to monitor the details of Mars400 nodes & the Ceph cluster. Use The Users option to manage UVS users. You can also configure the Notification for cluster events and send the notifications through emails. The Firmware update feature helps you to update UVS software easily.



Host Management

Host List

Enter key word to filter

Host Name	Model	Node ID	Cores	RAM Size	Network	# HDD	# SSD	# NVME	Roles	Status
ubuntu-2004-mars500-228	Mars500	-	64	93.67 GiB	192.168.3.228/24 192.168.4.228/24 192.168.5.228/24	N/A	N/A	10	OSD*8 RGW*2 MGR*1 MON*1	PKSS
ubuntu-2004-mars500-219	Mars500	-	64	93.67 GiB	192.168.3.219/24 192.168.4.219/24 192.168.5.219/24	N/A	N/A	10	OSD*8 MON*1 MGR*1	PKSS
ubuntu-2004-mars500-218	Mars500	-	64	93.67 GiB	192.168.3.218/24 192.168.4.218/24 192.168.5.218/24	N/A	N/A	10	MON*1 OSD*8 MGR*1 RGW*2	PKSS
node178-1bd1f	Mars400	8				1	1	N/A	OSD*1	PKSS
node177-1bdd	Mars400	7	4	3.84 GiB	192.168.3.177/24 192.168.4.177/24	1	1	N/A	OSD*1	PKSS
node176-1bdb	Mars400	6	4	3.84 GiB	192.168.3.176/24 192.168.4.176/24	1	1	N/A	OSD*1	PKSS
node175-1bd9	Mars400	5	4	3.84 GiB	192.168.3.175/24 192.168.4.175/24	1	1	N/A	OSD*1	PKSS
node174-1bd7	Mars400	4	4	3.84 GiB	192.168.3.174/24 192.168.4.174/24	1	1	N/A	OSD*1	PKSS
node173-1bd5	Mars400	3	4	3.84 GiB	192.168.3.173/24 192.168.4.173/24	1	1	N/A	OSD*1	PKSS
node172-1bd3	Mars400	2	4	3.84 GiB	192.168.3.172/24 192.168.4.172/24	1	1	N/A	OSD*1	PKSS

Showing 1 to 10 of 27 rows | 10 rows per page

[Add MON\(s\)](#)
[Add OSD\(s\)](#)
[Remove Host\(s\)](#)
[+ Add Host](#)

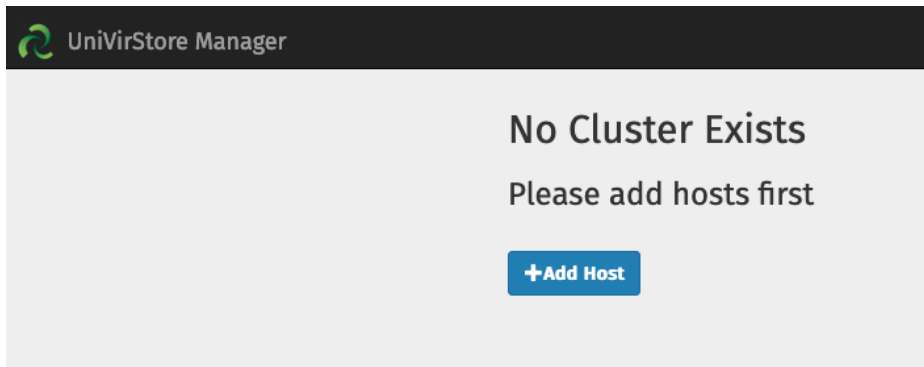
The UVS Manager uses the host page to allow the user to view and manage the host inventory. The features of this page are

1. Add hosts for deploying ceph services.
2. Open the web terminal to ssh to a Mars500 Linux shell.
3. Add monitor, OSD.
4. Displays detailed information of all host's models, Node ID (for Mars 400 only), CPU cores, RAM size, Network IP address, storage drive, Ceph service run on the host, and their status.
5. A filter box for searching host by a keyword.

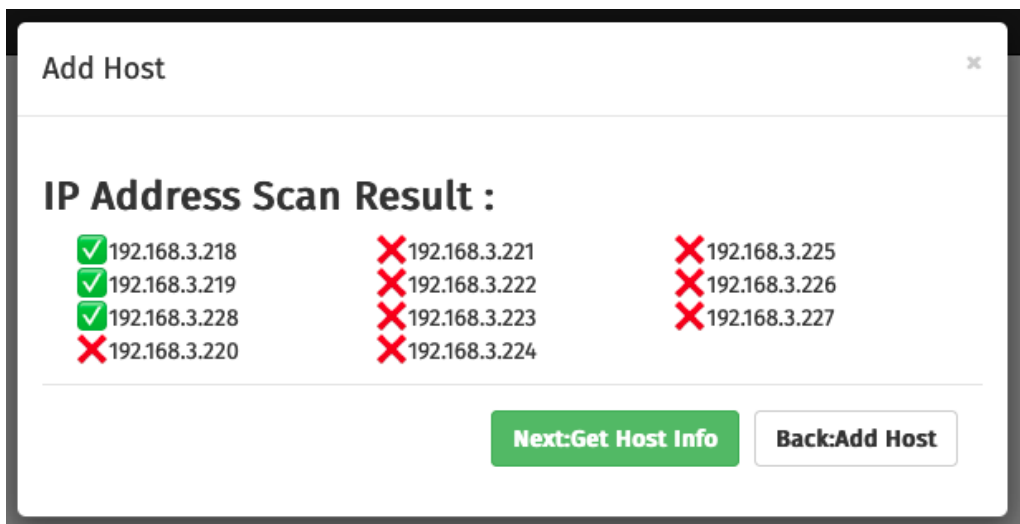
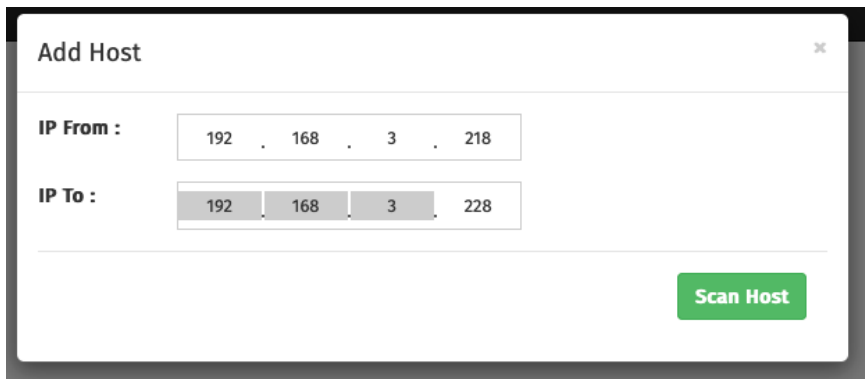
Add host & Initial a Cluster

Before deploying any ceph services, you shall add hosts to the host inventory. You can add multiple hosts by scanning an IP address range to find the available hosts. Hosts that have Ambedded UVS pre-installed can be managed by the UVS manager.

When you start to deploy a new cluster, open a browser from a computer that can access the management IP of Mars500 hosts. You will see the following output.



Click the **+Add Host** button to bring up the dialog box. Enter the **ceph public IP V4 address** range to the scan hosts.



Click **Next: Get Host Info**

Host List

click here to view device details

Host Name	Model	Node ID	Cores	RAM Size	Network	# HDD	# SSD	# NVME	Roles	Status
<input checked="" type="checkbox"/> mars500-218	Mars500	-	64	93.68 GiB	192.168.3.218/24 192.168.4.218/24 192.168.5.218/24	N/A	N/A	10		PASS
<input type="checkbox"/> mars500-219	Mars500	-	64	93.68 GiB	192.168.3.219/24 192.168.4.219/24 192.168.5.219/24	N/A	N/A	10		PASS
<input type="checkbox"/> mars500-228	Mars500	-	64	93.68 GiB	192.168.3.228/24 192.168.4.228/24 192.168.5.228/24	N/A	N/A	10		PASS

Showing 1 to 3 of 3 rows

click here to initial the cluster

Initial Cluster & Add MONs + Add Host

UniVirStore Manager

Host List

Host Name	Model	Node ID	Cores	RAM Size	Network	# HDD	# SSD	# NVME	Roles	Status
<input checked="" type="checkbox"/> mars500-218	Mars500	-	64	93.68 GiB	192.168.3.218/24 192.168.4.218/24 192.168.5.218/24	N/A	N/A	10		PASS
<input checked="" type="checkbox"/> mars500-219	Mars500	-	64	93.68 GiB	192.168.3.219/24 192.168.4.219/24 192.168.5.219/24	N/A	N/A	10		PASS
<input checked="" type="checkbox"/> mars500-228	Mars500	-	64	93.68 GiB	192.168.3.228/24 192.168.4.228/24 192.168.5.228/24	N/A	N/A	10		PASS

Showing 1 to 3 of 3 rows

Initial Cluster & Add MONs + Add Host

Select hosts to deploy monitors

Select correct public network, cluster network & Time Zone

Initial New Cluster & Create MONs

This will Initial New Cluster and Create MONs:

Add MON(s):

- 192.168.3.218 mars500-218
- 192.168.3.219 mars500-219
- 192.168.3.228 mars500-228

Public Network:

192.168.3.218/24

Cluster Network:

192.168.4.218/24

TimeZone:

Continent:

Asia

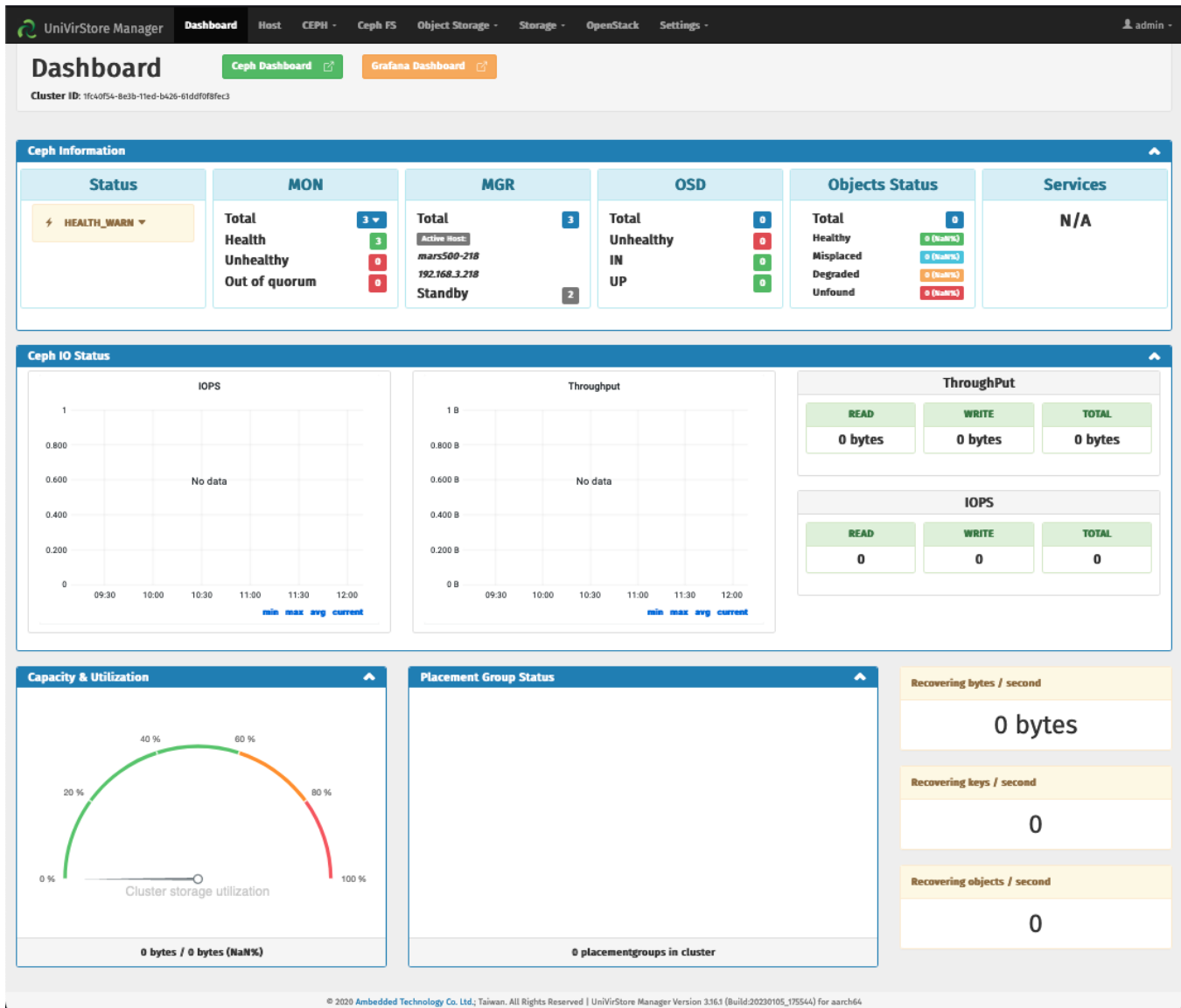
City (State):

UTC/GMT +08:00 - Asia/Taipei

Create Cluster

After initial cluster and add monitors, the UVS web UI will be available on any hosts that run Ceph monitor service.

Beside the UVS manager, you can also open the Ceph Dashboard and Grafana Dashboard by clicking the icons on top of the UVS manager dashboard.



UniVirStore Manager **Host** CEPH - Ceph FS Object Storage - Storage - OpenStack Settings - admin -

Host List

Host Name	Model	Node ID	Cores	RAM Size	Network	# HDD	# SSD	# NVME	Roles	Status
mars500-218	Mars500	-	64	93.68 GiB	192.168.3.218/24 192.168.4.218/24 192.168.5.218/24	N/A	N/A	10	MON*1 MGR*1	PASS
mars500-219	Mars500	-	64	93.68 GiB	192.168.3.219/24 192.168.4.219/24 192.168.5.219/24	N/A	N/A	10	MON*1 MGR*1	PASS
mars500-228	Mars500	-	64	93.68 GiB	192.168.3.228/24 192.168.4.228/24 192.168.5.228/24	N/A	N/A	10	MGR*1 MON*1	PASS

Showing 1 to 3 of 3 rows

[Add MON\(s\)](#)
[Add OSD\(s\)](#)
[Remove Host\(s\)](#)
[+ Add Host](#)

After the Cluster is deployed with Monitors and OSD, you can add Mars 500 and Mars 400 to the inventory.

When deploying a cluster with mix of Mars500 and Mars400, the Mars 400 nodes can only be used as OSD.

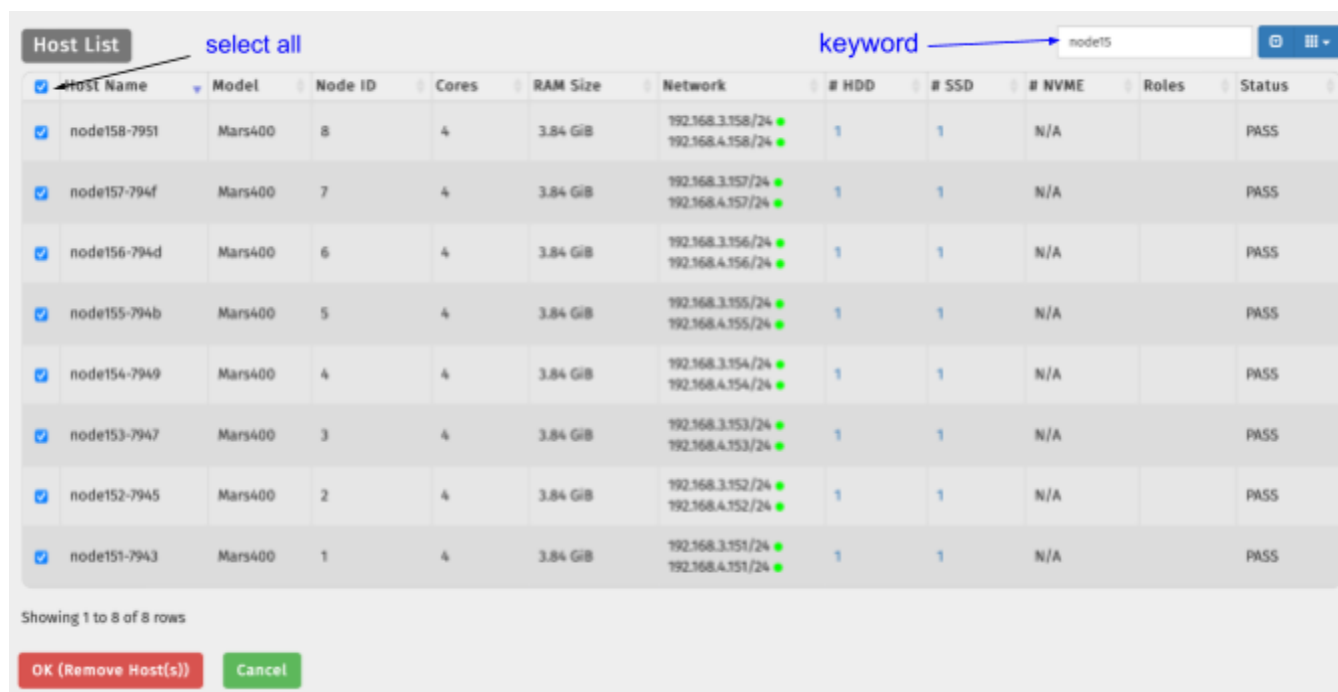
Open Web Terminal Command Line Interface

You can click the terminal icon beside the Mars500 host name to open the web terminal so you can log in to use the command line interface.

Remove hosts

You can remove hosts from the inventory. You can only remove hosts without running any Ceph service. Click the **Remove Host(s)** button, and the UVS manager will list the hosts can be removed. After UVS list the hosts can be removed, you can also use the filter to find out the hosts you want to remove.

For example, you have a cluster with mixing of Mars 500 and Mars 400 and you want to remove some Mars 400 node. You can put a keyword such as part of hostname to filter the hosts you want to remove. Click the check box to select all or the specific hosts you want to remove. Finally, click the **OK (Remove Host(s))** button to execute the remove.



The screenshot shows the 'Host List' interface. At the top, there is a 'Host List' tab and a 'select all' link. A search bar labeled 'keyword' contains the text 'node15'. Below the search bar is a table with columns: Host Name, Model, Node ID, Cores, RAM Size, Network, # HDD, # SSD, # NVME, Roles, and Status. The table lists 8 hosts, all of which are Mars400 models. Each row has a checkbox in the Host Name column, and all checkboxes are checked. At the bottom of the table, there is a 'Showing 1 to 8 of 8 rows' indicator and two buttons: 'OK (Remove Host(s))' and 'Cancel'.

Host Name	Model	Node ID	Cores	RAM Size	Network	# HDD	# SSD	# NVME	Roles	Status
<input checked="" type="checkbox"/> node158-7951	Mars400	8	4	3.84 GiB	192.168.3.158/24 192.168.4.158/24	1	1	N/A		PASS
<input checked="" type="checkbox"/> node157-794f	Mars400	7	4	3.84 GiB	192.168.3.157/24 192.168.4.157/24	1	1	N/A		PASS
<input checked="" type="checkbox"/> node156-794d	Mars400	6	4	3.84 GiB	192.168.3.156/24 192.168.4.156/24	1	1	N/A		PASS
<input checked="" type="checkbox"/> node155-794b	Mars400	5	4	3.84 GiB	192.168.3.155/24 192.168.4.155/24	1	1	N/A		PASS
<input checked="" type="checkbox"/> node154-7949	Mars400	4	4	3.84 GiB	192.168.3.154/24 192.168.4.154/24	1	1	N/A		PASS
<input checked="" type="checkbox"/> node153-7947	Mars400	3	4	3.84 GiB	192.168.3.153/24 192.168.4.153/24	1	1	N/A		PASS
<input checked="" type="checkbox"/> node152-7945	Mars400	2	4	3.84 GiB	192.168.3.152/24 192.168.4.152/24	1	1	N/A		PASS
<input checked="" type="checkbox"/> node151-7943	Mars400	1	4	3.84 GiB	192.168.3.151/24 192.168.4.151/24	1	1	N/A		PASS

Showing 1 to 8 of 8 rows

OK (Remove Host(s)) **Cancel**

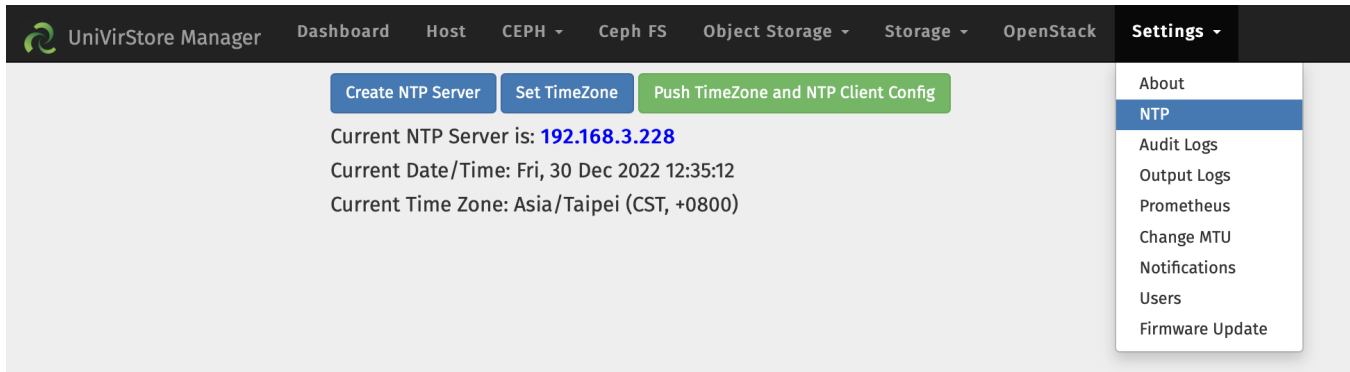
NTP Server & System Time Zone

After installing the first MON, the UVS manager also deploys an NTP server on the first-built monitor. You can change the NTP server to any other existing NTP server or keep using this NTP server for all nodes in this cluster. The NTP server is crucial for keeping a low clock skew between all Ceph MON nodes for proper operation.

NTP - Time Sync for Ceph Cluster

Time sync is critical in Ceph. Nodes need to have a clock skew of less than 50 ms. Otherwise, bad things can happen, especially on the monitor nodes. Note by default, the maximum allowed drift between nodes is 0.05 seconds!!! So, let's be sure the time is in sync!

To ensure this, you can create a local NTP server and sync other nodes with that. The NTP server will ensure our cluster works efficiently and correctly even when offline.



The screenshot shows the UniVirStore Manager web interface. The top navigation bar includes links for Dashboard, Host, CEPH, Ceph FS, Object Storage, Storage, OpenStack, and a Settings dropdown menu. The Settings menu is open, showing options like About, NTP (highlighted), Audit Logs, Output Logs, Prometheus, Change MTU, Notifications, Users, and Firmware Update. Below the navigation bar, there are three buttons: 'Create NTP Server', 'Set TimeZone', and 'Push TimeZone and NTP Client Config'. Below these buttons, the current NTP server is listed as 192.168.3.228, the current date/time is Fri, 30 Dec 2022 12:35:12, and the current time zone is Asia/Taipei (CST, +0800).

You have 2 options to configure or set up an NTP server.

Option 1: Use one of the Monitor nodes as the NTP server.

Option 2: Use an existing or external one as the NTP server.

Select the **NTP** option from the "**Settings**" menu.

You will reach a web page as shown below. As this is the first node, we don't have an NTP server in place. We need to create an NTP server or use an existing external NTP server.

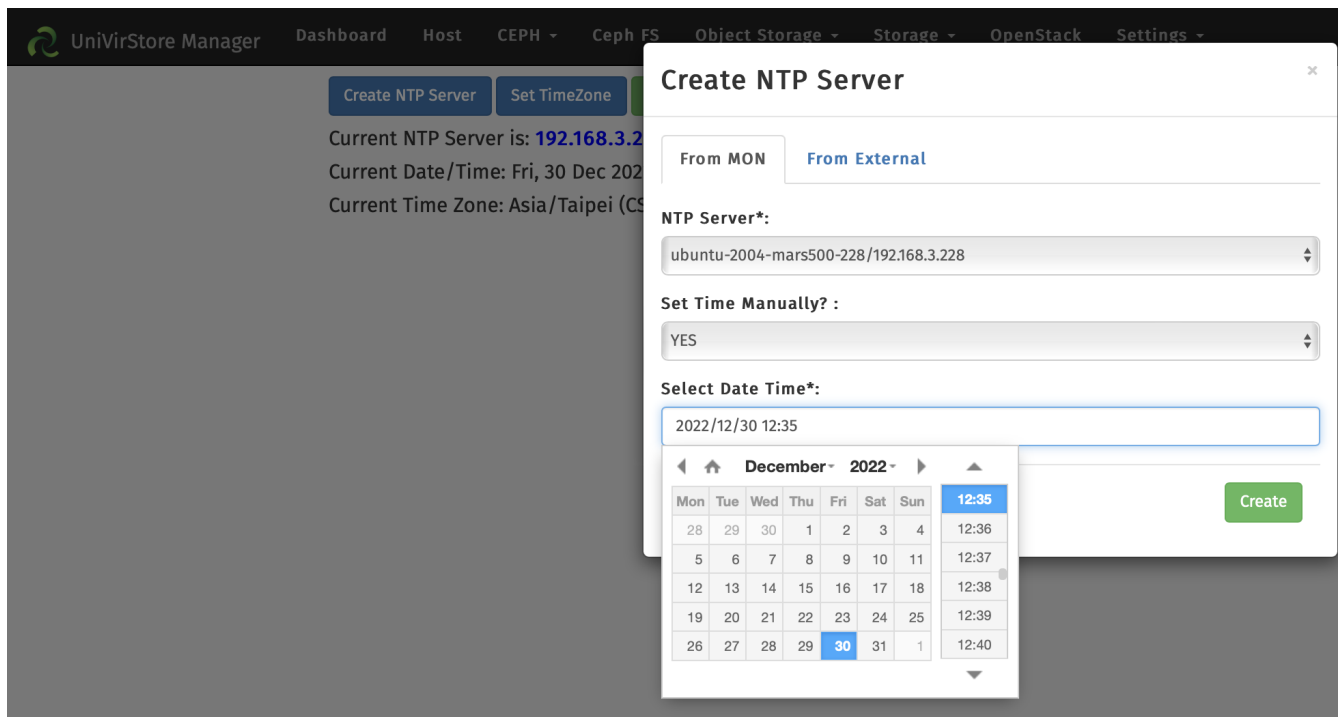
Option 1: Use one of the Monitor nodes as the NTP server.

You can keep using the NTP server on the MON node or later change the NTP server to one of the other new deployed monitor nodes.

Select an IP address for the Monitor node, which will act as an NTP server from the dropdown list.

If you Not to set time manually, UVS will use the time of the time of yor computer running the UVS web; else, you can select YES to "Set Time Manually".

Click "**Create**" to start the process.



UniVirStore Manager Dashboard Host CEPH Ceph FS Object Storage Storage OpenStack Settings

Create NTP Server Set TimeZone

Current NTP Server is: 192.168.3.228
Current Date/Time: Fri, 30 Dec 2022 12:35:00
Current Time Zone: Asia/Taipei (CST, +0800)

Create NTP Server

From MON From External

NTP Server*:
ubuntu-2004-mars500-228/192.168.3.228

Set Time Manually? :
YES

Select Date Time*:
2022/12/30 12:35

December 2022

Mon	Tue	Wed	Thu	Fri	Sat	Sun
28	29	30	1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	31	1

12:35
12:36
12:37
12:38
12:39
12:40

Create

Once the process is complete, we'll get the current NTP server and Date/Time information on the NTP page.

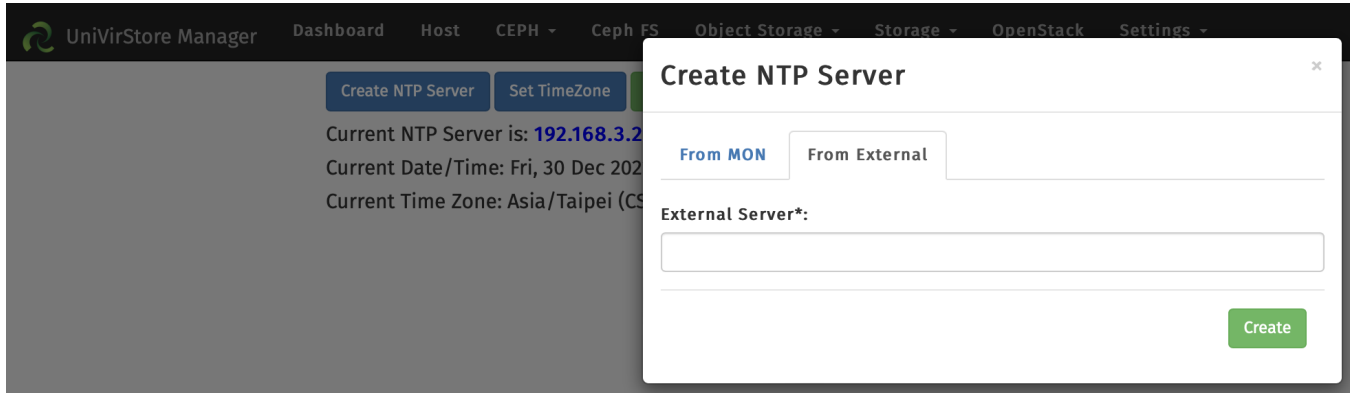


Create NTP Server Set TimeZone Push TimeZone and NTP Client Config

Current NTP Server is: 192.168.1.111
Current Date/Time: Mon, 22 Feb 2021 14:09:28
Current Time Zone: Asia/Taipei (CST, +0800)

Option 2: Use an existing or external one as the NTP server.

Click “**From External**” to set the IP address or domain name of an NTP server. Please ensure you have properly set the gateway and DNS IP address on all nodes that will join this Ceph cluster.

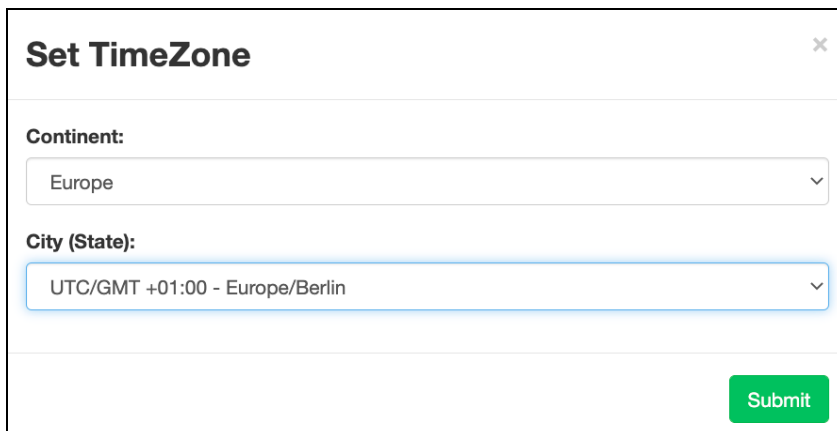


Once the process is complete, we will get the current NTP server and Date/Time information on the NTP page.



Set the Time Zone

Click The “Set TimeZone” button, and you can find the dialog box. Please select your Time Zone and submit it, then the Ceph cluster will apply the new settings in a few seconds.



[Create NTP Server](#)[Set TimeZone](#)[Push TimeZone and NTP Client Config](#)

Current NTP Server is: tw.pool.ntp.org

Current Date/Time: Mon, 22 Feb 2021 07:22:29

Current Time Zone: Europe/Berlin (CET, +0100)

Push Time Zone & NTP configuration to all nodes

By default, the Ceph monitors report health HEALTH_WARN clock skew detected on Monitor errors if the nodes' clocks differ by more than 50 milliseconds. If the cluster has time skew between monotors, you should push the correct time & time zone to all cluster nodes.

Click "**Push TimeZone and NTP Client Config**" to sync the settings.

```
==> [Change TimeZone to "Europe/Berlin"] *****
==> [Update Hardware Clock] *****
==> [fail the play if ntp_server is not defined] *****
==> [Copy Client Config] *****
==> [restart_chrony] *****
==> [Force NTP Sync] *****
==> [SUCCESS] *****
Exit status : 0
```

This will make all the nodes carry-on sync processes so that they all have the same time.

Deploy More Ceph Monitor Nodes

Ceph uses the Paxos algorithm to keep data consistent. The Ceph cluster requires an **odd number** of monitors. We recommend you create a minimum of three monitors on three Mars500 nodes to establish a monitor quorum and provide redundancy. If you have more nodes available, you can deploy more monitors on any Mars500 hosts.

Go to the “**Host**” page and click the **Add MON(s)** button. UVS will show the list of available hosts for deploying ceph monitors. Select the host you want to add monitors.

Three or five monitors are good enough for redundancy.

After deploying the Ceph monitors, the UVS manager will also deploy the Ceph manager services on the same node.

The Ceph manager daemon (ceph-mgr) is a daemon that runs alongside monitor daemons to provide monitoring and interfacing to external monitoring and management systems.

If there is no mgr daemon running, you will see a health warning to that effect, and some of the other information in the output of ceph status will be missing or stale until an mgr is started.

You can find the hosts run Ceph monitor and mgr in the host list in the Host page.

Deploy Ceph OSD services

Add OSD on Mars500

Go to “**Host**” then click “Add OSD (s)”

Click the **Add OSD(s)** button. You will get a pop-up as shown below. And click **List Host** You can deploy OSDs on one host at a time.

Select Candidate Hosts

Select Model:

Mars500

Select Host (Mars500):

mars500-218

List Hosts Close

Host Name	Model	Node ID	Cores	RAM Size	Network	# HDD	# SSD	# NVME	Roles	Status
mars500-218	Mars500	-	64	93.68 GiB	192.168.3.218/24 192.168.4.218/24 192.168.5.218/24	N/A	N/A	10	MON*1 MGR*1	PASS

Showing 1 to 1 of 1 rows

Click here to select all available drives

Click here to select a specific drive

Select/Unselect All

☒ mars500-p1

Type: NVME
Name: Micron_7400_MTFDCKBN6010Z
Capacity: 894.25 GiB
Status: Not Used

☒ mars500-p1

Type: NVME
Name: Micron_7400_MTFDCKBN6010Z
Capacity: 894.25 GiB
Status: Not Used

☒ mars500-p1

Type: NVME
Name: Micron_7400_MTFDCKBN6010Z
Capacity: 894.25 GiB
Status: Not Used

☒ mars500-p1

Type: NVME
Name: Micron_7400_MTFDCKBN6010Z
Capacity: 894.25 GiB
Status: Not Used

OSD Encryption:

NO

Select encrypt or not to encrypt the OSDs.

OK (Add OSD(s)) Cancel

After OSDs are deployed.

UniVirStore Manager Dashboard Host CEPH Ceph FS Object Storage Storage OpenStack Settings admin

Host List

New OSDs

Host Name	Model	Node ID	Cores	RAM Size	Network	# HDD	# SSD	# NVME	Roles	Status
mars500-218	Mars500	-	64	93.68 GiB	192.168.3.218/24 192.168.4.218/24 192.168.5.218/24	N/A	N/A	10	OSD*8 MON*1 MGR*1	PASS
mars500-219	Mars500	-	64	93.68 GiB	192.168.3.219/24 192.168.4.219/24 192.168.5.219/24	N/A	N/A	10	MON*1 MGR*1	PASS
mars500-228	Mars500	-	64	93.68 GiB	192.168.3.228/24 192.168.4.228/24 192.168.5.228/24	N/A	N/A	10	MGR*1 MON*1	PASS

Showing 1 to 3 of 3 rows

Add MON(s) Add OSD(s) Remove Host(s) + Add Host

If you check the ceph status right after new OSDs are deployed, you will see Health Warn. The stray daemons not managed by cephadm will be eliminated after several minutes. You can continue to deploy OSDs on other hosts.

The Degrade data redundancy: 1 pg undersized warning will be normal until OSDs are deployed on more than three hosts.

```
root@mars500-218:~# ceph -s
cluster:
  id:          1fc40f54-8e3b-11ed-b426-61ddf0f8fec3
  health: HEALTH_WARN
              8 stray daemon(s) not managed by cephadm
              Degraded data redundancy: 1 pg undersized

services:
  mon: 3 daemons, quorum mars500-218,mars500-219,mars500-228 (age 7m)
  mgr: mars500-218.keorzf(active, since 7m), standbys: mars500-219.dgdnxp,
mars500-228.bfaf1h
  osd: 8 osds: 8 up (since 5m), 8 in (since 6m); 1 remapped pgs

data:
  pools: 2 pools, 2 pgs
  objects: 1 objects, 32 KiB
  usage: 43 MiB used, 7.0 TiB / 7.0 TiB avail
  pgs: 1 active+clean
      1 active+undersized+remapped
```

Add OSD on Mars400

You can add Mars400 to join an existing Mars500 cluster. For this configuration, you can deploy OSDs on Mars400 but not Ceph monitor. Using Mars400 and Mars500 in one cluster gives the advantages of

1. Simplifying the Management of all host by one UVS manager and Ceph command interface.
2. Opportunity of use NVMe OSD pool as the cache tier of HDD pool

Each Mars400 appliance has eight hosts. You can deploy one osd per host. There are some steps to do before deploying OSD on Mars 400.

1. Create a new CRUSH root for separating the OSDs on Mars400 and Mars500. Using separated CRUSH root allows you to create Ceph pools with option of use NVMe or HDD.
2. Create CRUSH chassis buckets for every Mars400 host in the same chassis.
3. Create a CRUSH rule for Mars400 use.
4. Use Host management to include Mars400 hosts into host list.
5. Create CRUSH rules for using Mars400 HDD pool

Please refer to the details in the [Ceph Document](#) and [Manipulating CRUSH section](#) in this document.

Example:

CrushMap Tree

+ Create Bucket

Move Bucket

Crush Rule

Delete Bucket

Bucket Type Rename

LEGEND

11 root

10 region

9 zone

8 datacenter

7 room

6 pod

5 pdu

4 row

3 rack

2 chassis

1 host

0 osd

Expand All

Collapse All

- 11 mars400root

+ 2 mars400bmc109

+ 2 mars400bmc119

+ 2 mars400bmc129

- 11 default

+ 1 mars500-218

+ 1 mars500-219

+ 1 mars500-228

Create a root to host Mars400 chassis & multiple chassis for Mars400

UniVirStore Manager Dashboard Host CEPH Ceph FS Object Storage Storage OpenStack Settings admin										
Host List										
Host Name	Model	Node ID	Cores	RAM Size	Network	# HDD	# SSD	# NVME	Roles	Status
mars500-218	Mars500	-	64	93.68 GiB	192.168.3.218/24 192.168.4.218/24 192.168.5.218/24	N/A	N/A	10	OSD*8 MON*1 MGR*1	PASS
mars500-219	Mars500	-	64	93.68 GiB	192.168.3.219/24 192.168.4.219/24 192.168.5.219/24	N/A	N/A	10	OSD*8 MON*1 MGR*1	PASS
mars500-228	Mars500	-	64	93.68 GiB	192.168.3.228/24 192.168.4.228/24 192.168.5.228/24	N/A	N/A	10	OSD*8 MGR*1 MON*1	PASS
node151-cc9c	Mars400	1	4	3.84 GiB	192.168.3.151/24 192.168.4.151/24	1	1	N/A		PASS
node152-cc9e	Mars400	2	4	3.84 GiB	192.168.3.152/24 192.168.4.152/24	1	1	N/A		PASS
node153-cca0	Mars400	3	4	3.84 GiB	192.168.3.153/24 192.168.4.153/24	1	1	N/A		PASS
node154-cca2	Mars400	4	4	3.84 GiB	192.168.3.154/24 192.168.4.154/24	1	1	N/A		PASS
node155-cca4	Mars400	5	4	3.84 GiB	192.168.3.155/24 192.168.4.155/24	1	1	N/A		PASS
node156-cca6	Mars400	6	4	3.84 GiB	192.168.3.156/24 192.168.4.156/24	1	1	N/A		PASS
node157-cca8	Mars400	7	4	3.84 GiB	192.168.3.157/24 192.168.4.157/24	1	1	N/A		PASS
node158-ccaa	Mars400	8	4	3.84 GiB	192.168.3.158/24 192.168.4.158/24	1	1	N/A		PASS
node161-962d	Mars400	1	4	3.84 GiB	192.168.3.161/24 192.168.4.161/24	1	1	N/A		PASS
node162-962f	Mars400	2	4	3.84 GiB	192.168.3.162/24 192.168.4.162/24	1	1	N/A		PASS

Create Crush Rule ✕

Rule Name*:

Select Parent (Root)*:

Type:

Bucket:

Select Failure Domain*:

Select Device Class*:

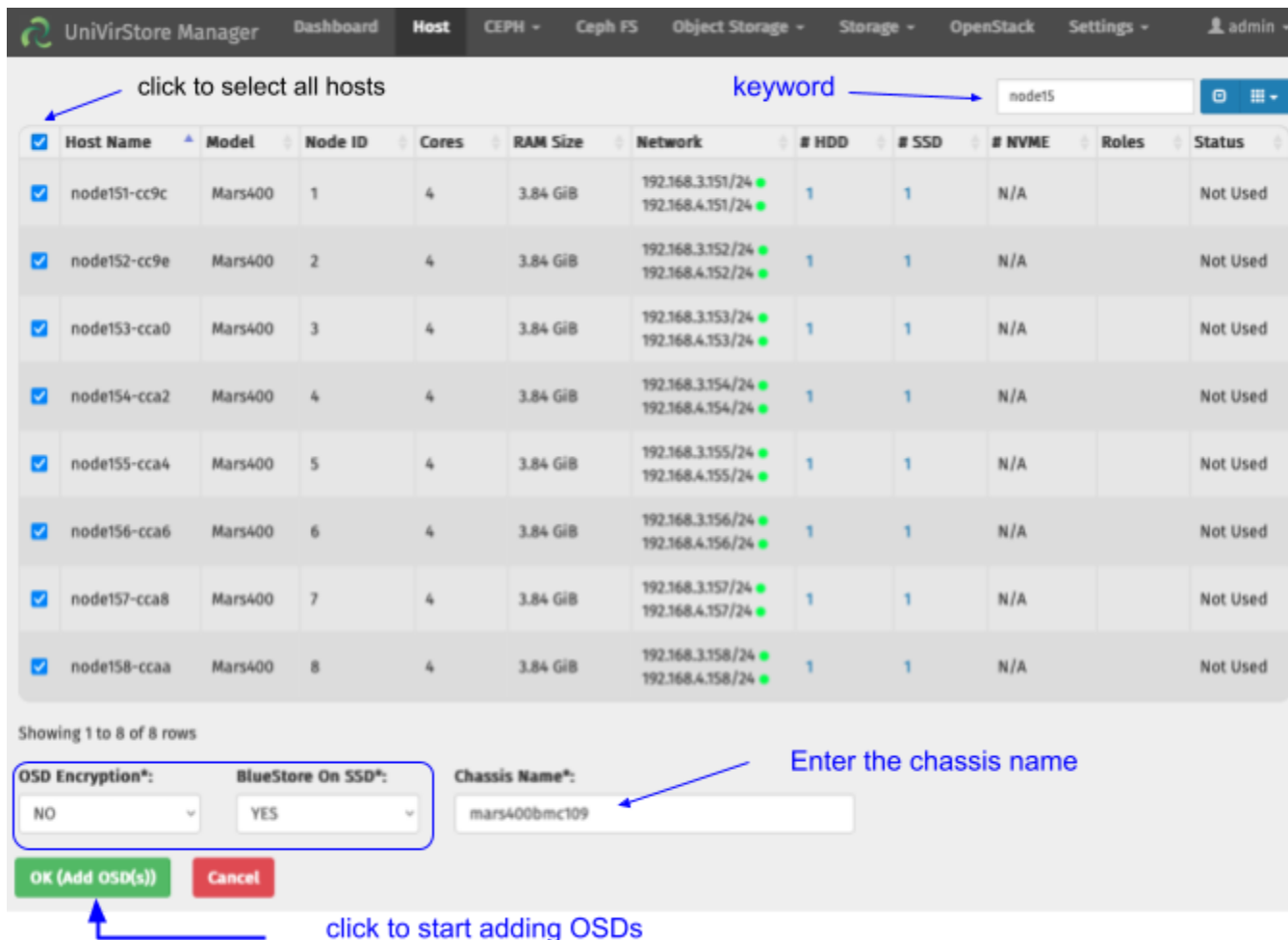
Steps for adding Mar400 OSDs

1. Click the **Add OSD(s)** button
2. Select Model
3. Click **List Hosts**
4. Use the keyword in search box to find out hosts in the chassis you want to add OSD. You can add OSD in the same chassis in one time.
5. Select encrypt or not to encrypt the OSD. You can not change this after the OSD id created.
6. Select using SSD as BlueStore log & DB device.
7. Enter the chassis name of the Mars400 appliance you want to create OSD.

Select Candidate Hosts ✕

Select Model:

Select Chassis (Mars400):



UniVirStore Manager Dashboard Host CEPH Ceph FS Object Storage Storage OpenStack Settings admin

click to select all hosts keyword node15

<input checked="" type="checkbox"/>	Host Name	Model	Node ID	Cores	RAM Size	Network	# HDD	# SSD	# NVME	Roles	Status
<input checked="" type="checkbox"/>	node151-cc9c	Mars400	1	4	3.84 GiB	192.168.3.151/24 192.168.4.151/24	1	1	N/A		Not Used
<input checked="" type="checkbox"/>	node152-cc9e	Mars400	2	4	3.84 GiB	192.168.3.152/24 192.168.4.152/24	1	1	N/A		Not Used
<input checked="" type="checkbox"/>	node153-cca0	Mars400	3	4	3.84 GiB	192.168.3.153/24 192.168.4.153/24	1	1	N/A		Not Used
<input checked="" type="checkbox"/>	node154-cca2	Mars400	4	4	3.84 GiB	192.168.3.154/24 192.168.4.154/24	1	1	N/A		Not Used
<input checked="" type="checkbox"/>	node155-cca4	Mars400	5	4	3.84 GiB	192.168.3.155/24 192.168.4.155/24	1	1	N/A		Not Used
<input checked="" type="checkbox"/>	node156-cca6	Mars400	6	4	3.84 GiB	192.168.3.156/24 192.168.4.156/24	1	1	N/A		Not Used
<input checked="" type="checkbox"/>	node157-cca8	Mars400	7	4	3.84 GiB	192.168.3.157/24 192.168.4.157/24	1	1	N/A		Not Used
<input checked="" type="checkbox"/>	node158-cca8	Mars400	8	4	3.84 GiB	192.168.3.158/24 192.168.4.158/24	1	1	N/A		Not Used

Showing 1 to 8 of 8 rows

OSD Encryption*: NO BlueStore On SSD*: YES Chassis Name*: mars400bmc109

OK (Add OSD(s)) Cancel

Enter the chassis name

click to start adding OSDs

Every Mars400 node has an M.2 SSD by default. Select the Journal on SSD to enable using SSD as BlueDB & WAL storage. You shall also select a chassis defined in the CRUSH map for these OSD to be added to a chassis directly. If you haven't defined the chassis for CRUSH, please create the chassis first even if you can use the default root now as an OSD location and change its place later. Please refer to the [CRUSH rule section](#) for how to configure and manage the CRUSH map and define the CRUSH rule with a failure domain.

OSD Encryption

You can enable the OSD disk encryption when you create the OSD. Ceph uses dm-crypt for encryption. OSD-encryption helps if a disk gets pulled out of the system. It is also a great way to ensure that no one can salvage sensitive data from a broken disk, which you return for RMA or recycle.

You can also view the SMART information of data disk & metadata disk connected to the OSD nodes.

OSD

+ Create New OSD

Service Restart


Node Reboot

Remove OSD

Move To Recycle

Search

<input type="checkbox"/>	Rank			Class	Node Name	IP Address	PG	Status	Data SMART	MetaData SMART
<input type="checkbox"/>	osd.0			hdd	demo114-1078	↑192.168.1.114	0	Up	<div></div> <div></div> <div></div>	<div></div> <div></div> <div></div>
<input type="checkbox"/>	osd.1			hdd	demo115-107a	↑192.168.1.115	0	Up	<div></div> <div></div> <div></div>	<div></div> <div></div> <div></div>
<input type="checkbox"/>	osd.2			hdd	demo116-107c	↑192.168.1.116	0	Up	<div></div> <div></div> <div></div>	<div></div> <div></div> <div></div>
<input type="checkbox"/>	osd.3			hdd	demo117-107e	↑192.168.1.117	0	Up	<div></div> <div></div> <div></div>	<div></div> <div></div> <div></div>
<input type="checkbox"/>	osd.4			hdd	demo118-1080	↑192.168.1.118	0	Up	<div></div> <div></div> <div></div>	<div></div> <div></div> <div></div>

Click the “Info”  icon to show the SMART information. If any HDD has developed some issue, it will be shown in complete detail when you click on the “Error” button in the Smart Status column in that OSD row. (Shown below).

Data Drive Details
<div>Health Status: PASSED</div> <div>SMART Info:</div> <div> <pre>smartctl 6.5 2016-05-07 r4318 [aarch64-linux-4.19.52-MARS400+] (local build) Copyright (C) 2002-16, Bruce Allen, Christian Franke, www.smartmontools.org === START OF INFORMATION SECTION === Device Model: ST4000VN0001-1SF178 Serial Number: Z4F08NQ2 LU WWN Device Id: 5 000c50 07bb8ccae Firmware Version: AN02 User Capacity: 4,000,787,030,016 bytes [4.00 TB] Sector Sizes: 512 bytes logical, 4096 bytes physical Rotation Rate: 7200 rpm Form Factor: 3.5 inches Device is: Not in smartctl database [for details use: -P showall] ATA Version is: ACS-3 T13/2161-D revision 3b SATA Version is: SATA 3.1, 6.0 Gb/s (current: 6.0 Gb/s) Local Time is: Mon Nov 18 10:32:04 2019 UTC SMART support is: Available - device has SMART capability. SMART support is: Enabled</pre> </div>

Data Drive Details

Health Status: Error (Err Code: 32)

```
smartctl 6.5 2016-05-07 r4318 [aarch64-linux-4.19.52-MARS400+] (local build)
Copyright (C) 2002-16, Bruce Allen, Christian Franke, www.smartmontools.org

=== START OF READ SMART DATA SECTION ===
SMART overall-health self-assessment test result: PASSED
Please note the following marginal Attributes:
ID# ATTRIBUTE_NAME          FLAG     VALUE WORST THRESH TYPE      UPDATED  WHEN_FAILED RAW_VALUE
190 Airflow_Temperature_Cel 0x0022   057     039   040    Old_age  Always    In_the_past 43 (Min/Max 34/49 #2)
```











































Locate Storage Devices

Inside the Mars400, each server node has two SATA devices. One device is a 3.5" or 2.5" SATA drive, which you use for OSD data storage, and another M.2 SSD, which we will use for the Bluestore WAL/DB or Monitor storage. If you want to find which chassis and where the storage devices are inside the chassis, you can click the LED icons on the UVS manager node page to make the LED beside the device blink. After you enable the LED blinking of a specified device, an LED on the chassis's front panel blinks too. This feature makes locating the physical location of the storage device very easy. Click the LED off icon to disable the blinking.

+ Create New OSD

Service Restart
Node Reboot
Remove OSD
Move To Recycle

Search

<input type="checkbox"/>	Rank		Node Name	IP Address	PG	Status	Data SMART	MetaData SMART
<input type="checkbox"/>	osd.3		demo15-107a	↑192.168.1.115	51	Up	  	  
<input type="checkbox"/>	osd.4		demo16-107c	↑192.168.1.116	58	Up	  	  
<input type="checkbox"/>	osd.5		demo17-107e	↑192.168.1.117	56	Up	  	  
<input type="checkbox"/>	osd.6		demo18-1080	↑192.168.1.118	55	Up	  	  
<input type="checkbox"/>	osd.0		demo12-1074	↑192.168.1.112	64	Up	  	  
<input type="checkbox"/>	osd.1		demo13-1076	↑192.168.1.113	53	Up	  	  
<input type="checkbox"/>	osd.2		demo14-1078	↑192.168.1.114	47	Up	  	  

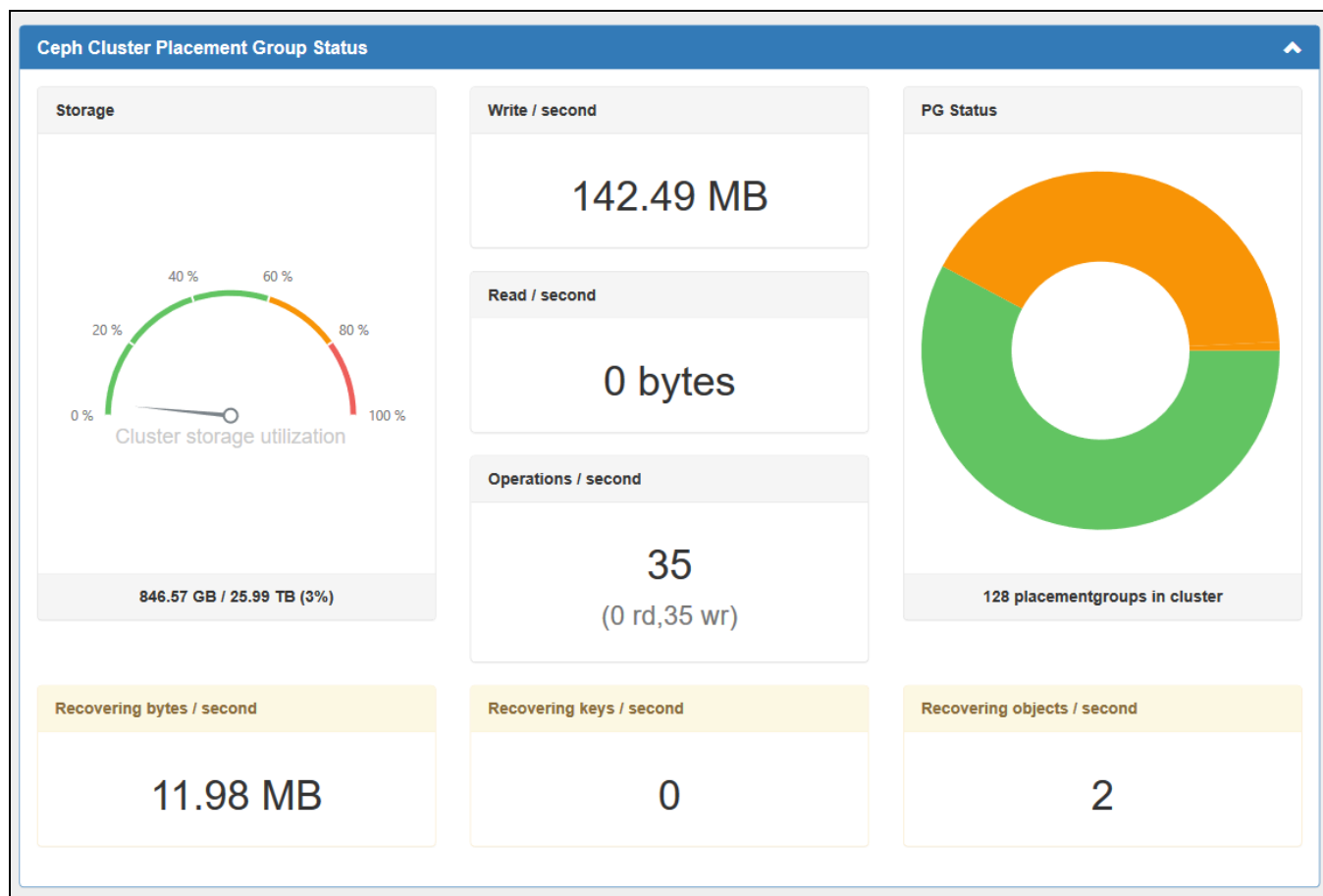
Remove and Recycle the OSD
























When you want to reduce the size of a cluster or replace OSD hardware, you may remove an OSD at runtime. With Mars400, the OSD is generally one Ceph OSD daemon for one storage drive within a microserver node. Generally, it's a good idea to check your cluster's capacity to see if you are reaching the upper end of its capacity. Ensure that your cluster is not at its near-full-ratio (85% of total capacity). For data safety, please remove one OSD at a time and make sure to remove it when the cluster is healthy.

UVS manager provides two ways to remove the OSD. You can remove the OSD directly by using “**Remove OSD**”. Removing OSD will delete the copy of the data stored in this OSD immediately. Before the data are re-healed, you have one copy of the data or erasure code chunk lost. Those placement groups which contain this OSD are in a degraded state until Ceph re-heal the deleted replication by other OSDs. After the cluster becomes healthy, you can replace the faulty hardware.

The alternative way to remove the OSD is to move the OSD to a recycle bin before destroying it. Moving OSD to recycle is a safer way of removing the OSD. When we click the “**Move To Recycle**” button, the UVS manager moves the OSD to trash, data are moved out to other OSDs until this OSD is empty. Then you can remove the OSD before you replace the defective hardware.



Check the box on the left-hand side of the OSD to select the OSD for removal. When moving the OSD to trash, you can find the cluster is recovering the data, and the cluster is in a healthy state. Ceph will reduce the number of PGs of this OSD gradually to zero.















































<input type="checkbox"/>	Rank			Node Name	IP Address	PG	Status	Data SMART	MetaData SMART
<input type="checkbox"/>	osd.2			demo14-1078	↑192.168.1.114	56	Up	  	  
<input type="checkbox"/>	osd.3			demo15-107a	↑192.168.1.115	57	Up	  	  
<input type="checkbox"/>	osd.4			demo16-107c	↑192.168.1.116	68	Up	  	  
<input type="checkbox"/>	osd.6			demo18-1080	↑192.168.1.118	64	Up	  	  
<input type="checkbox"/>	osd.0			demo12-1074	↑192.168.1.112	69	Up	  	  
<input type="checkbox"/>	osd.1			demo13-1076	↑192.168.1.113	65	Up	  	  
<input type="checkbox"/>	osd.5			demo17-107e	↑192.168.1.117	0	Up	  	  

In some cases, you may move the OSD in the recycle bin back to the cluster. You can select the OSD and click “Rejoin From Recycle.” The checkbox will give you a hint for rejoining OSD.

+ Create New OSD
Service Restart
Node Reboot
Remove OSD
Move To Recycle
Rejoin From Recycle


<input type="checkbox"/>	Rank			Node Name	IP Address	PG	Status	Data SMART	MetaData SMART
<input type="checkbox"/>	osd.2			demo14-1078	↑192.168.1.114	56	Up	  	  
<input type="checkbox"/>	osd.3			demo15-107a	↑192.168.1.115	57	Up	  	  
<input type="checkbox"/>	osd.4			demo16-107c	↑192.168.1.116	68	Up	  	  
<input type="checkbox"/>	osd.6			demo18-1080	↑192.168.1.118	64	Up	  	  
<input type="checkbox"/>	osd.0			demo12-1074	↑192.168.1.112	69	Up	  	  
<input type="checkbox"/>	osd.1			demo13-1076	↑192.168.1.113	65	Up	  	  
<input checked="" type="checkbox"/>	osd.5			demo17-107e	↑192.168.1.117	0	Up	  	  

Rejoin OSD(s) ×

The following Nodes Will Be Rejoin :

osd.5

Select Rejoin Chassis*:



Add

Close

Node Rolling Reboot

UVS manager versions after v2.0.14 control the node reboot of MON and OSD on this page. Rolling reboot avoids rebooting multiple nodes while the cluster is in a degraded status, backfilling or recovering. If there is any degrading condition, backfill, and recovery, the UVS manager will skip the reboot. If multiple nodes are selected for reboot while the cluster does not have the above errors, the UVS manager will do a rolling reboot one after another sequentially. The second node re will be held until the first node reboot is complete and back to a healthy condition. This is the same for other nodes waiting for reboot.

+ Create New OSD

Service Restart
Node Reboot
Remove OSD
Move To Recycle
Rejoin From Recycle

<input checked="" type="checkbox"/>	Rank			Node Name	IP Address	PG	Status	Data SMART	MetaData SMART
<input checked="" type="checkbox"/>	osd.2			demo14-1078	↑192.168.1.114	56	Up		
<input checked="" type="checkbox"/>	osd.3			demo15-107a	↑192.168.1.115	57	Up		
<input checked="" type="checkbox"/>	osd.4			demo16-107c	↑192.168.1.116	68	Up		
<input checked="" type="checkbox"/>	osd.6			demo18-1080	↑192.168.1.118	64	Up		
<input checked="" type="checkbox"/>	osd.0			demo12-1074	↑192.168.1.112	69	Up		
<input checked="" type="checkbox"/>	osd.1			demo13-1076	↑192.168.1.113	65	Up		
<input checked="" type="checkbox"/>	osd.5			demo17-107e	↑192.168.1.117	0	Up		

Now we have added all MONs and OSDs to the cluster. We can proceed to add pools, images, etc. to the cluster. However, before that, let us have a quick scroll down to the next page to look at Ceph Dashboard, and see how it looks populated.

UniVirStore Manager
Dashboard
CEPH
Object Storage
Storage
OpenStack
Settings
admin

Dashboard

Click here for Ceph-Mgr Dashboard (Login With Current UserID/Password)

Cluster ID: c5395871-d647-4c3d-906d-dc8ab1171e95

Ceph Cluster Status

✓ Cluster Status:HEALTH_OK

Ceph Cluster MON Status

✓ demo111-1072
MON : HEALTH_OK MGR : Active

✓ demo112-1074
MON : HEALTH_OK MGR : StandBy

✓ demo113-1076
MON : HEALTH_OK MGR : StandBy

Ceph Cluster OSD Status

Total	In	Up	Unhealthy
5	5	5	0

Ceph Cluster Placement Group Status

Storage



Cluster storage utilization

439.94 GB / 18.62 TB (2%)

Write / second

0 bytes

Read / second

0 bytes

Operations / second

0

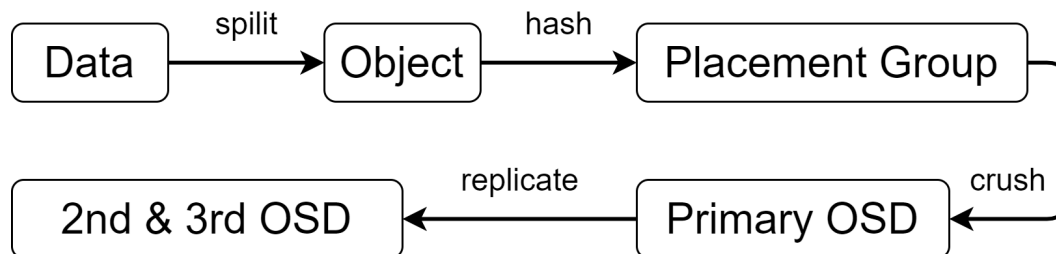
PG Status



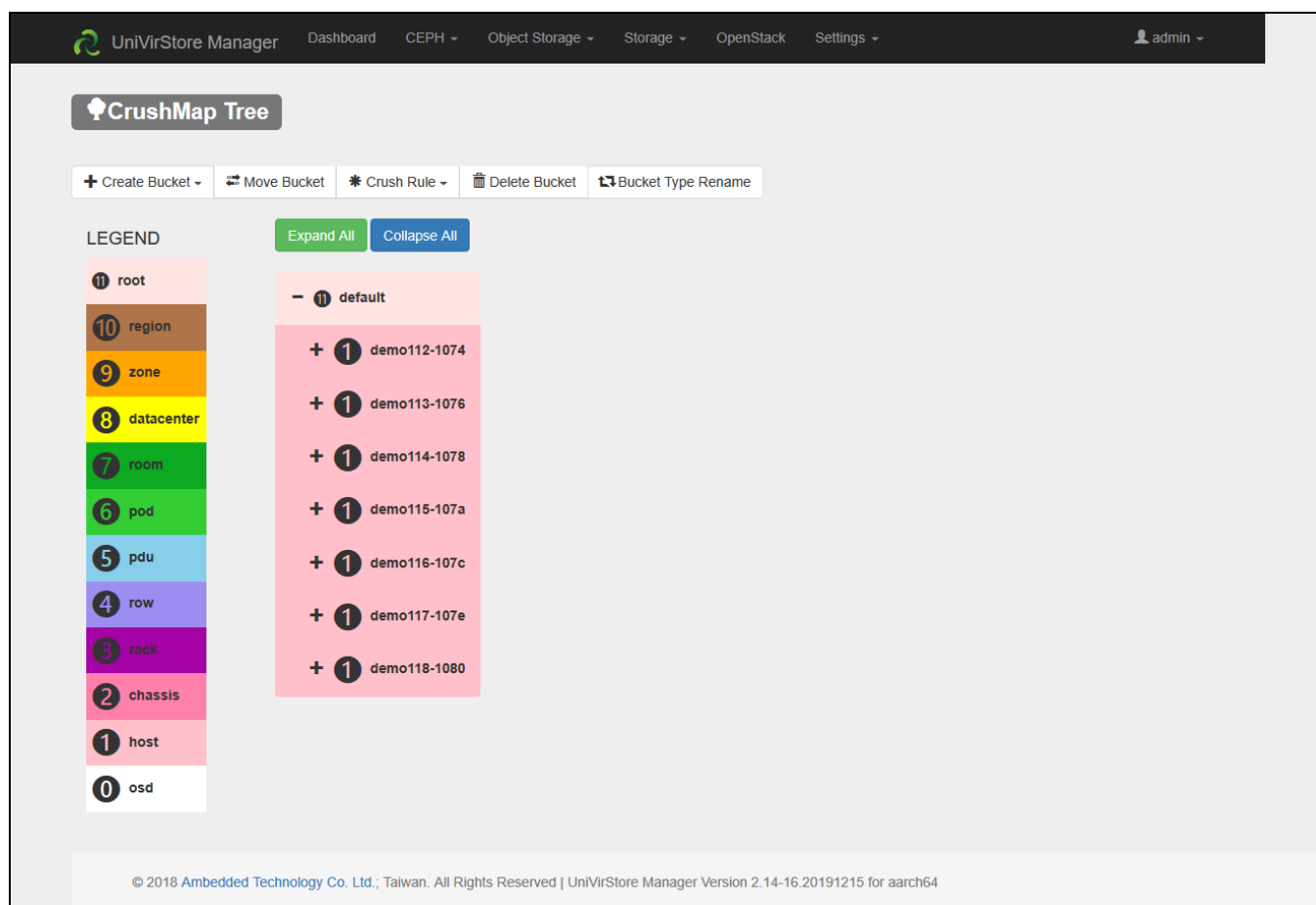
128 placementgroups in cluster

Manipulating CRUSH – Crown of the Ceph Storage Cluster

As CRUSH is the essence of Ceph, this utility to manipulate the CRUSH map is another feather in Ambedded UniVirStor Manager. CRUSH map is an integral part of Ceph, which tells OSD how to place data among the nodes. CRUSH(Controlled Replication Under Scalable Hashing) algorithm defines the working of a Ceph storage cluster. Ceph doesn't have any metadata lookup mechanism. It is the duty of the CRUSH algorithm to calculate the metadata and location of the object.



Viewing the CRUSH Map Tree – Graphic Representation



CrushMap Tree

+ Create Bucket
 Move Bucket
 * Crush Rule
 Delete Bucket
 Bucket Type Rename

LEGEND

- 11 root
- 10 region
- 9 zone
- 8 datacenter
- 7 room
- 6 pod
- 5 pdu
- 4 row
- 3 rack
- 2 chassis
- 1 host
- 0 osd

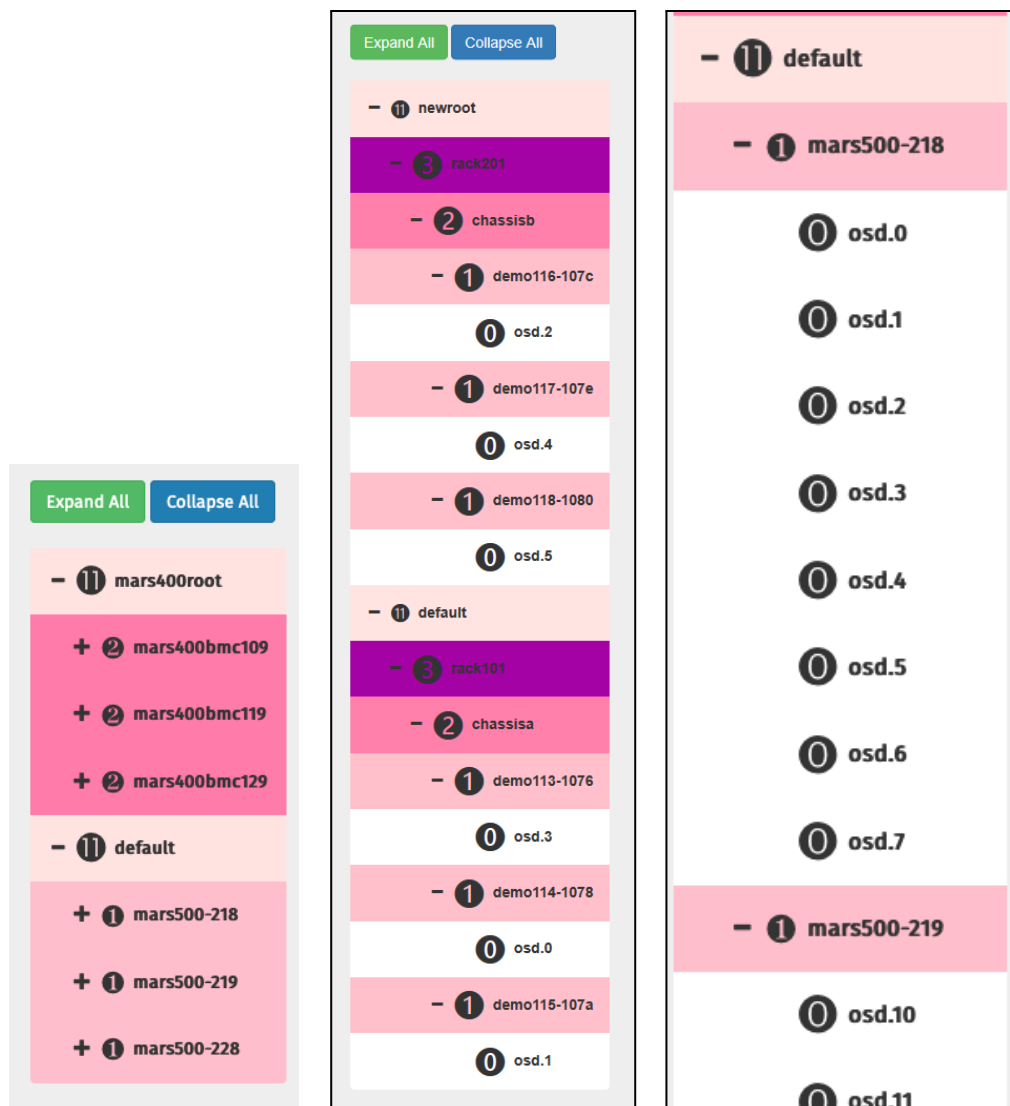
Expand All **Collapse All**

- 11 default
 - + 1 demo112-1074
 - + 1 demo113-1076
 - + 1 demo114-1078
 - + 1 demo115-107a
 - + 1 demo116-107c
 - + 1 demo117-107e
 - + 1 demo118-1080

© 2018 Ambedded Technology Co. Ltd.; Taiwan. All Rights Reserved | UniVirStore Manager Version 2.14-16.20191215 for aarch64

The CRUSH map for your storage cluster describes your device locations within CRUSH hierarchies and a ruleset for each hierarchy that determines how Ceph will store data. Use the “**Expand All**” and “**Collapse All**” buttons to Expand and Collapse the CRUSH tree view as per the requirement.

Expanded View of Mars400 & Mars500



The CRUSH map contains at least one hierarchy of nodes and leaves. The nodes of a hierarchy—called "buckets" in Ceph—are any aggregation of storage locations (e.g., rows, racks, chassis, hosts, etc.) as defined by their type. Each leaf of the hierarchy consists essentially of one of the storage devices in the list of storage devices.

The replication can happen based on various buckets. You can create various buckets (Root, Rack, and Chassis) as per the requirement. To map placement groups to OSDs across failure domains or performance domains, a CRUSH map defines a hierarchical list of bucket types. The purpose of creating a bucket hierarchy is to

segregate the leaf nodes by their failure domains and/or performance domains.

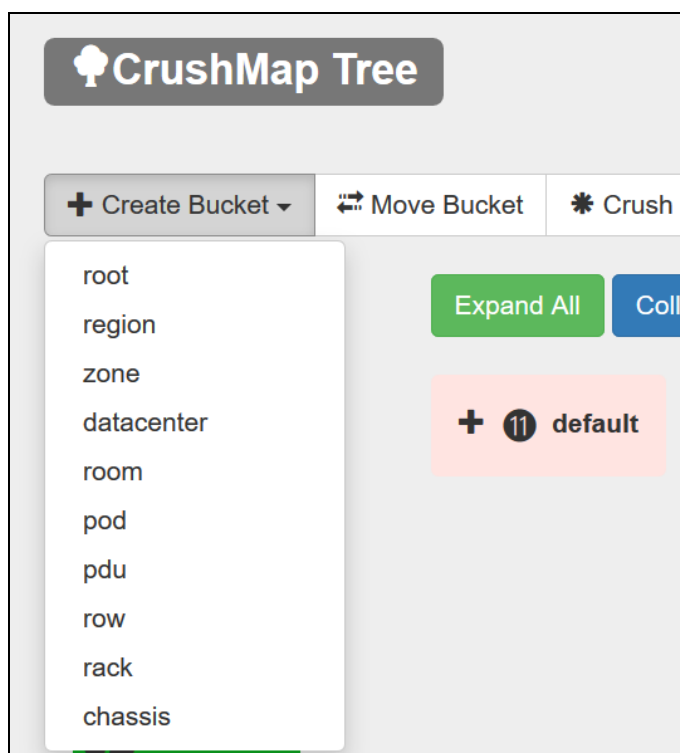
Please plan and create the CRUSH map hierarchy, bucket, and CRUSH rule before you deploy OSD. Changing your CRUSH map hierarchy after deploying OSD may trigger reorganize all placement groups and move data objects to new placements. During the period of moving data objects around the placement groups, it could reduce the client I/O performance.

After you created CRUSH map chassis, you can choose which chassis (Mars400) the new OSD is/are located.

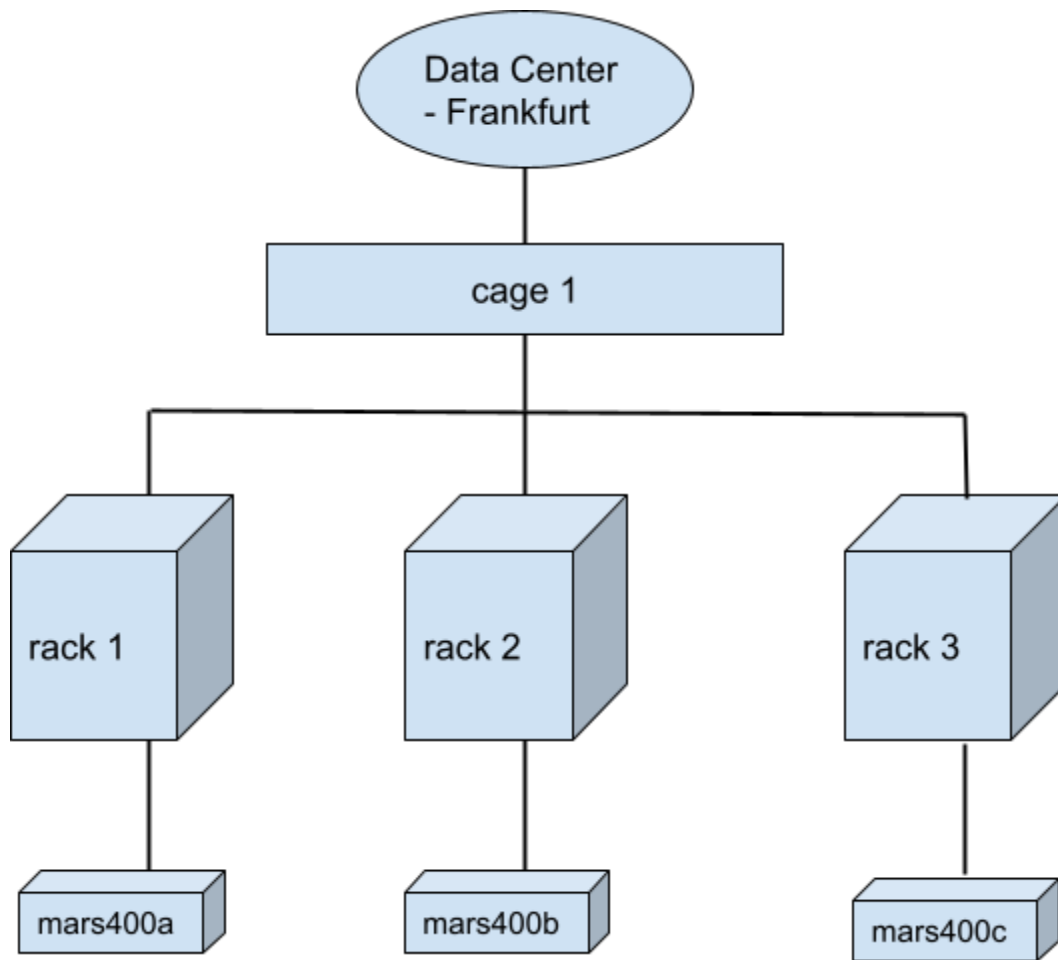
You can rename the default bucket name to what fits your actual naming conventions.

Creating CRUSH Buckets – segregate the leaf nodes

Click “**+Create Bucket**”, you will see the CRUSH bucket types. In the pop-up, please enter one or multiple bucket names that you want to create as the type that you just selected. Please enter the bucket names line by line.



The next step is selecting the parent of these new buckets. The parent of a bucket is any higher level bucket in the hierarchy where the new buckets are located. The bucket types listed in the “**+Create Bucket**” drop-down manual is in the order of high level to lower level buckets in the CRUSH map hierarchy. For example, if you are going to deploy the new Ceph cluster with the hierarchy like the following diagram.



First of all, you have to create frankfurt with the type “datacenter”. Then, you shall create a bucket named cage1 with any bucket type between the level of room and raw, e.g. room and choose frankfurt as its parent. You could rename the type “room” to “cage” for convenience or leave it as it is. The third step is creating rack1, rack2 and rack3 with type “rack” at one go. The last 3 steps are creating chassis mars400a, mars400b, and mars400c one by one, concerning for their parents, rack1, rack2, and rack3.

You can create a rack before you create a chassis or move a chassis to a rack later when a rack is available.

Create chassis Bucket

chassis*:

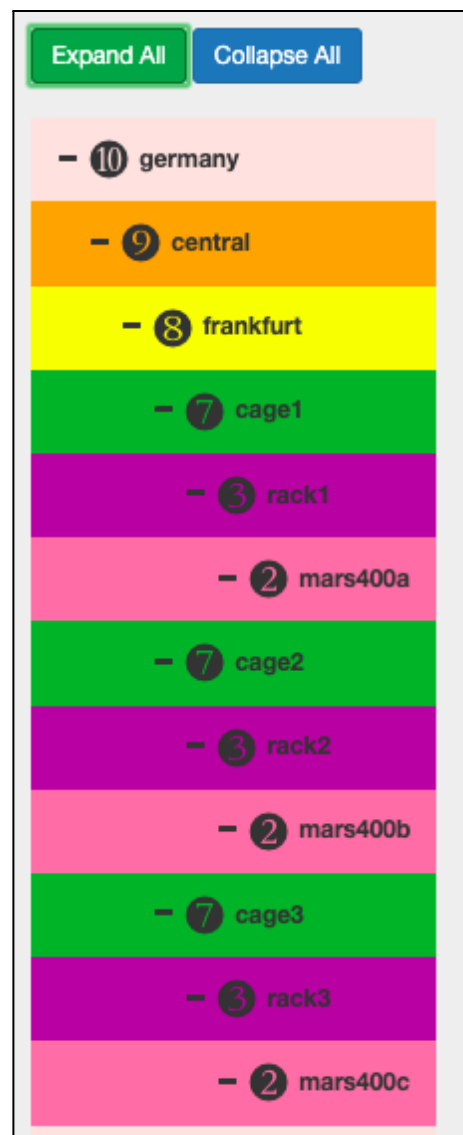
mars400a

Parent* :

rack / rack1

Create

Close



Moving CRUSH bucket

Since UVS v2.14.16, you could move any bucket to the higher level CRUSH bucket you want. If a new OSD is added to the CRUSH map at a suboptimal CRUSH location, or if your cluster topology changes, you may move an OSD in the CRUSH hierarchy to reflect its actual location. These tools also allow moving a host to different buckets to implement replication ruleset.

Warning: Changing your CRUSH map hierarchy after deploying OSD may trigger reorganize all placement groups and move data objects to new placements. During the period of moving data objects around the placement groups, it could reduce the client I/O performance.

Move Nodes

Warning !! Move Node will trigger the data movement and balancing

Select Move Bucket (Source)*:

Type:

chassis

Bucket:

mars400a

Select Parent Bucket (Destination)*:

Type:

root

Bucket:

default

Move

Close

Managing Rulesets – Create & List CRUSH Ruleset

The CRUSH rules define policy about how data is distributed across devices in the hierarchy. We can create new rulesets as well as view the existing CRUSH rulesets defined. We can create a new CRUSH ruleset using the CRUSH Rule create menu option. You need to specify a Rule name to identify the CRUSH ruleset. Identify the root node of the hierarchy you wish to use. Lastly the type of bucket you want to replicate across, (e.g., rack or chassis). Once the CRUSH ruleset is created, you can also list the same using the “**CRUSH Rule**” >> “**List**” CRUSH Rule menu sub-option.

Take the same CRUSH hierarchy in the section of [Create CRUSH Buckets](#) as an example. If you want Ceph to write replicated data to OSDs located in different chassis installed in different racks, you shall select the root as cage1 and the failure domain as the rack. You could also select the device class as HDD or SSD. For pools that use this CRUSH rule, data written to these pools will always be stored in the specified type of storage device.

Create Crush Rule

Rule Name*:

by_rack

Select Parent (Root)*:

Type:

cage

Bucket:

cage1

Select Failure Domain*:


rack

Select Device Class*:

hdd

Create

Close

 **CrushMap Tree**

+ Create Bucket

↔ Move Bucket

* Crush Rule

🗑 Delete Bucket

↕ Bucket Type Rename

LEGEND

Expand

List

Create

List Crush Rule

Search

ID	Name	Root	Failure Domain	Device Class	Action
0	replicated_rule	default	host		<div>Delete</div>
1	test	default	chassis	hdd	<div>Delete</div>
2	by_rack	frkdatacenter	rack	hdd	<div>Delete</div>

Showing 1 to 3 of 3 rows

Close

Rename Bucket Type

The default CRUSH hierarchy is the root, region, zone, datacenter, room, pod, PDU, row, rack, chassis & host. You may want to rename the bucket types and let it fit your environment. Click the Bucket Type Rename to open the window to change the CRUSH bucket type names.

Rename Bucket Type ✕

Current Type Name	Modified Type Name
Type 11 : root:	<input type="text" value="root"/>
Type 10 : region:	<input type="text" value="region"/>
Type 9 : zone:	<input type="text" value="zone"/>
Type 8 : datacenter:	<input type="text" value="datacenter"/>
Type 7 : room:	<input type="text" value="layerd"/>
Type 6 : pod:	<input type="text" value="layerc"/>
Type 5 : pdu:	<input type="text" value="layerb"/>
Type 4 : row:	<input type="text" value="layera"/>
Type 3 : rack:	<input type="text" value="rack"/>
Type 2 : chassis:	<input type="text" value="chassis"/>
Type 1 : host:	<input type="text" value="host"/>
Type 0 : osd:	<input type="text" value="osd"/>

Managing Ceph Pools – Your pool of resources to use

We have all MONs and OSDs added in the Ceph cluster. To use these resources, we need to create Ceph pools. The Ceph storage system supports the notion of “**Pools**”, which are logical partitions for storing objects. Ceph clients retrieve a cluster map from a Ceph monitor and write objects to pools. The pool’s size or the number of replicas, the CRUSH ruleset, and the number of placement groups determine how Ceph will place the data.

Pools set at least the following parameters:

- Ownership/Access to Objects
- The Number of Placement Groups, and
- The CRUSH Ruleset to Use.

Ceph Pools

+ Create Pool

✕ Delete Pool(s)

<input type="checkbox"/>	Pool Name	Type	Size	Application	EC Profile Crush Rule	Placement Groups	Quota	Compression	Mirror Status	Action
<input type="checkbox"/>	device_health_metrics	Replica	3	mgr_devicehealth	replicated_rule	1	Unlimited	Status : Disable	Disabled	<div>Edit</div> <div>+ Cache Tier</div>
<input type="checkbox"/>	iscsi	Erasure	3	rbd	k2m1	16	Unlimited	Status : Disable	Disabled	<div>Edit</div> <div>+ Cache Tier</div>
<input type="checkbox"/>	iscsi.meta	Replica	3	rbd	replicated_rule	32	Unlimited	Status : Disable	Disabled	<div>Edit</div> <div>+ Cache Tier</div>
<input type="checkbox"/>	rbd_pool	Replica	3	rbd	replicated_rule	16	Unlimited	Status : Disable	Disabled	<div>Edit</div> <div>+ Cache Tier</div>

Showing 1 to 4 of 4 rows

We can use this web-page to create, edit and delete the pools. Adding a Cache tier is also possible.

PG Number

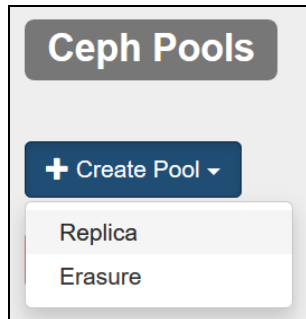
Each pool has a number of placement groups(PG). CRUSH maps PGs to OSDs dynamically. When a Ceph Client stores objects, CRUSH will assign each object to a placement group. Please carefully plan the number before you create a pool. Use the [Ceph PG Calculator](#) to help you plan the PG numbers. To get the best performance, let each OSD store 50 ~ 100 PGs, and make sure the PG number of each pool is the power of two. Please refer to the [ceph document](#) for how to optimize the number of PG. The table below is the recommendations of PG numbers in a cluster. For example, if the maximum number of PGs is 512 + 128, you could create a single pool with 512 PGs or build three pools with PGs 256, 256 & 128.

OSD Numbers	Pool Type	Max PGs in a cluster	Actual PGs per OSD
7	Replicated Pool, Size 3	128 + 64	82
7	Erasure Coded Pool, 4+2	128	110

21	Replicated Pool, Size 3	512 + 128	91
21	Erasure Coded Pool, 4+2	256	73

Creating Ceph Pool

We can create two types of pools based on choice from the drop-down menu.



Create Replicated Pool

This is the default choice when creating a pool, meaning every object is copied on multiple disks as per the replica defined while creating the pool.

Create Replicated Pool

Pools Name*:

rbd

PG*:

128

Replica Size*:

3

Application*:

Block Storage (rbd)

Crush Rule*:

replicated_rule

Quota(GiB)[0 to disable quota]:

2048

Compression Status:

Disable

Compression Required Ratio:

0.85

Create

Close

You need to specify the following while creating a Replicated pool.

- **Pool name:** the name of the replicated pool to create.
- ***PG:** number of placement groups. Each pool has a number of placement groups. CRUSH maps PGs to OSDs dynamically. When a Ceph Client stores objects, CRUSH will map each object to a placement group.
- ***Replica Size:** The Ceph Storage Cluster was designed to store at least two copies of an object (i.e., size = 2), which is the minimum requirement for data safety. For high availability, a Ceph Storage Cluster should store more than three copies of an object. The default pool size is 3.
- ***CRUSH Ruleset:** The CRUSH ruleset to use. CRUSH maps PGs to OSDs dynamically. When a Ceph Client stores objects, CRUSH will map each object to a placement group.
- ***Quota:** this can be used to set a quota on the pool for the maximum number of bytes.
- ***Compression Status:** You can click the “Disable” icon to enable the compression. Or, click again the “Enable” icon to disable the compression.
- ***Compression Required Ratio:** The ratio of the size of the data chunk after compression relative to the original size must be at least this small in order to store the compressed version. If the size of the data chunk is not reduced sufficiently it will not be used and the original (uncompressed) data will be stored. For example, if the compression required ratio is set to .85 then the compressed data must be 85% of the size of the original (or smaller).

Note:

- 1) You can edit those settings marked with an * after the pool is created by using the Edit feature.
- 2) Changing PG, Replica Size and CRUSH Ruleset will cause data to relocate. Make sure the change won't

impact on your storage service.

Erasure Code Profile



An erasure-coded pool stores each object as K+M chunks. It is divided into K data chunks and M coding chunks. The pool is configured to have a size of K+M so that each chunk is stored in an OSD in the acting set. For instance, an erasure-coded pool is created to use five OSDs ($K+M = 5$) and sustain the loss of two of them ($M = 2$).

Prior to creating an Erasure pool, we should have an Erasure profile to be used in place.

Create the Erasure Code Profiles

Since UVS 2.14-16, we choose "CLAY code plugin" as a default erasure coding algorithm. Compared to the previous algorithm jerasure plugin, the "CLAY code" improves the recovery speed and keeps read/write performance. UVS manager creates a default profile "ec4p2_host" with $k=4$ and $m=2$ for you. You can define new profiles to meet your requirements. (UVS manager automatically sets the parameter "d" of CLAY code to $(k+m-1)$ for best practice.

Click the **"Ceph"** >> **"Erasure Profile"** sub-menu option to create a new Erasure profile.

Erasure Profile						
<div>+ New Profile</div> <div>Search  </div>						
Profile Name	Object Chunk (K)	Coding Chunk (M)	Ruleset Failure Domain / ROOT	Device Class	Action	
ec4p2_host	4	2	host / default		 Delete	

Following inputs are required to create a new Erasure Profile:

- **Profile Name:** Name of the Erasure profile to create.
- **Number of Object Chunks (K):** The number of data chunks, i.e. the number of chunks in which the original object is divided.
- **Number of Coding Chunks (M):** The number of coding chunks, i.e. the number of additional chunks computed by the encoding functions.
- **Select Root:** Select the CRUSH root you want to apply to this erasure code profile.
- **Ruleset Failure Domain:** Select the failure domain which will apply to this profile.
- **Select Device Class:** The usable device classes are default, hdd & ssd. You shall specify the device class for this erasure code profile. If you have both SSD and HDD devices in your cluster, Ceph will store data in the specified device class for pools using this profile. If you select "default" as the device class, ceph ignores the device class for storing data.

Create New Erasure Code Profile

Profile Name*:

Number of Object Chunks(K)*:

Number of Coding Chunks(M)*:

Select Root*:

Ruleset Failure Domain*:

Select Device Class*:

****Note: K+M should not exceed total number of Ruleset Failure Domain Type**

Create

Close

Create Erasure Coded pool

Now that we have the Erasure profile, let us proceed to create an Erasure pool.

The following information needs to be entered while creating an Erasure-coded pool. Choosing the right profile is important because it cannot be modified after the pool is created.

- **Pool Name:** the name of the erasure pool to create.
- ***PG:** number of placement groups. Each pool has a number of placement groups. CRUSH maps PGs to OSDs dynamically. When a Ceph Client stores objects, CRUSH will map each object to a placement group.
- **Application:** There are three selections which are File Storage, Block Storage, and Object Storage.
- **Erasure Profile:** the name of the Erasure code profile to use.
- ***Quota:** this can be used to set a quota on the pool for the maximum number of bytes.
- ***Compression Status:** You can click the “Disable” icon to enable the compression. Or, click again the “Enable” icon to disable the compression.
- ***Compression Required Ratio:** The ratio of the size of the data chunk after compression relative to the original size must be at least this small in order to store the compressed version. If the size of the data chunk is not reduced sufficiently it will not be used and the original (uncompressed) data will be stored.

For example, if the compression required ratio is set to .85 then the compressed data must be 85% of the size of the original (or smaller).

Create Erasure Code Pool

Pools Name*:

ec42

PG*:

128

Application*:

Block Storage (rbd)

Erasure Profile*:

ec4p2_host

Metadata Pool Crush Rule*:

replicated_rule

Quota(GiB)[0 to disable quota]:

Compression Status:

Disable

Compression Required Ratio:

0.85

Create

Close

If the erasure code pool is created for RBD image (Block Device) use, UVS will create an extra replicated metadata pool.

Erasure code pool size is the sum of K & M. You have to select the proper failure domain which has a larger size than the required pool size. For example, if you want to create a K=4, M=2 erasure code pool, and you shall have more than six chassis to use for your failure domain. Otherwise, please select the host as its failure domain, defined in the erasure code profile if you have more than 6 OSD nodes.

<input type="checkbox"/>	Pool Name	Type	Size	EC Profile/Crush Rule	Placement Groups	Quota	Action
<input type="checkbox"/>	blockdevice	Erasure	6	ec42_host	512	100 GiB	Edit + Cache Tier
<input type="checkbox"/>	blockdevice.meta	Replica	3	replicated_rule	32	Unlimited	Edit + Cache Tier

Note:

- 1) You can edit those settings marked with an * after the pool is created by using the “**Edit**” feature.
- 2) Changing PG will cause data to relocate. Make sure the change won’t impact on your storage service.

Delete Ceph Pools

We can delete the Ceph pools not being used. Before deleting the pool, please ensure that there is no data in the pool.

Confirm Delete

All Images and its data in the below pools will be deleted.

Pool	Images
rbd	clone1 rbdimage1 rbdimage2

Delete
Cancel

Cache Tiering

A cache tier provides Ceph Clients with better I/O performance for a subset of the data stored in a backing storage tier. Cache tiering involves creating a pool of relatively fast/expensive storage devices (e.g., solid-state drives) configured to act as a cache tier, and a backing pool of either erasure-coded or relatively slower/cheaper devices configured to act as an economical storage tier. The Ceph handles where to place the objects and the tiering agent determines when to flush objects from the cache to the backing storage tier. So the cache tier and the backing storage tier are completely transparent to Ceph clients.

We can set the cache tier for Erasure pools from the "**Ceph**" >> "**Pools**" page. Click the "**+Cache Tier**" option in front of the respective pool to add a replicated pool as a Cache Tier for the same.

Add Cache Tier for Pool rbd

Pools Name*:
rbd

Cache Tier Pool*:
ssdpool

MAX Data in Cache [GiB]*:
100

Create

Close

Once you set any Replicated pool as Caching Tier for an Erasure pool, you get an option to remove Caching Tier, if ever needed in future operations.

<input type="checkbox"/>	rbd	Replica	3	rbd	replicated_rule	128	2 TiB	Status : Disable	Edit	- Cache Tier
<input type="checkbox"/>	ssdpool	Cache Tier of "rbd"	3	rbd	replicated_rule	128	Unlimited	Status : Disable	Edit	

Create RBD Images: Very own Block devices to map and use

Before you can add a block device to a client node, you must create an image for it in the Ceph Storage Cluster. RBD images are simple block devices that are striped over objects and stored in a RADOS object store. The size of the objects the image is striped over must be a power of two. Ceph Block Device images are thin provisioned. They don't actually use any physical storage until you begin saving data to them. However, they do have a maximum capacity that you set with the image size option.

After you map a Ceph block device to a client, it can be formatted with a file system and can be mounted. Or it can be used as a RAW partition. OpenStack components such as Cinder and Glance use block devices. Ceph block storage has many advantages. It supports snapshots and cloning.

You can create a new RBD image, Resize it, Delete an existing image or do Snapshot operations from "**Ceph**" >> "**Images**" sub-menu.

+ Create Image

Search

Pool Name	Image Name	Image Size	Object Size	Mirror Status	Action
nvme	IMAGE_p-test-01	100 GiB	4 MiB	Disabled	<div>Resize</div> <div>Delete</div> <div>Snapshot 0</div> <div>Watchers</div>
	IMAGE_p-test-02	100 GiB	4 MiB	Disabled	<div>Resize</div> <div>Delete</div> <div>Snapshot 0</div> <div>Watchers</div>
	IMAGE_p-test-03	100 GiB	4 MiB	Disabled	<div>Resize</div> <div>Delete</div> <div>Snapshot 0</div> <div>Watchers</div>
	IMAGE_p-test-04	100 GiB	4 MiB	Disabled	<div>Resize</div> <div>Delete</div> <div>Snapshot 0</div> <div>Watchers</div>
	IMAGE_p-test-05	100 GiB	4 MiB	Disabled	<div>Resize</div> <div>Delete</div> <div>Snapshot 0</div> <div>Watchers</div>
	IMAGE_p-test-06	100 GiB	4 MiB	Disabled	<div>Resize</div> <div>Delete</div> <div>Snapshot 0</div> <div>Watchers</div>
	IMAGE_p-test-07	100 GiB	4 MiB	Disabled	<div>Resize</div> <div>Delete</div> <div>Snapshot 0</div> <div>Watchers</div>
	IMAGE_p-test-08	100 GiB	4 MiB	Disabled	<div>Resize</div> <div>Delete</div> <div>Snapshot 0</div> <div>Watchers</div>
	IMAGE_p-test-09	100 GiB	4 MiB	Disabled	<div>Resize</div> <div>Delete</div> <div>Snapshot 0</div> <div>Watchers</div>
	IMAGE_p-test-10	100 GiB	4 MiB	Disabled	<div>Resize</div> <div>Delete</div> <div>Snapshot 0</div> <div>Watchers</div>

Showing 1 to 10 of 21 rows

10

rows per page

<

1

2

3

>

Creating a Ceph RBD Image

Click the **"+Create Image"** button to create a new RBD image.
Enter the following information to create a new RBD block image.

Create RBD Image

Select Pool*:

rbd

Image Name*:

rbdimage1

Image Size(GiB)*:

50

Object Size*:

4 MiB

Create

Close

- **Select Pool:** from the drop-down, select the Pool name in which the created image will be stored. Only the pools that set the application as Block Storage(rbd) are in the drop-down list.
- **Image Name:** the name to identify the RBD block image. This is helpful while mapping the same to the Ceph client as a raw block device.
- **Image Size:** The size of the image to be created.
- **Object Size:** Specifies the object size. The default object size is 4 MB, the smallest is 4K and the maximum is 32M.

Resizing RBD Images

Once you create an RBD Image, it can also be resized to allocate more space for the block device. Click the "**Resize**" button in front of the respective image and you will get a new pop-up, which allows you to enter the new Image Size.

Resize Image rbd/rbdimage1

Pool:

rbd

Image:

rbdimage1

Image Size[GiB]*:

50

Update

Close

Delete RBD Images

Any image, not being used can be deleted by clicking the Delete button in front of that particular image.

You will get a confirmation page with a warning that this process is irreversible and data deleted will not be recovered. Click "**Delete**" to delete the RBD image.

Do you wish to delete image rbd/rbdimage1

Warning! All data will be deleted and is irrecoverable

Cancel

Delete

RBD Snapshots

When we discuss the salient features of the Ambedded UniVirStor Manager, the discussion will not be complete without talking about Snapshots and clones.

A snapshot is a read-only copy of the state of an image at a particular point in time. One of the advanced features of Ceph block devices is that you can create snapshots of the images to retain a history of an image's state. Ceph also supports snapshot layering, which allows you to clone images.

Note: If a snapshot is taken while I/O is still in progress in an image, the snapshot might not get the exact or latest data of the image and the snapshot may have to be cloned to a new image to be mountable. So, we recommend stopping I/O before taking a snapshot of an image. If the image contains a filesystem, the filesystem must be in a consistent state before taking a snapshot. To stop I/O you can use the `fsfreeze` command. See `fsfreeze(8)` man page for more details. For virtual machines, `qemu-guest-agent` can be used to automatically freeze filesystems when creating a snapshot.

To create a snapshot of an RBD image, click the **"Snapshot"** option, which will take you to the Snapshot management web page.

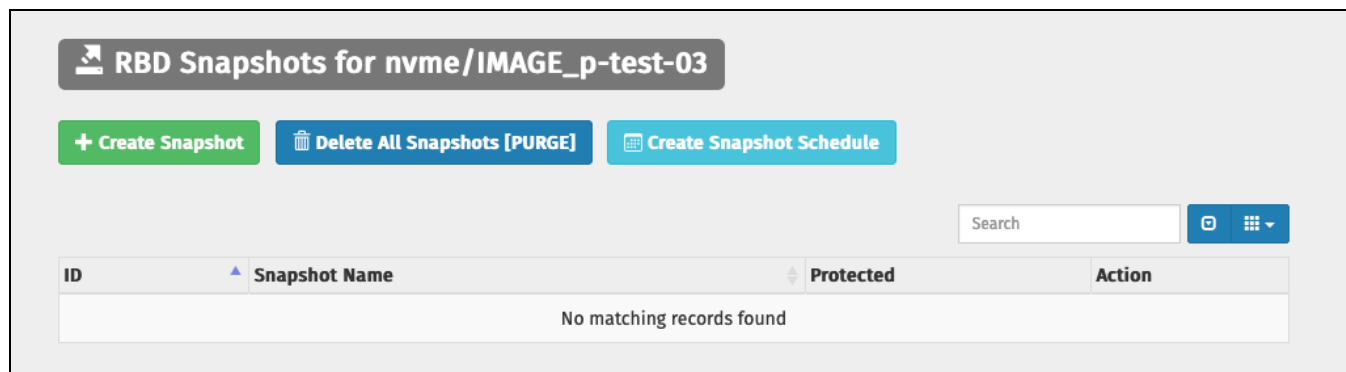
"Ceph" >> "Images" menu page could be used to create a new Snapshot, delete a snapshot, rollback to the earlier saved snapshot state or create a Clone from a snapshot.


RBD Snapshots Management

You can use UVS manager to create a snapshot manually or create a scheduled to do snapshots periodically. The features of snapshot management include:


- Create snapshots manually
- Create snapshot schedules
- Purge all snapshots
- Edit/Delete a schedule
- Rollback a snapshot
- Clone a snapshot

Click **"Snapshot"** beside a RBD image to switch to the snapshot management page.



 **RBD Snapshots for nvme/IMAGE_p-test-03**

+ Create Snapshot 🗑️ Delete All Snapshots [PURGE] 📅 Create Snapshot Schedule



ID	Snapshot Name	Protected	Action
No matching records found			

Create a Snapshot Manually

Click the **+ Create Snapshot** button to create a new snapshot.

Create Snapshot for ×

POOL/IMAGE

nvme/IMAGE_p-test-03

Snapshot Name:

Create

Once the snapshot is created, it will show in the list with other options for the same.

+ Create Snapshot

Delete All Snapshots [PURGE]

Create Snapshot Schedule

Search

ID	Snapshot Name	Protected	Action
14	202301071747	<input type="checkbox"/>	<div>Delete</div> <div>RollBack</div> <div>Clone</div>
15	202301071749	<input type="checkbox"/>	<div>Delete</div> <div>RollBack</div> <div>Clone</div>

Showing 1 to 2 of 2 rows

Create a Snapshot Schedule

Click the **Create Snapshot Schedule** button.

First, you have to give the schedule a name. The snapshot schedule name is used to identify a schedule. It is not related to the snapshot name. UVS manager will give every snapshot a name based on the time the snapshots are taken.

Then please Fill or select the frequency, weekdays, Start/End hour and maximum number of scheduled snapshots.

Create Snapshot Schedule

POOL/IMAGE

nvme/IMAGE_p-test-01

Snapshot Schedule Name

Allowed Characters 'A-Z','a-z','0-9','_','-','_'

every-hour

Snapshot Schedule Frequency

Day : 0

Hour : 1

Minutes : 0

Snapshot Schedule Work Weekday

☐ Sunday
☒ Monday
☒ Tuesday
☒ Wednesday
☒ Thursday
☒ Friday
☐ Saturday

Snapshot Schedule START / END Hour

24-hour clock

START : 8

END : 20

Snapshot Max Number


Max is 999 and 0 is no Limit


10

Create

When snapshot schedule is enabled, UVS manager will keep maximum number of snapshot set plus the protected snapshots.



Edit and Delete a Schedule

You can edit the schedule to change it by clicking the edit  button.

You can also delete schedule by clicking the  button.













































Multiple schedules for a RBD is allowed.

Snapshot Schedule :

	Name	Frequency	Work Weekday	Start/End Hour	Max Number
 	every-hour	1h	1,2,3,4,5	8-20	10

delete schedule
edit schedule




Snapshot protect switch

ID	Snapshot Name	Protected	Action
251	2023-01-08-19-00-08+0800		 Delete  RollBack  Clone
253	2023-01-09-08-00-03+0800		 Delete  RollBack  Clone
257	2023-01-09-09-00-04+0800		 Delete  RollBack  Clone
277	2023-01-09-09-43-41+0800		 Delete  RollBack  Clone
292	2023-01-09-10-00-04+0800		 Delete  RollBack  Clone
340	2023-01-09-11-00-04+0800		 Delete  RollBack  Clone
391	2023-01-09-12-00-05+0800		 Delete  RollBack  Clone
440	2023-01-09-13-00-05+0800		 Delete  RollBack  Clone
491	2023-01-09-14-00-05+0800		 Delete  RollBack  Clone
540	2023-01-09-15-00-06+0800		 Delete  RollBack  Clone
590	2023-01-09-16-00-06+0800		 Delete  RollBack  Clone

Showing 1 to 11 of 11 rows 25 rows per page

Multiple schedules can be set.

Snapshot Schedule :

	Name	Frequency	Work Weekday	Start/End Hour	Max Number
 	every-hour	1h	1,2,3,4,5	8-20	10
 	every-4hour	4h	0,6	0-0	10

Delete Snapshot

Selected snapshot can be deleted by clicking on the **Delete** button against the snapshot. You can also delete all snapshots at a time by using the purge option.

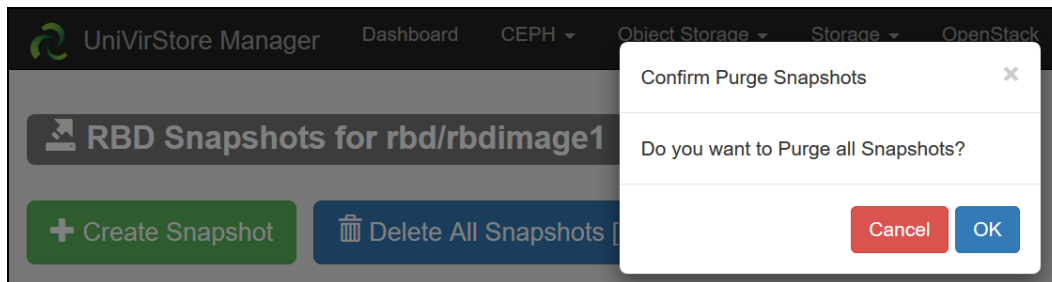
Note: If there is a clone image of a snapshot, you can't delete that snapshot. You have to flatten that cloned image before you delete the snapshot.

Delete Snapshot

Snapshot:

rbd/rbdimage1@snapshot1

Delete



Purge Snapshot

You can also delete all snapshots of an image by using the Purge Snapshot. Click the snapshot icon of an image and click the **“Delete All Snapshots [Purge]”**

Rollback

Rolling back an image to a snapshot means overwriting the current version of the image with data from a snapshot.

To Rollback to an earlier stage of RBD Image, click the **“Rollback”** button in front of Snapshot. A pop-up will come up with a warning. You shall unmount and un-map the image during the process.

Rollback Snapshot

All Data in the image will be rolled back to the snapshot saved state.
Please ensure that image is unmounted and unmapped.

Snapshot:

rbd/rbdimage1@snapshot1


Rollback


The time it takes to execute a rollback increases with the size of the image.



```
Rolling back to snapshot: 94% complete...
Rolling back to snapshot: 95% complete...
Rolling back to snapshot: 96% complete...
Rolling back to snapshot: 97% complete...
Rolling back to snapshot: 98% complete...
Rolling back to snapshot: 99% complete...
Rolling back to snapshot: 100% complete...done.
Exit status : 0
```

Snapshot Clone

This takes us to Clone from a Snapshot. This is also a very important feature of the Ambedded UVS Manager. Ceph supports the ability to create many copy-on-write (COW) clones of a block device snapshot. Snapshot layering enables Ceph block device clients to create images very quickly. A snapshot is read-only, so cloning a snapshot simplifies semantics—making it possible to create clones rapidly. Each cloned image (child) stores a reference to its parent image, which enables the cloned image to open the parent snapshot and read it. A COW clone of a snapshot behaves exactly like any other Ceph block device image. You can read to, write from, clone, and resize cloned images.

 **RBD Clone for rbd/rbdimage1@snapshot1**




Destination Pool	Image Name	Action
No matching records found		

Create a Clone – copy-on-write child


To clone a snapshot, specify you need to specify the parent pool, image and snapshot; and, the child pool and image name.

Create Clone for


RBD Snapshot:


rbd/rbdimage1@snapshot1

Destination Pool:

rbd


Destination Image Name:

clone1



Once cloning is complete, you will be able to see it in both the web-page of Clone and Ceph Image as shown

below.

RBD Clone for rbd/rbdimage1@snapshot1

+ Create Clone

Destination Pool	Image Name	Action
rbd	clone1	Flatten

Flatten a cloned image – Detaching child from a parent

Cloned images retain a reference to the parent snapshot. When you remove the reference from the child clone to the parent snapshot, you effectively “flatten” the image by copying the information from the snapshot to the clone.

To flatten an image, click the “**Flatten**” button against the Clone name.

Flatten Clone

Clone will be detached from its parent snapshot

Clone Image:

rbd/clone1

Flatten

Click on Flatten to start the process. The time it takes to flatten a clone increases with the size of the snapshot.

```

Image flatten: 94% complete...
Image flatten: 95% complete...
Image flatten: 96% complete...
Image flatten: 97% complete...
Image flatten: 98% complete...
Image flatten: 99% complete...
Image flatten: 100% complete...done.
Exit status : 0

```

Once the flattening process is complete, the clone image will be removed from the clone page but you can still find it in the selected pool as an independent image.

Ceph Images

+ Create Image

Search

Pool Name	Image Name	Image Size	Object Size	Action
rbd	clone1	50 GiB	4 MiB	<div>Resize</div> <div>Delete</div> <div>Snapshot</div> <div>Watchers</div>
	rbdimage1	50 GiB	4 MiB	<div>Resize</div> <div>Delete</div> <div>Snapshot</div> <div>Watchers</div>
	rbdimage2	100 GiB	8 MiB	<div>Resize</div> <div>Delete</div> <div>Snapshot</div> <div>Watchers</div>

Since a flattened image contains all the information from the snapshot, a flattened image will take up more storage space than a layered clone.

RBD Mirroring - Asynchronous replicate your block device to another Ceph cluster for disaster protection

The RBD Mirroring service mirrors RBD images asynchronously between two Ceph clusters. Since the Ceph Pacific version 16.2.x the new snapshot based RBD mirroring is stable for production.

Snapshot based mirroring uses periodically scheduled or manually created RBD image mirror-snapshots to replicate crash-consistent RBD images between clusters. The remote cluster will determine any data or metadata updates between two mirror-snapshots and copy the deltas to its local copy of the image. With the help of the RBD fast-diff image feature, updated data blocks can be quickly determined without the need to scan the full RBD image. Since this mode is not as fine-grained as journaling, the complete delta between two snapshots will need to be synced prior to use during a failover scenario. Any partially applied set of deltas will be rolled back at the moment of failover.

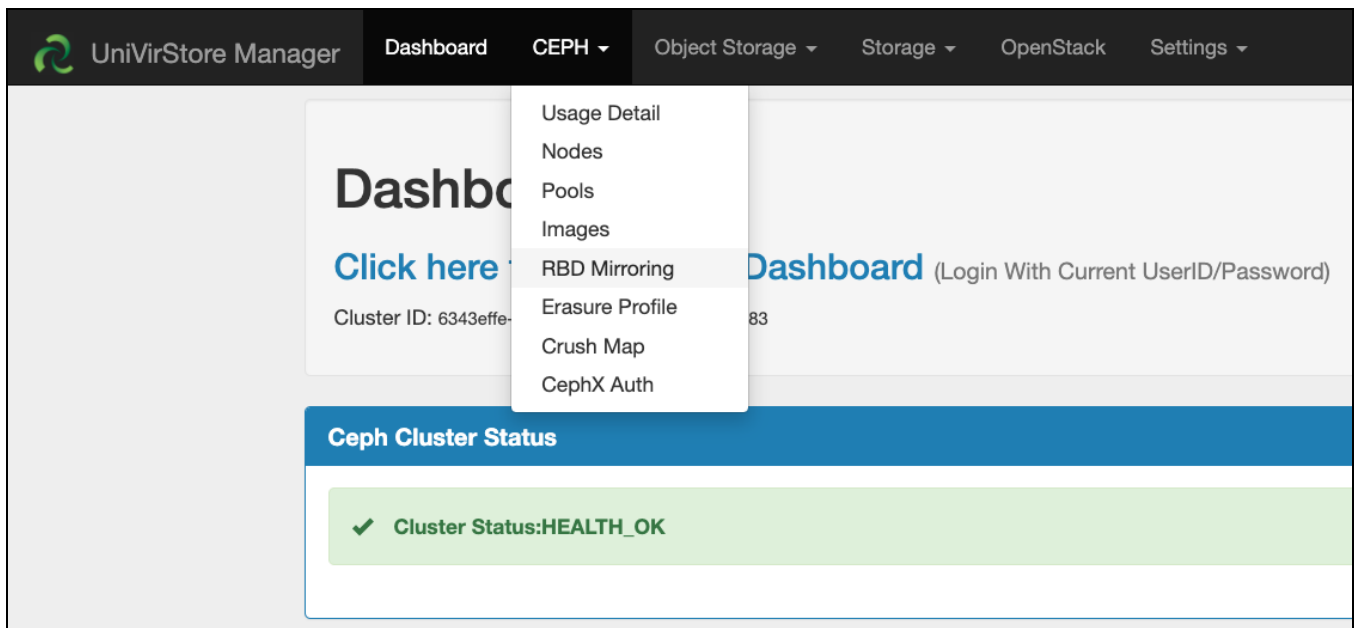
You can mirror all images in specific pools or the particular images from the primary cluster to the secondary cluster. The primary cluster is the cluster where its RBD images are replicated to the cluster on another site. The secondary or non-primary cluster is the cluster that stores the RBD images mirrored from the primary cluster. Please refer to the [Ceph document](#) for details of RBD mirroring.

Before getting started, there are things to keep in mind:

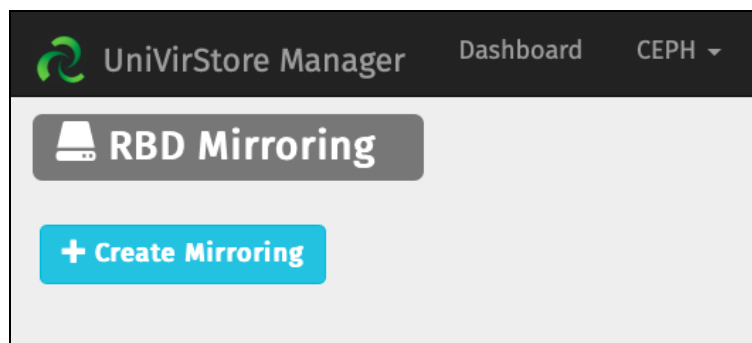
1. Each instance of the rbd-mirror daemon must be able to connect to both the local and remote Ceph clusters simultaneously (i.e. all monitor and OSD hosts). Additionally, the network must have sufficient bandwidth between the two data centers to handle the mirroring workload.
2. The rbd-mirroring service shall run on a monitor node or Linux server. You could use the Ceph monitor node or another server machine as the mirroring server. You can deploy the mirroring server on an independent machine to get better performance. We recommend you to deploy the mirroring server at the secondary site.
3. Since version 2.16.19, UVS manager uses the snapshot based mirroring to take the advantages of snapshot mode mirroring. It supports the kernel RBD and user space RBD. UVS manager provides the configurable automated snapshot schedule to synchronize the differences between snapshots.
4. During the RBD mirroring, the replicated image is operation-protected. Unless you do the promotion, you can't execute any operation on the replicated image.

By using UVS manager, you can deploy and manage the RBD mirroring with following features.

- ❖ Deploy rbd mirror service on Monitor node.
- ❖ Initial bidirectional mirroring between two Ceph clusters.
- ❖ Add multiple mirroring sites
- ❖ Mirror designated RBD images or all RBD images in the pools.
- ❖ Automatically create a pool at the secondary site if the target pool does not exist in the secondary cluster.
- ❖ Remove mirror peers
- ❖ Add/remove snapshot schedules.
- ❖ View snapshot and sync status
- ❖ Promote/Demote pools
- ❖ Destroy Mirroring



Create RBD Mirroring



Click the “**Create Mirroring**” icon. Fill the necessary information in the dialog box.

Create RBD Mirroring

MON IP Of Primary Cluster:* (Primary act the role of Mirror Data Provider)

192.168.1.118

Primary Site Name:*

LosAngeles

MON IP Of Secondary Cluster:* (Secondary act the role of Mirror Data Receiver)

192.168.1.111

Secondary Site Name:*

NewYork

Next

Close Dialog

Role: The primary site is the local site that you are using the UVS manager to create the mirroring. The secondary site is the remote target site you want to mirror the RBD image to.

MON IP of Primary Cluster: This is the monitor node IP address of the host that you are using the UVS manager. UVS manager will deploy a mirroring service on this host. The rbd-mirror daemon is responsible for pulling image updates from the remote peer cluster and applying them to the image within the local cluster.

Primary Site Name: This is the name you give it to the local site. Using the city name as the site name is a good choice. You can use capital letters without space for the site name. We use LosAngeles here as an example.

MON IP of Secondary Cluster: The monitor IP address of a monitor node at the secondary Ceph cluster. The UVS manager will deploy a mirroring service on that host to pull the image updates from the remote peer cluster and apply them to the image within the local cluster.

Secondary Site Name: This is the name you want to give it to the secondary site. We use NewYork as an example here.

Click the **Next** button to proceed with the process.

After creating the RBD Mirroring, the UVS manager will display the main page of RDB mirroring.

Create RBD Mirroring

×

Mirror Configuration for Primary/Secondary are Completed

Primary (LosAngeles)

IP : 192.168.1.118

HostName : node118-7a1d

SiteName : LosAngeles

Secondary(NewYork)

IP : 192.168.1.111

HostName : node111-7a0f

SiteName : NewYork

Go On Mirror Pool/Images Selection

Close Dialog

RBD Mirroring (LosAngeles)

Add Mirror Schedule ~

Destroy Mirroring

Add Site

LosAngeles=>NewYork

Site Data

Site Name	LosAngeles
MON IP	192.168.1.118
MON Hostname	node118-7a1d
Cluster Status	HEALTH_OK
Mirror Daemon(s)	(1258563) 192.168.1.118 [node118-7a1d]

Mirror Status :

Search

Role	Source	Target	Pool Name	Peer(s)	Mode	Image #	Status	Action
------	--------	--------	-----------	---------	------	---------	--------	--------

Mirror Image List :

Search

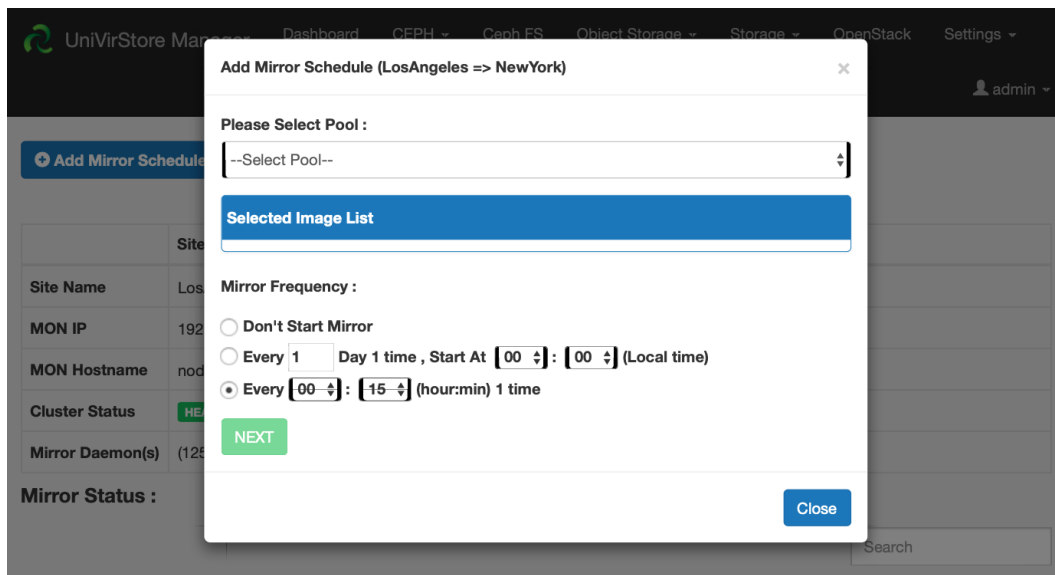
Pool	Image	Schedule	Description	Status
No matching records found				

Add Mirror Schedule

The next step is to add a Mirroring schedule to tell the UVS manager which RBD images you want to enable the mirroring and the period between two snapshots.

After setting the schedule, the UVS manager will immediately take a snapshot of all the specified images. Then, the mirroring service will start to sync data to the remote site. The time for completing the first time sync depends on the size of the RBD images and the network speed between the two clusters.

Click the **Add Mirror Schedule** button at the upper left corner to activate the dialog box.

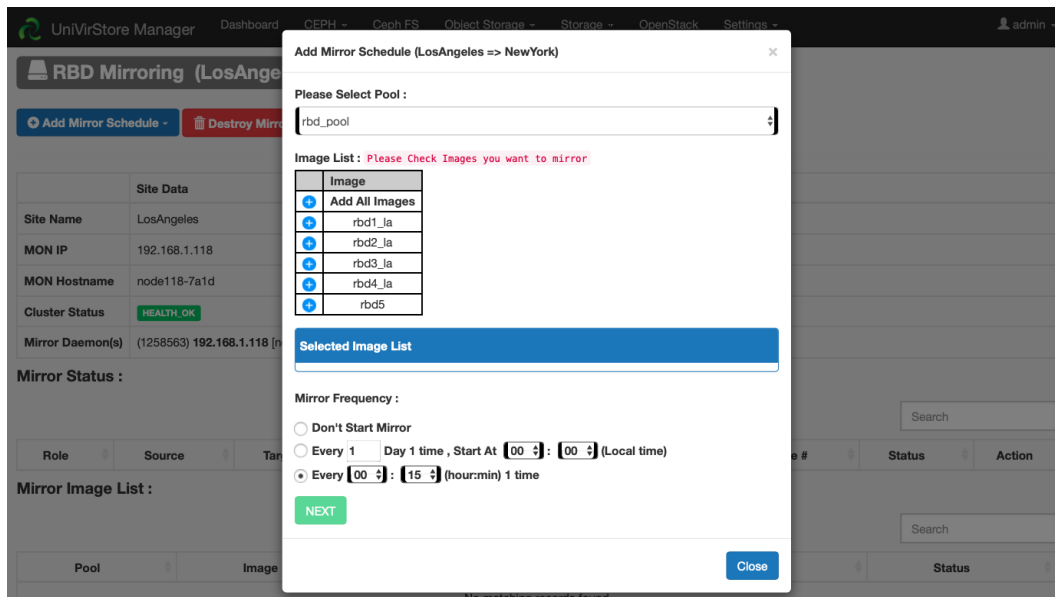




The screenshot shows the 'Add Mirror Schedule' dialog box in the UniVirStore Manager interface. The dialog box is titled 'Add Mirror Schedule (LosAngeles ==> NewYork)' and has a close button (X) in the top right corner. It contains the following elements:

- Please Select Pool :** A dropdown menu with the text '--Select Pool--'.
- Selected Image List**: A blue header for a list of selected images.
- Mirror Frequency :** Three radio button options:
 - ☐ Don't Start Mirror
 - ☐ Every 1 Day 1 time , Start At [00] : [00] (Local time)
 - ☒ Every [00] : [15] (hour:min) 1 time
- NEXT**: A green button at the bottom left.
- Close**: A blue button at the bottom right.

The background shows the UniVirStore Manager dashboard with a sidebar containing 'Add Mirror Schedule' and a table with columns like 'Site Name', 'MON IP', 'MON Hostname', 'Cluster Status', and 'Mirror Daemon(s)'. The 'Cluster Status' is highlighted in green.

After you select a pool, you will see the available images.



Click the  icon beside the Add All Images or the images you want to mirror. You can remove the RBD images from the list by clicking the  icon.

Set the mirror frequency and click the **NEXT** button to proceed to the mirroring schedule setting.

Add Mirror Schedule (LosAngeles => NewYork)

Please Select Pool :

rbd_pool

Image List : Please Check Images you want to mirror

	Image
	Add All Images
	rbd1_la
	rbd2_la
	rbd3_la
	rbd4_la
	rbd5

Selected Image List

rbd_pool/rbd1_la
 rbd_pool/rbd2_la
 rbd_pool/rbd3_la
 rbd_pool/rbd4_la
 rbd_pool/rbd5

Mirror Frequency :

☐ Don't Start Mirror
☐ Every 1 Day 1 time , Start At 00 : 00 (Local time)
☒ Every 00 : 15 (hour:min) 1 time

NEXT

After the UVS manager completes adding the schedule, you will see the Mirror Status. The icon **P** on the Role column states the images of that row are primary images.

RBD Mirroring (LosAngeles)

Add Mirror Schedule ~

Destroy Mirroring

Add Site


	Site Data
Site Name	LosAngeles
MON IP	192.168.1.118
MON Hostname	node118-7a1d
Cluster Status	HEALTH_OK
Mirror Daemon(s)	(1258563) 192.168.1.118 [node118-7a1d]



















Mirror Status :

Search

Role	Source	Target	Pool Name	Peer(s)	Mode	Image #	Status	Action
P	LosAngeles	NewYork	rbd_pool	Active	image	View Images (5)	OK	Remove Mirror Demote

Show Mirror Status Details

Click the [View Images](#) button to view the details of every image. You can remove an image from mirroring by clicking the cross icon .

Mirror Status :								
								<input type="text" value="Search"/>
Role	Source	Target	Pool Name	Peer(s)	Mode	Image #	Status	Action
	LosAngeles	NewYork	rbd_pool		image	View Images (5)		Remove Mirror Demote
Mirror Image List :								
								<input type="text" value="Search"/>
Pool	Image	Schedule	Description				Status	
rbd_pool	rbd1_la	 Sync Interval : 15m Next Mirror Time: 🕒 2021-08-06 10:15:00	Status : replaying bytes_per_second :0 Byte bytes_per_snapshot :0 Byte local_snapshot_timestamp :2021-08-06 09:59:07 remote_snapshot_timestamp :2021-08-06 09:59:07 replay_state :idle				 	
rbd_pool	rbd2_la	 Sync Interval : 15m Next Mirror Time: 🕒 2021-08-06 10:15:00	Status : replaying bytes_per_second :0 Byte bytes_per_snapshot :0 Byte local_snapshot_timestamp :2021-08-06 09:59:10 remote_snapshot_timestamp :2021-08-06 09:59:10 replay_state :idle				 	
rbd_pool	rbd3_la	 Sync Interval : 15m Next Mirror Time: 🕒 2021-08-06 10:15:00	Status : replaying bytes_per_second :0 Byte bytes_per_snapshot :0 Byte local_snapshot_timestamp :2021-08-06 09:59:12 remote_snapshot_timestamp :2021-08-06 09:59:12 replay_state :idle				 	
rbd_pool	rbd4_la	 Sync Interval : 15m Next Mirror Time: 🕒 2021-08-06 10:15:00	Status : replaying bytes_per_second :0 Byte bytes_per_snapshot :0 Byte local_snapshot_timestamp :2021-08-06 09:59:14 remote_snapshot_timestamp :2021-08-06 09:59:14 replay_state :idle				 	
rbd_pool	rbd5	 Sync Interval : 15m Next Mirror Time: 🕒 2021-08-06 10:15:00	Status : replaying bytes_per_second :0 Byte bytes_per_snapshot :0 Byte local_snapshot_timestamp :2021-08-06 09:59:16 remote_snapshot_timestamp :2021-08-06 09:59:16 replay_state :idle				 	

Add Additional Images for Mirroring

To add images to the mirroring, you can click the [Add Mirror Schedule](#) button and follow the steps you did for the first pool.

RBD Mirroring (LosAngeles)

Add Mirror Schedule

Destroy Mirroring

Add Site

Site Data	
Site Name	LosAngeles
MON IP	192.168.1.118
MON Hostname	node118-7a1d
Cluster Status	HEALTH_OK
Mirror Daemon(s)	(1258563) 192.168.1.118 [node118-7a1d]

Mirror Status :

Search

Role	Source	Target	Pool Name	Peer(s)	Mode	Image #	Status	Action
P	LosAngeles	NewYork	iscsi.meta	Active	image	View Images (4)	OK	Remove Mirror Demote
P	LosAngeles	NewYork	rbd_pool	Active	image	View Images (5)	OK	Remove Mirror Demote

Mirror Image List :

You can also view the mirror status from the remote cluster (New York). The icon **S** implies these images are non-primary images mirrored from the remote site (Los Angeles). You can view the image status details by clicking the [View Images](#) button. But, you can not make any changes to the non-primary images.

RBD Mirroring (NewYork)

Add Mirror Schedule

Destroy Mirroring

Add Site

Site Data	
Site Name	NewYork
MON IP	192.168.1.111
MON Hostname	node111-7a0f
Cluster Status	HEALTH_OK
Mirror Daemon(s)	(584109) 192.168.1.111 [node111-7a0f]

Mirror Status :

Search


Role	Source	Target	Pool Name	Peer(s)	Mode	Image #	Status	Action
S	LosAngeles	NewYork	iscsi.meta	Active	image	View Images (4)	OK	Promote
S	LosAngeles	NewYork	rbd_pool	Active	image	View Images (5)	OK	Promote

Mirror Image List :

Add Mirroring to Mirror Images from the Other Site

After creating the mirroring to replicate images from site A to site B, you can also set up mirroring to mirror images from site B to site A.

Use the UVS manager at site B to set up the mirroring from site B to site A.


RBD Mirroring (NewYork)

[Add Mirror Schedule](#)
[Destroy Mirroring](#)
[Add Site](#)

Site Data

Site Name

NewYork

MON IP

192.168.1.111

MON Hostname

node111-7a0f

Cluster Status

HEALTH_OK

Mirror Daemon(s)


(584109) 192.168.1.111 [node111-7a0f]

Mirror Status :

Role	Source	Target	Pool Name	Peer(s)	Mode	Image #	Status	Action
S	LosAngeles	NewYork	iscsi.meta	Active	image	View Images (4)	OK	Promote
S	LosAngeles	NewYork	rbd_pool	Active	image	View Images (5)	OK	Promote
P	NewYork	LosAngeles	NY_Pool1	Active	image	View Images (2)	OK	Remove Mirror Demote

Remove a Mirror

You can remove a mirror of all images in a pool by clicking the **Remove Mirror** button. After removing the mirror, the rbd mirror service will stop mirroring all images in the pool. The remote cluster will remain the images. These images' mirror status on the RBD page will be changed to **Disabled**.


RBD Mirroring (LosAngeles)

[Add Mirror Schedule](#)
[Destroy Mirroring](#)
[Add Site](#)

Site Data

Site Name

LosAngeles

MON IP

192.168.1.118

MON Hostname

node118-7a1d

Cluster Status

HEALTH_OK

Mirror Daemon(s)

(1258563) 192.168.1.118 [node118-7a1d]

Mirror Status :

Role	Source	Target	Pool Name	Peer(s)	Mode	Image #	Status	Action
P	LosAngeles	NewYork	iscsi.meta	Active	image	View Images (4)	OK	Remove Mirror Demote
P	LosAngeles	NewYork	rbd_pool	Active	image	View Images (5)	OK	Remove Mirror Demote
S	NewYork	LosAngeles	NY_Pool1	Active	image	View Images (2)	OK	Promote

Click remove Mirror to stop the mirroring

Failover & Failback - Promotion, Demotion

The RBD images which are mirroring to a remote site are defined as primary images. Primary images can be mounted/accessed by their clients. The non-primary images are those images mirrored from a remote site. You can find the Mirror status of each image on the UVS manager Ceph → Images page.

Ceph Images
[+ Create Image](#)

Pool Name	Image Name	Image Size	Object Size	Mirror Status	Action
iscsi.meta	blockimage	100 GiB	4 MiB	Primary (Mirror to NewYork)	↻ Resize 🗑 Delete 📷 Snapshot 👁 Watchers
	fileimage	50 GiB	4 MiB	Primary (Mirror to NewYork)	↻ Resize 🗑 Delete 📷 Snapshot 👁 Watchers
	lun1	100 GiB	4 MiB	Primary (Mirror to NewYork)	↻ Resize 🗑 Delete 📷 Snapshot 👁 Watchers
	lun2	10 GiB	4 MiB	Primary (Mirror to NewYork)	↻ Resize 🗑 Delete 📷 Snapshot 👁 Watchers
rbd_pool	rbd1_la	10 GiB	4 MiB	Primary (Mirror to NewYork)	↻ Resize 🗑 Delete 📷 Snapshot 👁 Watchers
	rbd2_la	15 GiB	4 MiB	Primary (Mirror to NewYork)	↻ Resize 🗑 Delete 📷 Snapshot 👁 Watchers
	rbd3_la	50 GiB	4 MiB	Primary (Mirror to NewYork)	↻ Resize 🗑 Delete 📷 Snapshot 👁 Watchers
	rbd4_la	20 GiB	4 MiB	Primary (Mirror to NewYork)	↻ Resize 🗑 Delete 📷 Snapshot 👁 Watchers
	rbd5	30 GiB	4 MiB	Primary (Mirror to NewYork)	↻ Resize 🗑 Delete 📷 Snapshot 👁 Watchers
	rbd6_la	10 GiB	4 MiB	Disabled	↻ Resize 🗑 Delete 📷 Snapshot 👁 Watchers
NY_Pool1	image1_NY	10 GiB	4 MiB	Non Primary (Mirrored From NewYork)	↻ Resize 🗑 Delete 📷 Snapshot 👁 Watchers
	image2_NY	100 GiB	4 MiB	Non Primary (Mirrored From NewYork)	↻ Resize 🗑 Delete 📷 Snapshot 👁 Watchers

After setting up the rbd-mirroring, you can find a “Promote” or “Demote” button at the Action column at the right side of each pool.

Role	Source	Target	Pool Name	Peer(s)	Mode	Image #	Status	Action
P	LosAngeles	NewYork	iscsi.meta	Active	image	View Images (4)	OK	Remove Mirror Demote
P	LosAngeles	NewYork	rbd_pool	Active	image	View Images (5)	OK	Remove Mirror Demote
S	NewYork	LosAngeles	NY_Pool1	Active	image	View Images (2)	OK	Promote

In a failover scenario, suppose the local cluster fails, and you cannot access the local images. Before the clients can mount/access the remote backup images, you have to promote the remote site non-primary images to primary images. To promote the images, please click the [Promote](#) icon. Then, clients can mount/access the images from the remote site. After the failed local cluster returns to normal, you have to sync the accumulated changes to the local RBD images before clients can switch to the local storage. Follow the steps below for the failback.

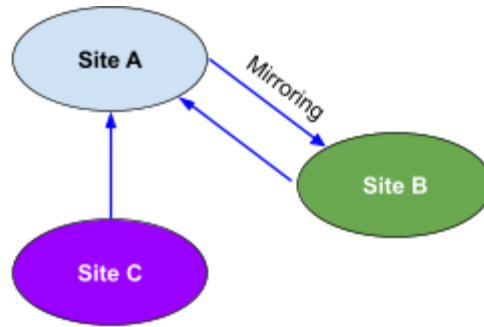
1. Open the local cluster UVS manager. Demote the local images by clicking the [Demote](#) icon.
2. After the demotion is complete, the mirroring service will start to copy the differences from the remote site to the local site at the next snapshot time .
3. Wait for the synchronization to be complete. The total time of completing the synchronization depends on the amount of accumulated differences during the failover.
4. Stop the client's IO (e.g., unmount images or turn off the VMs).
5. Promote the local images and demote the remote images.
6. Wait for the completion of image promotion and demotion.
7. The image mirror status display on the UVS manager RBD Mirroring page will show [OK](#) when the mirroring direction is successfully changed.
8. Clients now can switch to access the local images.

Add More Sites for RBD Mirroring

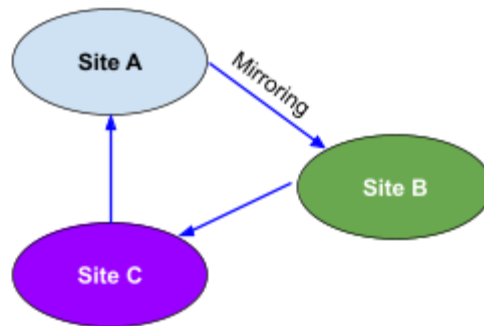
If you have more than two sites, you can use UVS manager to add the third or more sites to mirror their RBD images to other sites. The prerequisite is that you have set up the mirroring for two sites. For example, if you want to add the third site and back up the local images to one of the other sites. Follow the steps below to add a new site for mirroring.

1. Open the UVS manager of the site you want to mirror to.
2. Go to the RBD mirroring page and click [+ Add Site](#) to set up the mirroring.
3. Enter the IP address and the site name of the third site.
4. Wait for the UVS manager to complete the mirroring setup.
5. Open the third site's UVS manager and navigate to the RBD mirroring page.
6. Click the [Add Mirror Schedule](#) button and select the site you want to mirror your images to.
7. Select the pool/images you want to add to the mirroring.
8. Select the snapshot schedule and click the [NEXT](#) button to proceed to the mirroring schedule setting.

Option 1



Option 2



Add RBD Mirroring Site

MON IP :*


192.168.1.138

Site Name:*

Houston

Submit

CloseDialog


RBD Mirroring (Houston)


[+ Add Mirror Schedule](#)
[Destroy Mirroring](#)
[+ Add Site](#)

Site Data	
Site Name	Houston
MON IP	192.168.1.138
MON Hostname	node138-103c
Cluster Status	HEALTH_OK
Mirror Daemon(s)	(24089) 192.168.1.138 [node138-103c]

Mirror Status :

Search

Role	Source	Target	Pool Name	Peer(s)	Mode	Image #	Status	Action
P	Houston	NewYork	Houston_poo1	Active	image	View Images (2)	OK	Remove Mirror Demote


RBD Mirroring (NewYork)

[+ Add Mirror Schedule](#)
[Destroy Mirroring](#)
[+ Add Site](#)

Site Data	
Site Name	NewYork
MON IP	192.168.1.111
MON Hostname	node111-7a0f
Cluster Status	HEALTH_OK
Mirror Daemon(s)	(1154108) 192.168.1.111 [node111-7a0f]

Mirror Status :

Search

Role	Source	Target	Pool Name	Peer(s)	Mode	Image #	Status	Action
S	LosAngeles	NewYork	iscsi.meta	Active	image	View Images (4)	OK	Promote
S	LosAngeles	NewYork	rbid_pool	Active	image	View Images (5)	OK	Promote
P	NewYork	LosAngeles	NY_Pool1	Active	image	View Images (2)	OK	Remove Mirror Demote
S	Houston	NewYork	Houston_poo1	Active	image	View Images (2)	OK	Promote

Destroy Mirroring

If you want to disable all of the mirroring peer relationships, you can use the Destroy Mirroring feature to remove all mirroring peers. It will not delete the mirrored non-primary pools and images. The mirror status of all images will become disabled.

You can re-establish the mirroring after the mirroring is destroyed.

CephX Authentication

Ceph provides two authentication modes:

- No authentication: Any user can access data without authentication.
- Cephx: Ceph user authentication, similar to Kerberos.

UVS manager enables the CephX by default.

Note: If you disable authentication, you are at risk of a man-in-the-middle attack altering your client/server messages, which could lead to disastrous security effects. UVS manager enables CephX authentication as the default setting.

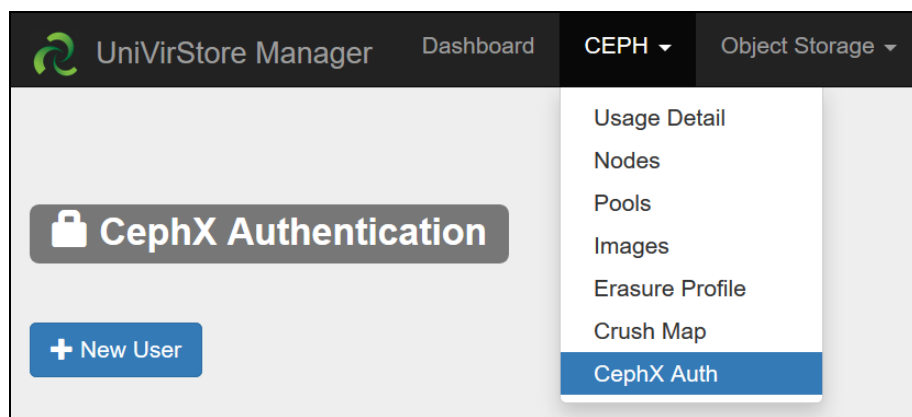
When the Cephx is enabled, Ceph looks for the keyring in the default search path, which is stored in `/etc/ceph/keyring`. You can override this location by adding a keyring option in the `[global]` section of `/etc/ceph/ceph.conf` configuration file.

Cephx uses shared secret keys for authentication, meaning both the client and the monitor cluster have a copy of the client's secret key. The authentication protocol is such that both parties can prove each other. They have a copy of the key without actually revealing it. This protocol provides mutual authentication, which means the cluster is sure the user possesses the secret key, and the user is sure that the cluster has a copy of the secret key.

For production environments, you should create users and assign pool access to the users. Ambedded UniVirStor Manager helps you create Ceph users and bind them to desired pools with permissions. That can allow a particular user to make changes, whereas restricting another user to be read-only access.

Click the "**CephX**" Auth option under CEPH main menu option. You will get to the CephX Auth web-page, which allows you to create CephX User, its associated keyring, bind to pool, and assign permissions to it.

Create CephX User



Click the "**+New User**" button to create a new CephX user.
Click "**Create**" to add the user.

Create User

User Name*:

client.x.

Create

User will be created and a keyring will be created with default permissions.

CephX Authentication

+ New User

Search

User Name	Key	Capabilities	Action
client.x.ambedded	AQDHNs5eFAYnJxAAPwpFnUgllbv0BjNaPalcNw==		<div>Delete</div> <div>Edit</div> <div>Download Key</div>

Once you have edited the capabilities as per requirement, you shall go ahead to edit his capability before download the keys and share them with the Ceph client nodes.

Edit CephX User

Click "**Edit**" to edit the user capabilities. The Capability field lets you edit the user capability of OSD to a specified pool.

Click the "**Allow Capabilities MDS**" for Ceph Filesystem users.

Some Examples:

allow rw pool=rbd ; allow the user to read and write to the rbd pool named rbd
allow * ; allow capability for administrator

Click "**Update**" to save the changes. (NOTE: Add capabilities one per line only)

Edit User ambedded

Capabilities [One Per Line]:

allow rwx pool=rbd

Allow MDS Capabilities: ☒

Update

After updating the user's capabilities, the user's keyring is ready for use by the client. For a client to access the Ceph pool using the created user, the client should have the User Keyring as well as the Ceph configuration file.

CephX Authentication

+ New User

Search

User Name	Key	Capabilities	Action
client.x.ambedded	AQDHNs5eFAYnJxAAPwpFnUgllbv0BjNaPalcNw==	allow rwx pool=rbd Allow MDS Capabilities	Delete Edit Download Key

Ambedded UVS Manager assists in downloading the created user Keyring and Ceph Configuration file by clicking the **"Download Key"** button against the required CephX user. The full name of the key is like client.x.\$username, and the id to the key is x.\$username. The commands below are the example for listing rbd pool.

```
# rbd ls -p rbd -c /root/ceph.conf -k /root/ceph.client.x.ambedded.keyring --name client.x.ambedded
# rbd ls -p rbd -c /root/ceph.conf -k /root/ceph.client.x.ambedded.keyring --id x.ambedded
# rbd ls -p rbd -c /root/ceph.conf -k /root/ceph.client.x.ambedded.keyring --user x.ambedded
```

If the user is not required anymore, you can delete the user by clicking the **"Delete"** button.

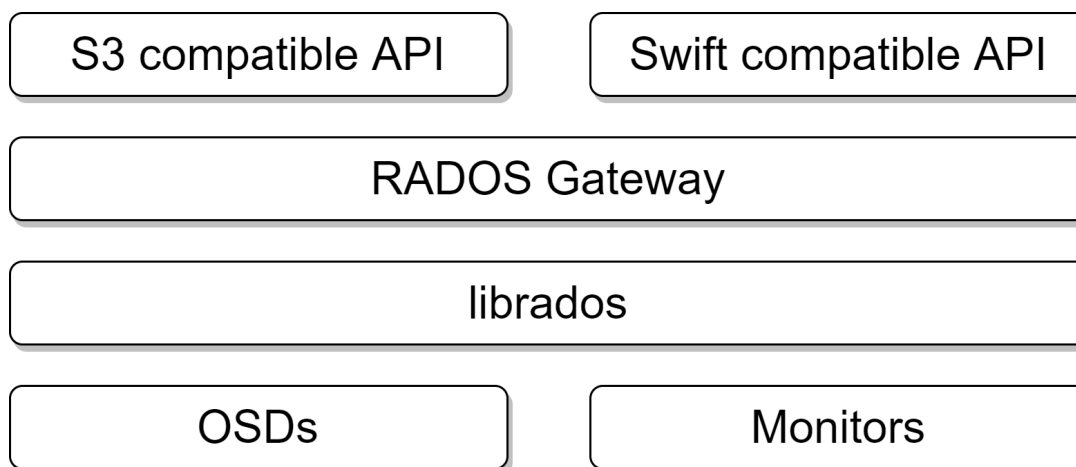
Delete User ambedded

Do you wish to delete

Delete

Ceph Object Storage

Ceph object gateway provides object storage to the Ceph cluster. Ceph object gateway is also known as Rados gateway. Rados gateway acts like a proxy that converts HTTP requests to RADOS requests and vice versa. It provides OpenStack Swift and Amazon S3 compatible object storage. The rados gateway daemon is used to interact with librados library and librados. The Ceph object store supports three interfaces. It provides an Amazon S3 compatible interface, OpenStack Swift compatible interface and an admin API which provides an HTTP restful API to access the Ceph cluster.



RADOS Gateway Management

Ceph Object Storage uses the Ceph Object Gateway daemon (radosgw) for interacting with a Ceph Storage Cluster. Since it provides interfaces compatible with OpenStack Swift and Amazon S3, the Ceph Object Gateway has its own user management.

UVS manager supports deploying RGW on Mars500 hist or external Arm64 and x86 VM/hosts.

The hardware recommendations for x86 server:

1. CPU: 4 cores or more
2. RAM: 32 GB
3. Network: 10Gbps

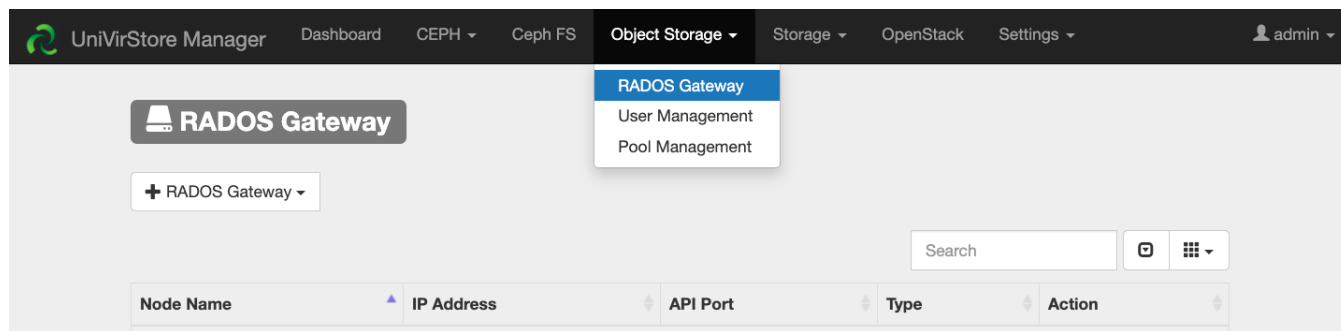
Prepare Software on External Server Nodes

UVS manager supports deploy RGW on Arm64 or x86 servers to get better performance. You have to install necessary software packages and make proper configurations before you can use the UVS manager to deploy and manage your external RADOS gateways. Please follow [this article](#) or [Appendix 2](#) to have the prerequisites ready for RGW. You don't need to install extra software on Mars 500 to deploy RGW on Mars500.

From the Ceph Pacific release onward, Ceph has to pull docker images with specific SHA. UVS deploy Ceph

services in a private network environment that needs a local registry server. So, UVS has a local registry server for Arm64 Docker images. Deploying RGW on external Arm64 servers can use the UVS local registry server. But, for deploying RGW on external x86 servers, please contact Ambedded Helpdesk to set up a local multi architecture registry server.

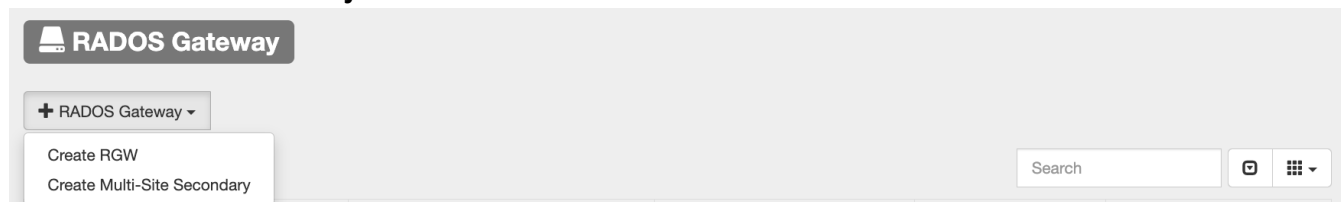
Click on “**Object Storage**” >> “**RADOS Gateway**” option to configure an RGW for the Ceph cluster.



New deployed RGWs are multi-site ready. You can use it for standalone usage or deploy another Ceph cluster in the future when you want to make the object storage support multi-site usage. UVS manager supports creating multi-site ready RADOS Gateways (RGW) on Mars nodes or external Arm64 and x86 servers.

Create RADOS Gateway

Click the “**+RADOS Gateway**” >> “**Create RGW**”



Select the node you want to make as RADOS Gateway and select a CRUSH ruleset for object placement strategy. Configure a PG number for the data pool. UVS has three options for SSL encryption, None, self-signed or uses your own certification keys.

If you select the **Internal**, you can select a monitor node to colocate the RGW from the drop-down list.

Create Master/Standalone Rados Gateway

Select Node*: ☒ Internal ☐ External

✓ mars500-218/192.168.3.218
mars500-219/192.168.3.219
mars500-228/192.168.3.228

Creating RADOS Gateways on external servers

If you want to create the RADOS gateways on external servers, choose **External** and provide it's IP address and root password.

Create Master/Standalone Rados Gateway

Select Node*: ☐ Internal ☒ External

External RGW IP Address*:

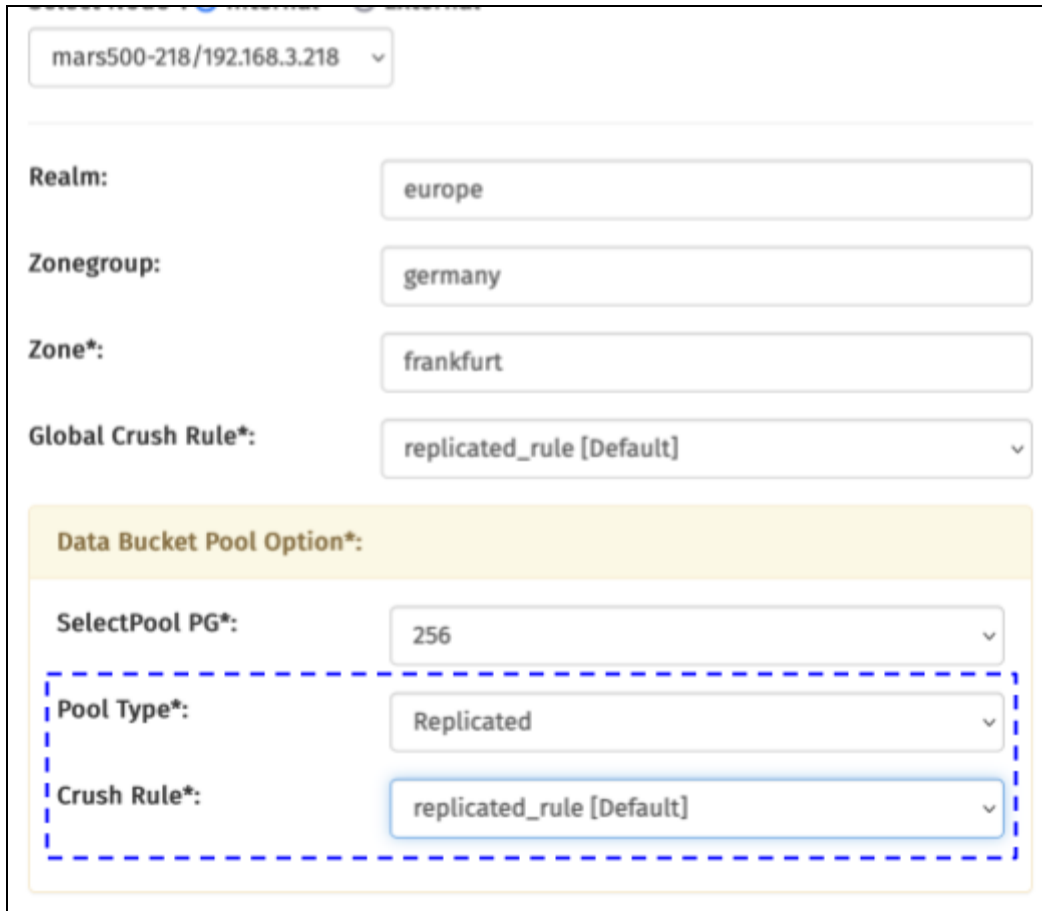
External RGW Root Password*:

Configure RGW Pools

Ceph RGW need RADOS pools for different purpose. The major pool to store object data is the bucket.data pool. UVS manager allows you to use a replicate pool or erasure code data pool. Please refer to the following picture.

- First, you shall fill the name of realm, zone group and zone. These names are for supporting multi site operation. Please refer to the ceph document for [Multi-Site operation details](#). If you are going to have single-site for now, you stail have to fill the realm, zone group and one names for this set up. This also allows you to enable the multi-site support in the future.
- The **Global CRUSH Rule** allows you to select the replica 3 CRUSH rule for all pools exclude the bucket.data as the bucket.data pool will be configured seperately. The UVS manager will set the proper number of PG for you.
- For the bucket data pool that majorly used for storing objects, you have choices to use replica or erasure code and give it a proper number of PG.

Please refer following picture of using replica and erasure code pool for bucket data.



Configuration interface showing options for a data bucket pool:

- Realm: europe
- Zonegroup: germany
- Zone*: frankfurt
- Global Crush Rule*: replicated_rule [Default]

Data Bucket Pool Option*:

- SelectPool PG*: 256
- Pool Type*: Replicated
- Crush Rule*: replicated_rule [Default]

Realm:	<input type="text" value="europe"/>
Zonegroup:	<input type="text" value="germany"/>
Zone*:	<input type="text" value="frankfurt"/>
Global Crush Rule*:	<input type="text" value="replicated_rule [Default]"/>
Data Bucket Pool Option*:	
SelectPool PG*:	<input type="text" value="256"/>
Pool Type*:	<input type="text" value="Erasure"/>
Erasure Code Profile*:	<input type="text" value="ec4p2_host"/>

SSL Encryption Options

You have three options for the SSL encryption:

1. NON SSL: http only.
2. Self-Signed SSL: The UVS manager will use self-signed SSL certificate
3. SSL with upload certificate: You can upload your own certificate and SSL key.

Multiple RGW instances

If you select Non-SSL encryption for your RADOS Gateway, you have the option to deploy multiple RADOS Gateway services in a host with predefined ports by the UVS manager. If you are using SSL encryption for the RGW, please contact Ambedded support team for the support.

SSL Encryption Option*:

☒ No SSL Encryption
☐ Self Signed Certification
☐ Upload CA & SSL Key

RGW Instances*:

SSL Encryption Option*:

☐ No SSL Encryption
☒ Self Signed Certification
☐ Upload CA & SSL Key

SSL Encryption Option*:

☐ No SSL Encryption
☐ Self Signed Certification
☒ Upload CA & SSL Key

CA: No file chosen

SSL Key: No file chosen

Click create to start the process of deploying the RGW.

After successfully deploying RGW on the selected host, you shall use the User Management function to create users for Amazon S3 and OpenStack Swift object storage. Then the RGW is ready for you to use.

You can test your RGW simply by entering its IP address and API port on any browser. For example, 192.168.1.111:7480. You will get a similar output on the browser.

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
<?xml version="1.0" encoding="UTF-8"?>
<ListAllMyBucketsResult xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <Owner>
    <ID>anonymous</ID>
    <DisplayName/>
  </Owner>
  <Buckets/>
</ListAllMyBucketsResult>
```

You can also create additional RADOS Gateways on other hosts by UVS manager.

Create Master/Standalone Rados Gateway

Select Node*: ☒ Internal
☐ External

mars500-219/192.168.3.219

Realm:

europa

Zonegroup:

germany

Note:

After the process of creating additional RGW is completed and The UVS manager return to the RGW page, you shall wait for about 30 seconds to have the new RGW up and running. Click the refresh button

RADOS Gateway

+ RADOS Gateway ▾

Pool Management

User Management

Zone Map

Download Config

Instances : 1

Change the number of RGW instances on all hosts

Search

Node Name	IP Address	API Port	Type	Action
mars500-218	192.168.3.218	TCP/7480(HTTP)	Standalone Gateway REALM = europe ZONEGROUP = germany ZONE = frankfurt	
mars500-219	192.168.3.219	TCP/7480(HTTP)	Standalone Gateway REALM = europe ZONEGROUP = germany ZONE = frankfurt	

Showing 1 to 2 of 2 rows

Delete RADOS Gateway

If you want to replace or remove RGW but keep the data, you shall use the “**Delete**” feature to remove and clean up the RGW container. This function purges only the ceph information on the RGW container, but no RADOS data will be deleted. Once you deleted the external RGW, you could deploy a new one anytime.

RADOS Gateway

+ RADOS Gateway ▾

Pool Management

User Management

Zone Map

Download Config

Search

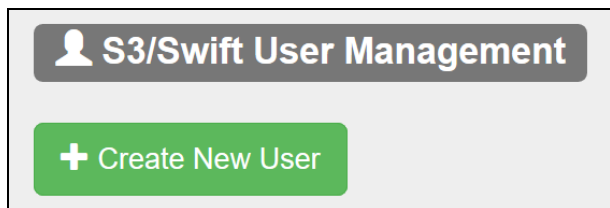
Node Name	IP Address	API Port	Type	Action
guestvm	192.168.1.110	TCP/7480(HTTP)	External Gateway REALM = asia ZONEGROUP = taiwan ZONE = taipei	
node111-1072	192.168.1.111	TCP/7480(HTTP)	Standalone Gateway REALM = asia ZONEGROUP = taiwan ZONE = taipei	
node113-1076	192.168.1.113	TCP/7480(HTTP)	Standalone Gateway REALM = asia ZONEGROUP = taiwan ZONE = taipei	

Showing 1 to 3 of 3 rows

Ceph Object Storage User Management

Create New Users

As mentioned earlier, Ceph Object Gateway has its own user management. You can access this menu from **“Object Storage” >> “User Management”** menu.



You will be presented with a pop-up to create a new Object Storage user for Swift and S3 compatible interfaces.

Create New User

UID*:

Name*:

Create

Enter the UID of the new user and a Full name to identify the new user. Once the user is successfully created, you will get the user listed with Swift and S3 keys.

UID	Name	Keys	Quota	Action
rgwuser	RGW User	S3 Access Key : 9JHR0EJS21UM8CHZ70WD S3 Secret Key : fSNI05D9uUSv6ZH2qPpiGShvyVbjbdN3gh0U2Cmo Swift UID : rgwuser:swift Swift Secret Key : FTHIbgmdYXJKBB6ZdEsoVp1XUzNT87NinOnvEbmg	Disabled	Edit Quota Delete

After you create the user, you can copy the keys and UID for S3 and Swift use.

Edit User Quota

Once the user is created, you can edit the user quota clicking the **“Edit Quota”** button against User. Put 0 (zero) in **“Quota”** to disable quota for the respective user.

Edit Quota

User ID:

rgwuser

Quota(GiB)[0 to disable quota]:

200

Update

Delete a User

The user can be deleted as well by clicking “**Delete**” in front of the user.

Confirm Delete User

Warning! All data & keys associated with the user will be deleted and is irrecoverable

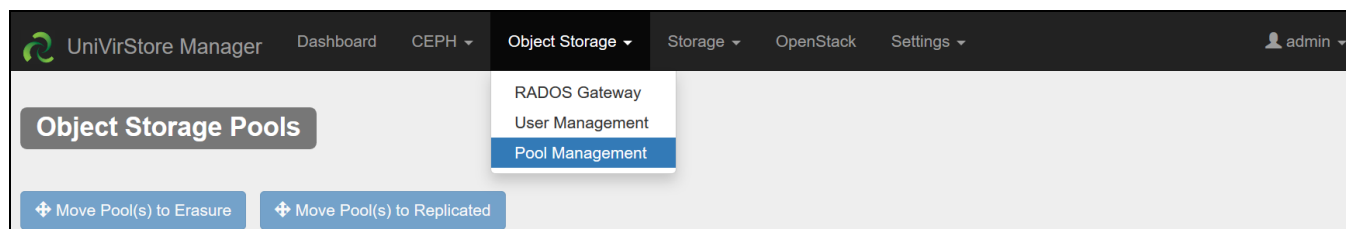
Cancel

Delete

Ceph Object Storage Pool management – RGW own pools

Ceph Object Gateways require Ceph Storage Cluster pools to store specific gateway data. If the user you created has permissions, the gateway will create the pools automatically. However, you should ensure that you have set an appropriate default number of placement groups per pool.

This task can be handled using Pool Management from “**Object Storage**” >> “**Pool Management**”



It takes you to the page from which you can edit the RADOSGW pools, including moving the pools from Replicated to Erasure and vice-versa. You can also modify the CRUSH Ruleset and number of Placement Groups for each pool.

By default, 19 RADOS Gateway pools are created and all the rgw pools have 8 Placement groups except default.rgw.buckets.data and default.rgw.buckets.index. The default.rgw.buckets.index has 32 placement groups. You can enter the number of placement group of default.rgw.buckets.data when you create the RGW.

Please refer to the on-line [Ceph PG calculator](#) for recommended PG numbers of all RGW pools. You can use the UVS manager Object Storage Pool Management edit function to adjust the PG number and other details of these RGW pools.

Object Storage Pools

Search

Pool Name

Type

Size

Application

EC Profile
Crush Rule

Placement
Groups

Quota

Compression

Mirror
Status

Action

.rgw.root

Replica

3

rgw

replicated_rule

8

Unlimited

Status : Disable

Not Support

+ Cache Tier

frankfurt.rgw.buckets.data

Replica

3

rgw

replicated_rule

256

Unlimited

Status : Disable

Not Support

+ Cache Tier

frankfurt.rgw.buckets.extra

Replica

3

rgw

replicated_rule

8

Unlimited

Status : Disable

Not Support

+ Cache Tier

frankfurt.rgw.buckets.index

Replica

3

rgw

replicated_rule

32

Unlimited

Status : Disable

Not Support

+ Cache Tier

frankfurt.rgw.buckets.non-ec

Replica

3

rgw

replicated_rule

8

Unlimited

Status : Disable

Not Support

+ Cache Tier

frankfurt.rgw.control

Replica

3

rgw

replicated_rule

8

Unlimited

Status : Disable

Not Support

+ Cache Tier

frankfurt.rgw.data.root

Replica

3

rgw

replicated_rule

8

Unlimited

Status : Disable

Not Support

+ Cache Tier

frankfurt.rgw.gc

Replica

3

rgw

replicated_rule

8

Unlimited

Status : Disable

Not Support

+ Cache Tier

frankfurt.rgw.intent-log

Replica

3

rgw

replicated_rule

8

Unlimited

Status : Disable

Not Support

+ Cache Tier

frankfurt.rgw.log

Replica

3

rgw

replicated_rule

8

Unlimited

Status : Disable

Not Support

+ Cache Tier

Showing 1 to 10 of 16 rows

10 rows per page

1

2

Editing RADOS Gateway Pools

The selected pool Replica size, CRUSH Ruleset and the number of Placement groups can be modified using the Edit button against each pool.

Edit Pool

Pools Name:

frankfurt.rgw.buckets.data

Replica Size*:

3

Quota(GiB)[0 to disable quota]:

0

Current Crush Rule:

replicated_rule

New Crush Rule*:

replicated_rule

Placement Groups:

256

Compression Status:

Disable

Compression Required Ratio:

0.85

Update

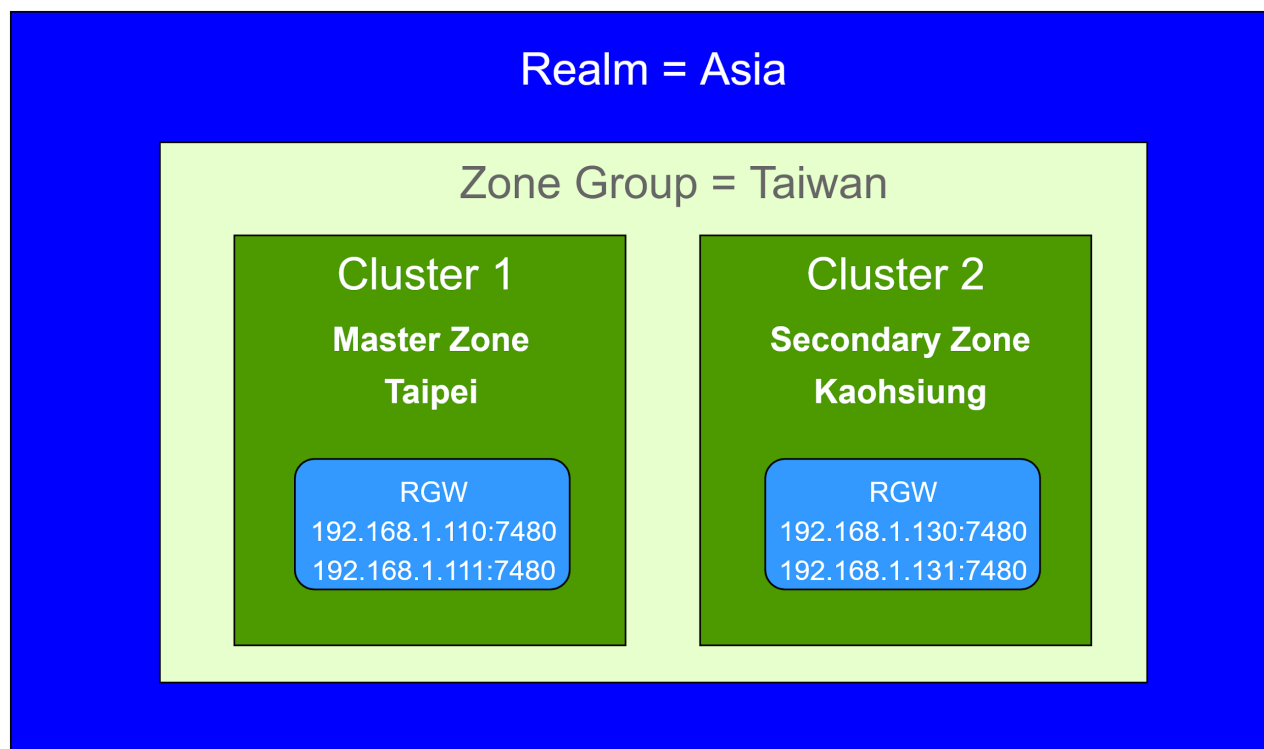
Close

This just saved you from the hassle of configuring RGW gateway and create/manage users to use the Gateway for interaction with Ceph cluster.

Setting Up Multi-Site RADOS Gateways

Ceph supports Active-Active multi-site installation to make data available on multiple different locations. Check the Ceph document for details. <https://docs.ceph.com/en/latest/radosgw/multisite/>

This user guide will help to set up MultiSite S3 Rados Gateway using UniVirStore Manager.



Steps to Build Multi-Site RADOS Gateways

1. You must have two or more Ceph clusters.
2. Rados gateways from different clusters should be able to ping each other.
3. The Ceph cluster at secondary zone shall not have RGW service.
4. Create one or more RGW from the Master Zone Ceph cluster.
5. Download the configuration file from the Master Zone
6. Create one or more Secondary RGW from the Secondary Zone Ceph cluster.

Creating Master RADOS Gateway

Please follow the steps in the chapter RADOS Gateway Management for how to deploy your RGW for the Master Zone.

Creating Secondary RADOS Gateway

To add another Ceph Cluster to an existing RADOS Cluster you need to create it as a Secondary RADOS Gateway. You will need the below information from the Master Zone.


- One of the Master Zone RGW IP
- REALM
- ZONEGROUP

Your secondary zones shall use the same Realm and Zone Group names as the Master Zone. And you shall use the different Zone names as the Master Zone.

RADOS Gateway					
<div> + RADOS Gateway Pool Management User Management Zone Map Download Config </div>					
<div> Search </div>					
Node Name	IP Address	API Port	Type	Action	
node111-1072	192.168.1.111	TCP/7480(HTTP)	Standalone Gateway REALM = asia ZONEGROUP = taiwan ZONE = taipei	Delete	
node112-1074	192.168.1.112	TCP/7480(HTTP)	Standalone Gateway REALM = asia ZONEGROUP = taiwan ZONE = taipei	Delete	

Showing 1 to 2 of 2 rows

1. Log in to the UVS Manager at Master Zone.
2. Click the **Download Config** from the Master Zone to download the configuration file.
3. Login to the UVS Manager at the Secondary Zone.
4. Under "**Object Storage**" >> "**RADOS Gateway**", Click on "+ **RADOS Gateway**" and Select "**MultiSite - Secondary**"



 **RADOS Gateway**

+ RADOS Gateway

Create RGW

Create Multi-Site Secondary

Search



Node Name	IP Address	API Port	Type	Action
No matching records found				

5. Select Internal or External host for deploying the RGW.

Create MultiSite Rados Gateway

Select Node*:

☒ Internal ☐ External


node138-103c/192.168.1.138

▼

Upload RGW Config File*:

Browse

No file selected

 Upload RGW Config

Close

Create MultiSite Rados Gateway ✕

Select Node*: ☐ Internal ☒ External

External RGW IP Address*:

External RGW Root Password*:

Upload RGW Config File*:

6. Click the Upload RGW Config
7. Select the Master RGW IP Address
8. The following page will ask you to enter **New Secondary Zone Name**, **CRUSH Rule** for RGW pools, the **number for PG for the RGW Bucket Data Pool** and **SSL Encryption Option**.

Create MultiSite Rados Gateway

Upload RGW Config

Master RGW IP Address*:

192.168.1.113

Master Realm:

asia

Existing Master Zonegroup*:

taiwan

New Secondary Zone*:

kaohsiung

Select Crush Rule*:

replicated_rule [Default]

Select RGW Bucket Data Pool PG*:

16

SSL Encryption Option*:

☒ No SSL Encryption

☐ Self Signed Certification

☐ Upload CA & SSL Key

Create

9. Click **“Create”**

3. This will create a MultiSite Secondary RADOS Gateway on the selected node.

UniVirStore Manager
Dashboard
CEPH
Object Storage
Storage
OpenStack
Settings
admin

RADOS Gateway

+ RADOS Gateway
Pool Management
User Management
Zone Map
Full Destroy RGW

Search

Node Name	IP Address	API Port	Type	Action
demo151-107a	192.168.1.151	TCP/7479(HTTP) TCP/7480(HTTPS)	Secondary Gateway REALM = mydatacentre ZONEGROUP = apac ZONE = taipei	Promote to Master

4. Click on “**Zone Map**” to see more details.

RGW MultiSite Zone Map

ZONEGROUP - apac

TYPE	ZONE	ENDPOINT
MASTER	singapore	http://192.168.1.111:7479
SECONDARY	taipei*	http://192.168.1.151:7479

* This Zone

Close

Promoting Secondary to Master

1. Under “**Object Storage**” >> “**RADOS Gateway**”, Click on “**Promote to Master**” to make this Secondary RADOS Gateway as Master RADOS Gateway

UniVirStore Manager
Dashboard
CEPH
Object Storage
Storage
OpenStack
Settings
admin

RADOS Gateway

+ RADOS Gateway
Pool Management
User Management
Zone Map
Full Destroy RGW

Search

Node Name	IP Address	API Port	Type	Action
demo151-107a	192.168.1.151	TCP/7479(HTTP) TCP/7480(HTTPS)	Secondary Gateway REALM = mydatacentre ZONEGROUP = apac ZONE = taipei	Promote to Master

RGW MultiSite Zone Map

ZONEGROUP - apac

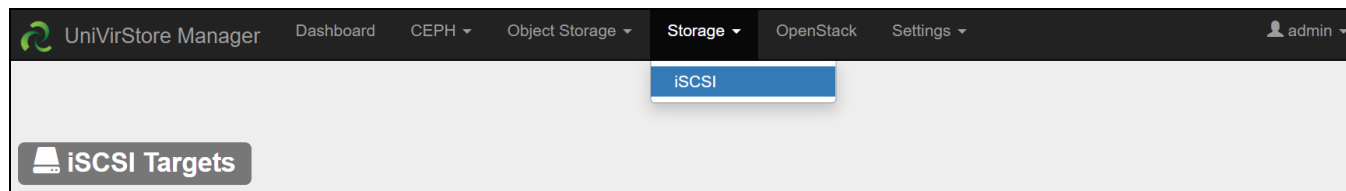
TYPE	ZONE	ENDPOINT
MASTER	taipei*	http://192.168.1.151:7479
SECONDARY	singapore	http://192.168.1.111:7479

* This Zone

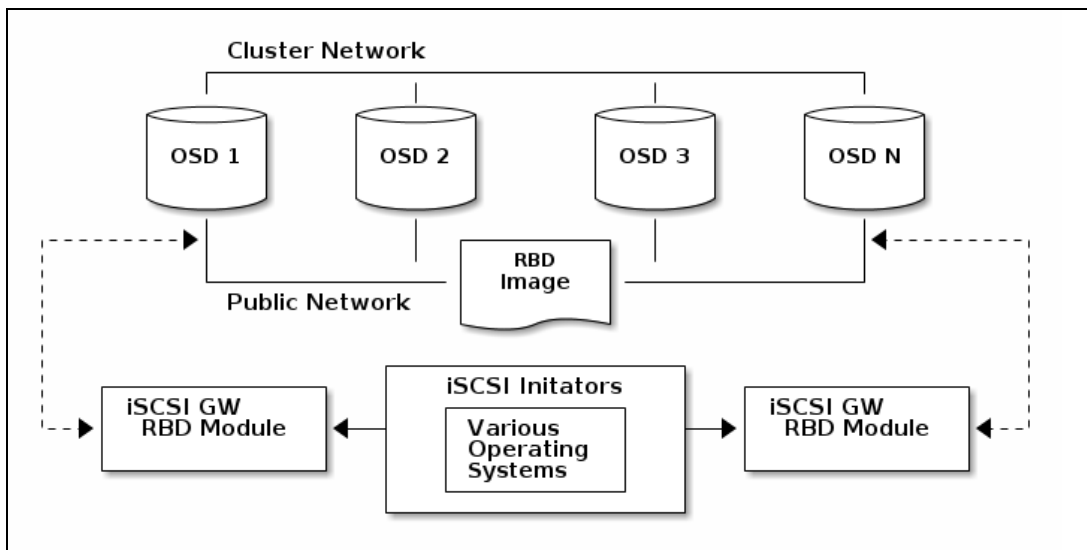
Close

iSCSI

Let us move to next section, which is Storage, more precisely, using iSCSI to mount RBD as Block device on client end.




The iSCSI gateway is integrating Ceph Storage with the iSCSI standard to provide a Highly Available (HA) iSCSI target that exports RADOS Block Device (RBD) images as SCSI disks. The iSCSI protocol allows clients (initiators) to send SCSI commands to SCSI storage devices (targets) over a TCP/IP network. This allows for heterogeneous clients, such as Microsoft Windows, to access the Ceph Storage cluster. With Ceph iSCSI gateway you can effectively run a fully integrated block-storage infrastructure with all the features and benefits of a conventional Storage Area Network (SAN).





With the Ceph RBD as iSCSI backend, iSCSI service has no single point of failure and can scale out its capacity and performance on demand. UVS manager support multiple iSCSI gateways failover and multi-path IO to provide high availability iSCSI service.


iSCSI Gateway



UVS manager support using internal Monitor nodes and external x86 servers as the iSCSI gateways. There are two options to set up your iSCSI gateways.


iSCSI Targets


Manage External Gateway



Create iSCSI LUN


Synchronize

Target	LUN	ACL	Action
No matching records found			

Run iSCSI gateways on Mars monitor nodes

1. Create iSCSI gateway on the Mars monitor nodes. Click the  Create iSCSI LUN button on the Storage → iSCSI page of the UVS manager.
2. Select a RBD image. (You must create the RBD image in advance.)
3. Select the Block or FileIO as the iSCSI back store.
4. Fill the CHAP and ACL authentication. (This is optional)
5. Click on the Create button. The LUN target will be available on all monitor nodes.

Create iSCSI LUN smaller text header

Create iSCSI LUN on*:

All CEPH Monitors

Backing Store (Pool/Image)*:

nvme/iscsi-lun1

Create iSCSI LUN Type*:

Block

CHAP Authentication and ACL:

Initiator Node IQN: Example: iqn.1993-08.org.debian:01:93c8fab31aa [ONE PER LINE]

Leave this filed blank if you do not want ACL based on Node IQN.
*For security reason, CHAP authentication shall also be used if you specify initiator IQN.

CHAP UserID: * If iSCSI initialtor is MS Windows , the String length should between 12-16 *

Leave Username and Password filed blank if you want to disable CHAP authentication.
*For security reason, CHAP authentication shall also be used if you specify initiator IQN.

CHAP Password: * If iSCSI initialtor is MS Windows , the String length should between 12-16 *

Create

Close

Target	LUN	ACL	Action
mars500-218 (192.168.3.218)	iqn.2013-02.org.nvme.iscsi-lun1.ambedded:sn.05c0fb7c POOL: nvme IMAGE: iscsi-lun1 Type: block Size: 100 GiB		Edit Delete Sessions
mars500-219 (192.168.3.219)	iqn.2013-02.org.nvme.iscsi-lun1.ambedded:sn.05c0fb7c POOL: nvme IMAGE: iscsi-lun1 Type: block Size: 100 GiB		Edit Delete Sessions
mars500-228 (192.168.3.228)	iqn.2013-02.org.nvme.iscsi-lun1.ambedded:sn.05c0fb7c POOL: nvme IMAGE: iscsi-lun1 Type: block Size: 100 GiB		Edit Delete Sessions

Showing 1 to 3 of 3 rows

Option 2: Use external servers as the iSCSI gateways

Using internal monitor nodes as iSCSI gateway has the advantages of saving cost on an extra server. However, colocate the iSCSI gateways will consume the Mars nodes' CPU, Memory and Network resources. You shall test the setup before you decide to use it for the production.

Create External iSCSI Gateway

Currently, Ubuntu 20.04 is supported as an External iSCSI Gateway.

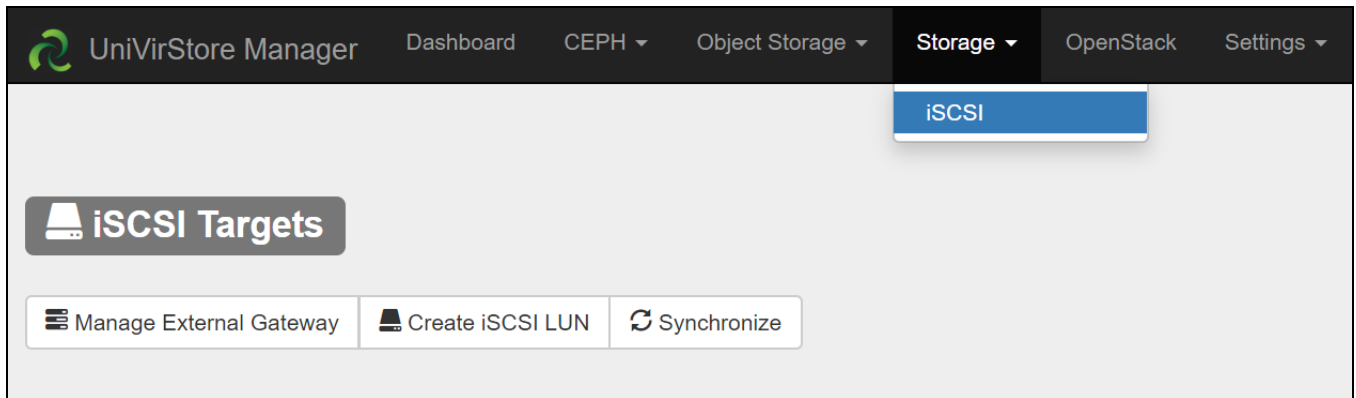
Prerequisite

Make sure before proceeding. Further, the below prerequisites are met

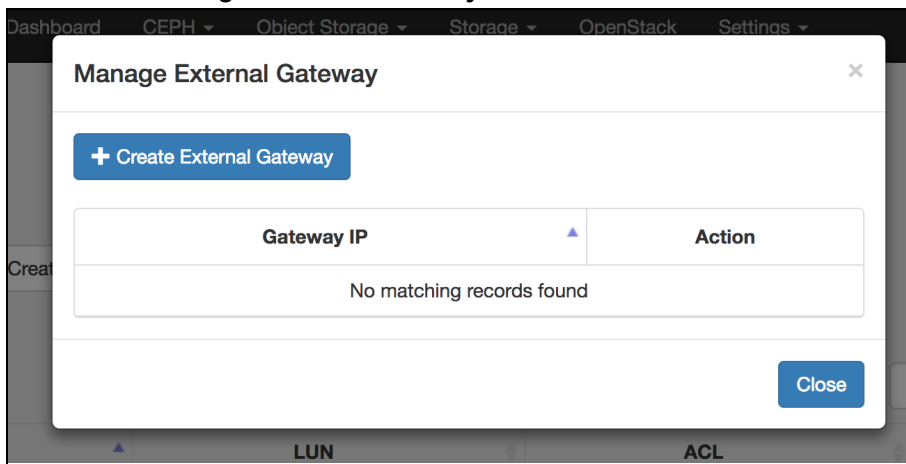
1. Ubuntu 20.04 should be installed on as bare metal or VM
2. Port 22 & root account should be allowed for SSH connection

Steps to Create External Gateway

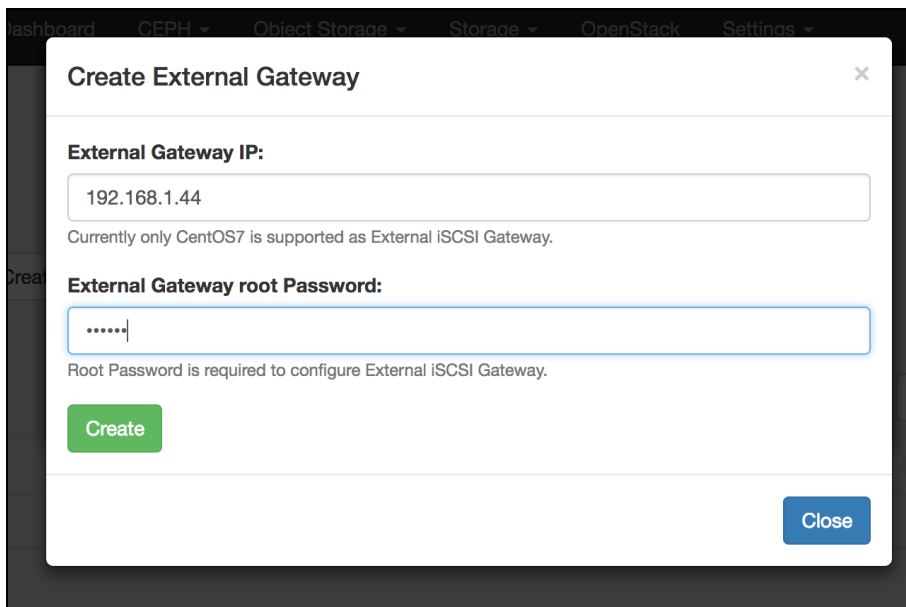
1. Once the above is done, log in to UVS Manager and navigate to **"Storage"** >> **"iSCSI"**



2. Click on “**Manage External Gateway**”



3. Click on “**Create External Gateway**” to add a new gateway

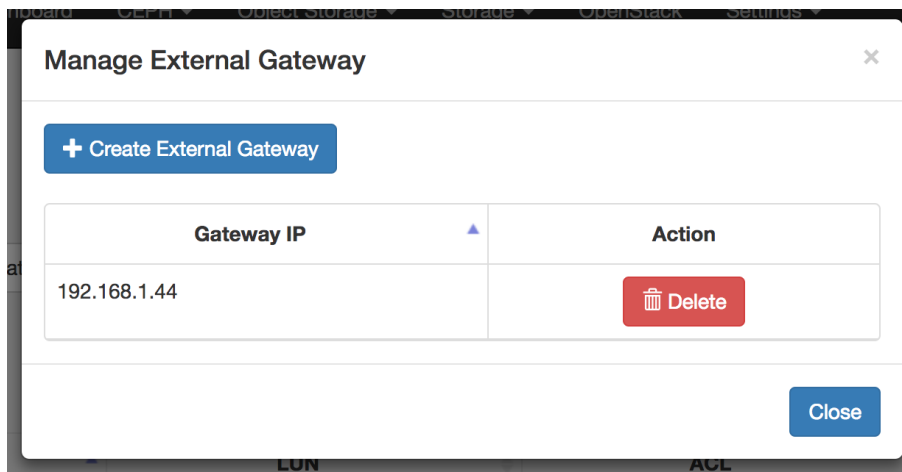


4. Type the “**IP Address**” and “**Root Password**” of External Gateway and click “**Create**”.

*Note: This may take time, depending on internet speed. If the page timeouts, don’t worry the operation continues in the background. You can click on “Manage External Gateway” again to check if the gateway is added to the list as shown below. If it is not shown, try adding it again. If still the issue persists, check the output. It should match the above output.

Delete External iSCSI Gateway

1. Navigate to “**Storage**” >> “**iSCSI**”
2. Click on “**Manage External Gateway**”
3. Make sure there are no iSCSI clients (initiators) connected to the gateway. Click on the “**Delete**” button next to the gateway IP Address.



Create LUN

Prerequisite

iSCSI LUN uses Ceph RBD image as it’s backend block storage. You have to create an RBD image for the LUN before you create the LUN.

LUNs can be created either on “All Monitor Nodes (Internal)” or “All External Gateways”. Once a LUN is created, it is ready to be used in HA by using iSCSI Multipath. You don’t have to create an iSCSI gateway on Monitor nodes as they are all ready to work when the UVS manager is installed.

1. Create iSCSI LUN on*: Choose “All CEPH Monitors” or “All External Gateways”.
2. Backing Store (Pool/Image)*: Select the RBD pool & image that you have created.
3. Create iSCSI LUN Type*: Choose “Block” or “FileIO” LUN type.

For security reasons, the UVS manager does not support access control with IQN only. If you wish to use the IQN-based ACL, you have to use both IQN and CHAP together. Type the Initiator IQN which can be found in your client (initiator node). E.g. /etc/iscsi/initiator_name.iscsi on the client. You can also allow multiple clients by entering initiator IQN one per line.

You can use CHAP without IQN. To use CHAP Authentication, please type the correct username and password. If you don’t want to use access control, you can leave the IQN, CHAP ID, and password blank.

4. Initiator Node IQN: The initiator IQN that is found in your clients.
5. CHAP UserID:
6. CHAP Password:

And click “Create”.

Hint: LUNs are created on multiple iSCSI gateways for multi-path IO use.

Create iSCSI LUN

Create iSCSI LUN on*:

All External Gateways

Backing Store (Pool/Image)*:

iscsi/i100

Create iSCSI LUN Type*:

Block

CHAP Authentication and ACL:

Initiator Node IQN: Example: iqn.1993-08.org.debian:01:93c8fab31aa [ONE PER LINE]

Leave this filed blank if you do not want ACL based on Node IQN.

CHAP UserID: * If iSCSI initialtor is MS Windows , the String length should between 12~16 *

Leave Username and Password fileds blank if you want to disable CHAP authentication.

CHAP Password: * If iSCSI initialtor is MS Windows , the String length should between 12~16 *

Create

Close

iSCSI Targets

Manage External Gateway

Create iSCSI LUN

Synchronize

Search

Target	LUN	ACL	Action
192.168.1.243 (192.168.1.243)	iqn.2013-02.org.iscsi-i200.ambedded:sn.17d50c45 POOL: iscsi IMAGE: i200 Type: fileio Size:200 GiB	INITIATOR IQN: iqn.1993-08.org.debian:01:93c8fab31aa	<div>Edit</div> <div>Delete</div> <div>Sessions</div>
192.168.1.243 (192.168.1.243)	iqn.2013-02.org.iscsi-i100.ambedded:sn.62490751 POOL: iscsi IMAGE: i100 Type: block Size:100 GiB	INITIATOR IQN: iqn.1991-05.com.microsoft:laptop-52g5r491	<div>Edit</div> <div>Delete</div> <div>Sessions</div>

Showing 1 to 2 of 2 rows

Client Sessions for Node 192.168.1.243

*NOTE: Client Session will be displayed only if Node ACL or CHAP Authentication is configured

alias: sid: 1 type: Normal session-state: LOGGED_IN
 name: iqn.1991-05.com.microsoft:laptop-52g5r491 (NOT AUTHENTICATED)
 mapped-lun: 0 backstore: block/iscsi_i100 mode: rw
 address: 192.168.1.22 (TCP) cid: 1 connection-state: LOGGED_IN
 Exit status : 0

Close

Delete LUN

The image shows that the LUN is created on all Internal/External Gateways. You can delete the by clicking the Delete button next to the LUN.

As this is a multi-path LUN it will also be deleted from all internal/External Gateways.

Configuring The iSCSI Initiators

iSCSI Initiator for Microsoft Windows

Follow the below Link

<https://docs.ceph.com/en/latest/rbd/iscsi-initiator-win/>

iSCSI Initiator for VMware ESX

Follow the below Link

<https://docs.ceph.com/en/latest/rbd/iscsi-initiator-esx/>

iSCSI Initiator for Linux

The below example demonstrates connecting a CentOS Client to External or Internal iSCSI gateway and uses Multipath

1. Install iSCSI Initiator Utils and multipathd

```
yum install iscsi-initiator-utils device-mapper-multipath
```

2. Enable Multipath

```
mpathconf --enable --with_multipathd y
```

```
systemctl enable multipathd  
systemctl start multipathd
```

3. If you are using CHAP authentication edit /etc/iscsi/iscsid.conf and add the below 3 lines

```
node.session.auth.authmethod = CHAP  
node.session.auth.username = username  
node.session.auth.password = password
```

and restart iscsid service

```
systemctl status iscsid
```

4. Discover All Target

```
[root@localhost ~]# iscsiadm -m discovery -t st -p 192.168.1.44  
192.168.1.44:3260,1 iqn.2003-01.org.linux-iscsi.aarch64:sn.00f3041e
```

```
[root@localhost ~]# iscsiadm -m discovery -t st -p 192.168.1.45
192.168.1.45:3260,1 iqn.2003-01.org.linux-iscsi.aarch64:sn.00f3041e
```

5. Login using the discovered LUN, iqn.2003-01.org.linux-iscsi.aarch64:sn.00f3041e in this example

```
[root@localhost ~]# iscsiadm -m node -T iqn.2003-01.org.linux-iscsi.aarch64:sn.00f3041e -l
Logging in to [iface: default, target: iqn.2003-01.org.linux-iscsi.aarch64:sn.00f3041e, portal: 192.168.1.44,3260]
(multiple)
Logging in to [iface: default, target: iqn.2003-01.org.linux-iscsi.aarch64:sn.00f3041e, portal: 192.168.1.45,3260]
(multiple)
Login to [iface: default, target: iqn.2003-01.org.linux-iscsi.aarch64:sn.00f3041e, portal: 192.168.1.44,3260]
successful.
Login to [iface: default, target: iqn.2003-01.org.linux-iscsi.aarch64:sn.00f3041e, portal: 192.168.1.45,3260]
successful.
```

6. The Above 2 LUNs should be visible in multipath and block device list

```
[root@localhost ~]# multipath -ll
mpatha (3600140590aa2c72b55644d4a1985a5bc) dm-2 LIO-ORG ,pool1_image4
size=10G features='0' hwhandler='0' wp=rw
|-+- policy='service-time 0' prio=1 status=active
|  ` 5:0:0:0 sdb 8:16 active ready running
`-+- policy='service-time 0' prio=1 status=enabled
   ` 4:0:0:0 sda 8:0  active ready running
```

```
[root@localhost ~]# lsblk
NAME        MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
sda         8:0    0 10G 0 disk
└─mpatha    253:2    0 10G 0 mpath
sdb         8:16    0 10G 0 disk
└─mpatha    253:2    0 10G 0 mpath
vda        252:0    0 30G 0 disk
├─vda1      252:1    0  1G 0 part /boot
├─vda2      252:2    0 29G 0 part
└─centos-root 253:0    0 26G 0 lvm  /
   └─centos-swap 253:1    0  3G 0 lvm  [SWAP]
```

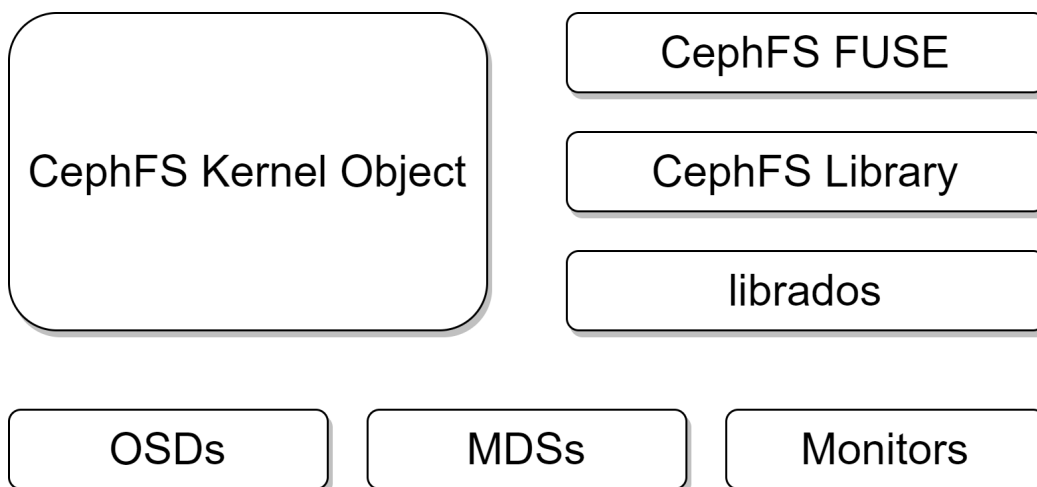
7. Notice that a device-mapper name was given to the LUN, mpatha in this example

*NOTE: DO NOT USE THE /dev/sda /dev/sdb BLOCK DEVICES IN ANY CASE. THIS WILL CORRUPT ALL THE DATE. INSTEAD USE /dev/mapper/mpatha

8. You can proceed further to format /dev/mapper/mpatha and just use it like any other device.

Ceph Filesystem

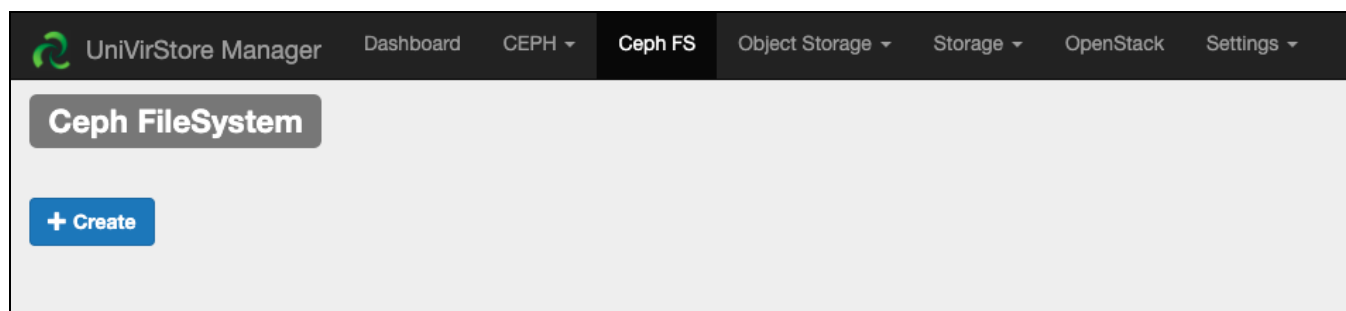
The Ceph Filesystem (CephFS) is a POSIX-compliant filesystem that uses a Ceph Storage Cluster to store its data. The Ceph filesystem uses the same Ceph Storage Cluster system as Ceph Block Devices, Ceph Object Storage. Using the Ceph Filesystem requires at least one Ceph Metadata Server in your Ceph Storage Cluster.



Ambedded UniVirStor Manager helps you to create MDS servers, create Ceph FS and bind Data and Metadata pools with the Ceph FS filesystem.

Creating CephFS

Click the "**CephFS**" menu to get to MDS / Ceph FS creation.



Under the CephFS drop-down, you get options to List the CephFS filesystem and Create a new CephFS filesystem.

1. Click "+ Create" to create the CephFS.
2. Fill the Filesystem Name you like.
3. You can choose Auto and Manual for creating data and metadata pools for CephFS use.

4. If you choose **Auto** mode, the UVS manager will automatically create two replica 3 pools with the default replicated CRUSH rule (by the host). This mode is only recommended for POC testing only. Manual mode enables you to get control of the pools.
5. If you choose **Manual** mode, you can use the existing pools or create new pools by giving the pool name, type (replica or erasure code), number of PG and CRUSH rule. Cephfs requires a replica pool for the metadata pool. You can't use the erasure code pool for metadata.
6. Enter one or more Mars node IP addresses to deploy active metadata servers. Please use the Mars nodes which do not have any Ceph daemon running on it as the active MDSs. You need at least one metadata server for using the Ceph file system. Once you deploy the active MDS, UVS manager will automatically deploy standby MDS on all Monitor nodes.

Auto Mode

Create Ceph File System

Filesystem Name*

mycephfs

Data Pool

☒ Automatic
 ☐ Manual

MetaData Pool

☒ Automatic
 ☐ Manual

MDS Nodes * (Seprate By Lines)

192.168.1.118

Create

UniVirStore Manager

Dashboard

CEPH

Ceph FS

Object Storage

Storage

OpenStack

Settings

admin

Ceph FileSystem

FileSystem Name	Metadata Pool	Data Pool	Action
mycephfs	mycephfs_pool_603865afbecfc.metadata	mycephfs_pool_603865afbecfc.data	How to Mount Destroy

MDS

Create New MDS

Node Name	IP Address	RANK	Status	Action
node118-1080	↑192.168.1.118	0	up:active	Remove
node111-1072	↑192.168.1.111	-1	up:standby	
node112-1074	↑192.168.1.112	-1	up:standby	
node113-1076	↑192.168.1.113	-1	up:standby	

Showing 1 to 4 of 4 rows

You can modify the data and metadata pools by using the Edit function on the Ceph → Pool page to change their replica size, CRUSH rule, number of PG, quota, and Compression settings.

UniVirStore Manager
Dashboard
CEPH
Ceph FS
Object Storage
Storage
OpenStack
Settings
admin

Ceph Pools

+ Create Pool
Delete Pool(s)

Search

<input type="checkbox"/>	Pool Name	Type	Size	Application	EC Profile Crush Rule	Placement Groups	Quota	Compression	Action
<input type="checkbox"/>	device_health_metrics	Replica	3	mgr_devicehealth	replicated_rule	1	Unlimited	Status : Disable	Edit + Cache Tier
<input type="checkbox"/>	mycephfs_pool_603865afbecfc.data	Replica	3	cephfs	replicated_rule	64	Unlimited	Status : Disable	Edit + Cache Tier
<input type="checkbox"/>	mycephfs_pool_603865afbecfc.metadata	Replica	3	cephfs	replicated_rule	32	Unlimited	Status : Disable	Edit + Cache Tier
<input type="checkbox"/>	rbd	Replica	3	rbd	replicated_rule	16	Unlimited	Status : Disable	Edit + Cache Tier

Showing 1 to 4 of 4 rows

Manual Mode

Create Ceph File System

Filesystem Name*

mycephfs

Data Pool

Automatic

Manual

Use Exist Pool :

Create New Pool

Create Pool

Pool Name* :

cephfs

Pool Type* :

Replicated

PG* :

64

Replicated Size* :

3

CRUSH

Crush Ruleset* :

replicated_rule

115 of 147

Metadata Pool

☐ Automatic
 ☒ Manual

☐ Use Exist Pool :
 ☒ Create New Pool

Create Pool

Pool Name* :

metadata

PG* :

16

Replicated Size* :

3

CRUSH

Crush Ruleset* :

replicated_rule

MDS Nodes * (Seprate By Lines)

192.168.1.118

Create

UniVirStore Manager

Dashboard

CEPH

Ceph FS

Object Storage

Storage

OpenStack

Settings

admin

Ceph FileSystem

FileSystem Name	Metadata Pool	Data Pool	Action
mycephfs	metadata	cephfs	How to Mount Destroy

MDS

Create New MDS

Node Name	IP Address	RANK	Status	Action
node118-1080	↑192.168.1.118	0	up:active	Remove
node111-1072	↑192.168.1.111	-1	up:standby	
node112-1074	↑192.168.1.112	-1	up:standby	
node113-1076	↑192.168.1.113	-1	up:standby	

Showing 1 to 4 of 4 rows

Create More Metadata Servers

You can create more active MDS to scale out the performance of metadata servers according to the load of client users at any time.

1. Click the Create New MDS button.
2. Fill the IP address of nodes you want to deploy the MDS.
3. Click Create.

Add MDS

IP Address*:

Mon(s) are default as StandBy MDS , Please Don't INPUT Mon(s)/OSD/MDS IP!!

Create

Close

MDS Failover and Failback

CephFS is a high available file system. If any active MDS fails due to any reason, one of the standby MDS will become active to keep the client services.

MDS

+ Create New MDS

CephFS

Search

Node Name	IP Address	RANK	Status	Action
demo111-1072	↑192.168.1.111	0	up:active	
demo112-1074	↑192.168.1.112	-1	up:standby	
demo113-1076	↑192.168.1.113	-1	up:standby	

Showing 1 to 3 of 3 rows

After the failed primary MDS goes online again, it will run as standby MDS. You can use the “Failback” feature to make the primary MDS ACTIVE again. Click the “Failback” and wait for one to two minutes, finally, all MDS on monitor nodes will become standby, and the primary MDS becomes Active.

MDS

+Create New MDS

CephFS

Search

Node Name	IP Address	RANK	Status	Action
demo118-1080	↑192.168.1.118	-1	up:standby	Remove Fallback
demo111-1072	↑192.168.1.111	0	up:active	
demo112-1074	↑192.168.1.112	-1	up:standby	
demo113-1076	↑192.168.1.113	-1	up:standby	

After failback, the primary MDS becomes active again.

MDS

+Create New MDS

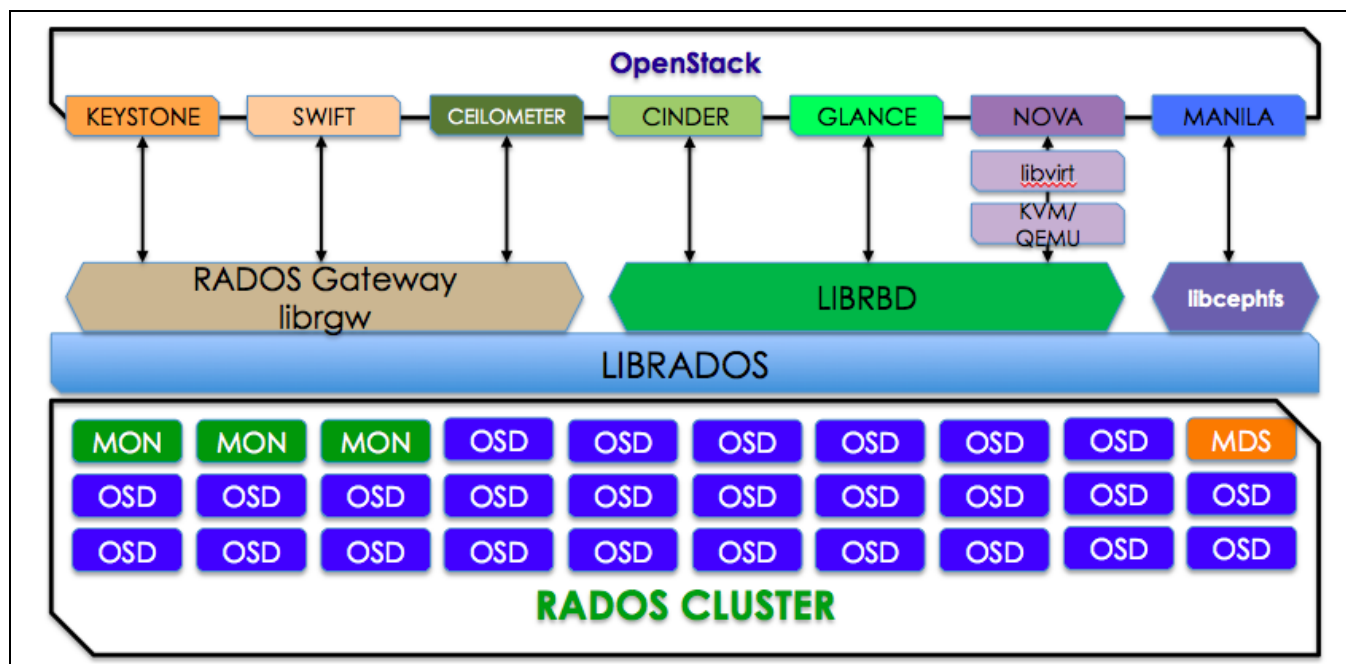
CephFS

Search

Node Name	IP Address	RANK	Status	Action
demo118-1080	↑192.168.1.118	0	up:active	Remove
demo111-1072	↑192.168.1.111	-1	up:standby	
demo112-1074	↑192.168.1.112	-1	up:standby	
demo113-1076	↑192.168.1.113	-1	up:standby	

Integrating Ceph with OpenStack – Reliable Storage Backend

Cloud platforms like OpenStack require a storage system that is reliable, scalable, unified and distributed. Ceph integrates easily with OpenStack components like cinder, glance, nova, and keystone. Ceph provides low-cost storage for OpenStack, which helps in getting the cost down. Another advantage of using Ceph is that it provides a unified storage solution for OpenStack. Ceph delivers file, object and block storage for OpenStack. The Ceph block storage has capabilities like thin provisioning, snapshot, cloning, which helps to spin up VM's quickly and makes backing up and cloning of VM's easy. The copy on write mechanism of Ceph allows OpenStack to spin up many instances at once, Ceph can provide persistent boot volumes for OpenStack instances. Ceph also provides API for swift and s3 storage interfaces.



For OpenStack components like Glance, Cinder and Nova to interact with Ceph, apart from configuration in OpenStack .conf files, they also need to have CephX authentication for OpenStack users. This utility configures the Users and authentication along with creating separate pools for each utility, namely; .glance for Glance, .cinder for Cinder and .nova for use with Nova OpenStack components.

OpenStack Backend Storage

+ Create Backend Pools & Auth Keys

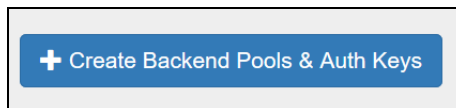
Search

OpenStack Component	Pool Name	CephX User Name	Key	Action
No matching records found				

119 of 147

Creating OpenStack Backend pools & Auth Keys

Click the "+ **Create Backend Pools & Auth keys**" button to create CephX authentication keys and respective pools.



Create Pools & Keys for OpenStack

PG*:

128

Replica Size*:

3

Crush Rule*:

replicated_rule

Create

Close

Once the process is complete, you will be presented with the page showing Ceph Users for Cinder, Glance and Nova along with keys and a Download button to download the keys.

OpenStack Backend Storage

Search

OpenStack Component	Pool Name	CephX User Name	Key	Action
CINDER	.cinder	cinder	AQDCRsNeB9KTDxAADmolGruM4RF15y/H5BI70g==	Download Key
GLANCE	.glance	glance	AQC8RsNeCsS/LhAAb/iBKaNvCYNxt+blhxaV2A==	Download Key
NOVA	.nova	nova	AQDHRsNe+GKPExAAEvDh8C/9gyEcES+LwU28BQ==	Download Key

Showing 1 to 3 of 3 rows

Download and save Keyring and ceph.conf file

UVS Manager helps you to generate the Ceph client keyring for client.glance, client.cinder and client.nova and also the ceph.conf file, which can be downloaded and copied for client access.

OpenStack Component	Pool Name	CephX User Name	Key	Action
CINDER	.cinder	cinder	AQDCRsNeB9KTDxAADmolGruM4RF15y/H5BI70g==	Download Key

Click the "**Download Key**" button against each user to download the compressed file, which contains the respective Ceph user authentication keyring along with the ceph.conf file

Creating OpenStack pools and users manually – Doing Ceph way

UVS Manager OpenStack option lets you create Pools and keys to use with OpenStack single click of key, however, it doesn't allow us to create pools and CephX users with your defined values (till as of now. We are working to get this done your way !).

We need to follow the manual way of creating pools individually and then associating them with CephX users along with managing capabilities required by Ceph for allowing OpenStack integration.

Refer [Creating Ceph Pool](#) for steps to create the pools.

Create Replicated Pool

Pools Name*:

ospcinder

PG*:

128

Replica Size*:

3

Application*:

Block Storage (rbd)

Crush Rule*:

replicated_rule

Quota(GiB)[0 to disable quota]:

500

Compression Status:

Disable

Compression Required Ratio:

0.85

Create

Create a pool with a custom-defined name (should be able to identify the purpose pool being used for).

	<input type="checkbox"/> Pool Name ▲	Type ▾	Size ▾	EC Profile/Crush Rule ▾	Placement Groups ▾	Quota ▾
+	<input type="checkbox"/> ospcinder	Replica	3	replicated_ruleset	128	Unlimited
+	<input type="checkbox"/> ospglance	Replica	3	replicated_ruleset	128	Unlimited
+	<input type="checkbox"/> ospnova	Replica	3	replicated_ruleset	128	Unlimited

Once the pools are created, we can go ahead to create the CephX users for respective pools association and adding capabilities.

Ref section [Create CephX User](#) for steps to create CephX users.

Create User

User Name*:

Create

You will get output similar to as shown below.

```
[client.x.ospcinder]
    key = AQDcbEBZEmvv0RAAxBhvu6+/Vt5nRve19HARKQ==
Exit status : 0
```

Notice the user being created. It has “.x” embedded to the user ID been created. We need to use the same username with OpenStack also.

Now that we have Pool to be used with OpenStack and also a CephX user, we need to associate the pool and CephX user using Ceph capabilities.

Refer [Edit CephX User](#) to edit or add capabilities to user.

Click Edit against the User to edit the User capabilities, i.e. assigning permissions/capabilities to pools.

Edit User ospcinder

Capabilities [One Per Line]:

```
allow rwx pool=ospcinder
allow rwx pool=ospnova
allow rwx pool=ospglance
```

Capabilities will vary from CephX user to user and OpenStack version to different versions. Once you have edited /added the capabilities, click Update. The output should be similar to as shown below.

```
updated caps for client.x.ospcinder
Exit status : 0
```

User Name	Key	Capabilities	Action
ospcinder	AQDcbEBZEmwORAAxBhvu6+/Vt5nRVei9HARKQ==	allow rwx pool=ospcinder allow rwx pool=ospnova allow rwx pool=ospglance	<div>Delete</div> <div>Edit</div> <div>Download Key</div>

We can follow the steps for other users as well as required by your OpenStack implementation.

User Name	Key	Capabilities	Action
ospcinder	AQDcbEBZEmwORAAxBhvu6+/Vt5nRVei9HARKQ==	allow rwx pool=ospcinder allow rwx pool=ospnova allow rwx pool=ospglance	<div>Delete</div> <div>Edit</div> <div>Download Key</div>
ospglance	AQD9LT9ZhguiGBAAzSRgDJTDwJ2+Zi23vhD0ig==	allow rwx pool=ospglance	<div>Delete</div> <div>Edit</div> <div>Download Key</div>

Mitaka introduced the support of RBD snapshots while doing a snapshot of a Nova instance, we need to allow the Cinder CephX user key write access as well to the Glance pool. Hence in the above snapshot, we see “ allow rwx pool=ospglance”.

The below table shows the capabilities to use for respective pools :

	Glance	allow rwx pool=ospglance
--	--------	--------------------------

Prior to Mitaka

	Cinder	allow rx pool=ospglance allow rwx pool=ospcinder allow rwx pool=ospnova
	Nova	allow rwx pool=ospnova allow rx pool=ospglance
After Mitaka	Glance	allow rwx pool=ospglance
	Cinder	allow rwx pool=ospglance allow rwx pool=ospcinder allow rwx pool=ospnova
	Nova	allow rwx pool=ospnova allow rx pool=ospglance

Once you have edited the capabilities as per requirement, you can go ahead and download the keys and share them with the OpenStack nodes.

Remember to use user-name as client.x.<username>. The key is also created in the same way, e.g.
ceph.client.x.ospcinder.keyring

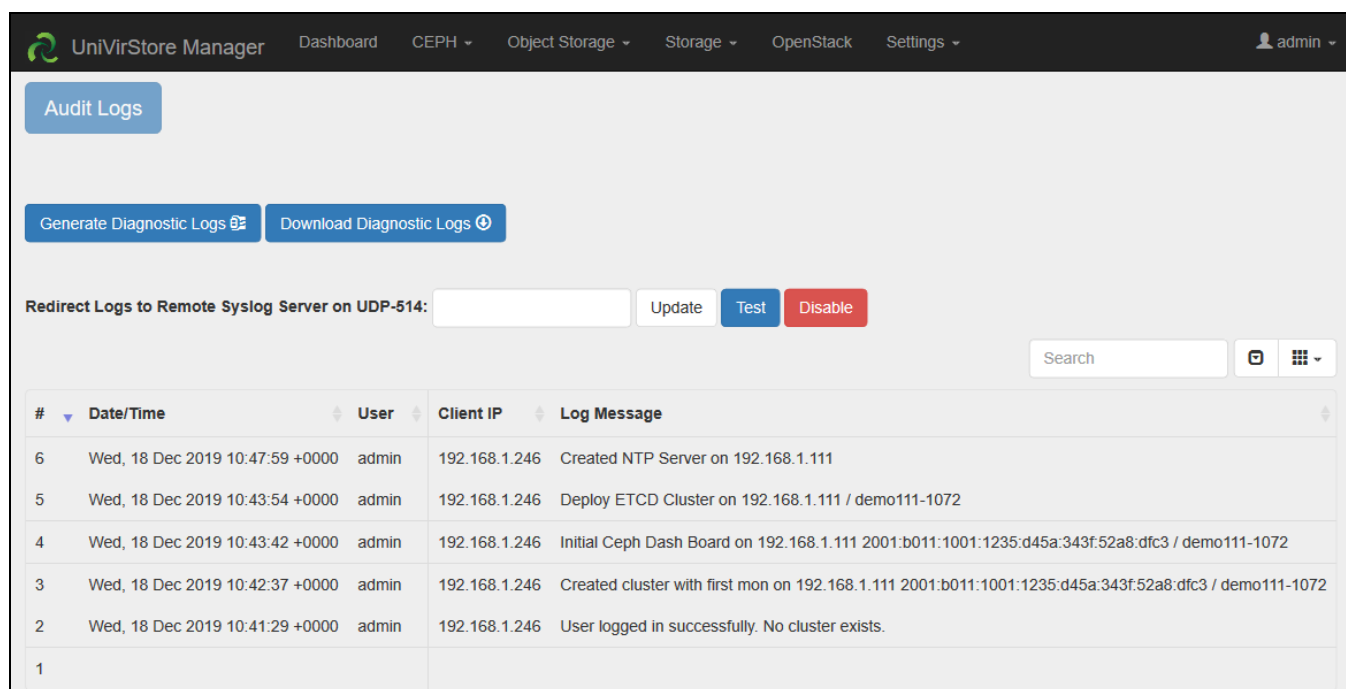
As we are nearing the end of this WebUI walk-through, we come to the menu option, which plays a core role in the working of the Ceph cluster.

Audit Logs – tracking the actions on WebUI

In this Audit Logs page, You can collect two types of logs.

Audit Logs

The first type is the audit log. The Audit Log can be used to automatically track every action undertaken by users on the Ceph by UVS manager. It can record what time a user logged on, which action they performed and status after the action was done.



UniVirStore Manager Dashboard CEPH Object Storage Storage OpenStack Settings admin

Audit Logs

Generate Diagnostic Logs Download Diagnostic Logs

Redirect Logs to Remote Syslog Server on UDP-514: Update Test Disable

Search

#	Date/Time	User	Client IP	Log Message
6	Wed, 18 Dec 2019 10:47:59 +0000	admin	192.168.1.246	Created NTP Server on 192.168.1.111
5	Wed, 18 Dec 2019 10:43:54 +0000	admin	192.168.1.246	Deploy ETCD Cluster on 192.168.1.111 / demo111-1072
4	Wed, 18 Dec 2019 10:43:42 +0000	admin	192.168.1.246	Initial Ceph Dash Board on 192.168.1.111 2001:b011:1001:1235:d45a:343f:52a8:dfc3 / demo111-1072
3	Wed, 18 Dec 2019 10:42:37 +0000	admin	192.168.1.246	Created cluster with first mon on 192.168.1.111 2001:b011:1001:1235:d45a:343f:52a8:dfc3 / demo111-1072
2	Wed, 18 Dec 2019 10:41:29 +0000	admin	192.168.1.246	User logged in successfully. No cluster exists.
1				

You can also use the syslog server to forward the Audit Log to an external Syslog server.



Redirect Logs to Remote Syslog Server on UDP-514: Update Test Disable

Please type the IP address of the external rsyslog server and test it.

success! 192.168.1.245 is reachable on 514/UDP

After test the rsyslog server, you shall click the **“Update”** button and apply it. Also, you could **“Disable”** the rsyslog function in the same page.

Redirect Logs to Remote Syslog Server on UDP-514:

#	Date/Time	User	Client IP	Log Message
12	Thu, 19 Dec 2019 06:21:36 +0000	admin	192.168.1.246	Updated remote syslog server to 192.168.1.245

Output Logs – the details to the UVS operations

In this page, you could see the output logs of the UVS operations that you ever executed before. There are three buttons, “**Detail**”, “**View Log**” & “**Download**”. These three functions help you to diagnostic the UVS operations.

UniVirStore Manager
Dashboard
CEPH
Object Storage
Storage
OpenStack
Settings
admin

Output Log List

#	Date/Time	Execute PHP	Execute At IP	Command
6	2020-05-19 02:38:51	/ntp.php	192.168.1.111	pushchrony.yml <input type="button" value="Detail"/> <input type="button" value="View Log"/> <input type="button" value="Download"/>
5	2020-05-19 02:35:58	/ntp.php	192.168.1.111	pushtimezone.yml <input type="button" value="Detail"/> <input type="button" value="View Log"/> <input type="button" value="Download"/>
4	2020-05-19 14:34:13	/ntp.php	192.168.1.111	pushtimezone.yml <input type="button" value="Detail"/> <input type="button" value="View Log"/> <input type="button" value="Download"/>
3	2020-05-19 14:27:47	/ntp.php	192.168.1.111	deployextntpsrver.yml <input type="button" value="Detail"/> <input type="button" value="View Log"/> <input type="button" value="Download"/>
2	2020-05-14 10:52:11	/ntp.php	192.168.1.111	pushchrony.yml <input type="button" value="Detail"/> <input type="button" value="View Log"/> <input type="button" value="Download"/>
1	2020-05-14 10:33:43	/nodes.php	192.168.1.111	osdadd.yml <input type="button" value="Detail"/> <input type="button" value="View Log"/> <input type="button" value="Download"/>

Showing 1 to 6 of 6 rows

© 2020 Ambedded Technology Co. Ltd.; Taiwan. All Rights Reserved | UniVirStore Manager Version 2.14.17 for aarch64

Command Detail
✕

```
TERM=xterm sudo /usr/bin/ansible-playbook /var/www/html/playbooks/pushchrony.yml -e ntp_server=tw.pool.ntp.org -e tzvalue="America/Detroit"|stdbuf -o0 grep TASK|stdbuf -o0 sed -e 's/TASK/==>/'
```

Close

Output Log

```

Command :TERM=xterm sudo /usr/bin/ansible-playbook /var/www/html/playbooks/pushchrony.yml -e ntp_server=tw.pool.ntp.org
==> [Change TimeZone to "America/Detroit"] *****
==> [fail the play if ntp_server is not defined] *****
==> [Copy Client Config] *****
==> [restart_chrony] *****
==> [Force NTP Sync] *****
==> [SUCCESS] *****
Exit status : 0


```

Close

Prometheus

UVS version 2.14.17 and newer versions integrate Prometheus Ceph & node exporters. Prometheus is an open-source project for monitoring and collecting system metrics with a dimensional data model.

Although Ceph has its own Prometheus module, for the high availability, we implement the UVS Prometheus feature by independent exporters & Prometheus main services. The Prometheus ceph & node exporters are the services that display the current status by the key-value. The Prometheus service collects the key-value and makes it into a time-series database. In the UVS “**Settings**” menu, you could find a “**Prometheus**” page; click it, and you’ll see the WebUI below.


UniVirStore Manager
Dashboard
CEPH
Object Storage
Storage
OpenStack
Settings
admin

Prometheus Configuration

Node Exporter : ☐ OFF
Ceph Exporter : ☐ OFF

Prometheus :

MONs

Retention Size

Retention Time

Please Select Node(s)
Auto GB
400 Days
Create Prometheus

Deploy On Nodes	Service	URL	Size(GB)	Time(Day)	Action
Apply Change(s)					

Exporter services

The exporter services capture the current system metrics and export their key-values.

You could **turn on/off** the switches to enable/disable the exporter services.

Node Exporter :	<input type="checkbox"/>	Ceph Exporter :	<input type="checkbox"/>
Node Exporter :	<input type="checkbox"/>	Ceph Exporter :	<input type="checkbox"/>

If you desire to know what kind of metrics the exporters share, please enable the services and check the URLs on node's port 9100 for node_exporter and port 9128 for ceph_exporter.

<p>← → ↺ ↻ 192.168.1.111:9100/metrics</p> <pre># HELP node_forks_total Total number of forks. # TYPE node_forks_total counter node_forks_total 5.4809789e+07 # HELP node_intr_total Total number of interrupts serviced. # TYPE node_intr_total counter node_intr_total 6.0753535e+08 # HELP node_load1 1m load average. # TYPE node_load1 gauge node_load1 1.27 # HELP node_load15 15m load average. # TYPE node_load15 gauge node_load15 1.62</pre>
<p>← → ↺ ↻ 192.168.1.111:9128/metrics</p> <pre>ceph_monitor_latency_seconds{cluster="ceph",monitor="demo112-1074"} 0.00510472 ceph_monitor_latency_seconds{cluster="ceph",monitor="demo113-1076"} 0.005612493 # HELP ceph_monitor_quorum_count The total size of the monitor quorum # TYPE ceph_monitor_quorum_count gauge ceph_monitor_quorum_count{cluster="ceph"} 3 # HELP ceph_osd_avail_bytes OSD Available Storage in Bytes # TYPE ceph_osd_avail_bytes gauge ceph_osd_avail_bytes{cluster="ceph",osd="osd.0"} 3.905178304e+12 ceph_osd_avail_bytes{cluster="ceph",osd="osd.1"} 3.90518272e+12 ceph_osd_avail_bytes{cluster="ceph",osd="osd.2"} 3.905107904e+12 ceph_osd_avail_bytes{cluster="ceph",osd="osd.3"} 3.905202176e+12 # HELP ceph_osd_average_utilization OSD Average Utilization # TYPE ceph_osd_average_utilization gauge ceph_osd_average_utilization{cluster="ceph"} 2.555441794885004 # HELP ceph_osd_backfill_full OSD Backfill Full Status # TYPE ceph_osd_backfill_full gauge ceph_osd_backfill_full{cluster="ceph",osd="osd.0"} 0</pre>

Prometheus service

Before you can use the Prometheus server, you shall use UVS to deploy the Prometheus service. First, you have to select at least one monitor node as a Prometheus server. Before you push the **“Create Prometheus”** button, you could decide the **retention size and time** for the Prometheus database.

Prometheus :

MONs
demo111-1072 ▾

Retention Size
Auto ▾ GB

Retention Time
400 ▾ Days

Create Prometheus

☐ Select all
☒ demo111-1072
☐ demo112-1074
☐ demo113-1076

Deploy On Nodes	Service	URL	Size(GB)	Time(Day)	Action
demo111-1072	ON	http://192.168.1.111:9090/	42 GB	400 Days	Cleanup

Apply Change(s)

After the creation of the Prometheus server, UVS shows details about the Prometheus servers. You could check the URL right now, and the link will take you to the Prometheus dashboard. Also, you could use the URL as a data source of Grafana, to build a comprehensive monitoring system.

Prometheus :

MONs
Please Select Node(s) ▾

Retention Size
Auto ▾ GB

Retention Time
400 ▾ Days

Create Prometheus

Deploy On Nodes	Service	URL	Size(GB)	Time(Day)	Action
demo111-1072	ON	http://192.168.1.111:9090/	42 GB	400 Days	Cleanup

Apply Change(s)

If you want to change the retention settings or disable the service, remember to click the “**Apply Changes**” button. And the “**Cleanup**” is a function that will destroy all of the Prometheus data and service in the selected node; please be careful.

Prometheus :

MONs
Please Select Node(s) ▾

Retention Size
Auto ▾ GB

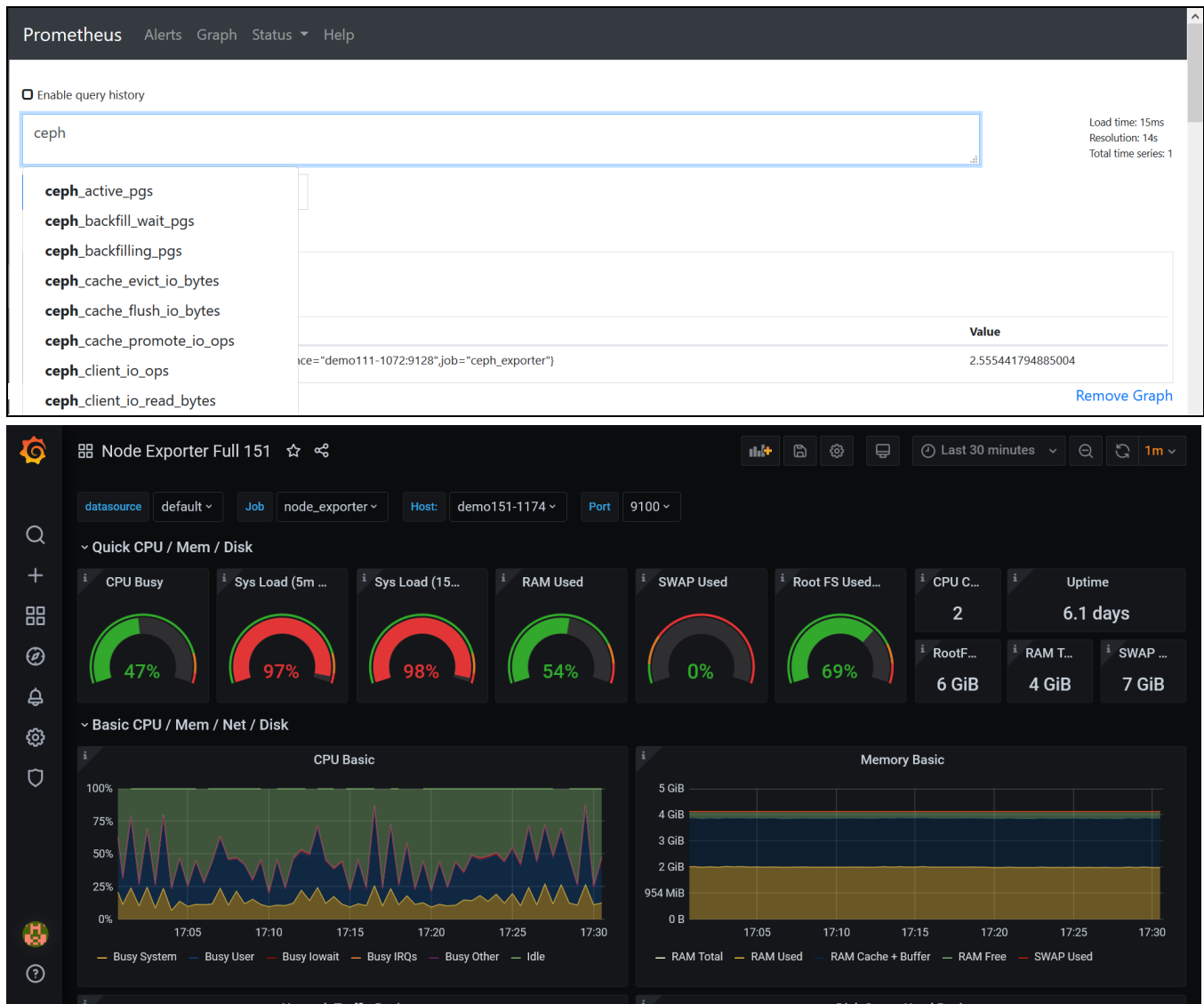
Retention Time
400 ▾ Days

Create Prometheus

Deploy On Nodes	Service	URL	Size(GB)	Time(Day)	Action
demo111-1072	ON	http://192.168.1.111:9090/	19 GB	240 Days	Cleanup

Apply Change(s)

In the Prometheus dashboard, you can search the metrics and see the time-series data. They are simple but limited. Hence, the alternative solution for cluster monitoring is the combination of Prometheus and Grafana. Use Mars nodes as the data source and build your Grafana server on another machine to get a robust monitoring system.



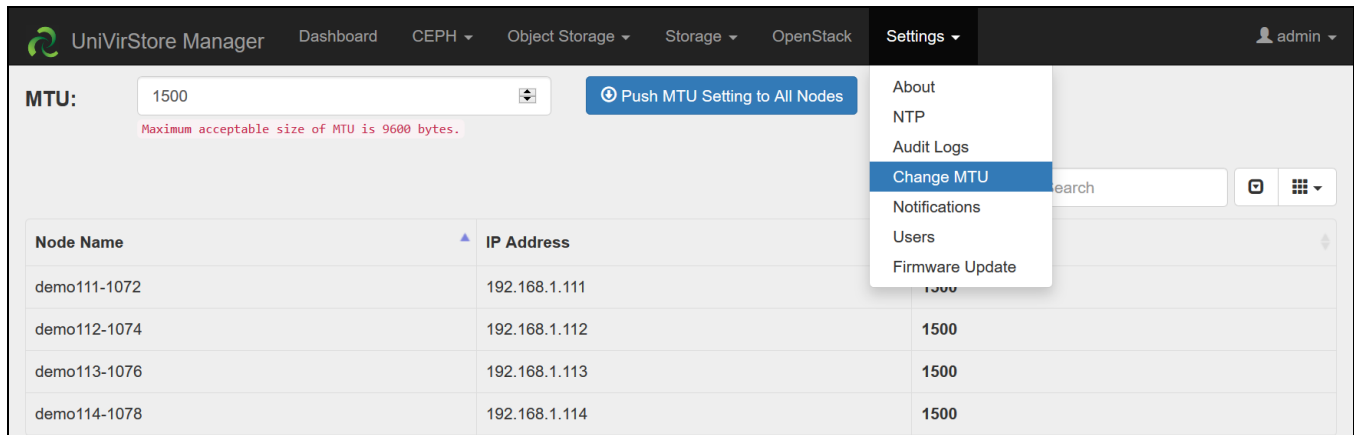
We've tested the following grafana dashboards for you. Please go to Grafana Labs and try the grafana IDs.

node exporter: 1860 & 8919

ceph exporter: 917

Change the MTU of Mars Nodes

The default MTU value of Mars 400 Nodes is **1500**. If you put the Mars 400 Ceph cluster into the jumbo frame environment, you shall adjust the MTU setting via this function.



UniVirStore Manager Dashboard CEPH Object Storage Storage OpenStack Settings admin

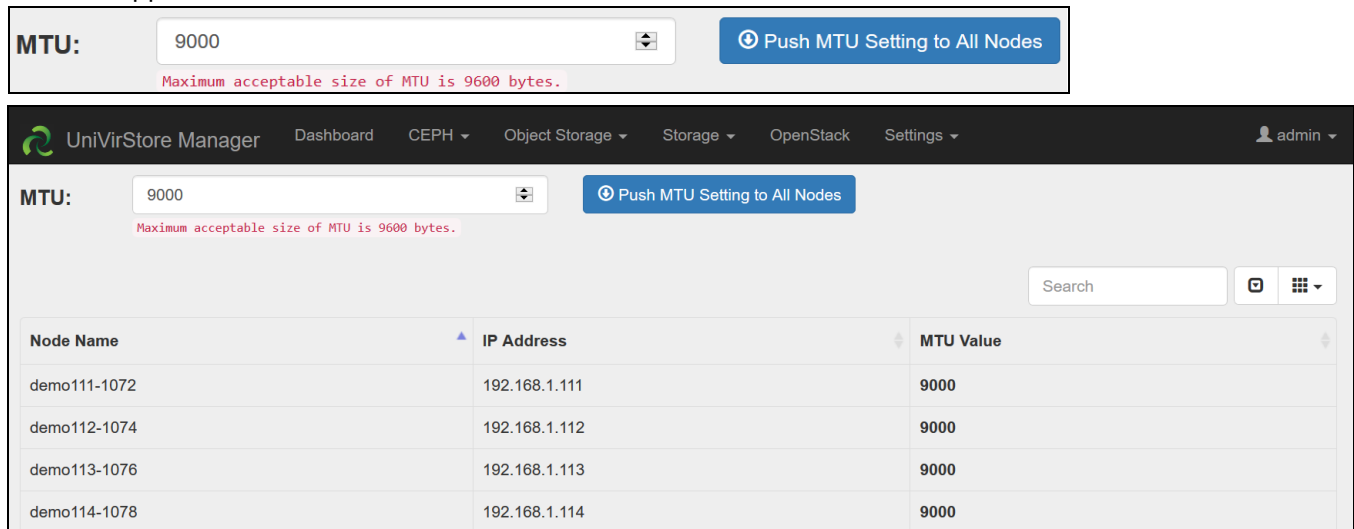
MTU: 1500 [Push MTU Setting to All Nodes](#)

Maximum acceptable size of MTU is 9600 bytes.

- About
- NTP
- Audit Logs
- Change MTU**
- Notifications
- Users
- Firmware Update

Node Name	IP Address	MTU Value
demo111-1072	192.168.1.111	1500
demo112-1074	192.168.1.112	1500
demo113-1076	192.168.1.113	1500
demo114-1078	192.168.1.114	1500

Fill the MTU value that you want and press the “Push MTU Setting to All Nodes” button. You will see the new MTU has applied.



UniVirStore Manager Dashboard CEPH Object Storage Storage OpenStack Settings admin

MTU: 9000 [Push MTU Setting to All Nodes](#)

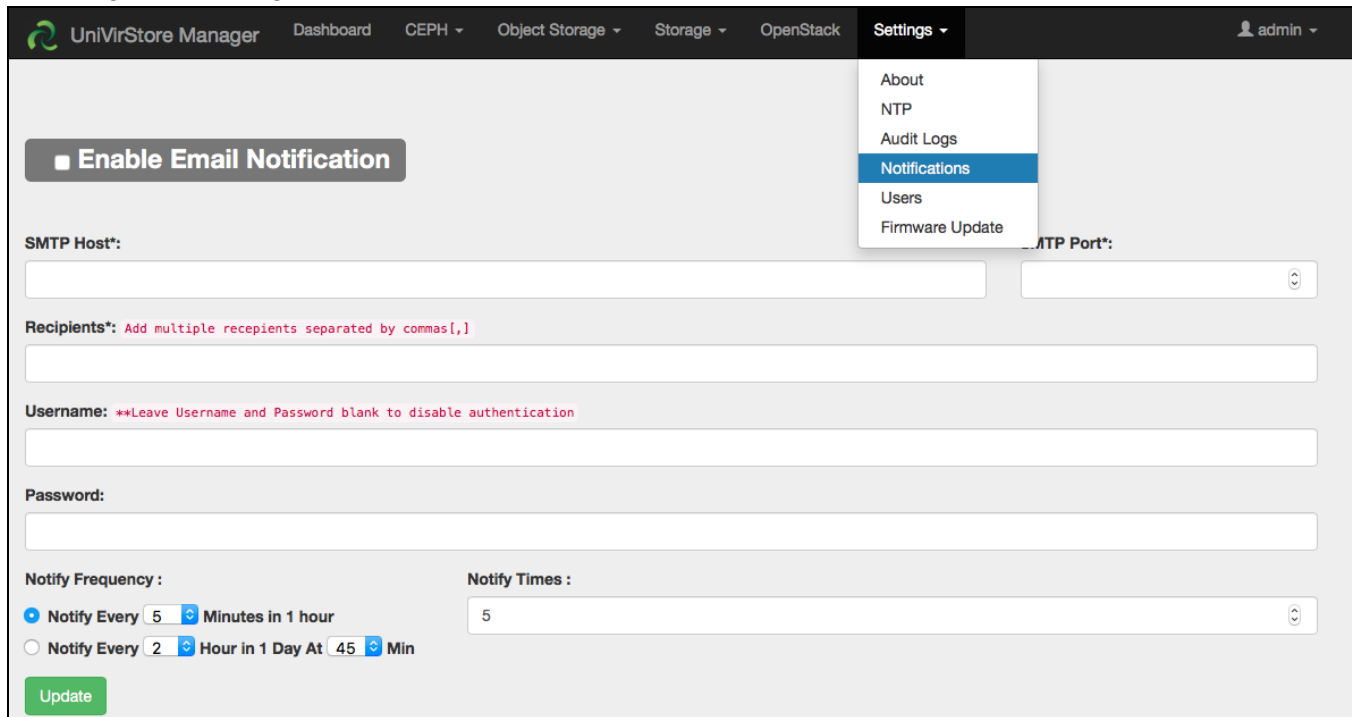
Maximum acceptable size of MTU is 9600 bytes.

Search

Node Name	IP Address	MTU Value
demo111-1072	192.168.1.111	9000
demo112-1074	192.168.1.112	9000
demo113-1076	192.168.1.113	9000
demo114-1078	192.168.1.114	9000

Notification – Alerts on email

The Ceph cluster has been configured to send the status update using Notification to configured e-mail addresses so that it is not required to have user monitor the Dashboard all the time. In case there is any warning and error reported from Ceph, the user will get an email notification alert on predefined email addresses according to the settings.



You need to fill the following required information to configure Notification feature after placing a Check-mark (✓) on Enable Email Notification button.

- ◆ SMTP Host: the name or the IP address of the SMTP server to be used for sending Notification email alert.
- ◆ SMTP Port: SMTP port for SMTP host.
- ◆ Recipients: e-mail address(es) of the recipient who should get the notification email alert.
- ◆ Username: e-mail address to be used to send an alert email message.
- ◆ Password: Password for the respective email address.

“Notification Times” is the number of repeat email notifications you would like to receive for every event. The default number is 5 and you can enter a number or click the up/down arrow at the right side of input field to change the setting.

You can use “Notify Frequency” to set the duration between each notification. There are two options.

1. UVS manager will send notifications every specified period until all repeated email notifications are sent. For example, you can ask the UVS manager to send 5 repeated emails every 5 minutes right after a new event happened.
2. UVS will repeatedly send email at the specified time after every specific hour(s). For example, when you

choose “Notify 5 times” and “Notify Every 2 Hours in 1 Day at 45 min” and an event happens at 3:20AM, UVS manager will send 5 emails at 3:20AM, 5:45AM, 7:45AM 9:45AM and 11:45AM.

Select the Change Password, enter new password and lastly click Update to update the password.

✓ Enable Email Notification

SMTP Host*:

smtp.gmail.com

SMTP Port*:

465

Recipients*:

Add multiple receipients separated by commas[,]

aaron@ambedded.com.tw

Username:

**Leave Username and Password blank to disable authentication

aaron@ambedded.com.tw

✓ Change Password:

Enter Password

Notify Frequency :

☒ Notify Every

5

Minutes in 1 hour

☐ Notify Every

2

Hour in 1 Day At

45

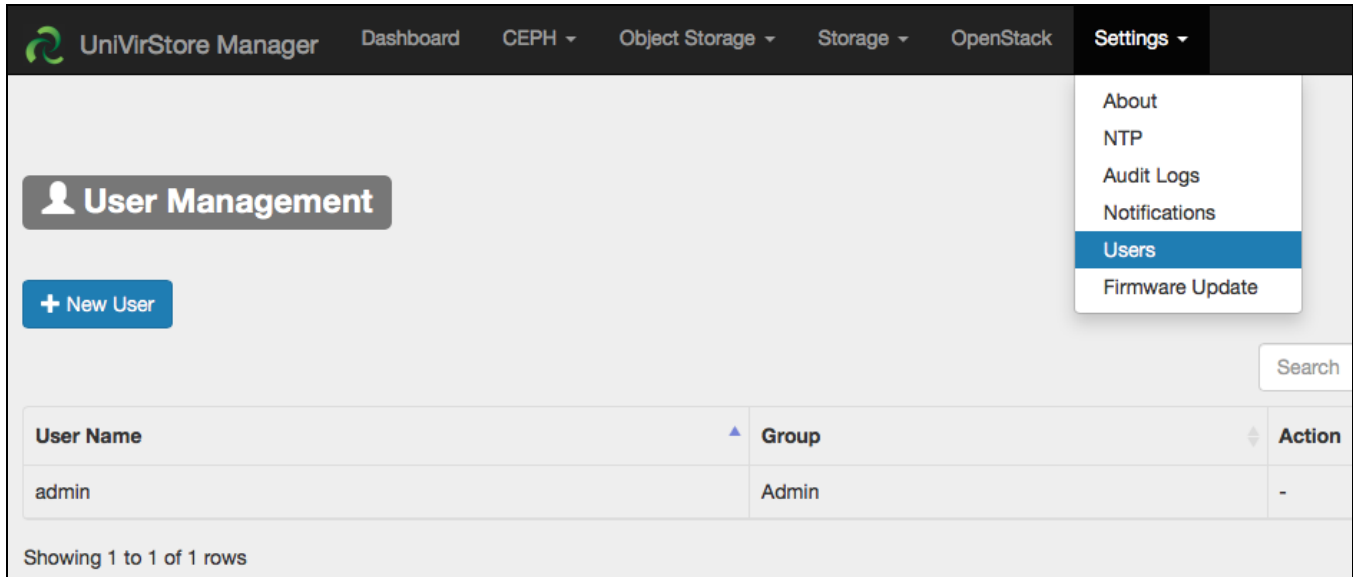
Min

Notify Times :

5

Update

Dashboard User – Edit and View rights



User Name	Group	Action
admin	Admin	-

Showing 1 to 1 of 1 rows

User admin is created when the UI is enabled. Additional users can be created by clicking the “**+New User**” button.



Create User


User Name*:
uvsuser

Password*:
.....



Group:
Admin

Create

Enter Username, Password and Group for the new user and click “**Create**”.


 **User Management**

[+ New User](#)

User Name	Group	Action
admin	Admin	-
uvsuser	View	Edit


The Password can be changed or user deleted by clicking the “**Edit**” button.

Edit User 

Reset Password*: uvsuser

Group:

Admin



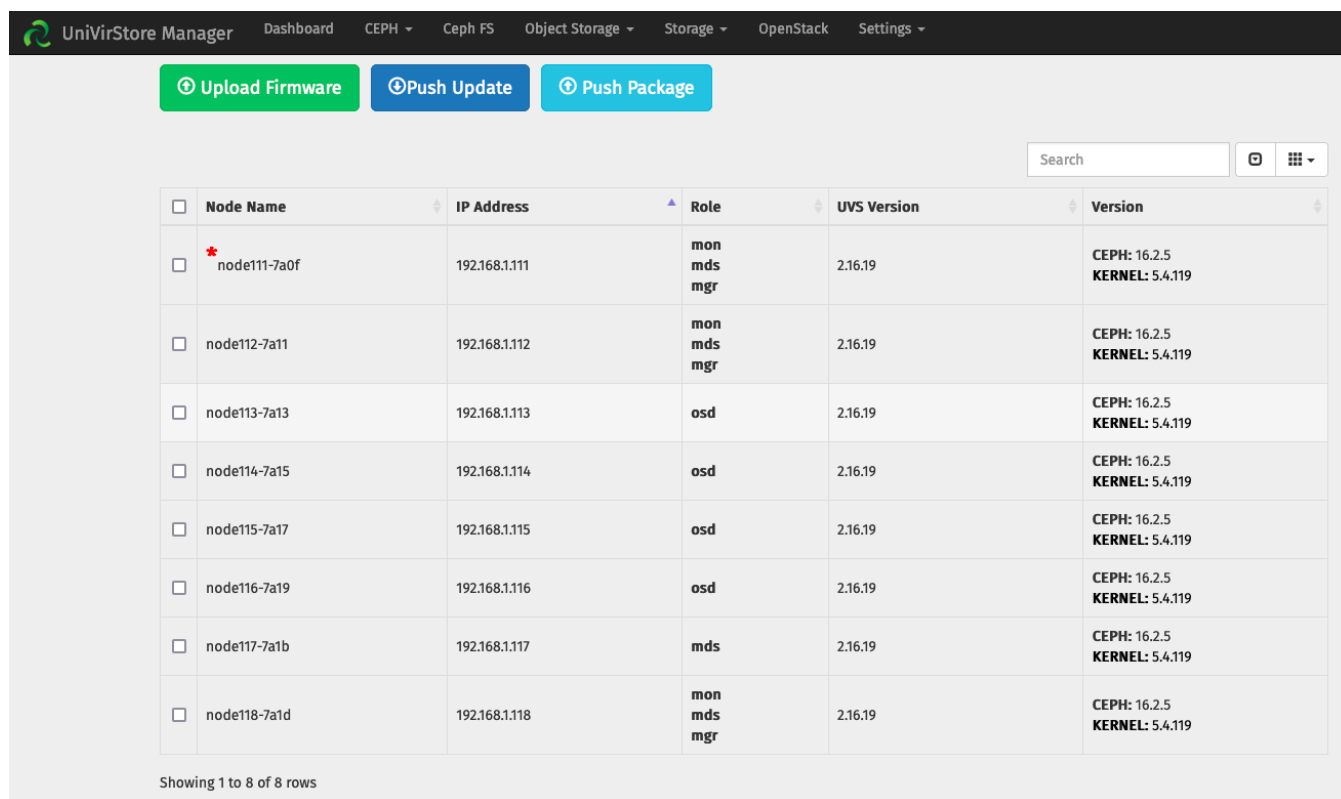
[Update](#)

Firmware Update

Notice:

[UVS Manager updates may include the upgrade for web UI firmware, Ceph software, Linux kernel and the Linux distribution. What is included in the update varies version by version. Please refer to the release notes for the details of the update procedure.](#)

In some cases, you have to upgrade the firmware on one of the Monitor nodes by the UVS admin console before you can update all other nodes' firmware by the UVS manager. Ther release note of every UVS release will have detailed information for the complete update procedure.



<input type="checkbox"/>	Node Name	IP Address	Role	UVS Version	Version
<input type="checkbox"/>	* node111-7a0f	192.168.1.111	mon mds mgr	2.16.19	CEPH: 16.2.5 KERNEL: 5.4.119
<input type="checkbox"/>	node112-7a11	192.168.1.112	mon mds mgr	2.16.19	CEPH: 16.2.5 KERNEL: 5.4.119
<input type="checkbox"/>	node113-7a13	192.168.1.113	osd	2.16.19	CEPH: 16.2.5 KERNEL: 5.4.119
<input type="checkbox"/>	node114-7a15	192.168.1.114	osd	2.16.19	CEPH: 16.2.5 KERNEL: 5.4.119
<input type="checkbox"/>	node115-7a17	192.168.1.115	osd	2.16.19	CEPH: 16.2.5 KERNEL: 5.4.119
<input type="checkbox"/>	node116-7a19	192.168.1.116	osd	2.16.19	CEPH: 16.2.5 KERNEL: 5.4.119
<input type="checkbox"/>	node117-7a1b	192.168.1.117	mds	2.16.19	CEPH: 16.2.5 KERNEL: 5.4.119
<input type="checkbox"/>	node118-7a1d	192.168.1.118	mon mds mgr	2.16.19	CEPH: 16.2.5 KERNEL: 5.4.119

Showing 1 to 8 of 8 rows

You will get to the window from where you can **upload** the firmware to the node and also **push** them to all the nodes in the cluster in one go. This can be thought of as a central firmware management console. UVS supports two methods for the firmware update. Ambedded releases an UPD file for UVS firmware update. And a RPM file is the patch for the UVS software patch.

UPD Update

First, click "**Upload Firmware**" to upload the firmware file.

Upload New Firmware

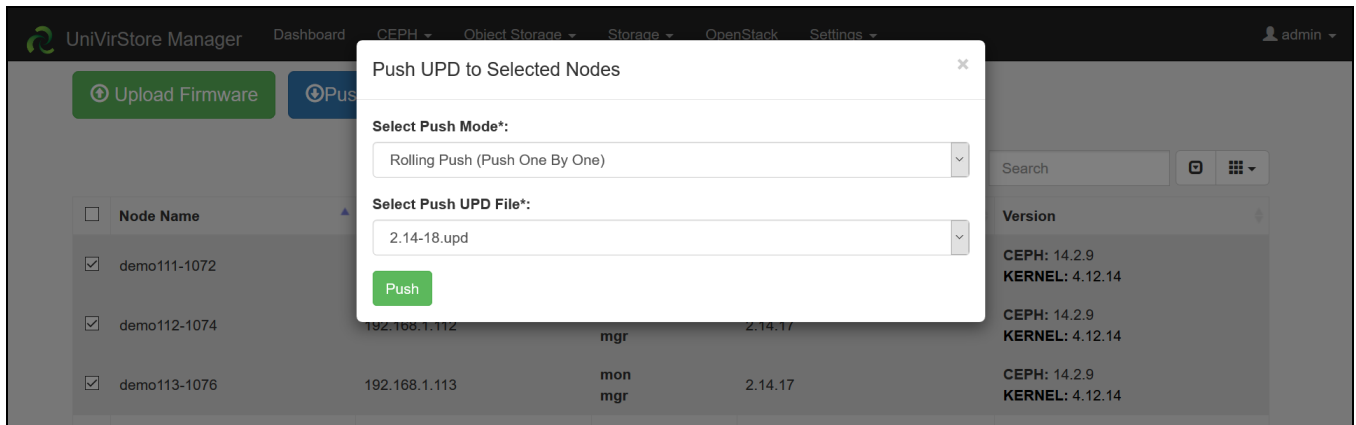
Select File*:

Browse... 2.14-18.upd

Upload

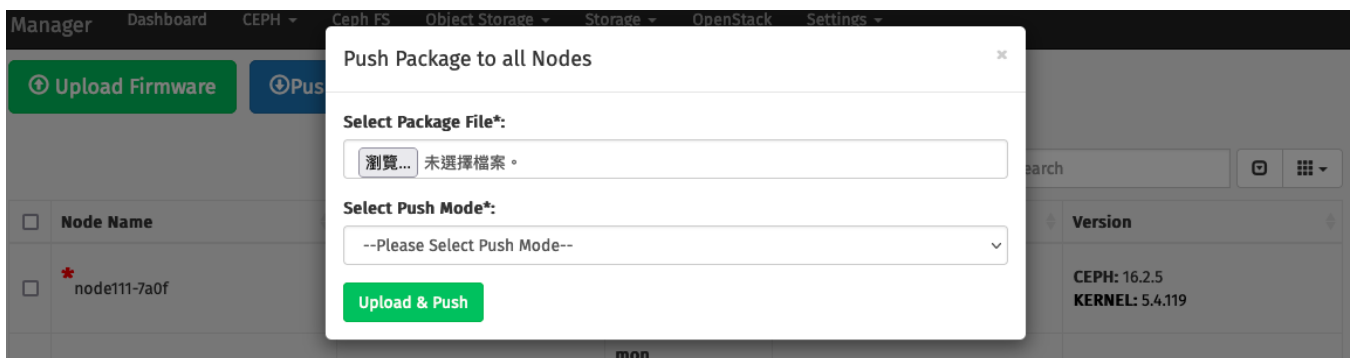
Existing Firmware in Cache

Then, **select the nodes** you would like to push and update firmware by checking the box on the left side of Node Name. Click "**Push Update**", select the push mode and upd file. Finally, click the "**Push**" button to push the uploaded firmware to the selected nodes in the cluster. UVS manager will automatically upgrade the firmware of these nodes.



Package Update

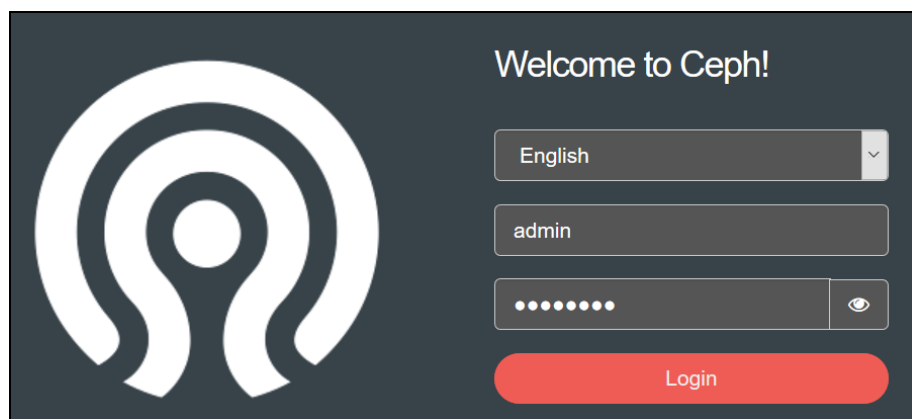
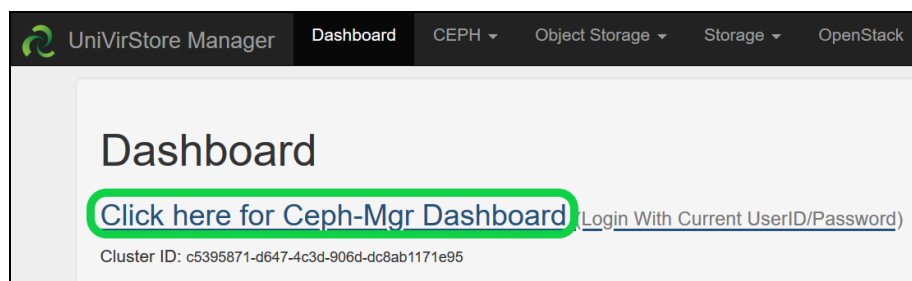
Select the nodes you would like to push and update firmware by checking the box on the left side of Node Name. Click "**Push Package**" to push and install the packages to all the nodes in the cluster.



Ceph-MGR Zabbix Module

Enable the Zabbix Module

To enable the ceph-mgr Zabbix module, we'd like to lead you to the Ceph-MGR dashboard. There is a link for the Ceph-MGR dashboard in the UVS Dashboard page. Click the link, you'll see the welcome page of the Ceph-MGR dashboard, and use the default account (admin/admin) login. Once you've logged in, we recommend changing the password immediately.



To manage the Ceph-MGR modules, you could go to the “**Cluster**” menu, “**Manager modules**” page. All of the Ceph-MGR modules are listed here. Now, we can enable the Zabbix module. Find the Zabbix module via the search bar, **edit** the settings, and enable it.

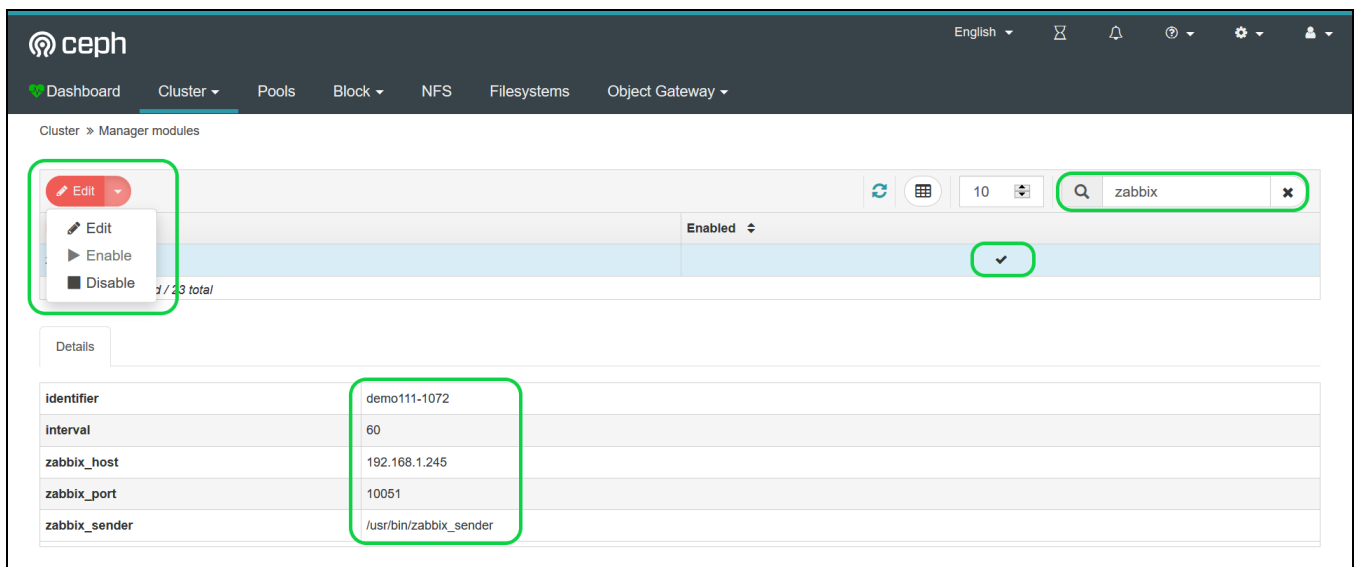
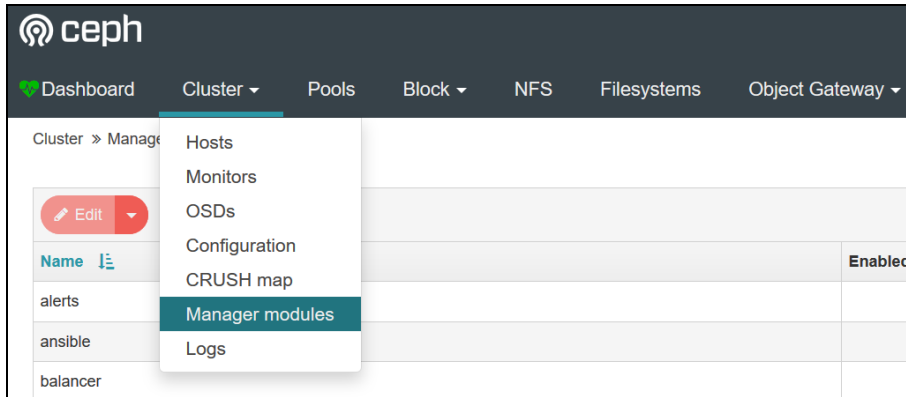
identifier: The name of the Zabbix sender. The monitor hostname is a good choice.

interval: The frequency of the data sending.

zabbix_host: The IP address of the Zabbix Server.

zabbix_port: The port of the Zabbix Service.

zabbix_sender: The path to the zabbix sender in the Ceph-MGR nodes.



The Zabbix Template

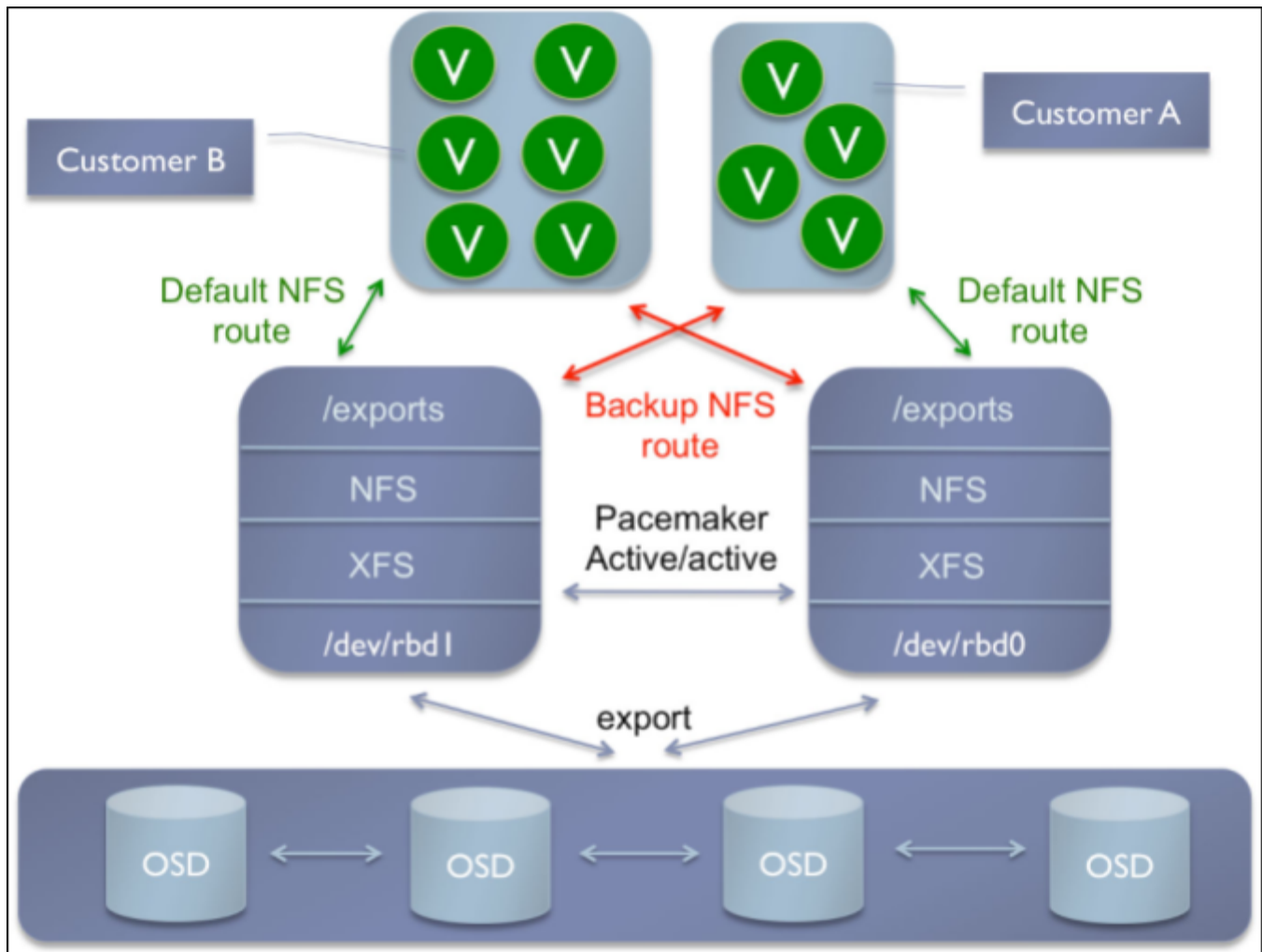
You could download the specific template XML file from the Ceph Github source. Also you could find it in the Mars400 nodes. Make sure the template matches the version to the Ceph cluster to get the best practice. If you want to check if the Zabbix module is workable, try the “ceph zabbix send” command in a console.

ceph zabbix send
Sending data to Zabbix

- **Github:** https://github.com/ceph/ceph/blob/v14.2.9/src/pybind/mgr/zabbix/zabbix_template.xml
- **Local path in Mars400 nodes:** /usr/share/ceph/mgr/zabbix/zabbix_template.xml

NFS & SAMBA protocol - using Ceph RBD Image

Ceph RBD images can be used as NFS exports or SAMBA share to share the Ceph resources between Linux clients. This is reliable as well as easy to setup. You need to create an RBD image and map it to the client.



Steps to create RBD Image and map to Linux client

- [Ref Section - Creating a Ceph RBD Image](#) for steps to create an image, which can be used as an NFS export or SAMBA share.

The keyring file will look as shown below : (actual content will vary from key to key)

```
[client.testuser]
```

```
key = AQBsvJFX0K5nDxAAHZb2RYXEFs7ZVIPDGmi7A==
```

- Depending on distro being used, you need to use yum / apt-get to download / install package ceph-common which will give us Ceph client tools.

- RHEL/CentOS # yum install ceph-common
- Ubuntu/Debian # apt-get install ceph-common

- Load RBD client module.

```
# modprobe rbd
```

Verify if the module has successfully loaded before proceeding further.

```
# lsmod | grep rbd
```

The output should be similar as shown below.

```
rbd                57093    2
```

```
libceph            194776    1 rbd
```

- List the mapped images on client machine
- Copy the ceph.conf and user-id keyring (filename – ceph.client.<user-id>.keyring)from UVS manager WebUI console to /etc/ceph/ directory on Client system.

```
# rbd ls <MON_IP_Address>:6789 --pool <pool-name> --id <User-ID> --keyring <absolute path of
```

keyring>

- Map the pool image to Ceph Client system

```
# rbd map -m <MON_IP_Address>:6789 --pool <pool-name> --id <User-ID> --keyring <absolute path of
```

keyring> <image name>

- Run command lsblk to verify the mount. You should be able to see the new Block device added.
lsblk
- Once the Ceph RBD block image is mapped, we can make a filesystem and mount the same as the local filesystem on Linux client.
mkfs -t ext4 /dev/rbd0
(/dev/rbd0, /dev/rbd1 etc will change depending on number of RBD images mapped)
#mkdir /myshare
#mount /dev/rbd0 /myshare

This prepares the ground for NFS or SAMBA to export the /myshare folder from Linux client. (The share name will change as per mount point used).

Making RBD Image mount persistent

NOTE: The previous mount command will create a temporary mount, which will not persist across reboot. For making it a permanent mount, we need to follow some additional steps -

We need to edit /etc/ceph/rbdmap and add RBD map details in the following format -

```
#poolname/imagename    id=client-name,keyring=<User keyring path>
```

To make the folder mount on mapped RBD image, you need to make entry to /etc/fstab file, as shown below and save the /etc/fstab file.

```
#vi /etc/fstab
```

```
/dev/rbd0 /myshare ext4 _netdev 0 0
```

Steps to create NFS Server

A) RHEL/CentOS

- i. Install nfs-utils package.

```
#yum install nfs-utils
```

- ii. edit /etc/exports file and add
/myshare *(rw, sync)
- iii. Save the file. Run `exportfs -r` to load the file.
- i. Enable and start the NFS service. (`chkconfig nfs-server on` and `service nfs-server start` / `systemctl enable nfs-server` && `systemctl start nfs-server`)

B) Debian / Ubuntu

- i. Install `nfs-kernel-server` package.
`#apt-get install nfs-kernel-server`
 - ii. Edit /etc/exports file and add
/myshare *(rw, sync)
 - iii. Save the file. Run `exportfs -r` to load the file.
- ```
exportfs -r
```
- iv. Enable and start the NFS service.

## Steps to create SAMBA Server

### A) RHEL/CentOS

- i. Install `samba` & `samba-client` package.

```
#yum install samba samba-client
```

- ii. Edit `/etc/samba/smb.conf` file and add share information at the end of the file as shown below -

```
[mysambashare]
path = /myshare
browsable = yes
writable = yes
guest ok = yes
read only = no
```

- iii. Save the file.
- i. Enable and start the SAMBA service. (`chkconfig smb on` and `service smb start` / `systemctl enable smb` && `systemctl start smb`)

### B) Debian / Ubuntu

Steps for RHEL / CentOS can be followed on Debian / Ubuntu to install SAMBA and export the share to clients.

```
/etc/init.d/smbd start
```

 can be used to start the `smbd` service.

## RBD-NBD – MAP RBD IMAGES TO NBD DEVICE

The classical methods to map an RBD image are mapping by rbd kernel module and mapping by librbd. Since the Ceph Jewel version, Ceph has supported the mapping by librbd using the NBD module. It's a highly efficient way for mapping an RBD image and it supports the most RBD features. If you're required to use RBD features, we recommend mapping the images by RBD-NBD.

### Install RBD-NBD Package & Module

Depending on the distro being used, you need to install the package rbd-nbd which will give us rbd-nbd tools.

- RHEL/CentOS # yum install rbd-nbd
- Ubuntu/Debian # apt-get install rbd-nbd

Besides the rbd-nbd package, you have to load the NBD kernel module. Unfortunately, the NBD kernel module isn't installed on everyone's server. Please check the NBD module in your server and try to compile the module if the server didn't preload it.

```
find /lib/modules/$(uname -r) -type f -name '*.ko' | grep nbd
/lib/modules/3.10.0-1062.18.1.el7.x86_64/kernel/drivers/block/nbd.ko
modprobe nbd
lsmod | grep nbd
nbd 17555 0
```

### Copy Ceph configurations & keyrings

The ceph.conf & ceph.keyring put on the Mars400 monitor nodes. You need to use SCP copying these two files to your servers.

```
scp ${mars400-mon-node}:/etc/ceph/ceph.* /etc/ceph
```

### Map the RBD Image

Map the RBD image via rbd-nbd is very similar using the rbd kernel mapping. A simple “rbd nbd map” command helps you mapping the RBD image.

```
rbd ls -p mirroring
image001
rbd nbd map mirroring/image001
/dev/nbd0
mkfs.xfs /dev/nbd0
mount /dev/nbd0 /mnt
df -h /mnt/
Filesystem Size Used Avail Use% Mounted on
/dev/nbd0 1.0T 33M 1.0T 1% /mnt
umount /mnt
rbd nbd unmap /dev/nbd0
```

We imitated the service rbdmap and created the nbdmap for mapping the rbd-nbd device automatically. The nbdmap service will map the list and mount it while booting. Once you map the rbd image and format it manually



for the first time, you do not need to worry about mapping the images every time. Please check the URL below to get more information.

<https://ambedded.freshdesk.com/a/solutions/articles/47001139253>

## Appendixes

### Appendix 1. Configure the network before deploying a Ceph cluster

Please visit [Ambedded Freshdesk](#) to download the Mars500 Deployment Guide.

### Appendix 2. Prepare the External RGW Hosts for Ceph Octopus

Before you can deploy Ceph RADOS Gateway for version 16.2.x (Pacific) by UVS Manager, please follow the following steps to prepare the necessary software packages on your x86 hosts.

1. Prepare a x86 VM or physical server with Ubuntu 18.04.5 installed.
2. download the packages file and upload to external host

[https://drive.google.com/file/d/1PZSxV9Q1kR9e5re63WINPbm1u-SK0v\\_K/view?usp=sharing](https://drive.google.com/file/d/1PZSxV9Q1kR9e5re63WINPbm1u-SK0v_K/view?usp=sharing)

3. Extract the tar file

```
tar -pxf /tmp/ext-host.tar.gz -C /opt
```

You will find 9 packages inside the pkgs folder.

4. Install the first 4 packages.

```
cd /opt/pkgs/<pkg name>
dpkg -i --force-all *deb
```

5. Run docker load -i tar file to import docker images

```
mv /opt/pkgs/4.docker-tarball /var/lib/docker/amb-image
cd /var/lib/docker/amb-image
docker load -i ceph.tar.gz
```

6. Run cephadm check-host to check if the host is ok to use.

```
cephadm check-host
rm -rf /opt/pkg
```

7. copy ceph tarball to mars400

```
scp /var/lib/docker/amb-image/ceph.tar.gz \
root@mars400 IP:/var/lib/docker/ceph-mars400-x86.tar.gz
```

8. add x86 container image to registry server

```
registry_ceph_x86="<registry IP>:<registry port>/ceph:amd64-v16"
docker load -i ./ceph-mars400-x86.tar.gz
docker tag ceph/ceph:amd64-v16 "${registry_ceph_x86}"
docker push "${registry_ceph_x86}"
```

9. Create manifest to combine the different arch

```
registry_target="<registry ip>:<registry port>"
docker manifest create \
"${registry_target}/ceph:v16" \
"${registry_target}/ceph:arm64-v16" \
"${registry_target}/ceph:amd64-v16"
docker manifest annotate \
"${registry_target}/ceph:v16" \
"${registry_target}/ceph:arm64-v16" \
--arch=arm64 --os linux
docker manifest annotate \
"${registry_target}/ceph:v16" \
"${registry_target}/ceph:amd64-v16" --arch=amd64 --os linux

push will return the important sha for ceph container image
docker manifest push \
--insecure --purge "${registry_target}/ceph:v16"
sha256:e6b4aca93f1b7127398933a3d2e26187b8b86e9fcb8ce08fa38d98fcb4aa3ac1
```

10. Setup new CEPH container ID to the cluster

```
target_path="<registry IP>:<registry Port>"
ceph config set global container_image
"${target_path}/ceph@sha256:e6b4aca93f1b7127398933a3d2e26187b8b86e9fcb8ce08fa38d98fcb4aa3ac1"
```

Tips: SHA only shows once after manifest push. If something is broken while pushing new container image, we need to remove the old registry server and push both arm and arm64 into registry again.

11. Copy mars400 daemon.json into the external host

```
scp root@<mars400 ip>:/etc/docker/daemon.json /etc/docker/daemon.json
systemctl restart docker
```

12. Pull registry ceph and run cephadm version to check if the return is correct

```
target="<registry ip>:<registry port>/ceph:v16"
docker pull "${target}"
cephadm veresion
Using recent ceph image
10.240.82.228:5000/ceph@sha256:e6b4aca93f1b7127398933a3d2e26187b8b86e9fcb8ce08fa38d98fcb4aa3ac1
ceph version 16.2.5 (0883bdea7337b95e4b611c768c0279868462204a) pacific (stable)
```