

# Global-Active Device Quorum on AWS Cloud

**iSCSI Target Using AWS Virtual Machine**

Hitachi Vantara

April 2022

# Table of Contents

<b>Notices and Disclaimer .....</b>	<b>2</b>
<b>About This Guide .....</b>	<b>3</b>
Intended Audience .....	3
Document Revisions .....	3
References .....	3
Contributors .....	3
Comments .....	3
<b>Configuration and Specifications .....</b>	<b>4</b>
Introduction .....	4
AWS Virtual Machine .....	5
<b>Amazon Virtual Machine Instance .....</b>	<b>6</b>
Deployment .....	6
Remote Access .....	12
Storage Repository .....	14
Firewall Exemption .....	17
<b>Targetcli .....</b>	<b>19</b>
Installation .....	19
Configuration .....	22
<b>Global-Active Device Quorums .....</b>	<b>26</b>
Create iSCSI Paths .....	26
Discover External Volumes .....	28
Define Global-Active Device Quorums .....	31

## Notices and Disclaimer

© 2022 Hitachi Vantara LLC. All rights reserved.

No part of this publication may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying and recording, or stored in a database or retrieval system for commercial purposes without the express written permission of Hitachi, Ltd., or Hitachi Vantara (collectively, "Hitachi"). Licensee may make copies of the Materials provided that any such copy is: (i) created as an essential step in utilization of the Software as licensed and is used in no other manner; or (ii) used for archival purposes. Licensee may not make any other copies of the Materials. "Materials" mean text, data, photographs, graphics, audio, video and documents.

Hitachi reserves the right to make changes to this Material at any time without notice and assumes no responsibility for its use. The Materials contain the most current information available at the time of publication.

Some of the features described in the Materials might not be currently available. Refer to the most recent product announcement for information about feature and product availability, or contact Hitachi Vantara at [https://support.HitachiVantara.com/en\\_us/contact-us.html](https://support.HitachiVantara.com/en_us/contact-us.html).

Notice: Hitachi products and services can be ordered only under the terms and conditions of the applicable Hitachi agreements. The use of Hitachi products is governed by the terms of your agreements with Hitachi Vantara.

By using this software, you agree that you are responsible for:

- 1) Acquiring the relevant consents as may be required under local privacy laws or otherwise from authorized employees and other individuals to access relevant data; and
- 2) Verifying that data continues to be held, retrieved, deleted, or otherwise processed in accordance with relevant laws.

Notice on Export Controls: The technical data and technology inherent in this Document may be subject to U.S. export control laws, including the U.S. Export Administration Act and its associated regulations, and may be subject to export or import regulations in other countries. Reader agrees to comply strictly with all such regulations and acknowledges that Reader has the responsibility to obtain licenses to export, re-export, or import the Document and any Compliant Products.

EXPORT CONTROLS: Licensee will comply fully with all applicable export laws and regulations of the United States and other countries, and Licensee shall not export, or allow the export or re-export of, the Software, API, or Materials in violation of any such laws or regulations. By downloading or using the Software, API, or Materials, Licensee agrees to the foregoing and represents and warrants that Licensee is not located in, under the control of, or a national or resident of any embargoed or restricted country.

Hitachi is a registered trademark of Hitachi, Ltd., in the United States and other countries.

AIX, AS/400e, DB2, Domino, DS6000, DS8000, Enterprise Storage Server, eServer, FICON, Flash Copy, IBM, Lotus, MVS, OS/390, PowerPC, RS6000, S/390, System z9, System z10, Tivoli, z/OS, z9, z10, z13, z/VM, BCPii™ and z/VSE are registered trademarks or trademarks of International Business Machines Corporation.

Active Directory, ActiveX, Bing, Excel, Hyper-V, Internet Explorer, the Internet Explorer logo, Microsoft, the Microsoft Corporate Logo, MS-DOS, Outlook, PowerPoint, SharePoint, Silverlight, SmartScreen, SQL Server, Visual Basic, Visual C++, Visual Studio, Windows, the Windows logo, Windows Azure, Windows PowerShell, Windows Server, the Windows start button, and Windows Vista are registered trademarks or trademarks of Microsoft Corporation. Microsoft product screenshots are reprinted with permission from Microsoft Corporation.

All other trademarks, service marks, and company names in this document or web site are properties of their respective owners.

**Global-Active Device Quorum on AWS Cloud**

**About This Guide**

This guide provides instructions for deploying a virtual machine in the Amazon Web Services (AWS) cloud and configuring it as an iSCSI target.



After the original publication of this whitepaper, Hitachi Vantara released a virtual machine image on Amazon Marketplace that automates the process of configuring targetcli. The solution is available for free on the Amazon Marketplace. However, you must pay for various AWS fees relating to running the virtual machine. The solution can be found at <https://aws.amazon.com/marketplace/pp/prodview-7yn64ltekjhjs>.

**Intended Audience**

This document is intended for Hitachi Vantara and Hitachi partner representatives who need a foundation of knowledge on this product to best represent it to potential buyers.

**Document Revisions**

Revision Number	Date	Details
1.0	November 2019	Initial release.
1.1	April 2022	Added recommendation to remove public IP address on the AWS virtual machine and added GAD Cloud Quorum solution. Also made wording corrections and formatting updates.

**References**

- [Hitachi Global-Active Device User Guide](#)

**Contributors**

The information included in this document represents the expertise, feedback, and suggestions of a number of skilled practitioners. The author (Dang Luong) wants to recognize and thank the following contributors and reviewers of this document (listed alphabetically by last name):

- Tom Attanese - Product Management
- Paul Romero - Global Product & Solutions Enablement

**Comments**

Please send your comments on this document to [gpse.replicationsoftware@hitachivantara.com](mailto:gpse.replicationsoftware@hitachivantara.com). Include the document title and number, including the revision level, and refer to specific sections and paragraphs whenever possible. All comments become the property of Hitachi Vantara. Thank You!

## Configuration and Specifications

### Introduction

This guide provides instructions for deploying a virtual machine in the Amazon Web Services (AWS) cloud and configuring it as an iSCSI target. We will use the Linux package “targetcli” to create and manage block devices on the virtual machine. The objective is to leverage volumes from the iSCSI target virtual machine running on AWS as quorum volumes for Global-active device (GAD).



Only use volumes from an iSCSI target virtual machine for global-active device quorums. Do not use them as data volumes.



This guide does not include instructions for establishing a VPN connection to AWS. Refer to the AWS documentation, such as [AWS Site-to-Site VPN](#).

Figure 1 illustrates the test environment. The on-premise datacenter is connected to the AWS cloud using a VPN tunnel. Network traffic is passed between the on-premise storage systems and the iSCSI target virtual machine in AWS using the VPN tunnel.

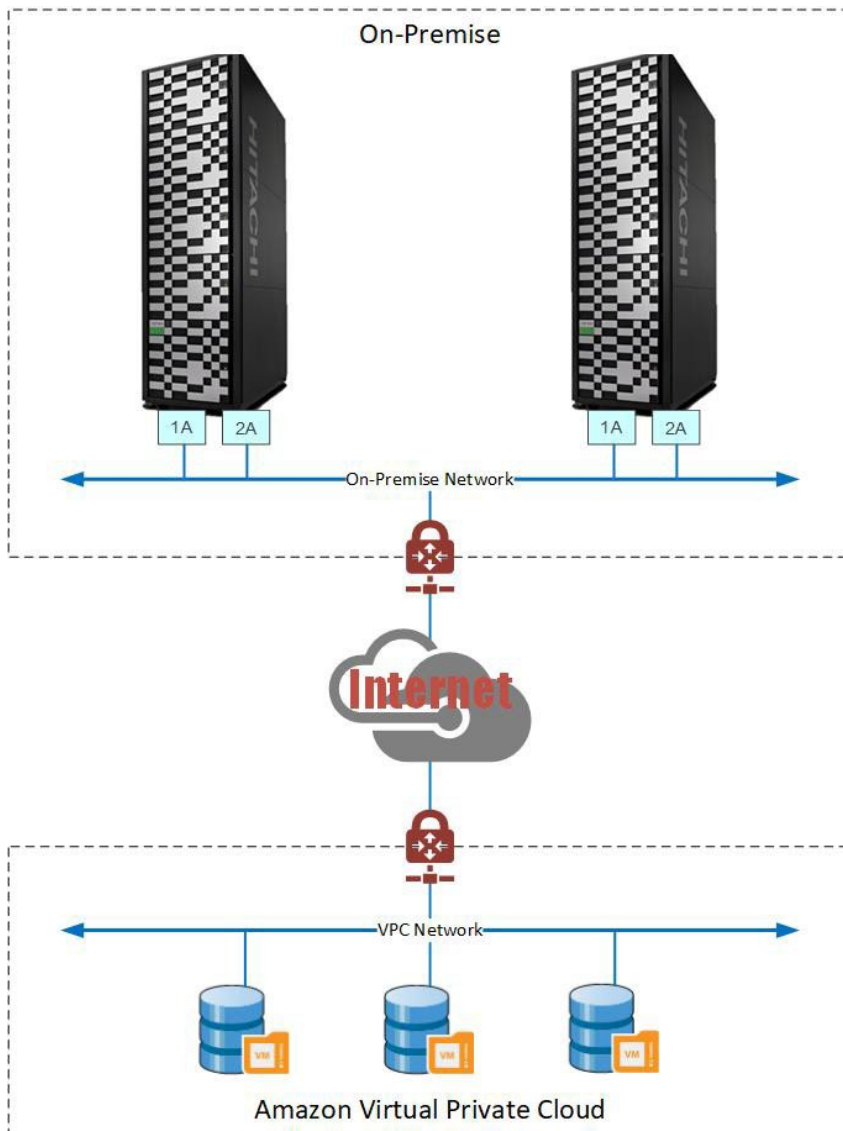


Figure 1. Test Environment

## Global-Active Device Quorum on AWS Cloud

### AWS Virtual Machine

The following settings were used for the virtual machine image:

- Operating system: Amazon Linux 2
- Kernel: 4.14.123-111.109.amzn2.x86\_64
- Instance type: t2.nano
  - CPU: Intel Xeon CPU E5-2676 v3 @ 2.40 GHz
  - Memory: 512 MB
- Targetcli version: targetcli-2.1.fb46-6.amzn2.noarch

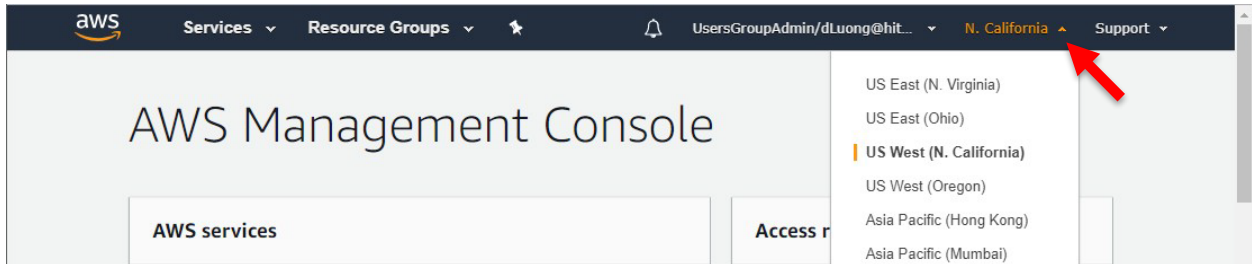
Global-Active Device Quorum on AWS Cloud

**Amazon Virtual Machine Instance**

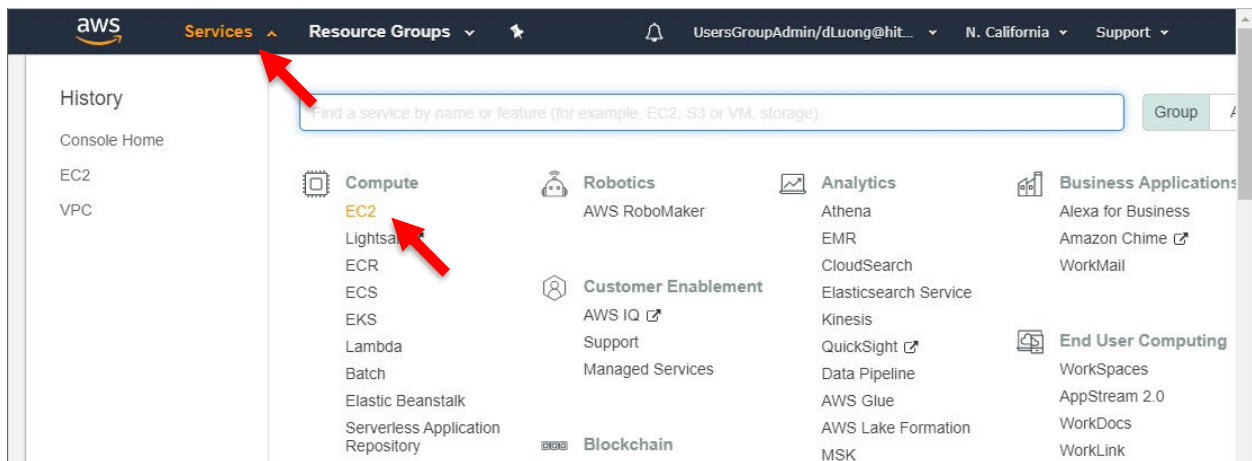
**Deployment**

This section provides instructions for deploying the virtual machine using an Amazon Machine Instance.

1. In the AWS Management Console, use the top-right shortcut to expand the **Region** list and select a region.



2. On the top left, select **Services > Compute > EC2**.

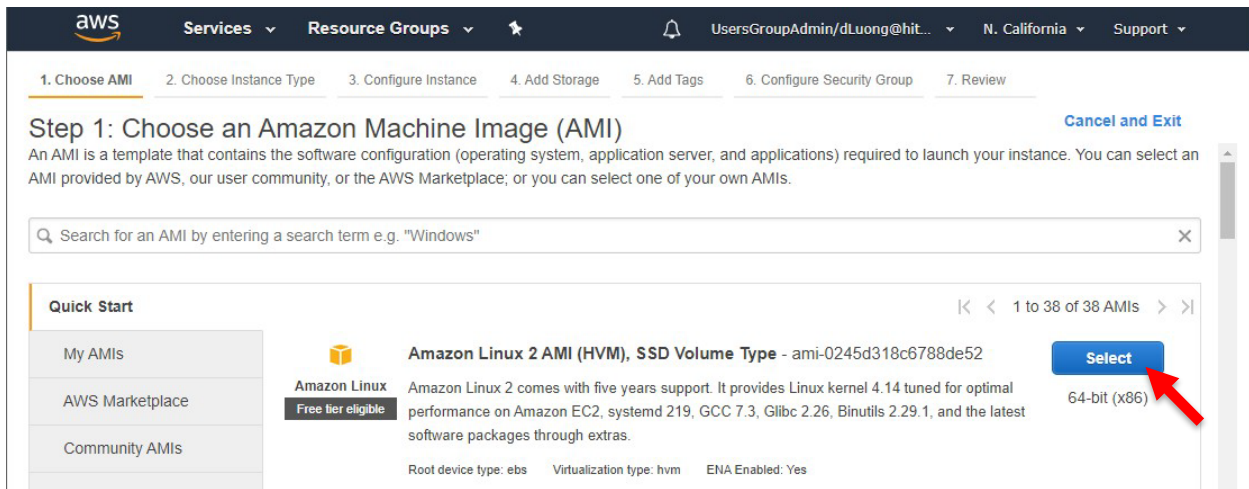


3. Click **Launch Instance**.

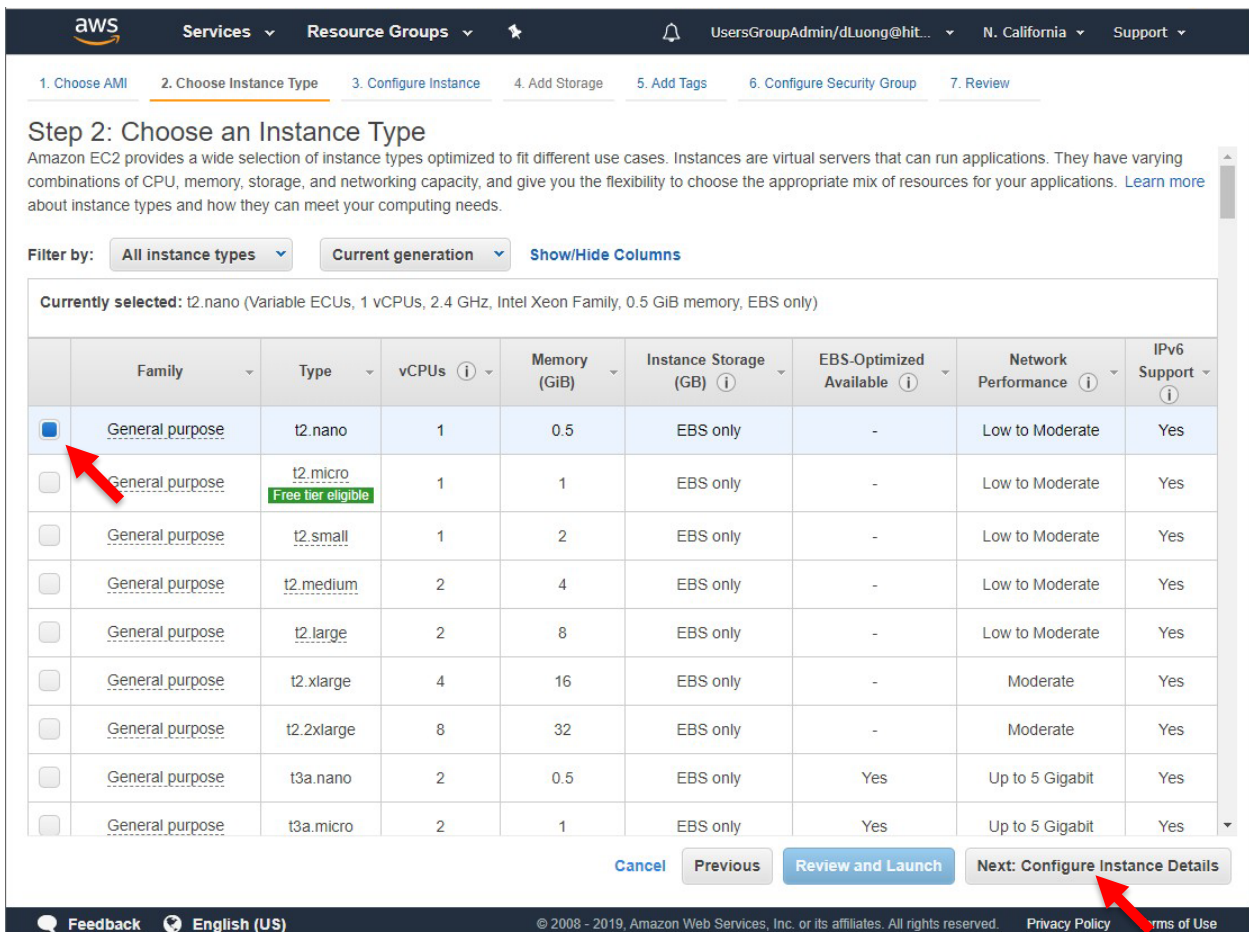


**Global-Active Device Quorum on AWS Cloud**

4. Locate **Amazon Linux 2 AMI** and click **Select**.



5. Select the instance type (we tested with the t2.nano type) and click **Next: Configure Instance Details**.



6. From the **Network** dropdown list, select a network. For the initial configuration, we enabled the **Auto-assign Public IP** option to remotely access the virtual machine and download targetcli packages. Click **Next: Add Storage**.



Global-Active Device Quorum on AWS Cloud

aws Services Resource Groups UsersGroupAdmin/dLuong@hit... N. California Support

1. Choose AMI 2. Choose Instance Type 3. Configure Instance 4. Add Storage 5. Add Tags 6. Configure Security Group 7. Review

### Step 3: Configure Instance Details

Configure the instance to suit your requirements. You can launch multiple instances from the same AMI, request Spot instances to take advantage of the lower pricing, assign an access management role to the instance, and more.

Number of instances: 1 [Launch into Auto Scaling Group](#)

Purchasing option:  Request Spot instances

Network: vpc-075201acbaecd164c [Create new VPC](#)  
No default VPC found. [Create a new default VPC.](#)

Subnet: subnet-0ab744f22f29e89a8 | us-west-1a [Create new subnet](#)  
65530 IP Addresses available

Auto-assign Public IP: Enable

Placement group:  Add instance to placement group

Capacity Reservation: Open [Create new Capacity Reservation](#)

IAM role: None [Create new IAM role](#)

Shutdown behavior: Stop

Enable termination protection:  Protect against accidental termination

Monitoring:  Enable CloudWatch detailed monitoring  
[Additional charges apply.](#)

Tenancy: Shared - Run a shared hardware instance  
[Additional charges will apply for dedicated tenancy.](#)

[Cancel](#) [Previous](#) [Review and Launch](#) [Next: Add Storage](#)

Feedback English (US) © 2008 - 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved. [Privacy Policy](#) [Terms of Use](#)

After successfully setting up the virtual machine as an iSCSI target, you must secure the solution by removing the public IP.

7. Click **Add New Volume.**

The new volume will provide backend storage for the quorums.

aws Services Resource Groups UsersGroupAdmin/dLuong@hit... N. California Support

1. Choose AMI 2. Choose Instance Type 3. Configure Instance 4. Add Storage 5. Add Tags 6. Configure Security Group 7. Review

### Step 4: Add Storage

Your instance will be launched with the following storage device settings. You can attach additional EBS volumes and instance store volumes to your instance, or edit the settings of the root volume. You can also attach additional EBS volumes after launching an instance, but not instance store volumes. [Learn more about storage options in Amazon EC2.](#)

Volume Type	Device	Snapshot	Size (GiB)	Volume Type	IOPS	Throughput (MB/s)	Delete on Termination	Encryption
Root	/dev/xvda	snap-06a692b9f21a43448	8	General Purpose S	100 / 3000	N/A	<input checked="" type="checkbox"/>	Not Encrypted

[Add New Volume](#)

## Global-Active Device Quorum on AWS Cloud

8. Enter the capacity for the new volume and click **Review and Launch**.

aws Services Resource Groups UsersGroupAdmin/dLuong@hit... N. California Support

1. Choose AMI 2. Choose Instance Type 3. Configure Instance 4. Add Storage 5. Add Tags 6. Configure Security Group 7. Review

### Step 4: Add Storage

Your instance will be launched with the following storage device settings. You can attach additional EBS volumes and instance store volumes to your instance, or edit the settings of the root volume. You can also attach additional EBS volumes after launching an instance, but not instance store volumes. [Learn more](#) about storage options in Amazon EC2.

Volume Type	Device	Snapshot	Size (GiB)	Volume Type	IOPS	Throughput (MB/s)	Delete on Termination	Encryption
Root	/dev/xvda	snap-06a692b9f21a43448	8	General Purpose S	100 / 3000	N/A	<input checked="" type="checkbox"/>	Not Encrypte
EBS	/dev/sdb	Search (case-insensit	100	General Purpose S	null	N/A	<input type="checkbox"/>	

**Add New Volume**

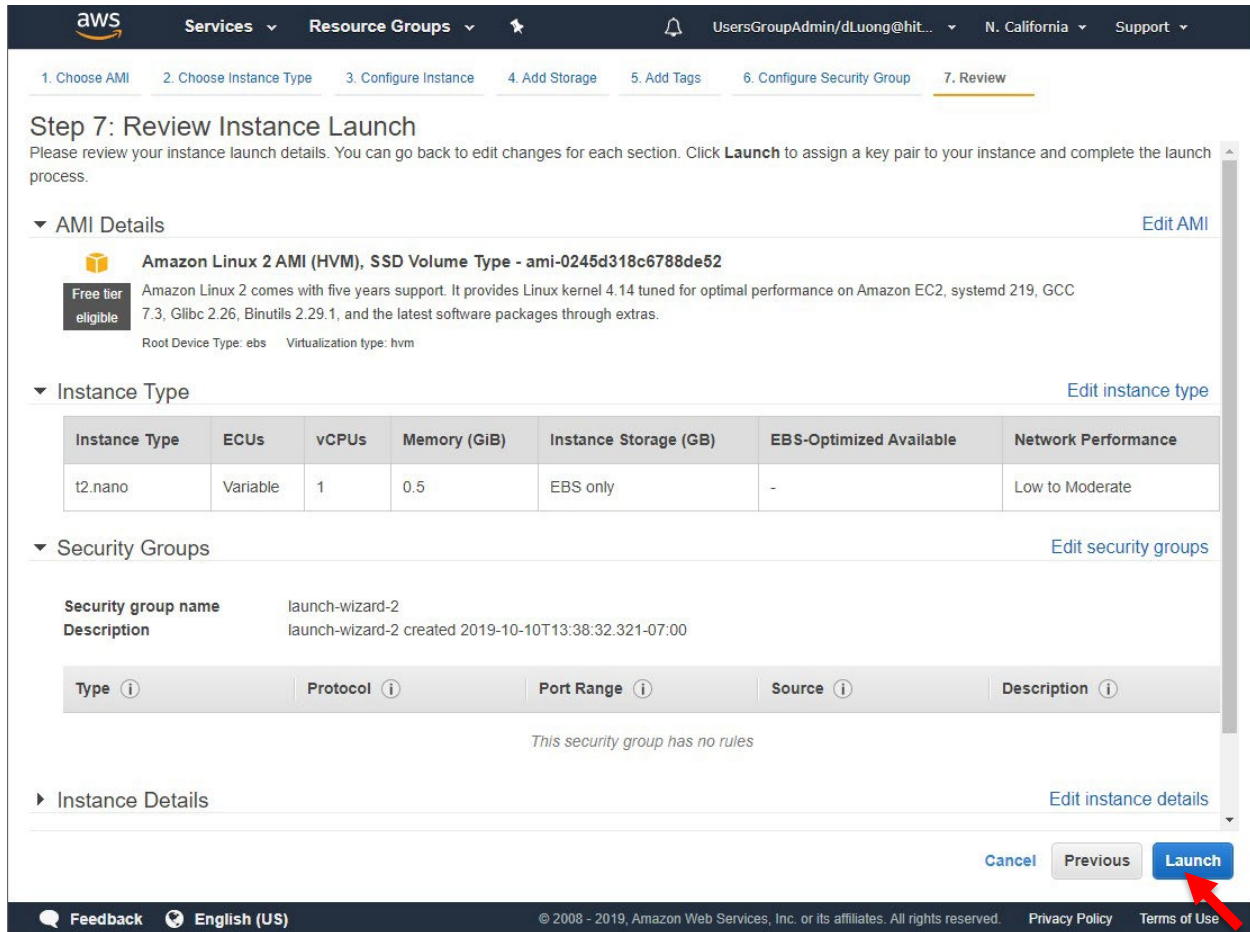
Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage. [Learn more](#) about free usage tier eligibility and usage restrictions.

Cancel Previous **Review and Launch** Next: Add Tags

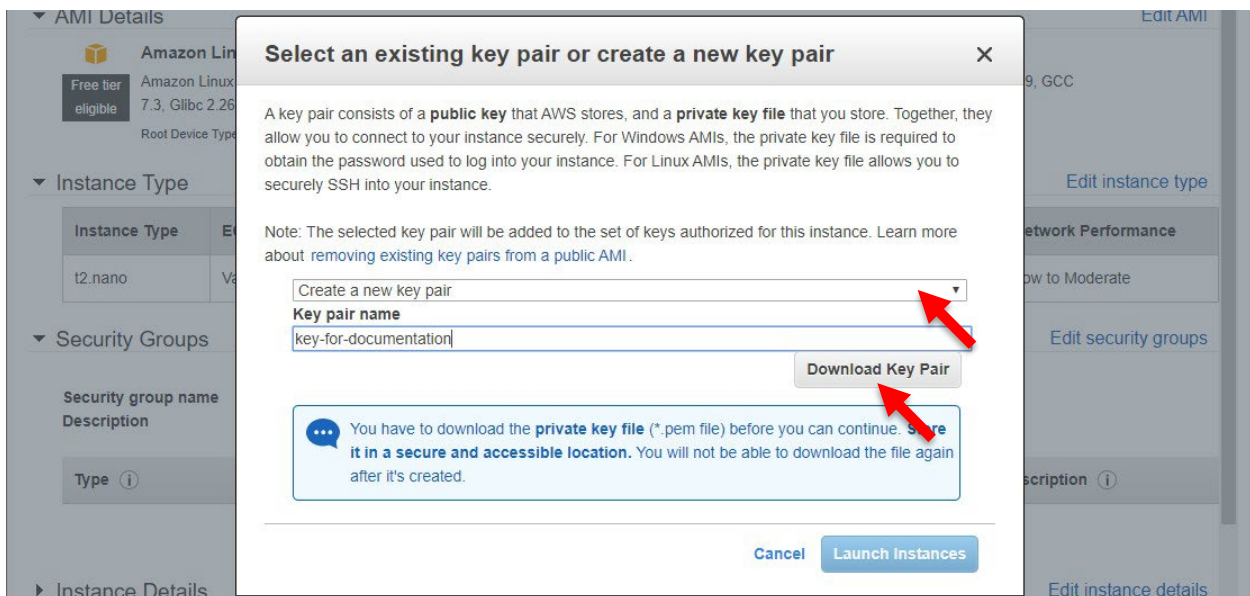
Feedback English (US) © 2008 - 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

**Global-Active Device Quorum on AWS Cloud**

9. Verify the details and click **Launch**.



10. If you do not have an existing key pair or do not want to use an existing key pair, use the dropdown list to select **Create a new key pair**. Enter a name for the pair and click **Download Key Pair**.



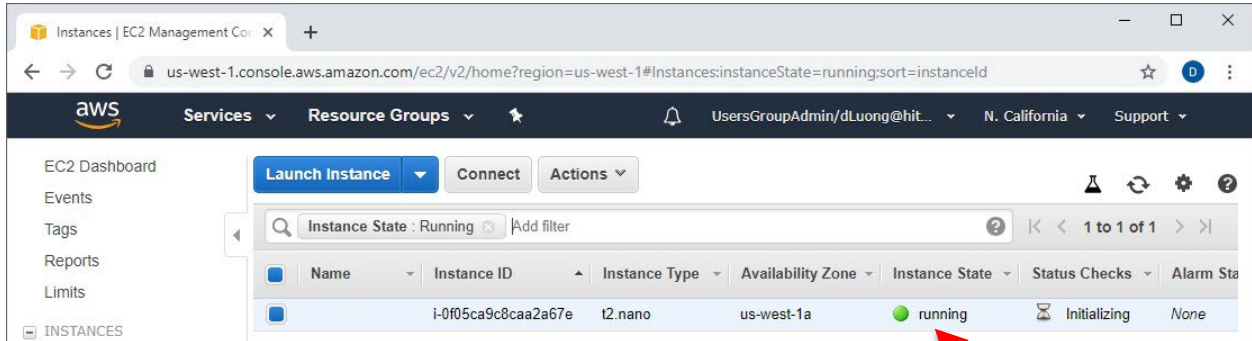
## Global-Active Device Quorum on AWS Cloud

11. Ensure that you download the pem file to your local machine and can locate it. Click **Launch Instances**.



You can convert the pem file to ppk format, which can be used by PuTTY. For instructions to convert, see: <https://tecadmin.net/convert-pem-to-ppk-via-putty/>

12. Verify that the **Instance State** of the new instance is running on the **Instances** screen. The new instance must be online and accessible.

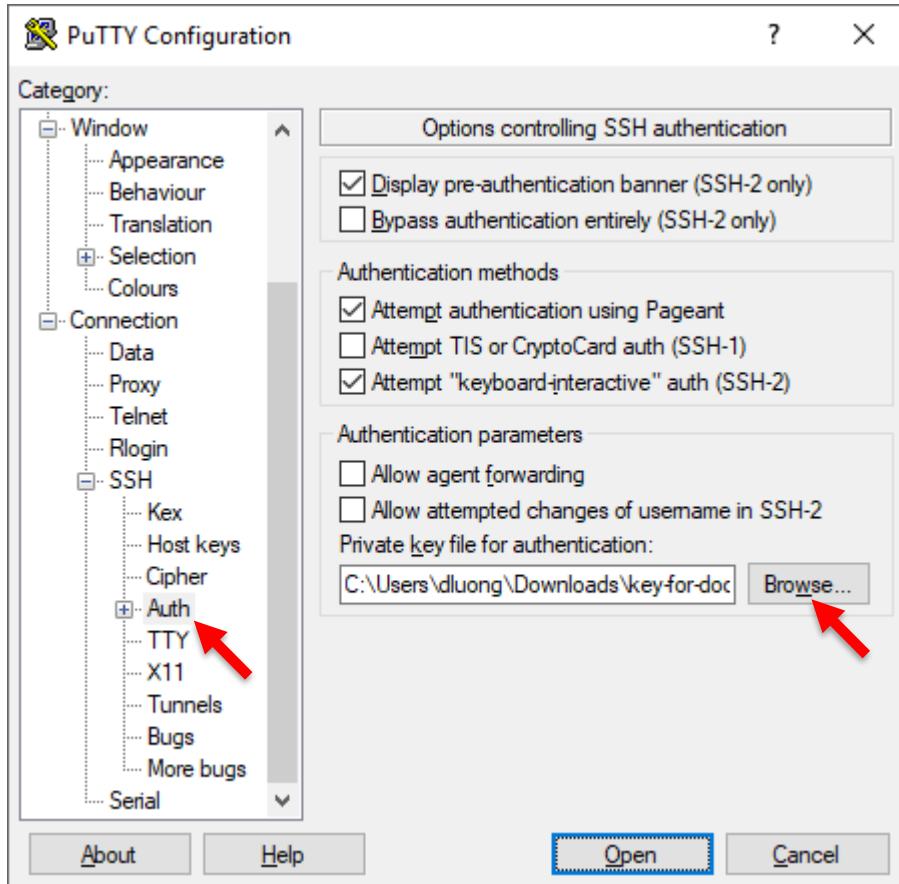


## Global-Active Device Quorum on AWS Cloud

### Remote Access

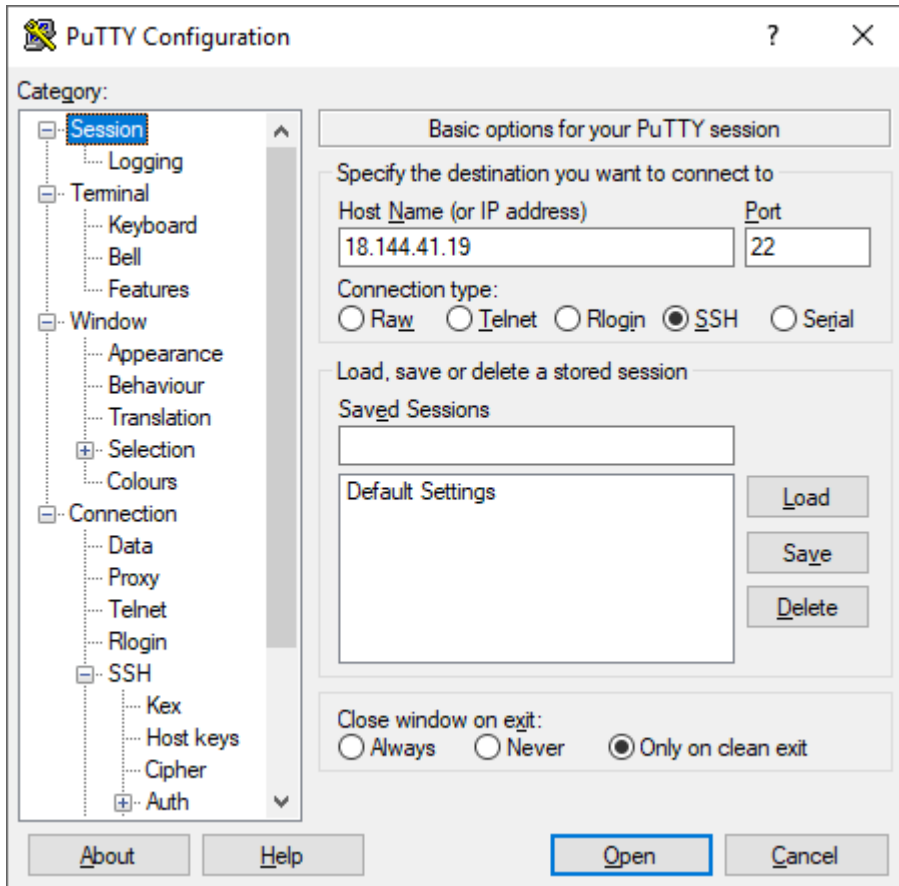
This section provides instructions for remotely accessing the new virtual machine using PuTTY.

1. On the PuTTY Configuration window, under **Category**, select **Connection > SSH > Auth**. Click **Browse** to locate the ppk file.

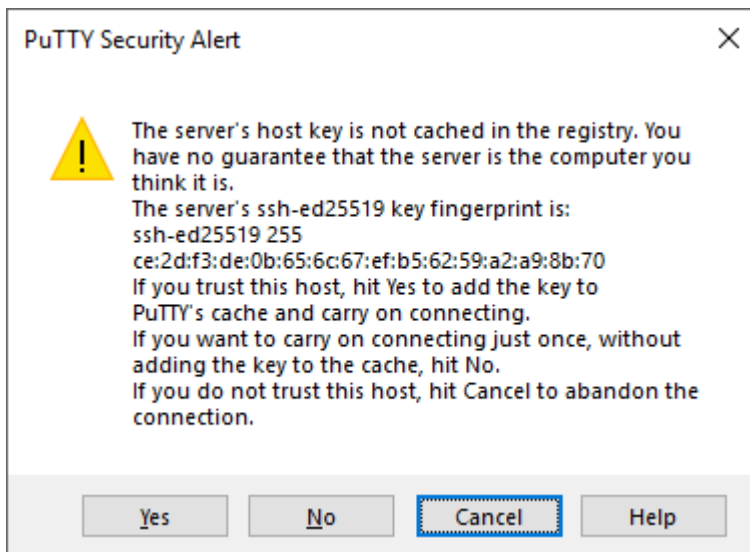


**Global-Active Device Quorum on AWS Cloud**

- Under **Category**, select **Session**, and enter the **IP address**. Under **Connection type**, select **SSH** and then click **Open**.



- To accept the host key, click **Yes**.



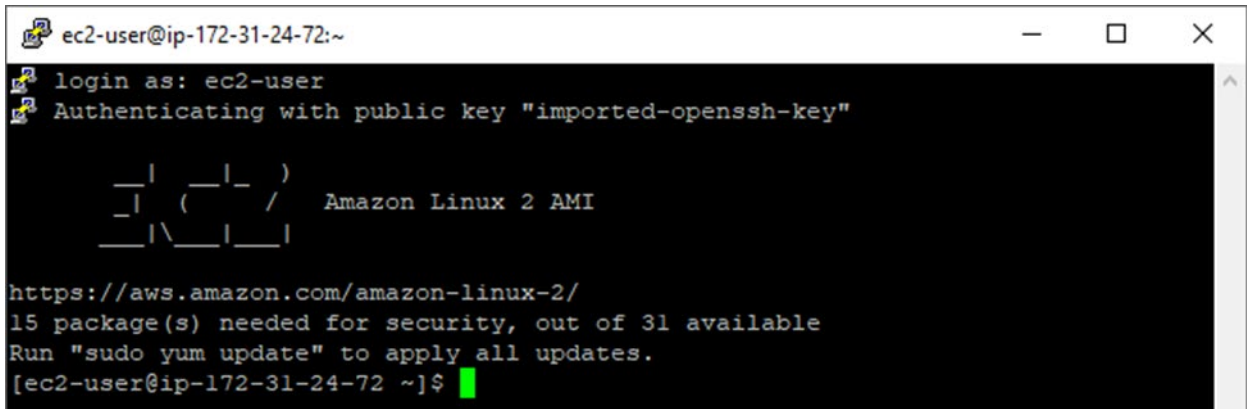
## Global-Active Device Quorum on AWS Cloud

- For the login name, enter: **ec2-user**.



```
18.144.41.19 - PuTTY
login as: ec2-user
```

The authentication is completed with the public key.



```
ec2-user@ip-172-31-24-72:~
login as: ec2-user
Authenticating with public key "imported-openssh-key"

  _ | _ | _ )
  _ | ( _ | _ /   Amazon Linux 2 AMI
  _ | \ _ | _ |

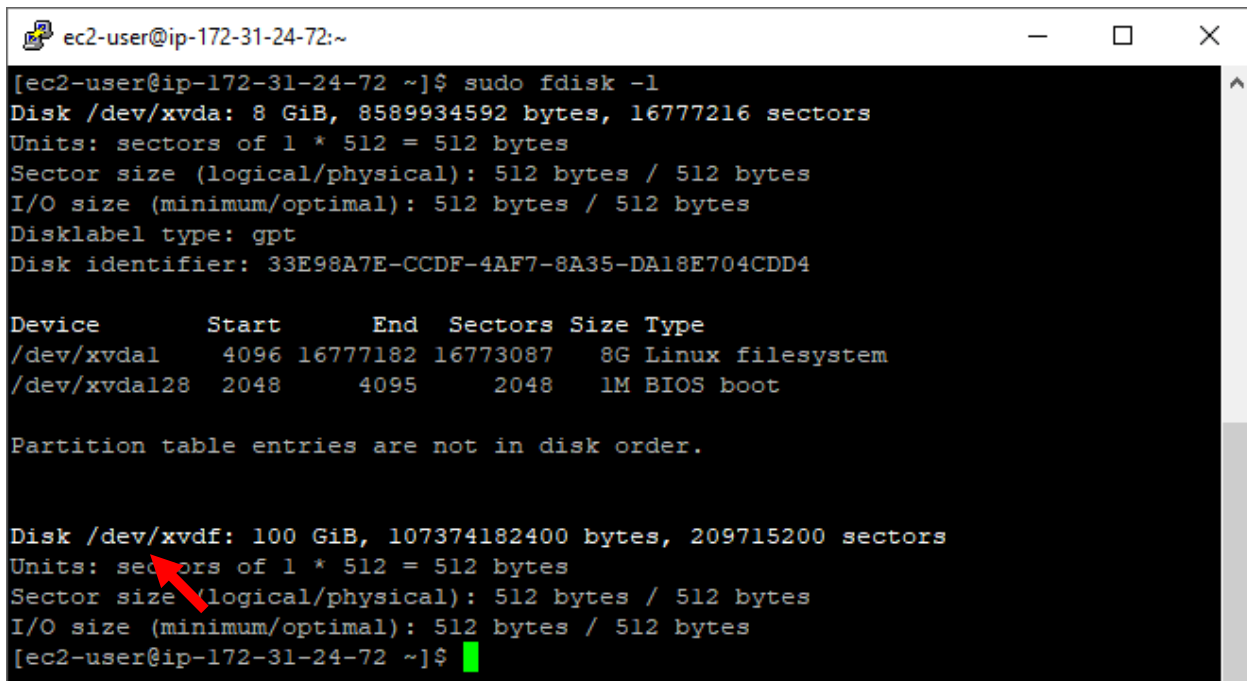
https://aws.amazon.com/amazon-linux-2/
15 package(s) needed for security, out of 31 available
Run "sudo yum update" to apply all updates.
[ec2-user@ip-172-31-24-72 ~]$
```

## Storage Repository

This section provides instructions for creating a storage repository for storing block devices that will be presented from the virtual machine.

- Verify that the second volume attached to the virtual machine exists by running the following command:

```
sudo fdisk -l
```



```
ec2-user@ip-172-31-24-72:~
[ec2-user@ip-172-31-24-72 ~]$ sudo fdisk -l
Disk /dev/xvda: 8 GiB, 8589934592 bytes, 16777216 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: gpt
Disk identifier: 33E98A7E-CCDF-4AF7-8A35-DA18E704CDD4

Device            Start      End  Sectors  Size Type
/dev/xvda1        4096 16777182 16773087   8G Linux filesystem
/dev/xvda128      2048     4095     2048    1M BIOS boot

Partition table entries are not in disk order.

Disk /dev/xvdf: 100 GiB, 107374182400 bytes, 209715200 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
[ec2-user@ip-172-31-24-72 ~]$
```

## Global-Active Device Quorum on AWS Cloud

2. Create a partition on the volume by running the following command:

```
sudo fdisk /dev/xvdf
```

3. Create a partition that fills up the entire volume:

- a. On the fdisk main menu, enter: **n**
- b. For Partition type, enter: **p**
- c. For Partition number, enter: **1**
- d. To accept default of 2048 for the first sector, press **Enter**.
- e. To accept default of max for the last sector, press **Enter**.

```
ec2-user@ip-172-31-24-72:~
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
[ec2-user@ip-172-31-24-72 ~]$ sudo fdisk /dev/xvdf

Welcome to fdisk (util-linux 2.30.2).
Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.

Device does not contain a recognized partition table.
Created a new DOS disklabel with disk identifier 0x9a9fbbf3.

Command (m for help): n
Partition type
   p   primary (0 primary, 0 extended, 4 free)
   e   extended (container for logical partitions)
Select (default p): p
Partition number (1-4, default 1): 1
First sector (2048-209715199, default 2048):
Last sector, +sectors or +size{K,M,G,T,P} (2048-209715199, default 209715199):

Created a new partition 1 of type 'Linux' and of size 100 GiB.

Command (m for help): █
```

4. To verify the new partition, enter **p**.

```
Command (m for help): p
Disk /dev/xvdf: 100 GiB, 107374182400 bytes, 209715200 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0x9a9fbbf3

Device      Boot Start      End  Sectors  Size Id Type
/dev/xvdf1  2048 209715199 209713152 100G 83 Linux

Command (m for help): █
```



**Global-Active Device Quorum on AWS Cloud**

- To write changes and close fdisk, enter **w**.

```
Command (m for help): w
The partition table has been altered.
Calling ioctl() to re-read partition table.
Syncing disks.

[ec2-user@ip-172-31-24-72 ~]$ █
```

- To create a volume group on top of the new partition, run the following command:

```
sudo vgcreate VG_quorums /dev/xvdf1
```

```
ec2-user@ip-172-31-24-72:~
[ec2-user@ip-172-31-24-72 ~]$ sudo vgcreate VG_quorums /dev/xvdf1
Physical volume "/dev/xvdf1" successfully created.
Volume group "VG_quorums" successfully created
[ec2-user@ip-172-31-24-72 ~]$ █
```

- Within the new volume group, to create a logical volume that spans 100% of the volume group, run the following command:

```
sudo lvcreate -l 100 VG_quorums
```

```
ec2-user@ip-172-31-24-72:~
[ec2-user@ip-172-31-24-72 ~]$ sudo lvcreate -l 100 VG_quorums
Logical volume "lv010" created.
[ec2-user@ip-172-31-24-72 ~]$ █
```

- To create an XFS file system on top of the logical volume, run the following command:

```
sudo mkfs.xfs /dev/VG_quorums/lvol0
```

```
ec2-user@ip-172-31-24-72:~
[ec2-user@ip-172-31-24-72 ~]$ sudo mkfs.xfs /dev/VG_quorums/lvol0
meta-data=/dev/VG_quorums/lvol0 isize=512    agcount=4, agsize=25600 blks
          =                       sectsz=512   attr=2, projid32bit=1
          =                       crc=1        finobt=1, sparse=0
data      =                       bsize=4096 blocks=102400, imaxpct=25
          =                       sunit=0      swidth=0 blks
naming    =version 2               bsize=4096 ascii-ci=0 ftype=1
log       =internal log           bsize=4096 blocks=855, version=2
          =                       sectsz=512   sunit=0 blks, lazy-count=1
realtime  =none                   extsz=4096 blocks=0, rtextents=0
[ec2-user@ip-172-31-24-72 ~]$ █
```

- To make a mount point, run the following command:

```
sudo mkdir /quorums
```

- To mount file system automatically during a reboot, add the following line in the `/etc/fstab` file:

```
/dev/VG_quorums/lvol0 /quorums          xfs          defaults    0          0
```

## Global-Active Device Quorum on AWS Cloud

11. Verify that the fstab addition works by running the following command:

```
sudo mount /quorums
df
```

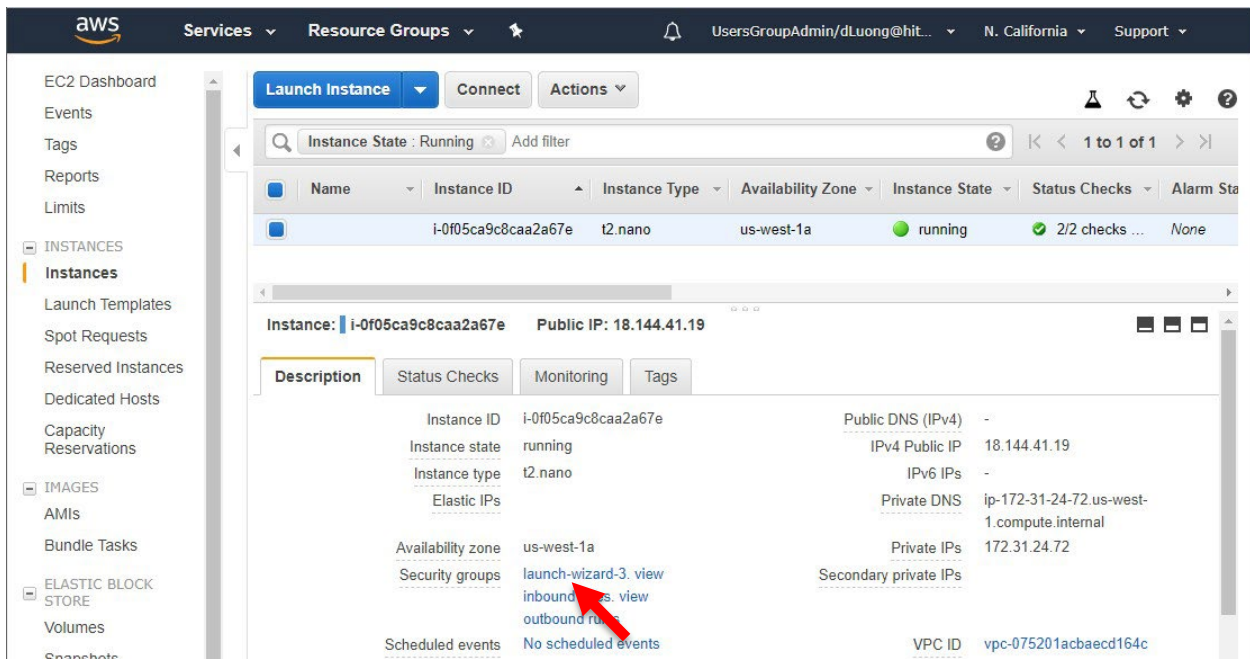
```

[ec2-user@ip-172-31-24-72 ~]$ sudo mount /quorums
[ec2-user@ip-172-31-24-72 ~]$ df
Filesystem            1K-blocks    Used Available Use% Mounted on
devtmpfs              227596         0   227596  0% /dev
tmpfs                 245624         0   245624  0% /dev/shm
tmpfs                 245624        400   245224  1% /run
tmpfs                 245624         0   245624  0% /sys/fs/cgroup
/dev/xvda1            8376300 1264724  7111576 16% /
tmpfs                 49128          0    49128  0% /run/user/1000
/dev/mapper/VG_quorums-lvol10 406180    21136  385044  6% /quorums
[ec2-user@ip-172-31-24-72 ~]$
  
```

## Firewall Exemption

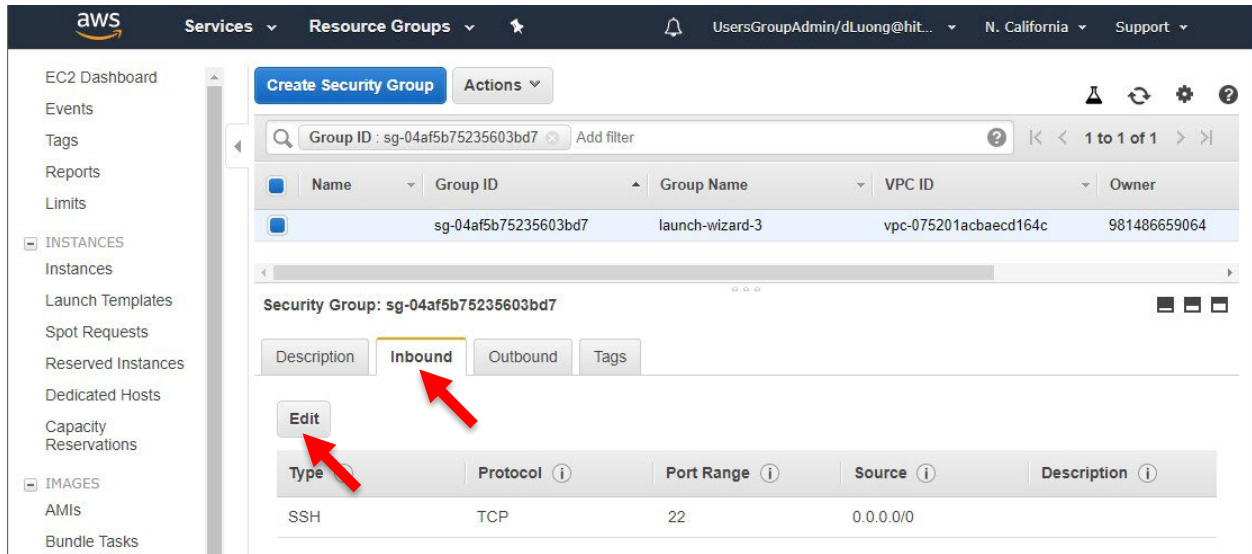
This section provides instructions for creating a firewall exemption so TCP traffic on port 3260 can enter the Virtual Private Cloud (VPC) where the virtual machine resides. Port 3260 is the default port used for iSCSI.

1. On the **Instances** page, select the virtual machine and click the security group attached to the instance.



**Global-Active Device Quorum on AWS Cloud**

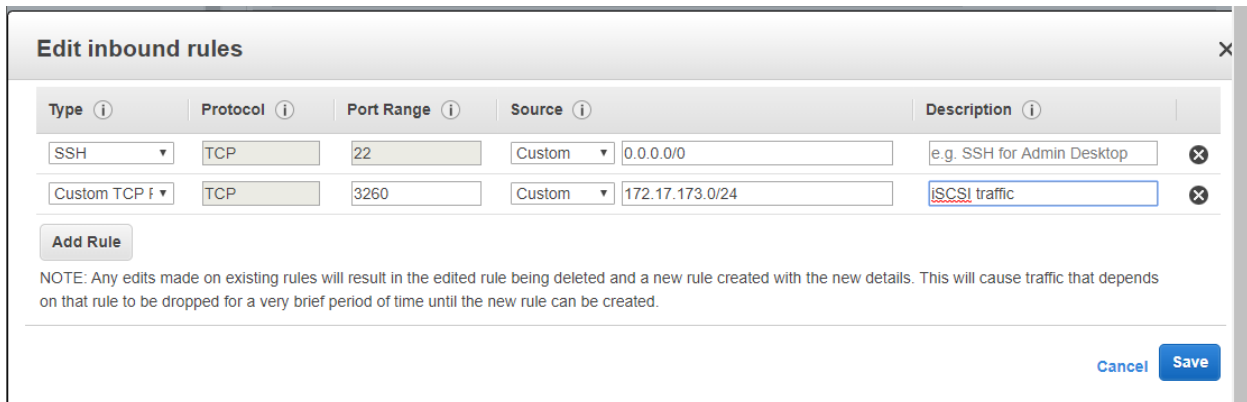
2. Select the **Inbound** tab and then click **Edit**.



3. Click **Add Rule**.

4. Set the new rule.

- For **Type**, select **Custom TCP Rule**.
- For **Port Range**, type: **3260**
- For **Source**, select **Custom**, and then enter the subnet of the storage system iSCSI ports.
- For **Description**, type: **iSCSI traffic**



5. Click **Save**. You do not need to create an outbound rule for TCP 3260.

## Targetcli

### Installation

This section provides instructions for installing targetcli on the virtual machine.

1. To install targetcli, run the following command:

```
sudo yum install -y targetcli
```

The following shows the output:

```
[ec2-user@ip-172-31-24-72 ~]$ sudo yum install -y targetcli
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Resolving Dependencies
--> Running transaction check
---> Package targetcli.noarch 0:2.1.fb46-6.amzn2 will be installed
--> Processing Dependency: python-rtslib >= 2.1.fb41 for package: targetcli-2.1.fb46-6.amzn2.noarch
--> Processing Dependency: python-ethtool for package: targetcli-2.1.fb46-6.amzn2.noarch
--> Processing Dependency: python-configshell for package: targetcli-2.1.fb46-6.amzn2.noarch
--> Running transaction check
---> Package python-configshell.noarch 1:1.1.fb23-4.amzn2 will be installed
--> Processing Dependency: python-urwid for package: 1:python-configshell-1.1.fb23-4.amzn2.noarch
--> Processing Dependency: pyparsing for package: 1:python-configshell-1.1.fb23-4.amzn2.noarch
---> Package python-ethtool.x86_64 0:0.8-5.amzn2.0.2 will be installed
--> Processing Dependency: libnl.so.1()(64bit) for package: python-ethtool-0.8-5.amzn2.0.2.x86_64
---> Package python-rtslib.noarch 0:2.1.fb63-12.amzn2 will be installed
--> Processing Dependency: python-pyudev for package: python-rtslib-2.1.fb63-12.amzn2.noarch
--> Processing Dependency: python-kmod for package: python-rtslib-2.1.fb63-12.amzn2.noarch
--> Running transaction check
---> Package libnl.x86_64 0:1.1.4-3.amzn2.0.2 will be installed
---> Package pyparsing.noarch 0:1.5.6-9.amzn2 will be installed
---> Package python-kmod.x86_64 0:0.9-4.amzn2.0.2 will be installed
---> Package python-pyudev.noarch 0:0.15-9.amzn2 will be installed
---> Package python-urwid.x86_64 0:1.1.1-3.amzn2.0.2 will be installed
--> Finished Dependency Resolution
```

Dependencies Resolved

```
=====
```

Package	Arch	Version	Repository	Size
Installing:				
targetcli	noarch	2.1.fb46-6.amzn2	amzn2-core	67 k
Installing for dependencies:				
libnl	x86_64	1.1.4-3.amzn2.0.2	amzn2-core	129 k
pyparsing	noarch	1.5.6-9.amzn2	amzn2-core	94 k
python-configshell	noarch	1:1.1.fb23-4.amzn2	amzn2-core	68 k
python-ethtool	x86_64	0.8-5.amzn2.0.2	amzn2-core	33 k
python-kmod	x86_64	0.9-4.amzn2.0.2	amzn2-core	74 k
python-pyudev	noarch	0.15-9.amzn2	amzn2-core	55 k
python-rtslib	noarch	2.1.fb63-12.amzn2	amzn2-core	100 k
python-urwid	x86_64	1.1.1-3.amzn2.0.2	amzn2-core	654 k

Transaction Summary

```
=====
```

Install 1 Package (+8 Dependent packages)

Total download size: 1.2 M

**Global-Active Device Quorum on AWS Cloud**

```

Installed size: 5.3 M
Downloading packages:
(1/9): pyparsing-1.5.6-9.amzn2.noarch.rpm | 94 kB 00:00
(2/9): libnl-1.1.4-3.amzn2.0.2.x86_64.rpm | 129 kB 00:00
(3/9): python-configshell-1.1.fb23-4.amzn2.noarch.rpm | 68 kB 00:00
(4/9): python-ethtool-0.8-5.amzn2.0.2.x86_64.rpm | 33 kB 00:00
(5/9): python-pyudev-0.15-9.amzn2.noarch.rpm | 55 kB 00:00
(6/9): python-kmod-0.9-4.amzn2.0.2.x86_64.rpm | 74 kB 00:00
(7/9): python-rtslib-2.1.fb63-12.amzn2.noarch.rpm | 100 kB 00:00
(8/9): python-urwid-1.1.1-3.amzn2.0.2.x86_64.rpm | 654 kB 00:00
(9/9): targetcli-2.1.fb46-6.amzn2.noarch.rpm | 67 kB 00:00
-----
Total 3.5 MB/s | 1.2 MB 00:00
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Installing : python-pyudev-0.15-9.amzn2.noarch 1/9
  Installing : pyparsing-1.5.6-9.amzn2.noarch 2/9
  Installing : python-kmod-0.9-4.amzn2.0.2.x86_64 3/9
  Installing : python-rtslib-2.1.fb63-12.amzn2.noarch 4/9
  Installing : libnl-1.1.4-3.amzn2.0.2.x86_64 5/9
  Installing : python-ethtool-0.8-5.amzn2.0.2.x86_64 6/9
  Installing : python-urwid-1.1.1-3.amzn2.0.2.x86_64 7/9
  Installing : 1:python-configshell-1.1.fb23-4.amzn2.noarch 8/9
  Installing : targetcli-2.1.fb46-6.amzn2.noarch 9/9
  Verifying : 1:python-configshell-1.1.fb23-4.amzn2.noarch 1/9
  Verifying : python-ethtool-0.8-5.amzn2.0.2.x86_64 2/9
  Verifying : python-urwid-1.1.1-3.amzn2.0.2.x86_64 3/9
  Verifying : python-rtslib-2.1.fb63-12.amzn2.noarch 4/9
  Verifying : libnl-1.1.4-3.amzn2.0.2.x86_64 5/9
  Verifying : python-kmod-0.9-4.amzn2.0.2.x86_64 6/9
  Verifying : pyparsing-1.5.6-9.amzn2.noarch 7/9
  Verifying : python-pyudev-0.15-9.amzn2.noarch 8/9
  Verifying : targetcli-2.1.fb46-6.amzn2.noarch 9/9

Installed:
  targetcli.noarch 0:2.1.fb46-6.amzn2

Dependency Installed:
  libnl.x86_64 0:1.1.4-3.amzn2.0.2
  pyparsing.noarch 0:1.5.6-9.amzn2
  python-configshell.noarch 1:1.1.fb23-4.amzn2
  python-ethtool.x86_64 0:0.8-5.amzn2.0.2
  python-kmod.x86_64 0:0.9-4.amzn2.0.2
  python-pyudev.noarch 0:0.15-9.amzn2
  python-rtslib.noarch 0:2.1.fb63-12.amzn2
  python-urwid.x86_64 0:1.1.1-3.amzn2.0.2

```

Complete!

- To install an additional python package that is required by targetcli, run the following command:

```
sudo yum install -y python-dbus
```

The following shows the output:

```

[ec2-user@ip-172-31-24-72 ~]$ sudo yum install -y python-dbus
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Resolving Dependencies
--> Running transaction check
--> Package dbus-python.x86_64 0:1.1.1-9.amzn2.0.2 will be installed
--> Processing Dependency: libdbus-glib-1.so.2()(64bit) for package: dbus-python-1.1.1-9.amzn2.0.2.x86_64
--> Running transaction check

```

**Global-Active Device Quorum on AWS Cloud**

```
---> Package dbus-glib.x86_64 0:0.100-7.2.amzn2 will be installed
--> Finished Dependency Resolution
```

Dependencies Resolved

```
=====
```

Package	Arch	Version	Repository	Size
Installing:				
dbus-python	x86_64	1.1.1-9.amzn2.0.2	amzn2-core	206 k
Installing for dependencies:				
dbus-glib	x86_64	0.100-7.2.amzn2	amzn2-core	103 k

Transaction Summary

```
=====
```

Install 1 Package (+1 Dependent package)

Total download size: 309 k

Installed size: 1.1 M

Downloading packages:

```
(1/2): dbus-python-1.1.1-9.amzn2.0.2.x86_64.rpm | 206 kB 00:00
(2/2): dbus-glib-0.100-7.2.amzn2.x86_64.rpm | 103 kB 00:00
```

```
-----
```

Total 1.8 MB/s | 309 kB 00:00

Running transaction check

Running transaction test

Transaction test succeeded

Running transaction

```
Installing : dbus-glib-0.100-7.2.amzn2.x86_64 1/2
Installing : dbus-python-1.1.1-9.amzn2.0.2.x86_64 2/2
Verifying : dbus-python-1.1.1-9.amzn2.0.2.x86_64 1/2
Verifying : dbus-glib-0.100-7.2.amzn2.x86_64 2/2
```

Installed:

```
dbus-python.x86_64 0:1.1.1-9.amzn2.0.2
```

Dependency Installed:

```
dbus-glib.x86_64 0:0.100-7.2.amzn2
```

Complete!

3. To start the targetcli daemon, run the following command:

```
sudo systemctl start target
```

4. To verify that the daemon is running, run the following command:

```
sudo systemctl status target
```

```

ec2-user@ip-172-31-24-72:~
[ec2-user@ip-172-31-24-72 ~]$ sudo systemctl status target
● target.service - Restore LIO kernel target configuration
   Loaded: loaded (/usr/lib/systemd/system/target.service; enabled; vendor prese
t: disabled)
   Active: active (exited) since Thu 2019-10-10 23:45:19 UTC; 10s ago
   Process: 559 ExecStart=/usr/bin/targetctl restore (code=exited, status=0/SUCCE
SS)
   Main PID: 559 (code=exited, status=0/SUCCESS)

Oct 10 23:45:19 ip-172-31-24-72.us-west-1.compute.internal systemd[1]: Starti...
Oct 10 23:45:19 ip-172-31-24-72.us-west-1.compute.internal systemd[1]: Starte...
Hint: Some lines were ellipsized, use -l to show in full.
[ec2-user@ip-172-31-24-72 ~]$

```

- To set targetcli to start automatically after it restarts, run the following command:

```
sudo systemctl enable target
```

```

ec2-user@ip-172-31-24-72:~
[ec2-user@ip-172-31-24-72 ~]$ sudo systemctl enable target
Created symlink from /etc/systemd/system/multi-user.target.wants/target.service
to /usr/lib/systemd/system/target.service.
[ec2-user@ip-172-31-24-72 ~]$

```

## Configuration

This section provides instructions for configuring targetcli to serve three 13 GB volumes over iSCSI.

- Log in to targetcli by running the following command:

```
sudo targetcli
```

```

ec2-user@ip-172-31-24-72:~
[ec2-user@ip-172-31-24-72 ~]$ sudo targetcli
targetcli shell version 2.1.fb46
Copyright 2011-2013 by Datera, Inc and others.
For help on commands, type 'help'.

/>

```

- Create three 13 GB volumes in the /quorums folder as follows:

- Create volume 1 by running the following command:

```
backstores/fileio create volume1 /quorums/volume1 13G
```

The following shows the output:

```
Created fileio volume1 with size 13958643712
```

- Create volume 2 by running the following command:

```
backstores/fileio create volume2 /quorums/volume2 13G
```

The following shows the output:

```
Created fileio volume2 with size 13958643712
```

**Global-Active Device Quorum on AWS Cloud**

- c. Create volume 3 by running the following command:

```
backstores/fileio create volume3 /quorums/volume3 13G
```

The following shows the output:

```
Created fileio volume3 with size 13958643712
```

3. To create an iSCSI qualified name, run the following commands:

```
cd /iscsi
```

```
create
```

The following shows the output:

```
Created target iqn.2003-01.org.linux-iscsi.ip-172-31-24-72.x8664:sn.a375a63a681c. Created TPG 1.
```

```
Global pref auto_add_default_portal=true
```

```
Created default portal listening on all IPs (0.0.0.0), port 3260.
```

4. Change the listening IP address from all to one specific IP address.

- a. Change the directory by running the following command:

```
cd iqn.2003-01.org.linux-iscsi.ip- 172-31-24-72.x8664:sn.a375a63a681c/tpg1/portals/
```

- b. Delete listening on all IP addresses by running the following command:

```
delete 0.0.0.0 3260
```

The following shows the output:

```
Deleted network portal 0.0.0.0:3260
```

- c. Set up listening on one specific IP address by running the following command:

```
create 172.31.24.72 3260
```

The following shows the output:

```
Using default IP port 3260
```

```
Created network portal 172.31.24.72:3260.
```

5. Map the volumes that you created earlier.

- a. Change the directory by running the following command:

```
cd /iscsi/iqn.2003- 01.org.linux-iscsi.ip-172-31-24-72.x8664:sn.a375a63a681c/tpg1/luns
```

- b. Map the first LUN by running the following command:

```
create /backstores/fileio/volume1
```

The following shows the output:

```
Created LUN 0.
```

- c. Map the second LUN by running the following command:

```
create /backstores/fileio/volume2
```

The following shows the output:

```
Created LUN 1.
```

- d. Map the third LUN by running the following command:

```
create /backstores/fileio/volume3/
```

The following shows the output:

```
Created LUN 2.
```

6. Mask the initiator IQNs of the storage systems to allow access to the LUNs. This adds four IQNs: two ports from each storage system.



**Global-Active Device Quorum on AWS Cloud**

- a. Change the directory by running the following command:

```
cd /iscsi/iqn.2003-01.org.linux-iscsi.ip-172-31-24-72.x8664:sn.a375a63a681c/tpg1/acls
```

- b. Create the first IQN by running the following command (your IQN will be different):

```
create iqn.1994-04.jp.co.hitachi:rsd.h8m.i.123ac6.1a
```

The following shows an example of the output:

```
Created Node ACL for iqn.1994-04.jp.co.hitachi:rsd.h8m.i.123ac6.1a
Created mapped LUN 2.
Created mapped LUN 1.
Created mapped LUN 0.
```

- c. Create the second IQN by running the following command (your IQN will be different):

```
create iqn.1994-04.jp.co.hitachi:rsd.h8m.i.123ac6.2a
```

The following shows an example of the output:

```
Created Node ACL for iqn.1994-04.jp.co.hitachi:rsd.h8m.i.123ac6.2a
Created mapped LUN 2.
Created mapped LUN 1.
Created mapped LUN 0.
```

- d. Create the third IQN by running the following command (your IQN will be different):

```
create iqn.1994-04.jp.co.hitachi:rsd.h8m.i.12afcd.1a
```

The following shows an example of the output:

```
Created Node ACL for iqn.1994-04.jp.co.hitachi:rsd.h8m.i.12afcd.1a
Created mapped LUN 2.
Created mapped LUN 1.
Created mapped LUN 0.
```

- e. Create the fourth IQN by running the following command (your IQN will be different):

```
create iqn.1994-04.jp.co.hitachi:rsd.h8m.i.12afcd.2a
```

The following shows an example of the output:

```
Created Node ACL for iqn.1994-04.jp.co.hitachi:rsd.h8m.i.12afcd.2a
Created mapped LUN 2.
Created mapped LUN 1.
Created mapped LUN 0.
```

- 7. To save the changes, run the following command:

```
cd /
saveconfig
```

The following shows the output:

```
Last 10 configs saved in /etc/target/backup/. Configuration saved to
/etc/target/saveconfig.json
```

- 8. To view the completed configuration, run the following command:

```
ls
```

The following is an example of the output (your URL for the LUNs will be different):

```
o- / ..... [..]
o- backstores ..... [..]
  | o- block ..... [Storage Objects: 0]
  | o- fileio ..... [Storage Objects: 3]
  | | o- volume1 ..... [/quorums/volume1 (13.0GiB) write-back activated]
  | | o- alua ..... [ALUA Groups: 1]
  | | | o- default_tg_pt_gp ..... [ALUA state: Active/optimized]
  | | o- volume2 ..... [/quorums/volume2 (13.0GiB) write-back activated]
```

**Global-Active Device Quorum on AWS Cloud**

```

| | | o- alua ..... [ALUA Groups: 1]
| | |   o- default_tg_pt_gp ..... [ALUA state: Active/optimized]
| | o- volume3 ..... [/quorums/volume3 (13.0GiB) write-back activated]
| |   o- alua ..... [ALUA Groups: 1]
| |     o- default_tg_pt_gp ..... [ALUA state: Active/optimized]
| o- pscsi ..... [Storage Objects: 0]
| o- ramdisk ..... [Storage Objects: 0]
o- iscsi ..... [Targets: 1]
| o- iqn.2003-01.org.linux-iscsi.ip-172-31-24-72.x8664:sn.a375a63a681c [TPGs: 1]
|   o- tpg1 ..... [no-gen-acls, no-auth]
|     o- acls ..... [ACLs: 4]
|       o- iqn.1994-04.jp.co.hitachi:rsd.h8m.i.123ac6.1a .... [Mapped LUNs: 3]
|         | o- mapped_lun0 ..... [lun0 fileio/volume1 (rw)]
|         | o- mapped_lun1 ..... [lun1 fileio/volume2 (rw)]
|         | o- mapped_lun2 ..... [lun2 fileio/volume3 (rw)]
|       o- iqn.1994-04.jp.co.hitachi:rsd.h8m.i.123ac6.2a .... [Mapped LUNs: 3]
|         | o- mapped_lun0 ..... [lun0 fileio/volume1 (rw)]
|         | o- mapped_lun1 ..... [lun1 fileio/volume2 (rw)]
|         | o- mapped_lun2 ..... [lun2 fileio/volume3 (rw)]
|       o- iqn.1994-04.jp.co.hitachi:rsd.h8m.i.12afcd.1a .... [Mapped LUNs: 3]
|         | o- mapped_lun0 ..... [lun0 fileio/volume1 (rw)]
|         | o- mapped_lun1 ..... [lun1 fileio/volume2 (rw)]
|         | o- mapped_lun2 ..... [lun2 fileio/volume3 (rw)]
|       o- iqn.1994-04.jp.co.hitachi:rsd.h8m.i.12afcd.2a .... [Mapped LUNs: 3]
|         o- mapped_lun0 ..... [lun0 fileio/volume1 (rw)]
|         o- mapped_lun1 ..... [lun1 fileio/volume2 (rw)]
|         o- mapped_lun2 ..... [lun2 fileio/volume3 (rw)]
|     o- luns ..... [LUNs: 3]
|       o- lun0 ..... [fileio/volume1 (/quorums/volume1) (default_tg_pt_gp)]
|       o- lun1 ..... [fileio/volume2 (/quorums/volume2) (default_tg_pt_gp)]
|       o- lun2 ..... [fileio/volume3 (/quorums/volume3) (default_tg_pt_gp)]
|     o- portals ..... [Portals: 1]
|       o- 172.31.24.72:3260 ..... [OK]
o- loopback ..... [Targets: 0]
/>

```



After successfully setting up the virtual machine as an iSCSI target, you must secure the solution by removing the public IP.

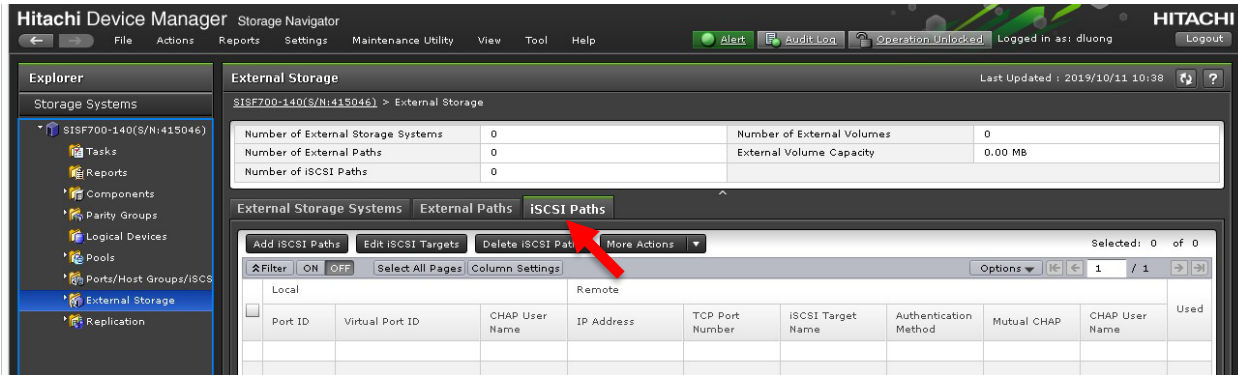
## Global-Active Device Quorum on AWS Cloud

### Global-Active Device Quorums

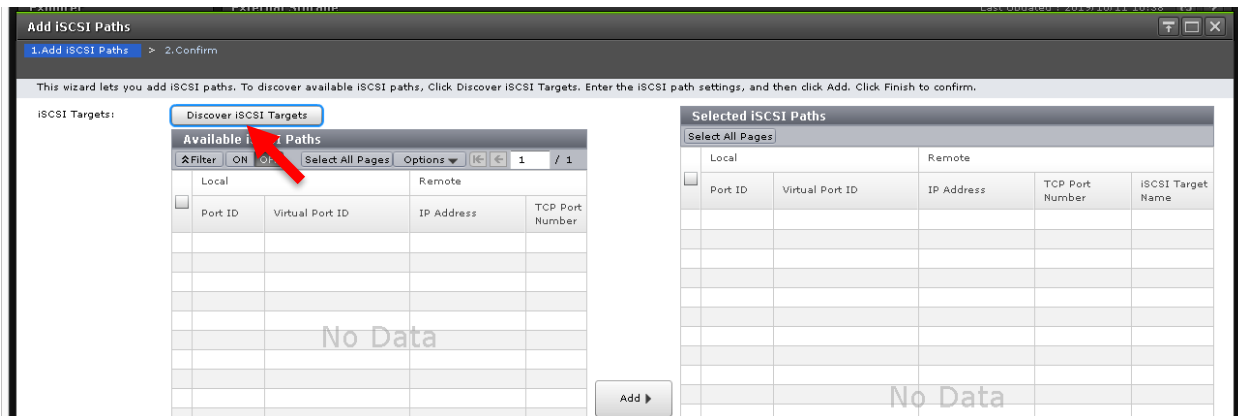
This section describes how to discover the volumes from the iSCSI target virtual machine and turn them into GAD quorums. The procedure is the same as it is to virtualize a physical Fibre Channel or iSCSI storage system.

#### Create iSCSI Paths

1. Log in to Storage Navigator.
2. On the left side, select **External Storage**, and then select the **iSCSI Paths** tab.



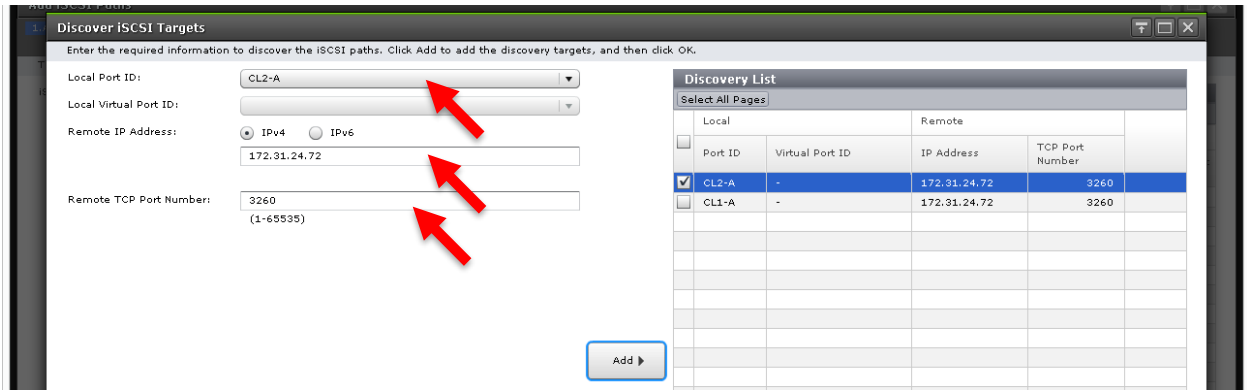
3. Click **Add iSCSI Paths**.
4. Click **Discover iSCSI Targets**.



5. Add both iSCSI paths. Repeat this step for both paths.
  - a. Select the storage port from the **Local Port ID** list.
  - b. Enter the private IP address of the AWS virtual machine.
  - c. For **Remote TCP Port Number**, enter **3260**.

**Global-Active Device Quorum on AWS Cloud**

d. Click **Add**.



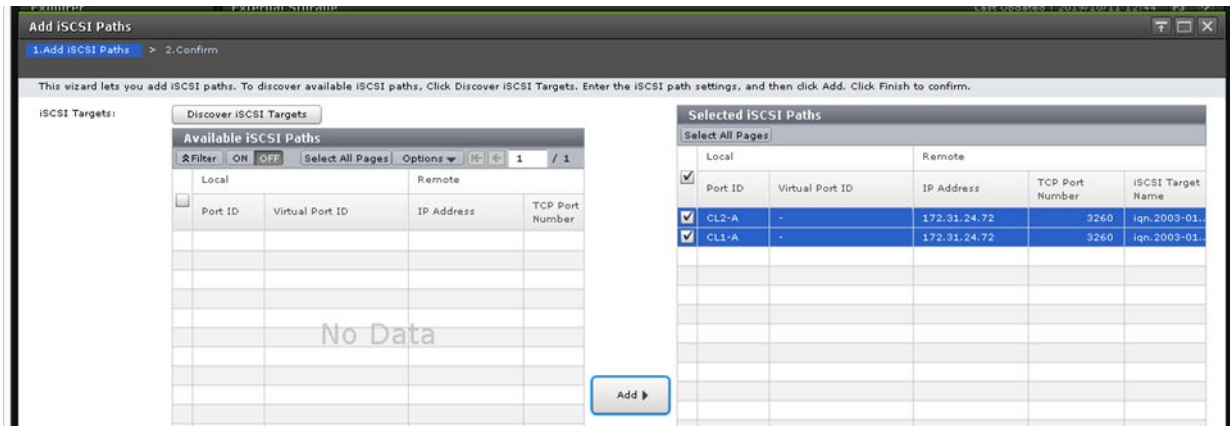
6. After creating both iSCSI paths, click **OK**.

7. On the **Add iSCSI Paths** window, set the following:

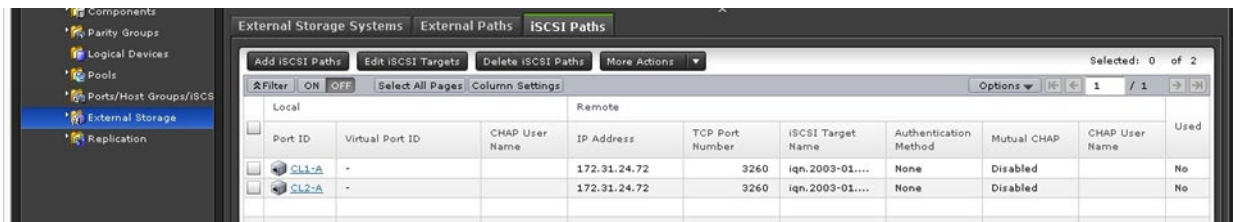
- a. From the **Authentication Method** dropdown list, click **None**.
- b. For **Mutual CHAP**, click **Disable**.



8. Click **Add** and then click **Finish**.



The following shows the created paths:

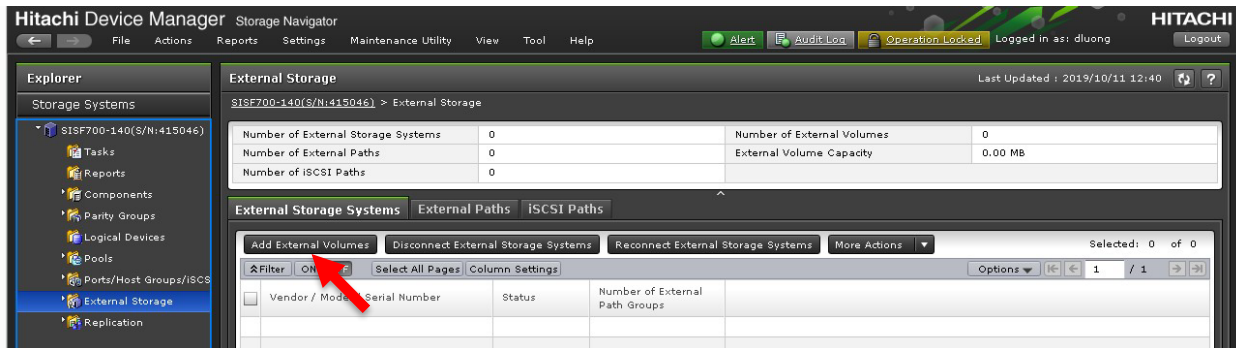


## Global-Active Device Quorum on AWS Cloud

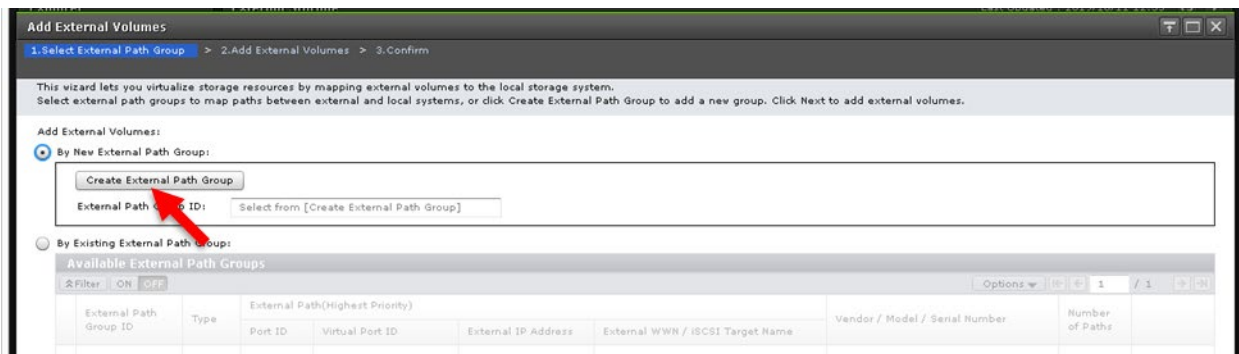
### Discover External Volumes

This section describes how to discover the volumes from the iSCSI virtual machine and virtualize them.

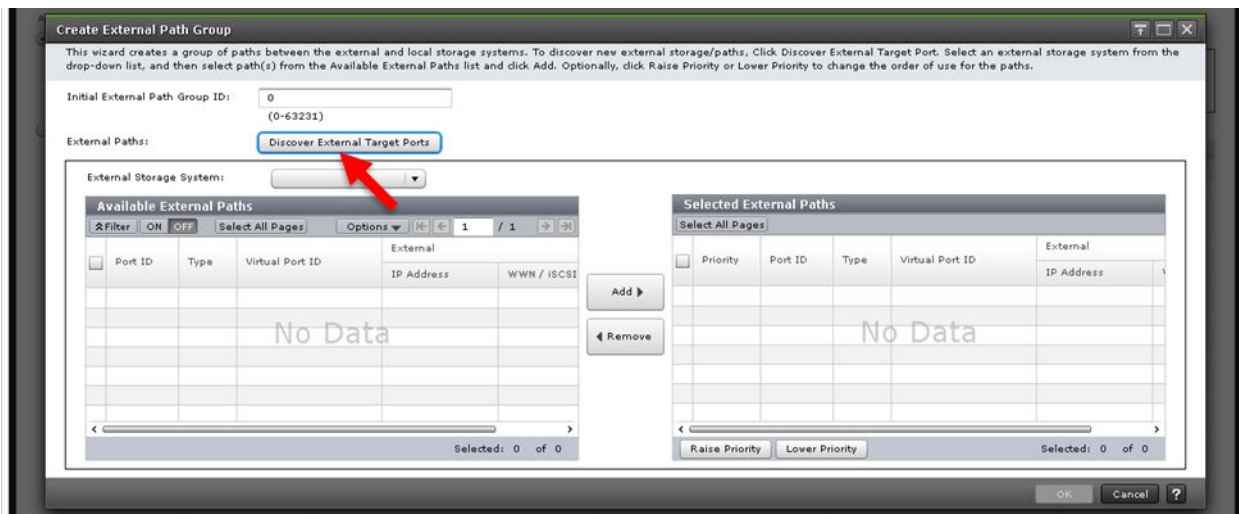
1. Select the **External Storage Systems** tab and then click **Add External Volumes**.



2. Click **Create External Path Group**.

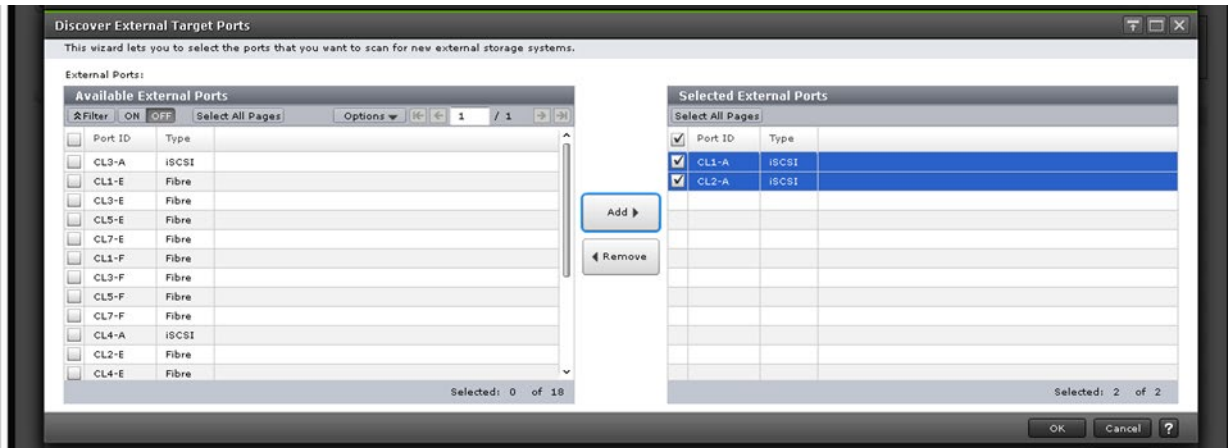


3. Click **Discover External Target Ports**.

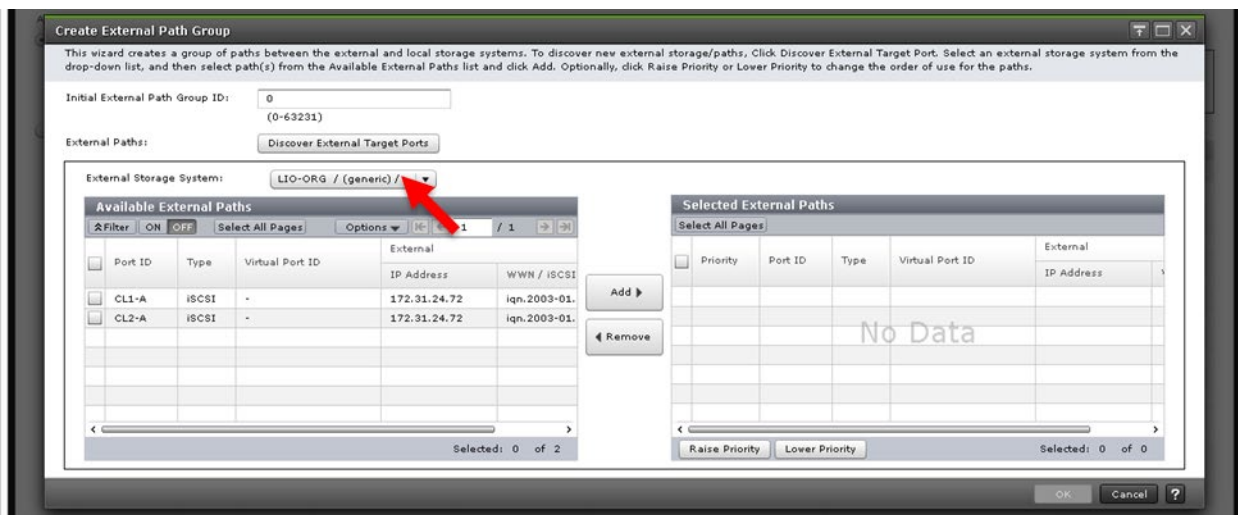


**Global-Active Device Quorum on AWS Cloud**

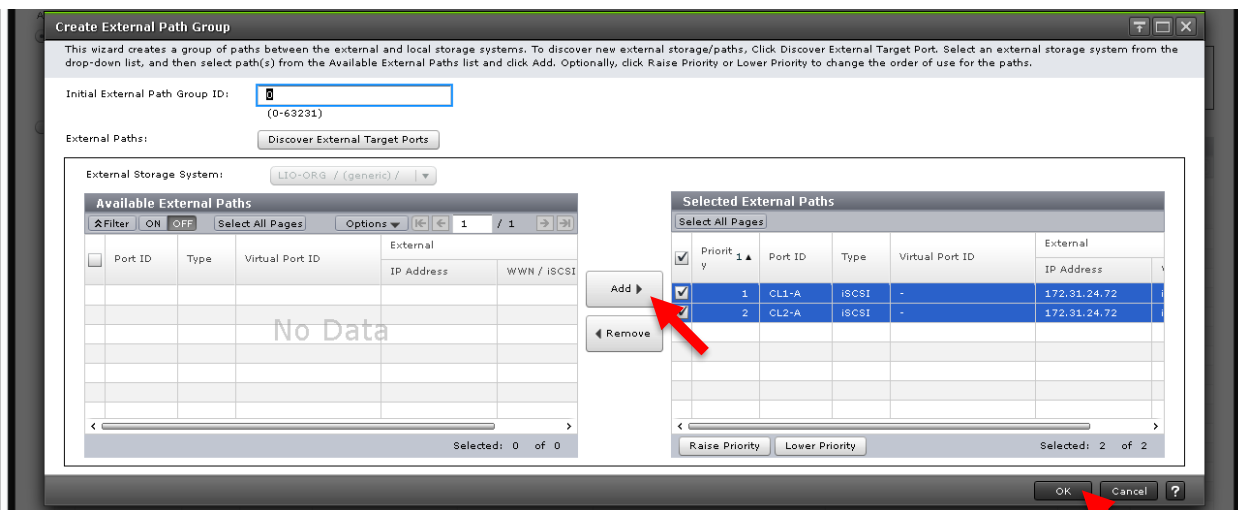
- Select the iSCSI ports, click **Add**, and then click **OK**.



If the discovery is successful, the virtual machine shows up as LIO-ORG.

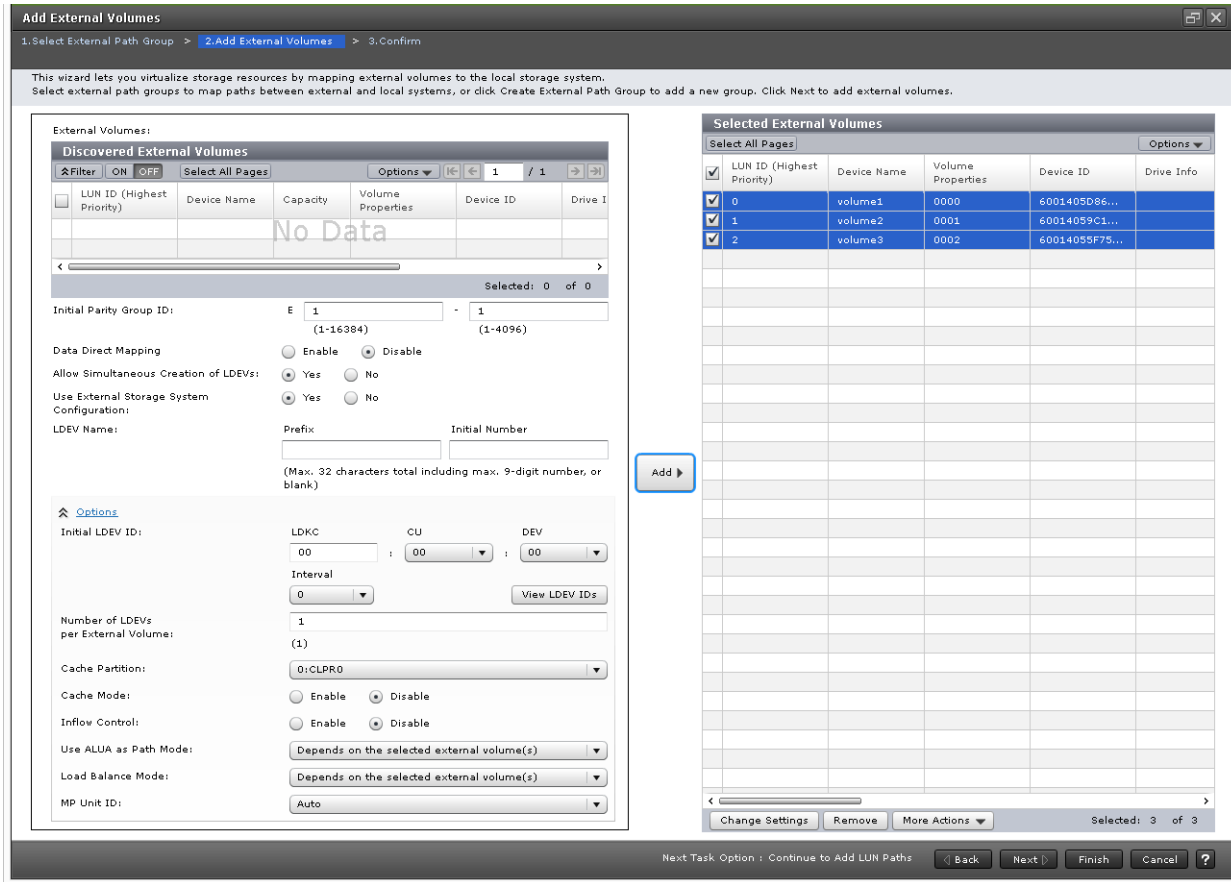


- Select the discovered external paths, click **Add**, and then click **OK**.



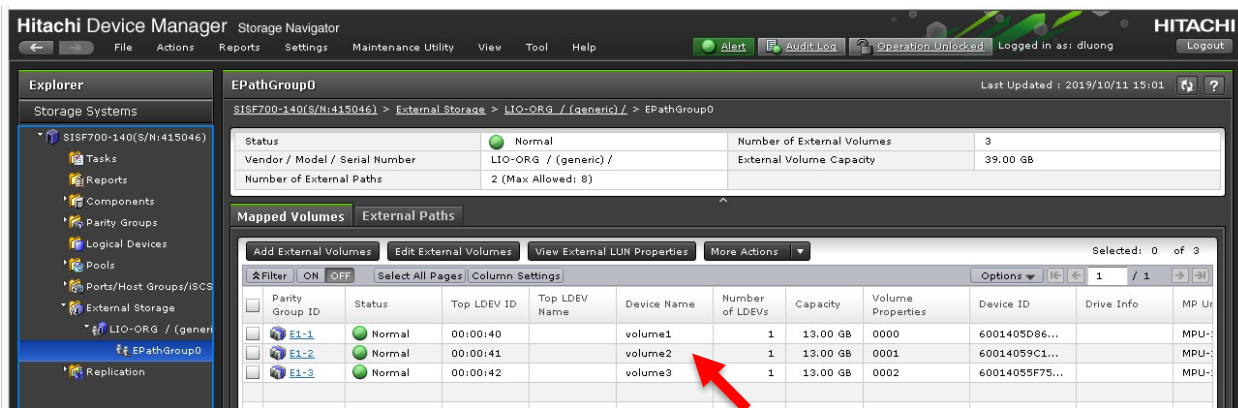
**Global-Active Device Quorum on AWS Cloud**

- On the Add External Volumes window, click **Next**.
- Select the discovered volumes and click **Add**.



- Click **Finish** and then click **Apply**.

The following shows the external volumes after they have been virtualized:

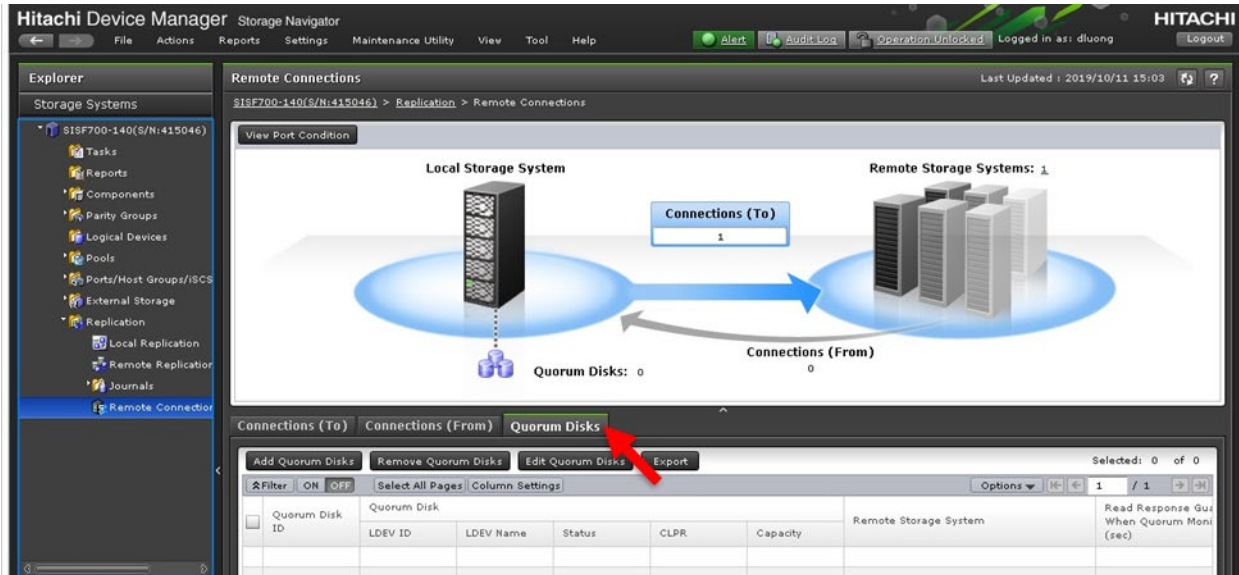


## Global-Active Device Quorum on AWS Cloud

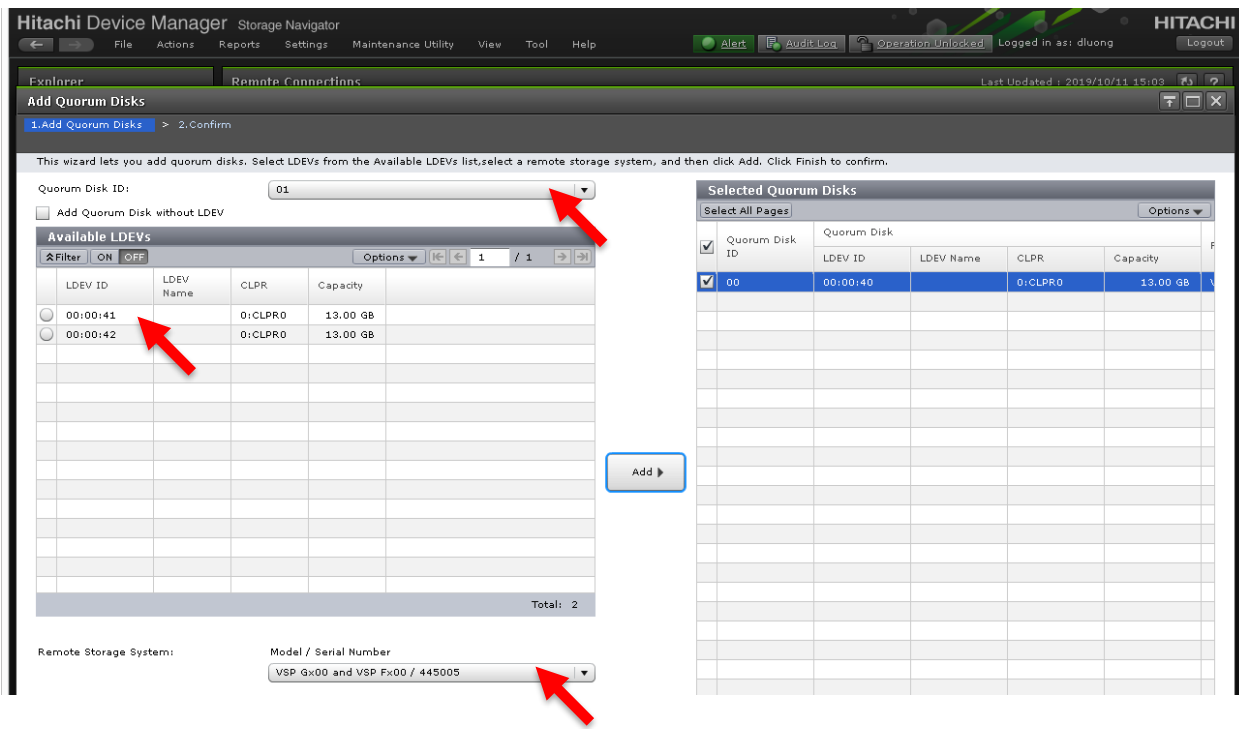
### Define Global-Active Device Quorums

This section describes how to turn the external volumes into GAD quorums. The procedure is the same as it is to a virtualized physical Fibre Channel or iSCSI storage system.

1. Select **Replication > Remote Connections**, and then select the **Quorum Disks** tab.



2. Click **Add Quorum Disks**.
3. In the Add Quorum Disks screen, choose the appropriate option from the **Quorum Disk ID** and the **Remote Storage System** list.
4. From the Available LDEVs table, select the external volume you want to use and click **Add**.

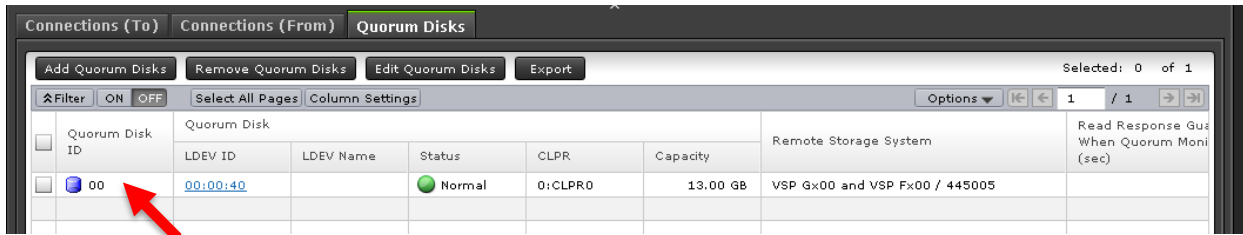


5. Click **Finish** and then click **Apply**.



## Global-Active Device Quorum on AWS Cloud

The following shows the quorum after it has been created:



The screenshot displays the 'Quorum Disks' management interface. At the top, there are tabs for 'Connections (To)', 'Connections (From)', and 'Quorum Disks'. Below the tabs are buttons for 'Add Quorum Disks', 'Remove Quorum Disks', 'Edit Quorum Disks', and 'Export'. The interface shows a table with the following columns: Quorum Disk ID, LDEV ID, LDEV Name, Status, CLPR, Capacity, Remote Storage System, and Read Response Gu... When Quorum Moni... (sec). The table contains one row with the following data: Quorum Disk ID: 00, LDEV ID: 00:00:40, LDEV Name: (empty), Status: Normal (with a green circle icon), CLPR: 0:CLPR0, Capacity: 13.00 GB, Remote Storage System: VSP Gx00 and VSP Fx00 / 445005, and Read Response Gu... When Quorum Moni... (sec): (empty). A red arrow points to the '00' in the 'Quorum Disk ID' column.

Quorum Disk ID	LDEV ID	LDEV Name	Status	CLPR	Capacity	Remote Storage System	Read Response Gu... When Quorum Moni... (sec)
00	00:00:40		Normal	0:CLPR0	13.00 GB	VSP Gx00 and VSP Fx00 / 445005	