

CLOUD CONCEPTS

(Elastic Block Storage and Elastic File System)

- In your system you have seen the hard disk drive (HDD) or SSD for data storage and installing OSes or other software. The types of the volume (alternate name of storage device) may vary but every type of hard drive (SSD, PATA, SCSI, and SATA) has a purpose.

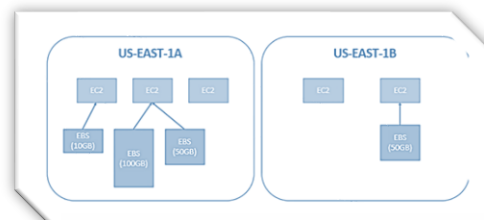
- An EBS (Elastic Block Store) Volume is a network drive you can attach to your instances while they run
- It allows your instances to persist data

- **EBS** is a hard drive that is attached to the server when the server is launched (we have seen that). It is the root volume (main drive) which is deleted on termination (when checked while launching the server). In case of accidental deletion or unexpected termination if there is a backup recovery plan you will get an email through SNS (Simple Notification Service). The service monitoring and creating alarms is cloud watch. Sometimes you need to store the data somewhere...!

EBSVolume

- It's a network drive (i.e. not a physical drive)
 - It uses the network to communicate the instance, which means there might be a bit of latency
 - It can be detached from an EC2 instance and attached to another one quickly
- It's locked to an Availability Zone (AZ)
 - An EBS Volume in us-east-1a cannot be attached to us-east-1b
 - To move a volume across, you first need to snapshot it

- EBS is a network drive which takes time due to the internet. You may see delays when it is attached to a server. It is important to note that EBS volume can be detached from the running server and attached with another server running in the same Availability Zone. If its server is running in another Availability Zone, the volume cannot be attached to that server. To attach the volume cross region firstly create a snapshot (copy of volume). Volume has provisioned capacity in terms of size (GBs) and in terms of input output operations (IOPS). Generally, SSD (high speed) costs more than HDD. Capacity of volume can be extended on need basis other option is to attach a new volume (partitions can be created).



You get billed for all the provisioned capacity

You can increase the capacity of the drive over time

- **SSD-backed volumes** are optimized for transactional workloads involving frequent read/write operations with small I/O size, where the dominant performance attribute is IOPS. SSD-backed volume types include General Purpose SSD and Provisioned IOPS SSD. The following is a summary of the use cases and characteristics of SSD-backed volumes.

	General Purpose SSD volumes		Provisioned IOPS SSD volumes		
Volume type	gp3	gp2	io2 Block Express ‡	io2	io1
Durability	99.8% - 99.9% durability (0.1% - 0.2% annual failure rate)		99.999% durability (0.001% annual failure rate)		99.8% - 99.9% durability (0.1% - 0.2% annual failure rate)
Use cases	<ul style="list-style-type: none">Transactional workloadsVirtual desktopsMedium-sized, single-instance databasesLow-latency interactive applicationsBoot volumesDevelopment and test environments		<div>Workloads that require:</div> <ul style="list-style-type: none">Sub-millisecond latencySustained IOPS performanceMore than 64,000 IOPS or 1,000 MiB/s of throughput	<ul style="list-style-type: none">Workloads that require sustained IOPS performance or more than 16,000 IOPSI/O-intensive database workloads	
Volume size	1 GiB - 16 TiB		4 GiB - 64 TiB		4 GiB - 16 TiB
Max IOPS per volume (16 KiB I/O)	16,000		256,000		64,000 †
Max throughput per volume	1,000 MiB/s	250 MiB/s *	4,000 MiB/s	1,000 MiB/s †	
Amazon EBS Multi-attach	Not supported		Supported		
Boot volume	Supported				

- **HDD-backed volumes** are optimized for large streaming workloads where the dominant performance attribute is throughput. HDD volume types include Throughput Optimized HDD and Cold HDD. The following is a summary of the use cases and characteristics of HDD-backed volumes.

	Throughput Optimized HDD volumes	Cold HDD volumes
Volume type	st1	sc1
Durability	99.8% - 99.9% durability (0.1% - 0.2% annual failure rate)	
Use cases	<ul style="list-style-type: none"> Big data Data warehouses Log processing 	<ul style="list-style-type: none"> Throughput-oriented storage for data that is infrequently accessed Scenarios where the lowest storage cost is important
Volume size	125 GiB - 16 TiB	
Max IOPS per volume (1 MiB I/O)	500	250
Max throughput per volume	500 MiB/s	250 MiB/s
Amazon EBS Multi-attach	Not supported	
Boot volume	Not supported	

- Whenever the term critical is used it means low latency and max number of I/O operations. If IOPs are less than 16K the volume is GP2 and if the IOPs are greater than 16K it refers to IO volume type. Both ST and SC are non-bootable and low-price volumes (HDD type) while GP and IO are bootable volumes and are bit expensive.

- GP2 (SSD): General purpose SSD volume that balances price and performance for a wide variety of workloads
- IO1 (SSD): Highest-performance SSD volume for mission-critical low-latency or high-throughput workloads
- ST1 (HDD): Low cost HDD volume designed for frequently accessed, throughput-intensive workloads
- SC1 (HDD): Lowest cost HDD volume designed for less frequently accessed workloads

- **Snapshot** of a volume means to create a copy of your volume. It is incremental which means that if the snapshot of a volume is created when volume size is 8 GB, in case the volume size is expanded to 10 GB then snapshot will only be created of 2GB incremented. It is recommended to take the snapshot when internet traffic is low. This process can be automated through Amazon Data Lifecycle Manager that will create snapshot as scheduled say 12 am at night. The volumes can be restored from the snapshot available and AMI can also be created using from the snapshots.

EBS Snapshots

- Incremental – only backup changed blocks
- EBS backups use IO and you shouldn't run them while your application is handling a lot of traffic
- Snapshots will be stored in S3 (but you won't directly see them)
- Not necessary to detach volume to do snapshot, but recommended
- Max 100,000 snapshots
- Can copy snapshots across AZ or Region
- Can make Image (AMI) from Snapshot
- EBS volumes restored by snapshots need to be pre-warmed (using fio or dd command to read the entire volume)
- Snapshots can be automated using Amazon Data Lifecycle Manager

EBS Migration

- EBS Volumes are only locked to a specific AZ
- To migrate it to a different AZ (or region):
- Snapshot the volume
- (optional) Copy the volume to a different region
- Create a volume from the snapshot in the AZ of your choice

Encryption: encrypt an unencrypted EBS volume

- Create an EBS snapshot of the volume
- Encrypt the EBS snapshot (using copy)
- Create new ebs volume from the snapshot (the volume will also be encrypted)
- Now you can attach the encrypted volume to the original instance

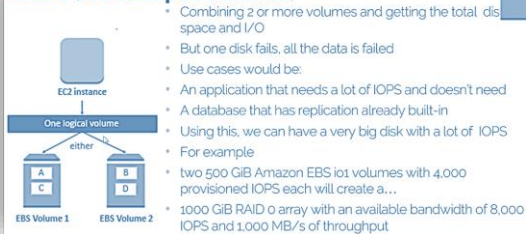
EBS Encryption

- When you create an encrypted EBS volume, you get the following
- Data at rest is encrypted inside the volume
- All the data in flight moving between the instance and the volume is encrypted
- All snapshots are encrypted
- All volumes created from the snapshot
- Encryption and decryption are handled transparently (you have nothing to do)
- Encryption has a minimal impact on latency
- EBS Encryption leverages keys from KMS (AES-256)
- Copying an unencrypted snapshot allows encryption
- Snapshots of encrypted volumes are encrypted

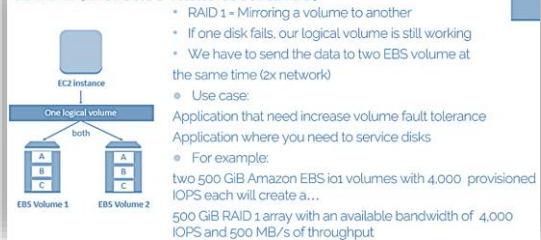
EBS RAID Options

- EBS is already redundant storage (replicated within an AZ)
- But what if you want to increase IOPS to say 100 000 IOPS?
- What if you want to mirror your EBS volumes?
- You would mount volumes in parallel in RAID settings!
- RAID is possible as long as your OS supports it
- Some RAID options are:
- RAID 0
- RAID 1
- RAID 5 (not recommended for EBS – see documentation)
- RAID 6 (not recommended for EBS – see documentation)
- RAID 0 (not recommended for EBS – see documentation)
- RAID 2 (not recommended for EBS – see documentation)
- RAID 7

RAID 0 (increase performance)



RAID 1 (increase fault tolerance)



Lab (Elastic Block Store)

- Initially no volume is created and no server is running. Snapshots = 0.

Resources

You are using the following Amazon EC2 resources in the US East (Ohio) region:

0 Running Instances	0 Elastic IPs
0 Dedicated Hosts	0 Snapshots
0 Volumes	0 Load Balancers
3 Key Pairs	6 Security Groups
0 Placement Groups	

- Launch a server with a simple script. See the default values of the attached volume. Add a new volume. Add tags, attach security groups and key pairs and launch the server. [Port 22 must be ON]

Volume Type	Device	Snapshot	Size (GiB)	Volume Type	IOPS	Throughput (MB/s)	Delete on Termination	Encryption
Root	/dev/xvda	snap-05741358b44a33b45	8	General Purpose SSD (gp2)	100 / 3000	N/A	<input checked="" type="checkbox"/>	Not Encrypt

Add New Volume

- See volumes two volumes can be seen 8GB and 2GB.

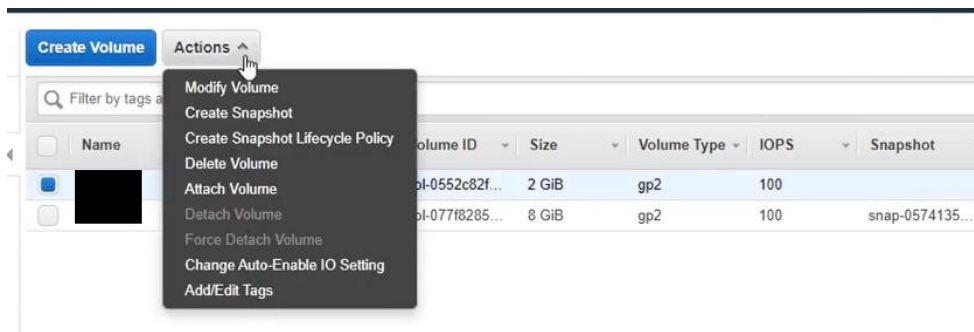
Name	Volume ID	Size	Volume Type	IOPS	Snapshot
	vol-0552c82f...	2 GiB	gp2	100	
	vol-077f8285...	8 GiB	gp2	100	snap-0574135...

- A volume attached with a running server can be detached. After detaching the volume don't forget to delete the volume.

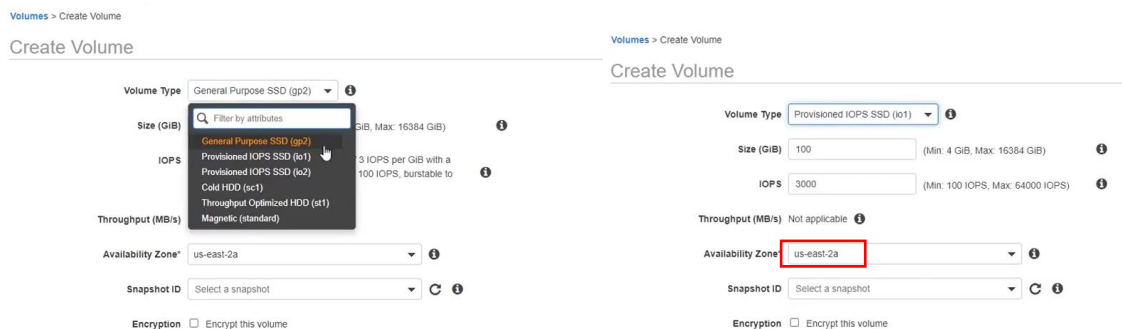
Create Volume Actions

- Modify Volume
- Create Snapshot
- Create Snapshot Lifecycle Policy
- Delete Volume
- Attach Volume
- Detach Volume
- Force Detach Volume
- Change Auto-Enable IO Setting
- Add/Edit Tags

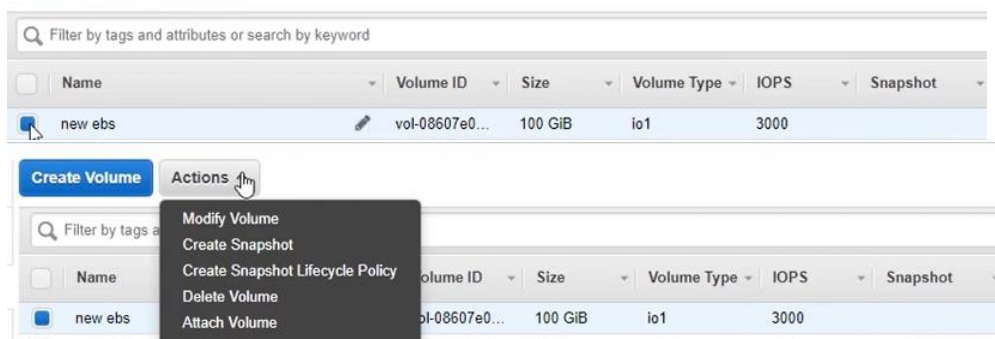
Name	Volume ID	Size	Volume Type	IOPS	Snapshot
	vol-0552c82f...	2 GiB	gp2	100	
	vol-077f8285...	8 GiB	gp2	100	snap-0574135...

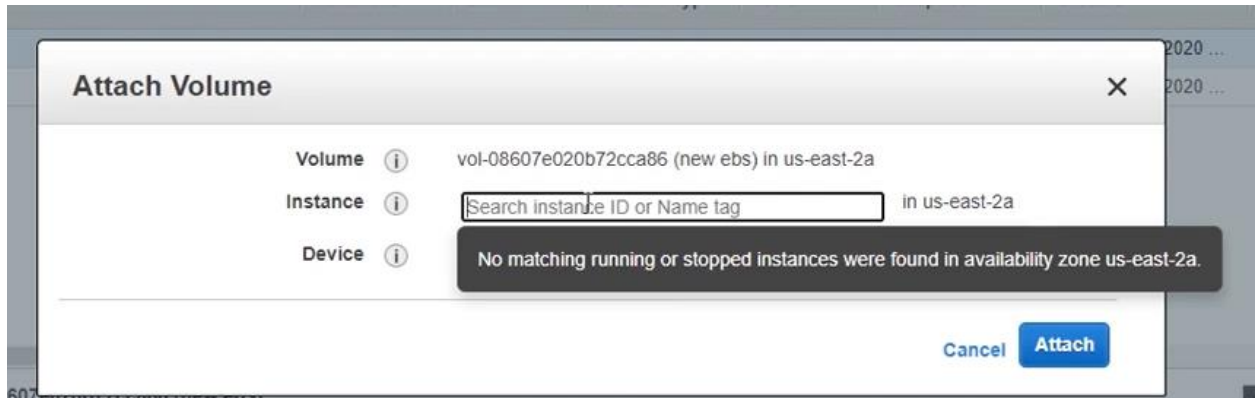


5. A new volume can be created by clicking on create volume. Set the type of volume and define the size. Check encryption if required. Create volume in different AZ i.e., different from the AZ of server. Click on Create Volume.



6. Name the new volume and attach it with the running server. It will show no instances running as the AZ of running server and the AZ of created volume are different. SO the server cannot be attached.





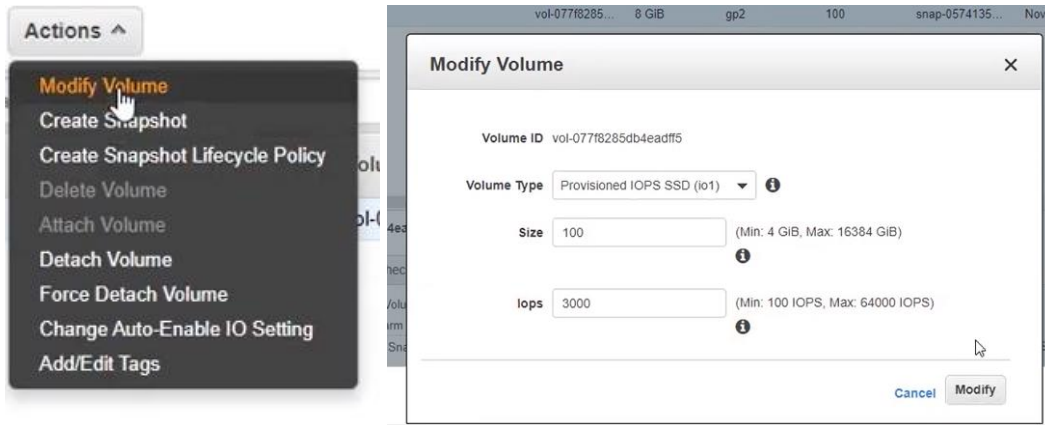
7. Delete this volume and **create a new volume in the same AZ and attach it with the server.**



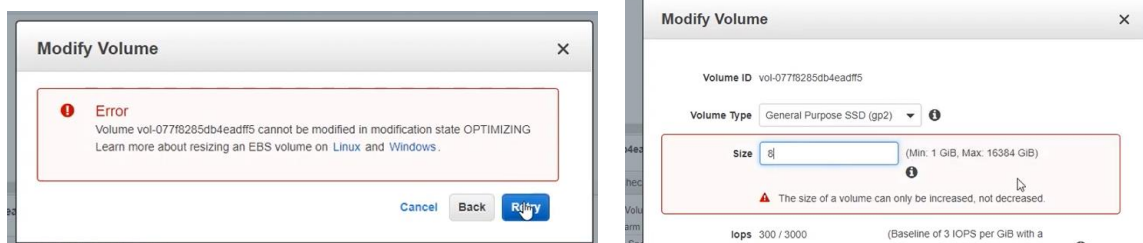
8. See the properties of new attached volume.

EBS-optimized	False	Owner	782982225188
Root device type	ebs	Launch time	November 6, 2020 at 9:29:02 AM UTC+5 (less than one hour)
Root device	/dev/xvda	Termination protection	False
Block devices	/dev/xvda /dev/sdf	Lifecycle	normal
		Monitoring	basic

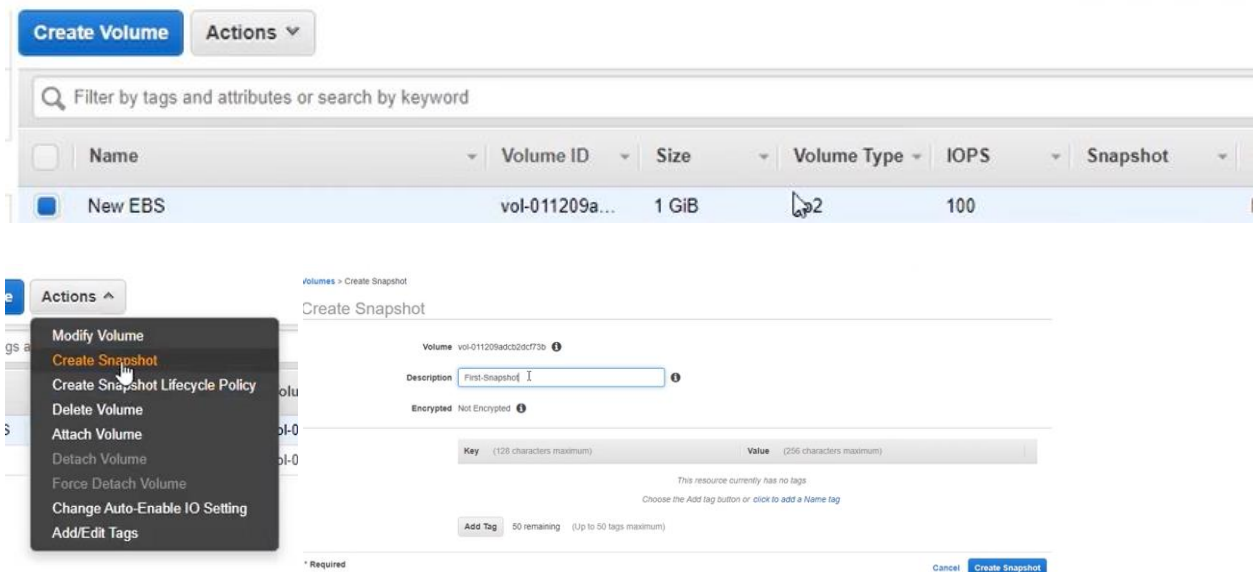
9. Now increase the capacity of the initial volume (8GB). Detach and delete the other volumes. Select the volume, click on Action and Modify Volume. Set the values and change the volume type (if required). The volume will be modified while the server is running.



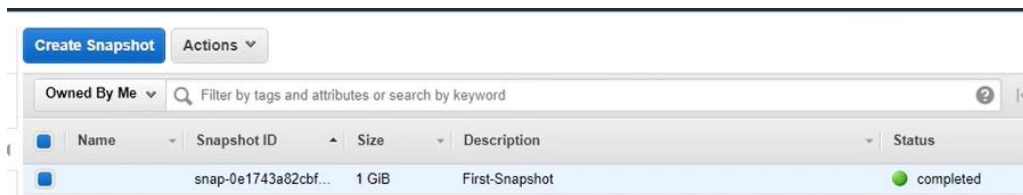
- ☛ It is not recommended to increase the size of the volume if requirement is temporary. It is better to create a new volume and delete it when requirements are fulfilled. Amazon allows the increase in size but donot allow to decrease the size. Even further increment will require a gap of 6hrs. Error may vary from what is shown.



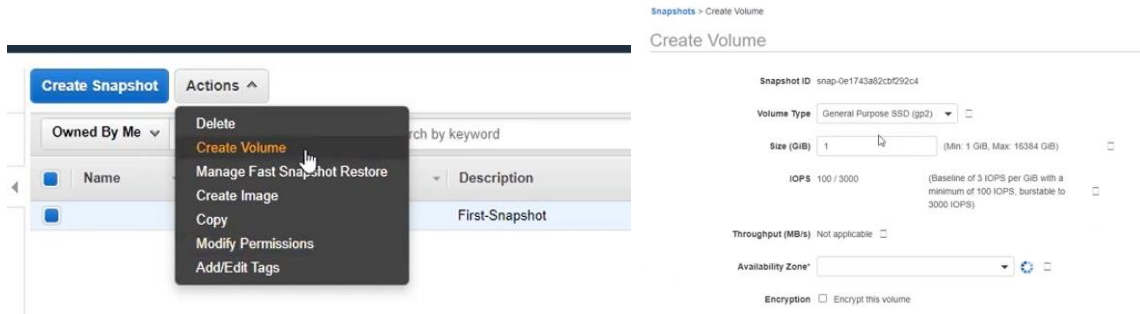
10. Create the snapshot of a volume. It is better to use 1GB size minimum to quickly create a copy. Larger the capacity greater will be the time. Click on Create Snapshot and write the description.



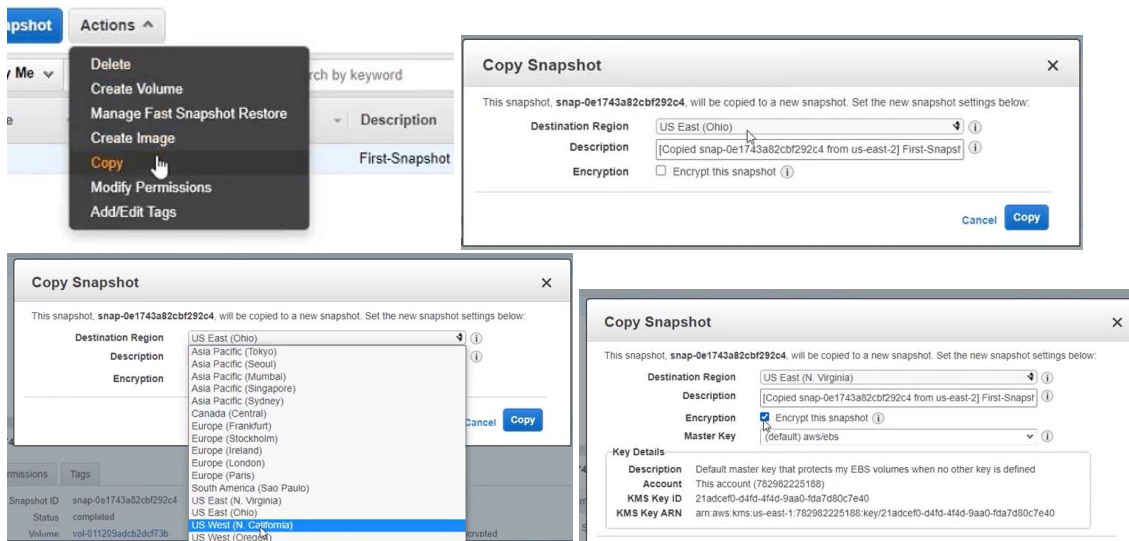
11. Click on snapshots to see the snapshots created.



12. A new volume can be created from the created snap shot.

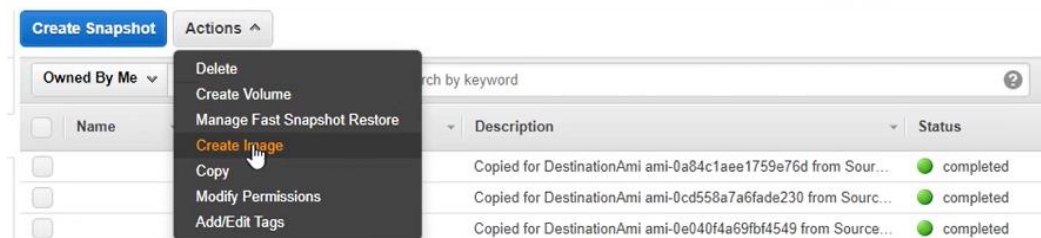


13. To migrate the snap shot, click Copy and select the region where you want the volume copy to be migrated (currently in Ohio, other region is North virginia). Encryption option is also available at this point.



14. Switch the region to see the snapshot in that region. Now you can create the machine image from this snapshot (seen in AMI) and launch the server using same steps.

DONOT FORGET TO DETACH VOLUMES, DELETE SNAPSHOTS/VOLUME and DEREGISTER AND DELETE AMIs and TERMINATE THE SERVERS.



The screenshot shows the 'Create Image from EBS Snapshot' dialog in the AWS console. The dialog has fields for Name, Description, Architecture (x86_64), Virtualization type (Hardware-assisted virtualization), Root device name (/dev/sda1), and Kernel ID (Use default). Below these is a 'Block Device Mappings' table with one entry for the root device. At the bottom are 'Cancel' and 'Create' buttons.

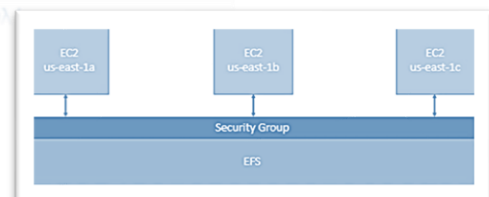
Volume Type	Device	Snapshot	Size (GiB)	Volume Type	IOPS	Throughput (MB/s)	Delete on Termination	Encrypted
Root	/dev/sda1	snap-0aabd61f79da44a7a	1	General Purpose	100 / 3000	N/A	<input checked="" type="checkbox"/>	<input type="checkbox"/> Not Encrypted

Name	AMI Name	AMI ID	Source	Owner	Visibility	Status
		ami-012ab99731588ef5f	782982225188/...	782982225188	Private	available

- **EFS** is like a shared drive that can be attached with multiple servers in different availability zones. Security groups are very important in EFS as they will define the servers to which this drive can be attached. It is not compatible with windows, can only be configured with linux AMI. Since this drive is used with multiple instances hence many input output operations are performed on this drive.

- Managed NFS (network file system) that can be mounted on many EC2
- EFS works with EC2 instances in multi-AZ
- Highly available, scalable, expensive (3x gp2), pay per use

- On premise means ones own physical data center. If the on premise data is to be shared with cloud data store EFS is synced.



- Use cases: content management, web serving, data sharing, Wordpress
- Uses NFSv4.1 protocol
- Uses security group to control access to EFS
- Compatible with Linux based AMI (not Windows)
- Performance mode:
- General purpose (default)
- Max I/O – used when thousands of EC2 are using the EFS
- EFS file sync to sync from on-premise file system to EFS
- Backup EFS-to-EFS (incremental – can choose frequency)
- Encryption at rest using KMS

<input type="checkbox"/>	EBS volume can be attached to one instance at a time
<input type="checkbox"/>	EBS volumes are locked at AZ level
<input type="checkbox"/>	EBS volumes can be migrated across AZ first by backing up/copying and create it in other AZ
<input type="checkbox"/>	EBS volume back up must not be created when application is handling a lot of traffic, although it is possible but not recommended
<input type="checkbox"/>	Root EBS volume of instances get terminated by default (if option is checked).
<input type="checkbox"/>	If disk IO is high then increase EBS volume size for GP2 or convert it to IOP
<input type="checkbox"/>	EFS can be mounted to thousands of instances
<input type="checkbox"/>	EFS is a shared volume, can share websites.
<input type="checkbox"/>	EBS GP2 can be optimized on cost
<input type="checkbox"/>	Custom AMI can be used for faster deployment

Lab (Elastic File System)

- Search for EFS and click on create file system. You can simply give the name with default values and create the volume. To explore other options, click on customize.

Amazon Elastic File System

Scalable, elastic, cloud-native NFS file system

Amazon Elastic File System (Amazon EFS) provides a simple, scalable, elastic file system for general purpose workloads for use with AWS Cloud services and on-premises resources.

Create file system

Create an EFS file system with service recommended settings.

[Create file system](#)

Pricing (US)

Create file system

Create an EFS file system with service recommended settings. [Learn more](#)

Name - optional
Name your file system.

Name must not be longer than 256 characters, and must only contain letters, numbers, and these characters: + - = . _ : /

Virtual Private Cloud (VPC)
Choose the VPC where you want EC2 instances to connect to your file system. [Learn more](#)

default

- Write the name and check enable automatic backup option if backup is to be created else it can be unchecked. By default this option is checked. Life cycle management will help to keep the cost optimized if files are not accessed in X time move it in infrequent access storage. GP or IO type can be selected. TAG if required.

The screenshot shows the AWS Elastic File System console with the following configuration details:

- Step 3 - optional:** File system policy
- Step 4:** Review and create
- Name your file system:** (Note: Name must not be longer than 256 characters, and must only contain letters, numbers, and these characters: + - _ . : /)
- Automatic backups:** ☒ Enable automatic backups
- Life cycle management:** Automatically save money as access patterns change by moving files into the EFS Infrequent Access storage class. (Learn more)
- Performance mode:**
 - ☒ **General Purpose**: Ideal for latency-sensitive use cases, like web serving environments and content management systems
 - ☐ **Max I/O**: Scale to higher levels of aggregate throughput and operations per second
- Encryption:** Choose to enable encryption of your file system's data at rest. Uses the AWS KMS service key (aws/elasticfilesystem) by default. ☒ Enable encryption of data at rest (Customize encryption settings)
- Tags - optional:** Add tags to associate key-value pairs to your resource. (Learn more)

Tag key	Tag value - optional	
<input type="text" value="Enter key"/>	<input type="text" value="Enter value"/>	<input type="button" value="Remove tag"/>

 (Add tag button, You can add 49 more tags)

- In next step VPC is selected. Default VPC is selected, you can select your own created VPC as well. Depending on the region you are in AZ can be selected remove the SG (def with VPC-def) from the AZs and attach your created security group. Proceed to next step, tag and click on create.
- **Required: Create a security group with outbound rules with all the traffic enabled.**

The screenshot shows the AWS Security Groups console with the following configuration details:

- Security group name:**
- Description:**
- VPC:**
- Security group rules:**
 - Inbound:** (selected)
 - Outbound:**

Type	Protocol	Port Range	Destination	Description
All traffic	All	0 - 65535	Custom 0.0.0.0/0	e.g. SSH for Admin
- Add Rule:** (button)

Step 1
File system settingsStep 2
Network accessStep 3 - optional
File system policyStep 4
Review and create

Network access

Network

Virtual Private Cloud (VPC)

Choose the VPC where you want EC2 instances to connect to your file system. [Learn more](#)vpc-029c3269
default

Mount targets

A mount target provides an NFSv4 endpoint at which you can mount an Amazon EFS file system. We recommend creating one mount target per Availability Zone. [Learn more](#)Availability
zone

us-east-2a

Subnet ID

subnet-2...

IP address

Automatic

Security groups

Choose s...

Remove

sg-
97f8fee
b
default

Virtual Private Cloud (VPC)

Choose the VPC where you want EC2 instances to connect to your file system. [Learn more](#)vpc-029c3269
defaultvpc-029c3269
defaultA mount target provides an NFSv4 endpoint at which you can mount an Amazon EFS file system. We recommend creating one mount target per Availability Zone. [Learn more](#)Availability
zone

us-east-2a

Subnet ID

subnet-2...

IP address

Automatic

Security groups

Choose s...

Remove

sg-
97f8fee
b
default

us-east-2b

subnet-a...

Automatic

Choose s...

Remove

sg-
97f8fee
b
default

us-east-2b

subnet-a...

Automatic

Choose s...

Remove

sg-
021d9a
56a2bb
552fe
efs-demo

us-east-2c

subnet-b...

Automatic

Choose s...

Remove

sg-
021d9a
56a2bb
552fe
efs-demo

Amazon EFS > File systems > Create

Step 1
File system settings

Step 2
Network access

Step 3 - optional
File system policy

Step 4
Review and create

File system policy - optional

Policy options

Select one or more of these common policy options, or create a more advanced policy using the editor. [Learn more](#)

☐ Prevent root access by default*

☐ Enforce read-only access by default*

☐ Enforce in-transit encryption for all clients

* Identity-based policies can override these default permissions.

Policy editor

Clear

1

Cancel

Previous

Create

Amazon EFS > File systems

File systems (1)

View details

Delete

Create file system

Filter by property values

Name	File system ID	Encrypted	Total size	Size in EFS Standard	Size in EFS IA	Provisioned Throughput (MiB/s)
efs_demo	fs-866c0efe	Encrypted	6 KiB	6 KiB	0 Bytes	-

- Launch two servers in different availability zones. With SG as shown.

: Configure Instance Details

Purchasing option ⓘ ☐ Request Spot instances

Network ⓘ vpc-029c3269 (default) ⓘ [Create new VPC](#)

Subnet ⓘ subnet-242ef84f | Default in us-east-2a ⓘ [Create new subnet](#)
4090 IP Addresses available

Auto-assign Public IP ⓘ Use subnet setting (Enable) ⓘ

Placement group ⓘ ☐ Add instance to placement group

Capacity Reservation ⓘ Open ⓘ

Domain join directory ⓘ No directory ⓘ [Create new directory](#)

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

CPU options ⓘ ☐ Specify CPU options

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

Step 6: Configure Security Group

A security group is a set of firewall rules that control the traffic for your instance. On this page, you can add rules to allow specific traffic to reach your instance. For example, if you want to set up a web server and allow Internet traffic to reach your instance, add rules that allow unrestricted access to the HTTP and HTTPS ports. You can create a new security group or select from an existing one below. [Learn more](#) about Amazon EC2 security groups.

Assign a security group: ☒ Create a new security group
☐ Select an existing security group

Security group name: ec2-for-efs

Description: launch-wizard-1 created 2020-11-20T09:05:32.366+05:00

Type ⓘ	Protocol ⓘ	Port Range ⓘ	Source ⓘ	Description ⓘ
SSH	TCP	22	Custom 0.0.0.0/0	e.g. SSH for Admin Desktop

[Add Rule](#)

Warning

Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

[Cancel](#) [Previous](#) [Review and Launch](#)

Step 5: Add Tags

A tag consists of a case-sensitive key-value pair. For example, you could define a tag with key = Name and value = Webserver.

A copy of a tag can be applied to volumes, instances or both.

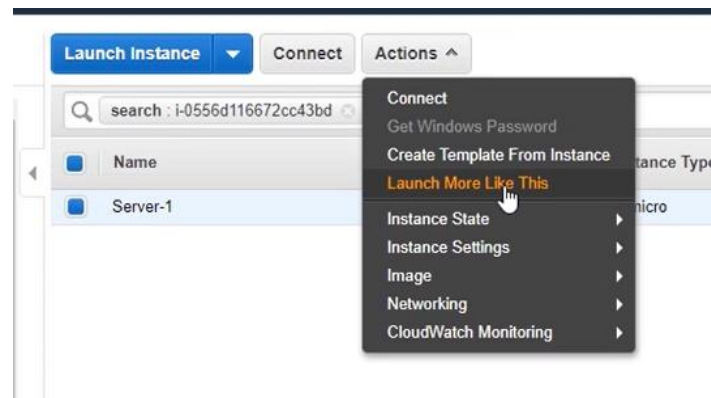
Tags will be applied to all instances and volumes. [Learn more](#) about tagging your Amazon EC2 resources.

Key (128 characters maximum)	Value (256 characters maximum)	Instances ⓘ	Volumes ⓘ
Name	Server-1	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

[Add another tag](#) (Up to 50 tags maximum)

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status
Server-1	i-0556d116672cc43bd	t2.micro	us-east-2a	running	Initializing	None

- You can duplicate the instance by clicking on Actions. Change the name and follow the default configuration, just change the AZ different from the previous one. Access both servers through there public IP through SSH (power shell/putty).



Step 7: Review Instance Launch

SSH TCP 22 0.0.0.0/0

▼ Instance Details [Edit instance details](#)

Number of instances 1 Purchasing option On demand

Network vpc-029c3269
Subnet subnet-242ef84f
EBS-optimized No
Monitoring No
Termination protection No
Shutdown behavior Stop
Hibernation behavior Disabled
Capacity Reservation open
IAM role None
Domain join directory None
Tenancy default
Credit specification Use default
Host ID
Host resource group name

<input type="checkbox"/>	Server-2	i-04525368b83241c6f	t2.micro	us-east-2b	running
<input checked="" type="checkbox"/>	Server-1	i-0556d116672cc43bd	t2.micro	us-east-2a	running

```
ec2-user@ip-172-31-28-223:~$
login as: ec2-user
Authenticating with public key "imported-openssh-key"

  _l  ( _l  /
 _l  ( _l  /  Amazon Linux 2 AMI
 _l  \ _l  \

https://aws.amazon.com/amazon-linux-2/
```

```
ec2-user@ip-172-31-3-252:~$
login as: ec2-user
Authenticating with public key "imported-openssh-key"

  _l  ( _l  /
 _l  ( _l  /  Amazon Linux 2 AMI
 _l  \ _l  \

https://aws.amazon.com/amazon-linux-2/
```

- Now to create a shared file that will be automatically shared with other server when created.

```
sudo yum install -y amazon-efs-utils
```

```
sudo mkdir /efs
```

```
sudo mount -t efs fs-866c0efe:/ /efs
```

```
cd /efs
```

- First command will install the required packages.
- mkdir is used to create a directory.
- mount is used to attach the volume with server
- cd will change the present working directory.

- Run the following commands on **both servers**.

```
ec2-user@ip-172-31-28-223:~
login as: ec2-user
Authenticating with public key "imported-openssh-key"

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

https://aws.amazon.com/amazon-linux-2/
No packages needed for security; 3 packages available
Run "sudo yum update" to apply all updates.
[ec2-user@ip-172-31-28-223 ~]$ sudo yum install -y amazon-efs-utils
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Resolving Dependencies
--> Running transaction check
--> Package amazon-efs-utils.noarch 0:1.28.2-1.amzn2 will be installed
--> Processing Dependency: stunnel >= 4.56 for package: amazon-efs-utils-1.28.2-1.amzn2.noarch
```

- Then create a directory on **both servers**.

```
ec2-user@ip-172-31-3-252:~
[ec2-user@ip-172-31-3-252 ~]$
[ec2-user@ip-172-31-3-252 ~]$ sudo mkdir /efs
```

- Now attach the EFS using the command on both servers as shown. Please note that you must write the id of your created EFS volume. NOTE that no output will be shown. It is due to the attached security group where no inbound rule is defined. For this edit the attached security group with EFS and ON the port for NFS (Network File System). Attach the security group create for EC2 instances with that port as source. **This will sync both servers through security group any modification in EFS will be shared with both servers.**

```
ec2-user@ip-172-31-3-252:~
[ec2-user@ip-172-31-3-252 ~]$
[ec2-user@ip-172-31-3-252 ~]$ sudo mkdir /efs
[ec2-user@ip-172-31-3-252 ~]$ sudo mount -t efs fs-866c0efe:/ /efs
```

File systems (1)						
<input type="text" value="Filter by property values"/>						
Name	File system ID	Encrypted	Total size	Size in EFS Standard	Size in EFS IA	
efs_demo	fs-866c0efe	Encrypted	6 KiB	6 KiB	0 Bytes	

Edit inbound rules

Type	Protocol	Port Range	Source	Description
NFS	TCP	2049	Custom sg-077b17ab897d3e6f9	e.g. SSH for Admin Desktop

Add Rule

NOTE: Any edits made on existing rules will result in the edited rule being deleted and a new rule created with the new details. This will cause traffic that depends on that rule to be dropped for a very brief period of time until the new rule can be created.

Cancel Save

- Press Ctrl+C to stop the running command (putty) and run the same command on both servers after configuring SG inbound rule.

```
ec2-user@ip-172-31-3-252:~
[ec2-user@ip-172-31-3-252 ~]$
[ec2-user@ip-172-31-3-252 ~]$ sudo mkdir /efs
[ec2-user@ip-172-31-3-252 ~]$ sudo mount -t efs fs-866c0efe:/ /efs
^CTraceback (most recent call last):
  File "/sbin/mount.efs", line 1988, in <module>
    main()
  File "/sbin/mount.efs", line 1984, in main
    mount_nfs(dns_name, path, mountpoint, options)
  File "/sbin/mount.efs", line 1009, in mount_nfs
    out, err = proc.communicate()
  File "/usr/lib64/python2.7/subprocess.py", line 483, in communicate
    return self._communicate(input)
  File "/usr/lib64/python2.7/subprocess.py", line 1124, in _communicate
    stdout, stderr = self._communicate_with_poll(input)
  File "/usr/lib64/python2.7/subprocess.py", line 1178, in _communicate_with_poll
    ready = poller.poll()
KeyboardInterrupt
[ec2-user@ip-172-31-3-252 ~]$ sudo mount -t efs fs-866c0efe:/ /efs
```

- Now change the current directory to efs directory created in mounted EFS volume.

```
ec2-user@ip-172-31-3-252:/efs
[ec2-user@ip-172-31-3-252 ~]$
[ec2-user@ip-172-31-3-252 ~]$ cd /efs
[ec2-user@ip-172-31-3-252 efs]$
```

- If you run any command then it will not be executed. **Permission Denied. Error...!**

```
[ec2-user@ip-172-31-3-252 ~]$
[ec2-user@ip-172-31-3-252 ~]$ cd /efs
[ec2-user@ip-172-31-3-252 efs]$ echo "Welcom toEFS" > hello.txt
-bash: hello.txt: Permission denied
[ec2-user@ip-172-31-3-252 efs]$
```

- Now be a super user, from ec2 user to a root user and execute the commands.

```
root@ip-172-31-3-252:/efs
[ec2-user@ip-172-31-3-252 ~]$
[ec2-user@ip-172-31-3-252 ~]$ cd /efs
[ec2-user@ip-172-31-3-252 efs]$ echo "Welcom toEFS" > hello.txt
-bash: hello.txt: Permission denied
[ec2-user@ip-172-31-3-252 efs]$ sudo su
[root@ip-172-31-3-252 efs]# echo "Welcome to EFS" > hello.txt
[root@ip-172-31-3-252 efs]# cat hello.txt
Welcome to EFS
[root@ip-172-31-3-252 efs]#
```

- Echo is used to print what is written to screen or to a file
- > is used to write to a file
- cat will display the contents of the file

- Now go to server 2 and run the commands as shown. You will see that file created through the server 1 is also available in server 2. It is because both servers are synced via SG. **It is important to note that only servers attached SG for EC2 instances will have the file access. No other port is open ...!**

```
root@ip-172-31-28-223:/efs
[ec2-user@ip-172-31-28-223 ~]$ sudo mkdir /efs
[ec2-user@ip-172-31-28-223 ~]$ sudo mount -t efs fs-866c0efe:/ /efs
[ec2-user@ip-172-31-28-223 ~]$ cd /efs
[ec2-user@ip-172-31-28-223 efs]$ sudo su
[root@ip-172-31-28-223 efs]# cat hello.txt
Welcome to EFS
```

- Now create a file in server two and access it from server 1 to see the syncing.

```
root@ip-172-31-28-223:/efs
[ec2-user@ip-172-31-28-223 ~]$ sudo mkdir /efs
[ec2-user@ip-172-31-28-223 ~]$ sudo mount -t efs fs-866c0efe:/ /efs
[ec2-user@ip-172-31-28-223 ~]$ cd /efs
[ec2-user@ip-172-31-28-223 efs]$ sudo su
[root@ip-172-31-28-223 efs]# cat hello.txt
Welcome to EFS
[root@ip-172-31-28-223 efs]# echo "Welcome to Cloud Computing" > abc.txt
[root@ip-172-31-28-223 efs]# cat abc.txt
Welcome to Cloud Computing
```



```
root@ip-172-31-3-252:/efs
[ec2-user@ip-172-31-3-252 ~]$
[ec2-user@ip-172-31-3-252 ~]$ cd /efs
[ec2-user@ip-172-31-3-252 efs]$ echo "Welcom toEFS" > hello.txt
-bash: hello.txt: Permission denied
[ec2-user@ip-172-31-3-252 efs]$ sudo su
[root@ip-172-31-3-252 efs]# echo "Welcome to EFS" > hello.txt
[root@ip-172-31-3-252 efs]# cat hello.txt
Welcome to EFS
[root@ip-172-31-3-252 efs]# cat abc.txt
Welcome to Cloud Computing
[root@ip-172-31-3-252 efs]#
```

- Close the sessions, delete the security groups of server or terminate the servers and delete all other SGs

Amazon EFS > File systems

File systems (1) View details Delete Create file system

Filter by property values

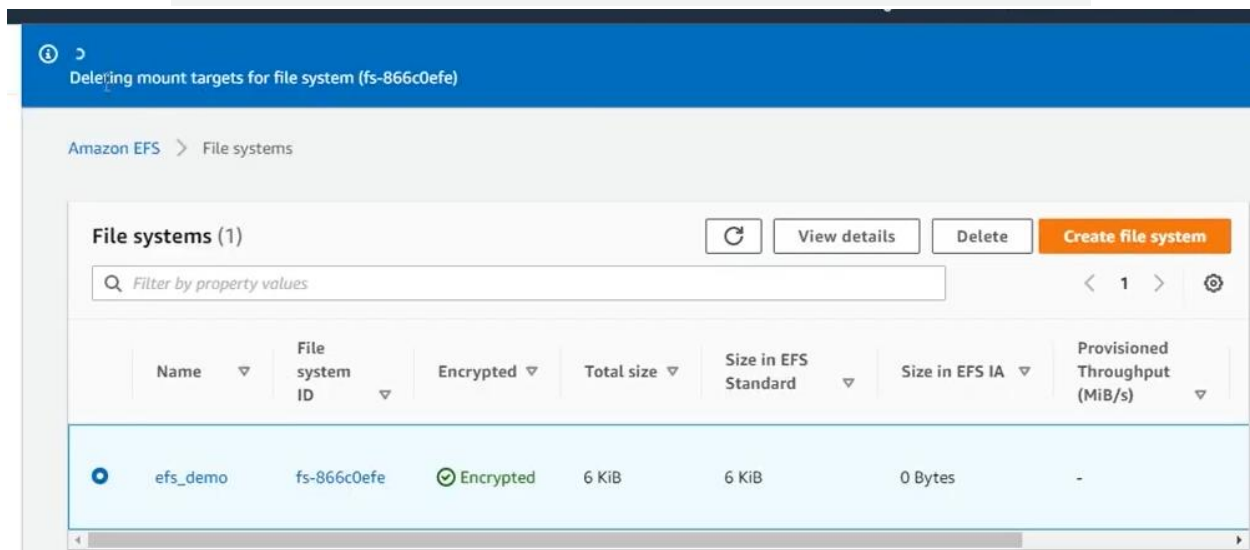
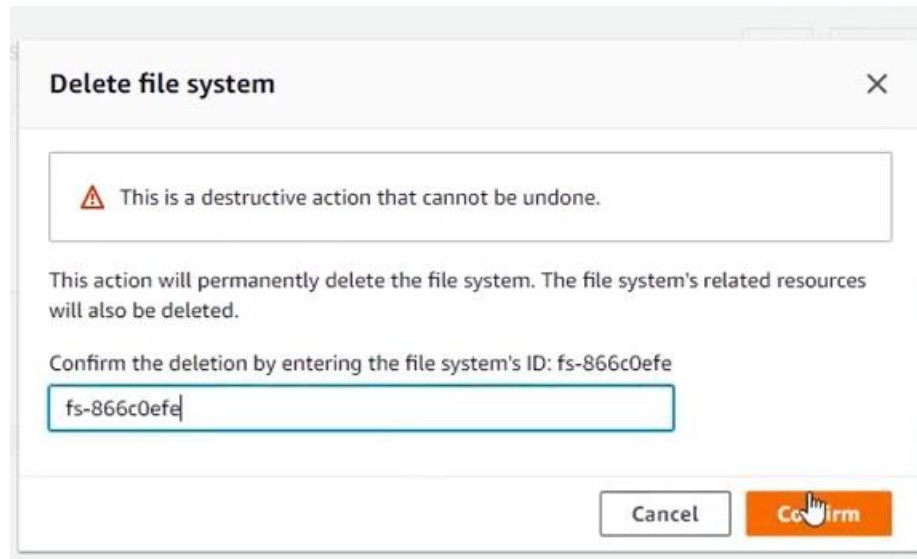
Name	File system ID	Encrypted	Total size	Size in EFS Standard	Size in EFS IA	Provisioned Throughput (MiB/s)
efs_demo	fs-866c0efe	Encrypted	6 KiB	6 KiB	0 Bytes	-

Amazon EFS > File systems

File systems (1) View details Delete Create file system

Filter by property values

Name	File system ID	Encrypted	Total size	Size in EFS Standard	Size in EFS IA	Provisioned Throughput (MiB/s)
efs_demo	fs-866c0efe	Encrypted	6 KiB	6 KiB	0 Bytes	-



- You might get this error if the servers to which EFS is mounted is not terminated. So firstly delete the instances and then delete the mounted volume.

