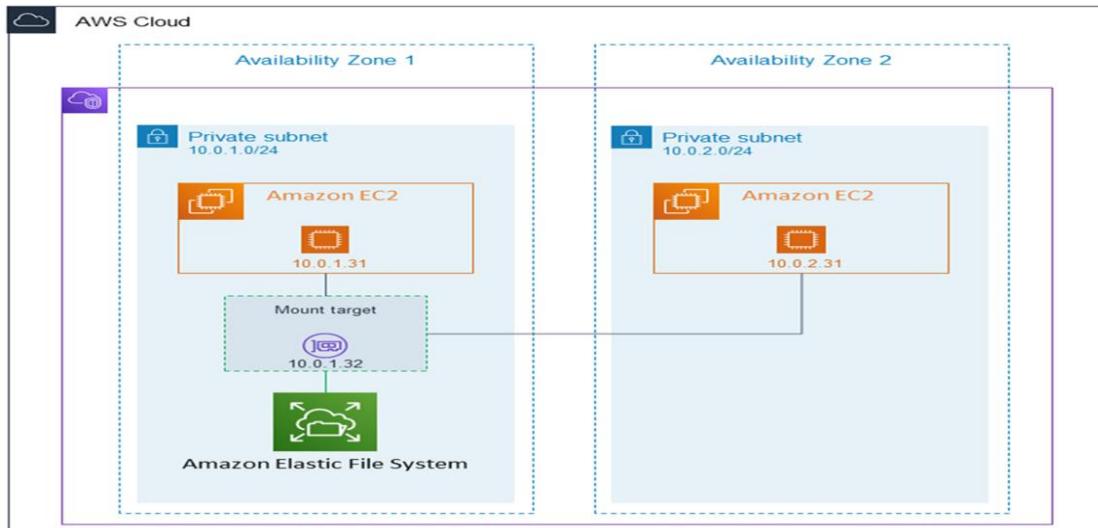


Elastic File Storage

Amazon Elastic File System (EFS) is a scalable and fully managed file storage service provided by Amazon Web Services (AWS). It allows you to create and configure file systems that can be accessed concurrently from multiple EC2 instances, providing a highly available and scalable storage solution for your applications and workloads.



Amazon EFS supports **for only Amazon Linux instance**:

Step 1: Launching EC2 Instances

1. Sign in to AWS Management Console.
2. Go to the EC2 dashboard.
3. Click on "Launch Instance".
4. Configure instance details:
 - Instance Name: **EFS-1**
5. Choose an Amazon Machine Image (AMI) **for Linux**.
6. Select instance type: **t2.micro**.
 - Key Pair: **EFS**
 - Network: **Choose Subnet-1a**
 - Security Group: **Create a new security group (SG) and add NFS and allow from anywhere.**

7. Launch the instance.

8. Repeat the above steps to launch another instance named **EFS-2** in **Subnet-1b** with the same configuration.

The screenshot shows the 'Launch an instance' wizard in the AWS Management Console. The 'Network settings' section is open, showing a VPC dropdown set to 'vpc-0d1325c8014a7178b' (default) and a Subnet dropdown set to 'subnet-02e8d41cf03965d25'. The 'Auto-assign public IP' field is set to 'Enable'. Under 'Firewall (security groups)', the 'Create security group' radio button is selected. A new security group named 'launch-wizard-32-SGG2' is being created. The 'Description - required' field is left empty. On the right, the 'Summary' panel shows 1 instance, the software image as Amazon Linux 2023 AMI 2023.10, and the virtual server type as t2.micro. The 'Launch instance' button is highlighted.

The screenshot shows the 'Launch an instance' wizard with detailed security group rules. Two rules are defined:

- Security group rule 3 (TCP: 80, 0.0.0.0/0)**: Type: HTTP, Protocol: TCP, Port range: 80. Source type: Anywhere. Description: e.g. SSH for admin desktop.
- Security group rule 4 (TCP: 2049, 0.0.0.0/0)**: Type: NFS, Protocol: TCP, Port range: 2049. Source type: Anywhere. Description: e.g. SSH for admin desktop.

A warning message at the bottom states: "⚠️ 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." On the right, the 'Summary' panel shows 1 instance, the software image as Amazon Linux 2023 AMI 2023.10, and the virtual server type as t2.micro. The 'Launch instance' button is highlighted.

The screenshot shows the AWS EC2 Instances page. On the left, there's a navigation sidebar with sections like EC2, Dashboard, EC2 Global View, Events, Instances (selected), Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity Reservations, Capacity Manager, Images, AMIs, AMI Catalog, and Elastic Block Store. The main content area displays a table of instances. The table has columns for Name, Instance ID, Instance state, Instance type, Status check, Alarm status, Availability Zone, and Public IPv4. There are two entries: EFS-1 (i-0d15af5467a53200) and EFS-2 (i-0c3be69de26945ba3). Both are running t2.micro instances. Below the table, there's a detailed view for instance i-0d15af5467a53200 (EFS-1), showing details like Public IPv4 address (54.242.34.166), Private IPv4 addresses (172.31.39.32), and Public DNS (ec2-54-242-34-166.compute-1.amazonaws.com).

Step 2: Creating an EFS File System

1. Go to the EFS service in the AWS Management Console.
2. Click on "Create file system".
3. Specify details:
 - Name: Optional
 - VPC: Default
 - Enable region button.
 Click on "Next".
4. In the Network settings:
 - Delete all security groups.
 - **Select the newly created security group (NFS).---which is created while creating instance**
 Click on "Next" and review the configuration.
- Click on "Create file system".

aws Search [Alt+S] United States (N. Virginia) Account ID: 6844-8522-0151 vclabs/user4617815=gsaisaatvik@gmail.com

Elastic File System

File systems Access points

AWS Backup AWS DataSync AWS Transfer

Documentation

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 recommended settings.

Create file system

Pricing

With EFS, there are no minimum fees. You pay only for the storage that you use, the data that you read and write, and any additional throughput that you provision.

Estimate your cost using the [AWS Pricing Calculator](#)

Learn more about pricing

What is Amazon Elastic File System?

Amazon Elastic File System - Scalable, Elastic, Clo... Copy link

Amazon Elastic File System

CloudShell Feedback © 2026, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

aws Search [Alt+S] United States (N. Virginia) Account ID: 6844-8522-0151 vclabs/user4617815=gsaisaatvik@gmail.com

Amazon EFS > File systems > Create

File system settings

Step 2 Network access Step 3 - optional File system policy Step 4 Review and create

General

Name FEFS

Name can include letters, numbers, and +-=_.:/ symbols, up to 256 characters.

File system type

Choose to either store data across multiple Availability Zones or within a single Availability Zone. [Learn more](#)

Regional Offers the highest levels of availability and durability by storing file system data across multiple Availability Zones within an AWS Region.

One Zone Provides continuous availability to data within a single Availability Zone within an AWS Region.

Automatic backups

Automatically backup your file system data with AWS Backup using recommended settings. Additional pricing applies. [Learn more](#)

Enable automatic backups

Lifecycle management

Automatically save money as access patterns change by moving files into the Infrequent Access (IA) or Archive storage class. [Learn more](#)

Transition into Infrequent Access (IA)
Transition files to IA based on the time since they were last accessed in Standard storage.
30 day(s) since last access ▾

Transition into Archive
Transition files to Archive based on the time since they were last accessed in Standard storage.
90 day(s) since last access ▾

Transition into Standard
Transition files back to Standard storage based on when they are first accessed in IA or Archive storage.
None ▾

CloudShell Feedback © 2026, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#Home.

AWS Search [Alt+S] United States (N. Virginia) Account ID: 6844-8522-0151 vclabs/user4617815=gaisaatvik@gmail.com

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

Network access

Network

Virtual Private Cloud (VPC) | [Learn more](#)

Choose the VPC where you want EC2 instances to connect to your file system.

vpc-0d1325c8014a7178b
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 | Subnet ID | IP address type | IPv4 address | IPv6 address | Security groups |
|-------------------|-------------|-----------------|--------------|--------------|-------------------------------------|
| us-east-1a | subnet-0... | IPv4 only | Optional | - | Choose se... Remove |
| us-east-1b | subnet-0... | IPv4 only | Optional | - | Choose se... Remove |

sg-0dd6af10fc
db275b9
launch-wizard-52-
SGG1

sg-02822ef7e
53fcc5fb
launch-wizard-52-
SGG2

CloudShell Feedback © 2026, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#Home.

AWS Search [Alt+S] United States (N. Virginia) Account ID: 6844-8522-0151 vclabs/user4617815=gaisaatvik@gmail.com

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 custom policy using the editor. [Learn more](#)

Prevent root access by default*
 Enforce read-only access by default*
 Prevent anonymous access
 Enforce in-transit encryption for all clients

* Identity-based policies can override these default permissions.

▶ **Grant additional permissions**

Policy editor {JSON}

Clear

Manual changes will prevent the use of the policy options on the left until the editor is cleared.

Cancel Previous Next

CloudShell Feedback © 2026, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

The screenshot shows the AWS EFS console interface. On the left, there's a navigation sidebar with 'Elastic File System' selected. The main area has a green success banner at the top stating 'Success! File system (fs-0066d7a62fa373f00) is available.' Below this is a table titled 'File systems (1)' with columns: Name, File system ID, Encrypted, Total size, Size in Standard, Size in IA, Size in Archive, Provisioned Throughput (MiB/s), and File system state. A single row is shown for 'EFS' with the ID 'fs-0066d7a62fa373f00', marked as encrypted, with 6.00 KB total size and 0 Bytes provisioned throughput. The status is 'Available'. At the bottom of the page, there are links for CloudShell, Feedback, and cookie preferences.

Step 3: Accessing the two EC2 instances named EFS-1& EFS -2 in **two different PowerShell** sessions and performing the specified tasks:

Accessing EFS-1 Instances in Two Different PowerShell Sessions:

1. Open Two PowerShell Sessions:

- Open two separate PowerShell windows or tabs on your local machine.

For each Instance:

2. SSH into the Instance:

- Use the SSH command to connect to the EFS-1 instance

```
ssh -i [path-to-your-keypair.pem] ec2-user@[instance-public-ip]
```

3. Switch to Root User:

- Gain root access by executing the following command:

```
sudosu
```

4. Create a Directory:

- Make a directory named "efs" using the following command:

```
mkdirs
```

5. Install Amazon EFS Utilities:

- Use yum package manager to install the Amazon EFS utilities:

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

6. List Files:

- Execute the following command to list files in the current directory:

```
ls
```

7. Verify Installation (Optional):

- Optionally, you can verify the installation of the Amazon EFS utilities by checking the version:

```
efs-utils --version
```

Repeat Steps 2-7 for ----- EFS-2 Instance.

By following these steps, you'll have accessed each EFS-1 instance in separate PowerShell sessions, switched to the root user, created a directory named "efs," installed the Amazon EFS utilities, and listed files in the directory. This setup allows you to configure and manage each instance individually as needed.

us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#Home:

aws Search [Alt+S] United States (N. Virginia) Account ID: 6844-8522-0151 vclabs/user4617815=gsaatvik@gmail.com

EC2 Instances i-0d15af5467a53200 Connect to instance

Connect Info

Connect to an instance using the browser-based client.

EC2 Instance Connect Session Manager SSH client EC2 serial console

Instance ID: i-0d15af5467a53200 (EFS-1)

Connection type:

- Connect using a Public IP
Connect using a public IPv4 or IPv6 address
- Connect using a Private IP
Connect using a private IP address and a VPC endpoint

Public IPv4 address 54.242.34.166

IPv6 address

Username: ec2-user

Note: In most cases, the default username, ec2-user, is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI username.

Cancel Connect

CloudShell Feedback © 2026, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

aws Search [Alt+S] United States (N. Virginia) Account ID: 6844-8522-0151 vclabs/user4617815=gsaatvik@gmail.com

```
Amazon Linux 2023
https://aws.amazon.com/linux/amazon-linux-2023

[ec2-user@ip-172-31-7-61 ~]$ sudo su
[root@ip-172-31-7-61 ec2-user]# sudo mkdir efs
[root@ip-172-31-7-61 ec2-user]# ls
efs
[root@ip-172-31-7-61 ec2-user]# yum install -y amazon-efs-utils
```

i-0c3be69de26945ba3 (EFS-2) X

Public IPs: 98.80.191.176 Private IPs: 172.31.7.61

CloudShell Feedback © 2026, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

The screenshot shows a terminal session in AWS CloudShell. The user is installing the Amazon EFS utilities package on an Amazon Linux 2023 instance. The terminal output includes:

```

Amazon Linux 2023
https://aws.amazon.com/linux/amazon-linux-2023

[ec2-user@ip-172-31-39-32 ~]$ sudo su
[root@ip-172-31-39-32 ec2-user]# mkdir efs
[root@ip-172-31-39-32 ec2-user]# ls
efs

[root@ip-172-31-39-32 ec2-user]# yum install -y amazon-efs-utils
Amazon Linux 2023 Kernel Livepatch repository
Dependencies resolved.

Transaction Summary
Install 2 Packages

Total download size: 4.9 M
Installed size: 10 M

```

i-0d15af5467a53200 (EFS-1)

PublicIPs: 54.242.34.166 PrivateIPs: 172.31.39.32

CloudShell Feedback © 2026, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Step 4: Attaching EFS to EC2 Instances

1. Go to the EFS service in the AWS Management Console.
2. Click on the target EFS file system.
3. Click on the "Attach" button.
4. Choose "**Mount via DNS**" option.
5. Copy the displayed command.,on both ec2 instances(powershell)
6. Paste and execute the copied command in the terminal to mount the EFS file system onto the instance.

```

Running transaction test
Transaction test succeeded.
Running transaction
Preparing : 1/1
Installing : stunnel-5.58-1.amzn2023.0.2.x86_64 1/2
Running scriptlet: stunnel-5.58-1.amzn2023.0.2.x86_64 1/2
Installing : amazon-efs-utils-2.4.1-1.amzn2023.x86_64 2/2
Running scriptlet: amazon-efs-utils-2.4.1-1.amzn2023.x86_64 2/2
Verifying : amazon-efs-utils-2.4.1-1.amzn2023.x86_64 1/2
Verifying : stunnel-5.58-1.amzn2023.0.2.x86_64 2/2

WARNING:
A newer release of "Amazon Linux" is available.

Available Versions:
Version 2023.10.20260120:
Run the following command to upgrade to 2023.10.20260120:
dnf upgrade --releasever=2023.10.20260120

Release notes:
https://docs.aws.amazon.com/linux/al2023/release-notes/relnotes-2023.10.20260120.html

=====
Installed:
amazon-efs-utils-2.4.1-1.amzn2023.x86_64           stunnel-5.58-1.amzn2023.0.2.x86_64
Complete!
[root@ip-172-31-7-61 ec2-user]#

```

i-0c3be69de26945ba3 (EFS-2)

PublicIPs: 98.80.191.176 PrivateIPs: 172.31.7.61

CloudShell Feedback © 2026, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

File systems > fs-0066d7a62fa373f00

Elastic File System < EFS (fs-0066d7a62fa373f00) Delete Attach

Mount your Amazon EFS file system on a Linux instance. Learn more

Mount via DNS Mount via IP

Using the EFS mount helper:

```
sudo mount -t efs -o tls fs-0066d7a62fa373f00:/ efs
```

Using the NFS client:

```
sudo mount -t nfs4 -o nfsvers=4.1,rsize=1048576,wsize=1048576,hard,timeo=600,retrans=2,noresvport fs-0066d7a62fa373f00.efs.us-east-1.amazonaws.com:/ efs
```

See our user guide for more information. Learn more

Metered size

Total size

CloudShell Feedback © 2026, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Step 5: Verify and Test EFS

1. Change the directory to EFS on both instances.
2. Create a file on one instance.
3. Verify that the file automatically synchronizes and appears on the other instance.

The screenshot shows a terminal session in AWS CloudShell. The user has run several commands to create an EFS file system, attach it to an EC2 instance, and verify the setup.

```
aws | [Alt+S] Search | United States (N. Virginia) | Account ID: 6844-8522-0151 | vclabs/user4617815=gsaisaatvik@gmail.com | 2/2 | 1/2 | 2/2

Running scriptlet: amazon-efs-utils-2.4.1-1.amzn2023.x86_64
Verifying      : amazon-efs-utils-2.4.1-1.amzn2023.x86_64
Verifying      : stunnel-5.58-1.amzn2023.0.2.x86_64

=====
WARNING:
A newer release of "Amazon Linux" is available.

Available Versions:
Version 2023.10.20260120:
Run the following command to upgrade to 2023.10.20260120:
dnf upgrade --releasever=2023.10.20260120
Release notes:
https://docs.aws.amazon.com/linux/al2023/release-notes/relnotes-2023.10.20260120.html

=====
Installed:
amazon-efs-utils-2.4.1-1.amzn2023.x86_64                               stunnel-5.58-1.amzn2023.0.2.x86_64

Complete!
[root@ip-172-31-7-61 ec2-user]# sudo mount -t efs -o tls fs-0066d7a62fa373f00:/ efs
[root@ip-172-31-7-61 ec2-user]# cd efs
[root@ip-172-31-7-61 efs]# ls
f1
[root@ip-172-31-7-61 efs]# touch f2
[root@ip-172-31-7-61 efs]# ls
f1  f2
[root@ip-172-31-7-61 efs]# 
```

i-0c3be69de26945ba3 (EFS-2)

Public IPs: 98.80.191.176 Private IPs: 172.31.7.61

CloudShell Feedback © 2026, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

By above steps, we can successfully create EC2 instances, configured them, created an EFS file system, and attached it to the instances in the same availability zone in the Mumbai region. Now, the instances can seamlessly access and share files stored in the EFS file system.