

EFS (*Elastic File System*)

What is EFS used for in AWS?

Amazon Elastic File System (Amazon EFS) provides serverless, fully elastic file storage so that you can share file data without provisioning or managing storage capacity and performance.

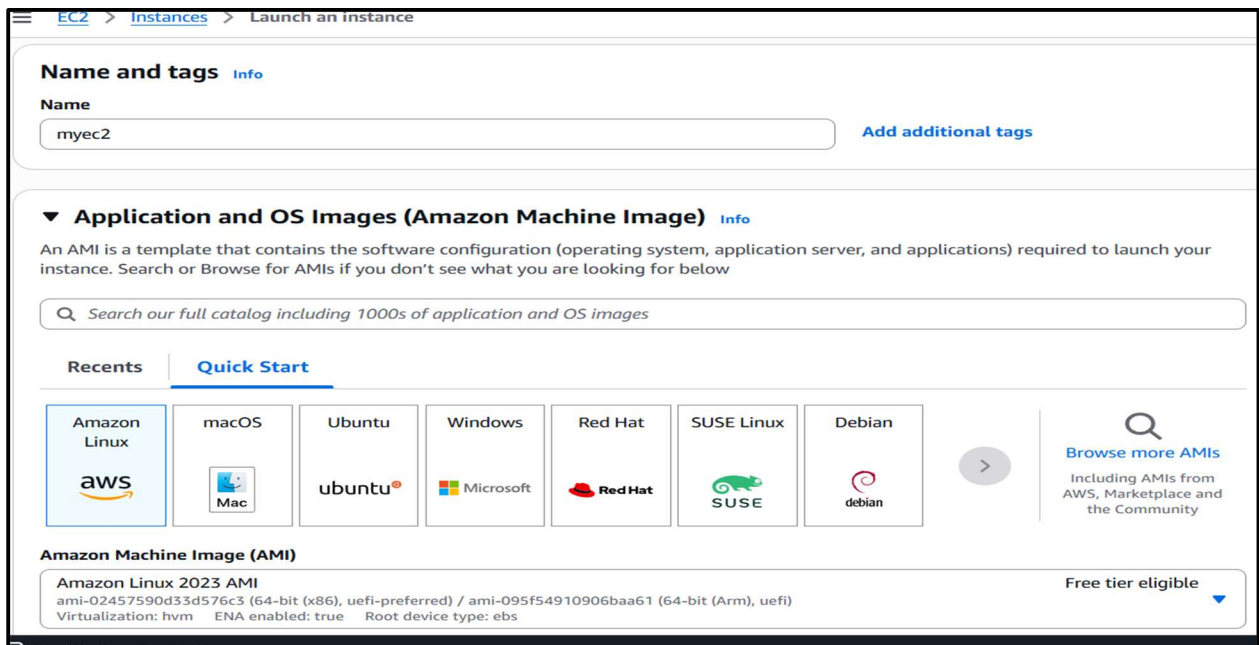
Steps for EFS ::

Step1: Sign in to AWS Management Console

- Navigate to the AWS Console and sign in using the IAM Username and Password provided.
- Ensure your selected AWS Region is US East (N. Virginia) us-east-1.

Step2: Launching two EC2 Instances

- **Instance type:**
- Select t2. From the top menu, navigate to Services > Compute > EC2.
- Click on Instances in the left navigation pane, then click the Launch instances button.
- **Name and tags:**
- For Name, enter myec2.



- **Application and OS Images (AMI):**
- In the search bar, type Amazon Linux and select the Amazon Linux 2023 AMI. It is free tier eligible.

EFS (*Elastic File System*)

- micro (Free tier eligible) from the dropdown list.
- Key pair (login)
- Click Create new key pair
- Key pair name: *myec2key*
- Key pair type: *RSA*
- Private key file format: *pem*
- Click Create key pair and save the downloaded file.

▼ Key pair (login) Info

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name - required

myec2key ▼

↻ Create new key pair

▼ Network settings Info

VPC - required Info

vpc-04145a1273dee2ce4 (default) ▼

172.31.0.0/16

Subnet Info

No preference ▼

↻ Create new subnet

Auto-assign public IP Info

Enable ▼

Additional charges apply when outside of free tier allowance

Firewall (security groups) Info

A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

☒ Create security group

☐ Select existing security group

Security group name - required

- **Network settings:**
- Click Edit.
- For Auto-assign public IP, select Enable.
- Under Firewall (security groups), select Create security group.
- Security group name: EFS Security Group
- The SSH rule should be present by default. Set its Source type to Anywhere.
- Click Add security group rule.
- Type: Choose NFS.
- Source: Choose Anywhere.

EFS (*Elastic File System*)

Type Info	Protocol Info	Port range Info
ssh	TCP	22
Source type Info	Source Info	Description - optional Info
Anywhere	<input type="text" value="Add CIDR, prefix list or security group"/> 0.0.0.0/0	e.g. SSH for admin desktop
▼ Security group rule 2 (TCP, 2049, 0.0.0.0/0) Remove		
Type Info	Protocol Info	Port range Info
NFS	TCP	2049
Source type Info	Source Info	Description - optional Info
Anywhere	<input type="text" value="Add CIDR, prefix list or security group"/> 0.0.0.0/0	e.g. SSH for admin desktop

- **Summary panel:**
- Find Number of instances and enter 2

▼ Summary

Number of instances | Info

When launching more than 1 instance, **consider EC2 Auto Scaling**

Software Image (AMI)

Amazon Linux 2023 AMI 2023.7.2...[read more](#)

- **Click Launch Instance.**
- **Click View all Instances to see your instances being created.**
- **Select each instance and give them unique names in the “Name” column: myec2-1 and myec2-2.**
- **Note the IPv4 Public IP addresses for both instances for later use. N**

<input type="checkbox"/>	myec2-1	i-0c8f1176fb348ebfe	Running		t2.micro	2/2 checks passed	View alarms	us-east-1c
<input type="checkbox"/>	myec2-2	i-08f7b2ddc37082b69	Running		t2.micro	2/2 checks passed	View alarms	us-east-1c

Step3: Creating an Elastic File System

EFS (*Elastic File System*)

- **Navigate to Services > Storage > EFS.**
- **Click Create file system.**

- **Configure the file system:**
- Name: myefs

General

Name - optional
Name your file system.

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**

- **Virtual Private Cloud (VPC): Ensure the default VPC is selected.**
- **Storage class: Leave as Standard**
- **Click Customize :**
- On the File system settings page, uncheck Enable automatic backups and click Next.
- **On the Network access page:**
- Under Mount targets, the default VPC should already be selected.
- For each Availability Zone listed, remove the default security group by clicking the 'X'.
- In the now-empty Security groups box for each Availability Zone, select the **EFS Security Group** you created earlier.
- Click Next.

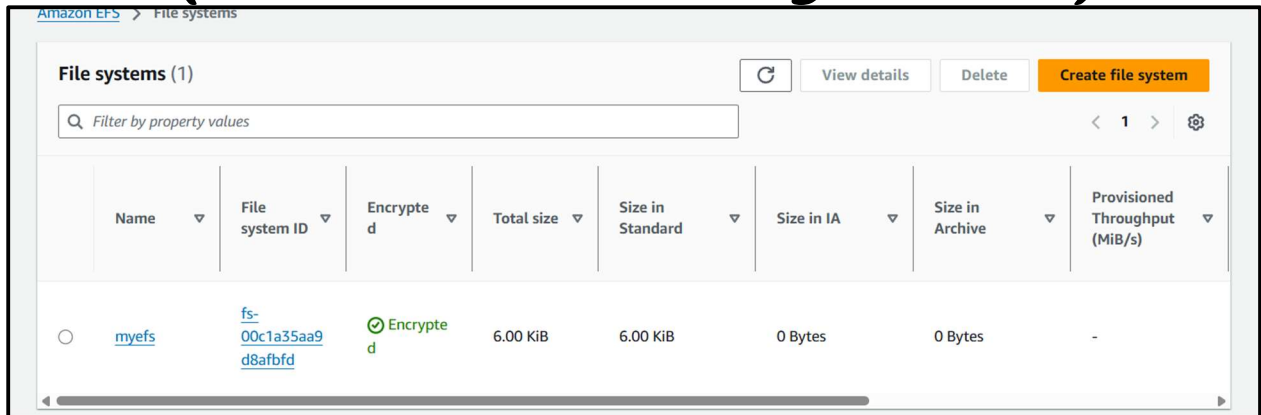
Mount targets

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	-	<div>sg-03e081c70913552ca EFS Security Group</div>	Remove
us-east-1b	subnet-0...	IPv4 only	Optional	-	<div>Choose s...</div>	Remove
us-east-1c	subnet-0...	IPv4 only	Optional	-	<div>sg-03e081c70913552ca EFS Security Group</div>	Remove

- On the File system policy page, leave the policy empty and click Next.
- Review your settings and click Create

EFS (*Elastic File System*)



The screenshot shows the Amazon EFS console interface. At the top, there's a breadcrumb 'Amazon EFS > File systems'. Below this, a header bar contains 'File systems (1)', a refresh button, 'View details', 'Delete', and a prominent orange 'Create file system' button. A search bar with the placeholder 'Filter by property values' is also present. The main content is a table with columns: Name, File system ID, Encryption status, Total size, Size in Standard, Size in IA, Size in Archive, and Provisioned Throughput (MiB/s). One file system is listed: 'myefs' with ID 'fs-00c1a35aa9d8afbdf', marked as 'Encrypted', and showing 6.00 KiB in Standard storage and 0 Bytes in IA and Archive storage. The Provisioned Throughput is listed as '-'. A pagination bar at the bottom shows '1' of 1 items.

Name	File system ID	Encryption	Total size	Size in Standard	Size in IA	Size in Archive	Provisioned Throughput (MiB/s)
myefs	fs-00c1a35aa9d8afbdf	Encrypted	6.00 KiB	6.00 KiB	0 Bytes	0 Bytes	-

Step4: Mount the File System to myec2-1 Instance

- SSH into the myec2-1 instance using its public IP address and the **.pem** key you downloaded.
- **Once connected, switch to the root user: `sudo -s`**
- **Update the instance and install the EFS utilities package:**
- **`Yum -y update`**
- **`Yum install -y amazon-efs-utils`**
- **Create a directory to mount the file system to: `mkdir efs`**

- ***In the AWS EFS Console, select your EFS-Demo file system and click the Attach button in the top-right.***
- A window will pop up. Copy the command shown under Using the EFS mount helper. It will look similar to this:
- **`Sudo mount -t efs -o tls fs-xxxxxxxxxxxxxxxx:/ efs`**
- Paste and run this command in your SSH session.
- Verify the mount was successful by running `df -h`. You should see the EFS file system listed at the bottom.

EFS (*Elastic File System*)

```
Running transaction test
Transaction test succeeded.
Running transaction
  Preparing      : 
  Installing     : stunnel-5.58-1.amzn2023.0.2.x86_64
  Running scriptlet: stunnel-5.58-1.amzn2023.0.2.x86_64
  Installing     : amazon-efs-utils-2.3.0-1.amzn2023.x86_64
  Running scriptlet: amazon-efs-utils-2.3.0-1.amzn2023.x86_64
  Verifying      : amazon-efs-utils-2.3.0-1.amzn2023.x86_64
  Verifying      : stunnel-5.58-1.amzn2023.0.2.x86_64

Installed:
  amazon-efs-utils-2.3.0-1.amzn2023.x86_64                                stunnel-5.58-1.amzn2023.0.2.x86_64

Complete!
[root@ip-172-31-89-247 ec2-user]# mkdir efs
[root@ip-172-31-89-247 ec2-user]# sudo mount -t efs -o tls fs-00cla35aa9d8afbfd:/ efs
[root@ip-172-31-89-247 ec2-user]# df -h
Filesystem      Size  Used Avail Use% Mounted on
devtmpfs        4.0M   0   4.0M   0% /dev
tmpfs           475M   0   475M   0% /dev/shm
tmpfs           190M  500K   190M   1% /run
/dev/xvda1       8.0G  1.6G   6.4G  20% /
tmpfs           475M   0   475M   0% /tmp
/dev/xvda128     10M   1.3M   8.7M  13% /boot/efi
tmpfs           95M   0    95M   0% /run/user/1000
127.0.0.1:/      8.0E   0   8.0E   0% /home/ec2-user/efs
[root@ip-172-31-89-247 ec2-user]#
```

Step5: Mount the File System to MyEC2-2 Instance

- Follow the exact same steps as in Step4, but for the MyEC2-2 instance.
- SSH into MyEC2-2.
- Switch to root user (`sudo -s`).
- Install EFS utilities (`yum install -y amazon-efs-utils`).
- Create the mount directory (`mkdir efs`).
- Use the same mount command you copied from the EFS console's Attach button.
- Verify with `df -h`.

Step6: Testing the File System

- Have two terminal windows open, one connected to MyEC2-1 and the other to MyEC2-2.
- In both terminals, switch to the root user (`sudo -s`) and navigate into the shared EFS directory:
`cd efs`
- In the MyEC2-1 terminal, create a test file: `touch hello.txt`
- List the files: `ls -ltr`
- You should see hello.txt.
- In the MyEC2-2 terminal, list the files without creating anything: `ls -ltr`
- You should see the hello.txt file appear automatically. This confirms both instances are sharing the same EFS file system

EFS (*Elastic File System*)

```
root@ip-172-31-91-46 ec2-user]# sudo -s
root@ip-172-31-91-46 ec2-user]# cd efs
root@ip-172-31-91-46 efs]# touch hallo.txt
root@ip-172-31-91-46 efs]# ls -lrt
total 4
-rw-r--r--. 1 root root 0 Jun 10 15:46 hallo.txt
root@ip-172-31-91-46 efs]# cd efs
bash: cd: efs: No such file or directory
root@ip-172-31-91-46 efs]# touch my-first-efs
root@ip-172-31-91-46 efs]# ls -l
total 8
-rw-r--r--. 1 root root 0 Jun 10 15:46 hallo.txt
-rw-r--r--. 1 root root 0 Jun 10 15:48 my-first-efs
root@ip-172-31-91-46 efs]#
```

```
[root@ip-172-31-91-46 ec2-user]# sudo -s
[root@ip-172-31-91-46 ec2-user]# cd efs
[root@ip-172-31-91-46 efs]# touch hallo.txt
[root@ip-172-31-91-46 efs]# ls -lrt
total 4
-rw-r--r--. 1 root root 0 Jun 10 15:46 hallo.txt
[root@ip-172-31-91-46 efs]# cd efs
bash: cd: efs: No such file or directory
[root@ip-172-31-91-46 efs]# touch my-first-efs
[root@ip-172-31-91-46 efs]# ls -l
total 8
-rw-r--r--. 1 root root 0 Jun 10 15:46 hallo.txt
-rw-r--r--. 1 root root 0 Jun 10 15:48 my-first-efs
[root@ip-172-31-91-46 efs]#
```