

# WHAT IS EFS? AMAZON ELASTIC FILE SYSTEM EXPLAINED



Amazon's Elastic File System (EFS) is a scalable storage solution that can be used for general purpose workloads. An EFS can be attached to multiple [Amazon Web Services \(AWS\) compute instances](#) and on-premises servers, providing a common resource for applications and data storage in many different environments.

Let's look at how an EFS functions and the benefits it provides organizational computing environments, including:

- What is an EFS?
- Advantages to using an EFS
- Disadvantages to using an EFS
- Use cases for EFS volumes
- Creating an EFS
- Mounting an EFS on an on-premises server
- EFS pricing
- Creating & using an EFS

*(This article is part of our [AWS Guide](#). Use the right-hand menu to navigate.)*

## What is an EFS?

An EFS is a Network File System (NFS) that organizes data in a logical file hierarchy. Data is stored in a path-based system, where data files are organized in folders and sub-folders.

Mapped file server drives and detachable USB drives both use hierarchical file systems, so the

concept should be familiar to anyone who has ever dealt with personal computers and servers.

EFSs are ideal candidates for storing:

- Organizational data
- File server
- Individual data
- Application data

Amazon states that a single EFS can be simultaneously connected to thousands of Elastic Compute Cloud (EC2) instances or on-premise resources, allowing you to share EFS data with as many resources as needed.

Access to shared EFS folders and data is provided through native operating system interfaces.

## Advantages of using AWS EFS

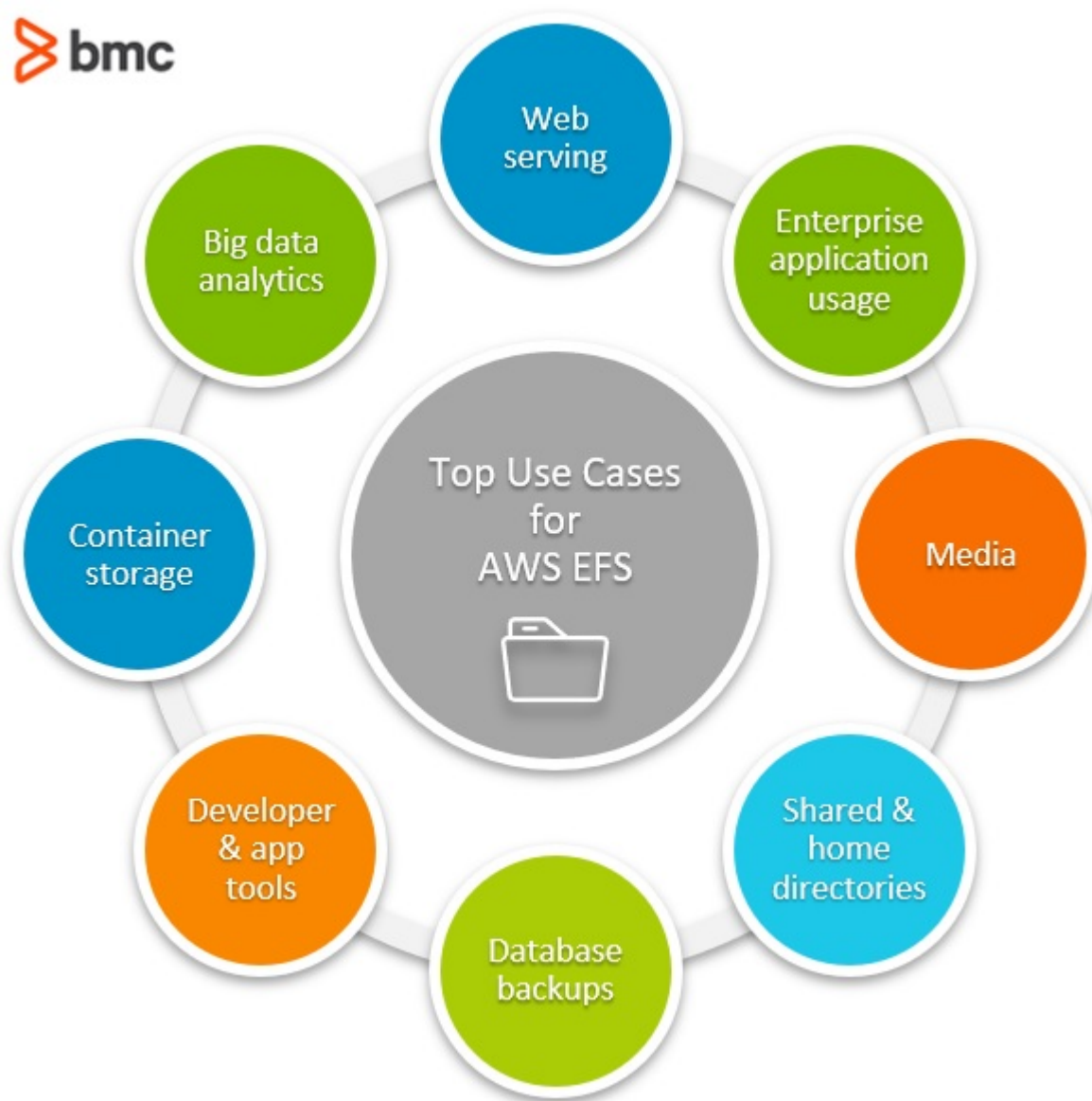
An Amazon EFS is elastic. That means its storage capacity can be automatically scaled up (add more storage) or scaled down (shrink storage capacity) as folders and files are added to or removed from the system. This is a major advantage over traditional storage solutions—you can add or remove capacity without disrupting users or applications.

Importantly, EFS storage is permanent. When attached to an AWS compute instance, data will not disappear when that instance is relaunched.

## Disadvantages of using Amazon EFS

Amazon EFSs do have a couple limitations:

- **No Windows instances.** Amazon EFSs are not supported on AWS Windows EC2 instances. EFS volumes can only be used with non-Windows instances, such as Linux, that support NFS volumes.
- **No system boot volumes.** Amazon EFS volumes also cannot be used for system boot volumes. AWS EC2 instances must use [Elastic Block Store \(EBS\) volumes](#) for booting their systems. EBS volumes are like EFS volumes with one exception. An EBS volume can only be connected to one EC2 instance or server, while EFS volumes can be connected to several EC2 instances and on-premises resources.



## Use cases for EFS volumes

An EFS is suitable for the following use cases:

- Web serving and content management
- Enterprise application usage
- Media and entertainment
- Shared and home directories
- Database backups
- Developer and application tools
- [Container storage](#)
- [Big data analytics](#)
- Other applications where you need to connect a common data source to a single server or multiple servers

## Creating an Elastic File System (EFS)

An EFS is created within an AWS Virtual Private Cloud (AWS VPC) and must be attached to EC2 instances within the same VPC.

All the resources associated with an EFS—VPC, EC2 instances, and the EFS itself—must reside in the [same AWS region](#). To host an EFS, you can use a default VPC or a custom VPC.

(Learn how VPCs work in our [AWS VPCs introduction](#).)

You can create and manage an EFS using:

- The AWS Management Console
- An AWS Command Line Interface (CLI)
- AWS API interfaces

Setup is relatively easy. AWS EC2 instances can mount existing EFSs to store and access data.

## Mounting an EFS on an on-premises server

You can also mount an EFS on an on-premises server using AWS Direct Connect or AWS VPN. Using an EFS from on-premises servers allows you to:

- Migrate data to EFS
- Backup on-premises storage
- Perform additional tasks

## Amazon EFS backup options

Backing up Amazon EFS is an essential practice. Doing so ensures your data is protected and you stay in compliance with protocols and regulations, and if disaster strikes, you can quickly recover. There are several EFS backup options:

### 1. AWS DataSync

When you establish this always-on connection between data stored on site and Amazon EFS, you can move data between them with ease. It works for data migrations, replication, and recovery.

### 2. AWS Backup

Using Amazon's managed backup services is another option. You can configure AWS resources and automate backups when they need to occur.

### 3. AWS Transfer Family

These services include Secure File Transfer Protocol, File Transfer Protocol over SSL, and File Transfer Protocol. You can use them to move files in and out of Amazon EFS.

### 4. EFS-to-EFS backup

When you want to execute incremental backups, this service is ideal. Define resources as code so that it is easy to configure, use, and manage many AWS resources.

### 5. Amazon S3

S3 is short for Simple Storage Service. Use it to store data as objects within many storage containers.

It supports automatic scaling for ever larger collections of data and large size objects.

## 6. Amazon Glacier

This service gives you secure storage for long-term data archiving and backups. It is best when you don't need to access data often, but you want a durable repository for your archives.

## EFS vs. EBS

You will find important distinctions between Amazon EFS and Amazon EBS.

### Shared file access

- **EFS:** You can share file access across multiple instances.
- **EBS:** For when you need separate storage that is accessible by a single instance at a time.

### Access:

- **EFS:** You can have multiple instances access the storage system simultaneously.
- **EBS:** You can only have a single instance access the storage system at a time.

### Throughput

- **EFS:** For when you want to deliver aggregate throughput to thousands of clients at the same time.
- **EBS:** Provides high availability and low latency.

## EFS pricing

EFS File system throughput scales automatically as capacity grows. Throughput can also be provisioned independent of capacity, if needed.

EFS storage can be provisioned under several different AWS pricing models, depending on:

- Whether your EFS data will be accessed frequently or infrequently
- Whether an EFS will store data in one or several AWS zones
- Whether you are using throughput provisioning
- What AWS storage class is used

## How to practice creating & using an EFS

AWS offers a [Getting Started exercise](#) where you can create a sample Amazon Elastic File System, attach it to an EC2 instance, and transfer files to your new EFS using AWS DataSync (AWS account required).

This exercise also provides instructions for deleting the AWS sample resources you create, so your account is not charged after your EFS testing.

## Create file system

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

**Name - optional**  
Name your file system.

test\_efs

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](#)

vpc-a431edd9  
default

**Availability and Durability**  
Choose Regional (recommended) to create a file system using regional storage classes. Choose One Zone to create a file system using One Zone storage classes. [Learn more](#)

☒ **Regional**  
Stores data redundantly across multiple AZs

☐ **One Zone**  
Stores data redundantly within a single AZ

Cancel

Customize

Create

An Elastic File System can be created through the AWS Management Console

For more information on using an Amazon EFS for your data needs, consult the [Amazon site](#) directly.

## Related reading

- [BMC Multi-Cloud Blog](#)
- [An Introduction to AWS Certifications](#)
- [The AWS Well-Architected Framework: 5 Pillars & Best Practices](#)
- [Cloud Growth, Trends & Outlook](#)
- [How to Run Self Hosted Azure DevOps Build/Release Agents](#)