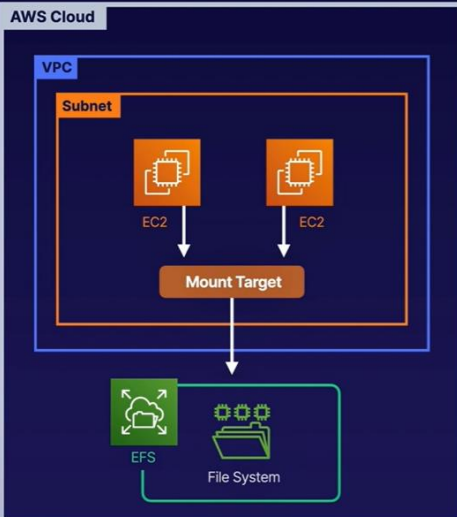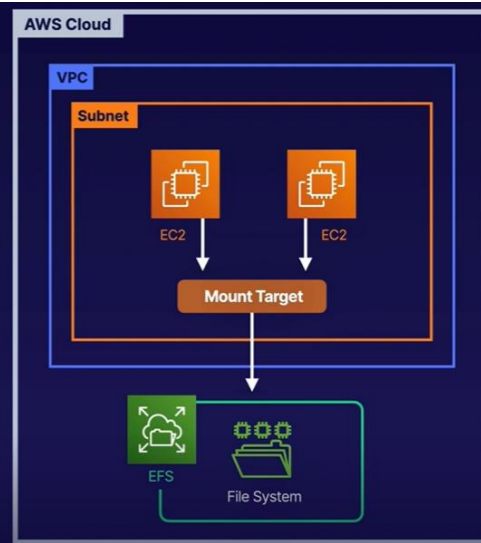EFS overview:

Amazon Elastic File System

EFS

✓ Managed NFS (network file system) that can be mounted on many EC2 instances.

✓ EFS works with EC2 instances in multiple Availability Zones.

✓ Highly available and scalable; however, it is expensive.

EFS
Use Cases

Content Management

Great fit for content management systems, as you can easily share content between EC2 instances.

Web Servers

Also a great fit for web servers. Have just a single folder structure for your website.

EFS Overview

✓ Uses NFSv4 protocol

✓ Compatible with Linux-based AMI (Windows not supported at this time)

✓ Encryption at rest using KMS

✓ File system scales automatically; no capacity planning required

✓ Pay per use

EFS Performance

EFS has amazing performance capabilities

1000s
Concurrent Connections

EFS can support thousands of concurrent connections (EC2 instances).

10 Gbps
Throughput

EFS can handle up to 10 Gbps in throughput.

Petabytes
Scaling

Scale your storage to petabytes.

Controlling Performance

When creating an EFS file system, you can set what performance characteristics you want.

General Purpose

Used for things like web servers, CMS, etc.

Max I/O

Used for big data, media processing, etc.

Storage Tiers

EFS comes with storage tiers and lifecycle management, allowing you to move your data from one tier to another after X number of days.

Standard
For frequently accessed files

Infrequently Accessed
For files not frequently accessed

What to Remember about EFS

🏆 Supports the Network File System version 4 (NFSv4) protocol

🏆 Can support thousands of concurrent NFS connections

🔖 Only pay for the storage you use (no pre-provisioning required)

🔖 Data is stored across multiple AZs within a region

🛡 Can scale up to petabytes

🛡 Read-after-write consistency

💡 BONUS TIP

If you have a scenario-based question around **highly scalable shared storage using NFS**, think EFS.

⏭ **Next lesson**
FSx Overview

FSx Overview:

What Is FSx for Windows?

Amazon FSx for Windows File Server provides a **fully managed native Microsoft Windows file system** so you can **easily move** your Windows-based applications that require file storage to AWS.

AMAZON FSX IS BUILT ON WINDOWS SERVER.

How is FSx for Windows different from EFS?

FSx for Windows

- A managed Windows Server that **runs Windows Server Message Block** (SMB)-based file services.

- **Designed for Windows** and Windows applications.

- **Supports** AD users, access control lists, groups, and security policies, along with Distributed File System (DFS) namespaces and replication.

EFS

- A managed NAS filer for **EC2 instances based on Network File System** (NFS) version 4.

- One of the **first network file sharing protocols** native to Unix and Linux.

Amazon FSx for Lustre

A fully managed file system that is optimized for compute-intensive workloads

✓ High Performance Computing
✓ Machine Learning
✓ Media Data Processing Workflows
✓ Electronic Design Automation

FSx for Lustre Performance

With Amazon FSx, you can **launch and run a Lustre file system that can process massive datasets at up to hundreds of gigabytes per second** of throughput, millions of IOPS, and sub-millisecond latencies.

In the exam, you'll be given **different scenarios** and asked to choose whether you should use **EFS**, **FSx for Windows**, or **FSx for Lustre**.

1. **EFS:** When you need distributed, highly resilient storage for Linux instances and Linux-based applications.

2. **Amazon FSx for Windows:** When you need centralized storage for Windows-based applications, such as SharePoint, Microsoft SQL Server, Workspaces, IIS Web Server, or any other native Microsoft application.

3. **Amazon FSx for Lustre:** When you need high-speed, high-capacity distributed storage. This will be for applications that do high performance computing (HPC), financial modeling, etc. Remember that FSx for Lustre can store data directly on S3.

AMI EBS VS Instance Store:

5 Things You Can Base Your AMI On

- ✓ Region
- ✓ Operating system
- ✓ Architecture (32-bit or 64-bit)
- ✓ Launch permissions
- ✓ Storage for the root device (root device volume)

All AMIs are categorized as either backed by:

Amazon EBS

The root device for an instance launched from the AMI is an Amazon EBS volume created from an Amazon EBS snapshot.

Instance Store

The root device for an instance launched from the AMI is an instance store volume created from a template stored in Amazon S3.

♀ STUDY TIP

Instance Store Volumes

Instance store volumes are sometimes called ephemeral storage. Instance store volumes **cannot be stopped**. If the underlying host fails, you will lose your data. You can, however, reboot the instance without losing your data.

If you delete the instance, you will lose the instance store volume.

EBS Volumes

EBS-backed instances **can be stopped**. You will not lose the data on this instance if it is stopped. You can also reboot an EBS volume and not lose your data.

By default, the root device volume will be deleted on termination. However, you can tell AWS to keep the root device volume with EBS volumes.

AMIs: EBS vs. Instance Store

Instance store volumes are sometimes called ephemeral storage.

You can reboot both EBS and instance store volumes and you will not lose your data.

Instance store volumes **cannot be stopped**. If the underlying host fails, you will lose your data.

By default, both root volumes will be deleted on termination. However, with EBS volumes, you can tell AWS to keep the root device volume.

🔆 BONUS TIP

An AMI is just a blueprint for an EC2 instance.

EBS-backed instances **can be stopped**. You will not lose the data on this instance if it is stopped.