

AMAZON EFS: CREACIÓN DE UN SISTEMA DE FICHEROS NFS

Amazon Elastic File System (EFS) es un servicio completamente administrado, de alta disponibilidad y durabilidad, que proporciona un almacenamiento de ficheros totalmente elástico y sin servidor que permite compartir datos de archivos sin necesidad de aprovisionar o administrar la capacidad de almacenamiento y el rendimiento. Los sistemas de ficheros de Amazon EFS escalan automáticamente hasta PBytes de datos y, a nivel de productividad, permiten que miles de servidores, instancias o máquinas virtuales puedan acceder simultáneamente y de forma consistente.

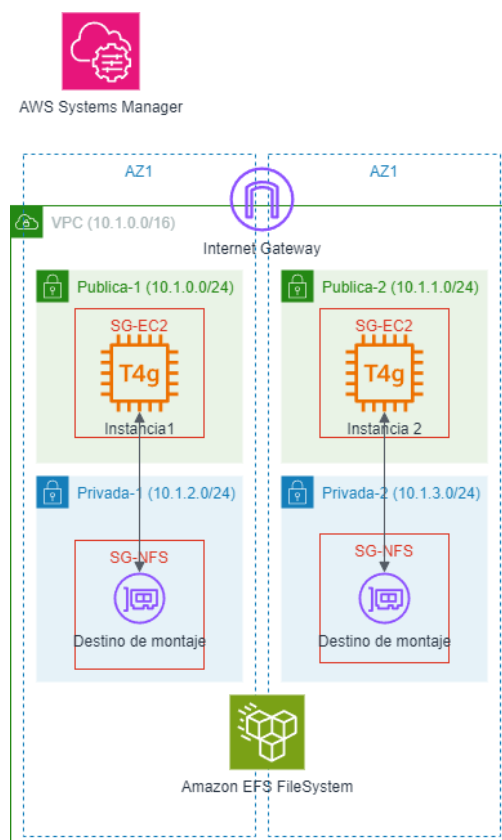
En esta práctica, exploraremos el servicio de Amazon EFS y aprenderemos cómo crear un sistema de archivos NFS (Network File System) utilizando Amazon EFS en una infraestructura de red prediseñada. Amazon EFS es un sistema de archivos elástico y sin servidor que nos permite crear y montar sistemas de archivos en instancias de Amazon EC2.

A continuación, veremos paso a paso cómo configurar y utilizar Amazon EFS, montándolo en dos instancias de Amazon EC2. También exploraremos cómo crear puntos de montaje en una VPC para acceder al sistema de archivos y cómo se proporciona una dirección IP estática y un nombre DNS para el punto de conexión NFSv4.

Requerimientos:

- Disponer de acceso a los recursos de AWS a través de un *sandbox* de AWS Academy
- Disponer de una máquina con la AWS CLI configurada con las credenciales de AWS Academy Learner Lab

Arquitectura propuesta:



Realización:**CONFIGURACIÓN DE LA INFRAESTRUCTURA DE RED**

- 1) El objetivo de la presente práctica no es configurar la infraestructura de red por lo que automatizaremos el despliegue de dicha infraestructura mediante el servicio de AWS CloudFormation. Para ello previamente, descargaremos los dos archivos *practica-efs.yaml* y *vpc.yaml* que se encuentran en la siguiente carpeta de un repositorio de GitHub:

<https://github.com/jose-emilio/aws-academy-fp-asir/tree/main/resources/efs>

- 2) Desde la AWS CLI, creamos un bucket de Amazon S3 para alojar nuestros recursos de AWS CloudFormation (el nombre del bucket no debe existir previamente; también podemos utilizar algún bucket de S3 que ya dispongamos de él) mediante la orden siguiente, sustituyendo el *placeholder* por el nombre del bucket:

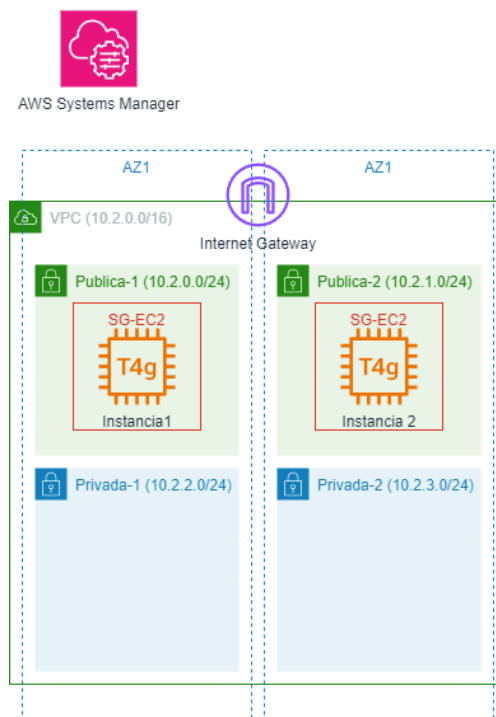
```
aws s3 mb s3://<nombre-bucket> --region us-east-1
```

- 3) A continuación, nos ubicamos en el directorio donde se encuentran los archivos descargados y ejecutamos las siguientes instrucciones para instruir al servicio AWS CloudFormation para que despliegue la infraestructura necesaria para la práctica:

```
aws cloudformation package --s3-bucket <nombre-bucket> --template-file practica-efs.yaml --output-template-file packaged.yaml --region us-east-1
```

```
aws cloudformation deploy --template-file packaged.yaml --stack-name efs-stack --region us-east-1
```

Tras esperar unos minutos, se habrá completado el despliegue que incluirá la siguiente arquitectura:

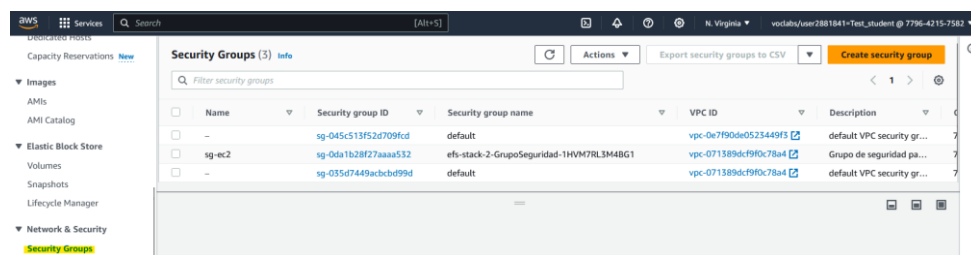


La arquitectura automatizada contiene los siguientes recursos:

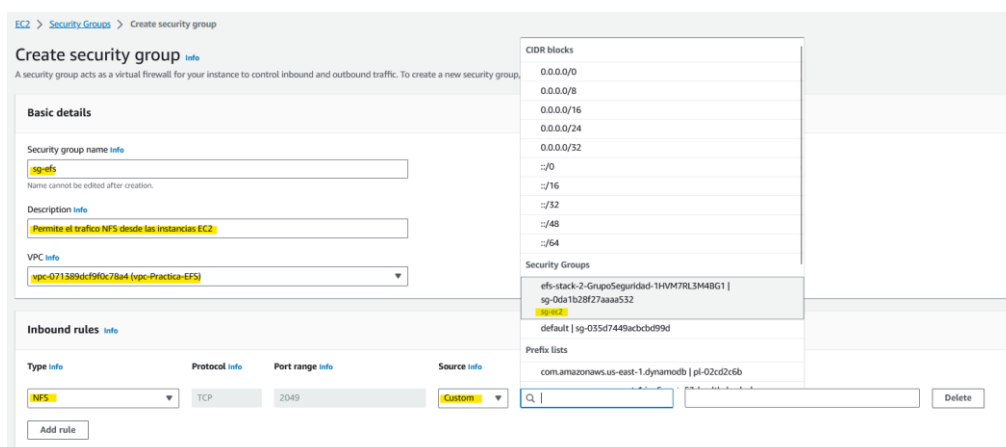
- Una red de Amazon VPC (*vpc-Practica-EFS*) con un bloque CIDR 10.1.0.0/16, que abarca dos zonas de disponibilidad, con una subred pública y una subred privada en cada zona de disponibilidad.
 - *Practica-EFS-publica-1*, bloque CIDR 10.1.0.0/24, *us-east-1a*
 - *Practica-EFS-publica-2*, bloque CIDR 10.1.2.0/24, *us-east-1b*
 - *Practica-EFS-privada-1*, bloque CIDR 10.1.1.0/24, *us-east-1a*
 - *Practica-EFS-privada-2*, bloque CIDR 10.1.3.0/24, *us-east-1b*
- Dos instancias EC2, llamadas *Instancia1* e *Instancia2*, desplegadas en cada una de las subredes públicas *Practica-EFS-publica-1* y *Practica-EFS-publica-2*
- Un grupo de seguridad *sg-ec2* asignado a las instancias EC2 anteriores, que no permite el tráfico de entrada y permite todo el tráfico de salida

CREACIÓN DEL SISTEMA DE FICHEROS MEDIANTE AMAZON EFS

- 4) Amazon EFS permite desplegar sistemas de ficheros administrados que pueden ser accedidos mediante el protocolo NFS (*Network File System*). Este protocolo hace que el servicio Amazon EFS escuche las solicitudes por el puerto 2049 TCP, por lo que procederemos previamente a crear un grupo de seguridad que permita el tráfico entrante por este puerto. Además, siguiendo las mejores prácticas de seguridad, sólo permitiremos el tráfico que provenga de nuestras instancias EC2. Para ello, desde la consola de Amazon EC2, accederemos al menú lateral **Network & Security / Security Groups** y presionaremos el botón **Create security group**:



- 5) En la pantalla de creación del grupo de seguridad introducimos su nombre (*efs-sg*), así como la descripción. También seleccionamos en el campo **VPC** el valor *vpc-Practica-EFS*. Añadimos una nueva regla de entrada en el apartado **Inbound rules**, seleccionando para el menú desplegable **Type** el valor *NFS* y seleccionando en el campo **Source** la opción *Custom* e indicando el grupo de seguridad etiquetado como *sg-ec2*, tal y como se muestra en la imagen siguiente:



Por último, en el apartado **Tags** añadimos una nueva etiqueta, indicando en el apartado **Key** el valor *Name* y en el apartado **Value** el valor *sg-efs*. Presionamos el botón **Create security group**.

Tags - optional
A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Key: Value - optional: Remove

Add new tag
You can add up to 49 more tags

Cancel Create security group

Tras esta operación, desde la opción del menú lateral **Network & Security / Security Groups** podremos comprobar que ya disponemos de los grupos de seguridad necesarios para la práctica:

Security Groups (4) Info Actions Export security groups to CSV Create security group

Filter security groups

	Name	Security group ID	Security group name	VPC ID	Description	Owner
<input type="checkbox"/>	-	sg-045c513f52d709fcd	default	vpc-0e7f90de0523449f3	default VPC security gr...	779642157582
<input type="checkbox"/>	-	sg-035d7449acbcd99d	default	vpc-071389dcf9f0c78a4	default VPC security gr...	779642157582
<input type="checkbox"/>	sg-ec2	sg-0da1b28f27aaaa532	efs-stack-2-GrupoSegu...	vpc-071389dcf9f0c78a4	Grupo de seguridad pa...	779642157582
<input type="checkbox"/>	sg-efs	sg-0fe189e42e8ea76e	efs-sg	vpc-071389dcf9f0c78a4	Permite el trafico NFS ...	779642157582

- 6) Ahora procedemos a configurar nuestro sistema de ficheros compartido. Para ello, desde la consola del servicio de Amazon EFS, accedemos a la opción **File Systems** del menú lateral, y presionamos el botón **Create file system**:

Elastic File System × Amazon EFS > File systems

File systems (0) View details Delete Create file system

Filter by property values

Name	File system ID	Encrypte d	Total size	Size in Standard / One Zone	Size in Standard-IA / One Zone-IA	Provisioned Throughput (MiB/s)	File system state
No resources							

Create file system

- 7) Desde la siguiente ventana, presionamos directamente el botón **Customize** sin introducir ningún otro valor:

Create file system ×

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

Name - optional
Name your file system.

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

Virtual Private Cloud (VPC)
Choose the VPC where you want EC2 instances to connect to your file system.

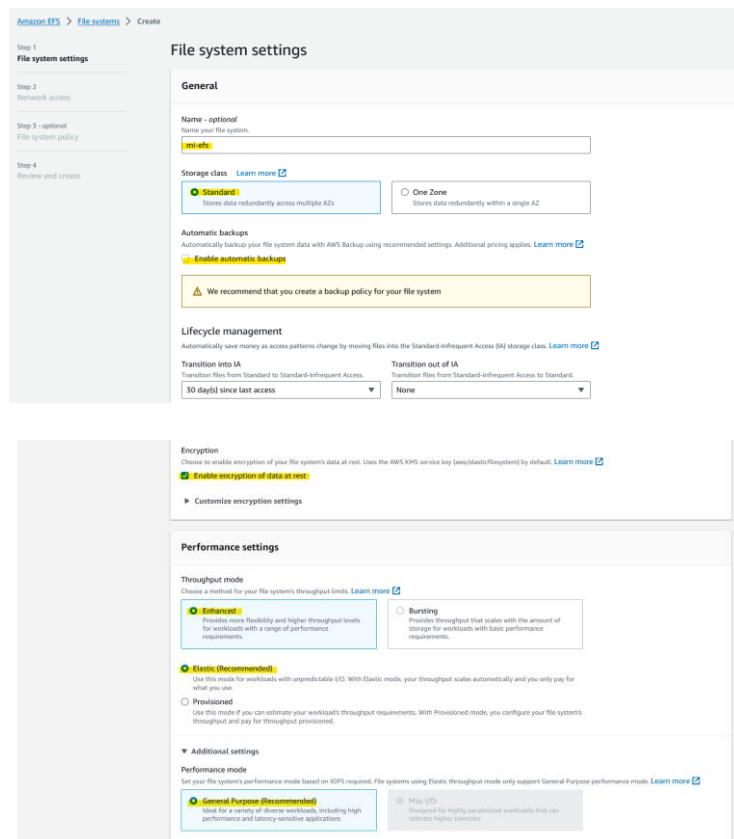
default

Cancel Customize Create

8) Se abrirá un asistente de 4 pasos que nos permitirá configurar nuestro sistema de ficheros en Amazon EFS. En el primer paso, aparecerá los siguientes campos:

- **Name:** Nombre del sistema de ficheros. En realidad, es una etiqueta opcional. En esta práctica introduciremos el valor *mi-efs*.
- **Storage class:** Permite elegir entre las opciones *Standard* y *One Zone*, en función de si deseamos que los datos almacenados se repliquen en más de una zona de disponibilidad o sólo en una zona de disponibilidad, respectivamente. En nuestro caso, elegiremos la opción *Standard*.
- **Automatic backups:** Permite realizar copias automatizadas de nuestro sistema de ficheros NFS utilizando el servicio AWS Backup. Desmarcaremos esta opción para la práctica.
- **Lifecycle management:** Amazon EFS dispone de dos capas de almacenamiento, una frecuente y otra infrecuente, con el objeto de poder reducir los costes de almacenamiento. Esta característica va a permitir transicionar los ficheros entre ambas capas de almacenamiento en función de su patrón de utilización. Para el desarrollo de esta práctica no modificaremos los valores por defecto.
- **Encryption:** Permite habilitar o deshabilitar el cifrado en reposo de los datos. Para ello utiliza el servicio de AWS KMS (*Key Management Service*). Dejaremos el valor por defecto.
- **Throughput mode:** Determina la productividad de nuestro sistema de ficheros. Existen tres modos diferentes:
 - **Elastic.** Se utiliza para cargas de trabajo impredecibles en las que es complicado prever los requerimientos de rendimiento. Es la opción recomendada en la mayor parte de los casos de uso.
 - **Provisioned.** Permite ajustar el rendimiento expresado en MiB/s. Es útil cuando se conocen los requerimientos de rendimiento de la carga de trabajo.
 - **Bursting.** Permite escalar la productividad para que escale con la cantidad de almacenamiento del sistema de ficheros

En nuestro caso, seleccionaremos la opción *Elastic*.



The screenshot displays the 'File system settings' page in the Amazon EFS console, showing the configuration steps and options for creating a new file system.

Step 1: File system settings

General

- Name - optional:** Name your file system.
- Storage class:** [Learn more](#)
 - ☒ **Standard** (Selected): Stores data redundantly across multiple AZs.
 - ☐ **One Zone**: Stores data redundantly within a single AZ.
- Automatic backups:** Automatically backup your file system data with AWS Backup using recommended settings. Additional pricing applies. [Learn more](#)
 - ☐ **Enable automatic backups**
 - We recommend that you create a backup policy for your file system**
- Lifecycle management:** Automatically save money as access patterns change by moving files into the Standard-Infrequent Access (IA) storage class. [Learn more](#)
 - Transition into IA:** Transition files from Standard to Standard-Infrequent Access.
 - Transition out of IA:** Transition files from Standard-Infrequent Access to Standard.

Encryption

Choose to enable encryption of your file system's data at rest. Uses the AWS KMS service key (aws/elasticfilesystem) by default. [Learn more](#)

- ☒ **Enable encryption of data at rest:**
- [Customize encryption settings](#)

Performance settings

Throughput mode

Choose a method for your file system's throughput limits. [Learn more](#)

- ☒ **Enhanced** (Selected): Provides more flexibility and higher throughput levels for workloads with a range of performance requirements.
- ☐ **Elastic (Recommended)**: Use this mode for workloads with unpredictable I/O. With Elastic mode, your throughput scales automatically and you only pay for what you use.
- ☐ **Provisioned**: Use this mode if you can estimate your workload's throughput requirements. With Provisioned mode, you configure your file system's throughput and pay for throughput provisioned.
- ☐ **Bursting**: Provides throughput that scales with the amount of storage for workloads with basic performance requirements.

Additional settings

Performance mode

Set your file system's performance mode based on IOPS required. File systems using Elastic throughput mode only support General Purpose performance mode. [Learn more](#)

- ☒ **General Purpose (Recommended)**: Ideal for a variety of diverse workloads, including high performance and latency-sensitive applications.
- ☐ **Max I/O**: Designed for highly parallelized workloads that can tolerate higher latencies.

Por último, presionamos el botón **Next**.

- 9) En el segundo paso del asistente, debemos configurar la información de la VPC donde se desplegará nuestro sistema de ficheros. Para ello, en el apartado **Virtual Private Cloud (VPC)** seleccionamos la VPC creada de forma automática en el apartado **3)** de la práctica, llamada *vpc-Practica-EFS*. Desde el apartado **Mount targets** debemos asegurar que, en el subapartado **Subnet ID** de cada zona de disponibilidad, están marcadas las subredes privadas (*Practica-EFS-privada-1* y *Practica-EFS-privada-2*). Además, en el subapartado **Security groups** debemos eliminar el grupo de seguridad *default* y añadir nuestro grupo de seguridad etiquetado como *efs-sg*, tal y como se muestra en las siguientes figuras:

Una vez configurada esta sección, presionamos el botón **Next**.

- 10) En el tercer paso del asistente, podemos configurar una política basada en el recurso (política de sistema de ficheros) para nuestro recurso de Amazon EFS. Esta política permite controlar el acceso a los datos de nuestro sistema de ficheros utilizando permisos de AWS IAM (*Identity and Access Management*). Por defecto la política no utiliza IAM para la autenticación y otorga permisos

completos a cualquier cliente anónimo que pueda conectar con el sistema de ficheros utilizando un destino de montaje (*mount target*). En nuestro caso, para simplificar la realización de esta práctica, no configuraremos la política. Presionamos el botón **Next**.

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

Grant file system permissions to additional AWS IAM principals. [Learn more](#)

Policy editor (JSON)

Clear

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

- 11) Por último, en el paso 4 podremos revisar la configuración elegida. Para terminar y proceder a la creación del sistema de ficheros, presionamos el botón **Create**. En cuestión de pocos segundos, nuestro sistema de fichero ya estará creado y podremos visualizarlo desde la consola de Amazon EFS:

File systems (1)

Filter by property values

	Name	File system ID	Encrypted	Total size	Size in Standard / One Zone	Size in Standard-IA / One Zone-IA	Provisioned Throughput (MiB/s)	File system state	Creation time	Availability Zone
<input type="radio"/>	mi-efs	fs-0fcb538da7525d5b0	<input checked="" type="checkbox"/> Encrypted	6.00 KiB	6.00 KiB	0 Bytes	-	<input checked="" type="checkbox"/> Available	Wed, 01 Nov 2023 11:05:44 GMT	Standard

- 12) Para comprobar que la configuración de red se ha implementado correctamente, accedemos desde la ventana anterior al enlace indicado en el nombre del sistema de ficheros y activamos la pestaña **Network**, tal y como se muestra en la imagen:

Amazon EFS > File systems > fs-0fcb538da7525d5b0

mi-efs (fs-0fcb538da7525d5b0)

Delete Attach

Edit

General

Performance mode
General Purpose

Throughput mode
Elastic

Lifecycle management
Transition into IA: 30 day(s) since last access
Transition out of IA: None

Availability zone
Standard

Automatic backups
☐ Disabled

Encrypted
34fb1c8b-4bf2-4d1c-b120-f9721278eb7b (aws/elasticfilesystem)

File system state
☒ Available

DNS name
[fs-0fcb538da7525d5b0.efs.us-east-1.amazonaws.com](#)

Metered size | Monitoring | Tags | File system policy | Access points | **Network** | Replication

Network

Manage

Availability zone	Mount target ID	Subnet ID	Mount target state	IP address	Network interface ID	Security groups
us-east-1a	fsmt-010b1ed3119220ea	subnet-01efe3b5433e37fe9	<input checked="" type="checkbox"/> Available	10.1.1.168	eni-088af7a867c157db3	sg-0fe189e42e8eaa76e (efs-sg)
us-east-1b	fsmt-02364f468bb6c93aa	subnet-01cf9066eaa737060	<input checked="" type="checkbox"/> Available	10.1.3.26	eni-0d8f5256529683376	sg-0fe189e42e8eaa76e (efs-sg)

Desde la ventana anterior, podemos comprobar que se ha creado un destino de montaje (*mount target*) en cada una de las dos zonas de disponibilidad, en concreto, dentro de las subredes privadas *Practica-EFS-privada-1* y *Practica-EFS-privada-2*. Además, cada punto de montaje está implementado mediante un **Interfaz de Red Elástico** (*ENI*, *Elastic Network Interface*) que tiene asignada una IP privada dentro del bloque CIDR de la subred correspondiente.

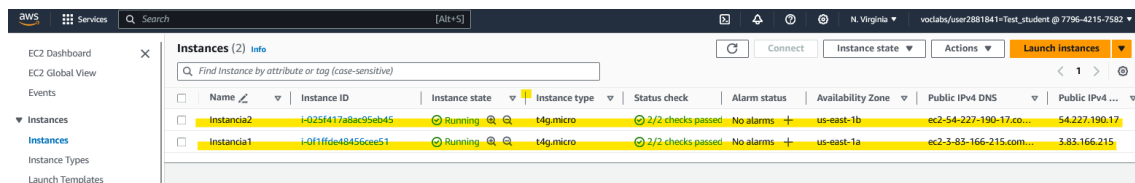
Además, se ha asignado un nombre DNS a nuestro sistema de ficheros. Este nombre DNS se resolverá de forma privada, condicionalmente desde la subred de donde se resuelva, es decir, los clientes ubicados en la zona de disponibilidad *us-east-1* resolverán a la IP del punto de montaje ubicado en dicha zona de disponibilidad, y así, sucesivamente.

Copiamos el nombre DNS del sistema de ficheros (o el ID del sistema de ficheros, es indiferente) ya que lo utilizaremos en los siguientes pasos.

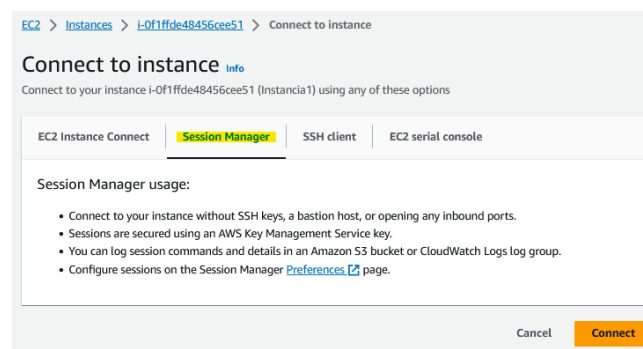
MONTAJE DEL SISTEMA DE FICHEROS EN INSTANCIAS EC2

En este apartado se procederá al montaje del sistema de ficheros en las instancias EC2 con sistema operativo Amazon Linux. Además, se creará un **punto de acceso** sobre el sistema de ficheros creado para hacer que una de las instancias EC2 tenga limitado el acceso a uno de los directorios del sistema de ficheros de Amazon EFS.

- 13) Accedemos a la consola de Amazon EC2. En ella podremos comprobar que hay dos instancias EC2 iniciadas:



- 14) Desde la pantalla anterior seleccionamos la instancia etiquetada como *Instancia1* y presionamos el botón **Connect**. Tras ello, nos ubicamos en la pestaña **Session Manager** y presionamos el botón **Connect**:



- 15) Desde el intérprete de órdenes, instalamos previamente el paquete *amazon-efs-utils*, que nos permitirá montar de una forma sencilla nuestro sistema de ficheros EFS. Para ello ejecutamos la orden:

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


Session ID: user2881841=Test_student-04f1a05964f70cc9c Instance ID: i-0f1ffde48456cee51

Terminate

```
sh-4.2$ sudo yum install -y amazon-efs-utils
Loaded plugins: extras_suggestions, langpacks, priorities, update-modd
amazon-core | 3.6 kB 00:00:00
Resolving Dependencies
--> Running transaction check
--> Package amazon-efs-utils.noarch 0:1.35.0-1.amzn2 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

=====================================================================================================================================
Package Arch Version Repository Size
=====================================================================================================================================
Installing:
amazon-efs-utils noarch 1.35.0-1.amzn2 amazon-core 57 k

Transaction Summary
=====================================================================================================================================
Install 1 Package

Total download size: 57 k
Installed size: 216 k
Downloading packages:
amazon-efs-utils-1.35.0-1.amzn2.noarch.rpm | 57 kB 00:00:00
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
Installing : amazon-efs-utils-1.35.0-1.amzn2.noarch 1/1
Verifying : amazon-efs-utils-1.35.0-1.amzn2.noarch 1/1

Installed:
amazon-efs-utils.noarch 0:1.35.0-1.amzn2

Complete!
sh-4.2$
```

- 16) A continuación, creamos el directorio `/mnt/efs` y montamos nuestro sistema de ficheros EFS en dicho directorio. Para ello ejecutamos las siguientes órdenes, sustituyendo el *placeholder* por el nombre DNS del sistema de ficheros EFS:

```
$ fs=<nombre-dns-efs>
$ sudo mkdir /mnt/efs
$ sudo mount -t efs -o tls $fs:/ /mnt/efs
```

Session ID: user2881841=Test_student-04f1a05964f70cc9c Instance ID: i-0f1ffde48456cee51

Terminate

```
sh-4.2$ fs=fs-0fch53de7525d5b0.efs.us-east-1.amazonaws.com
sh-4.2$ sudo mkdir /mnt/efs
sh-4.2$ sudo mount -t efs -o tls $fs:/ /mnt/efs
sh-4.2$
```

Tras el proceso anterior, el sistema EFS estará preparado para que nuestra instancia EC2 pueda acceder a él.

- 17) A continuación, introduciremos datos en el sistema de ficheros, para ello crearemos un directorio llamado *instancia2* e introduciremos archivos tanto dentro de dicho directorio como en el directorio raíz del sistema EFS. Ejecutamos las siguientes órdenes:

```
$ sudo mkdir /mnt/efs/instancia2
$ sudo bash -c "echo 'Fichero 1. Sólo accesible por la instancia1' > /mnt/efs/fichero1.txt"
$ sudo bash -c "echo 'Fichero 2. Accesible por ambas instancias' > /mnt/efs/instancia2/fichero2.txt"
```

Session ID: user2881841=Test_student-04f1a05964f70cc9c Instance ID: i-0f1ffde48456cee51

Terminate

```
sh-4.2$ sudo mkdir /mnt/efs/instancia2
sh-4.2$ sudo bash -c "echo 'Fichero 1. Sólo accesible por la instancia1' > /mnt/efs/fichero1.txt"
sh-4.2$ sudo bash -c "echo 'Fichero 2. Accesible por ambas instancias' > /mnt/efs/instancia2/fichero2.txt"
sh-4.2$
```

- 18) Para visualizar la estructura de ficheros creada en el sistema de ficheros EFS ejecutamos la orden siguiente:

```
$ tree /mnt/efs
```

Session ID: user2881841=Test_student-04f1a05964f70cc9c Instance ID: i-0f1ffde48456cee51

```
sh-4.2$ tree /mnt/efs
/mnt/efs
├── fichero1.txt
└── instancia2
    └── fichero2.txt

1 directory, 2 files
sh-4.2$
```

- 19) A continuación, vamos a montar el sistema de ficheros de Amazon EFS en la instancia etiquetada como *Instancia2*. Sin embargo, en esta ocasión, deseamos que el punto de montaje del sistema de ficheros en esta instancia tome como raíz el directorio llamado *instancia2*, de forma que nuestra instancia sólo tendrá visibilidad sobre dicha rama del sistema de ficheros. Para ello, procedemos a abrir una sesión contra la *Instancia2* mediante AWS SSM Session Manager, tal y como se realizó en el apartado 14). Desde aquí volvemos a instalar los paquetes necesarios, tal y como se explicó en el apartado 15):

Session ID: user2881841=Test_student-02a2c97da5f512f6d Instance ID: i-025f417a8ac95eb45 Terminate

```
sh-4.2$ sudo yum install -y amazon-efs-utils tree
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core
Resolving Dependencies
--> Running transaction check
--> Package amazon-efs-utils.noarch 0:1.35.0-1.amzn2 will be installed
--> Package tree.aarch64 0:1.6.0-10.amzn2.0.1 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

Package Arch Version Repository Size
Installing:
amazon-efs-utils noarch 1.35.0-1.amzn2 amzn2-core 57 k
tree aarch64 1.6.0-10.amzn2.0.1 amzn2-core 47 k

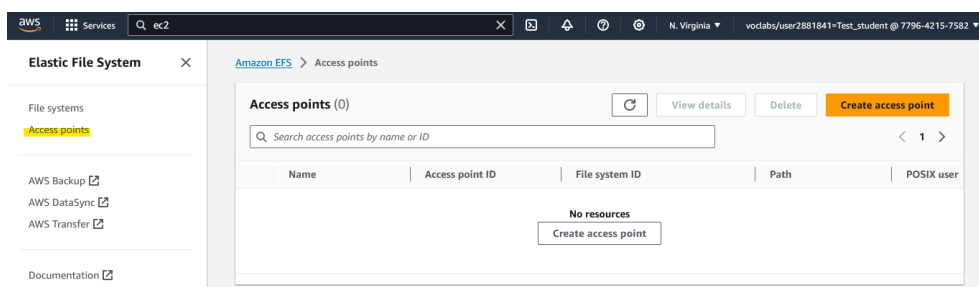
Transaction Summary
Install 2 Packages

Total download size: 103 k
Installed size: 311 k
Downloading packages:
(1/2): amazon-efs-utils-1.35.0-1.amzn2.noarch.rpm | 57 kB 00:00:00
(2/2): tree-1.6.0-10.amzn2.0.1.aarch64.rpm | 47 kB 00:00:00
Total 1.2 MB/s | 103 kB 00:00:00
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
Installing : amazon-efs-utils-1.35.0-1.amzn2.noarch 1/2
Installing : tree-1.6.0-10.amzn2.0.1.aarch64 2/2
Verifying : tree-1.6.0-10.amzn2.0.1.aarch64 1/2
Verifying : amazon-efs-utils-1.35.0-1.amzn2.noarch 2/2

Installed:
amazon-efs-utils.noarch 0:1.35.0-1.amzn2 tree.aarch64 0:1.6.0-10.amzn2.0.1

Complete!
sh-4.2$
```

- 20) En esta ocasión, no podemos montar directamente el sistema de ficheros, ya que la *Instancia2* requiere una visión parcial, limitada al directorio *instancia2*. Para ello, es necesario crear desde el servicio Amazon EFS un **punto de acceso** sobre nuestro sistema de ficheros. Vamos a la consola de Amazon EFS y, desde el menú lateral elegimos la opción **Access points** y presionamos el botón **Create access point**, tal y como se muestra en la imagen:



- 21) En la siguiente pantalla, elegimos en el campo **File system** el sistema de ficheros creado, etiquetado como *mi-efs*. Introducimos en el campo **Name** el valor “*Punto de acceso para Instancia2*” y en el campo **Root directory path** indicamos el valor */instancia2*, tal y como se muestra en la imagen.

El grupo de opciones **POSIX user** permite determinar la identidad POSIX en el punto de acceso que se utilizar para todas las operaciones a nivel de fichero realizadas por los clientes NFS. El grupo de opciones **Root directory creation permissions** instruirá al servicio Amazon EFS para que cree el directorio raíz especificado en el punto de acceso con los permisos indicados en caso de que el directorio no exista. Dejaremos todas las opciones en sus valores por defecto.

Por último, presionamos el botón **Create access point**.

En breves instantes se habrá creado el punto de acceso y podremos visualizarlo en la consola de Amazon EFS.

Name	Access point ID	File system ID	Path
Punto de acceso para Instancia2	fsap-0a47323c8e3dfe98a	fs-0fcb538da7525d5b0	/instancia2

- 22) A continuación, copiamos el ID del punto de acceso asignado por el servicio Amazon EFS. Volvemos a la consola de la *Instancia2* y montamos nuestro sistema de ficheros EFS, pero en esta ocasión utilizando el punto de acceso mediante las siguientes órdenes, sustituyendo los *placeholders* por el nombre DNS del sistema de ficheros y el ID del punto de acceso, respectivamente:

```
$ fs=<nombre-dns-efs>
$ fsap=<ID-punto-acceso>
$ sudo mkdir /mnt/efs
$ sudo mount -t efs -o tls,accesspoint=$fsap $fs:/ /mnt/efs
```

Session ID: user2881841=Test_student-02a2c97da5f512f6d Instance ID: i-025f417a8ac95eb45

Terminate

```
sh-4.2$ fs=fs-0fcb538da7525d5b0.efs.us-east-1.amazonaws.com
sh-4.2$ fsap=fsap-0a47323c8e3dfe98a
sh-4.2$ sudo mkdir /mnt/efs
sh-4.2$ sudo mount -t efs -o tls,accesspoint=$fsap $fs:/ /mnt/efs
sh-4.2$
```

- 23) Si accedemos ahora al directorio */mnt/efs*, podremos comprobar que sobre dicho directorio local se ha montado el directorio */instancia2* del sistema de ficheros EFS:

Session ID: user2881841=Test_student-02a2c97da5f512f6d Instance ID: i-025f417a8ac95eb45

Terminate

```
sh-4.2$ ls -l /mnt/efs/
total 4
-rw-r--r-- 1 root root 42 Nov  2 09:53 fichero2.txt
sh-4.2$
```

- 24) La solución adoptada para montar el sistema de ficheros EFS en las instancias EC2 es temporal, es decir, si reiniciásemos las instancias EC2 habría que montar de nuevo el sistema de ficheros. Para hacer que el sistema de ficheros EFS se monte cada vez que se inician las instancias EC2, es necesario añadir una línea en el archivo `/etc/fstab`. En el caso de la primera instancia añadiremos una línea de configuración mediante la siguiente orden:

```
$ sudo bash -c "echo '$fs:/mnt/efs nfs4
nfsvers=4.1,rsize=1048576,hard,timeo=600,retrans=2,noresvport,_netdev 0 0dev' >>
/etc/fstab"
```

Tras la ejecución anterior podremos visualizar el archivo `/etc/fstab` y comprobar su contenido mediante la orden:

```
cat /etc/fstab
```

Session ID: user2881841=Test_student-03cf94e1ea931bd10 Instance ID: i-0f1ffde48456cee51

Terminate

```
sh-4.2$ cat /etc/fstab
#
UUID=05d38dcb-9526-442a-bc49-ed462a3ad823 / xfs defaults,noatime 1 1
UUID=9e1d-3b7a /boot/efi vfat defaults,noatime,uid=0,gid=0,umask=0077,shortname=winnt 0 0
fs-0fcb538da7525d5b0.efs.us-east-1.amazonaws.com:/mnt/efs nfs4 nfsvers=4.1,rsize=1048576,hard,timeo=600,retrans=2,noresvport,_netdev 0 0dev
sh-4.2$
```

Si, a continuación, reiniciamos la instancia con la orden `sudo reboot`, podremos comprobar que el sistema de ficheros EFS se ha montado automáticamente:

Session ID: user2881841=Test_student-037b9ee9e46714cdb Instance ID: i-0f1ffde48456cee51

```
sh-4.2$ tree /mnt/efs/
/mnt/efs/
├── fichero1.txt
└── instancia2
    └── fichero2.txt

1 directory, 2 files
sh-4.2$
```

- 25) En el caso de la *Instancia2*, al estar montado el sistema de ficheros EFS mediante un punto de acceso, la sintaxis es ligeramente diferente. Para ello, ejecutamos la siguiente orden:

```
$ sudo bash -c "echo '$fs /mnt/efs efs _netdev,tls,accesspoint=$fsap 0 0' >> /etc/fstab"
```

Session ID: user2881841=Test_student-0bcca964967b35a8a Instance ID: i-025f417a8ac95eb45

Terminate

```
sh-4.2$ fs=fs-0fcb538da7525d5b0.efs.us-east-1.amazonaws.com
sh-4.2$ fsap=fsap-0a47323c8e3dfa98a
sh-4.2$ sudo bash -c "echo '$fs /mnt/efs efs _netdev,tls,accesspoint=$fsap 0 0' >> /etc/fstab"
sh-4.2$ cat /etc/fstab
#
UUID=05d38dcb-9526-442a-bc49-ed462a3ad823 / xfs defaults,noatime 1 1
UUID=9e1d-3b7a /boot/efi vfat defaults,noatime,uid=0,gid=0,umask=0077,shortname=winnt 0 0
fs-0fcb538da7525d5b0.efs.us-east-1.amazonaws.com /mnt/efs efs _netdev,tls,accesspoint=fsap-0a47323c8e3dfa98a 0 0
sh-4.2$
```

- 26) Si reiniciamos la *Instancia2* y posteriormente abrimos una sesión, podremos comprobar que se ha vuelto a montar automáticamente el sistema de ficheros EFS utilizando el punto de acceso:

Session ID: user2881841=Test_student-0176c89f1fdd262f3

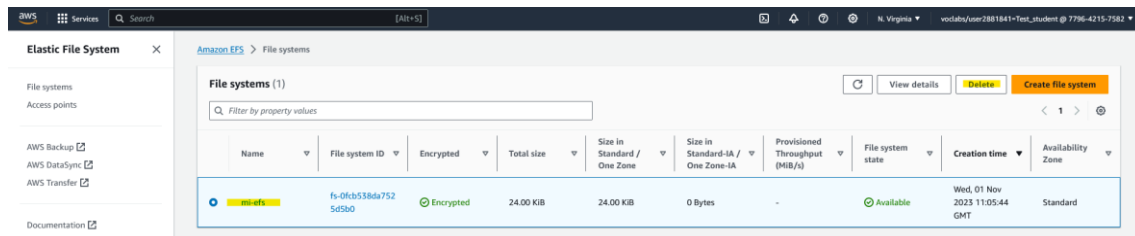
Instance ID: i-025f417a8ac95eb45

```
sh-4.2$ tree /mnt/efs/  
/mnt/efs/  
└── fichero2.txt  
  
0 directories, 1 file  
sh-4.2$
```

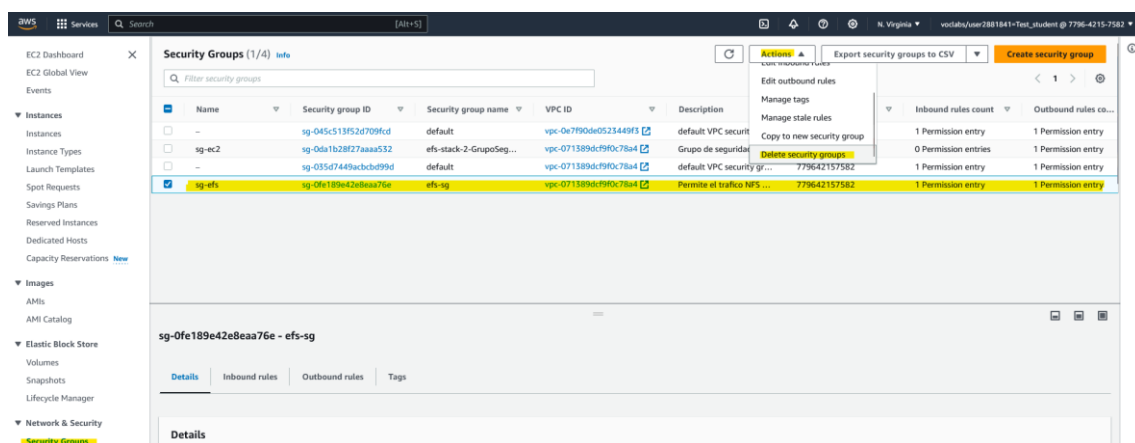
Limpieza de la Práctica:

Para terminar esta práctica y liberar los recursos creados, evitando así el consumo de créditos de AWS Academy Learner Labs, simplemente debemos dar los siguientes pasos:

- **Eliminar el sistema de ficheros de Amazon EFS.** Para ello, desde la consola del servicio Amazon EFS seleccionamos el sistema de ficheros y, presionamos el botón **Delete**.



- **Eliminar únicamente el grupo de seguridad asignado al sistema de ficheros EFS.** Para ello, desde la consola del servicio de Amazon EC2 accedemos al menú lateral **Network & Security / Security Groups**, seleccionando el grupo de seguridad etiquetado como *sg-efs* y presionar el botón **Actions / Delete security groups**.



- **Desaprovisionar la infraestructura de red.** Para ello utilizaremos el servicio AWS CloudFormation. Accedemos a la consola de nuestra máquina local y ejecutamos la orden siguiente de la AWS CLI:

```
aws cloudformation delete-stack --stack-name efs-stack --region us-east-1
```

(NOTA: La instrucción anterior no devuelve resultado alguno, pero en cuestión de pocos minutos nuestra infraestructura de red e instancias EC2 se habrán desaprovisionado)