

Desafío número 3

Josue Reyes

Objetivos del desafío:

Gestión de Cuentas y Usuarios

Creación y Configuración de Cuentas:

- Proceso de crear una cuenta en AWS, fundamental para cualquier trabajo en la nube.

Gestión de Identidades y Accesos (IAM):

- Creación de usuarios con permisos específicos.
- Otorgar permisos de administrador mediante políticas de acceso (`admin full access`).
- Importancia de iniciar sesión con diferentes usuarios para tareas específicas.

Organización y Etiquetado de Recursos

Uso de Tags:

- Cómo definir y aplicar tags para organizar y gestionar recursos.
- Importancia de etiquetar recursos con información relevante como propietario, equipo y proyecto.

Implementación y Configuración de Servicios Específicos

Amazon EC2:

- Lanzamiento de instancias EC2 dentro de los límites del free tier.
- Automatización de configuraciones iniciales usando scripts en `User Data`.
- Configuración de Security Groups para gestionar el tráfico de red.

Conexión Remota a Instancias:

- Configuración y verificación de conexiones remotas utilizando SSM, llaves SSH o desde una VM con Linux.

Amazon S3:

- Creación de buckets con nombres únicos.
- Subida y gestión de archivos en un bucket S3.
- Verificación de la funcionalidad del bucket y de los archivos subidos.

Amazon EBS:

- Creación y vinculación de volúmenes EBS a instancias EC2.
- Configuración del sistema de archivos (`ext4`), incluyendo formateo y montaje.
- Actualización del `FSTAB` para montaje automático y verificación de la capacidad de escritura en el volumen.

Operaciones Básicas y Buenas Prácticas

Automatización y Scripting:

- Uso de scripts para instalar software y realizar configuraciones iniciales.
- Descarga de archivos desde S3 usando herramientas como AWS CLI o `wget`.

Configuración de Seguridad:

- Configuración de reglas de seguridad para gestionar acceso a los recursos.
- Importancia de asegurar conexiones remotas y permisos adecuados.

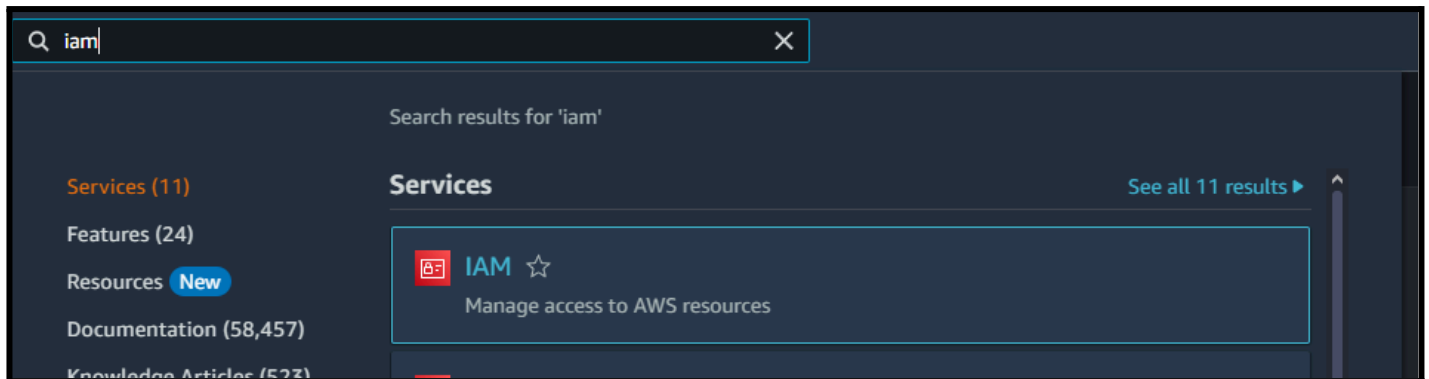
Gestión de Almacenamiento:

- Montaje y verificación de sistemas de archivos.
- Movilidad de datos entre diferentes servicios (por ejemplo, desde S3 a un volumen EBS).

Creación del usuario IAM.

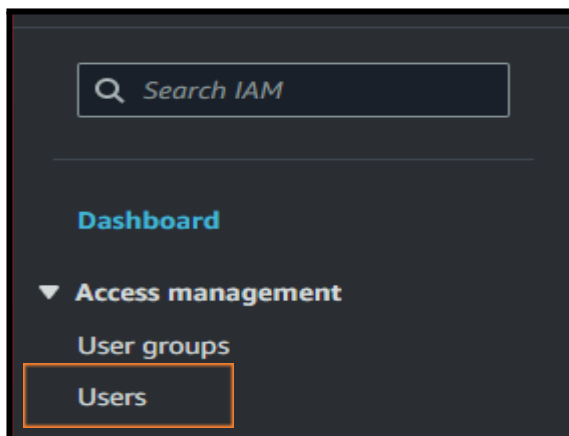
Crearemos el usuario siguiendo los siguientes pasos:

Paso 1.



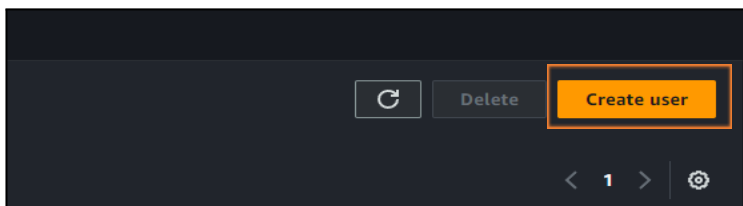
Buscamos el recurso en la barra de búsqueda de la plataforma con las palabras AIM y seleccionamos el recurso.

Paso 2.



Buscamos User y le damos click.

Paso 3.



Seleccionamos **Create user**

Paso 4.

User name

User-Actividad-AWS

The user name can have up to 64 characters. Valid characters: A-Z, a-z, 0-9, and + = , . @ _ - (hyphen)

☒ **Provide user access to the AWS Management Console - optional**
If you're providing console access to a person, it's a [best practice](#) to manage their access in IAM Identity Center.

Are you providing console access to a person?

User type

☐ **Specify a user in Identity Center - Recommended**
We recommend that you use Identity Center to provide console access to a person. With Identity Center applications.

☒ **I want to create an IAM user**
We recommend that you create IAM users only if you need to enable programmatic access through access keys, or a backup credential for emergency account access.

En la casilla de User name le asignaremos un nombre de usuario y seleccionaremos los siguientes parámetros:

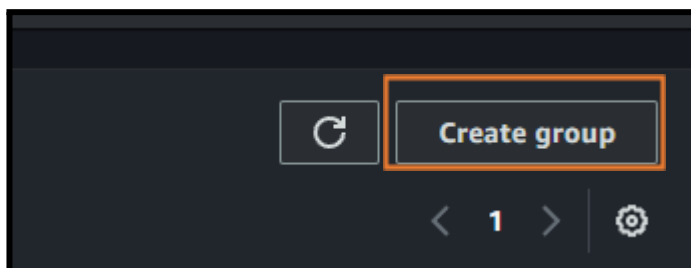
- ☐ **Provide user access to the AWS Management Console**
- ☐ **I want to create an IAM user**
- ☐ **Custom password (y creamos una contraseña personalizada)**

Click en botón Next

Paso 5.

En este paso le daremos los permisos al usuario en caso que ya se tengan un grupo creado se lo asignaremos. En nuestro caso crearemos uno nuevo.

para los cual vamos al botón Create Group



User group name
Enter a meaningful name to identify this group.

Grupo-2


Maximum 128 characters. Use alphanumeric and '+=, @-_' characters.


Le asignamos un nombre, buscamos y le asignaremos los siguientes permisos:


- ☐ **AmazonEC2FullAccess**
- ☐ **AmazonS3FullAccess**
- ☐ **AmazonEBSCSIDriverPolicy**
- ☐ **ROSAAmazonEBSCSIDriverOperatorPolicy**


Search

☐ Policy name

☐ ☐  [AmazonEBSCSIDriverPolicy](#)

☐ ☐  [AmazonEC2FullAccess](#)

☐ ☐  [AmazonS3FullAccess](#)

☐ ☐  [ROSAAmazonEBSCSIDriverOperatorPolicy](#)

Paso 6.

Agregarles las Tags

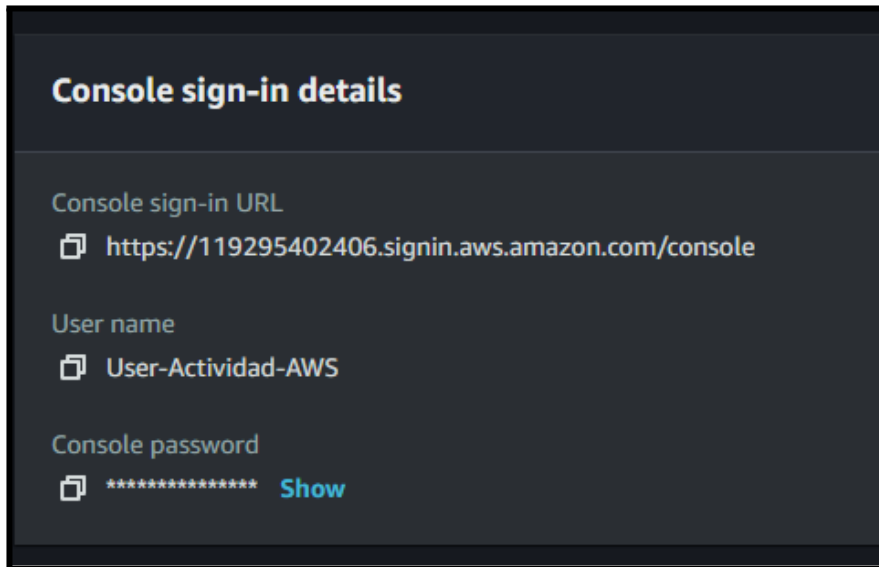
Tags - optional
Tags are key-value pairs you can add to AWS resources to help identify, organize, or search for resources. Choose any tags you want to associate with this user.

Key	Value - optional
<input type="text" value="Owner"/>	<input type="text" value="Josue Reyes"/>
<input type="text" value="E-mail"/>	<input type="text" value="josuereydev@gmail.com"/>
<input type="text" value="Team"/>	<input type="text" value="Grupo-2"/>
<input type="text" value="Proyecto-1"/>	<input type="text" value="Actividad-AWS"/>

Use "Actividad-AWS"

Y por ultimo le damos **Create User**

Y nos abrirá la siguiente ventana con la información del usuario creado

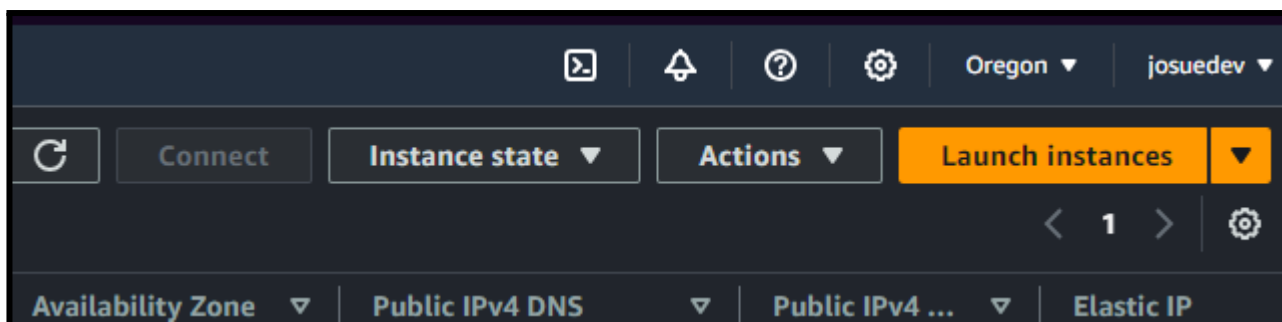


Lanzar instancias en EC2

Para lanzar una instancia en EC2 es importante tener en cuenta la región o zona en la que se lanzará, en nuestro caso lo haremos en **us-east-1 (N. Virginia)**

En la barra de búsqueda escribimos EC2 y vamos a **Launch instances**

Paso 1.



Paso 2. Elegir un nombre para nuestra instancia

Launch an instance Info

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

Name and tags Info

Name

Add additional tags

Paso 3. Buscamos y seleccionamos el recurso o la AMI

ubuntu

Ubuntu

Free tier eligible

Verified provider

Ubuntu Server 20.04 LTS (HVM), SSD Volume Type

ami-08e2c1a8d17c2fe17 (64-bit (x86)) / ami-05f290e7e87696c29 (64-bit (Arm))

Ubuntu Server 20.04 LTS (HVM),EBS General Purpose (SSD) Volume Type. Support available from Canonical (<http://www.ubuntu.com/cloud/services>).

Platform: ubuntu

Root device type: ebs

Virtualization: hvm

ENA enabled: Yes

Select

☒ 64-bit (x86)

☐ 64-bit (Arm)

Paso 4. Elegir el tipo de Instancia (En este caso elegiremos la permitida en la capa gratuita de AWS)

▼ Instance type Info | Get advice

Instance type

t2.micro

Free tier eligible

Family: t2

1 vCPU

1 GiB Memory

Current generation: true

On-Demand Linux base pricing: 0.0116 USD per Hour

On-Demand SUSE base pricing: 0.0116 USD per Hour

On-Demand Windows base pricing: 0.0162 USD per Hour

On-Demand RHEL base pricing: 0.0716 USD per Hour

☐ All generations

Compare instance types

Additional costs apply for AMIs with pre-installed software

Paso 5. Creamos un Key Pair Login (*Este Key Pair lo podemos utilizar en cualquier máquina desplegada en la región de Virginia*)

The screenshot shows the 'Create key pair' dialog box. At the top, it says 'Create key pair' with a close button. Below, the 'Key pair name' section has a text input field containing 'KP-Grupo2-DesafioAWS'. A note below the field states: 'The name can include up to 255 ASCII characters. It can't include leading or trailing spaces.' The 'Key pair type' section has two options: 'RSA' (selected with a radio button) and 'ED25519'. The RSA option is described as 'RSA encrypted private and public key pair'. The ED25519 option is described as 'ED25519 encrypted private and public key pair'. The 'Private key file format' section has one option: '.pem' (selected with a radio button), described as 'For use with OpenSSH'.

Paso 6. Configuramos los parámetros de Red

- Predeterminado Create security Group
- Allow SSH traffic from **Anywhere** (*Dejarlo en Anywhere es una mala práctica porque permitiría que haya conexión de cualquier lugar*) de momento lo dejaremos así.

The screenshot shows the 'Create security group' dialog box. At the top, there are two radio buttons: 'Create security group' (selected) and 'Select existing security group'. Below, it says 'We'll create a new security group called 'launch-wizard-2' with the following rules:'. There are three checked checkboxes: 'Allow SSH traffic from' (with a note 'Helps you connect to your instance' and a dropdown menu showing 'Anywhere' and '0.0.0.0/0'), 'Allow HTTPS traffic from the internet' (with a note 'To set up an endpoint, for example when creating a web server'), and 'Allow HTTP traffic from the internet' (with a note 'To set up an endpoint, for example when creating a web server'). At the bottom, there is a yellow warning box with a triangle icon and the text: '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.' with a close button.

Paso 7. Configurar el almacenamiento.

▼ **Configure storage** [Info](#)

Advanced

1x GiB ▼ Root volume (Not encrypted)

❗

Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage

×

Paso 8. Instalamos Apache2 desde nuestra Instancia de la siguiente manera

En Advanced Details escribiremos el siguiente script.

User data - optional [Info](#)

Upload a file with your user data or enter it in the field.

📁

 Choose file

```
#!/bin/bash
apt-get update -y
apt-get install apache2 -y
echo "Instalacion de Apache2-Server"
systemctl start apache2
```

Paso 9. Lanzar la Instancia

En la sección de **Instances** podemos verificar que la instancia creada está corriendo.

Instances (1/2) Info			
<input type="text" value="Find Instance by attribute or tag (case-sensitive)"/>			All states ▼
<input type="checkbox"/>	Name ✎	Instance ID	Instance state
<input checked="" type="checkbox"/>	ActividadAWS-Linux22	i-074824a61891b678d	🟢 Running
<input type="checkbox"/>	UbutuServer-DesafioAWS	i-0aea28ebd79faf8d3	⏸ Stopped

Conectarse a la Instancia por SSH desde nuestra VM

Hay varias formas para conectarnos a la instancia en esta ocasión lo haremos por medio de conexión SSH desde nuestra terminal de Linux

Paso 1. Copiamos nuestro clave codificada o KeyPair en algún fichero en nuestra VM

una vez en ese directorio corremos el comando **chmod 400** para darle permisos y luego corremos el comando **ssh -i** + el contenido de nuestro archivo .pem

EC2 Instance Connect

Session Manager

SSH client

EC2 serial console

Instance ID
i-074824a61891b678d (ActividadAWS-Linux22)

1. Open an SSH client.

2. Locate your private key file. The key used to launch this instance is KP-Grupo2-DesafioAWS.pem

3. Run this command, if necessary, to ensure your key is not publicly viewable.
chmod 400 "KP-Grupo2-DesafioAWS.pem"

4. Connect to your instance using its Public DNS:
ec2-3-93-187-245.compute-1.amazonaws.com

Example:
ssh -i "KP-Grupo2-DesafioAWS.pem" ubuntu@ec2-3-93-187-245.compute-1.amazonaws.com

Si los datos son correctos nos conectaremos a la instancia EC2 que hemos creado.

```
/home/linux/Documentos/AWS-credenciales
root@ubuntu20:/home/linux/Documentos/AWS-credenciales# chmod 400 "KP-Grupo2-DesafioAWS.pem"
root@ubuntu20:/home/linux/Documentos/AWS-credenciales# ssh -i "KP-Grupo2-DesafioAWS.pem" ubuntu@ec2-3-93-187-245.compute-1.amazonaws.com
Welcome to Ubuntu 22.04.4 LTS (GNU/Linux 6.5.0-1017-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/pro

System information as of Sun May  5 23:40:33 UTC 2024

System load:  0.0               Processes:            102
Usage of /:   25.2% of 7.57GB   Users logged in:     1
Memory usage: 21%              IPv4 address for eth0: 172.31.16.65
Swap usage:   0%

Expanded Security Maintenance for Applications is not enabled.

7 updates can be applied immediately.
7 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable
```

Paso 2. Verificamos si el servidor Apache2 fue instalado con el comando:

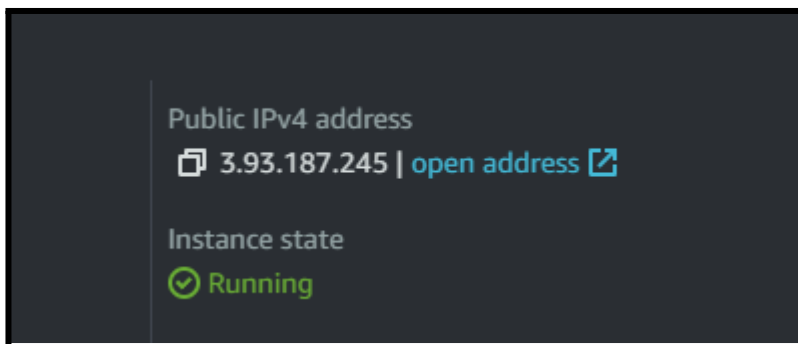
sudo systemctl status apache2.

```
Last login: Sun May  5 23:28:06 2024 from 186.22.245.136
ubuntu@ip-172-31-16-65:~$ sudo systemctl reload apache2
ubuntu@ip-172-31-16-65:~$ sudo systemctl status apache2
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enabled)
   Active: active (running) since Sun 2024-05-05 23:19:00 UTC; 22min ago
     Docs: https://httpd.apache.org/docs/2.4/
   Process: 346 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SUCCESS)
   Process: 925 ExecReload=/usr/sbin/apachectl graceful (code=exited, status=0/SUCCESS)
  Main PID: 411 (apache2)
    Tasks: 55 (limit: 1121)
   Memory: 7.5M
      CPU: 150ms
   CGroup: /system.slice/apache2.service
           └─411 /usr/sbin/apache2 -k start
             └─929 /usr/sbin/apache2 -k start
               └─930 /usr/sbin/apache2 -k start
```

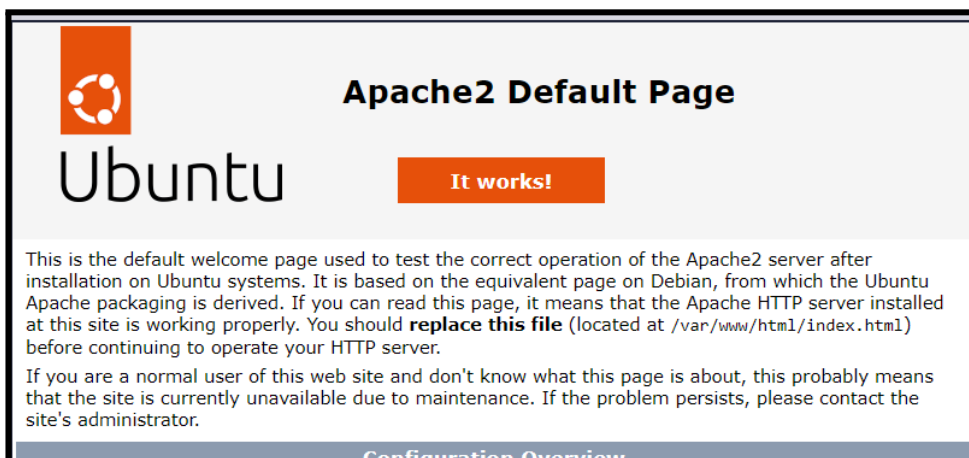
La imagen del servidor indica que está corriendo.

Paso 3. Verificar en un navegador

Para verificar que podemos navegar en el servidor lo hacemos con la dirección IP Pública de nuestra Instancia.



Copiamos la dirección en la barra de navegación y nos abrirá el servidor.



Crear un bucket en el servicio S3

Una vez logueados al usuario que creamos con los permisos de **AmazonS3FullAccess** podremos crear un bucket para almacenamiento.

Paso 1.

Ingresamos un nombre para el bucket y dejamos todos los servicios predeterminados, por último cliqueamos en el botón **Crear Bucket**.

Configuración general

Región de AWS

EE. UU. Este (Norte de Virginia) us-east-1

Tipo de bucket [Información](#)



Uso general

Recomendado para la mayoría de los casos de uso y patrones de acceso. Los buckets de uso general son del tipo de bucket de S3 original. Permiten una combinación de clases de almacenamiento que almacenan objetos de forma redundante en múltiples zonas de disponibilidad.



Directorio: *nuevo*

Recomendado para casos de uso de baja latencia. Estos buckets utilizan únicamente la clase de almacenamiento S3 Express One Zone, que proporciona un procesamiento más rápido de los datos dentro de una única zona de disponibilidad.

Nombre del bucket [Información](#)

bucketgrupo2

Configuración de bloqueo de acceso público para este bucket

Se concede acceso público a los buckets y objetos a través de listas de control de acceso (ACL), políticas de bucket, políticas de puntos de acceso o todas las anteriores. A fin de garantizar que se bloquee el acceso público a todos sus buckets y objetos, active Bloquear todo el acceso público. Esta configuración se aplica exclusivamente a este bucket y a sus puntos de acceso. AWS recomienda activar Bloquear todo el acceso público, pero, antes de aplicar cualquiera de estos ajustes, asegúrese de que las aplicaciones funcionarán correctamente sin acceso público. Si necesita cierto nivel de acceso público a los buckets u objetos, puede personalizar la configuración individual a continuación para adaptarla a sus casos de uso de almacenamiento específicos. [Más información](#)

☒ Bloquear *todo* el acceso público

Activar esta configuración equivale a activar las cuatro opciones que aparecen a continuación. Cada uno de los siguientes ajustes son independientes entre sí.



Bloquear el acceso público a buckets y objetos concedido a través de *nuevas* listas de control de acceso (ACL)

S3 bloqueará los permisos de acceso público aplicados a objetos o buckets agregados recientemente, y evitará la creación de nuevas ACL de acceso público para buckets y objetos existentes. Esta configuración no cambia los permisos existentes que permiten acceso público a los recursos de S3 mediante ACL.



Bloquear el acceso público a buckets y objetos concedido a través de *cualquier* lista de control de acceso (ACL)

S3 ignorará todas las ACL que conceden acceso público a buckets y objetos.



Bloquear el acceso público a buckets y objetos concedido a través de políticas de bucket y puntos de acceso públicas *nuevas*

S3 bloqueará las nuevas políticas de buckets y puntos de acceso que concedan acceso público a buckets y objetos. Esta configuración no afecta a las políticas ya existentes que permiten acceso público a los recursos de S3.

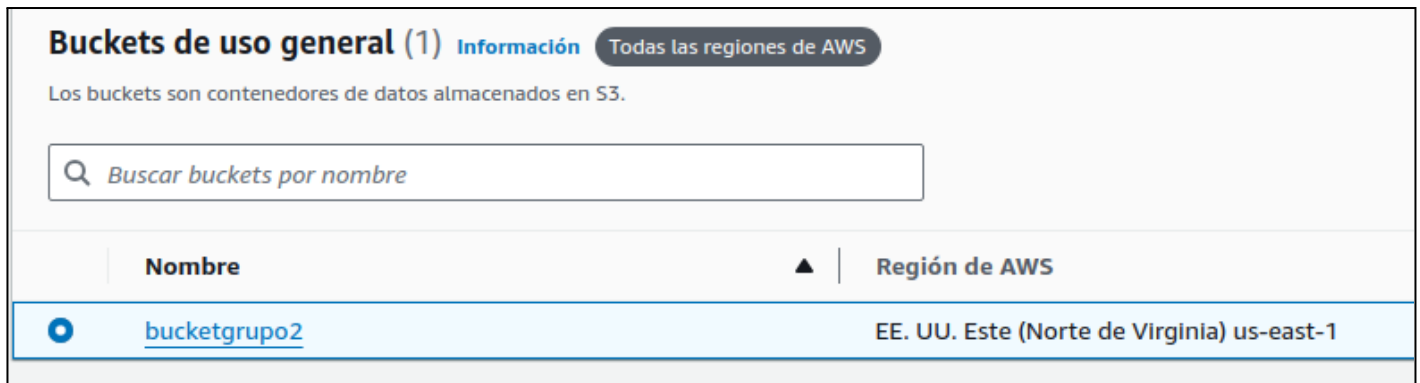


Bloquear el acceso público y entre cuentas a buckets y objetos concedido a través de *cualquier* política de bucket y puntos de acceso pública

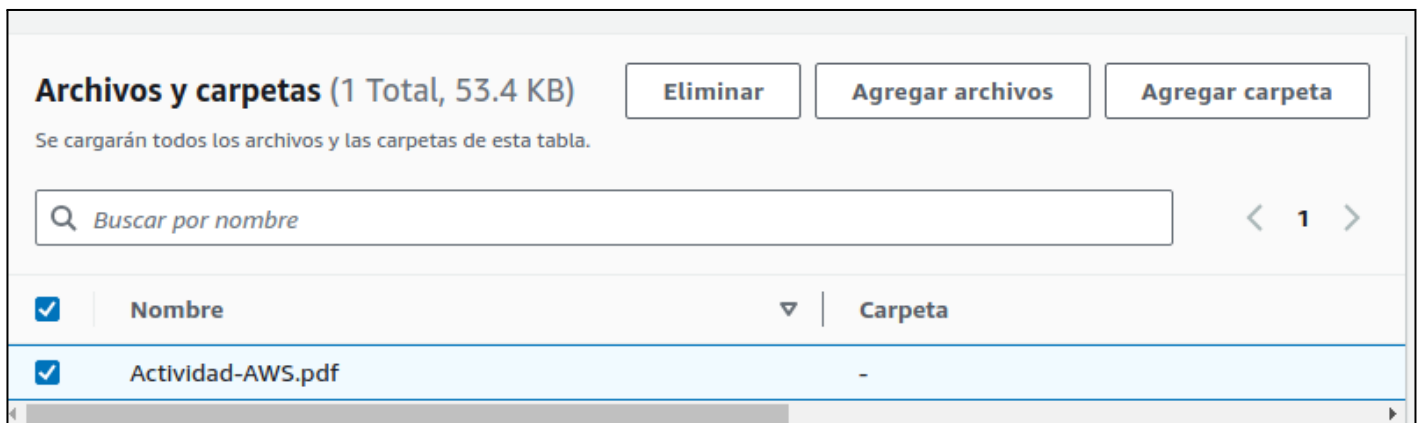
S3 ignorará el acceso público y entre cuentas en el caso de buckets o puntos de acceso que tengan políticas que concedan acceso público a buckets y objetos.

Paso 2. Subir un archivo al bucket

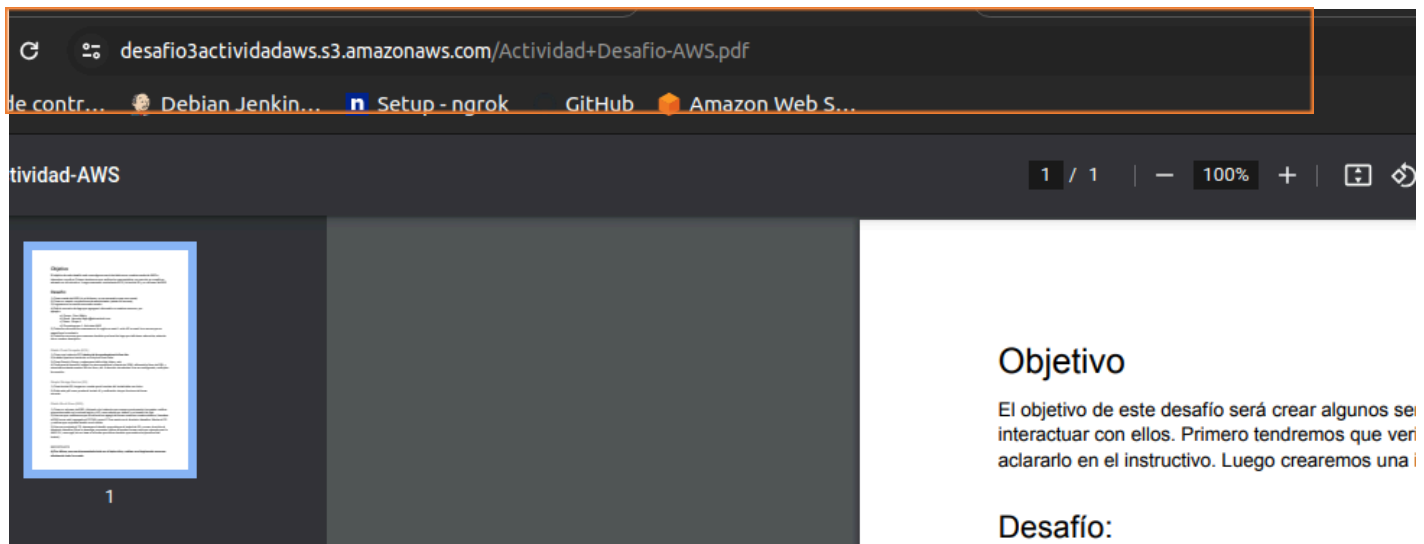
Una vez creado el bucket listamos e ingresamos al bucket haciendo clic sobre el bucket.



Paso 3. Cargar o subir el archivo.



Paso 4. Verificación Podemos ver que el link corresponde a un bucket del S3



Elastic Block Store (EBS)

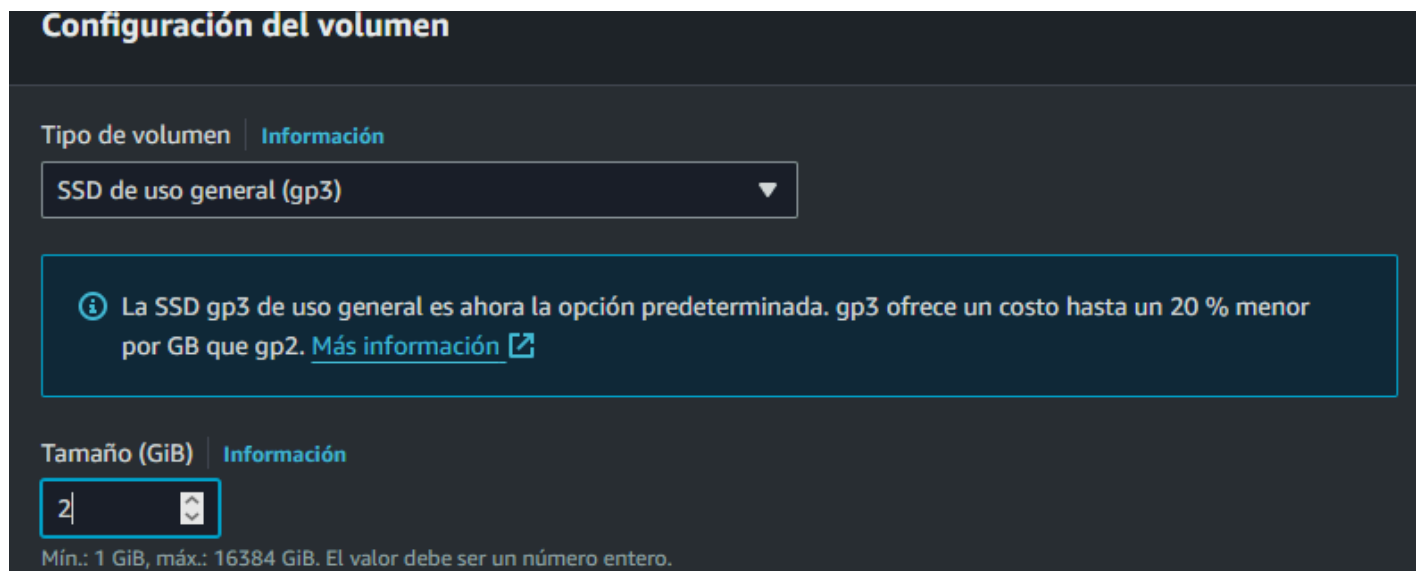
Crear un volumen de EBS y linkearlo a la instancia que creamos previamente.

Paso 1.

En el panel del servicio EC2 nos dirigimos al apartado Elastic Block Store (EBS), Volúmenes damos click al botón **Crear Volumen**.

Le asignaremos un tamaño, para este ejercicio le asignamos un espacio de 3 GB ponemos cuidado que este en la misma zona de disponibilidad que nuestra instancia EC2 es decir **us-east-1a**.

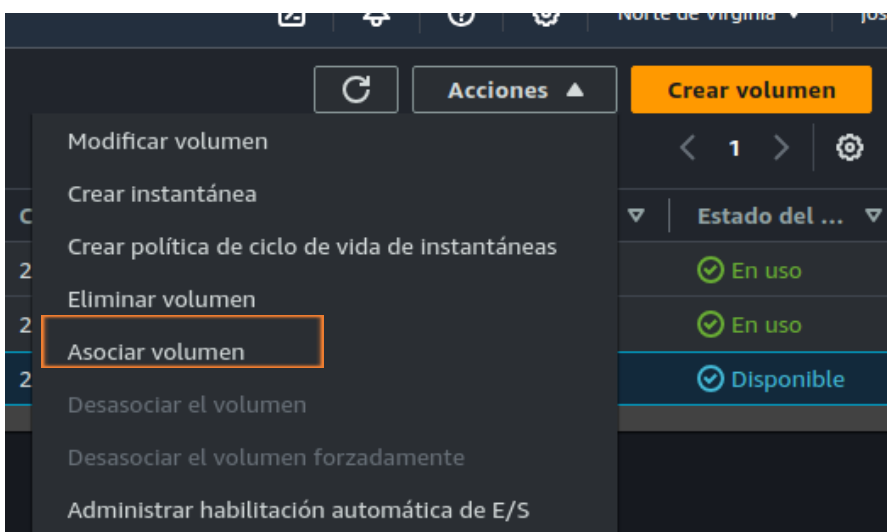
Ciframos el volumen y por último Creamos el Volumen



The screenshot shows the 'Configuración del volumen' (Volume Configuration) page in the AWS Management Console. It features a dark theme. At the top, there's a header 'Configuración del volumen'. Below it, there are two tabs: 'Tipo de volumen' (selected) and 'Información'. Under 'Tipo de volumen', a dropdown menu shows 'SSD de uso general (gp3)'. A blue information box below this states: 'La SSD gp3 de uso general es ahora la opción predeterminada. gp3 ofrece un costo hasta un 20 % menor por GB que gp2. [Más información](#)'. Further down, there's another section for 'Tamaño (GiB)' with a tab 'Información'. A text input field contains the number '2', and a small up/down arrow icon is to its right. Below the input field, a note reads: 'Mín.: 1 GiB, máx.: 16384 GiB. El valor debe ser un número entero.'

Paso 2. Asociar el Volumen

Una vez creado el volumen en el panel principal de volúmenes lo seleccionamos y dentro de **Acciones** hacemos clic en la opción **Asociar Volumen**.



Dentro de los detalles Asociamos nuestra Instancia EC2 y al volumen seleccionamos **/dev/sdf**

Zona de disponibilidad
us-east-1a

Instancia | Información

I-01ece390a9ff8161f

Solo se muestran las instancias de la misma zona de disponibilidad que el volumen seleccionado.

Nombre del dispositivo | Información

/dev/sdf

Nombres de dispositivos recomendados para Linux: /dev/sda1 para el volumen raíz. /dev/sd[f-p] para los volúmenes de datos.

Por último le damos clic en en el botón **Asociar Volumen**.

Paso 3. Listar nuestras unidades

Nos conectamos a la instancia por el método que hayamos elegido en mi caso lo hice por SSH y listamos nuestras unidades con el siguiente comando **lsblk**

```
ubuntu@ip-172-31-40-23:~$ lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINTS
loop0        7:0    0  25.2M  1 loop /snap/amazon-ssm-agent/7983
loop1        7:1    0  55.7M  1 loop /snap/core18/2812
loop2        7:2    0  55.7M  1 loop /snap/core18/2823
loop3        7:3    0  63.9M  1 loop /snap/core20/2264
loop4        7:4    0  63.9M  1 loop /snap/core20/2318
loop5        7:5    0   87M  1 loop /snap/lxd/27948
loop6        7:6    0   87M  1 loop /snap/lxd/28373
loop7        7:7    0  39.1M  1 loop /snap/snapd/21184
loop8        7:8    0  38.7M  1 loop /snap/snapd/21465
xvda        202:0    0    8G   0 disk
├─xvda1     202:1    0   7.9G   0 part /
├─xvda14    202:14   0    4M   0 part
└─xvda15    202:15   0  106M   0 part /boot/efi
xvdf        202:80   0    2G   0 disk
```


Paso 4. Formatear el volumen EBS

Formateamos el volumen con el siguiente comando: **sudo mkfs -t ext4 /dev/xvdf**

(/xvdf corresponde al volumen que le asignamos en el paso No. 2)

```
ubuntu@ip-172-31-40-23:~$ sudo mkfs -t xfs /dev/xvdf
meta-data=/dev/xvdf            isize=512    agcount=4, agsize=196608 blks
        =                       sectsz=512    attr=2, projid32bit=1
        =                       crc=1        finobt=1, sparse=1, rmapbt=0
        =                       reflink=1    bigtime=0 inobtcount=0
data      =                       bsize=4096   blocks=786432, imaxpct=25
        =                       sunit=0      swidth=0 blks
naming    =version 2           bsize=4096   ascii-ci=0, ftype=1
log        =internal log      bsize=4096   blocks=2560, version=2
        =                       sectsz=512   sunit=0 blks, lazy-count=1
realtime  =none                extsz=4096   blocks=0, rtextents=0
ubuntu@ip-172-31-40-23:~$
```

Verificamos nuevamente con el comando **lsblk -f**

Paso 5. Crear el directorio /desafíos

sudo mkdir /desafios

Paso 6. Agregar el volumen al archivo FSTAB

nano /etc/fstab (también podemos usar vim)

```
GNU nano 6.2
LABEL=cloudimg-rootfs /          ext4    discard,errors=remount-ro      0 1
LABEL=UEFI            /boot/efi  vfat    umask=0077                    0 1
/dev/xvdf /mnt/ebs  ext4    defaults,nofail               0 2
```

/dev/xvdf /desafíos ext4 defaults,nofail 0 2

Paso 7. Montar el disco

sudo mount -a

Paso 8. Verificó que el volumen se haya montado

lsblk -f

```
ubuntu@ip-172-31-40-23:~$ lsblk -f
NAME        FSTYPE    FSVER LABEL          UUID                                  FSAVAIL FSUSE% MOUNTPOINTS
loop0
loop1
loop2
loop3
loop4
loop5
loop6
loop7
loop8       squashfs  4.0
xvda
├─xvda1     ext4      1.0   cloudimg-rootfs db5e27f6-7377-40b0-9756-df259213cbb0  4.7G    38%    /
├─xvda14
├─xvda15    vfat      FAT32 UEFI           692D-8804                      98.3M    6%     /boot/efi
└─xvdf      ext4      1.0   46ca0d54-c276-4463-b25b-01cb6a58ada5  1.8G    0%     /mnt/ebs
                                     /desafios
```

Paso 9. Descargar el fichero ActividadDesafioAWS.pdf de nuestro Bucket con el comando wget

```
ubuntu@ip-172-31-40-23:~$ wget https://desafio3actividadaws.s3.amazonaws.com/Actividad+Desafio-AWS.pdf
--2024-05-19 22:38:24-- https://desafio3actividadaws.s3.amazonaws.com/Actividad+Desafio-AWS.pdf
Resolving desafio3actividadaws.s3.amazonaws.com (desafio3actividadaws.s3.amazonaws.com)... 16.182.34.17
Connecting to desafio3actividadaws.s3.amazonaws.com (desafio3actividadaws.s3.amazonaws.com)|16.182.34.17|:443:
HTTP request sent, awaiting response... 200 OK
Length: 54633 (53K) [application/pdf]
Saving to: 'Actividad+Desafio-AWS.pdf.1'

Actividad+Desafio-AWS.pdf.1          100%[=====]
2024-05-19 22:38:24 (36.6 MB/s) - 'Actividad+Desafio-AWS.pdf.1' saved [54633/54633]
```

```
ubuntu@ip-172-31-40-23:~$ ls
Actividad+Desafio-AWS.pdf  aws  awscliv2.zip
ubuntu@ip-172-31-40-23:~$
```

Paso 10. Movemos el fichero al volumen EBS

```
ubuntu@ip-172-31-40-23:~$ sudo mv Actividad+Desafio-AWS.pdf /desafios
ubuntu@ip-172-31-40-23:~$ ls
aws  awscliv2.zip
ubuntu@ip-172-31-40-23:~$ cd /desafios
ubuntu@ip-172-31-40-23:/desafios$ ls
Actividad+Desafio-AWS.pdf  lost+found
ubuntu@ip-172-31-40-23:/desafios$
```