**Indicaciones**

**Amazon S3 - AWS Elastic Block Store (EBS)**

1. Las respuestas deben ser explicadas, solo colocar resultados sin ninguna referencia no puntúa en las preguntas de la evaluación.
2. Realiza una copia de este documento y coloca todas tus respuestas y sube a tu repositorio personal de github en formato markdown. Presenta capturas de pantalla del procedimiento y las explicaciones necesarias. No puntúa si solo se hace la presentación de imágenes.
3. De preferencia adiciona un video adicional explicando los pasos realizados. Utiliza el sandbox de AWS usado en la práctica anterior.
4. Sube a la plataforma de Blackboard el enlace de github donde están todas tus respuestas. No olvides colocar tu nombre y apellido antes de subir el enlace de tus respuestas a la plataforma
5. Cualquier evidencia de copia elimina el examen se informará de la situación a la coordinación.

S3

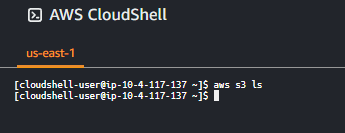
En este laboratorio, se estudiará el almacenamiento de Amazon S3. Utilizarás los comandos aws s3 y s3api para administrar datos en Amazon S3. Amazon S3 es un almacenamiento de objetos accesible a través de Internet.

# Parte 1: Operaciones básicas con S3

Suponga que su directorio actual es /home/aws\_user (puedes cambiarlo). Envíe las siguientes instrucciones y responde las preguntas que siguen.

1. Enumere todos los buckets propiedad del usuario a través del siguiente comando ls. aws s3 ls

¿Cuál es la salida?



El comando “aws s3 ls” lista los archivos y directorios en un bucket de Amazon S3. En este caso, como no tenemos un bucket. la salida será vacío.

Para ello, vamos a hacer un bucket

1. Haz un bucket a través del siguiente comando

mb. aws s3 mb s3://tu\_nombre\_de\_usuario

¿Cuál es la salida?



**El comando “aws s3 mb s3://rodrigorodriguez” está indicando a la AWS CLI que deseas crear un nuevo bucket en Amazon S3 con un nombre. El mb significa make bucket y el prefijo s3:// indica que se trata de un bucket de S3**

**Al darle enter al comando, la salida será un mensaje: “make\_bucket: rodrigorodriguez” . Eso quiere decir que el bucket se creó correctamente en Amazon S3 con el nombre rodrigorodriguez.**

1. Enumera el contenido del bucket a través del siguiente comando ls

aws s3 ls s3://tu\_nombre\_de\_usuario

¿Cuál es la salida?



**La salida está vacía porque recién he creado un bucket y no tiene objetos/archivos en el bucket.**

1. Crea un directorio llamado páginas web (mkdir webpages) y cd en ese directorio.



**Se crea un directorio con el uso del comando mkdir y para cambiar el**

**directorio actual al directorio creado usaremos el comando cd**

Crea un archivo html simple llamado hello.html con el siguiente contenido.

<html><body>

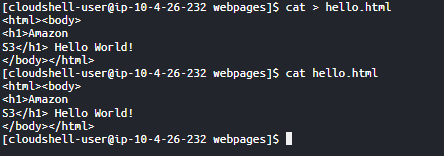
<h1>Amazon S3</h1> Hello World!

</body></html>

**Para crear el archivo hello.html y colocar mensaje al hello.html al mismo tiempo**

**Vamos a utilizar el comando echo para crear y colocarle mensaje. Después usaremos**

**el comando cat para verificar si los mensajes están en el archivo hello.html**



Carga el archivo en tu bucket s3 y póngalo a disposición del público con lo siguiente. aws s3 cp hello.html s3://tu\_nombre\_de\_usuario --acl public-read

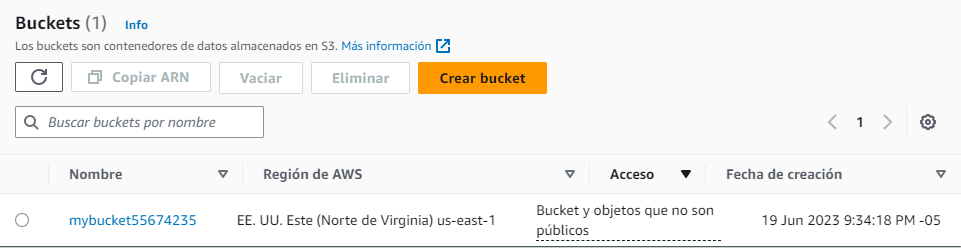
¿Cuál es la salida?

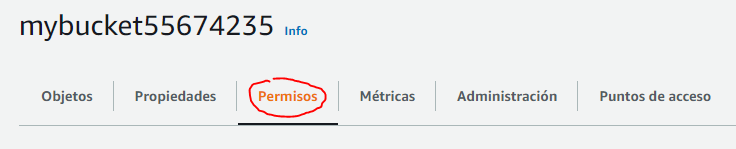


**Nos sale error, pero para solucionarlo iremos a Amazon S3 y buscamos nuestro bucket**

**Se desactiva el bloqueo público para que funcione el código.**

**En mi caso, yo entré a aws S3 para desactivar el bloqueo público. OJO. esto es un ejemplo**

**de lo que hice  
  
Primero busque el bucket creado:**  


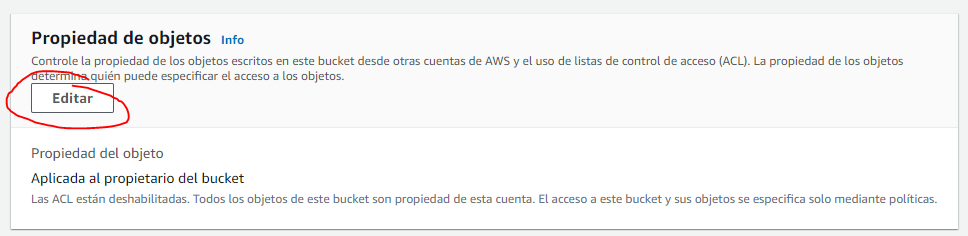
**Le damos click al nombre del bucket, y vamos a permisos.  
**

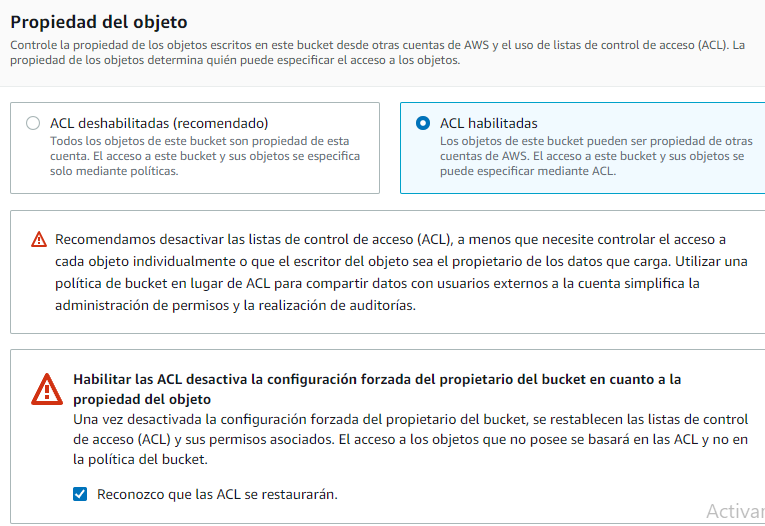
**Vamos a editar 3 permisos : Bloquear acceso público,propiedad de objeto y ACL**

**1- Bloquear acceso publico: Click en el check y le damos guardar**

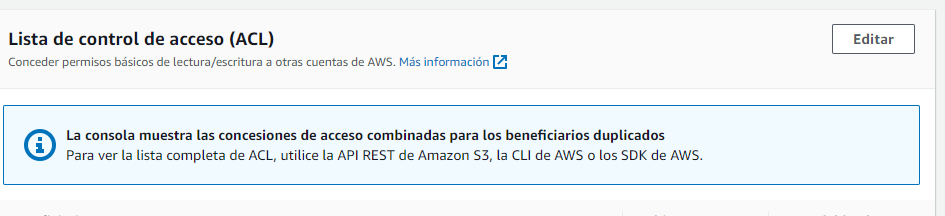
****

**2- Propiedad de objetos: le damos a editar**

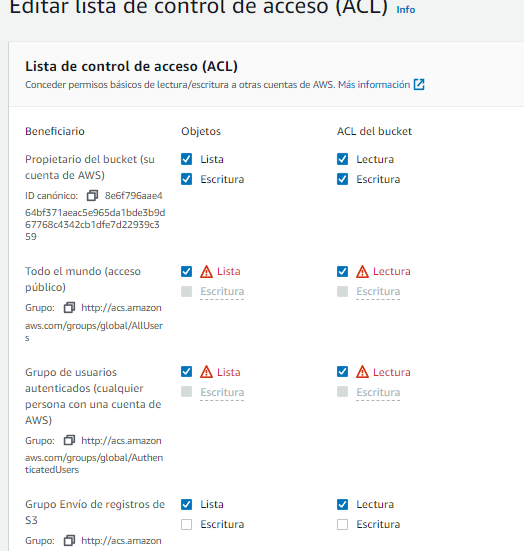


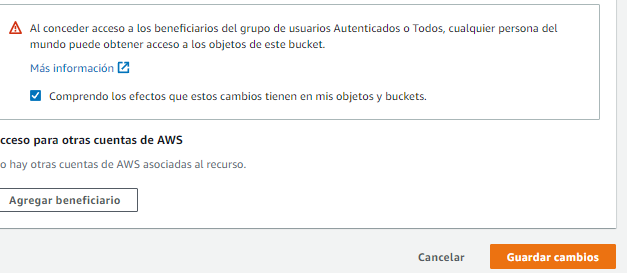
**Habilitamos el ACL, le click a reconocer que las acl se restauraran y por último guardar.**  
  


**3- ACL : click en editar**

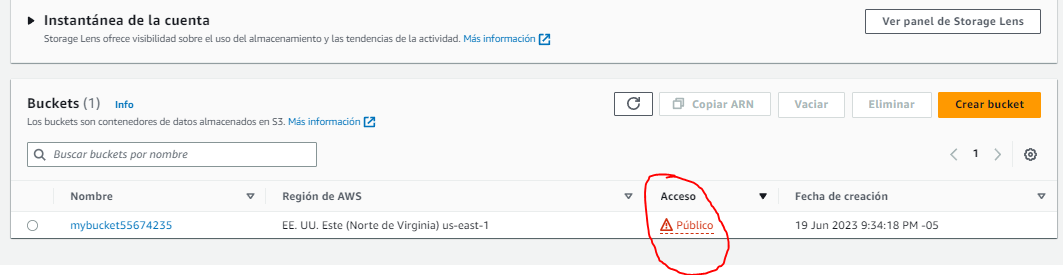
****

**Le damos check a todos, click en el cuadrito de comprendo los efectos y por ultimo guardar.**

****



Después de haber desactivado los permisos privados, ahora iremos a bucket y veremos que la instancia estara publica (No olvidar darle a la ruedita de cargar para que se actualice el bucket)

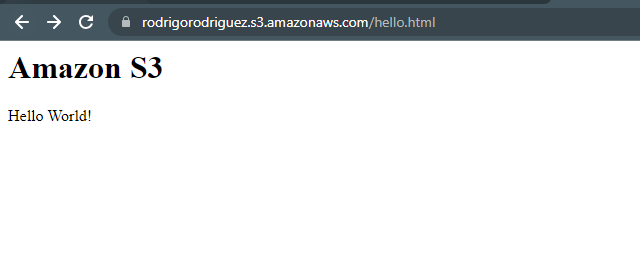


**Regresamos a cloudshell y ejecutamos ahora el código**



**La salida indica que se ha realizado una operación de carga exitosa del archivo "hello.html" al bucket de S3 llamado "rodrigorodriguez"**.

1. Dado que se puede acceder a tu objeto s3 a través de Internet, probémoslo. En el navegador web de tu máquina virtual (u otra9 accede a la URL [http://s3.amazonaws.com/tu\_nombre\_de\_usuario/hello.html.](http://s3.amazonaws.com/tu_nombre_de_usuario/hello.html) ¿Qué viste en el navegador?



**Nos aparece el mensaje del Hello.html Hello world!**

# Parte 2: alojamiento de sitios web estáticos con S3

1. Podemos usar el bucket como almacenamiento de sitios web estáticos. Experimentamos con eso aquí. Crea dos archivos html en el directorio actual llamados index.html y error.html. El contenido de los dos archivos se muestra a continuación.

<html><body>

This is an index page!

</body></html>

<html><body>

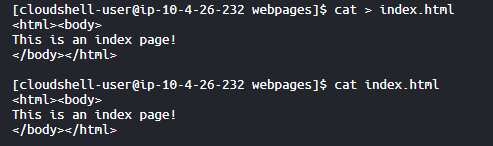
Sorry, we can't find that page!

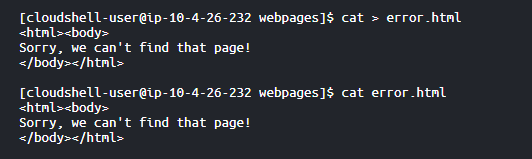
</body></html>

**Usamos el comando cat para crear archivo y colocarle mensaje. Y asi, creamos**

**los archivos con sus respectivos mensajes. y verificamos con el mismo comando**

**solo que le quitamos el >.** >



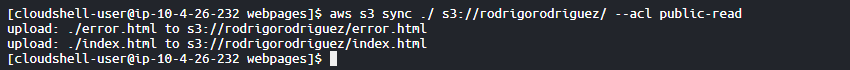


**MÁS ABAJO SE ENCUENTRA LAS PREGUNTAS. NO SE PORQUE  
EL WORD SE PUSO ASI**

El comando sync compara el directorio de origen con tu bucket S3 y carga solo archivos nuevos o modificados. Entonces puedes cargar ambos archivos fácilmente a través del siguiente comando.

aws s3 sync ./ s3://tu\_nombre\_de\_usuario/ --acl public-read

¿Cuál es la salida?



**La salida nos dice que se ha cargado correctamente los archivos error.html y index.html al bucket S3 rodrigorodriguez.**

Ahora habilitamos el bucket para alojamiento de sitios web estáticos con las siguientes instrucciones.

aws s3 website s3://tu\_nombre\_de\_usuario/

--index-document index.html

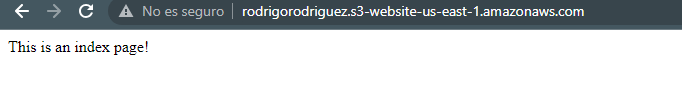
--error-document error.html



Observa cómo la instrucción enlaza ambos archivos con sus usos. En el navegador web de tu VM, acceda a la URL

http://tu\_nombre\_de\_usuario.s3-website-us-east-1.amazonaws.com/

¿Qué viste en el navegador? ¿Por qué?

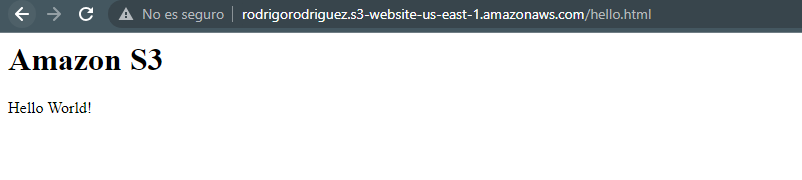


**Lo que vi en el navegador fue el mensaje del archivo index.html This is an index page!**

**Porque solo vemos eso?**

**Porque como no se ha presentado un error. Entonces solo se ejecuta el archivo index.html.**

Ahora, acceda a http://tu\_nombre\_de\_usuario.s3-website-us-east-1.amazonaws.com/ hello.html



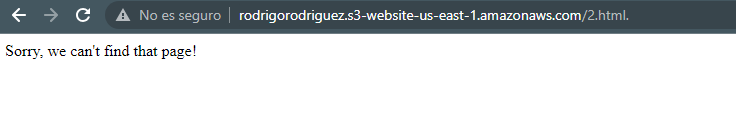
¿Qué viste en el navegador?

**Vi el mensaje Hello world del archivo hello.html**

A continuación, acceda a

http://tu\_nombre\_de\_usuario.s3-website-us-east-1.amazonaws.com/2.html.

¿Qué viste en el navegador? ¿Por qué?



**Nos sale el mensaje del archivo error.html**

**¿Por qué se muestra esto?**

**El comportamiento esperado en esta configuración es que cuando accedes a cualquier URL que no corresponda a un archivo existente en tu sitio web, se mostrará el contenido del archivo error.html. Además, ese archivo 2.html no existe en el bucket y en el bucket solo hemos creado tres archivos (Hello.html,index.html y error.html)**

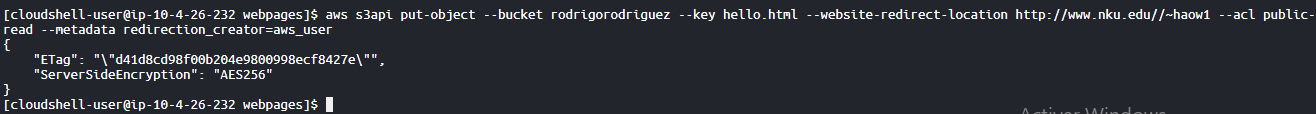
1. Podemos definir reglas de redirección y agregar metadatos a los objetos en el bucket. Ejecuta el siguiente comando para hacerlo. Observa que este comando usa s3api, no s3.

aws s3api put-object --bucket *tu\_nombre\_de\_usuario*

--key hello.html

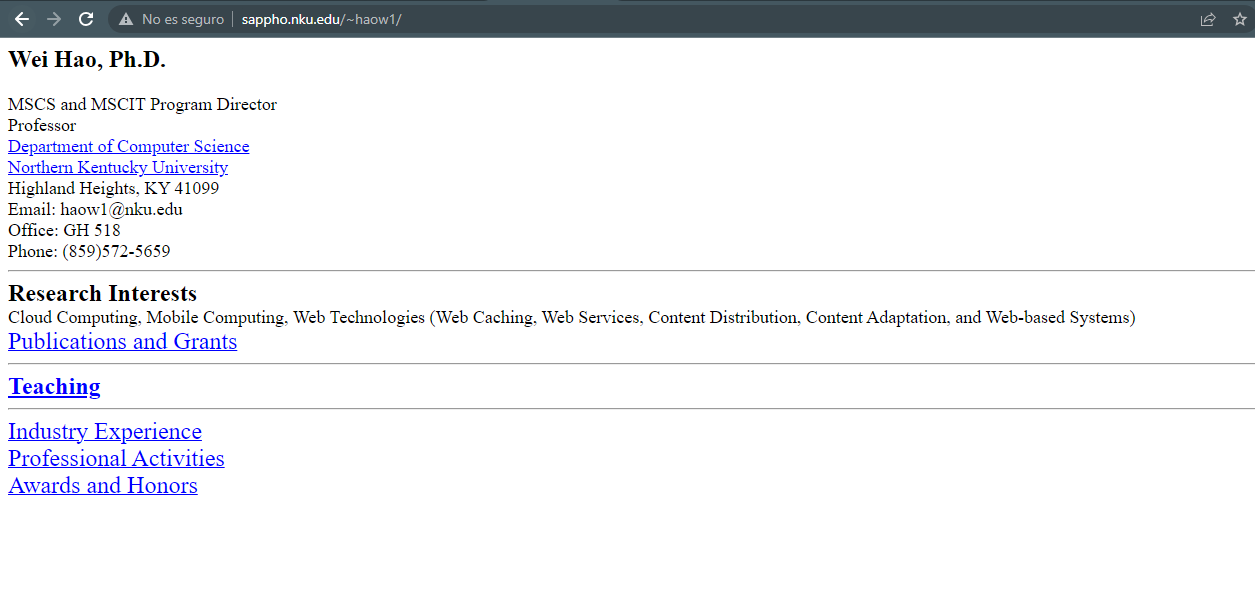
--website-redirect-location <http://www.nku.edu/~haow1>--acl public-read

--metadata redirection\_creator=aws\_user



Ahora http://*tu\_nombre\_de\_usuario*.s3-website-us-east-1.amazonaws.com/hello.html

**¿Qué ves en el navegador? ¿Por qué?.**

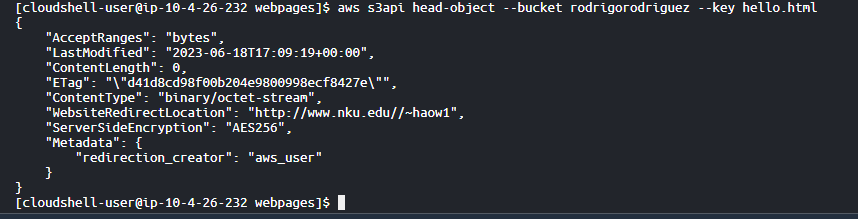


**Veo que ya no sale el mensaje Hello word y aparece como que una información de una persona llamada wei hao. Este código indica la especificación del nombre y la clave para actualizarlo y redirigir a un link (del wei hao por ejemplo) al usuario cuando entra al hello.html**

1. Para recuperar los metadatos de un objeto, usamos el subcomando head-object. Emite la siguiente instrucción.

aws s3api head-object --bucket tu\_nombre\_de\_usuario --key hello.html

¿Cuál es la salida?



**La salida del comando aws s3api head-object –bucket rodrigorodriguez –key hello.html proporciona información sobre el objeto específico en el bucket de Amazon S3. Como por ejemplo: La fecha y hora de la última modificación, el tipo de contenido, la metadata,etc. El código es larguísimo así que lo he cortado .**

# Parte 3: Limpieza

1. Podemos eliminar objetos usando rm. Elimina tu página de índice de la siguiente manera.

aws s3 rm s3://tu\_nombre\_de\_usuario/index.html

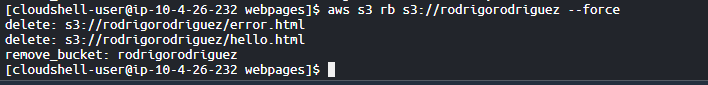


¿Cuál es la salida?

**La salida del código es un mensaje diciendo que se eliminó el archivo index.html del bucket rodrigo rodriguez correctamente**

1. Y podemos quitar el bucket como un todo. Usa lo siguiente. aws s3 rb s3://tu\_nombre\_de\_usuario --force

¿Cuál es la salida? ¿Qué hace --force?



**La salida de este código es un mensaje sobre la eliminación de los archivos y el bucket.**

**El –force es un comando que se utiliza para forzar la eliminación del bucket con todo sus archivos**

EBS

En este laboratorio, se utilizará la CLI de AWS para crear un volumen y una instantánea de Amazon EBS y configurar tu almacenamiento de EBS como un arreglo RAID.

# Parte 1. Crea un nuevo volumen de EBS

1. Inicia sesión en el sandbox del curso. Crea un nuevo volumen de EBS con el siguiente comando.

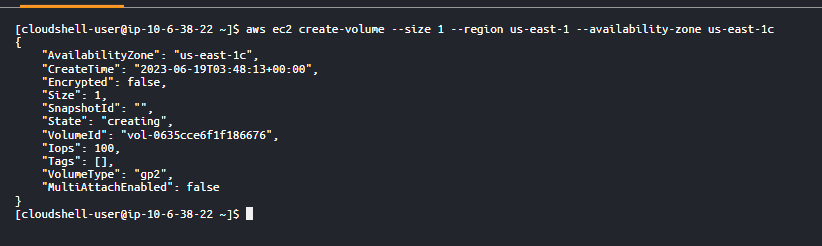
aws ec2 create-volume --size 1 --region us-east-1

--availability-zone us-east-1c

¿Qué significa este comando?

**Este código significa que estamos creando un volumen con características como el tamaño del volumen en 1 gigabytes (--size 1) , la región que se encuentra (--region us-east-1) y su zona de disponibilidad (--availability-zone us-east-1c)**

¿Cuál es la salida?

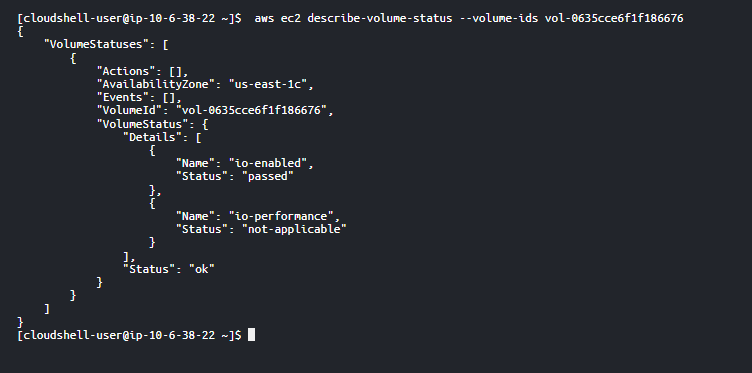


La salida de este código, nos devuelve información del volumen creado como el tiempo de creación, el estado del volumen, la id del volumen, etc.

1. Utiliza el siguiente comando para ver la información de tu volumen de EBS donde se te proporcionó volume\_id en el resultado del comando anterior.

aws ec2 describe-volume-status --volume-ids volume\_id

¿Cuál es la salida?



**Se utiliza para obtener información sobre el estado del volumen que indicaste de Amazon EBS en Amazon EC2.**

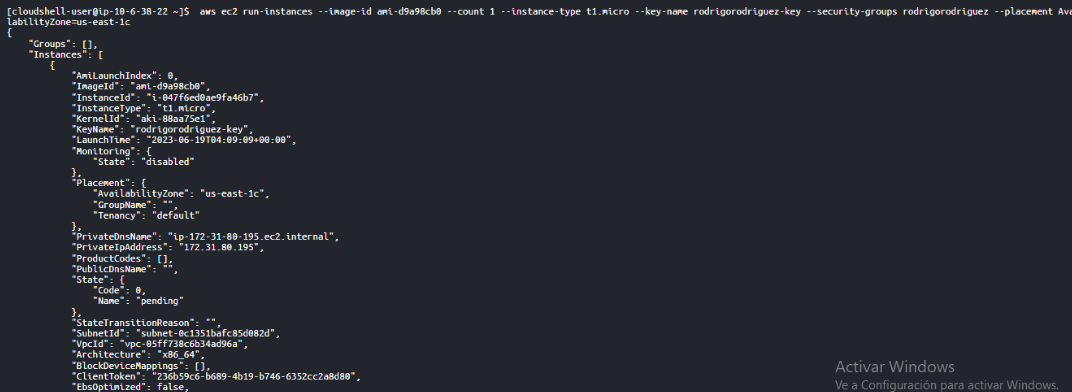
1. Para crear una instancia de EBS, hazlo siguiente.

aws ec2 run-instances --image-id ami-d9a98cb0 --count 1

–instance-type t1.micro –key-name

tu\_nombre\_de\_usuario-key --security-groups tu\_nombre\_de\_usuario

--placement AvailabilityZone=us-east-1c



**La salida muestra los detalles de la instancia que se ha lanzado.**

Ahora, adjunta el volumen de EBS a la instancia. Esto lo colocas en el directorio

/dev/sdf en tu instancia EC2.

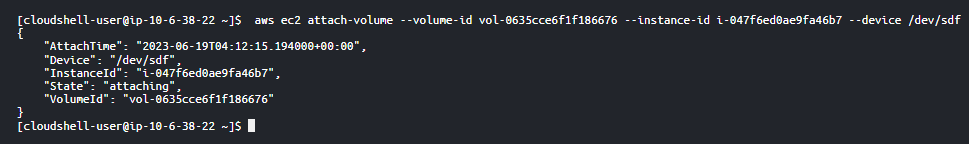
aws ec2 attach-volume --volume-id volume\_id --instance-id id\_instance --device /dev/sdf

**id de la instancia: id i-01d83fc2e514ed147**

**id del volumen :vol-0566fced4d002f392**

¿Cuál es la salida?

aws ec2 attach-volume --volume-id vol-0566fced4d002f392 --instance-id i-01d83fc2e514ed147 --device /dev/sdf



**La salida muestra los detalles de la operación de adjuntar el volumen**

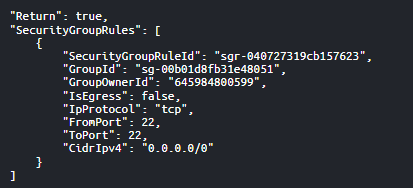
**Por ejemplo: el tiempo de adjunte , su estado, el ID, su dispositivo y el id**

**del ID**

1. Inicia sesión en la instancia EC2 a través de ssh. En tu instancia EC2, cambie a root.

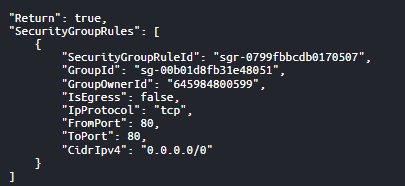
**Para iniciar sesión en la instancia EC2 a través de ssh. Vamos a activar los puertos** **22 y 80 para permitir la conexión ssh. Para ello utilizamos los siguientes codigos.**

**aws ec2 authorize-security-group-ingress --group-name rodrigorodriguez --protocol tcp --port 22 --cidr 0.0.0.0/0**



**Para el puerto 80 utilizaremos el mismo codigo que utilizamos para el puerto 22**

**aws ec2 authorize-security-group-ingress --group-name rodrigorodriguez --protocol tcp --port 80 --cidr 0.0.0.0/0**



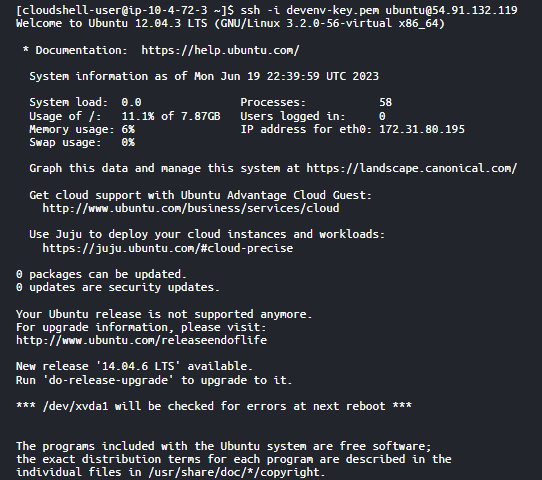
**Después usamos este codigo:**

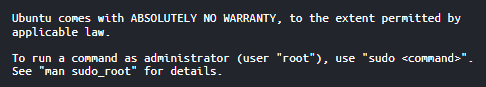


**Al establecer los permisos a "400" en el archivo devenv-key.pem, estás dando permisos de lectura únicamente al propietario del archivo,**

**Teniendo ya los dos puertos. Ahora podemos iniciar sesión de la instancia en ssh.**

**Y lo hacemos con el siguiente código:**





**La salida será una conexión SSH establecida con la instancia EC2. Verás mensajes de autenticación**

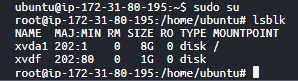
**Cambiamos al modo root con el código sudo su**



Ahora queremos crear un sistema de archivos en el volumen de EBS (el volumen de EBS es básicamente un dispositivo de almacenamiento en blanco). Luego necesitamos montar el volumen para que sea accesible. Utiliza los siguientes comandos desde tu EC2. Ten en cuenta que, según el controlador del dispositivo de bloque del kernel, el dispositivo puede estar conectado con un nombre diferente al que ha especificado. Por ejemplo, si especificas un nombre de dispositivo de

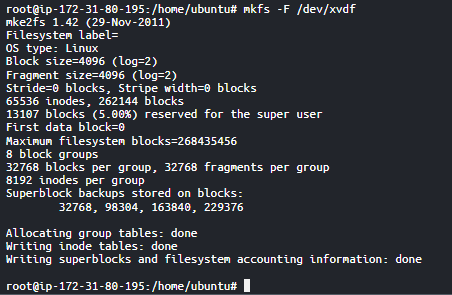
/dev/sdf, el kernel podría cambiar el nombre de tu dispositivo a /dev/xvdf, en la mayoría de los casos, la letra final sigue siendo la misma.

Ejecuta lsblk en tu terminal para ver tus dispositivos de disco disponibles y tus puntos de montaje (si corresponde) para ayudarte a determinar el nombre de dispositivo correcto que debe usar. Suponga que el kernel cambia el nombre del dispositivo a /dev/xvdf.



mkfs -F /dev/xvdf

¿Cuál es la salida?



mkdir /data

mount /dev/xvdf/data cd /data/

df

¿Cuál es la salida?

mkdir/data:



La salida del codigo indica que el comando se ejecutó correctamente y se creó el directorio "data" en el directorio raíz (/).

(Desde aqui quiero indicar que me confundido con el codigo, ya que estaban mal escrito y coloque el &data en xdvf asi que lo hare desde xdvg)

mount /dev/xvdg/data



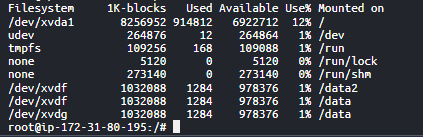
la salida indica la montura del dispositivo en la dirección /data

cd /data/



se pasa de directorio / a directorio /data

df



la salida de este código muestra las filas del sistema, su disponibilidad, el % de uso,etc

# Parte 2. Instantáneas de EBS

1. Crea un archivo llamado aws\_user.txt y escribe lo que desees en el archivo. Ahora, veremos cómo crear una copia de seguridad de todo tu volumen de EBS. El primer paso es asegurarte de que todos los datos en memoria se hayan escrito en el volumen (disco), ya que es posible que el archivo creado aún no se haya guardado en el disco. Para forzar que esto suceda, usamos el comando sync (sincronización). En la ventana de tu terminal para su instancia EC2, ejecuta las siguientes instrucciones.

root@ip-10-45-185-154:/data# sync

Creamos el archivo aws\_user.txt en el modo root usando cat



Y como nos indica el documento usaremos el codigo sync



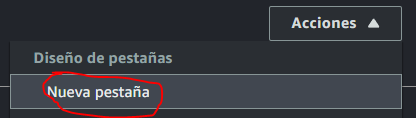
Como no sale nada, significa que se ha ejecutado correctamente.   
Lo que hace sync es que garantiza que los datos en memoria se

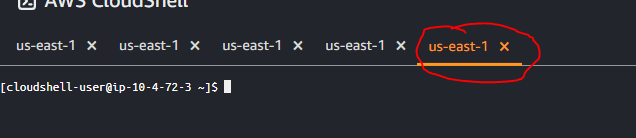
sincronicen con el disco, lo que reduce la pérdida de datos en ca-

sos que haya error en el sistema

Abre una segunda ventana de terminal en tu máquina virtual. Emite el siguiente comando.





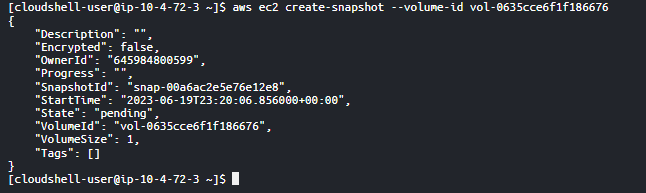


aws ec2 create-snapshot --volume-id volume\_id

--description "Esta es mi instantánea de volumen".

donde volume\_id es el id obtenido del paso 1. ¿Cuál es el resultado?

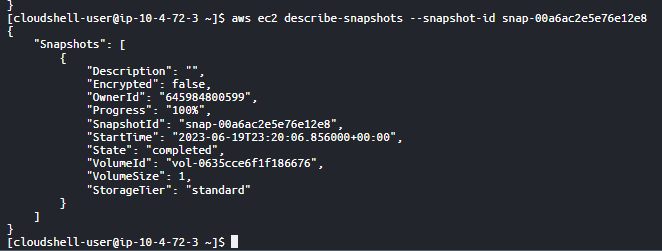
Volumen id : vol-0635cce6f1f186676



Nos sale la descripción del snapshot creado como su ID, el volumen donde esta el snapshot, etc

Puedes verificar el estado de tu instantánea usando las siguientes instrucciones.

aws ec2 describe-snapshots --snapshot-id snapshot\_id



El snapshot\_id debe ser parte de la salida de la instrucción de creación de instantáneas que acaba de ejecutar.

¿Cuál es el resultado del comando describe-snapshot?

**La salida es la información detallada sobre el snapshots (instantáneas) de Amazon Elastic Block Store (EBS). La salida de este comando proporciona información como el ID del snapshot, el volumen asociado, el tamaño del snapshot, la fecha y hora de creación, entre otros detalles.**

Continúa repitiendo este comando hasta que vea que el estado de la instantánea cambia a "completado", lo que significa que se ha realizado una copia de seguridad del volumen.

Como podemos ver la imagen de arriba, vemos que el estado del snapshot

esta completo

1. Dada una instantánea, podemos usarla para crear un nuevo volumen. Ejecuta el siguiente comando. Utiliza el ID de instantánea del paso 5.

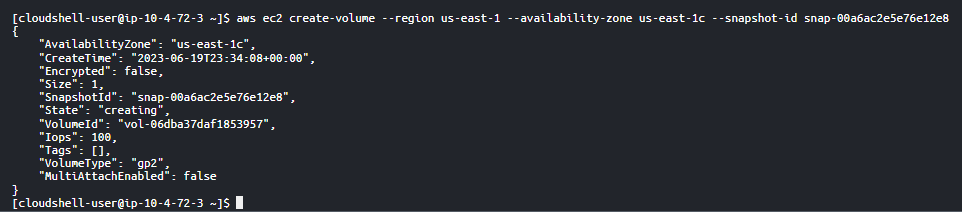
aws ec2 create-volume --region us-east-1

--availability-zone us-east-1c

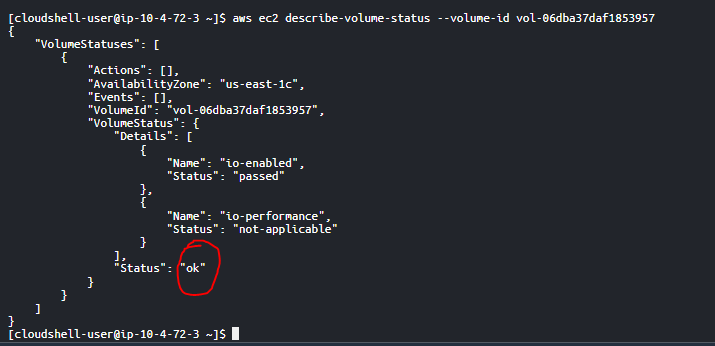
--snapshot-id snapshot\_id

snapshot\_id = snap-00a6ac2e5e76e12e8

¿Cuál es la salida?



**La salida es la información sobre la creacion del volumen con el uso de la id de la instantanea, la región y la zona de disponibilidad**



¿Qué comando ejecutar para verificar el estado?

**Utilice el comando describe-volume-status –volume-id volume**

¿Cuál es la salida?

**La salida nos indica la información detallada del volumen creado como**

**para saber su estado , su id , etc**

1. Repite el comando de adjuntar volumen del paso 3 para adjuntar este nuevo volumen. El ID de volumen será el que se devolvió al obtener el estado 6, mientras que el ID de instancia es el de tu instancia EC2 que obtuvo en el paso 3.

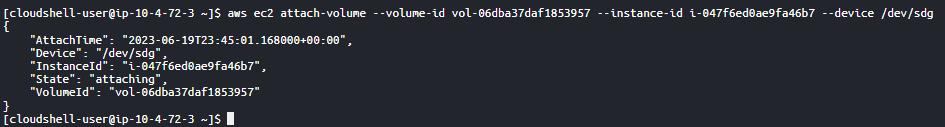
aws ec2 attach-volumen --volume-id volume\_id

--instance-id instance\_id --device /dev/sdg

**Volumen id = vol-06dba37daf1853957**

**instancia id =** [**i-047f6ed0ae9fa46b7**](https://us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#InstanceDetails:instanceId=i-047f6ed0ae9fa46b7)

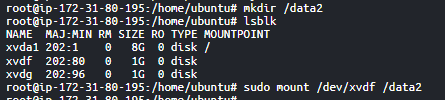
¿Cuál es la salida?

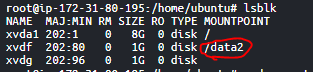


**La salida del codigo nos muestra los detalles cuando el volumen se adjunta con la instancia y sale su estado para verificar si se completo el adjunte o no (attaching)**

1. Vuelve a la ventana de la terminal en la que se tiene ssh en tu instancia EC2. Desde ese terminal, crea un punto de montaje llamado /data2 y monte el nuevo volumen allí.







¿Qué comandos se ejecutaron para lograr ambas tareas?

mkdir

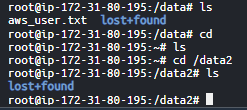
lsblk

sudo mount

Cambia el directorio de su instancia EC2 a /data2.



¿Viste el archivo aws\_user.txt?



No, ya que el aws:user.txt lo creamos en el directorio /data

1. Ahora queremos desmontar nuestros volúmenes, para lo cual usamos el comando unmount. Luego separaremos los volúmenes de la instancia EC2 y los destruiremos. Los siguientes son los comandos a ejecutar. Ten en cuenta que los primeros tres comandos están en su instancia EC2 y el resto está en tu VM.

root@ip-10-45-185-154:/data3# cd /

root@ip-10-45-185-154:/# unmount /dev/xvdf root@ip-10-45-185-154:/# unmount /dev/xvdg

**En mi caso, como el /data esta en /dev/xdvg. Entonces primero desmontamos el**

**/dev/xvdg y luego el /dev/xvdf**





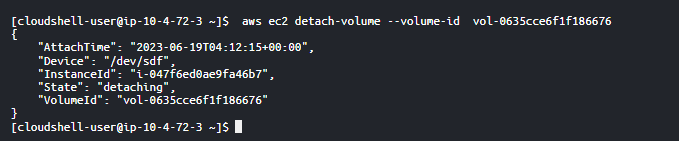
**Ahora desconecta y elimina el primer volumen, cuyo volume\_id obtuvo en el paso 1. Espera unos 10 segundos después de desconectar antes de intentar eliminar.**

**Salimos del modo root y vamos a una nueva pestaña para colocar los siguientes codigos:**

**aws ec2 detach-volume --volume-id volume\_id**

**aws ec2 delete-volume --volume-id volume\_id**

¿Cuáles son las salidas?





La salida del codigo detach es los detalles del volumen desadjuntando con la instancia y la 2da imagen no sale nada porque se ha eliminado correctamente el volumen creado

Repite estos dos comandos para el segundo volumen, cuyo volume\_id

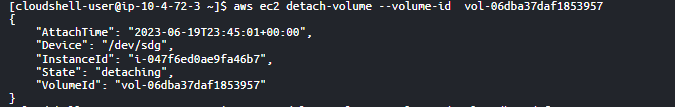
deberías haber obtenido del paso 6.

¿Qué comandos usastes?

**aws ec2 detach-volume --volume-id volume\_id:**

**aws ec2 delete-volume --volume-id volume\_id:**

¿Cuáles son las salidas?





**Lo mismo que el anterior paso, salen detalles del volúmenes desadjuntando con la instancias y su eliminacion**

1. Elimina la instantánea con lo siguiente usando su snapshot\_id del paso 5.

aws ec2 delete-snapshot --snapshot-id *snapshot\_id.*

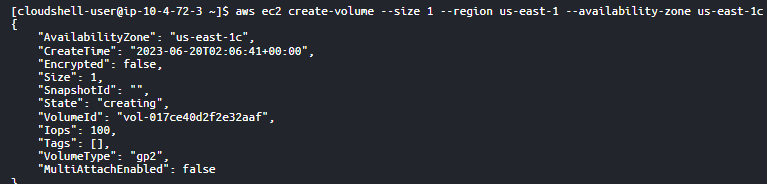
**

1. Cambie a la terminal. De lo que aprendiste en la parte 1, crea dos volúmenes de 1 GB en la zona de disponibilidad us-east-1c.

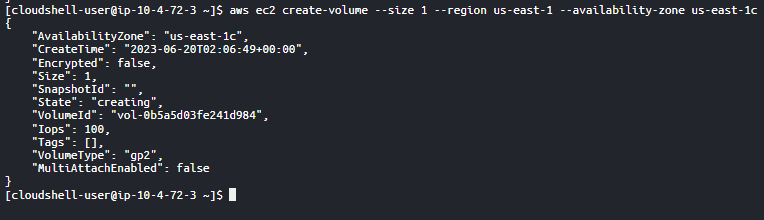
¿Qué comandos ejecutaste?

**aws ec2 create-volume --size 1 --region us-east-1--availability-zone us-east-1c**

¿Cuáles son las salidas?



Las informaciones detalladas del volumen creado



La informacion de la creacion del volumen con el uso de tamaño de gigabyte, su region , etc

Adjunta ambos volúmenes a tu instancia EC2, haciendo que aparezcan como /dev/sdh1 y /dev/sdh2, respectivamente. instancia ID= i-047f6ed0ae9fa46b7

Volumen1= vol-017ce40d2f2e32aaf

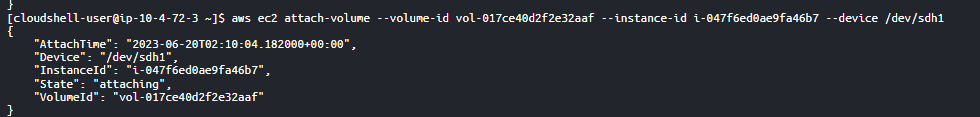
Volumen2=vol-0b5a5d03fe241d984

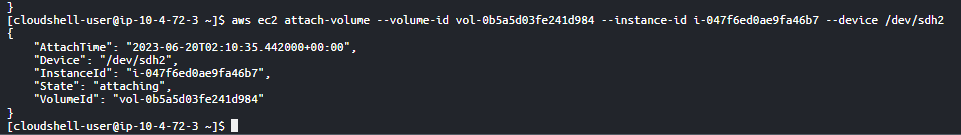
¿Qué comandos ejecutaste?

**aws ec2 attach-volume --volume-id volume\_id --instance-id id\_instance --device /dev/sdh1**

**aws ec2 attach-volume --volume-id volume\_id --instance-id id\_instance --device /dev/sdh2**

¿Cuáles son las salidas?





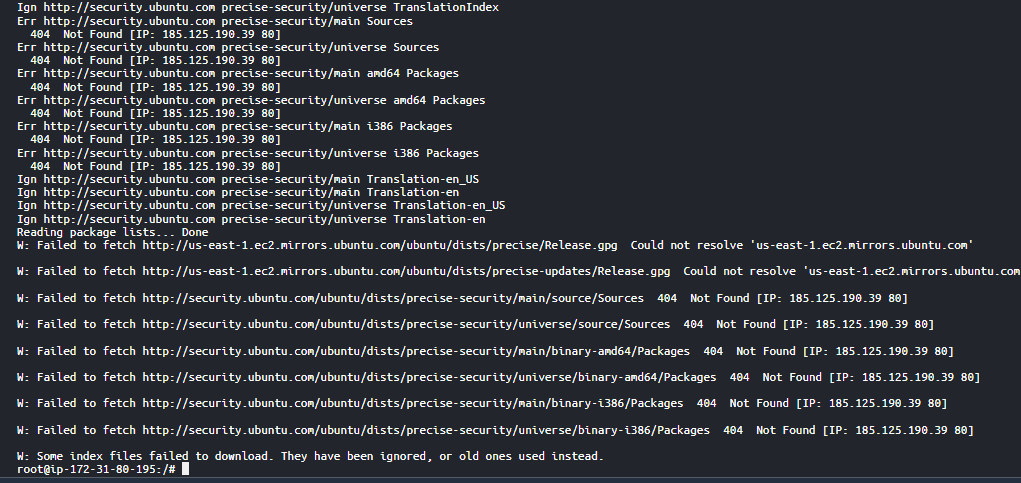
1. Cambia al terminal de la instancia EC2. Usaremos el programa mdadm de Linux para configurar los volúmenes en una configuración RAID. Instala mdadm de la siguiente manera.

apt-get update

apt-get install mdadm



Escribe "y" y presiona enter cuando se te solicite, seleccione "No configuration" cuando se te solicite y presiona enter.



Ahora ejecutamos mdadm para crear un arreglo RAID 0 en los dos volúmenes. Ejecuta lo siguiente. Donde vea "renamed\_/dev/sdh1" y "renamed\_/dev/shd2", usa los nombres que se te proporcionó AWS en el paso 11.

mdadm --create /dev/md0 --level 0 --metadata=1.1

--raid-devices 2 renamed\_/dev/sdh1 renamed\_/dev/sdh2

¿Cuál es la salida?

1. Ahora, podemos comprobar el estado de la matriz RAID 0. Emite lo siguiente. mdadm --detail /dev/md0

¿Cuál es la salida? Tenemos que agregar un sistema de archivos al arreglo RAID 0. Entonces queremos montarlo. Haz lo siguiente.

mkfs /dev/md0 mkdir /data3

mount /dev/md0 /data3

El comando df de Linux muestra información sobre los sistemas de archivos montados. ¿Cuál es la salida?

1. Finalizamos este laboratorio deteniendo el arreglo RAID 0, separando y eliminando ambos volúmenes de EBS y luego finalizando la instancia EC2. Para detener el arreglo RAID 0, haz lo siguiente desde su instancia EC2.

cd /

unmount /dev/md0 mdadm --stop /dev/md0

Ahora, cambia a tu terminal. Separa y elimina ambos volúmenes de EBS. ¿Qué comandos ejecutaste? ¿Cuáles son las salidas? Finaliza tu instancia EC2. ¿Qué comando ejecutaste? ¿Cuál es la salida?