**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

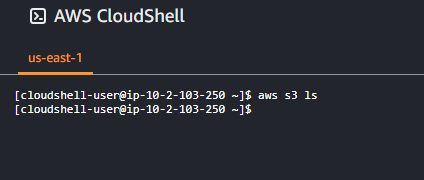
En este laboratorio, se estudiará el almacenamiento de Amazon S3. Utilizarás los comandos aws s3 y s3api para administrar datos en Amazon S3. Amazon S3 es un almacenamiento de objetos accesible a través de Internet.

# Parte 1: Operaciones básicas con S3

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

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

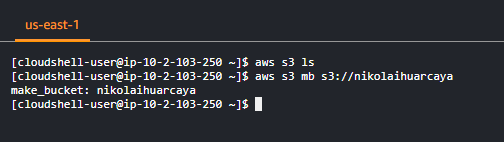
¿Cuál es la salida?



**Este comando se utiliza para listar los objetos y los buckets que se encuentran en nuestra cuenta de Amazon S3. Al ejecutar este comando, se recibirá una lista de los buckets existentes y los objetos almacenados dentro de cada bucket.**

1. Haz un bucket a través del siguiente comando mb. aws s3 mb s3://tu\_nombre\_de\_usuario

¿Cuál es la salida?

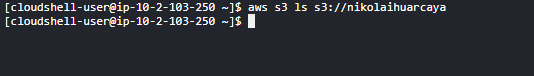
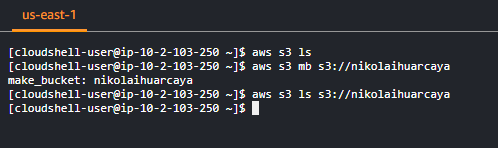


**Este comando nos está indicando a la AWS CLI que deseas crear un nuevo bucket en Amazon S3 con un nombre en especifico y en este caso nosotros pusimos como *nombre\_de\_usuario nikolaihuarcaya.*. Primero nosotros tenemos que especificar que es mb, este significa make bucket y el prefijo s3:// indica que se trata de un bucket de S3 Al darle enter al comando, la salida será un mensaje: “make\_bucket:nikolaihuarcaya” . Eso nos da a entender que el bucket se creó correctamente en Amazon S3 con el nombre nikolaihuarcaya**

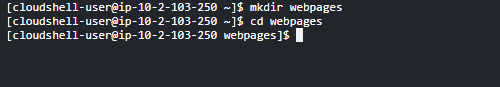
1. Enumera el contenido del bucket a través del siguiente comando l

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

¿Cuál es la salida?



**Lo que podemos notar con el comando que pusimos es que no hay una salida ya que está vacío porque recién se ha creado un bucket y como es un bucket recién creado, no tiene objeto/archivos.**

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

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

**Para la creación del archivo hello.html usaremos el comando cat > hello.html**

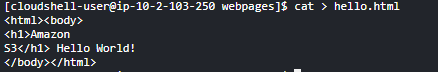
****

**Para que posteriormente dentro de este archivo pongamos lo siguiente :**

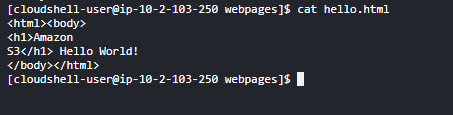
<html><body>

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

</body></html>



**Si es que queremos verificar que lo que pusimos en el archivo hello.html se encuentre correctamente solo pondremos el comando cat hello.html**

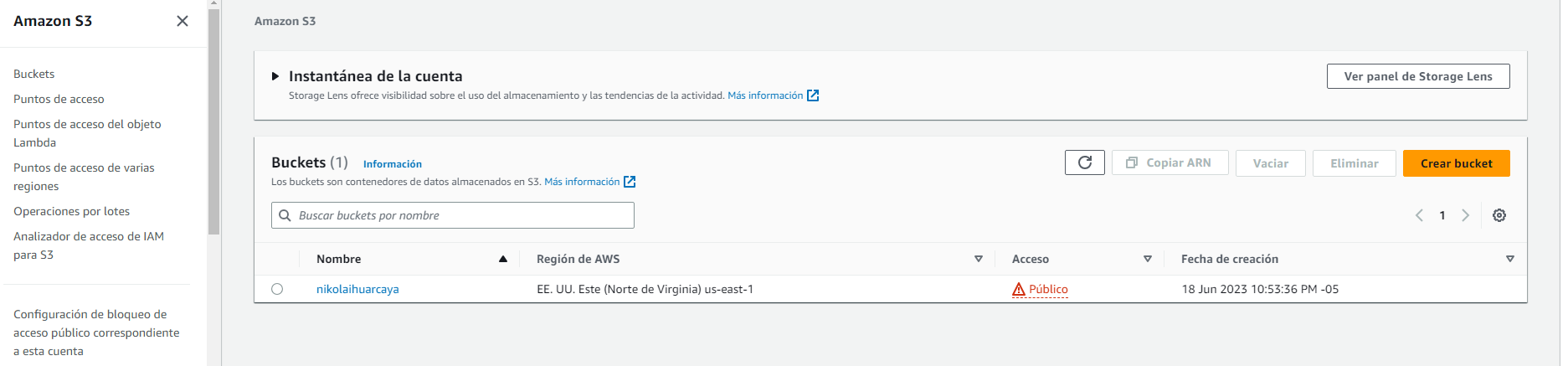
****

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

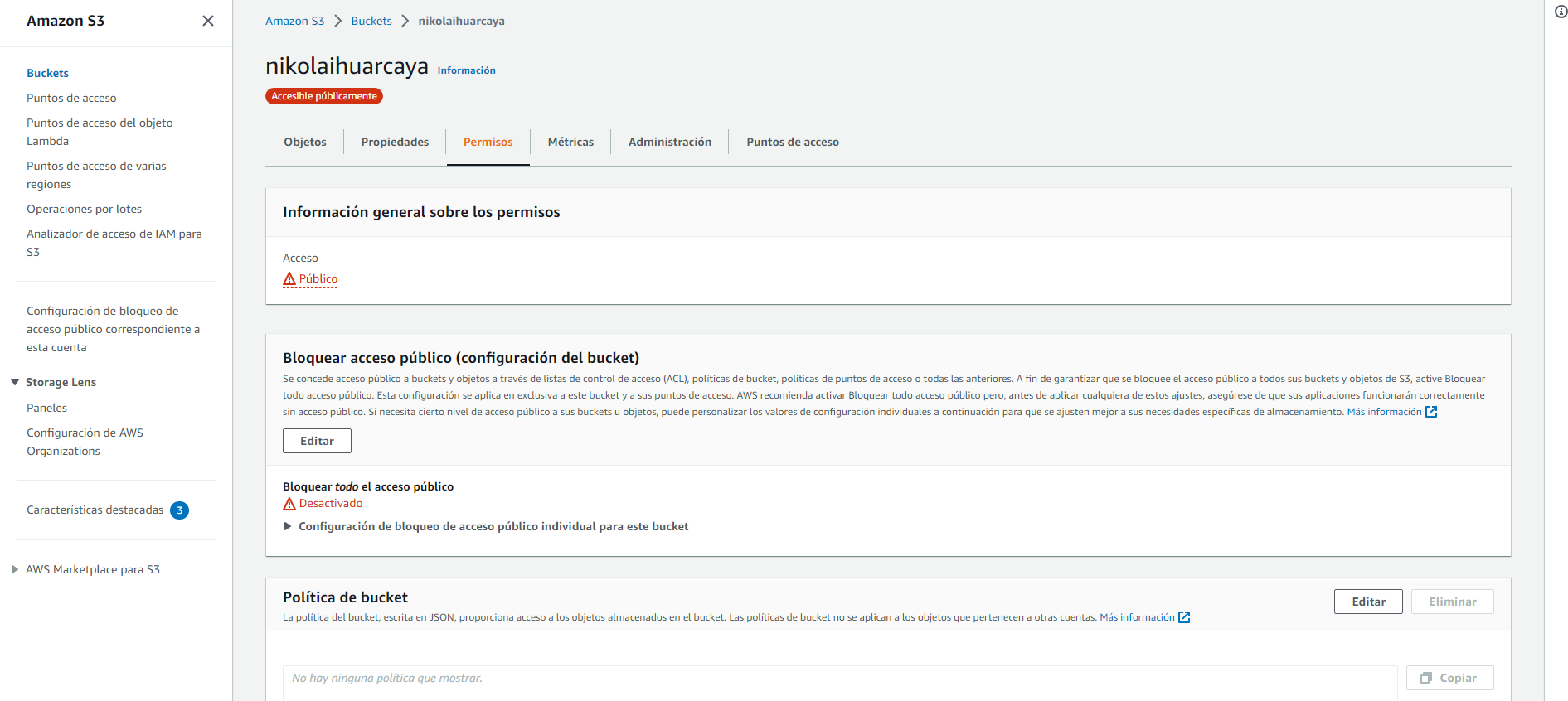
¿Cuál es la salida?

Primeramente nos saldrá error file y para que este bucket esté en público tendremos que hacer los siguientes pasos

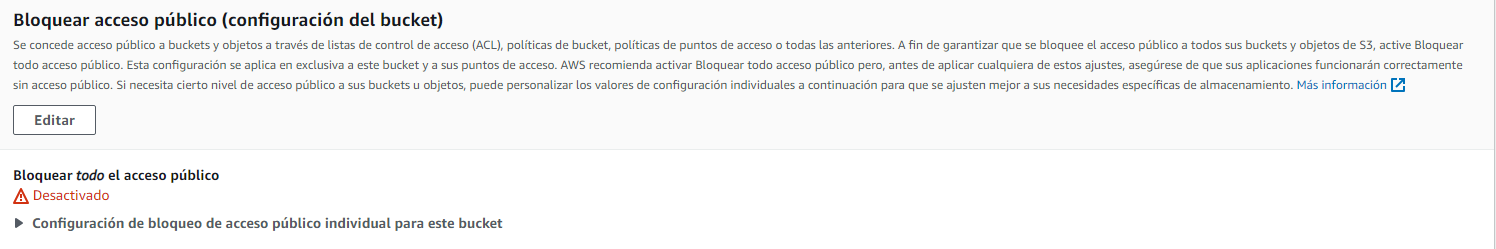
**4.1.- Tenemos que dirigirnos a Amazon S3 / en la imagen ya sale publico por que ya se hizo el cambio pero se mencionara los pasos para ponerlo en público.**

****

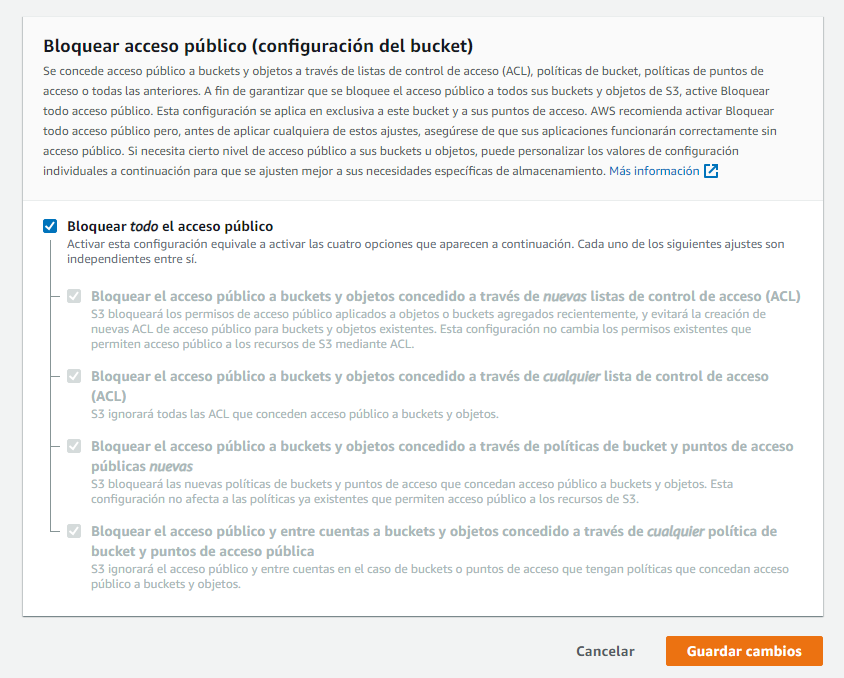
**4.2.- Hacemos click en el Buckets creado (nikolaihuarcaya) para luego dirigirnos a permisos .**

****

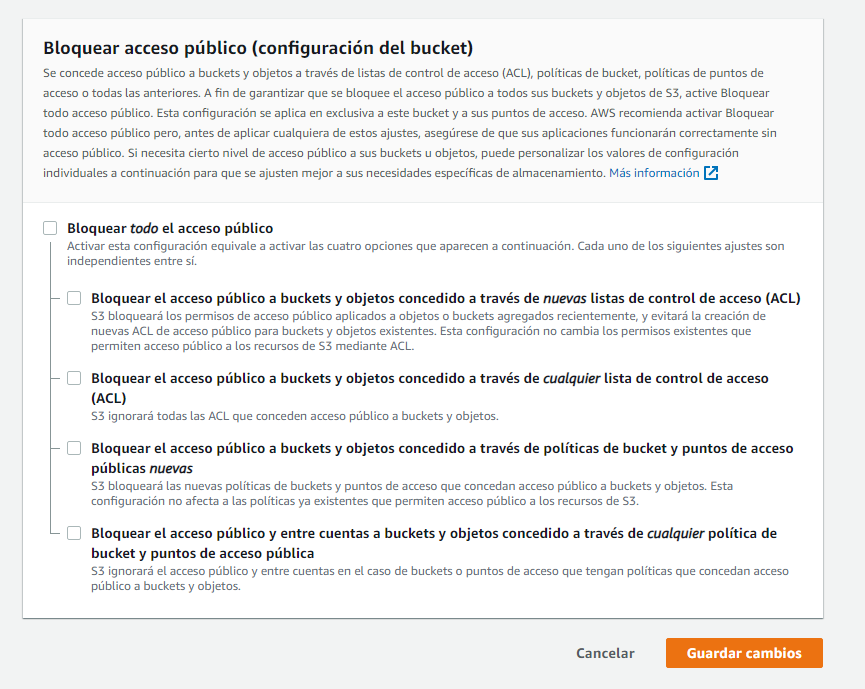
**4.3.-Vamos a “Bloquear acceso público” y hacemos click en Editar para editarlo.**

****

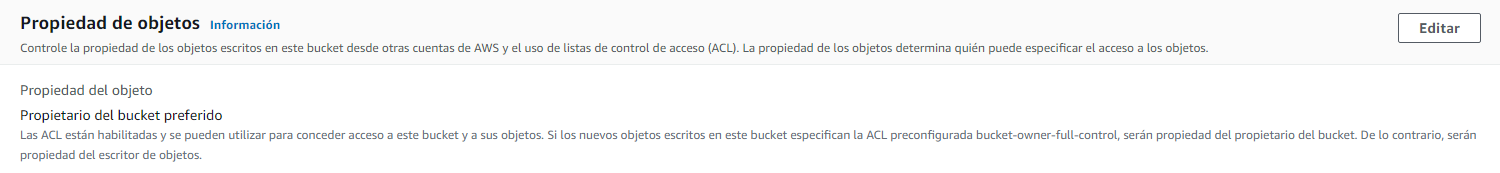
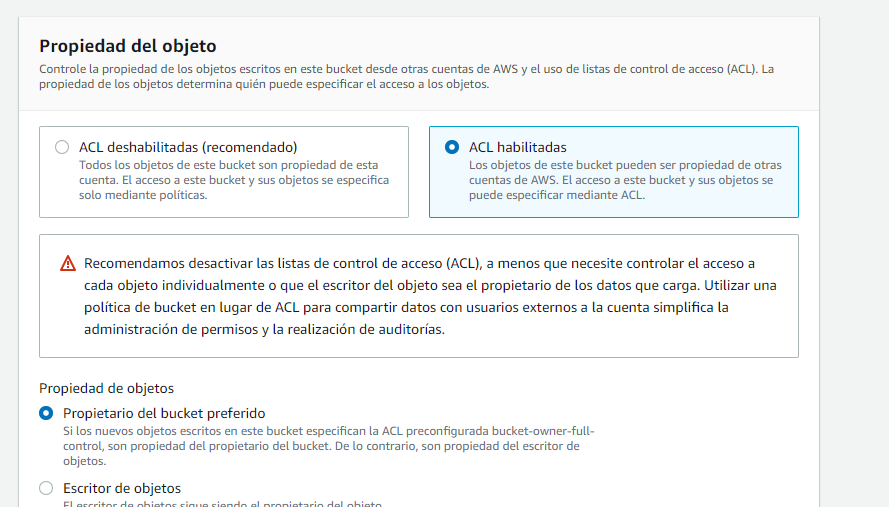
**4.4.-De lo que está marcado en “Bloquear todo el acceso público” lo desmarcamos y ponemos en guardar cambios , tenemos que esperar un momento para que suceda los cambios.**

****

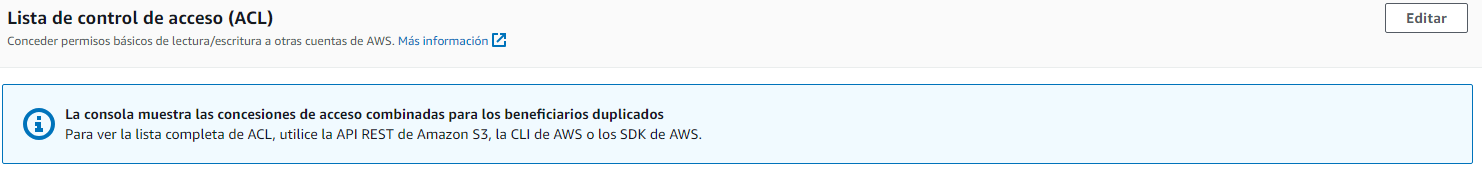
**4.5.-Cuando ya se realizó el cambio volvemos a entrar y podemos revisar que se guardó los cambios.**

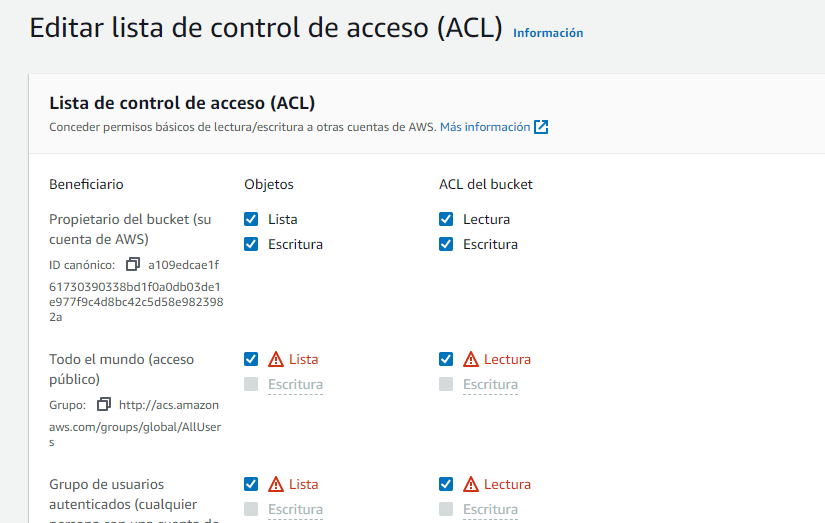
****

**4.6.- Ahora nos dirigimos a propiedades de objetos y hacemos click en editar para habilitar el ACL.**

****

**4.7.- Por último nos dirigimos a la lista de control de acceso (ACL) , para luego habilitarlos.**

****

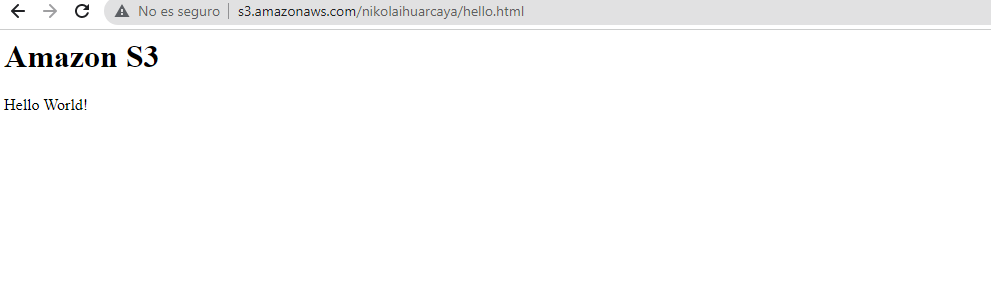


Ahora ponemos el comando para ver si funcionaba los cambios:

aws s3 cp hello.html s3://tu\_nombre\_de\_usuario --acl public-read



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?



Al momento de acceder a la página web con el URL que nos dio y obviamente modificándolo nos sale **Amazon S3 / Hello World!**

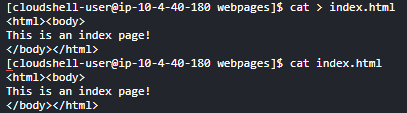
# Parte 2: alojamiento de sitios web estáticos con S3

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!

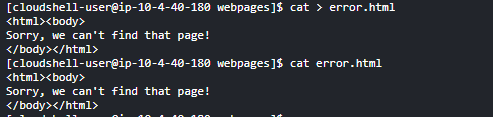
</body></html>

**Primero creamos para index con cat > index.html y para verificar que esté todo bien lo llamamos con cat.**

<html><body>

Sorry, we can't find that page!

</body></html>

**Ahora creamos para error con cat > error.html y para verificar que esté todo bien lo llamamos con cat.**

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



La salida nos dice que se ha cargado correctamente los archivos **error.html y index.html al bucket S3 nikolaihuarcaya,también quiero agregar que se encuentra el hello por que se hizo en el webpages.**

¿Cuál es la salida? 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

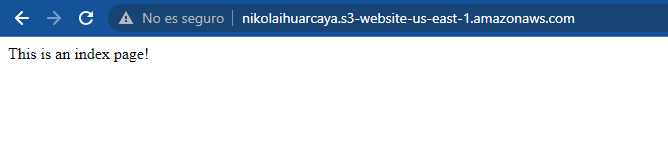
--error-document error.html



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

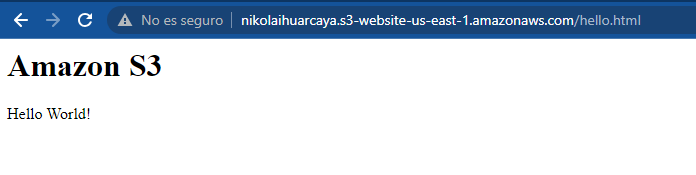
http://tu\_nombre\_de\_usuario.s3-website-us-east-1.amazonaws.com/.

¿Qué viste en el navegador? ¿Por qué?



**Lo que se puede observar en el navegador fue el mensaje del archivo index.html que se puso con anterioridad.**

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

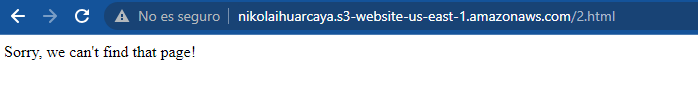


¿Qué viste en el navegador?

**Lo que podemos observar es el mensaje que pusimos en el 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? 

Lo que se puede observar es el mensaje que pusimos en el error.html

¿Por qué?

**Porque se está esperando en la configuración es que cuando accedes a cualquier URL que no corresponda a un archivo existente en tu sitio web estático.**

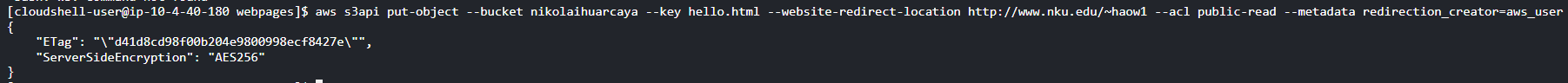
1. Podemos definir reglas de redirección y agregar metadatos a los objetos en el bucket. Ejecuta el siguiente comando para hacerlo. Observa que este comando usa s3api, no s3.

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

--key hello.html

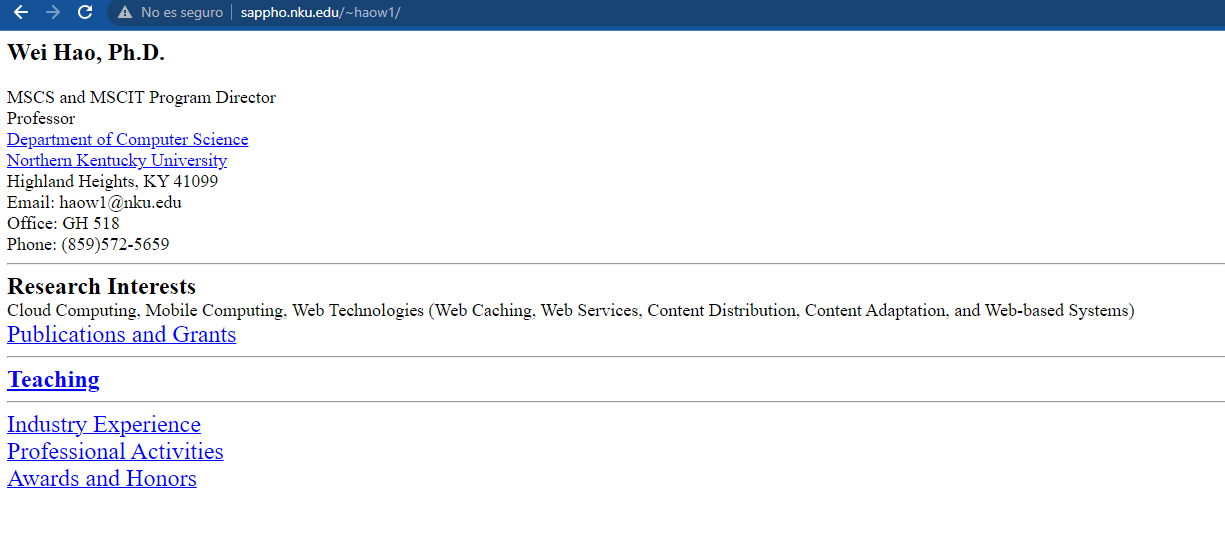
--website-redirect-location <http://www.nku.edu/~haow1>--acl public-read

--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?.

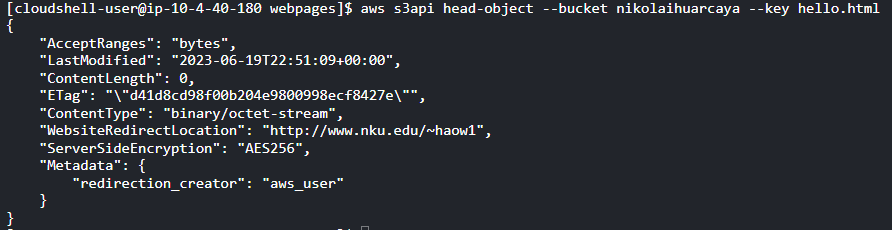


**Veo que ya no sale el mensaje Hello word y aparece como que una información de una persona llamada wei hao.**

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?

  
(EXPLICACIÓN MÁS ADELANTE)

# 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

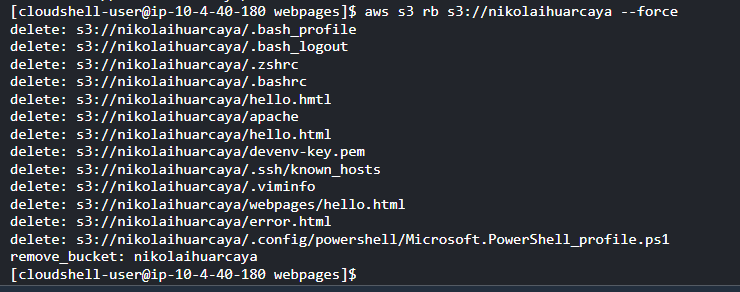


¿Cuál es la salida?

**La salida que nos muestra el comando 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 que se crearon con anterioridad y el bucket.También hay que agregar que 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

**Primero tenemos que hacer una aclaración ya que hay**

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

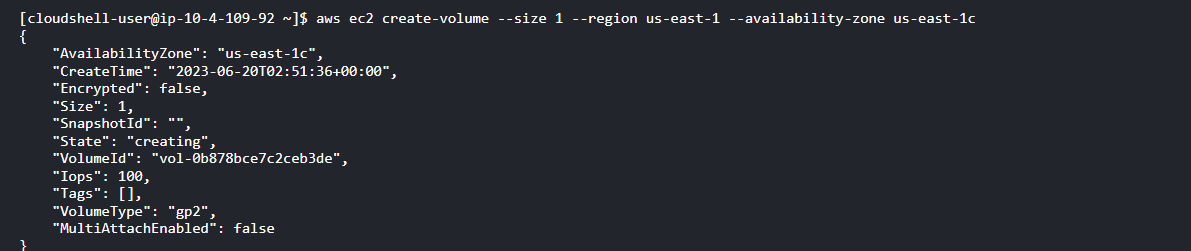
aws ec2 create-volume --size 1 --region us-east-1

--availability-zone us-east-1c

¿Qué significa este comando?

**Lo que que significa este comando es que se está creando un volumen con características como el tamaño del volumen en 1 gigabytes (--size 1) la región en la que se encuentra (--region us-east-1) y su zona de disponibilidad (--availability-zone us-east-1c).**

¿Cuál es la salida?



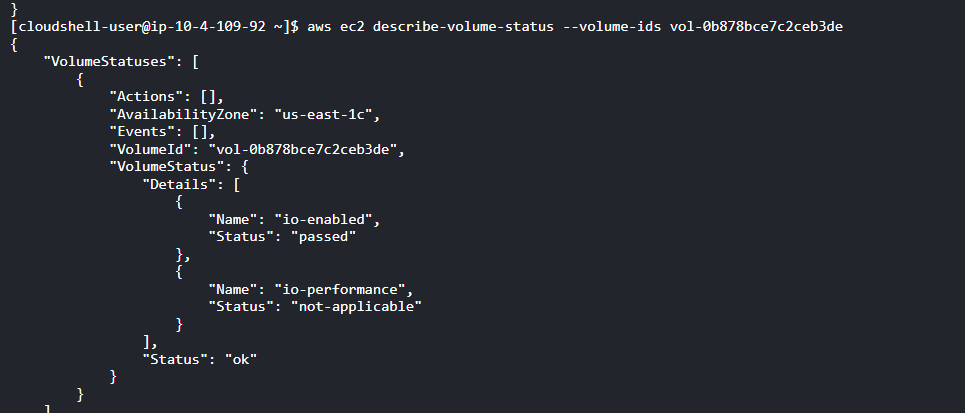
**También hay que mencionar que tenemos que acordarnos del VolumenId ya que lo vamos a necesitar para la siguiente pregunta para que así podamos hacer correr el comando correctamente.**

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?

Cómo se mencionó con anterioridad el VolumenId se pegó en la parte donde decir volume\_id , ya que tenemos que reemplazarlo con nuestro id de volumen.



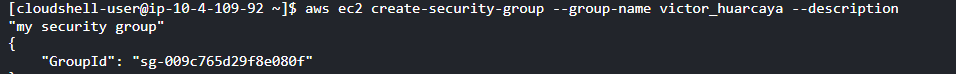
1. Para crear una instancia de EBS, hazlo siguiente.

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

–instance-type t1.micro –key-name

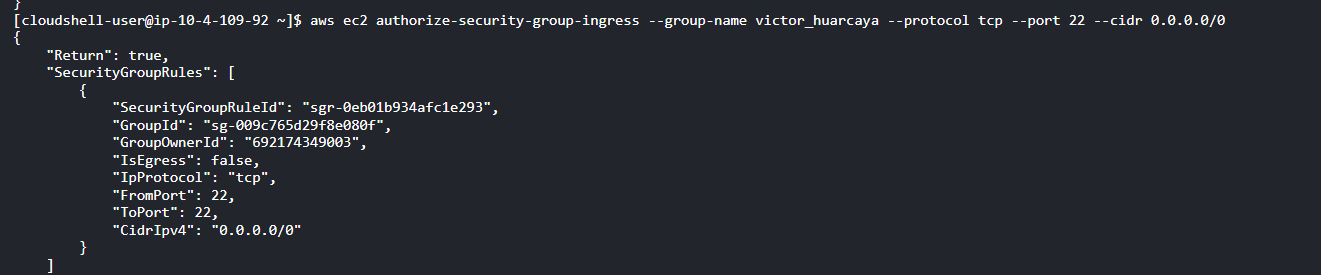
tu\_nombre\_de\_usuario-key --security-groups tu\_nombre\_de\_usuario

--placement AvailabilityZone=us-east-1c

**Antes de ello tenemos que crear nuestro usuario y los puertos 22 y 80 para que al final podamos correr el comando que nos pide, esto lo podemos hacer gracias a la práctica calificada 3 ya que usamos los comandos necesarios para crear una instancia EBS.**

Ahora creamos el puerto 22 y 80

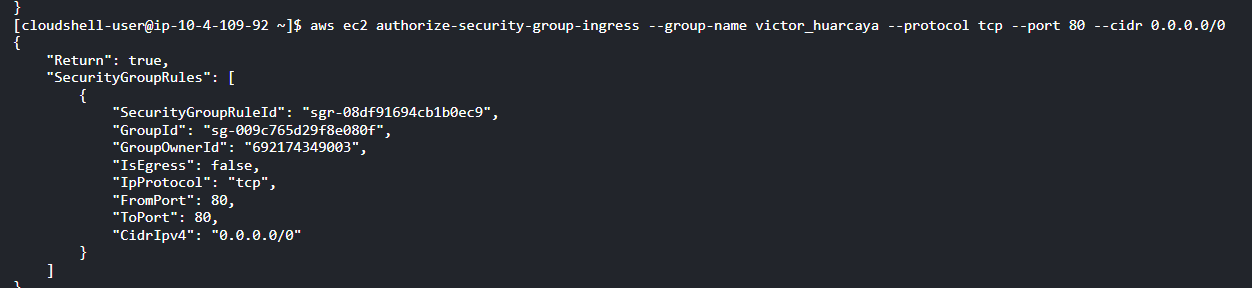
**Puerto 22: Es el puerto utilizado por el protocolo SSH (Secure Shell)**

aws ec2 authorize-security-group-ingress --group-name victor\_huarcaya --protocol tcp --port 22 --cidr 0.0.0.0/0

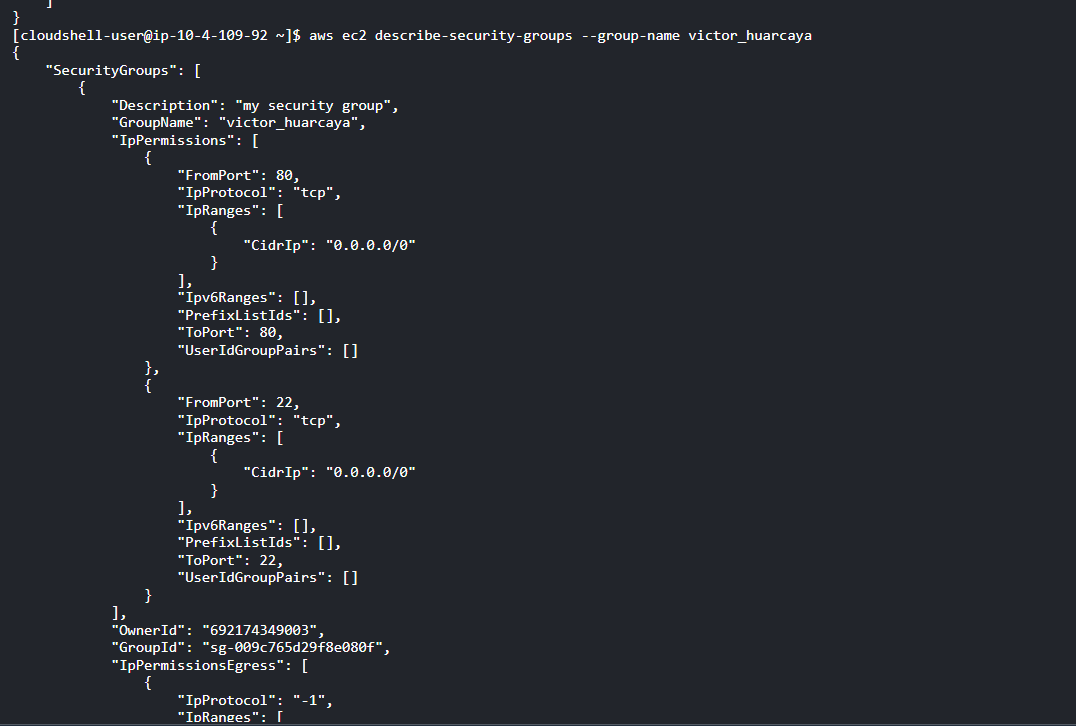
**Puerto 80 : Ee refiere a HTTP o al servicio Web**

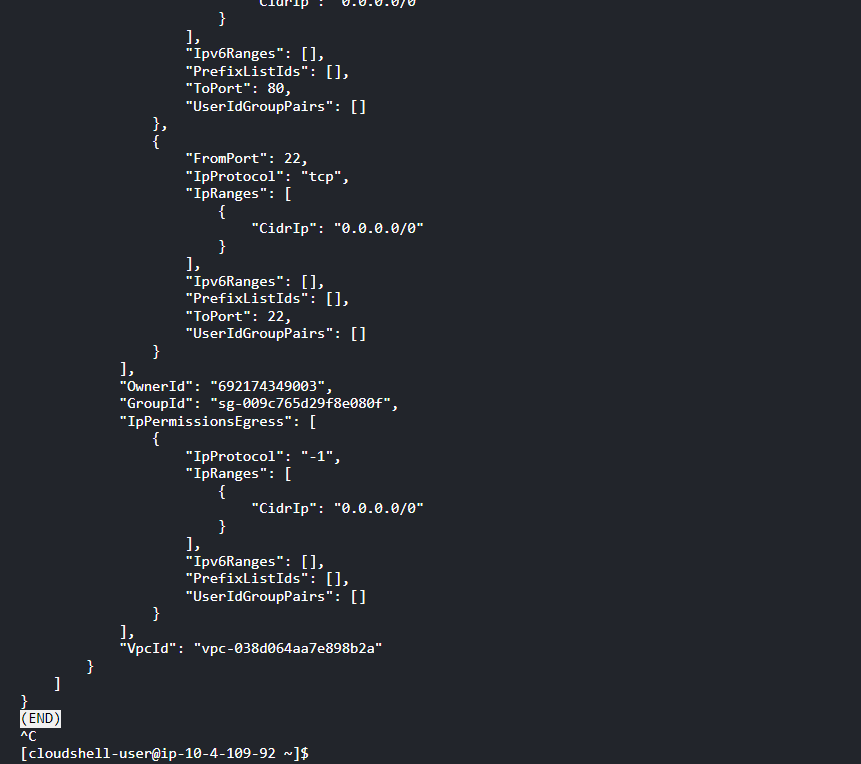
Para el puerto 80 utilizaremos el mismo código que utilizamos para el puerto 22

aws ec2 authorize-security-group-ingress --group-name victor\_huarcaya --protocol tcp --port 80 --cidr 0.0.0.0/0

****

Ahora podemos iniciar sesión de la instancia en ssh. Y lo hacemos con el siguiente código:

**Describe los grupos de seguridad especificados o todos sus grupos de seguridad**



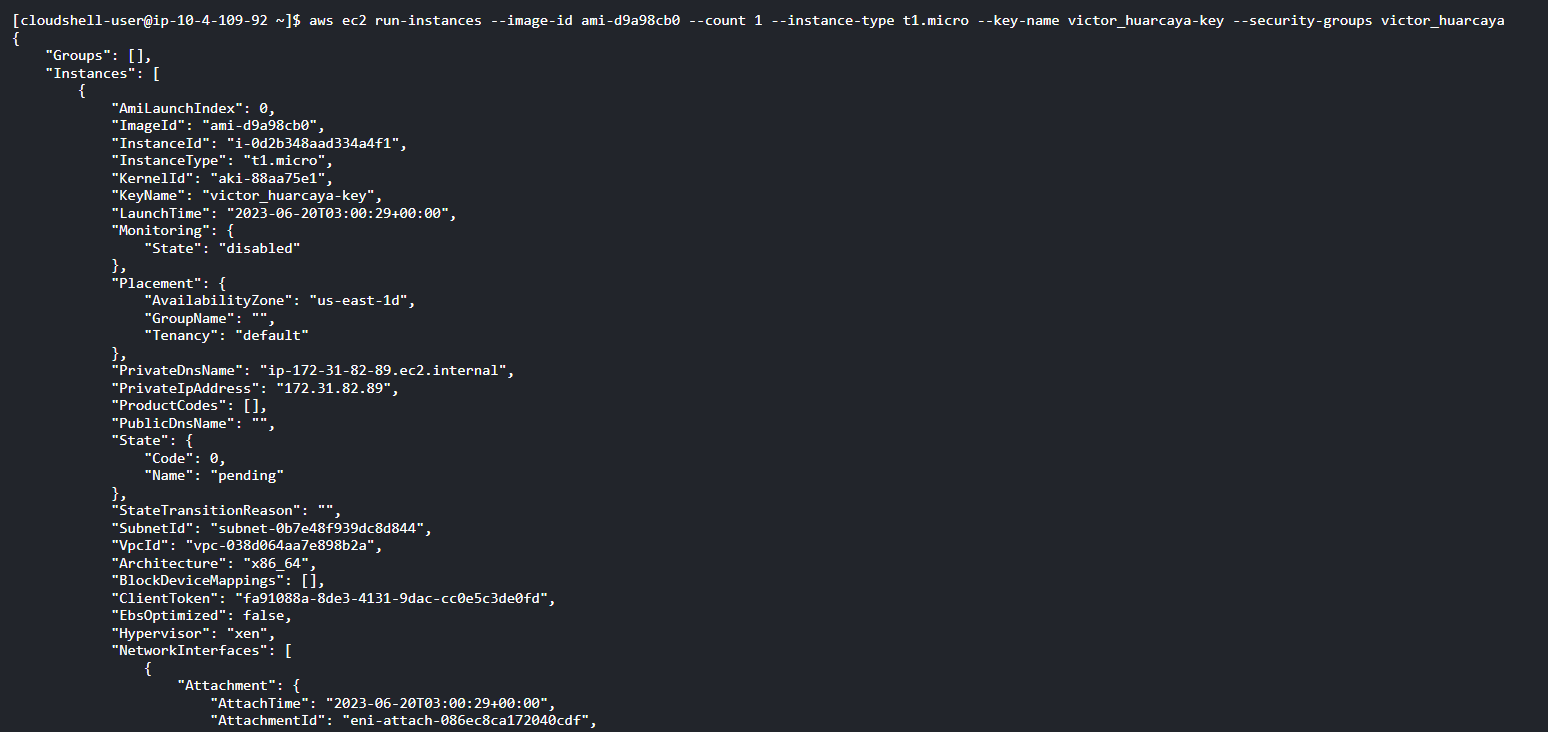
aws ec2 run-instances --image-id ami-d9a98cb0 --count 1

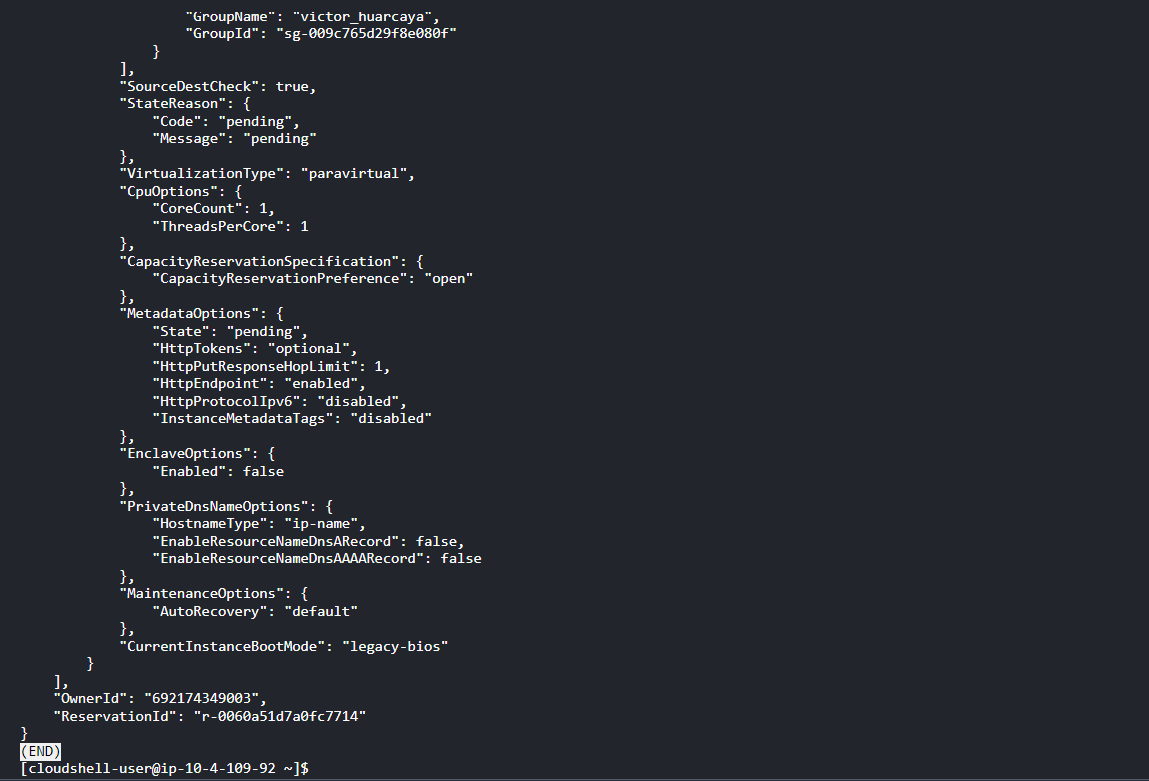
–instance-type t1.micro –key-name

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

--placement AvailabilityZone=us-east-1c

**Ahora recién podremos correr este comando , porque ya cumplimos con los pasos anteriores.**

****

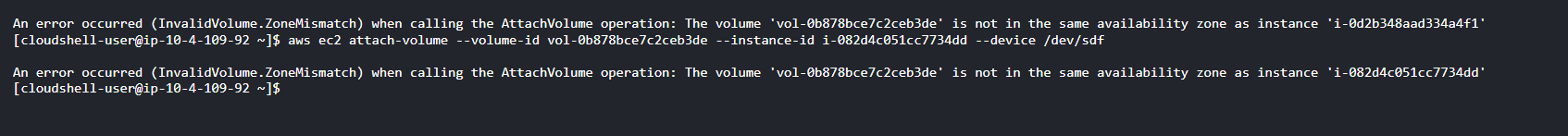
****

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

¿Cuál es la salida?



No se pudo por el tema de la localización del archivo .

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

# 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

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? 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? 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.

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 --región us-east-1

--availability-zone us-east-1c

--snapshot-id snapshot\_id

¿Cuál es la salida? Comprueba el estado del volumen. ¿Qué comando ejecutaste para verificar el estado? ¿Cuál es la salida?

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

¿Cuál es la salida?

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 ejecutó para lograr ambas tareas? Cambia el directorio de su instancia EC2 a /data2. ¿Viste el archivo aws\_user.txt?
2. Ahora queremos desmontar nuestros volúmenes, para lo cual usamos el comando umount. 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

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.

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

¿Cuáles son las salidas? Repite estos dos comandos para el segundo volumen, cuyo volume\_id deberías haber obtenido del paso 6. ¿Qué comandos usastes?

¿Cuáles son las salidas?

1. Elimina la instantánea con lo siguiente usando su snapshot\_id del paso 5. aws ec2 delete-snapshot --snapshot-id *snapshot\_id.}*
2. 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? ¿Cuáles son las salidas? Adjunta ambos volúmenes a tu instancia EC2, haciendo que aparezcan como /dev/sdh1 y /dev/sdh2, respectivamente. ¿Qué comandos ejecutaste? ¿Cuáles son las salidas?
3. 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?