



183- [JAWS] - Lab - Administración del almacenamiento

Datos Generales:

Nombre: Tomás Alfredo Villaseca Constantinescu

País: Chile

Fecha: 06/11/2023

Contacto: tomas.villaseca.c@gmail.com

Al finalizar este laboratorio, podrá hacer lo siguiente:

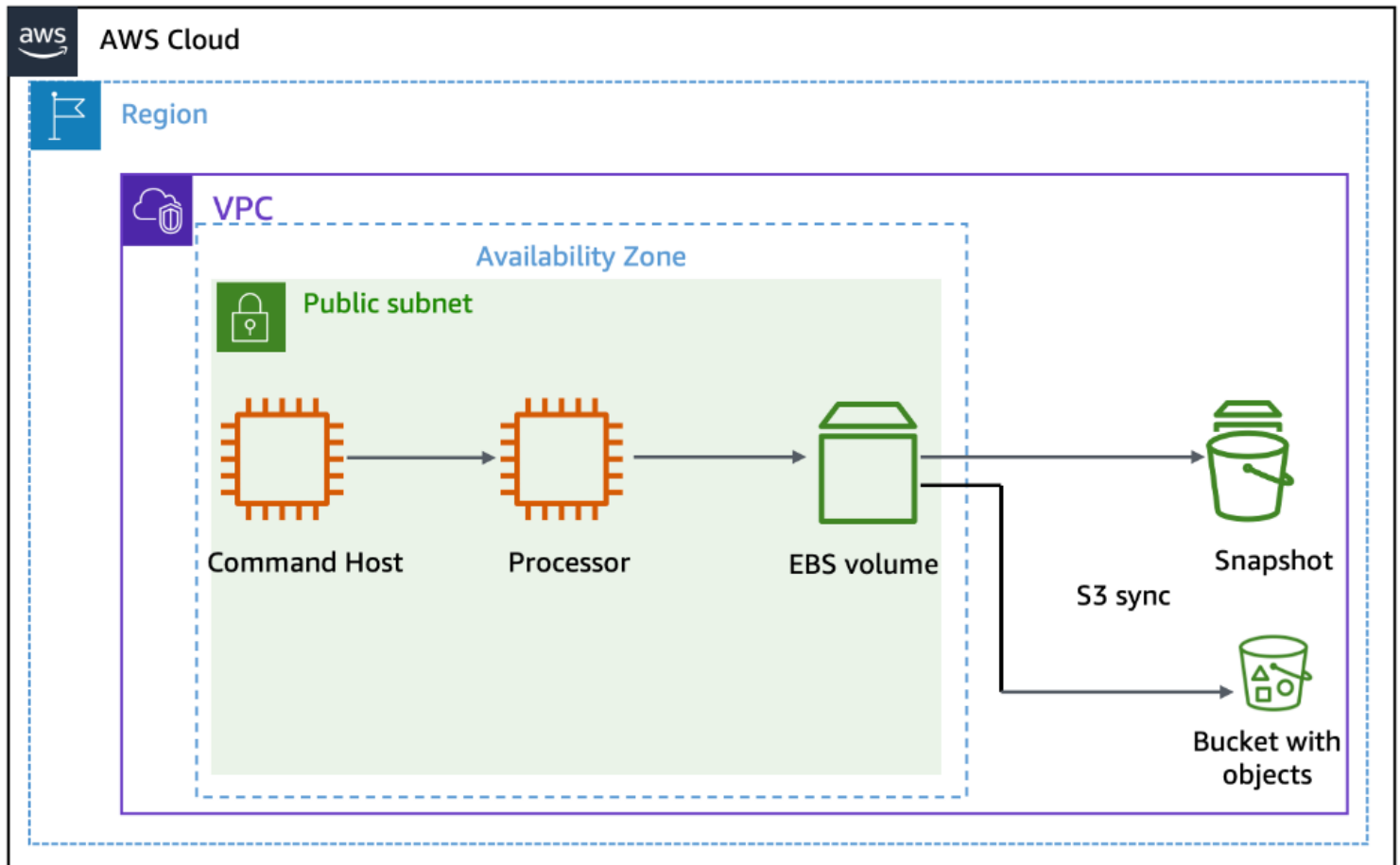
- Crear y mantener snapshots para instancias de Amazon EC2.
- Utilizar la sincronización de Amazon S3 para copiar archivos de un volumen de EBS a un bucket de S3.
- Utilizar el control de versiones de Amazon S3 para recuperar archivos eliminados.

Resumen Laboratorio:

AWS ofrece varias maneras de administrar datos en volúmenes de Amazon EBS.

En este laboratorio, utilizará la CLI de AWS para crear snapshots de un volumen de EBS y configurar un programador para ejecutar scripts de Python con el fin de eliminar snapshots antiguos.

En la sección de desafíos de este laboratorio, deberá sincronizar el contenido de un directorio de un volumen EBS con un bucket de Amazon S3 mediante un comando de sincronización de Amazon S3.



Su entorno de laboratorio consiste en una VPC con una subred pública.

Las instancias de Amazon EC2 denominadas "Command Host" y "Processor" ya se han creado en esta VPC para usted como parte de este laboratorio.

La instancia "Command Host" se utilizará para administrar los recursos de AWS, incluida la instancia "Processor".

Tarea 1: Crear y configurar recursos

Tarea 1.1 – Crear un S3 Bucket

En esta tarea, se creará un bucket S3 para sincronizar archivos desde un volumen EBS.

Paso 1: S3 → Create Bucket

- Bucket name → labbucket123123123
- Region → Default

Bucket name

labbucket123123123

Bucket name must be unique within the global namespace and follow the bucket naming rules. [See rules for bucket naming.](#)

AWS Region

US West (Oregon) us-west-2

	Name	AWS Region
●	labbucket123123123	US West (Oregon) us-west-2

Tarea 1.2 – Adjuntar perfil de instancia a Processor

En esta tarea, se adjuntará un rol de IAM creado previamente como perfil de instancia a la instancia EC2 "Processor", otorgándole los permisos para interactuar con otros servicios de AWS como volúmenes EBS y buckets S3.

Paso 1: EC2 → Instances → Processor → Actions → Security → Modify IAM role

	Name	Instance ID	Instance status
✓	Processor	i-045ebfc846fd72a4e	Running

- IAM role → S3BucketAccess
- Update IAM role

IAM role

Select an IAM role to attach to your instance or create a new role if you haven't created any. The role you select replaces any roles that are currently attached to your instance.

S3BucketAccess






[Create new IAM role](#)

Tarea 2: Crear Snapshots de su instancia

Tarea 2.1 – Conectarse a instancia Command Host

Paso 1: EC2 → Instances → Command Host → Connect → EC2 Instance Connect

EC2 Instance Connect	Session Manager	SSH client	EC2 serial console
<p>Instance ID</p> <p> i-0a2b0d9c60865bc77 (Command Host)</p> <p>Connection Type</p> <div><p><input checked="" type="radio"/> Connect using EC2 Instance Connect</p><p>Connect using the EC2 Instance Connect browser-based client, with a public IPv4 address.</p></div> <div><p><input type="radio"/> Connect using EC2 Instance Connect Endpoint</p><p>Connect using the EC2 Instance Connect browser-based client, with a private IPv4 address and a VPC endpoint.</p></div> <p>Public IP address</p> <p> 35.89.225.124</p> <p>User name</p> <p>Enter the user name defined in the AMI used to launch the instance. If you didn't define a custom user name, use the default user name, ec2-user.</p> <div><input type="text" value="ec2-user"/></div> <div><p> Note: In most cases, the default user name, ec2-user, is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI user name.</p></div>			

```
[ec2-user@ip-10-5-0-7 ~]$
```

```
#  
~\_##### Amazon Linux 2  
~~\_#####  
~~\_###| AL2 End of Life is 2025-06-30.  
~~\_#/V~'-'>  
~~~~A newer version of Amazon Linux is available!  
~~~~_-./-/Amazon Linux 2023, GA and supported until 2028-03-15.  
_/m/'-/_https://aws.amazon.com/linux/amazon-linux-2023/  
  
[ec2-user@ip-10-5-0-7 ~]$
```

Tarea 2.2 – Tomar una Snapshot inicial

En esta tarea, se identificará el volumen EBS adjunto a la instancia "Processor" y se tomará una snapshot inicial.

Paso 1: Para obtener el EBS Volume-id, ejecutar el siguiente comando:

```
aws ec2 describe-instances --filter 'Name=tag:Name,Values=Processor' --query 'Reservations[0].Instances[0].BlockDeviceMappings[0].Ebs.{volumeId:volumeId}'
```

```
[ec2-user@ip-10-5-0-7 ~]$ aws ec2 describe-instances --filter 'Name=tag:Name,Values=Processor' --query 'Reservations[0].Instances[0].BlockDeviceMappings[0].Ebs.{volumeId:volumeId}'
{
  "VolumeId": "vol-0237a0fa8f47724a3"
}
[ec2-user@ip-10-5-0-7 ~]$
```

- EBS Volume-id → vol-0237a0fa8f47724a3

Paso 2: Para obtener el Instance ID de la instancia Processor, ejecutar el siguiente comando:

```
aws ec2 describe-instances --filters 'Name=tag:Name,Values=Processor' --query 'Reservations[0].Instances[0].InstanceId'
```

```
[ec2-user@ip-10-5-0-7 ~]$ aws ec2 describe-instances --filters 'Name=tag:Name,Values=Processor' --query 'Reservations[0].Instances[0].InstanceId'
"i-045ebfc846fd72a4e"
[ec2-user@ip-10-5-0-7 ~]$
```

- Instance ID → i-045ebfc846fd72a4e

Paso 3: Para apagar la instancia Processor, ejecutar el siguiente comando:

- Reemplazar Instance ID copiado anteriormente.

```
aws ec2 stop-instances --instance-ids INSTANCE-ID
```

```
[ec2-user@ip-10-5-0-7 ~]$ aws ec2 stop-instances --instance-ids i-045ebfc846fd72a4e
{
  "StoppingInstances": [
    {
      "CurrentState": {
        "Code": 64,
        "Name": "stopping"
      },
      "InstanceId": "i-045ebfc846fd72a4e",
      "PreviousState": {
        "Code": 16,
        "Name": "running"
      }
    }
  ]
}
[ec2-user@ip-10-5-0-7 ~]$
```

Paso 4: Para verificar que la instancia Processor fue detenida, ejecutar el siguiente comando:

- Reemplazar Instance ID copiado anteriormente.

```
aws ec2 wait instance-stopped --instance-id INSTANCE-ID
```

✓	Processor	i-045ebfc846fd72a4e	⊖ Stopped
---	-----------	---------------------	-----------

Paso 5: Para crear un snapshot inicial de la instancia Processor, ejecutar el siguiente comando:

- Reemplazar VOLUME-ID por EBS Volume-id copiado anteriormente.

```
aws ec2 create-snapshot --volume-id VOLUME-ID
```

```
[ec2-user@ip-10-5-0-7 ~]$ aws ec2 create-snapshot --volume-id vol-0237a0fa8f47724a3
{
  "Description": "",
  "Encrypted": false,
  "OwnerId": "686324501633",
  "Progress": "",
  "SnapshotId": "snap-02d52fad1b1b1a8ab",
  "StartTime": "2023-11-06T21:05:12.279Z",
  "State": "pending",
  "VolumeId": "vol-0237a0fa8f47724a3",
  "VolumeSize": 8,
  "Tags": []
}
[ec2-user@ip-10-5-0-7 ~]$
```

- Snapshot ID → snap-02d52fad1b1b1a8ab

Paso 6: Para revisar el estado del snapshot, ejecutar el siguiente comando:

- Reemplazar Snapshot ID por el copiado anteriormente.

```
aws ec2 wait snapshot-completed --snapshot-id SNAPSHOT-ID
```

```
[ec2-user@ip-10-5-0-7 ~]$ aws ec2 wait snapshot-completed --snapshot-id snap-02d52fad1b1b1a8ab
[ec2-user@ip-10-5-0-7 ~]$
```

Paso 7: Para reiniciar la instancia Processor, ejecutar el siguiente comando:

- Reemplazar Instance ID copiado anteriormente.

```
aws ec2 start-instances --instance-ids INSTANCE-ID
```

```
[ec2-user@ip-10-5-0-7 ~]$ aws ec2 start-instances --instance-ids i-045ebfc846fd72a4e
{
  "StartingInstances": [
    {
      "CurrentState": {
        "Code": 0,
        "Name": "pending"
      },
      "InstanceId": "i-045ebfc846fd72a4e",
      "PreviousState": {
        "Code": 80,
        "Name": "stopped"
      }
    }
  ]
}
```

Tarea 2.3 – Programar la creación de Snapshots posteriores

Utilizando el sistema de programación de Linux (cron), puede configurar un proceso recurrente de snapshots para que las nuevas snapshot de sus datos se tomen automáticamente.

Para los fines de este laboratorio, usted programa la creación de una instantánea cada minuto para que pueda verificar los resultados de su trabajo.

En esta tarea, se creará un trabajo cron para gestionar el número de snapshots que se mantienen para un volumen.

Paso 1: Para crear y programar una entrada cron que ejecute un trabajo cada minuto, ejecute el siguiente comando:

- Reemplazar VOLUME-ID por EBS Volume-id copiado anteriormente.

```
echo "* * * * * aws ec2 create-snapshot --volume-id VOLUME-ID 2>&1 >> /tmp/cronlog" > cronjob
crontab cronjob
```

```
[ec2-user@ip-10-5-0-7 ~]$ echo "* * * * * aws ec2 create-snapshot --volume-id vol-0237a0fa8f4772
[ec2-user@ip-10-5-0-7 ~]$ crontab cronjob
[ec2-user@ip-10-5-0-7 ~]$
```


Paso 2: Para verificar que los snapshots posteriores se están creando, ejecutar el siguiente comando:

- Reemplazar VOLUME-ID por EBS Volume-id copiado anteriormente.

```
aws ec2 describe-snapshots --filters "Name=volume-id,Values=VOLUME-ID"
```

```
[ec2-user@ip-10-5-0-7 ~]$ aws ec2 describe-snapshots --filters "Name=volume-id,Values=vol-0237a0fa8f47724a3"
{
  "Snapshots": [
    {
      "Description": "",
      "Encrypted": false,
      "OwnerId": "686324501633",
      "Progress": "47%",
      "SnapshotId": "snap-031b032ec779bccbc",
      "StartTime": "2023-11-06T21:11:03.022Z",
      "State": "pending",
      "VolumeId": "vol-0237a0fa8f47724a3",
      "VolumeSize": 8,
      "StorageTier": "standard"
    },
    {
      "Description": "",
      "Encrypted": false,
      "OwnerId": "686324501633",
      "Progress": "100%",
      "SnapshotId": "snap-0d347d10e672371b8",
      "StartTime": "2023-11-06T21:10:03.095Z",
      "State": "completed",
      "VolumeId": "vol-0237a0fa8f47724a3",
      "VolumeSize": 8,
      "StorageTier": "standard"
    },
    {
      "Description": "",
      "Encrypted": false,
      "OwnerId": "686324501633",
      "Progress": "100%",
      "SnapshotId": "snap-02d52fad1b1b1a8ab",
      "StartTime": "2023-11-06T21:05:12.279Z",
      "State": "completed",
      "VolumeId": "vol-0237a0fa8f47724a3",
      "VolumeSize": 8,
      "StorageTier": "standard"
    }
  ]
}
```


Tarea 2.4 – Retener las últimas dos Snapshots

En esta tarea, se ejecutará un script de Python que mantiene sólo las dos últimas snapshots de cualquier volumen EBS.

Paso 1: Para detener el trabajo cron, ejecutar el siguiente comando:

```
crontab -r
```

```
[ec2-user@ip-10-5-0-7 ~]$ crontab -r
[ec2-user@ip-10-5-0-7 ~]$
```

Paso 2: Para examinar el contenido del script de Python “snapshotter_v2.py”, ejecutar el siguiente comando:

```
more /home/ec2-user/snapshotter_v2.py
```

```
[ec2-user@ip-10-5-0-7 ~]$ more /home/ec2-user/snapshotter_v2.py
#!/usr/bin/env python

import boto3

MAX_SNAPSHOTS = 2    # Number of snapshots to keep

# Create the EC2 resource
ec2 = boto3.resource('ec2')

# Get a list of all volumes
volume_iterator = ec2.volumes.all()

# Create a snapshot of each volume
for v in volume_iterator:
    v.create_snapshot()

# Too many snapshots?
snapshots = list(v.snapshots.all())
if len(snapshots) > MAX_SNAPSHOTS:

    # Delete oldest snapshots, but keep MAX_SNAPSHOTS available
    snap_sorted = sorted([(s.id, s.start_time, s) for s in snapshots], key=lambda k: k[1])
    for s in snap_sorted[:-MAX_SNAPSHOTS]:
        print("Deleting snapshot", s[0])
        s[2].delete()
[ec2-user@ip-10-5-0-7 ~]$
```

El script busca todos los volúmenes EBS asociados a la cuenta del usuario actual y toma instantáneas. Luego, examina el número de instantáneas asociadas al volumen, ordena las instantáneas por fecha y elimina todas las instantáneas excepto las dos más recientes.

Paso 3: Para obtener los ID de los multiples snapshots generados, ejecutar el siguiente comando:

- Reemplazar VOLUME-ID por EBS Volume-id copiado anteriormente.

```
aws ec2 describe-snapshots --filters "Name=volume-id, values=VOLUME-ID" --query 'Snapshots[*].SnapshotId'
```

```
[ec2-user@ip-10-5-0-7 ~]$ aws ec2 describe-snapshots --filters
[
    "snap-031b032ec779bccbc",
    "snap-0d347d10e672371b8",
    "snap-02d52fad1b1b1a8ab",
    "snap-0e6a71327b3a9db91"
]
[ec2-user@ip-10-5-0-7 ~]$
```

Paso 4: Ejecutar el script de Python snapshotter_v2.py:

```
python3 snapshotter_v2.py
```

```
[ec2-user@ip-10-5-0-7 ~]$ python3 snapshotter_v2.py
/usr/local/lib/python3.7/site-packages/boto3/compat.py:35: PythonDeprecationWarning: The boto3.utils.get_service_name function is deprecated. Please use boto3.client('service_name').service_name.
  warnings.warn(warning, PythonDeprecationWarning)
Deleting snapshot snap-02d52fad1b1b1a8ab
Deleting snapshot snap-0e6a71327b3a9db91
Deleting snapshot snap-0d347d10e672371b8
[ec2-user@ip-10-5-0-7 ~]$
```

Paso 5: Para examinar el número de snapshots para el volumen actual, ejecutar el siguiente comando:

- Reemplazar VOLUME-ID por EBS Volume-id copiado anteriormente.

```
aws ec2 describe-snapshots --filters "Name=volume-id, values=VOLUME-ID" --query 'Snapshots[*].SnapshotId'
```

```
[ec2-user@ip-10-5-0-7 ~]$ aws ec2 descri
[
    "snap-031b032ec779bccbc",
    "snap-044bc4fce07e91d04"
]
[ec2-user@ip-10-5-0-7 ~]$
```

Tarea 3: Sincronizar archivos con Amazon S3

En esta tarea, se debe sincronizar el contenido de un directorio con el bucket de Amazon S3 que creó anteriormente.

Tarea 3.1 – Descargar y descomprimir archivos de muestra

El paquete de archivos de muestra contiene una carpeta con tres archivos de texto:

- archivo1.txt
- archivo2.txt
- archivo3.txt.

Estos son los archivos que sincronizará con su bucket de Amazon S3.

Paso 1: EC2 → Instances → Processor → Connect → EC2 Instance Connect

Name

Instance ID

Instance state

Processor

i-045ebfc846fd72a4e

Running

EC2 Instance Connect

Session Manager

SSH client

EC2 serial console

Instance ID

i-045ebfc846fd72a4e (Processor)

Connection Type

☒ Connect using EC2 Instance Connect

Connect using the EC2 Instance Connect browser-based client, with a public IPv4 address.

☐ Connect using EC2 Instance Connect Endpoint

Connect using the EC2 Instance Connect browser-based client, with a private IPv4 address and a VPC endpoint.

Public IP address

34.210.28.52

User name

Enter the user name defined in the AMI used to launch the instance. If you didn't define a custom user name, use the default user name, ec2-user.

ec2-user

Note: In most cases, the default user name, ec2-user, is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI user name.

Cancel

Connect

```

#
~\##### Amazon Linux 2
~\#####
~\#####| AL2 End of Life is 2025-06-30.
~\#/
~V~'-'>
~~~~
~..-
~/m/'-/_

```

A newer version of Amazon Linux is available!

Amazon Linux 2023, GA and supported until 2028-03-15.
<https://aws.amazon.com/linux/amazon-linux-2023/>

```
[ec2-user@ip-10-5-0-44 ~]$
```

Paso 2: Para descargar los archivos de muestra, ejecutar el siguiente comando:

```
wget https://aws-tc-largeobjects.s3.us-west-2.amazonaws.com/CUR-TF-100-RSJAWS-1-23732/183-lab-JAWS-managing-storage/s3/files.zip
```

```
[ec2-user@ip-10-5-0-44 ~]$ wget https://aws-tc-largeobjects.s3.us-west-2.amazonaws.com/CUR-TF-100-RSJAWS-1-23732/183-lab-JAWS-managing-storage/s3/files.zip
--2023-11-06 21:16:31-- https://aws-tc-largeobjects.s3.us-west-2.amazonaws.com/CUR-TF-100-RSJAWS-1-23732/183-lab-JAWS-managing-storage/s3/files.zip
Resolving aws-tc-largeobjects.s3.us-west-2.amazonaws.com (aws-tc-largeobjects.s3.us-west-2.amazonaws.com) [aws-tc-largeobjects.s3.us-west-2.amazonaws.com]
Connecting to aws-tc-largeobjects.s3.us-west-2.amazonaws.com (aws-tc-largeobjects.s3.us-west-2.amazonaws.com) [aws-tc-largeobjects.s3.us-west-2.amazonaws.com]:443
HTTP request sent, awaiting response... 200 OK
Length: 72110 (70K) [application/zip]
Saving to: 'files.zip'

100%[=====] 72110/72110 (40.9 MB/s)

2023-11-06 21:16:31 (40.9 MB/s) - 'files.zip' saved [72110/72110]

[ec2-user@ip-10-5-0-44 ~]$
```

Paso 3: Para descomprimir los archivos descargados, ejecutar el siguiente comando:

```
unzip files.zip
```

```
[ec2-user@ip-10-5-0-44 ~]$ ls
files.zip
[ec2-user@ip-10-5-0-44 ~]$ unzip files.zip
Archive:  files.zip
  inflating: files/file1.txt
  inflating: files/file2.txt
  inflating: files/file3.txt
[ec2-user@ip-10-5-0-44 ~]$ ls
files  files.zip
[ec2-user@ip-10-5-0-44 ~]$
```

Tarea 3.2 – Sincronizar archivos

Paso 1: Para activar el versioning del Bucket S3, ejecutar el siguiente comando:

- Reemplazar S3-Bucket-Name por Bucket Name copiado anteriormente.

```
aws s3api put-bucket-versioning --bucket S3-BUCKET-NAME --versioning-configuration Status=Enabled
```

```
[ec2-user@ip-10-5-0-44 ~]$ aws s3api put-bucket-versioning --bucket labbucket123123123 --versioning-configuration Status=Enabled
[ec2-user@ip-10-5-0-44 ~]$
```

Paso 2: Para sincronizar el contenido de la carpeta de archivos con tu bucket S3, ejecutar el siguiente comando:

- Reemplazar S3-Bucket-Name por Bucket Name copiado anteriormente.

```
aws s3 sync files s3://S3-BUCKET-NAME/files/
```

```
[ec2-user@ip-10-5-0-44 ~]$ aws s3 sync files s3://labbucket123123123/files/
upload: files/file2.txt to s3://labbucket123123123/files/file2.txt
upload: files/file1.txt to s3://labbucket123123123/files/file1.txt
upload: files/file3.txt to s3://labbucket123123123/files/file3.txt
[ec2-user@ip-10-5-0-44 ~]$
```

Paso 3: Para confirmar el estado de tus archivos, ejecutar el siguiente comando:

- Reemplazar S3-Bucket-Name por Bucket Name copiado anteriormente.

```
aws s3 ls s3://S3-BUCKET-NAME/files/
```

```
[ec2-user@ip-10-5-0-44 ~]$ aws s3 ls s3://labbucket123123123/files/
2023-11-06 21:18:47      30318 file1.txt
2023-11-06 21:18:47      43784 file2.txt
2023-11-06 21:18:47      96675 file3.txt
[ec2-user@ip-10-5-0-44 ~]$
```

Paso 4: Para eliminar uno de los archivos en el drive local, ejecutar el siguiente comando:

```
rm files/file1.txt
```

```
[ec2-user@ip-10-5-0-44 ~]$ rm files/file1.txt
[ec2-user@ip-10-5-0-44 ~]$
```

Paso 5: Para eliminar el mismo archivo en el Bucket S3, ejecutar el siguiente comando:

- Reemplazar S3-Bucket-Name por Bucket Name copiado anteriormente.

```
aws s3 sync files s3://S3-BUCKET-NAME/files/ --delete
```

```
[ec2-user@ip-10-5-0-44 ~]$ aws s3 sync files s3://labbucket123123123/files/ --delete
delete: s3://labbucket123123123/files/file1.txt
[ec2-user@ip-10-5-0-44 ~]$
```

Paso 6: Para verificar que el archivo fue eliminado del bucket S3, ejecutar el siguiente comando:

- Reemplazar S3-Bucket-Name por Bucket Name copiado anteriormente.

```
aws s3 ls s3://S3-BUCKET-NAME/files/
```

```
[ec2-user@ip-10-5-0-44 ~]$ aws s3 ls s3://labbucket123123123/files/
2023-11-06 21:18:47      43784 file2.txt
2023-11-06 21:18:47      96675 file3.txt
[ec2-user@ip-10-5-0-44 ~]$
```

Paso 7: Para ver una lista de las versiones anteriores del archivo eliminado (file1.txt), ejecutar el siguiente comando:

- Reemplazar S3-Bucket-Name por Bucket Name copiado anteriormente.

```
aws s3api list-object-versions --bucket S3-BUCKET-NAME --prefix files/file1.txt
```

```
"Versions": [
  {
    "ETag": "\"b76b2b775023e60be16bc332496f8409\"",
    "Size": 30318,
    "StorageClass": "STANDARD",
    "Key": "files/file1.txt",
    "VersionId": "1MA0ehMV.ffC84QmRe4TgtSg4Bfv8AVC",
    "IsLatest": false,
    "LastModified": "2023-11-06T21:18:47.000Z",
    "Owner": {
      "DisplayName": "awslabsc0w6482174t1697178688",
      "ID": "e49766083dcf02c2b94d2c7567b20ed8dbf21a459fe5dd1987e30e5d24076df6"
    }
  }
]
```

- VersionID → 1MA0ehMV.ffC84QmRe4TgtSg4Bfv8AVC

Paso 8: Dado que no existe un comando directo para restaurar una versión anterior de un objeto de Amazon S3 en su propio bucket, deberá volver a descargar la versión anterior y sincronizarla de nuevo con Amazon S3.

Para descargar la versión anterior de file1.txt, ejecutar el siguiente comando:

- Reemplazar S3-Bucket-Name por Bucket Name copiado anteriormente.
- Reemplazar Version-ID por VersionID copiado anteriormente.

```
aws s3api get-object --bucket S3-BUCKET-NAME --key files/file1.txt --version-id VERSION-ID files/file1.txt
```

```
[ec2-user@ip-10-5-0-44 ~]$ aws s3api get-object --bucket S3-BUCKET-NAME --key files/file1.txt --version-id VERSION-ID files/file1.txt
{
  "AcceptRanges": "bytes",
  "LastModified": "Mon, 06 Nov 2023 21:18:47 GMT",
  "ContentLength": 30318,
  "ETag": "\"b76b2b775023e60be16bc332496f8409\"",
  "VersionId": "1MA0ehMV.ffc84QmRe4TgtSg4Bfv8AVC",
  "ContentType": "text/plain",
  "ServerSideEncryption": "AES256",
  "Metadata": {}
}
```

Paso 9: Para verificar que el archivo fue restaurado localmente, ejecutar el siguiente comando:

```
ls files
```

```
[ec2-user@ip-10-5-0-44 ~]$ ls files
file1.txt  file2.txt  file3.txt
[ec2-user@ip-10-5-0-44 ~]$
```


Paso 10: Para re-sincronizar el contenido de los archivos con el Bucket S3, ejecutar el siguiente comando:

- Reemplazar S3-Bucket-Name por Bucket Name copiado anteriormente.

```
aws s3 sync files s3://S3-BUCKET-NAME/files/
```

```
[ec2-user@ip-10-5-0-44 ~]$ aws s3 sync files s3://labbucket123123123/files/  
upload: files/file1.txt to s3://labbucket123123123/files/file1.txt  
[ec2-user@ip-10-5-0-44 ~]$
```

Paso 11: Para verificar que la nueva versión de file1.txt fue ingresada en el Bucket S3, ejecutar el siguiente comando:

- Reemplazar S3-Bucket-Name por Bucket Name copiado anteriormente.

```
aws s3 ls s3://S3-BUCKET-NAME/files/
```

```
[ec2-user@ip-10-5-0-44 ~]$ aws s3 ls s3://labbucket123123123/files/  
2023-11-06 21:25:06      30318 file1.txt  
2023-11-06 21:18:47      43784 file2.txt  
2023-11-06 21:18:47      96675 file3.txt  
[ec2-user@ip-10-5-0-44 ~]$
```

Laboratorio Completado

