



Infraestructura II

## Práctica: Automatizar la creación de un recurso cloud

En esta actividad vamos a poner a prueba nuestros conocimientos. Nuestro desafío será la creación automatizada de un recurso de AWS, en este caso un bucket S3 para poder guardar nuestros archivos de forma privada y ejecutar este template en nuestra cuenta cloud. Tengamos en cuenta las siguientes consideraciones:

- Tiene que recibir el nombre del bucket por parámetro.
- El bucket debe ser privado.
- Tenemos que activar el versionado del bucket.

El código de resolución, que te mostramos en la siguiente página, puede ser levemente distinto al que escribiste e igualmente ambos ser correctos. ¡Comencemos!

## Resolución

Nuestro template está compuesto por: una referencia a la versión del template de los módulos que vamos a utilizar. La versión "2010-09-09" es la vigente en 2022. Además, una pequeña descripción de qué vamos a hacer.

AWSTemplateFormatVersion: "2010-09-09"

Description: Automatizamos Bucket S3





Vamos a ingresar el nombre del bucket por parámetro, es decir, por fuera del template al momento que lo ejecutamos.

```
Parameters:

BucketName:

Type: String

Description: El nombre de mi Bucket S3
```

En el espacio de Metadata se define la información que mostramos en la interfaz gráfica, la forma en que vamos a ejecutar nuestro template (recordemos que también se puede ejecutar con AWS CLI desde nuestras computadoras por línea de comandos).

```
Metadata:

AWS::CloudFormation::Interface:

ParametersGroup:

- Label:

default: Bucket S3

Parameters:

- BucketName

ParametersLabel:

BucketName:

default: 'Nombre de mi Bucket: '
```

Por último, pero no menos importante, el recurso a crear. El código escrito anteriormente es para darle un contexto al recurso que necesitamos, nuestro bucket S3. En este espacio definimos que queremos crear el bucket y dentro de sus propiedades:





- El nombre se recibe por parámetro (!REF + nombre de variable, definida arriba).
- Deshabilitar la visibilidad pública para que nuestro bucket sea privado.
- Habilitamos el versionado de los archivos que vamos a almacenar.

```
Resources:

S3Bucket:

Type: AWS::S3::Bucket

Properties:

BucketName: !Ref BucketName

PublicAccessBlockConfiguration:

BlockPublicAcls: true

BlockPublicPolicy: true

IgnorePublicAcls: true

RestrictPublicBuckets: true

VersioningConfiguration:

Status: Enabled
```

El código completo es el siguiente:

```
AWSTemplateFormatVersion: "2010-09-09"

Description: Automatizamos Bucket S3

Parameters:

BucketName:

Type: String

Description: El bucket de mi Bucket S3

Metadata:

AWS::CloudFormation::Interface:
```





```
default: Bucket S3
    - BucketName
 ParametersLabel:
     default: 'Nombre de mi Bucket: '
S3Bucket:
 Type: AWS::S3::Bucket
    BucketName: !Ref BucketName
     BlockPublicAcls: true
      IgnorePublicAcls: true
      RestrictPublicBuckets: true
      Status: Enabled
```

Ahora veamos cómo ejecutarlo. En CloudFormation seleccionamos la opción "Crear pila".







Una vez dentro, tenemos que elegir la opción de "Cargar un archivo de una plantilla" y lo seleccionamos de nuestra computadora. En nuestro caso, se llama "s3\_dh.yaml".



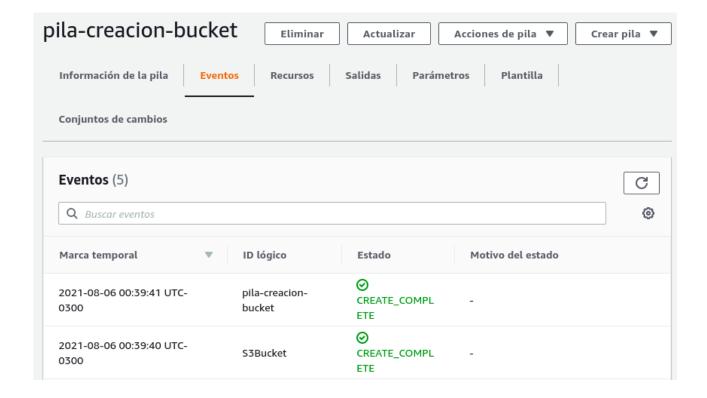
En el paso anterior, se realizaron validaciones de que el template sea correcto. Una vez superadas las validaciones, vamos a poder visualizar la información de nuestro código. Escribir el nombre que queremos para nuestro bucket, en la sección "Parámetros".







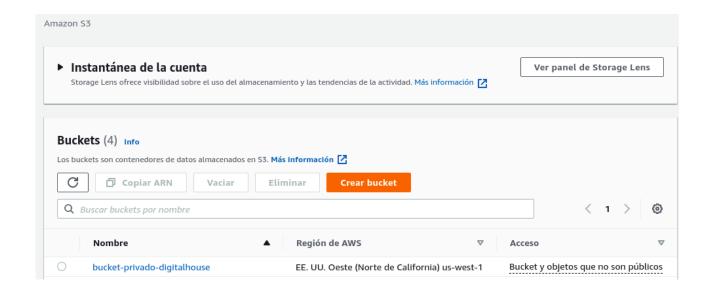
¡Nuestro bucket ya se está creando! Podemos ver en la sección de eventos cómo se crea.



Chequemos el servicio S3, ¿estará ahí?







¡La creación fue un éxito! Ahora que nuestro desafío está completo, lo queremos eliminar para no generar costos extras en nuestra cuenta de AWS:



Tengamos en cuenta que al eliminar una pila también se eliminan todos los recursos.





