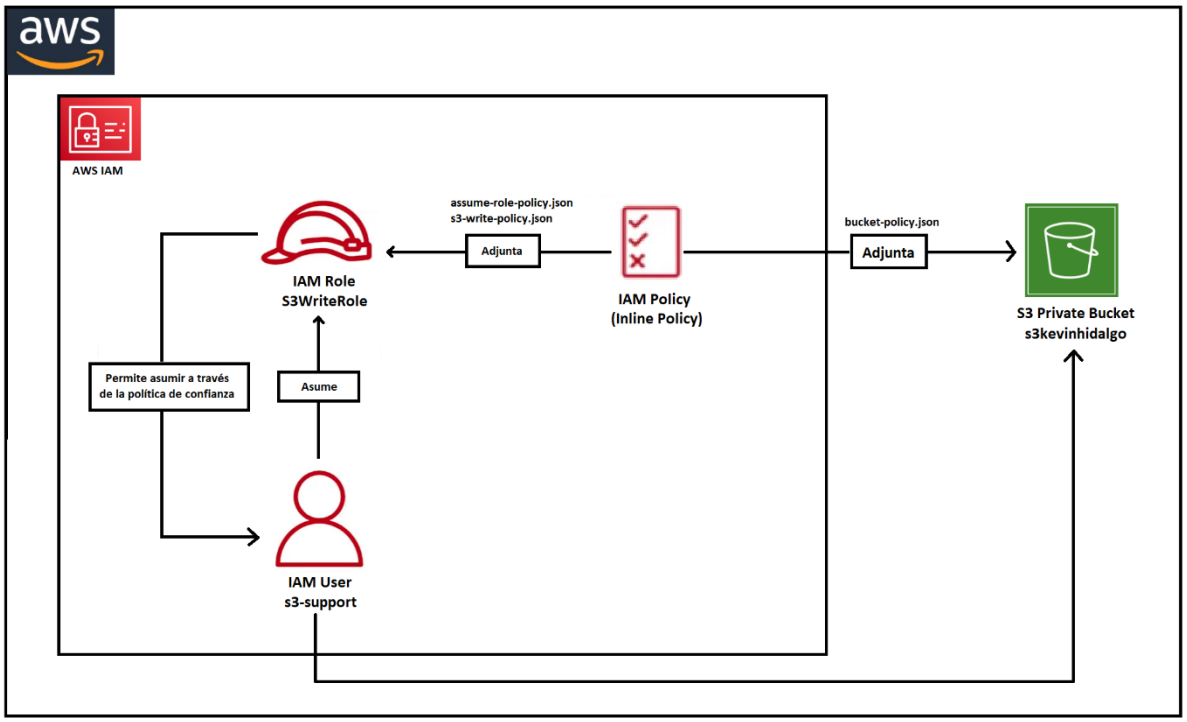
DESAFIO 3

Kevin Damian Hidalgo – 99405

1. Introducción

Este documento detalla el proceso para configurar un rol de IAM que permita a un usuario asumir el rol creado y escribir archivos en un bucket cerrado de Amazon S3 desde la CLI de AWS.

Diagrama conceptual



En este sistema, los componentes IAM User, IAM Role, y IAM Policy trabajan juntos para gestionar el acceso a un bucket S3 de forma segura:

1. **IAM Role (S3WriteRole)** tiene dos políticas inline adjuntas:
    - **assume-role-policy.json**: Es una política de confianza que define quién puede asumir el rol. En este caso, permite que el usuario IAM **s3-support** asuma el rol **S3WriteRole**.
    - **s3-write-policy.json**: Esta política otorga permisos específicos al rol para realizar operaciones de escritura en el bucket S3 llamado **s3kevinhidalgo**.
2. **IAM User (s3-support)** asume el rol **S3WriteRole** utilizando la política de confianza definida en **assume-role-policy.json**. Esta política permite al usuario obtener credenciales temporales para asumir el rol.
3. Una vez que el usuario **s3-support** asume el rol **S3WriteRole**, obtiene los permisos definidos en **s3-write-policy.json**, lo que le permite escribir en el bucket privado de S3 (**s3kevinhidalgo**).
4. **El bucket S3 (s3kevinhidalgo)** tiene una política inline adjunta (**bucket-policy.json**) que controla quién puede realizar operaciones sobre el bucket. Esta política puede restringir el acceso, permitiendo solo a roles y usuarios específicos interactuar con el bucket.

En resumen, el rol **S3WriteRole** y sus políticas inline (**assume-role-policy.json** y **s3-write-policy.json**) permiten que el usuario **s3-support** obtenga permisos temporales para escribir en el bucket **s3kevinhidalgo**, y la política del bucket (**bucket-policy.json**) aplica restricciones adicionales para garantizar el acceso seguro.

2. Crear un bucket en S3

Primero, debes crear un bucket en S3 con un nombre único, ya que el nombre de los buckets debe ser globalmente único en AWS.

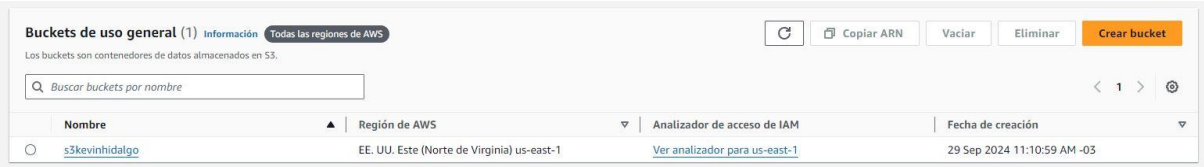
Comando para crear un bucket S3:

aws s3api create-bucket --bucket nombre-unico-del-bucket --region us-east-1

Reemplaza **nombre-unico-del-bucket** por un nombre único que no esté siendo utilizado por otros usuarios de AWS.

En mi caso el nombre del bucket seria **s3kevinhidalgo**.

```
[cloudshell-user@ip-10-130-79-136 ~]$ aws s3api create-bucket --bucket s3kevinhidalgo --region us-east-1
{
    "Location": "/s3kevinhidalgo"
}
[cloudshell-user@ip-10-130-79-136 ~]$
```



2.1. Deshabilitar el Acceso Público

S3 ofrece varias configuraciones que pueden ayudar a restringir el acceso público. Deshabilitar las configuraciones de acceso público es crucial para asegurarte de que el bucket esté "cerrado".

Usa el siguiente comando para deshabilitar el acceso público:

aws s3api put-bucket-public-access-block --bucket nombre-unico-del-bucket --public-access-block-configuration BlockPublicAcls=true,IgnorePublicAcls=true,BlockPublicPolicy=true,RestrictPublicBuckets=true

Este comando establece las siguientes restricciones:

- **BlockPublicAcls**: Bloquea ACLs públicas.
- **IgnorePublicAcls**: Ignora todas las ACLs públicas que intenten aplicarse al bucket.
- **BlockPublicPolicy**: Bloquea políticas públicas en el bucket.
- **RestrictPublicBuckets**: Restringe el acceso si hay configuraciones públicas en el bucket.

```
[cloudshell-user@ip-10-130-79-136 ~]$ aws s3api put-public-access-block --bucket s3kevinhidalgo --public-access-block-configuration BlockPublicAcls=true,IgnorePublicAcls=true,BlockPublicPolicy=true,RestrictPublicBuckets=true
[cloudshell-user@ip-10-130-79-136 ~]$
```

Este comando no retorna nada en consola, pero si se accede a la pestaña **Permisos** del bucket en cuestión se podrá visualizar la sección **Bloquear acceso público (configuración del bucket)** en donde estará activado, es decir, se convirtió en un bucket cerrado.

2.2. Configurar Política de Bucket para Permitir Acceso Solo a Usuarios o Roles Específicos

Es recomendable aplicar una política de bucket para permitir el acceso solo a los usuarios, roles o servicios específicos dentro de tu cuenta.

Crea un archivo **bucket-policy.json** con el siguiente contenido:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Deny",
      "Principal": "*",
      "Action": "s3:*",
      "Resource": [
        "arn:aws:s3:::nombre-unico-del-bucket",
        "arn:aws:s3:::nombre-unico-del-bucket/*"
      ],
      "Condition": {
        "Bool": {
          "aws:SecureTransport": "false"
        }
      }
    },
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::123456789012:user/s3-support"
      },
      "Action": [
        "s3:PutObject",
```

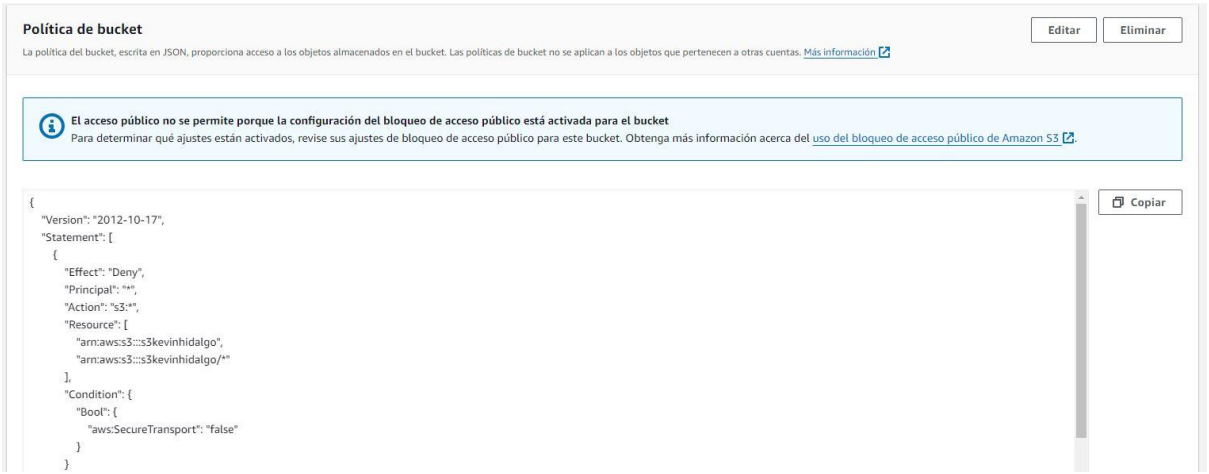
```
            "s3:GetObject"
        ],
        "Resource": [
            "arn:aws:s3:::nombre-unico-del-bucket/*"
        ]
    }
  ]
}
```

Esta política:

- Deniega todo el acceso si las solicitudes no están realizadas a través de HTTPS (**aws:SecureTransport**).
- Permite solo al usuario o rol IAM especificado (reemplaza el **arn:aws:iam::123456789012:user/s3-support** con el ARN del usuario o rol adecuado) realizar acciones de escritura.
- Aplica la política de bucket con el siguiente comando:

aws s3api put-bucket-policy --bucket nombre-unico-del-bucket --policy file://bucket-policy.json

Reemplazar el valor de ***nombre-unico-del-bucket*** por el nombre de tu bucket, y la ruta del archivo json a configurar en el comando dependerá de en donde se encuentren.



3. Crear un usuario IAM llamado s3-support y generar credenciales programáticas

Para este ejercicio se debe crear un usuario con el nombre s3-support el cual se utilizara para subir archivos al bucket.

Para crear el usuario en cuestión se debe realizar el siguiente comando:

aws iam create-user --user-name s3-support

```
[cloudshell-user@ip-10-130-79-136 ~]$ aws iam create-user --user-name s3-support
{
    "User": {
        "Path": "/",
        "UserName": "s3-support",
        "UserId": "AIDAYM7POCWYBR7BFT6MV",
        "Arn": "arn:aws:iam::577638372784:user/s3-support",
        "CreateDate": "2024-09-29T14:13:08+00:00"
    }
}
[cloudshell-user@ip-10-130-79-136 ~]$
```

Comando para crear las credenciales programáticas:

aws iam create-access-key --user-name s3-support

Este comando generará el **AccessKeyId** y el **SecretAccessKey**, que debes resguardar para configurar el acceso con dicho usuario.

```
[cloudshell-user@ip-10-130-79-136 ~]$ aws iam create-access-key --user-name s3-support
{
    "AccessKey": {
        "UserName": "s3-support",
        "AccessKeyId": "AKIAYM7POCWYA662PEYA",
        "Status": "Active",
        "SecretAccessKey": "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",
        "CreateDate": "2024-09-29T14:45:06+00:00"
    }
}
[cloudshell-user@ip-10-130-79-136 ~]$
```

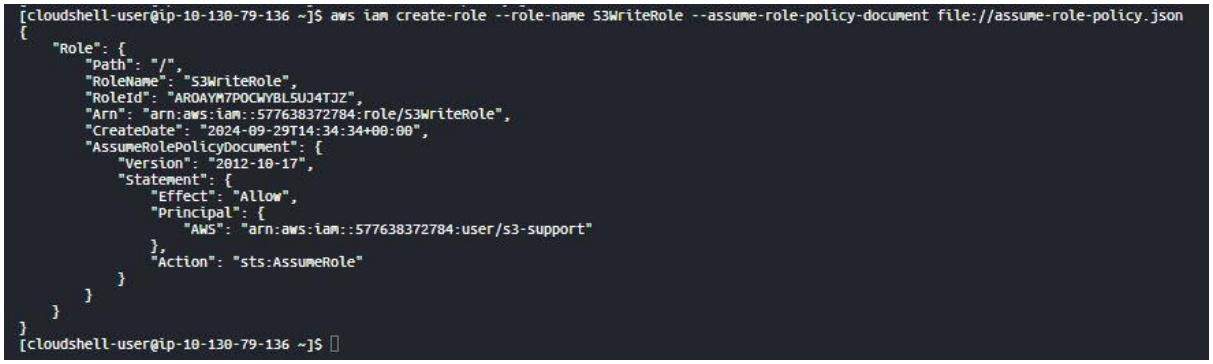
4. Crear un rol con una política que permita escribir en el bucket

4.1. Crear el rol de IAM

Para que el usuario s3-support pueda tener permisos sobre el bucket de S3, primero se debe crear el rol de IAM que será asumido por el usuario.

Comando para crear el rol:

aws iam create-role --role-name S3WriteRole --assume-role-policy-document file://assume-role-policy.json



IMPORTANTE: El comando **create-role** exige especificar el archivo **–assume-role-policy-document** el cual debe estar creado previamente, donde **assume-role-policy.json** es un archivo que contiene la política que permite asumir el rol:

```
{
 "Version": "2012-10-17",
 "Statement": {
  "Effect": "Allow",
  "Principal": {
   "AWS": "arn:aws:iam::123456789012:user/s3-support"
  },
  "Action": "sts:AssumeRole"
 }
}
```

Asegúrate de reemplazar el ID de cuenta 123456789012 con el ID de tu cuenta de AWS. (creado en el paso 3)

4.2. Crear la política de permisos para escribir en el bucket

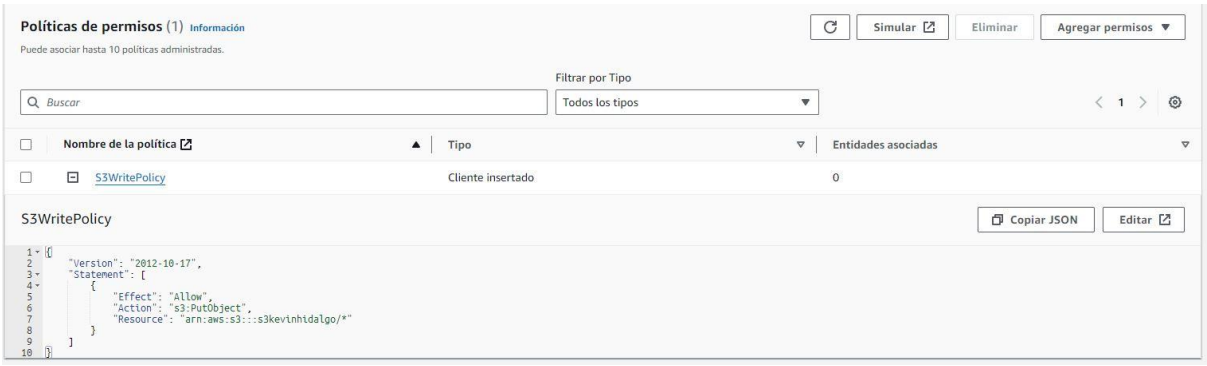
Luego, debes crear el archivo **s3-write-policy.json** el cual es una política que le permitira escribir en el bucket creado:

```
{
 "Version": "2012-10-17",
 "Statement": [
  {
   "Effect": "Allow",
   "Action": "s3:PutObject",
   "Resource": "arn:aws:s3:::nombre-unico-del-bucket/*"
  }
 ]
}
```

Debe reemplazar el valor **nombre-unico-del-bucket** por el nombre del bucket creado (Paso 1).

Una vez creado se debe adjuntar la política al rol:
aws iam put-role-policy --role-name S3WriteRole --policy-name S3WritePolicy --policy-document file://s3-write-policy.json

```
[cloudshell-user@ip-10-130-79-136 ~]$ aws iam put-role-policy --role-name S3WriteRole --policy-name S3WritePolicy --policy-document file://s3-write-policy.json
[cloudshell-user@ip-10-130-79-136 ~]$
```



Se le agrega al Role S3WriteRole una política con el nombre S3WritePolicy en la cual se especifica que solo se podrán escribir archivos en el bucket.

5. Conectar el CLI con las credenciales del usuario s3-support

Una vez creado el Bucket, Usuario, Rol y las respectivas Políticas, se procede a usar las credenciales generadas anteriormente (Punto 3) para configurar el CLI con el usuario s3-support.

Comando para configurar el perfil:

aws configure --profile s3-support

Durante la configuración, introduce el **AWS_ACCESS_KEY_ID** (**AccessKeyId**), el **AWS_SECRET_ACCESS_KEY** (**SecretAccessKey**) del usuario, la región en donde está ubicado el bucket (en este caso us-east-1) y el formato de salida (depende de la necesidad del usuario).

```
[cloudshell-user@ip-10-130-79-136 ~]$ aws configure --profile s3-support
AWS Access Key ID [None]: AKIAYM7POCWYA662PEYA
AWS Secret Access Key [None]: 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
Default region name [None]: us-east-1
Default output format [None]: json
```

6. Asumir el rol

Una vez que hayas configurado el perfil, puedes asumir el rol utilizando el siguiente comando:

aws sts assume-role --role-arn arn:aws:iam::123456789012:role/S3WriteRole --role-session-name S3WriteSession --profile s3-support

Se debe reemplazar el ID **123456789012** del ARN del Rol **S3WriteRole** con el ID correspondiente.

Este comando devolverá las credenciales temporales (**AccessKeyId**, **SecretAccessKey**, y **SessionToken**) que debes usar para interactuar con los recursos.

Guarda las credenciales temporales y configura temporalmente el CLI con ellas:

aws configure set aws_access_key_id TEMP_ACCESS_KEY --profile s3-support

aws configure set aws_secret_access_key TEMP_SECRET_KEY --profile s3-support

aws configure set aws_session_token TEMP_SESSION_TOKEN --profile s3-support

Si el cambio esta automatizado, en las variables TEMP deberían guardar los datos correspondientes obtenidos del comando anterior, y si el cambio se sesión es realizado manualmente, se deben reemplazar las variables TEMP por los datos correspondientes.

7. Escribir en el Bucket

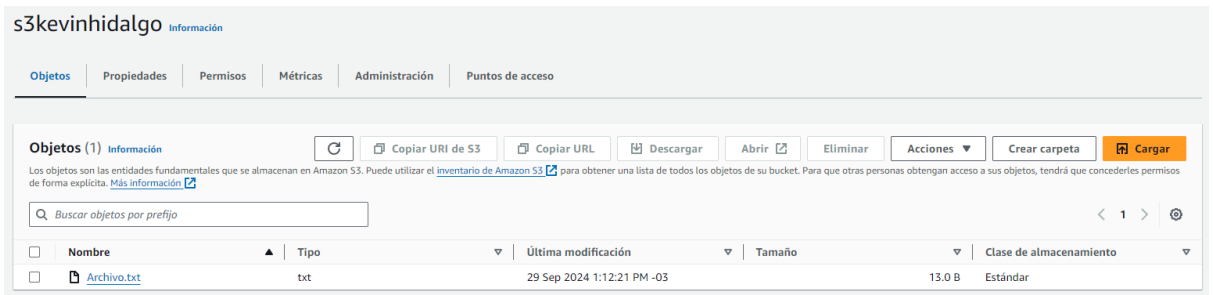
Finalmente, intenta escribir un archivo en el bucket utilizando el perfil temporal:

Comando para subir un archivo:

aws s3 cp /ruta/al/archivo.txt s3://nombre-unico-del-bucket/ --profile s3-support

Reemplazar ***/ruta/al/archivo.txt*** con la ruta/archivo que desees escribir y ***nombre-unico-del-bucket*** por el nombre del bucket correspondiente.

Si todo se ha configurado correctamente, deberías ver un mensaje de éxito que indica que el archivo se ha subido al bucket S3.



8. Realizar un Unset Role

Una vez finalizado el uso del Role para subir los archivos al bucket, se recomienda realizar un unset role para evitar cualquier acceso indebido.

Existen 3 formas de realizar un Unset el cual varía dependiendo de la forma de cómo se crearon las credenciales temporales.

8.1. Eliminar las Variables de Entorno

Si asumiste el rol utilizando las credenciales temporales y las configuraste en las variables de entorno (AWS_ACCESS_KEY_ID, AWS_SECRET_ACCESS_KEY, AWS_SESSION_TOKEN), puedes simplemente eliminarlas para "salir" del rol.

Ejecuta lo siguiente en tu terminal:

- En Linux/macOS**:**

unset AWS_ACCESS_KEY_ID AWS_SECRET_ACCESS_KEY AWS_SESSION_TOKEN

- En Windows (Command Prompt)**:**

set AWS_ACCESS_KEY_ID=

set AWS_SECRET_ACCESS_KEY=

set AWS_SESSION_TOKEN=

- En Windows (PowerShell)**:**

Remove-Item Env:AWS_ACCESS_KEY_ID, Env:AWS_SECRET_ACCESS_KEY, Env:AWS_SESSION_TOKEN

Esto hará que el AWS CLI deje de utilizar las credenciales temporales y vuelva a las predeterminadas (generalmente configuradas con aws configure).

8.2. Cambiar de Perfil de AWS

Si estás utilizando un perfil específico al asumir el rol y deseas salir del contexto de ese perfil, puedes cambiar de perfil o volver al predeterminado:

export AWS_PROFILE=default

Esto dejará de usar el perfil asociado con las credenciales del rol asumido.

8.3. Cerrar la Sesión Actual de la Terminal

Otra forma de salir del rol asumido es simplemente cerrar la sesión de terminal en la que estás trabajando. Si vuelves a abrir una nueva sesión, tus credenciales volverán a ser las predeterminadas.

Al salir de un rol asumido, tu AWS CLI volverá a usar las credenciales predeterminadas configuradas en el archivo **~/.aws/credentials.**