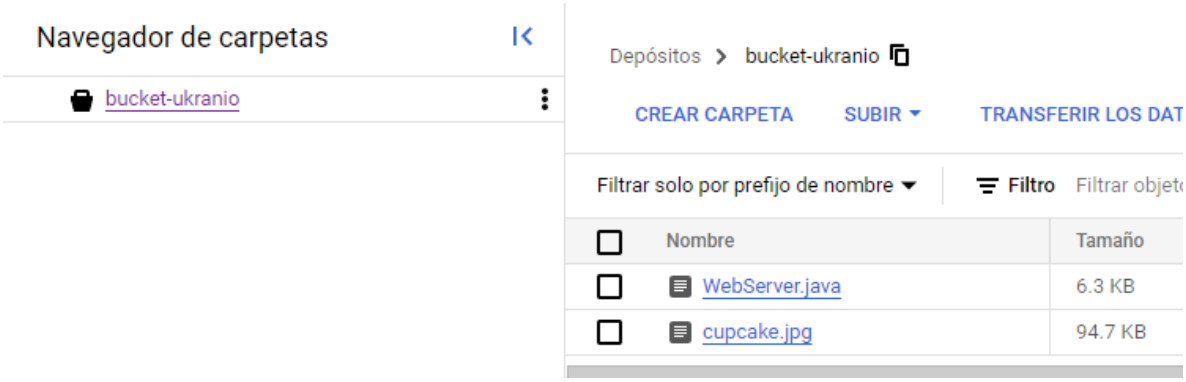
Google Cloud Storage JSON API

Elaborada por: M. en C. Ukranio Coronilla

Para el almacenamiento de datos en la nube de Google utilizaremos el servicio **Google Cloud Storage (GCS)**. Este permite almacenar datos no estructurados, es decir objetos o archivos de cualquier tipo y tamaño de hasta 5TB.

Para interactuar con este servicio usaremos la Google Cloud Storage JSON API, la cual es una API RESTful que utiliza solicitudes HTTP para acceder y gestionar los datos en GCS.

Vamos a subir un par de archivos al bucket para usarlos de ejemplo dando click a SUBIR , con lo que en mi caso me ha quedado así:



Ahora proceda a habilitar el servicio de Google Cloud Storage JSON API:

Dado que esta API implica acceso a datos y no sólo es un servicio de procesamiento como la Traslation API, requiere un sistema mas complejo de credenciales y permisos de acceso a usuarios.

Identity and Access Management

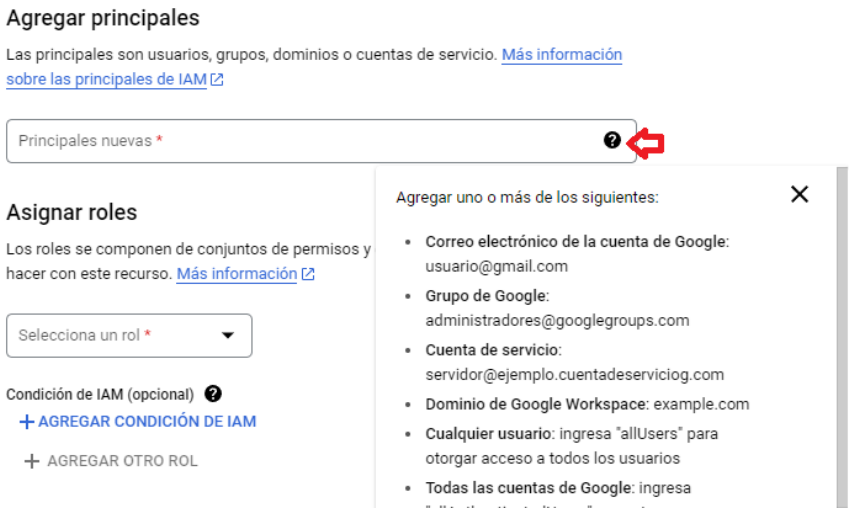
Google Cloud dispone de un sistema para controlar quién puede acceder a los objetos en un bucket y qué acciones pueden realizarse sobre dichos objetos. Este sistema se denomina Identity and Access Management (IAM).

Para mostrar como funciona veremos como otorgar permisos de lectura al público en general para el bucket que acabamos de crear, aunque lo podríamos hacer también sólo para un objeto dentro del bucket.

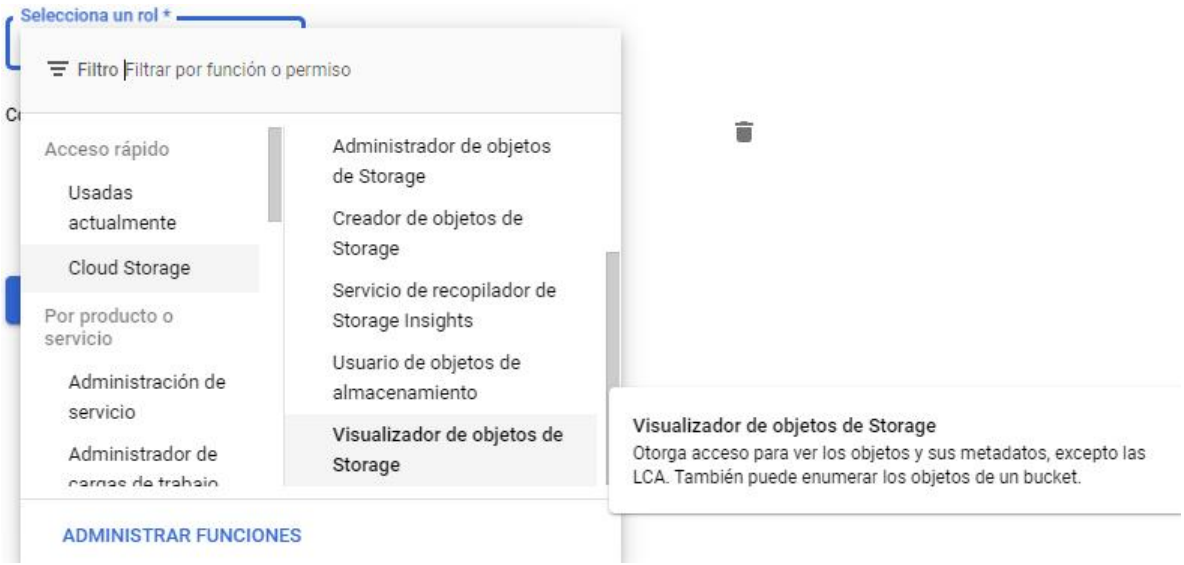
Primero le damos click a la hamburguesa ☰ y seleccionamos: **Cloud Storage->Buckets** y posteriormente le damos click al bucket de interés. Damos click a la pestaña Permisos y para poder hacer el bucket público damos click en: Quitar la Prevención de acceso público y después en CONFIRMAR.

Ahora vamos a otorgar el acceso a un **PRINCIPAL**, lo cual hace referencia a una cuenta, un grupo en Google o a una cuenta de servicio, lo cual es similar a un propietario en un sistema UNIX. Para ello damos click en:

Y en **Agregar principales** podemos poner el mouse sobre el signo de interrogación para que nos muestre todas las opciones disponibles:

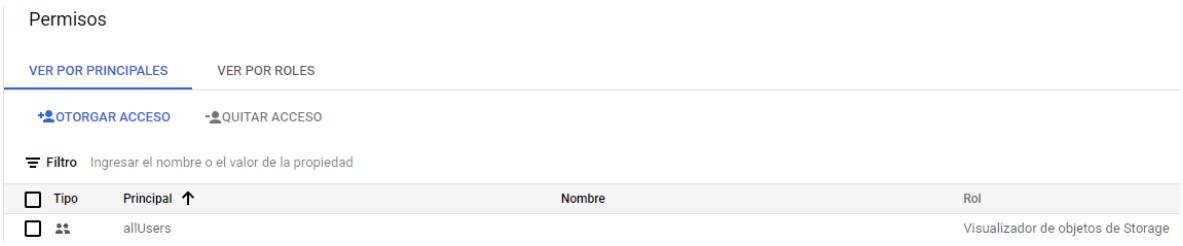


En nuestro caso dado que queremos dar acceso a cualquier usuario ingresamos **allUsers** como PRINCIPAL. Posteriormente tenemos que asignarle un rol o lo que puede hacer el PRINCIPAL con este recurso y dado que solo le queremos dar permisos de lectura seleccionamos **Visualizador de objetos de Storage** como se muestra en la imagen:



Finalmente damos click en 

. Podemos ver que se ha creado este nuevo permiso:



Si seleccionamos la casilla de verificación del permiso en la parte izquierda podremos quitar el permiso dando click en 

si así lo deseamos.

Ahora para probar el acceso público y de acuerdo con la referencia disponible de la API:

https://cloud.google.com/storage/docs/json_api

Podemos poner en el navegador la URI:

https://storage.googleapis.com/storage/v1/b/bucket-ukranio/o/cupcake.jpg

donde tendría que substituir el nombre de su bucket en amarillo y el nombre de su archivo en verde. Con lo cual le devolvería un JSON con los metadatos del archivo.

Para descargar el archivo agregamos el parámetro alt=media, quedando la URI para este caso como:

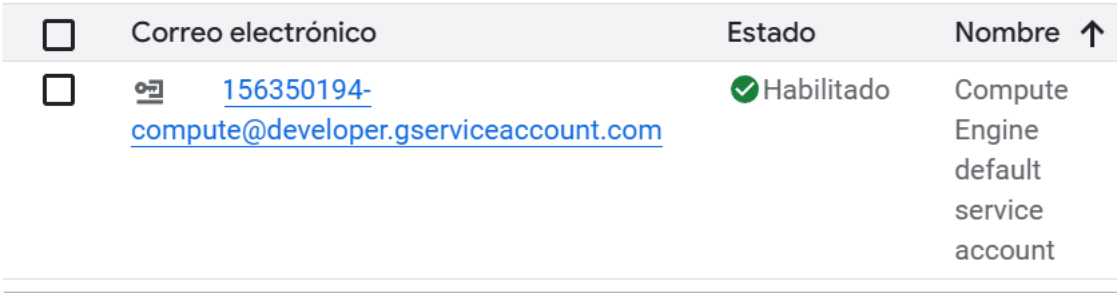
https://storage.googleapis.com/storage/v1/b/bucket-ukranio/o/cupcake.jpg?alt=media

Para reutilizar el bucket en la siguiente parte de la práctica, elimine el acceso público al bucket. Con esto no tendríamos que poder acceder al recurso en el bucket (pruébelo).

Al programar aplicaciones en la nube en general sólo queremos que nuestro código pueda acceder al bucket y nadie más pueda hacerlo. Para lograrlo primero necesitamos obtener una cuenta de servicio que es básicamente una cuenta impersonal vinculada a su sesión en la plataforma de GCP.

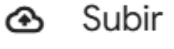
Cuenta de servicio

Para obtener la cuenta de servicio le damos click a la hamburguesa ☰ y seleccionamos: **IAM y Administración->Cuentas de servicio** y aquí veremos una lista de las cuentas de servicio existentes, pero en nuestro caso sólo existirá la cuenta de servicio por default:



Le damos click a la cuenta de servicio lo cual nos muestra el correo electrónico asociado (copiarlo porque se utilizará después) y el ID. Le damos click a la pestaña Claves, posteriormente damos click en Agregar clave y seleccionamos **Crear clave nueva**. En nuestro caso creamos la clave privada en JSON y nos descarga el archivo de credenciales JSON.

Ahora como al inicio de la práctica vaya a su bucket y de click en Otorgar acceso para agregar como PRINCIPAL la cuenta de servicio y asigne el rol de **Lector de objetos heredados de almacenamiento**.

Lo que sigue es activar la cuenta de servicio para lo cual accedemos al Cloud Shell  y subimos el archivo de credenciales JSON dando click en ⋮ y después en  . Posteriormente activamos la cuenta de servicio ejecutando el comando (substituya en amarillo el nombre del archivo que subió):

```
gcloud auth activate-service-account --key-file=nombre_del_archivo.json
```

Hecha la activación podemos obtener un token de acceso con el siguiente comando:

```
gcloud auth print-access-token
```

Con lo cual nos devuelve un token de acceso que podemos usar para acceder al bucket. Copie dicho token y en la terminal UNIX de su LAP podrá acceder al recurso mediante curl como sigue:

```
curl -X GET -H "Authorization: Bearer token"
"https://storage.googleapis.com/storage/v1/b/bucket-ukranio/o/cupcake.jpg"
```

En azul va el token de acceso, en amarillo el nombre de su bucket y en verde el nombre del objeto dentro de su bucket (no se observa, pero existe un espacio en blanco entre las dos expresiones encerradas en comillas).

Cabe mencionar que el token de acceso tiene un periodo de vigencia de una hora a partir del momento en que se emite. Es posible especificar un tiempo mayor con el parámetro lifetime hasta un máximo de 12 horas con:

```
gcloud auth print-access-token --lifetime=43200s
```

Ejercicio

Modifique el cliente HTTP síncrono para acceder al recurso de la misma manera que lo hicimos con curl. Envíe el código y la captura de pantalla con la respuesta como comprobantes de esta práctica.