Tot plugin GCP IAM

Control de versiones **2**
Modelo de integración **3**
Gobierno activo 3
Credenciales requeridas **3**
Creación de la cuenta de servicio 3
Gobierno activo 5
Ejemplo de configuración **6**

Control de versiones

Versión	Fecha de modificación	Responsable	Aprobador	Resumen de cambios
1.0	04/03/2024	Anjana Producto	Anjana Producto	Creación del documento

Modelo de integración

Gobierno activo

De forma general los DSA de Anjana Data se representan como grupos y los firmantes de dichos DSA son miembros de dichos grupos.

Anjana Data crea y elimina los grupos de forma automática, al igual que incluye y excluye a usuarios de cada grupo con el objetivo de materializar la adhesión o desadherencia de un usuario a un DSA.

Para gestionar los permisos y roles, el plugin se conecta a una instancia de Google IAM[1], la cual da acceso a una API que proporciona Google en su plataforma Cloud que gestiona el control de acceso e identidades sobre los recursos de la propia plataforma.

El contrato está representado por un rol custom con los permisos genéricos de acceso a la tecnología preconcedidos, los usuarios se asocian a dicho rol (adquieren dichos permisos) usando políticas en las cuales se termina de especificar el grano fino a nivel elemento en las tecnologías que lo permitan.

Las acciones que se aplican sobre GCP son la siguientes:

- Creación/modificación/eliminación de roles custom.
- Asignación de roles a usuarios mediante políticas IAM con condiciones de aplicabilidad (para gestionar acceso a nivel elemento). En el plugin GCP IAM se recupera el nombre de los roles y es en otros plugins como el de GCP BigQuery donde se asignan/eliminan dichos roles para los usuarios.

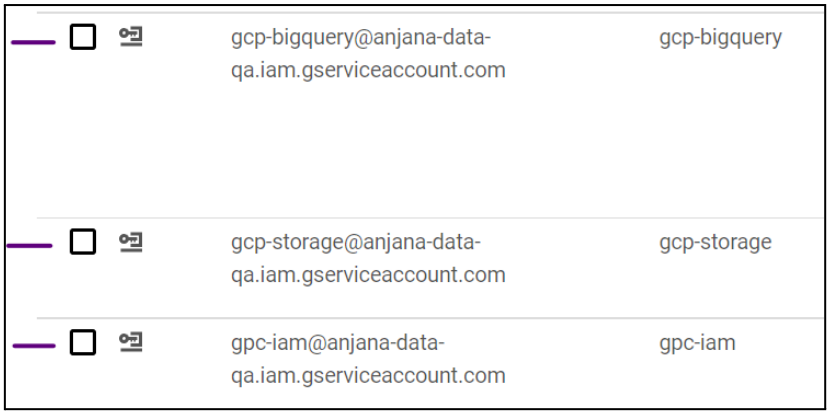
Credenciales requeridas

Las credenciales requeridas se deben configurar en el fichero yaml en la propiedad “totplugin.connection.credentialsContent”.

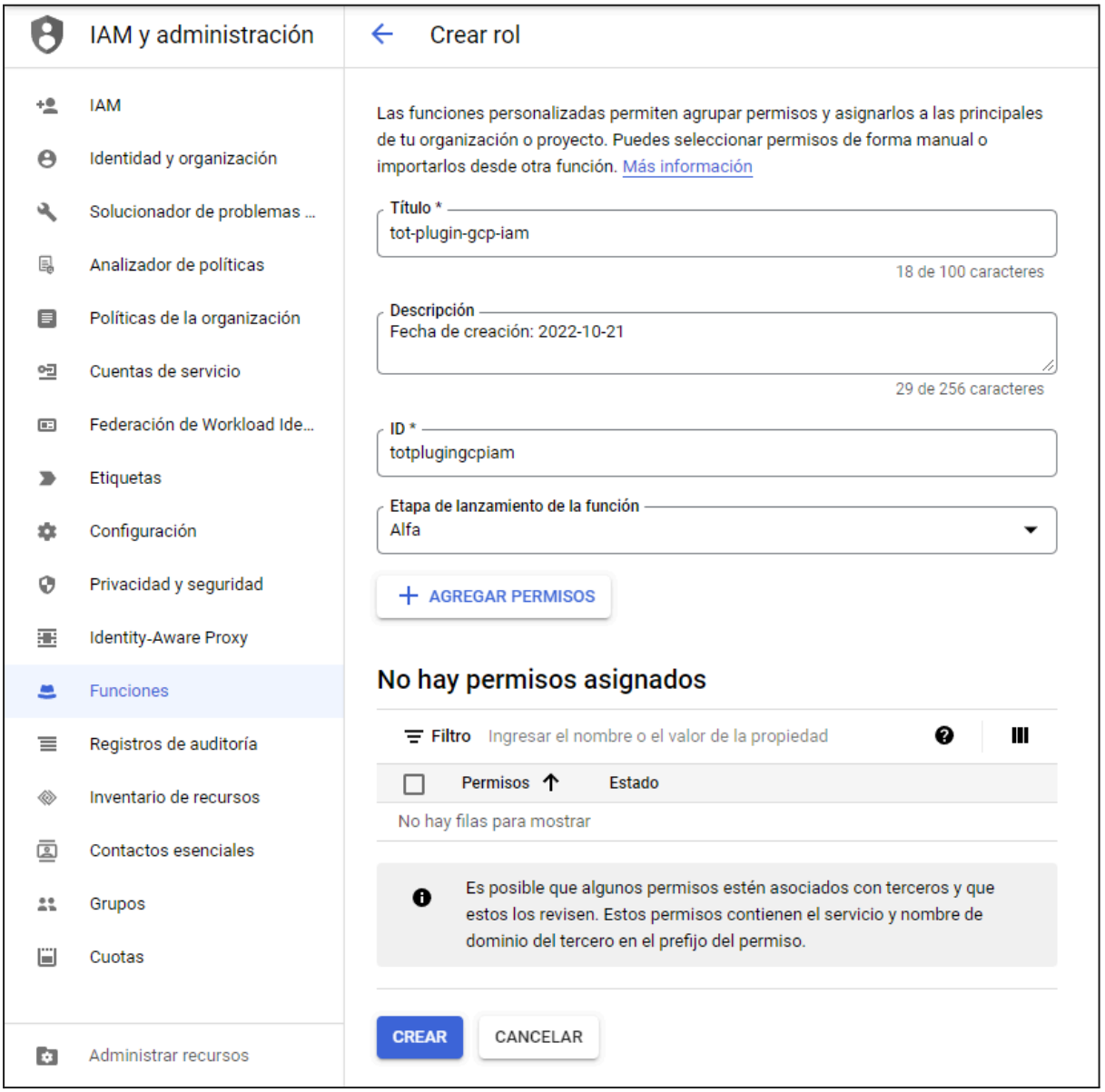
Creación de la cuenta de servicio

Para GCP es necesario crear una cuenta de servicio en IAM para cada plugin de forma individual y, tras eso, asignarle los permisos necesarios para la ejecución de las tareas específicas de cada plugin.

[1] Documentación Google IAM: https://cloud.google.com/iam/docs



Para personalizar los permisos de forma adecuada es necesaria la creación de roles personalizados en los cuáles se engloban los permisos que luego son asociados a las cuentas de servicio.



Gobierno activo

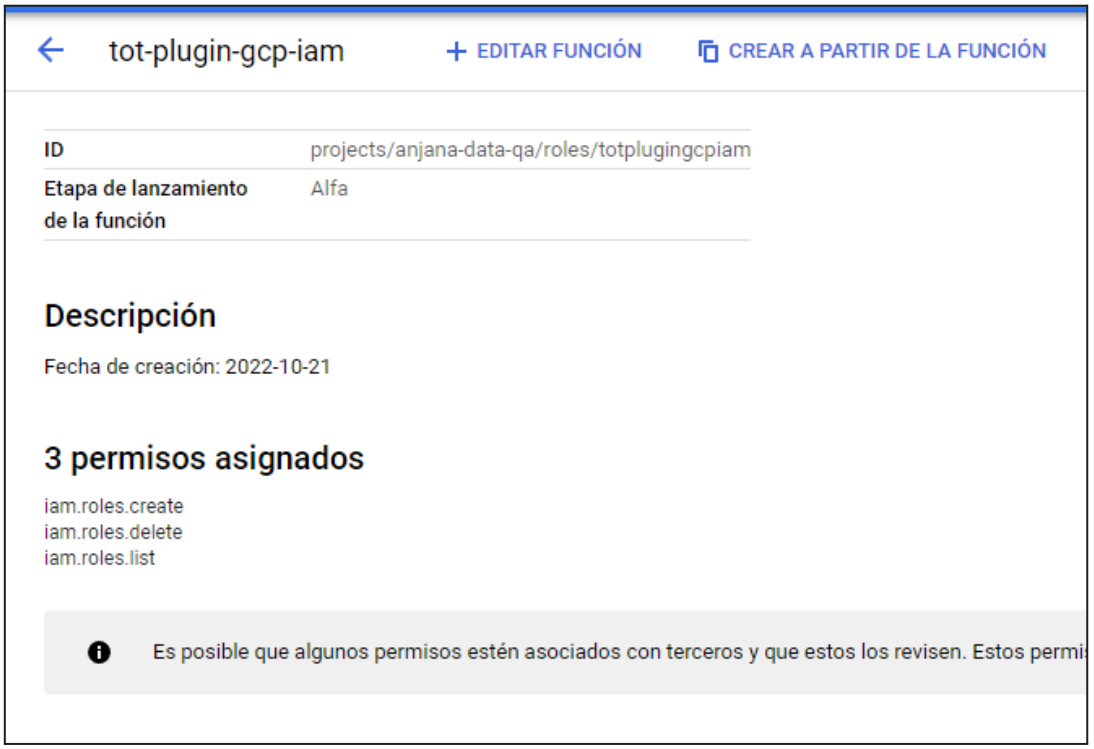
Los permisos utilizados son los siguientes:

- iam.roles.create
- iam.roles.delete
- iam.roles.list

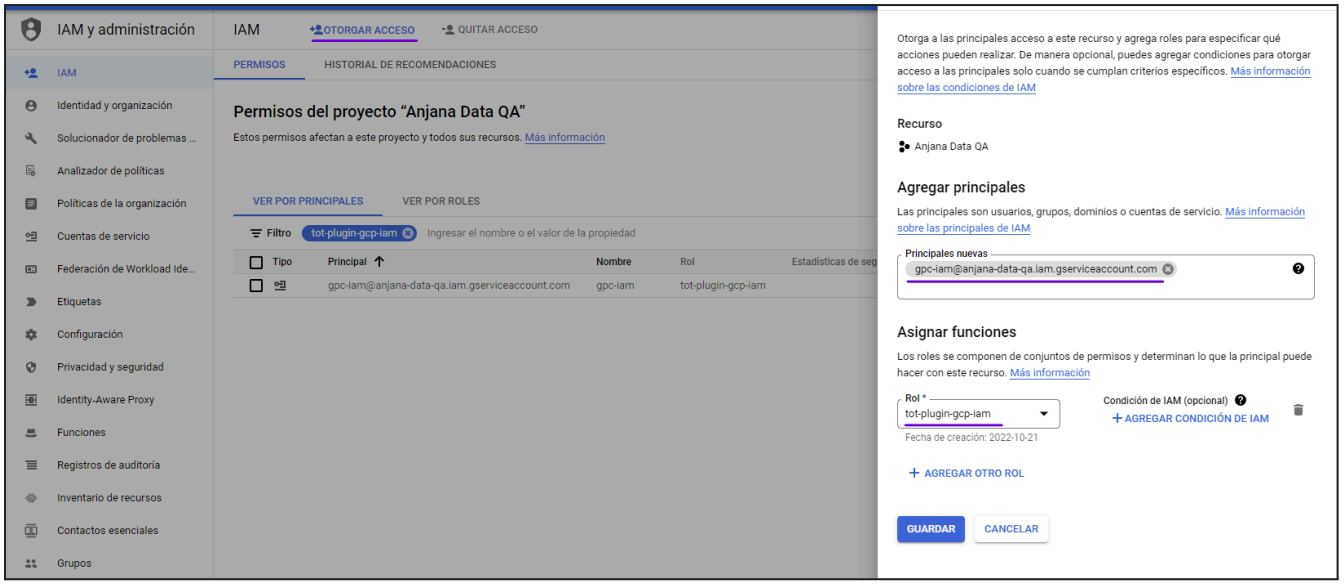
Apis requeridas en proyecto:

- Identity and Access Management (IAM) API
- Admin API SDK

En resumen los permisos utilizados para el rol personalizado serán los siguientes:



Para asignar los permisos a la cuenta de servicio de BigQuery es necesario asignar el rol con los permisos al usuario:



Ejemplo de configuración

Aquí se incluye el detalle de la configuración específica del plugin.
En la Guía de Configuración técnica se explica la configuración común.

Configuraciones específicas:

- connection:
  - credentialsContent: Credenciales de acceso a GCP.
  - project: Nombre del proyecto

```
server:
  port: 15010

totplugin:
  server:
    urls:
      - http://totserver:15000/tot/

  aris:
    - ari: "anja:totplugin:im:/Google/gcpIam/devQA/"
  groupPrefix: Dsa_
  connection:
    project: "projects/anjana-data-qa"
    credentialsContent: |
      {
        "type": "service_account",
        "project_id": "anjana-data",
        "private_key_id": "**********************",
```

```
        "private_key": "-----BEGIN PRIVATE KEY-----\n-----END PRIVATE
KEY-----\n",
        "client_email": "gpc-iam@*****.com",
        "client_id": "**********************",
        "auth_uri": "https://accounts.google.com/o/oauth2/auth",
        "token_uri": "https://oauth2.googleapis.com/token",
                                          "auth_provider_x509_cert_url":
"https://www.googleapis.com/oauth2/v1/certs",
                                                "client_x509_cert_url":
"https://www.googleapis.com/robot/v1/metadata/x509/gpc-iam%40*****.co
m"
      }

eureka:
  client:
    serviceUrl:
      defaultZone: http://totserver:15000/tot/eureka
```