Creating and managing service accounts

cloud.google.com/iam/docs/creating-managing-service-accounts

This page explains how to create and manage service accounts using the Identity and Access Management (IAM) API, the Google Cloud Console, and the `gcloud` command-line tool.

By default, each project can have up to 100 service accounts that control access to your resources. You can request a quota increase if necessary. Learn more about quotas and limits.

Before you begin

- Understand IAM service accounts
- Install the `gcloud` command-line tool

Required permissions

To allow a user to manage service accounts, grant one of the following roles:

- Service Account User (`roles/iam.serviceAccountUser`): Includes permissions to list service accounts, get details about a service account, and impersonate a service account.
- Service Account Admin (`roles/iam.serviceAccountAdmin`): Includes permissions to list service accounts and get details about a service account. Also includes permissions to create, update, and delete service accounts, and to view or change the IAM policy on a service account.

To learn more about these roles, see Service Accounts roles.

IAM basic roles also contain permissions to manage service accounts. You should not grant basic roles in a production environment, but you can grant them in a development or test environment.

Creating a service account

Creating a service account is similar to adding a member to your project, but the service account belongs to your applications rather than an individual end user.

When you create a service account, you must provide an alphanumeric ID (*`SA_NAME`* in the samples below), such as `my-service-account` . The ID must be between 6 and 30 characters, and can contain lowercase alphanumeric characters and dashes. After you create a service account, you cannot change its name.

The service account's name is a unique identifier; it will appear in the service account's email address that is provisioned during creation, such as `SA_NAME@PROJECT_ID.iam.gserviceaccount.com`.

Each service account also has a unique numeric ID, which is generated automatically.

You also provide the following information when you create a service account:

- `SA_DESCRIPTION` is an optional description for the service account.
- `SA_DISPLAY_NAME` is a friendly name for the service account.
- `PROJECT_ID` is the ID of your Google Cloud project.

To create a service account, at minimum the user must be granted the Service Account Admin role (`roles/iam.serviceAccountAdmin`) or the Editor basic role (`roles/editor`). You should not grant basic roles in a production environment, but you can grant them in a development or test environment.

After you create a service account, you might need to wait for 60 seconds or more before you use the service account. If you try to use a service account immediately after you create it, and you receive an error, you can retry the request with exponential backoff.

1. To create the service account, run the `gcloud iam service-accounts create` command:

   ```
   gcloud iam service-accounts create SERVICE_ACCOUNT_ID \
       --description="DESCRIPTION" \
       --display-name="DISPLAY_NAME"
   ```

   Replace the following values:

   - `SERVICE_ACCOUNT_ID` : The ID for the service account.
   - `DESCRIPTION` : Optional. A description of the service account.
   - `DISPLAY_NAME` : A service account name to display in the Cloud Console.

2. Optional: To grant your service account an IAM role on your project, run the `gcloud projects add-iam-policy-binding` command:

   ```
   gcloud projects add-iam-policy-binding PROJECT_ID \
       --
   member="serviceAccount:SERVICE_ACCOUNT_ID@PROJECT_ID.iam.gserviceaccount.com"
   \
       --role="ROLE_NAME"
   ```

   Replace the following values:

   - `PROJECT_ID` : The project id.
   - `SERVICE_ACCOUNT_ID` : The service account ID.
   - `ROLE_NAME` : A role name, such as `roles/compute.osLogin` .

3. Optional: To allow users to impersonate the service account, run the `gcloud iam service-accounts add-iam-policy-binding` command to grant a user the Service Account User role (`roles/iam.serviceAccountUser`) on the service account:

```
gcloud iam service-accounts add-iam-policy-binding \
    SERVICE_ACCOUNT_ID@PROJECT_ID.iam.gserviceaccount.com \
    --member="user:USER_EMAIL" \
    --role="roles/iam.serviceAccountUser"
```

Replace the following values:

- `PROJECT_ID` : The project ID.
- `SERVICE_ACCOUNT_ID` : The service account ID.
- `USER_EMAIL` : The email address for the user.

After you create a service account, grant one or more roles to the service account so that it can act on your behalf.

Also, if the service account needs to access resources in other projects, you usually must enable the APIs for those resources in the project where you created the service account.

Listing service accounts

When listing service accounts, you can specify parameters to limit the number of service accounts to include in the response. You can then use `ListServiceAccountsResponse.next_page_token` in a subsequent request to list the remaining service accounts.

Use this method to audit service accounts and keys, or to build custom tools for managing service accounts.

To list service accounts, at minimum the user must be granted the Service Account User role (`roles/iam.serviceAccountUser`) or the Viewer basic role (`roles/viewer`).

Execute the `gcloud iam service-accounts list` command to list all service accounts in a project.

Command:

```
gcloud iam service-accounts list
```

The output is the list of all service accounts in the project:

```
NAME                    EMAIL
SA_DISPLAY_NAME_1       SA_NAME_1@PROJECT_ID.iam.gserviceaccount.com
SA_DISPLAY_NAME_2       SA_NAME_2@PROJECT_ID.iam.gserviceaccount.com
```

Updating a service account

The display name (friendly name) and description of a service account are commonly used to capture additional information about the service account, such as the purpose of the service account or a contact person for the account.

To update the name or description of a service account, at minimum the user must be granted the Service Account Admin role (`roles/iam.serviceAccountAdmin`) or the Editor basic role (`roles/editor`).

Execute the `gcloud iam service-accounts update` command to update a service account.

Command:

```
gcloud iam service-accounts update \
    SA_NAME@PROJECT_ID.iam.gserviceaccount.com \
    --description="UPDATED_SA_DESCRIPTION" \
    --display-name="UPDATED_DISPLAY_NAME"
```

The output is the renamed service account:

```
description: UPDATED_SA_DESCRIPTION
displayName: UPDATED_DISPLAY_NAME
name:
projects/PROJECT_ID/serviceAccounts/SA_NAME@PROJECT_ID.iam.gserviceaccount.com
```

Disabling a service account

Similar to deleting a service account, when you disable a service account, applications will no longer have access to Google Cloud resources through that service account. If you disable the default App Engine and Compute Engine service accounts, the instances will no longer have access to resources in the project. If you attempt to disable an already disabled service account, it will have no effect.

Unlike deleting a service account, disabled service accounts can easily be re-enabled as necessary. We recommend disabling a service account before deleting it to make sure no critical applications are using the service account.

To disable a service account, at minimum the user must be granted the Service Account Admin role (`roles/iam.serviceAccountAdmin`) or the Editor basic role (`roles/editor`).

ConsolegcloudRESTC#GoJavaPython
Execute the `gcloud iam service-accounts disable` command to disable a service account.

Command:

```
gcloud iam service-accounts disable SA_NAME@PROJECT_ID.iam.gserviceaccount.com
```

Output:

```
Disabled service account SA_NAME@PROJECT_ID.iam.gserviceaccount.com
```

Enabling a service account

After enabling a disabled service account, applications will regain access to Google Cloud resources through that service account.

You can enable a disabled service account whenever you need to. If you attempt to enable an already enabled service account, it will have no effect.

To enable a service account, at minimum the user must be granted the Service Account Admin role (`roles/iam.serviceAccountAdmin`) or the Editor basic role (`roles/editor`).

ConsolegcloudRESTC#GoJavaPython
Execute the `gcloud iam service-accounts enable` command to enable a service account.

Command:

```
gcloud iam service-accounts enable SA_NAME@PROJECT_ID.iam.gserviceaccount.com
```

Output:

```
Enabled service account SA_NAME@PROJECT_ID.iam.gserviceaccount.com
```

Deleting a service account

When you delete a service account, applications will no longer have access to Google Cloud resources through that service account. If you delete the default App Engine and Compute Engine service accounts, the instances will no longer have access to resources in the project.

Delete with caution; make sure your critical applications are no longer using a service account before deleting it. If you're not sure whether a service account is being used, we recommend disabling the service account before deleting it. Disabled service accounts can be easily re-enabled if they are still in use.

When a service account is deleted, its role bindings are not immediately removed; they are automatically purged from the system after a maximum of 60 days.

Deleted service accounts do not count towards your service account quota.

To delete a service account, at minimum the user must be granted the Service Account Admin role (`roles/iam.serviceAccountAdmin`) or the Editor basic role (`roles/editor`).

Execute the `gcloud iam service-accounts delete` command to delete a service account.

Command:

```
gcloud iam service-accounts delete \
    SA_NAME@PROJECT_ID.iam.gserviceaccount.com
```

Output:

```
Deleted service account SA_NAME@PROJECT_ID.iam.gserviceaccount.com
```

Undeleting a service account

In some cases, you can use the `undelete` command to undelete a deleted service account. You can usually undelete a deleted service account if it meets these criteria:

- **The service account was deleted less than 30 days ago.**

  After 30 days, IAM permanently removes the service account. Google Cloud cannot recover the service account after it is permanently removed, even if you file a support request.

- **There is no existing service account with the same name as the deleted service account.**

  For example, suppose that you accidentally delete the service account `my-service-account@project-id.iam.gserviceaccount.com`. You still need a service account with that name, so you create a new service account with the same name, `my-service-account@project-id.iam.gserviceaccount.com`.

  The new service account does not inherit the permissions of the deleted service account. In effect, it is completely separate from the deleted service account. However, you cannot undelete the original service account, because the new service account has the same name.

  To address this issue, delete the new service account, then try to undelete the original service account.

If you are not able to undelete the service account, you can create a new service account with the same name; revoke all of the roles from the deleted service account; and grant the same roles to the new service account. For details, see Policies with deleted members.

Finding a deleted service account's numeric ID

When you undelete a service account, you must provide its numeric ID. The numeric ID is a 21-digit number, such as `123456789012345678901`, that uniquely identifies the service account. For example, if you delete a service account, then create a new service account with the same name, the original service account and the new service account will have different numeric IDs.

If you know that a binding in an IAM policy includes the deleted service account, you can get the policy, then find the numeric ID in the policy. The numeric ID is appended to the name of the deleted service account. For example, in this policy, the numeric ID for the deleted service account is `123456789012345678901`:

```
{
  "version": 1,
  "etag": "BwUjMhCsNvY=",
  "bindings": [
    {
      "members": [
        "deleted:serviceAccount:my-service-account@project-
id.iam.gserviceaccount.com?uid=123456789012345678901"
      ],
      "role": "roles/iam.serviceAccountUser"
    },
  ]
}
```

Numeric IDs are only appended to the names of deleted members.

Alternatively, you can search your audit logs for the `DeleteServiceAccount` operation that deleted the service account:

1. In the Cloud Console, go to the **Logs explorer** page.

   Go to Logs explorer

2. In the query editor, enter the following query, replacing *`SERVICE_ACCOUNT_EMAIL`* with the email address of your service account (for example, `my-service-account@project-id.iam.gserviceaccount.com`):

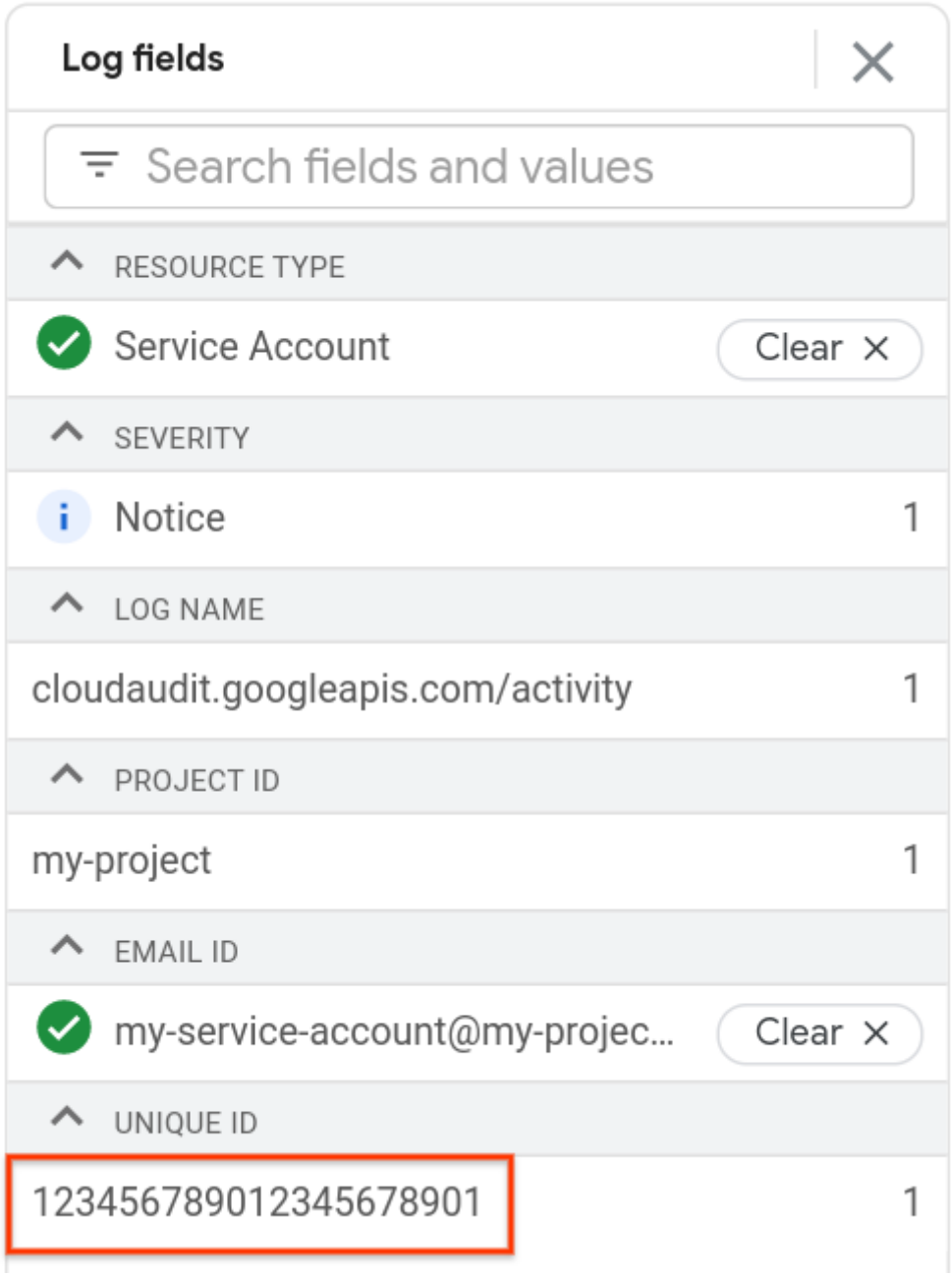
   ```
   resource.type="service_account"
   resource.labels.email_id="SERVICE_ACCOUNT_EMAIL"
   "DeleteServiceAccount"
   ```

3. If the service account was deleted more than an hour ago, click schedule **Last 1 hour**, select a longer period of time from the drop-down list, then click **Apply**.

4. Click **Run query**. The Logs Viewer displays the `DeleteServiceAccount` operations that affected service accounts with the name you specified.

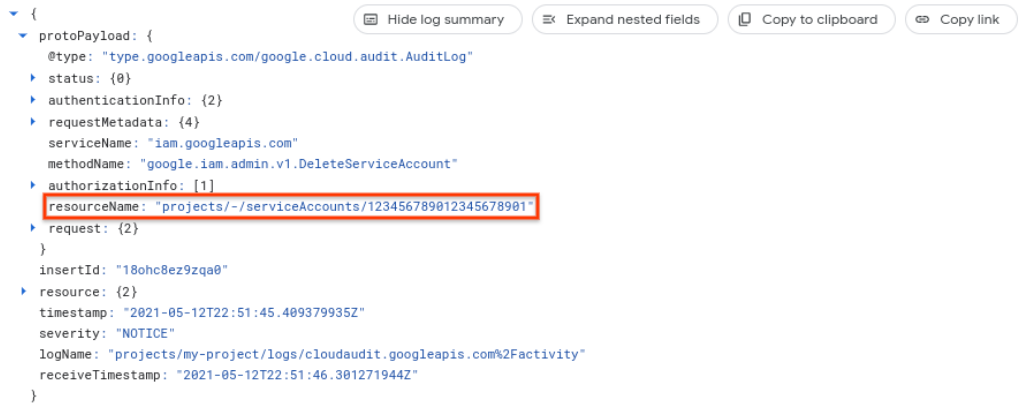
5. Find and note the numeric ID of the deleted service account by doing one of the following:

    - If the search results include only one `DeleteServiceAccount` operation, find the numeric ID in the **Unique ID** field of the **Log fields** pane.
    - If the search results show more than one log, do the following:



        1. Find the correct log entry. To find the correct log entry, click the keyboard_arrow_right expander arrow next to a log entry. Review the details of the log entry and determine whether the log entry shows the operation that you want to undo. Repeat this process until you find the correct log entry.

2. In the correct log entry, locate the service account's numeric ID. To locate the numeric ID, expand the log entry's `protoPayload` field, then find the `resourceName` field.



The numeric ID is everything after `serviceAccounts` in the `resourceName` field.

Undeleting the service account by numeric ID

After you find the numeric ID for the deleted service account, you can try to undelete the service account.

gcloudREST

Execute the `gcloud beta iam service-accounts undelete` command to undelete a service account.

Command:

```
gcloud beta iam service-accounts undelete ACCOUNT_ID
```

Output:

```
restoredAccount:
  email: SA_NAME@PROJECT_ID.iam.gserviceaccount.com
  etag: BwWWE7zpApg=
  name:
projects/PROJECT_ID/serviceAccounts/SA_NAME@PROJECT_ID.iam.gserviceaccount.com
  oauth2ClientId: '123456789012345678901'
  projectId: PROJECT_ID
  uniqueId: 'ACCOUNT_ID'
```

What's next

- Learn how to create and manage service account keys.
- Review the process for granting IAM roles to all types of members, including service accounts.
- Explore how you can use role recommendations to downscope permissions for all members, including service accounts.
- Understand how to allow members to impersonate service accounts.

Rate and review