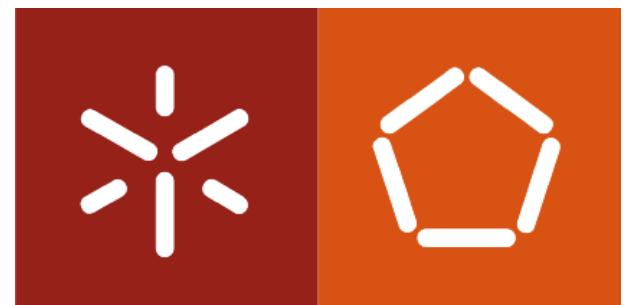


Cloud Computing Applications and Services

(Aplicações e Serviços de Computação em Nuvem)

Google Kubernetes Engine

University of Minho
2025-2026

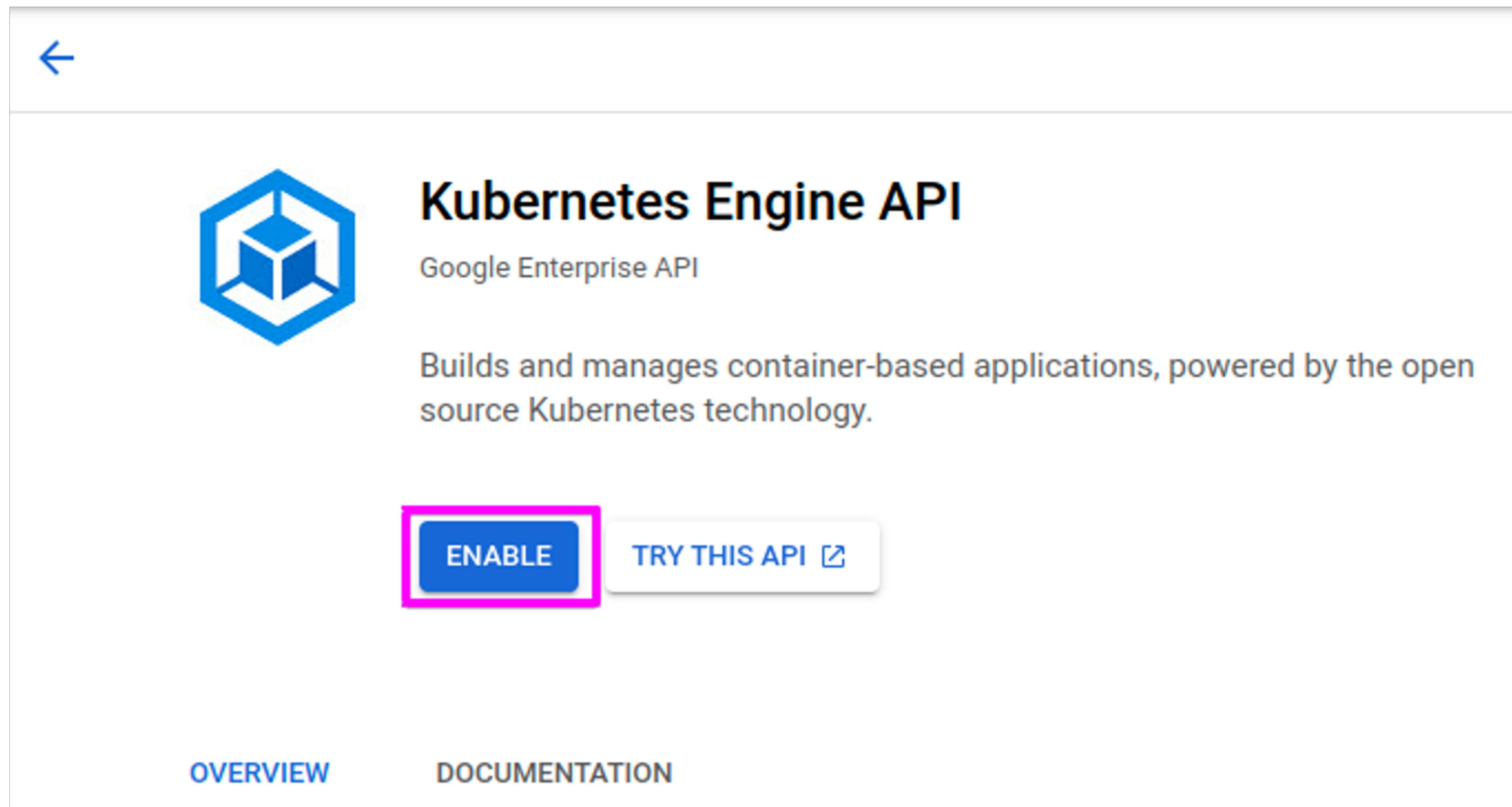


Bootstrap

- Access Google Cloud Console: <https://console.cloud.google.com/>
- Create a project
 - ▶ Associate billing account (check e-mail for coupon)
 - ▶ Each account has 50\$

Enable Kubernetes Engine API

(Side bar → Kubernetes Engine → Enable)



Create a Service Account

(Side bar → IAM & Admin → Service accounts)

+ CREATE SERVICE ACCOUNT

1 Service account details

Fill the *Service account name* and click on “*Create and continue*”

1 Service account details

Service account name x c

Display name for this service account

Service account ID * x c

Email address: sa-example@ascn2223.iam.gserviceaccount.com ✉

Service account description

Describe what this service account will do

[CREATE AND CONTINUE](#)

2 Grant this service account access to project (optional)

3 Grant users access to this service account (optional)

[DONE](#) [CANCEL](#)

Create a Service Account

(Side bar → IAM & Admin → Service accounts)

2 Grant this service account access to project

Grant the following Roles to the service account:

- ▶ Compute Admin
- ▶ Kubernetes Engine Admin
- ▶ Service Account User

The screenshot shows the 'Service account details' section of the Google Cloud IAM & Admin Service Accounts interface. It lists three roles assigned to the service account:

- Role:** Compute Admin (IAM condition optional) [+ ADD IAM CONDITION](#) [Delete](#)
Full control of all Compute Engine resources.
- Role:** Kubernetes Engine Admin (IAM condition optional) [+ ADD IAM CONDITION](#) [Delete](#)
Full management of Kubernetes Clusters and their Kubernetes API objects.
- Role:** Service Account User (IAM condition optional) [+ ADD IAM CONDITION](#) [Delete](#)
Run operations as the service account.

At the bottom, there are 'CONTINUE' and 'DONE' buttons, and a 'CANCEL' button.

3 Grant users access to this service account (optional)

Create a Service Account

(Side bar → IAM & Admin → Service accounts)

3 Grant users access to this service account

Add the email of each element of the group to both “*Service account users role*” and “*Service account admins role*”

The screenshot shows the third step of creating a service account. It includes a list of optional steps, a main input field for user emails, and two dropdown menus for roles.

- Service account details
- Grant this service account access to project (optional)
- 3 Grant users access to this service account (optional)

Grant users access to this service account (optional)
Grant access to users or groups that need to perform actions as this service account. [Learn more](#)

Service account users role

- studentA@gmail.com X
- studentB@gmail.com X
- studentC@gmail.com X

Grant users the permissions to deploy jobs and VMs with this service account

Service account admins role

- studentA@gmail.com X
- studentB@gmail.com X
- studentC@gmail.com X

Grant users the permission to administer this service account

DONE CANCEL

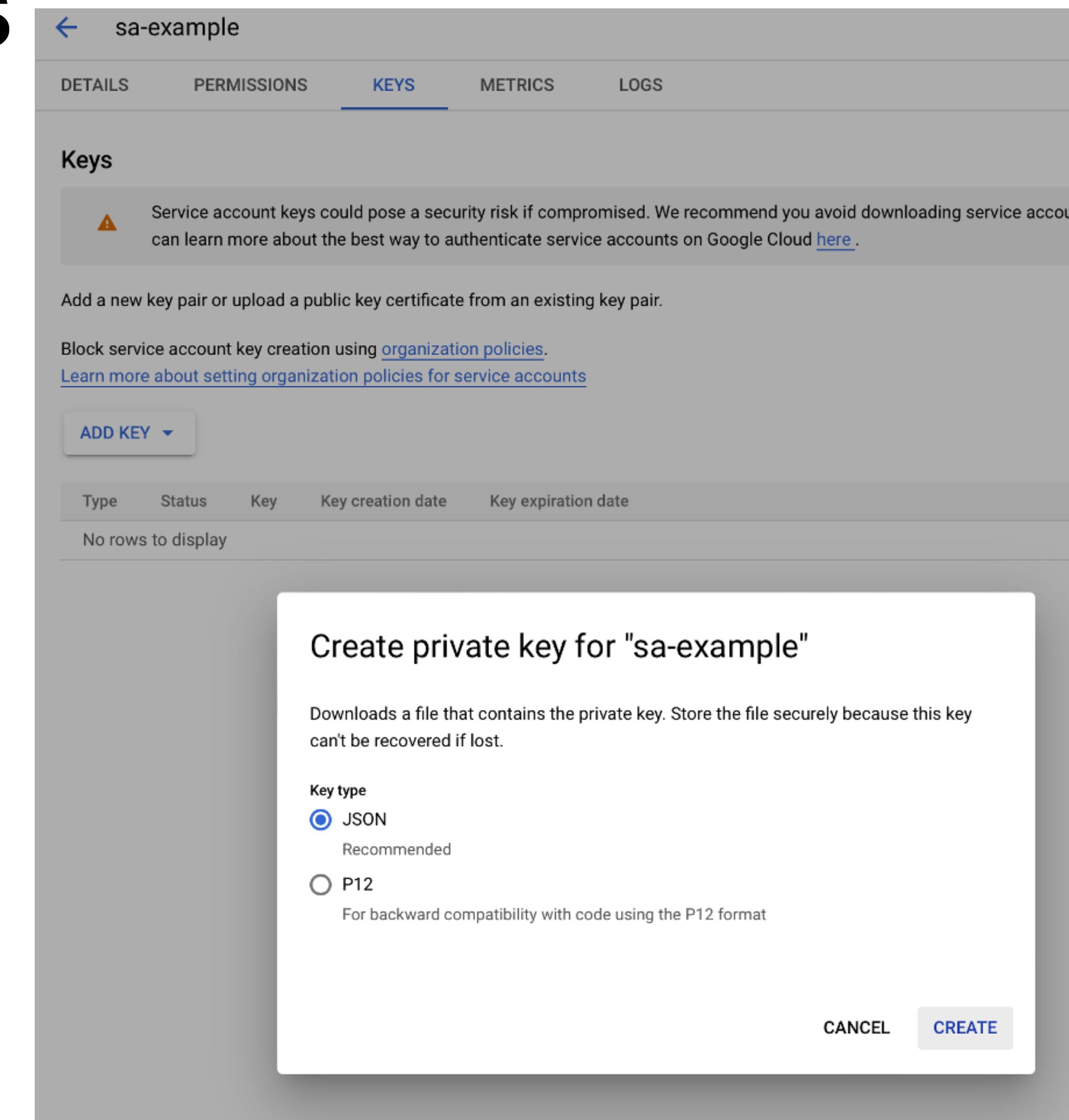
Get Service Account Keys

(Side bar → IAM & Admin → Service accounts)

- 1 Select your service account
- 2 On top select “Keys”
- 3 Select “ADD KEY” → “Create new key”
- 3 Select “JSON” → “CREATE”

Note: The downloaded file should be used as the credential file by the [GKE_cluster_create](#) playbook.

Important: This file has **private** credentials and **should not** be added to your project’s GitHub repository.



Install and Configure Google Cloud CLI

- The Google Cloud CLI must be installed and configured on the machine where the Ansible playbooks will run.
- Access the link <https://cloud.google.com/sdk/docs/install> and follow the installation instructions for the corresponding operating system.
- Configure the Google Cloud CLI with the following command:

```
gcloud init
```

GKE - Kubernetes Cluster

- Kubernetes cluster resources (e.g., control plane and nodes) are managed by Google Cloud.
- GKE has access to other Google services (e.g., load balancing, storage, monitoring, ...).
- Users can interact directly with the cluster through the gcloud CLI tool (e.g., create, configure, destroy cluster).
- gcloud CLI installs and configures kubectl so that users can deploy K8s objects at the GKE cluster (e.g., pods, services, ...).
- Ansible gcp_container_cluster module can be used to interact with gcloud CLI.

GKE - Kubernetes Cluster

