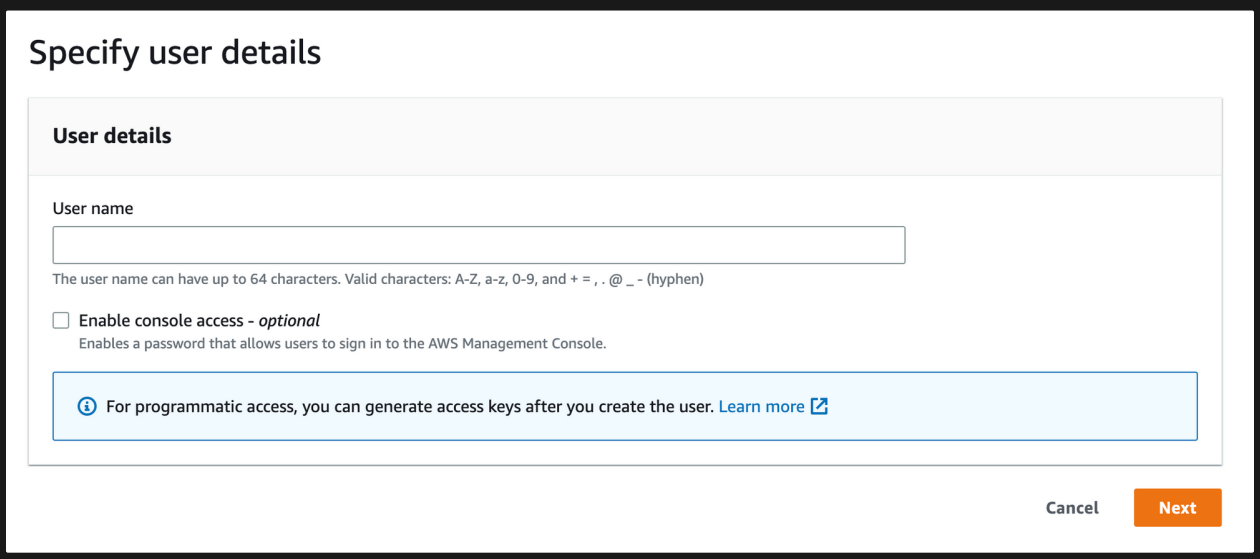
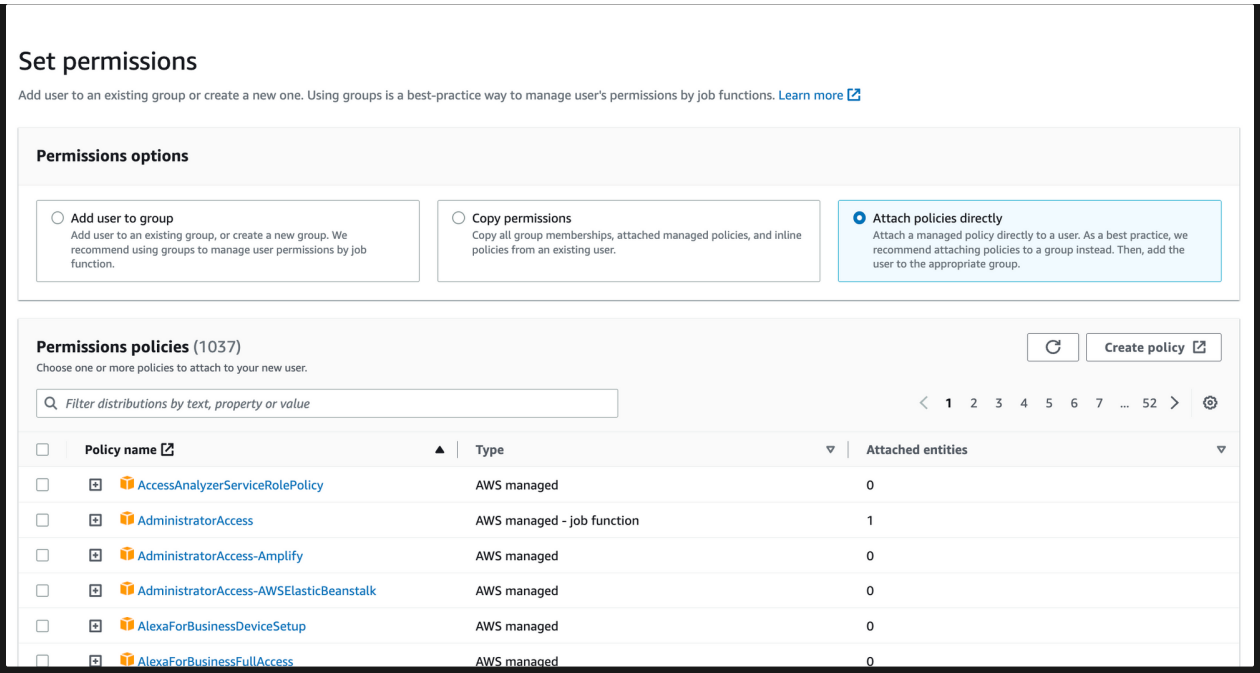
Steps to implement Hands-on Project - Mission 1

Amazon Web Services

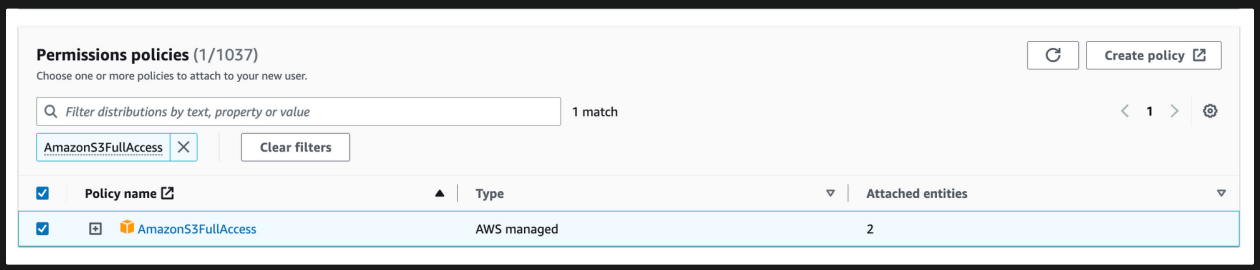
- Access AWS console and go to IAM service
- Under Access management, Click in "Users", then "Add users". Insert the User name **terraform-en-1** and click in **Next** to create a programmatic user.



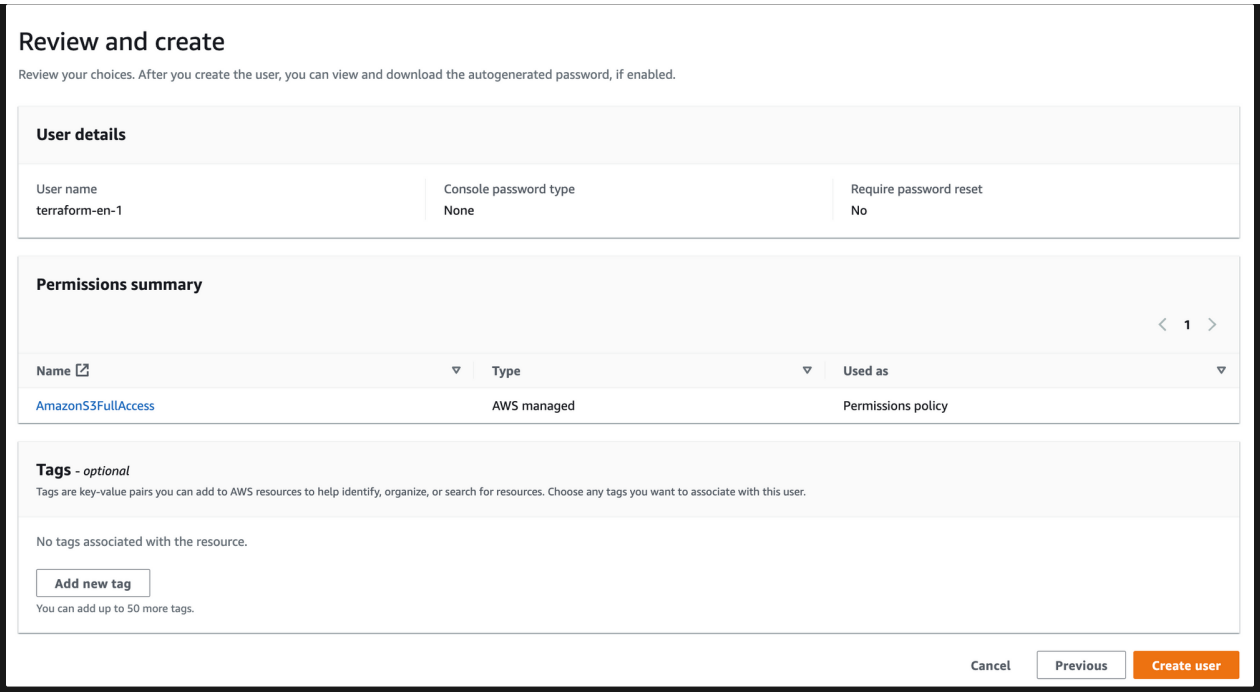
- On Set permissions, Permissions options, click in "Attach policies directly" button.



- Type **AmazonS3FullAccess** in **Filter distributions by text, property or value**, press Enter.
- Select AmazonS3FullAccess



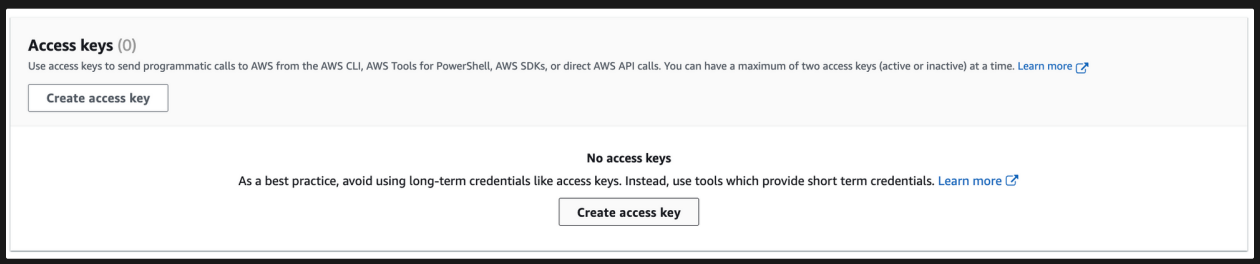
- Click in Next
- Review all details



- Click in Create user

[NEW] AWS has recently changed the way to download the key. Follow the new steps:

- Click on the user you have created.
- Click on Security credentials.
- Scroll down and go to Access keys section.
- Click on Create access key



- Select **Command Line Interface (CLI)** and **I understand the above recommendation and want to proceed to create an access key** checkbox.
- Click Next
- Click on Create access key
- Click on Download .csv file

[TIP] Access key best practices

- Never store your access key in plain text, in a code repository, or in code.
- Disable or delete access key when no longer needed.
- Enable least-privilege permissions.
- Rotate access keys regularly.

- After download, click Done.
- Now, rename .csv file downloaded to accessKeys.csv

Google Cloud Platform (GCP)

- CLICK HERE to download the hands-on files.
- Access GCP Console and open Cloud Shell
- Upload accessKeys.csv and .zip hands-on file to GCP Cloud Shell
- Hands-on files preparation

```
mkdir mission1_en mv mission1.zip mission1_en cd mission1_en unzip
mission1.zip mv ~/accessKeys.csv mission1/en cd mission1/en chmod +x *.sh
```

- Run the following commands to prepare AWS and GCP environment. Authorize when asked.

```
./aws_set_credentials.sh accessKeys.csv gcloud config set project
<project_id>
```

- Execute the command below

```
./gcp_set_project.sh
```

- Enable the Container Registry API, Kubernetes Engine API and the Cloud SQL API

```
gcloud services enable containerregistry.googleapis.com gcloud services
enable container.googleapis.com gcloud services enable
sqladmin.googleapis.com
```

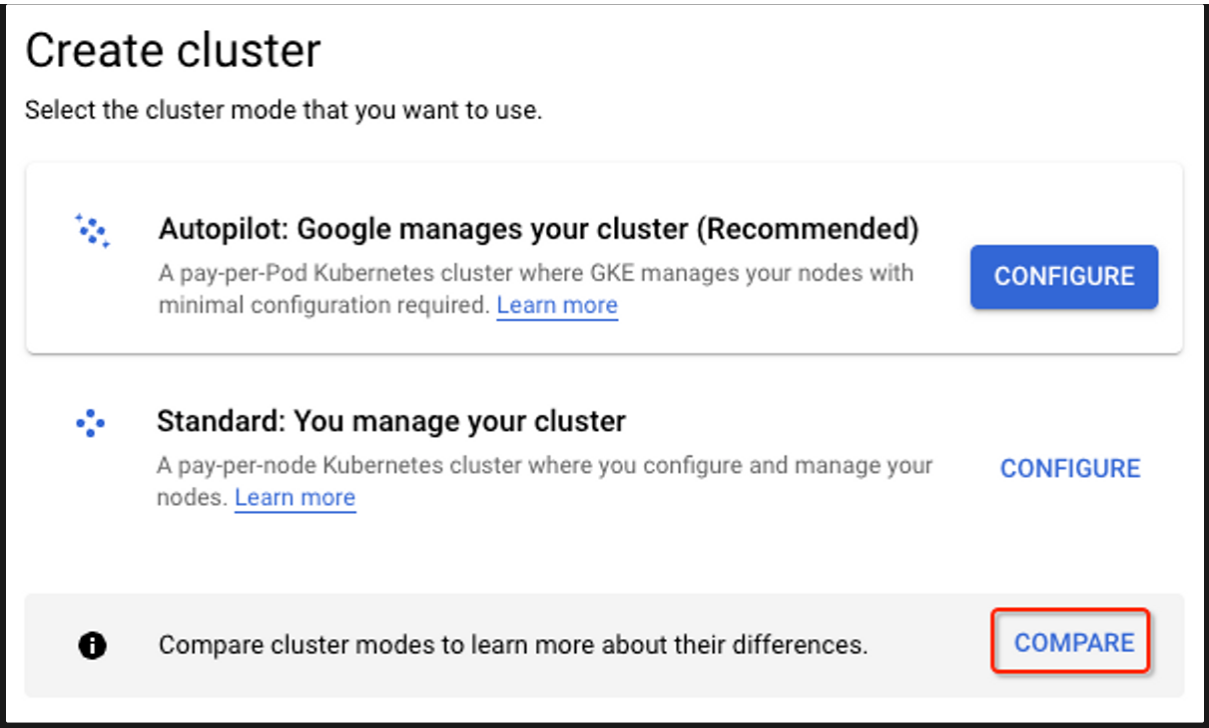
IMPORTANT (DO NOT SKIP):

- **Before executing the Terraform commands, open the Google Editor and update the file tcb_aws_storage.tf replacing the bucket name with an unique name (AWS requires unique bucket names).**
  - Open the tcb_aws_storage.tf using Google Editor
  - On line 4 of the file tcb_aws_storage.tf:
    - Replace xxxx with your name initials plus two random numbers: Example: luxxy-covid-testing-system-pdf-en-jr29

- Run the following commands to finish provision infrastructure steps

```
cd ~/mission1_en/mission1/en/terraform/ terraform init terraform plan
terraform apply Type Yes and go ahead.
```

💡 After access the GKE Service to create a cluster, click on the **Compare** button to "**Compare cluster modes to learn more about their differences**".


Create cluster
Select the cluster mode that you want to use.
Autopilot: Google manages your cluster (Recommended)
A pay-per-Pod Kubernetes cluster where GKE manages your nodes with minimal configuration required. Learn more
CONFIGURE
Standard: You manage your cluster
A pay-per-node Kubernetes cluster where you configure and manage your nodes. Learn more
CONFIGURE
Compare cluster modes to learn more about their differences.
COMPARE

Create cluster

Select the cluster mode that you'd like to use. Learn more

	Autopilot mode	Standard mode
	Optimized Kubernetes cluster with a hands-off experience	Kubernetes cluster with node configuration flexibility
	CONFIGURE TRY THE DEMO	CONFIGURE TRY THE DEMO
Scaling	Automatic based on workload	You configure scaling
Nodes	Google manages and configures your nodes	You manage and configure your nodes
Configuration	Streamlined configuration ready to use	You can configure all options
Workloads supported	Most workloads except these limitations	All Kubernetes workloads
Billing method	Pay per pod	Pay per node (VM)
SLA	Kubernetes API and node availability	Kubernetes API availability

View all

💡 Download Visual Studio Code used by Jean during the Training HERE

SQL Network Configuration