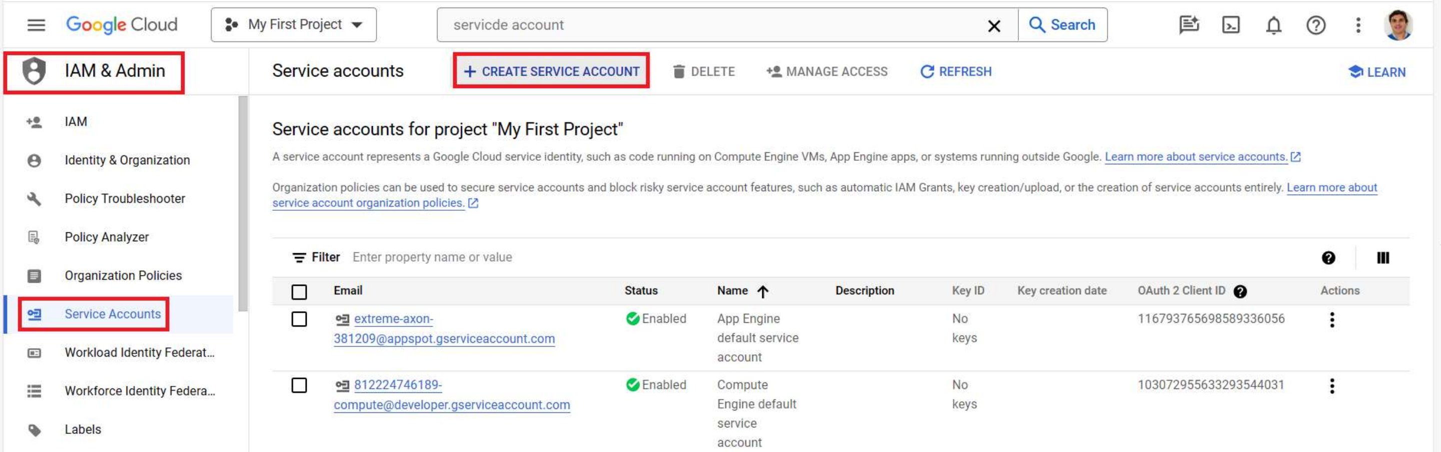
GithubActions: How to create .NET8 WebAPI Docker image and upload to Google Cloud Artifacts Registry

You can see this example source code in this github repo:

https://github.com/luiscoco/GithubActions_dotNET8WebAPI_Create_DockerImage_Upload_to_Google_Cloud_Artifacts_Registry

1. Create a Service Account in Google Cloud Platform

Go to the GCP Console: Open the Google Cloud Console and log in to your account.



Service accounts for project "My First Project"

A service account represents a Google Cloud service identity, such as code running on Compute Engine VMs, App Engine apps, or systems running outside Google. Learn more about service accounts.

Organization policies can be used to secure service accounts and block risky service account features, such as automatic IAM Grants, key creation/upload, or the creation of service accounts entirely. Learn more about service account organization policies.

□	Email	Status	Name ↑	Description	Key ID	Key creation date	OAuth 2 Client ID	Actions
□	extreme-axon-381209@appspot.gserviceaccount.com	✓ Enabled	App Engine default service account		No keys		116793765698589336056	⋮
□	812224746189-compute@developer.gserviceaccount.com	✓ Enabled	Compute Engine default service account		No keys		103072955633293544031	⋮

Select Your Project: Make sure you have the correct project selected in which you want to create the service account.

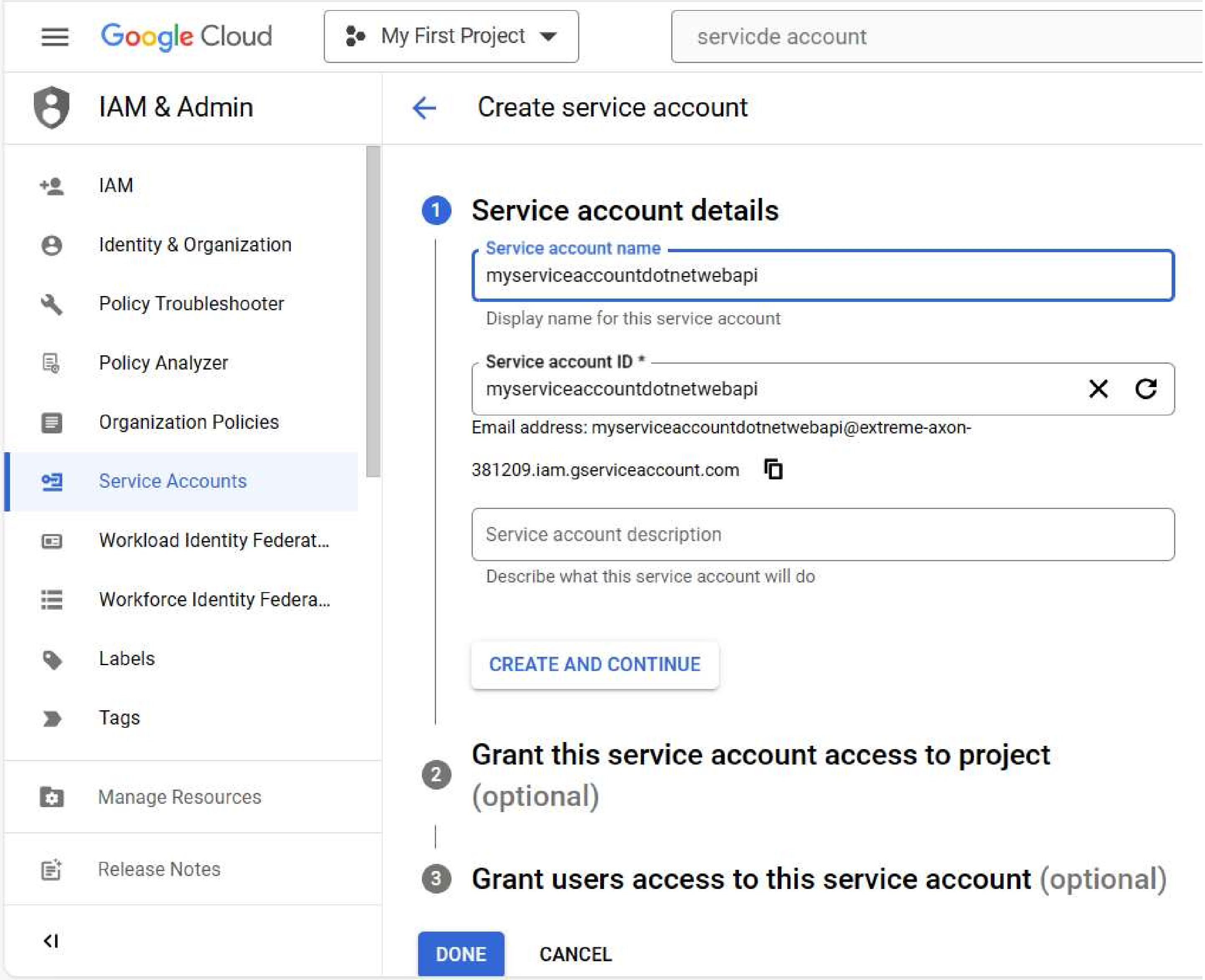
Navigate to IAM & Admin: In the left-hand menu, click on "IAM & Admin", then select "Service Accounts".

Create Service Account: Click on "Create Service Account" and fill in the necessary details:

Name: Give your service account a name.

ID: This is filled automatically based on the name but can be customized.

Description: (Optional) Add a description for your service account.



Create service account

1 Service account details

Service account name: myserviceaccountdotnetwebapi

Display name for this service account

Service account ID *: myserviceaccountdotnetwebapi

Email address: myserviceaccountdotnetwebapi@extreme-axon-381209.iam.gserviceaccount.com

Service account description

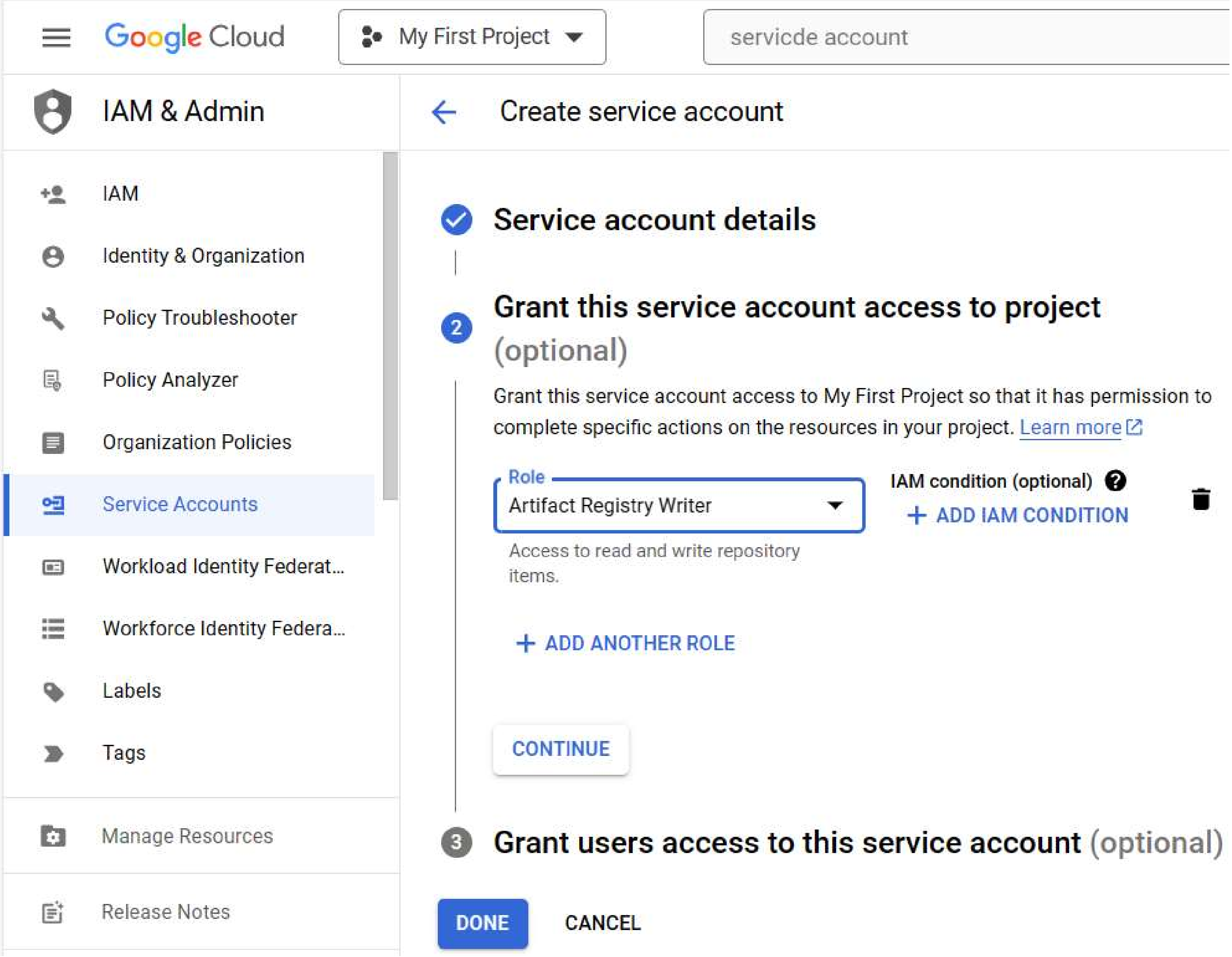
Describe what this service account will do

2 Grant this service account access to project (optional)

3 Grant users access to this service account (optional)

Grant Access: Assign the service account appropriate roles. For Docker images push we can assing the role: "**Artifact Registry Writer**"

Other similar roles could be: "**Storage Admin**" or "**Artifact Registry Administrator**"



Create service account

Service account details

2 Grant this service account access to project (optional)

Grant this service account access to My First Project so that it has permission to complete specific actions on the resources in your project. Learn more

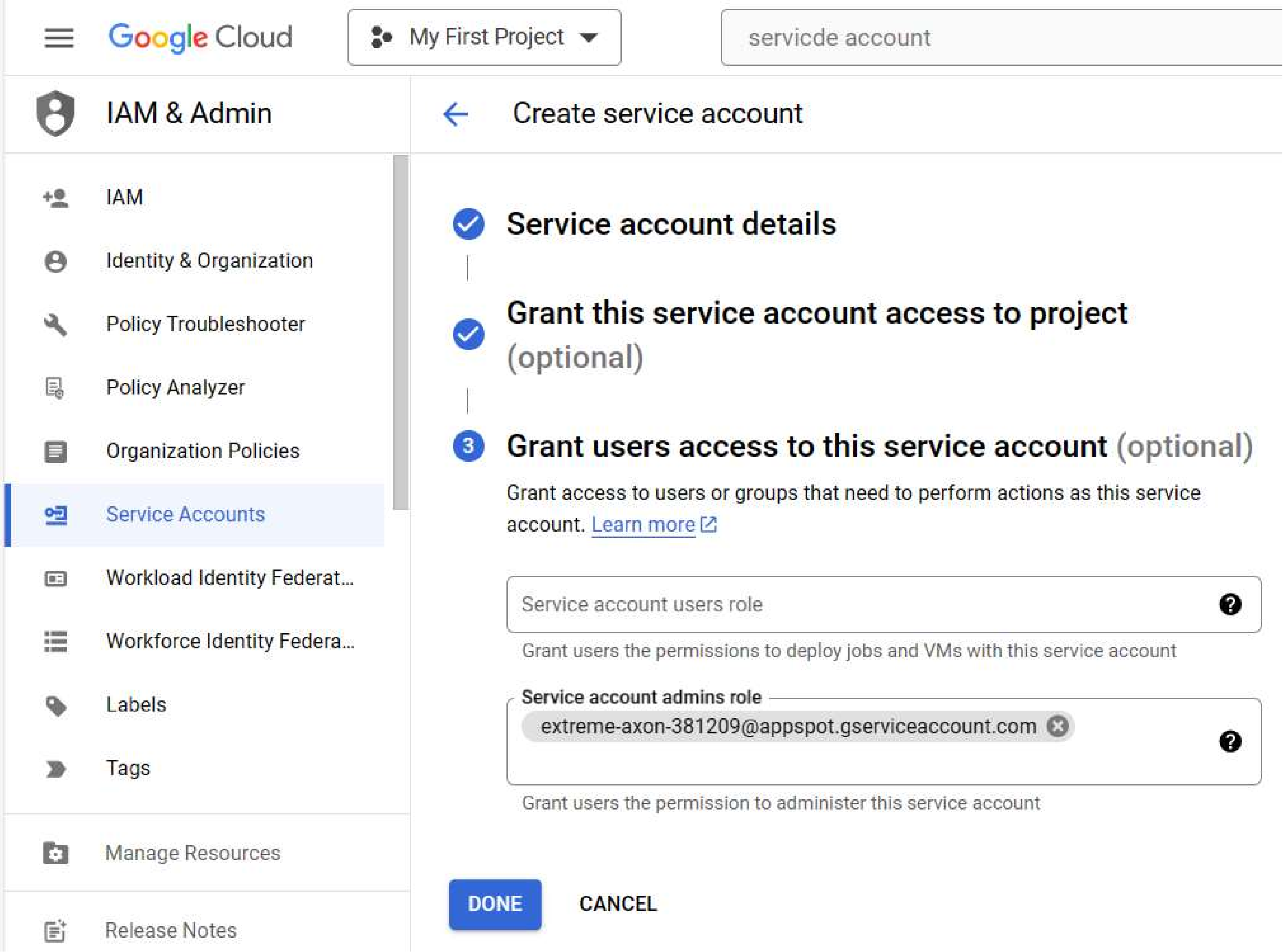
Role: Artifact Registry Writer

Access to read and write repository items.

IAM condition (optional)

3 Grant users access to this service account (optional)

Do not forget to set the project ID (for this example: extreme-axon-381209) in the **Service account admin role**: extreme-axon-381209@appspot.gserviceaccount.com App Engine default service account



Create service account

✓ Service account details

✓ Grant this service account access to project (optional)

3 Grant users access to this service account (optional)

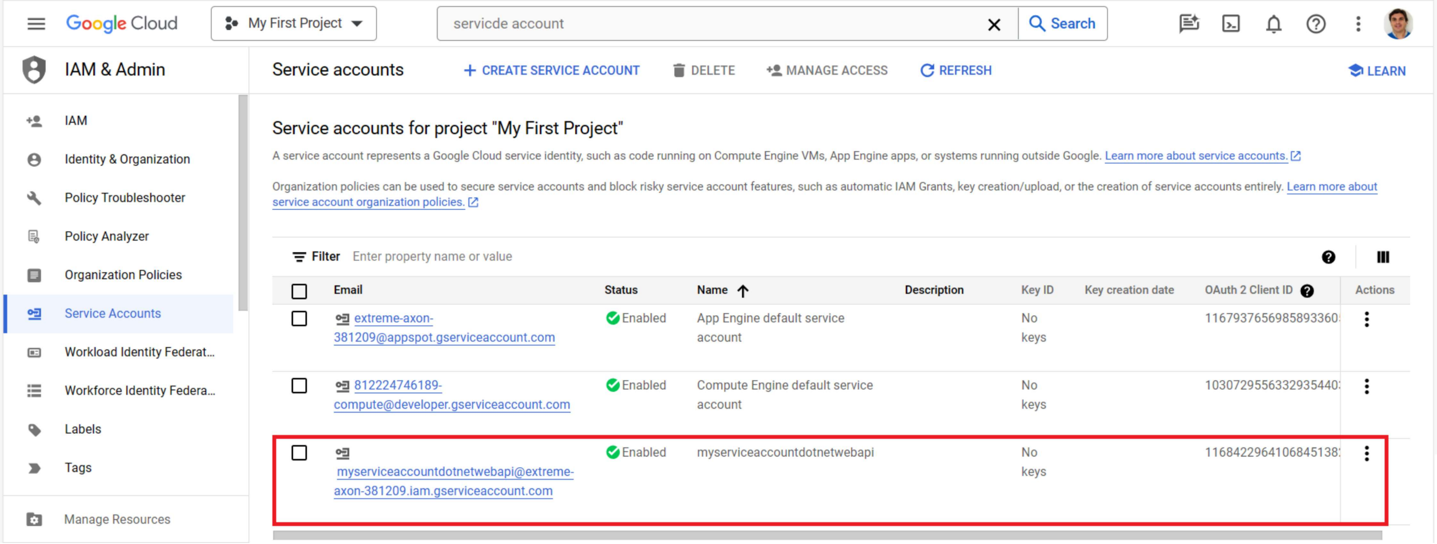
Grant access to users or groups that need to perform actions as this service account. Learn more

Service account users role

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

Service account admins role: extreme-axon-381209@appspot.gserviceaccount.com

Grant users the permission to administer this service account



Service accounts for project "My First Project"

A service account represents a Google Cloud service identity, such as code running on Compute Engine VMs, App Engine apps, or systems running outside Google. Learn more about service accounts.

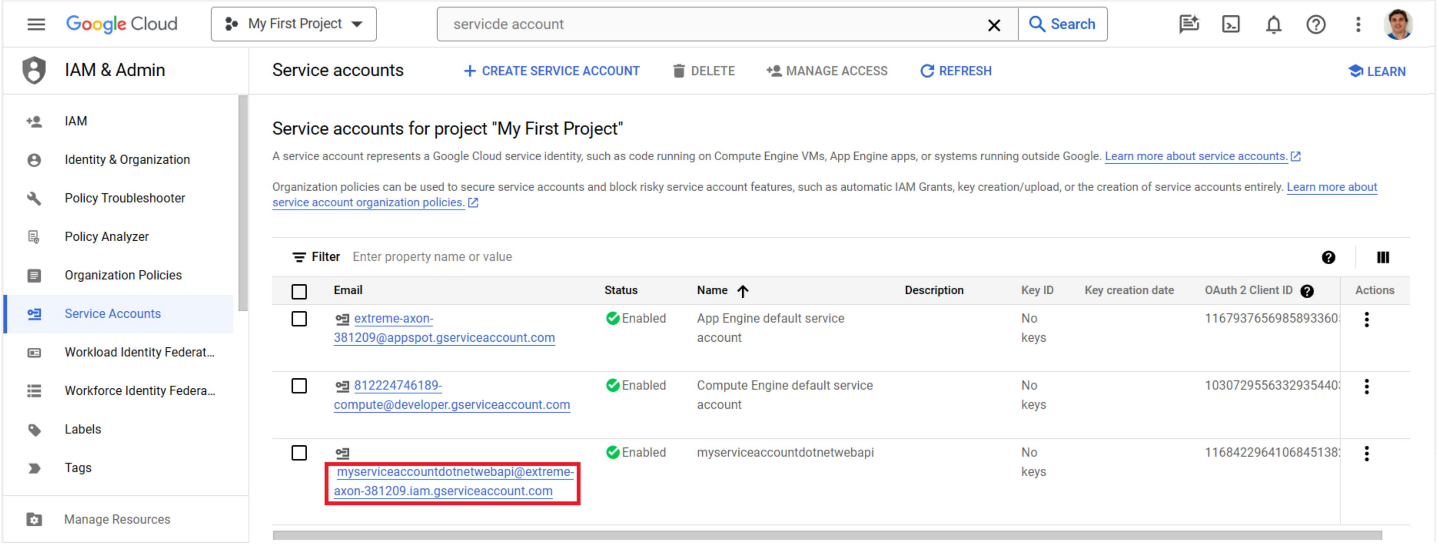
Organization policies can be used to secure service accounts and block risky service account features, such as automatic IAM Grants, key creation/upload, or the creation of service accounts entirely. Learn more about service account organization policies.

□	Email	Status	Name ↑	Description	Key ID	Key creation date	OAuth 2 Client ID	Actions
□	extreme-axon-381209@appspot.gserviceaccount.com	✓ Enabled	App Engine default service account		No keys		1167937656985893360	⋮
□	812224746189-compute@developer.gserviceaccount.com	✓ Enabled	Compute Engine default service account		No keys		1030729556332935440	⋮
□	myserviceaccountdotnetwebapi@extreme-axon-381209.iam.gserviceaccount.com	✓ Enabled	myserviceaccountdotnetwebapi		No keys		1168422964106845138	⋮

Be cautious with permissions to follow the principle of least privilege.

Create Key: After creating the service account, click on it to open its details. Under the "Keys" tab, click "Add Key", then select "Create new key".

Choose "JSON" as the key type and click "Create". This will download the JSON key file to your computer.



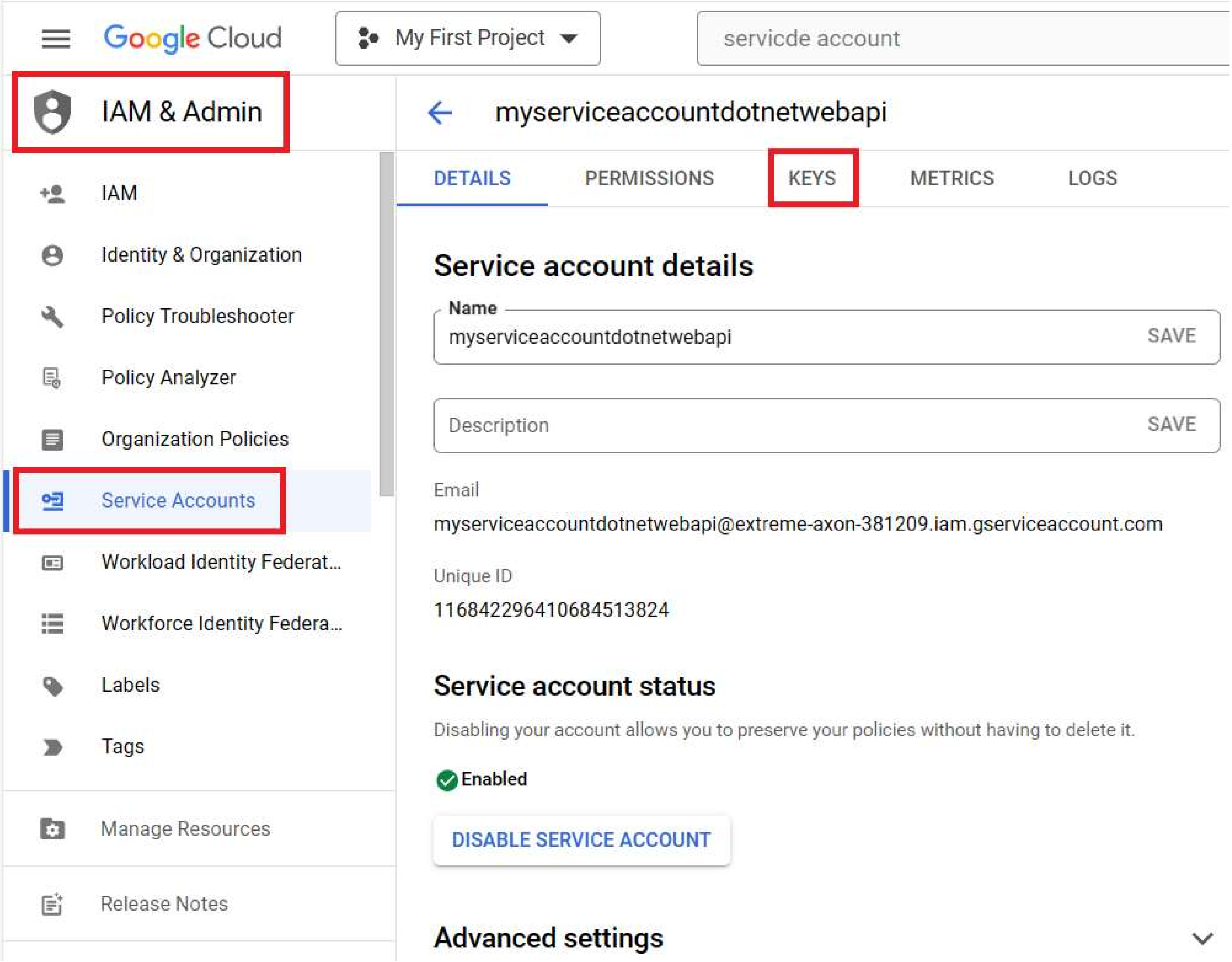
Service accounts for project "My First Project"

A service account represents a Google Cloud service identity, such as code running on Compute Engine VMs, App Engine apps, or systems running outside Google. Learn more about service accounts.

Organization policies can be used to secure service accounts and block risky service account features, such as automatic IAM Grants, key creation/upload, or the creation of service accounts entirely. Learn more about service account organization policies.

Filter Enter property name or value

□	Email	Status	Name ↑	Description	Key ID	Key creation date	OAuth 2 Client ID	Actions
□	extreme-axon-381209@appspot.gserviceaccount.com	✓ Enabled	App Engine default service account		No keys		116793765698589336O	⋮
□	812224746189-compute@developer.gserviceaccount.com	✓ Enabled	Compute Engine default service account		No keys		10307295563329354O	⋮
□	myserviceaccountdotnetwebapi@extreme-axon-381209.iam.gserviceaccount.com	✓ Enabled	myserviceaccountdotnetwebapi		No keys		116842296410684513	⋮



Service account details

Name: myserviceaccountdotnetwebapi SAVE

Description SAVE

Email

myserviceaccountdotnetwebapi@extreme-axon-381209.iam.gserviceaccount.com

Unique ID

116842296410684513824

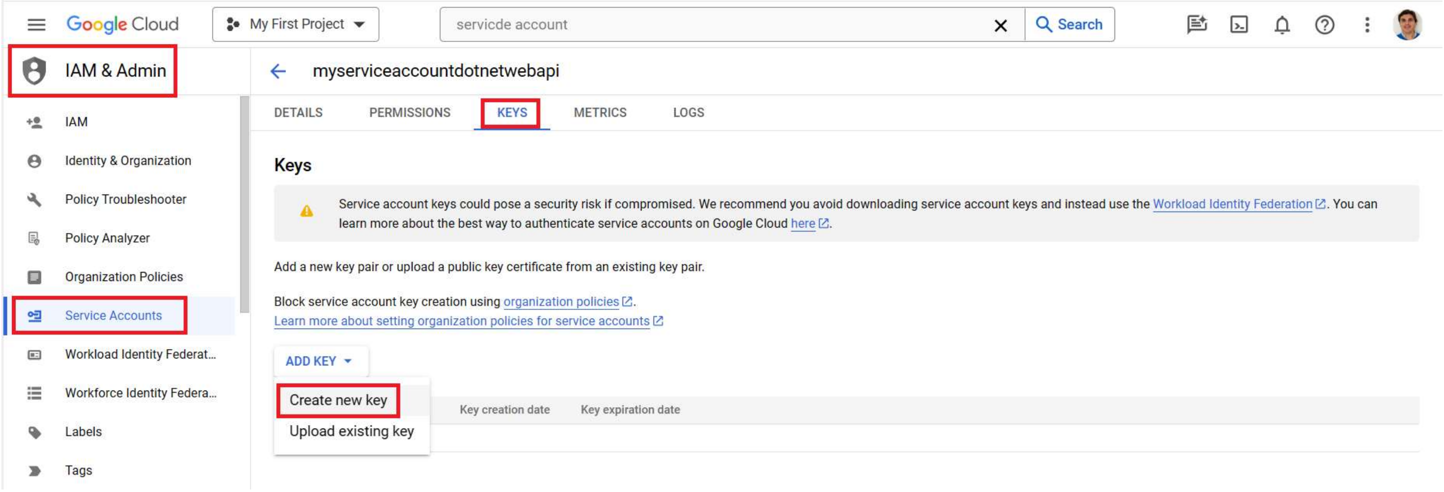
Service account status

Disabling your account allows you to preserve your policies without having to delete it.

✓ Enabled

DISABLE SERVICE ACCOUNT

Advanced settings

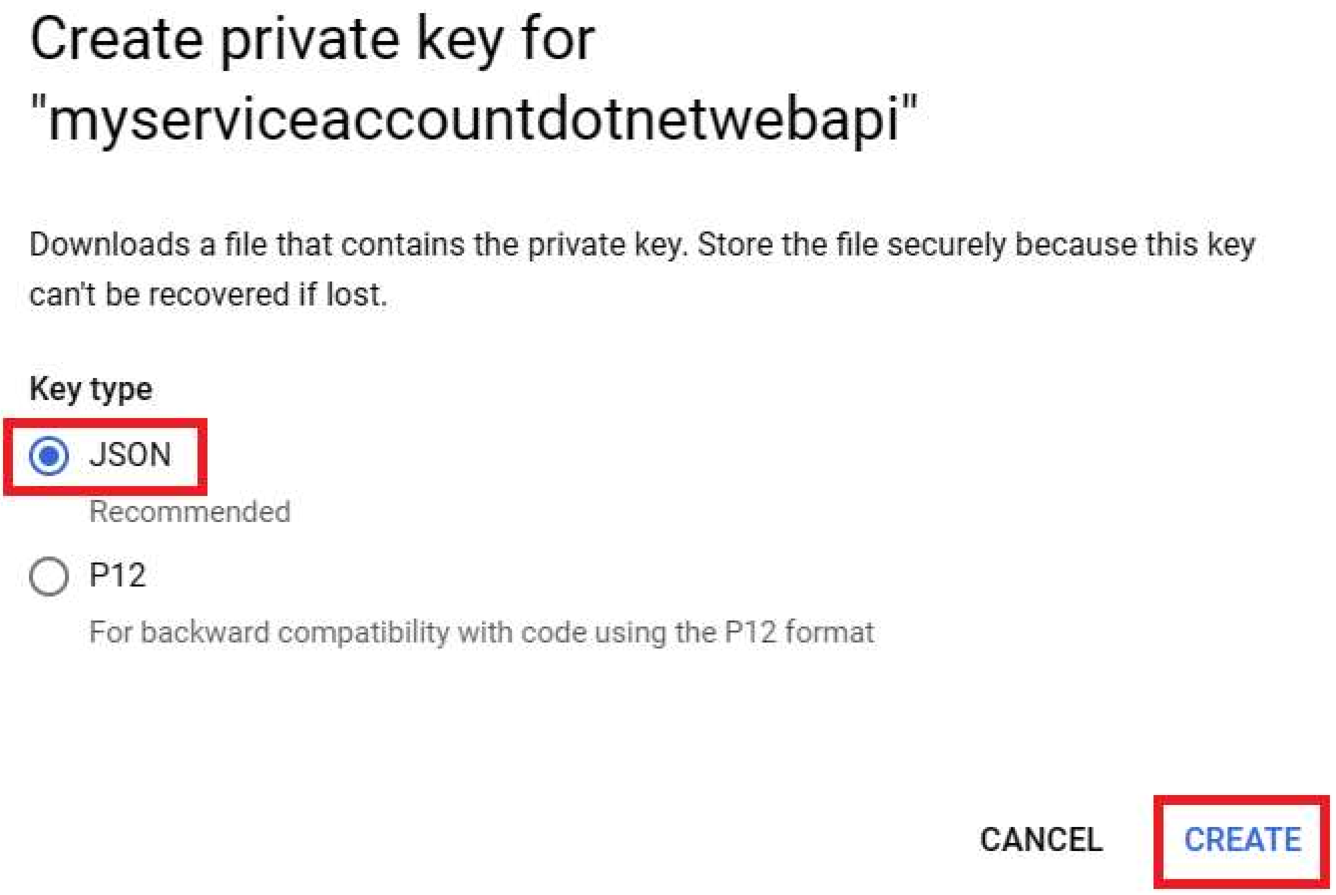


Keys

Service account keys could pose a security risk if compromised. We recommend you avoid downloading service account keys and instead use the Workload Identity Federation. You can learn more about the best way to authenticate service accounts on Google Cloud here.

Add a new key pair or upload a public key certificate from an existing key pair.

Block service account key creation using organization policies.
Learn more about setting organization policies for service accounts



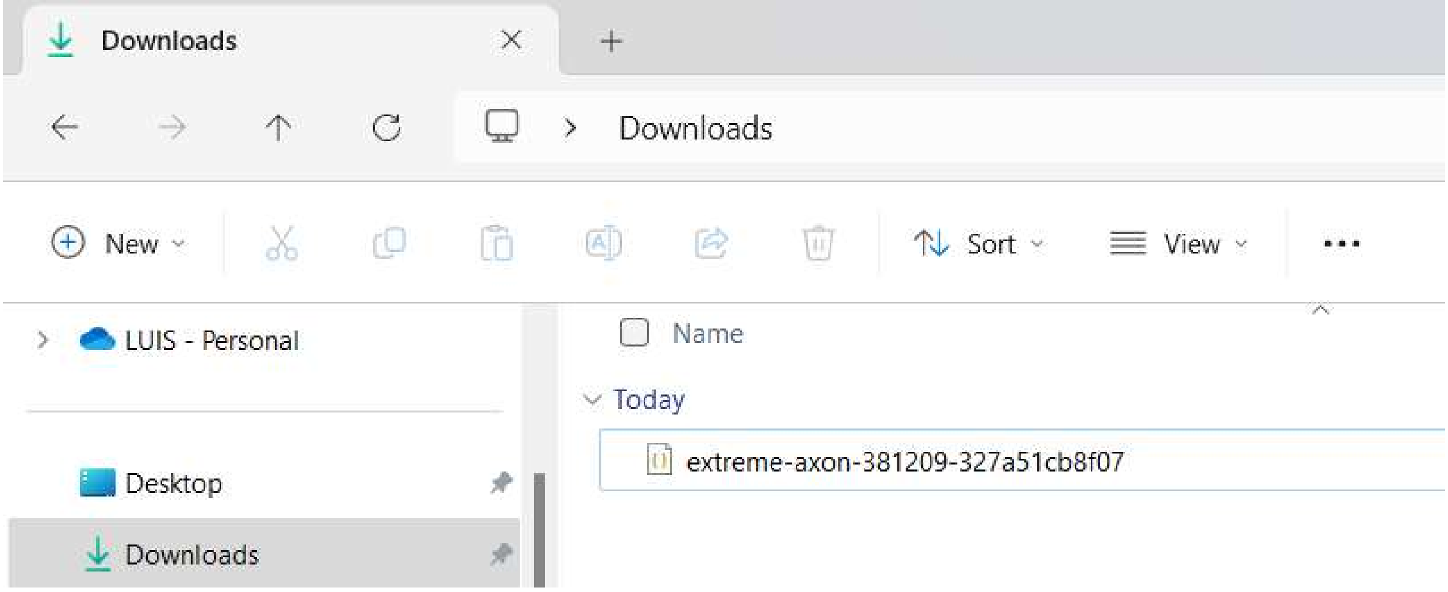
Create private key for "myserviceaccountdotnetwebapi"

Downloads a file that contains the private key. Store the file securely because this key can't be recovered if lost.

Key type

● JSON
Recommended

○ P12
For backward compatibility with code using the P12 format



extreme-axon-381209-327a51cb8f07

2. Add the Key as a Secret in your GitHub Repository

Go to Your GitHub Repository: Open your GitHub repository in a web browser.

Navigate to Settings: Click on "**Settings**" in the top menu of your repository.

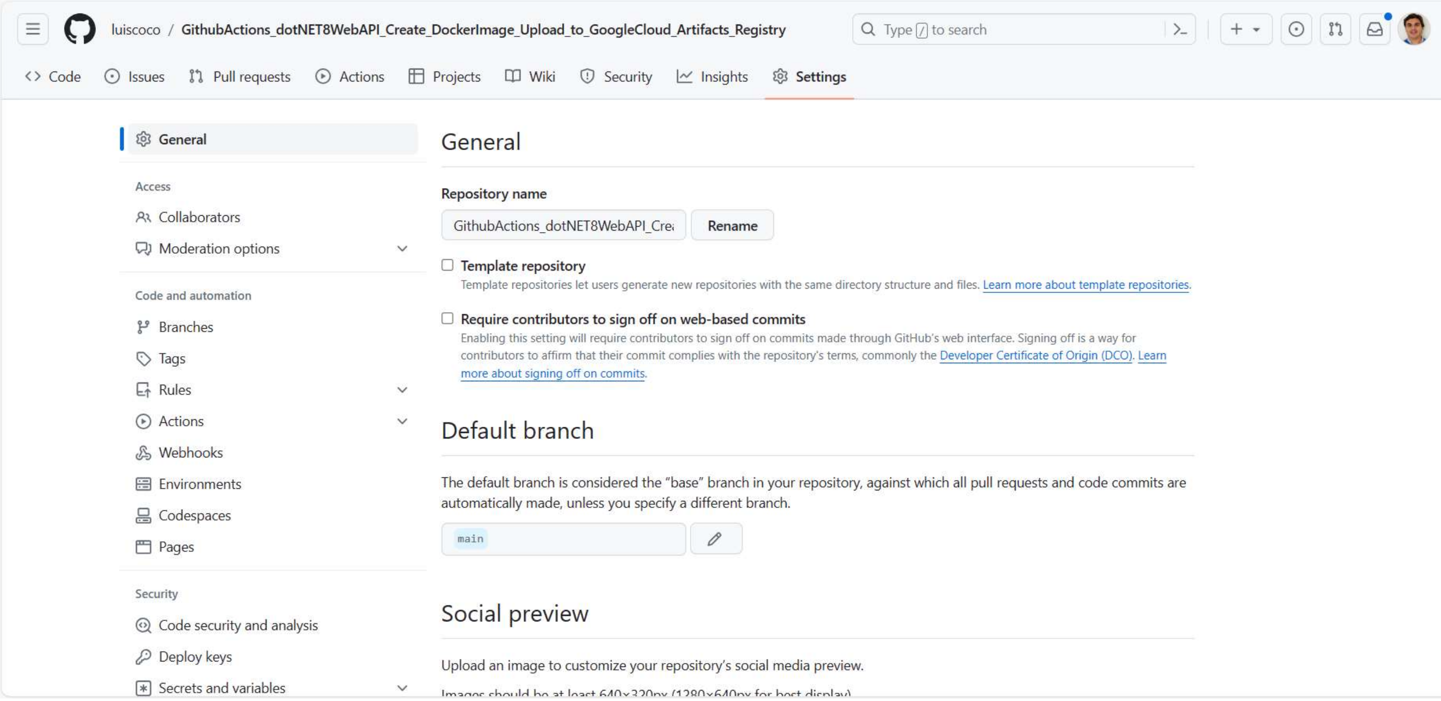
Access Secrets: In the left-hand sidebar, click on "**Secrets**", then select "**Actions**".

Add a New Secret: Click on "**New repository secret**".

Name Your Secret: Enter **GOOGLE_CLOUD_CREDENTIALS** as the name.

Paste the Key Content: Open the **JSON** key file you downloaded from GCP in a text editor, copy all its contents, and paste them into the secret's value field in GitHub.

Save the Secret: Click "Add secret" to save your new secret.



General

Repository name

GithubActions_dotNET8WebAPI_Cre: Rename

□ **Template repository**
Template repositories let users generate new repositories with the same directory structure and files. Learn more about template repositories.

□ **Require contributors to sign off on web-based commits**
Enabling this setting will require contributors to sign off on commits made through GitHub's web interface. Signing off is a way for contributors to affirm that their commit complies with the repository's terms, commonly the Developer Certificate of Origin (DCO). Learn more about signing off on commits.

Default branch

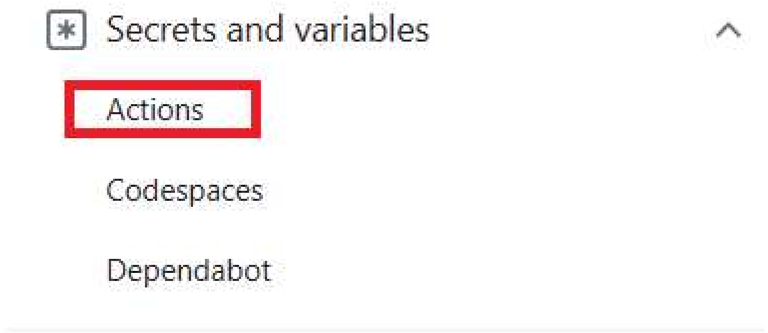
The default branch is considered the "base" branch in your repository, against which all pull requests and code commits are automatically made, unless you specify a different branch.

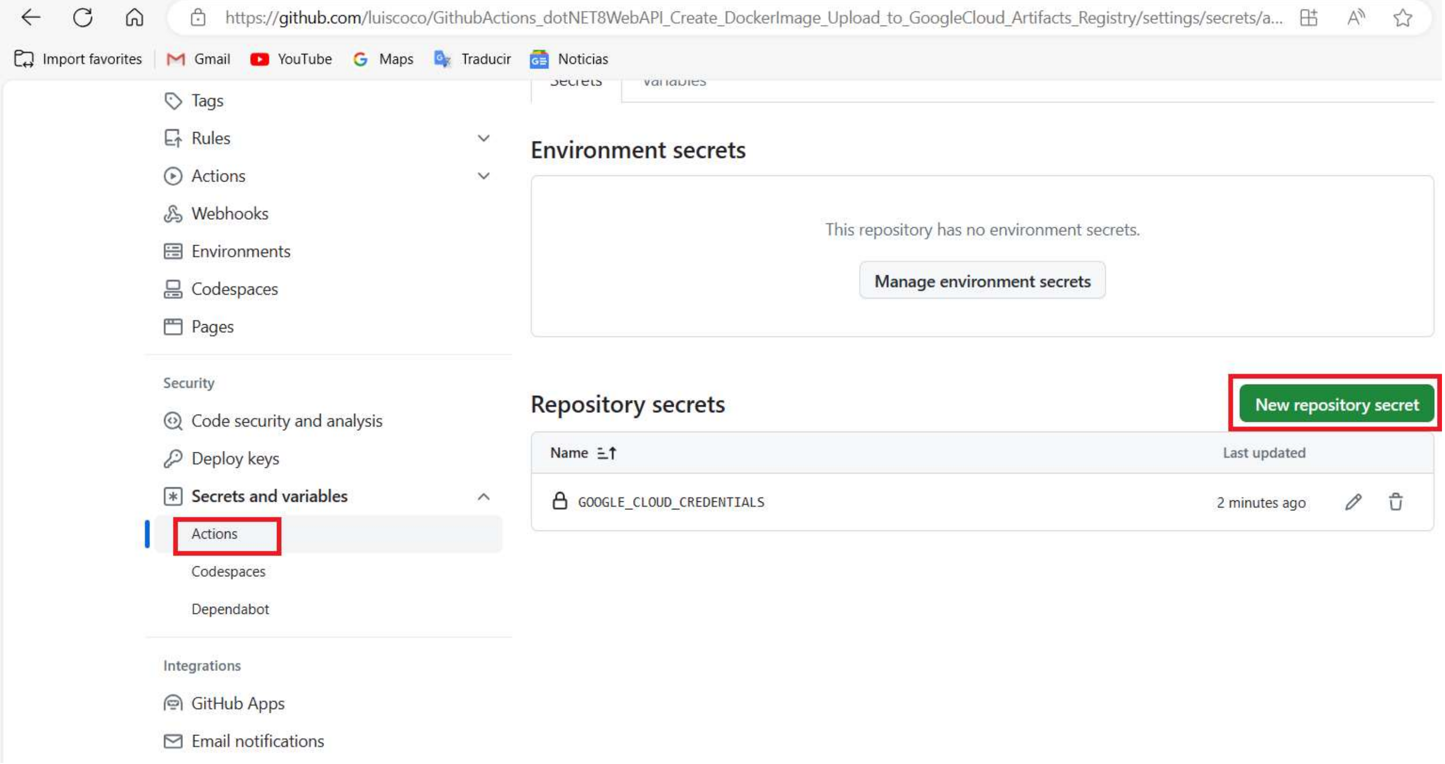
main

Social preview

Upload an image to customize your repository's social media preview.

Images should be at least 640×320px (1280×640px for best display).



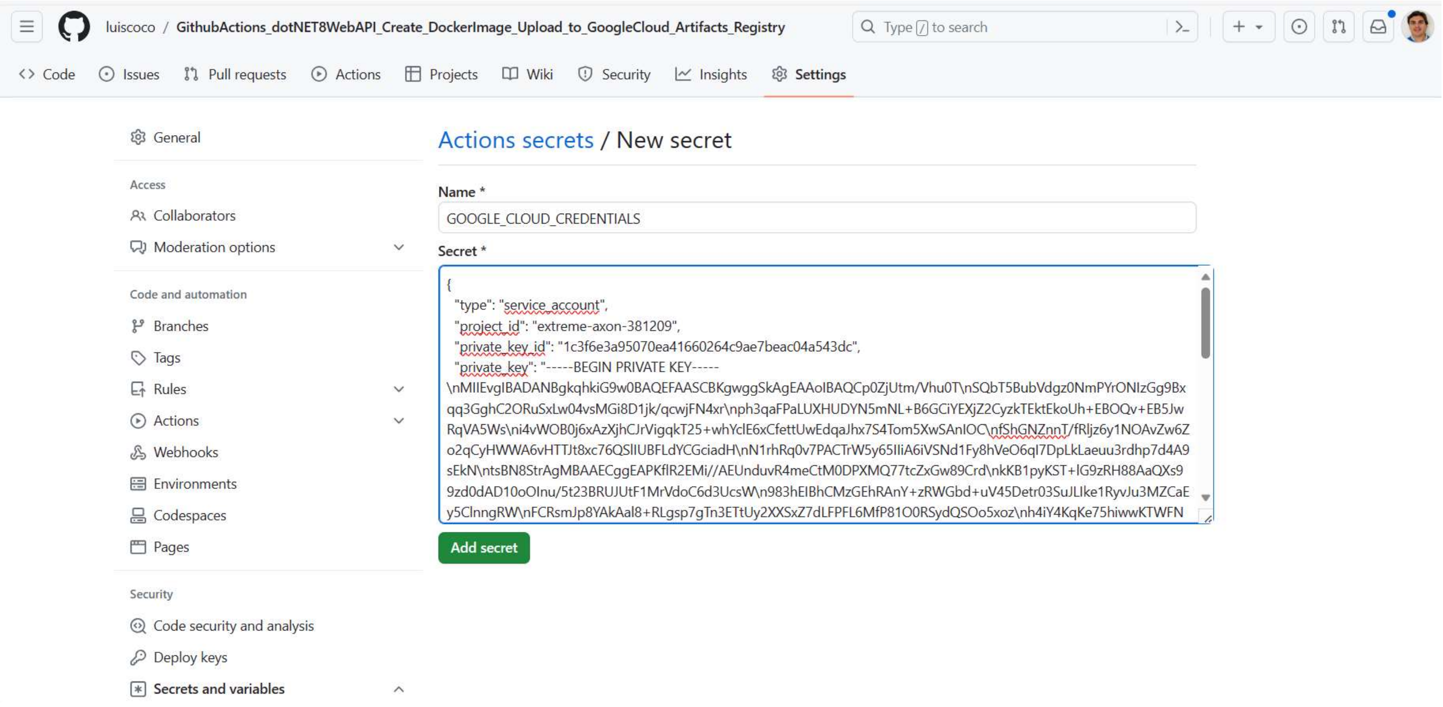


Environment secrets

This repository has no environment secrets.

Repository secrets

Name	Last updated
GOOGLE_CLOUD_CREDENTIALS	2 minutes ago



Actions secrets / New secret

Name *

GOOGLE_CLOUD_CREDENTIALS

Secret *

```
{
  "type": "service_account",
  "project_id": "extreme-axon-381209",
  "private_key_id": "1c3f6e3a95070ea41660264c9ae7beac04a543dc",
  "private_key": "-----BEGIN PRIVATE KEY-----
\nMIIEvgIBADANBgkqhkiG9w0BAQEFAASCBKgwggSkAgEAAoIBAQCp0ZjUtm/Vhu0T\nSQbT5BubVdgz0NmPYrONIzGg9Bx
qq3GghC2ORuSxLw04vsMGi8D1jk/qcwjFN4xr\nph3qaFPaLUXHUDYN5mNL+B6GCiYEXjZ2CyzkTEktEkoUh+EBOQv+EB5Jw
RqVA5Ws\ni4vWOB0j6xAzXjhCJrVigqkT25+whYclE6xCfettUwEdqaJhx7S4Tom5XwSAnIOC\nfShGNZnnT/fRljz6y1NOAvZw6Z
o2qCyHWWA6vHTTJt8xc76QSlIUBFLdYCGciadH\nN1rhRq0v7PACTrW5y65IIiA6iVSNd1Fy8hVeO6qI7DpLkLaeuu3rdhp7d4A9
sEkN\ntsBN8StrAgMBAAECggEAPKflR2EMi//AEUnduvR4meCtM0DPXMQ77tcZxGw89Crd\nkKB1pyKST+lG9zRH88AaQXs9
9zd0dAD10oOInu/5t23BRUJUtF1MrVdoC6d3UcsW\n983hEIBhCMzGEhRAnY+zRWGbd+uV45Detr03SuJLIke1RyvJu3MZCaE
y5ClnngRW\nFCRsmJp8YAkAal8+RLgsp7gTn3ETtUy2XXSxZ7dLFPFL6MfP81O0RSydQSOo5xoz\nh4iY4KqKe75hiwwKTWFN
```

Now, your GitHub Actions workflow can use this **secret** to authenticate with Google Cloud services. In your workflow file, you can reference this secret as ${{ secrets.GOOGLE_CLOUD_CREDENTIALS }}.

3. Create the main.yml file for Github actions workflow

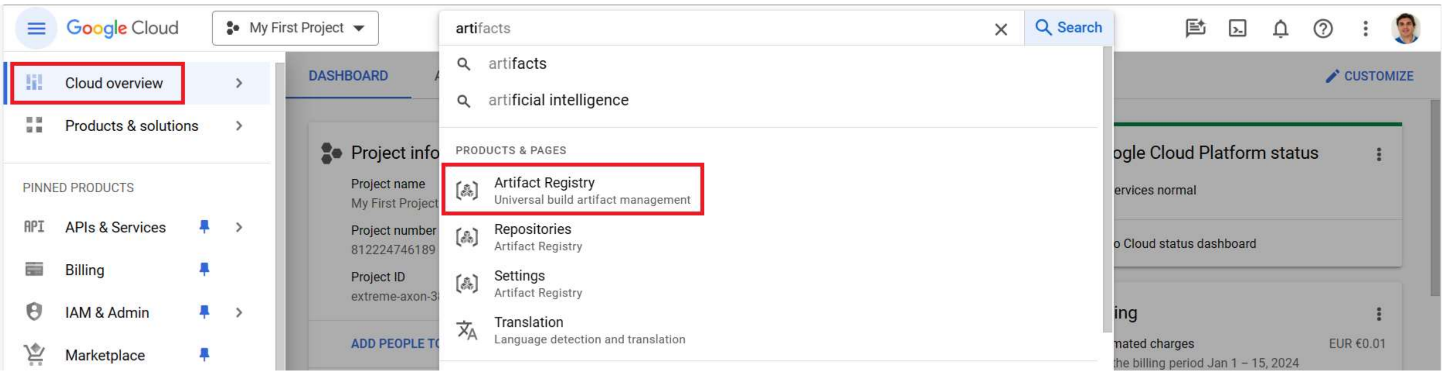
Below is the **main.yml** file tailored for your requirements.

This workflow assumes you have already set up Google Cloud credentials as secrets in your GitHub repository

```
name: Build and Push Docker Image

on:
  push:
    branches:
      - main

env:
  PROJECT_ID: extreme-axon-381209
  IMAGE_NAME: my-dotnetwebapi
  REPOSITORY: europe-southwest1-docker.pkg.dev/extreme-axon-381209/myfirstrepo
  TAG: latest

jobs:
  build-and-push:
    runs-on: ubuntu-latest

    steps:
    - name: Checkout code
      uses: actions/checkout@v4

    - name: Authenticate to Google Cloud
      uses: google-github-actions/auth@v2
      with:
        credentials_json: ${{ secrets.GOOGLE_CLOUD_CREDENTIALS }}

    - name: Configure Docker for Google Cloud Artifact Registry
      run: |
        echo '${{ secrets.GOOGLE_CLOUD_CREDENTIALS }}' | gcloud auth activate-service-account
        gcloud auth configure-docker europe-southwest1-docker.pkg.dev --quiet
    - name: Build Docker image
      run: |
        docker build -t ${{ env.REPOSITORY }}/${{ env.IMAGE_NAME }}:${{ env.TAG }} .
    - name: Push Docker image
      run: |
        docker push ${{ env.REPOSITORY }}/${{ env.IMAGE_NAME }}:${{ env.TAG }}
    - name: Verify the image was pushed
      run: |
        gcloud artifacts docker images list ${{ env.REPOSITORY }}/${{ env.IMAGE_NAME }}
```

4. Verify the Docker image uploaded to Google Cloud

Navigate to Google Cloud Artifacts Registry repo



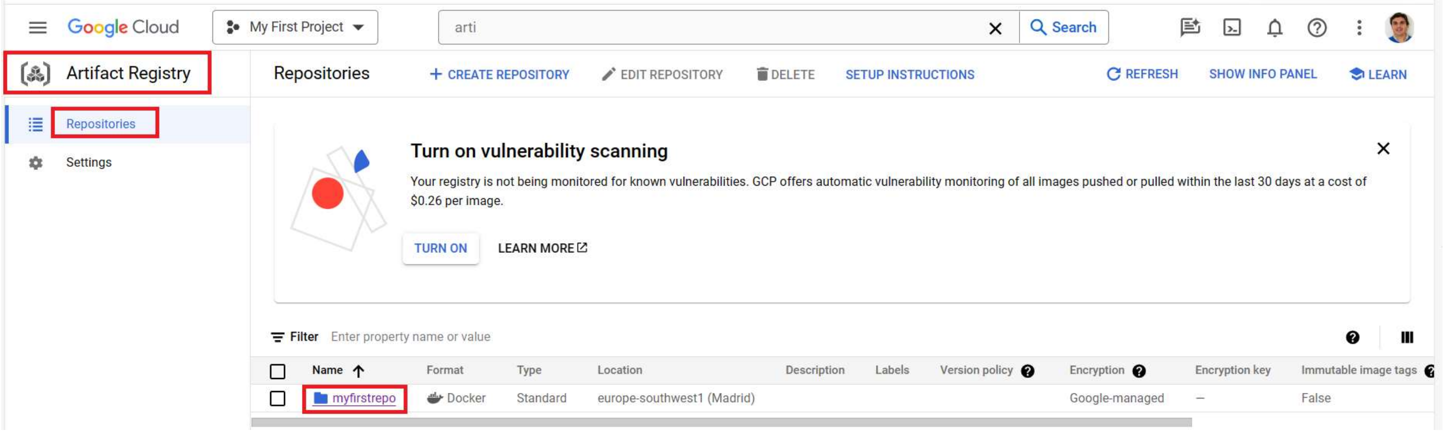
Project name
My First Project

Project number
812224746189

Project ID
extreme-axon-3

nated charges EUR €0.01
he billing period Jan 1 – 15, 2024

We can see inside the repo the uploaded Docker image

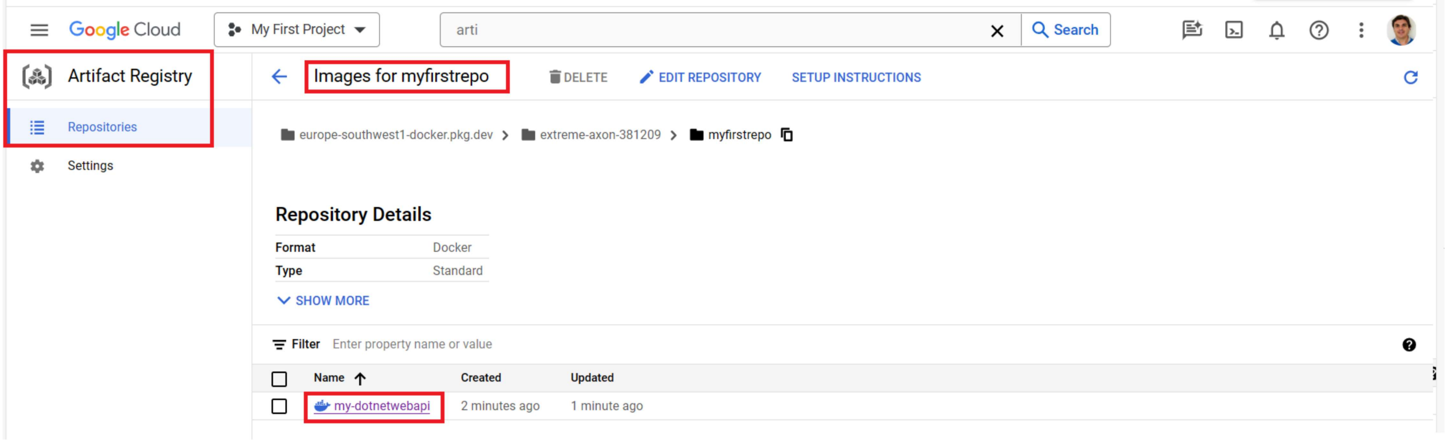


Turn on vulnerability scanning

Your registry is not being monitored for known vulnerabilities. GCP offers automatic vulnerability monitoring of all images pushed or pulled within the last 30 days at a cost of $0.26 per image.

Filter Enter property name or value

□	Name ↑	Format	Type	Location	Description	Labels	Version policy	Encryption	Encryption key	Immutable image tags
□	myfirstrepo	Docker	Standard	europe-southwest1 (Madrid)				Google-managed	–	False



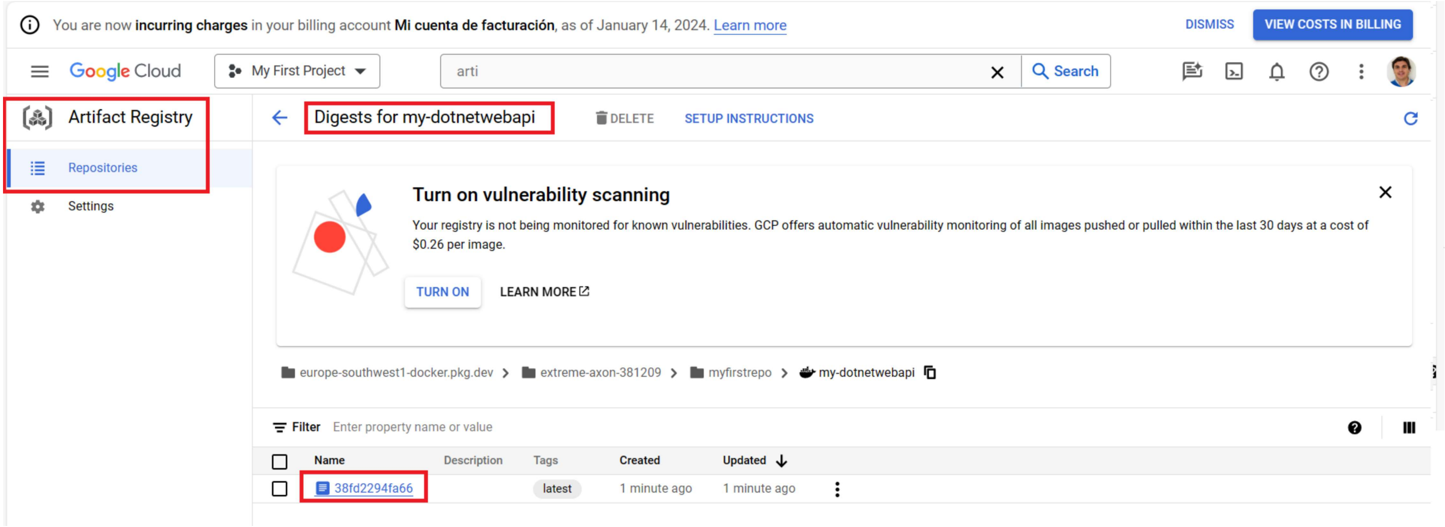
europe-southwest1-docker.pkg.dev > extreme-axon-381209 > myfirstrepo

Repository Details

Format	Docker
Type	Standard

Filter Enter property name or value

□	Name ↑	Created	Updated
□	my-dotnetwebapi	2 minutes ago	1 minute ago



You are now **incurring charges** in your billing account **Mi cuenta de facturación**, as of January 14, 2024. Learn more

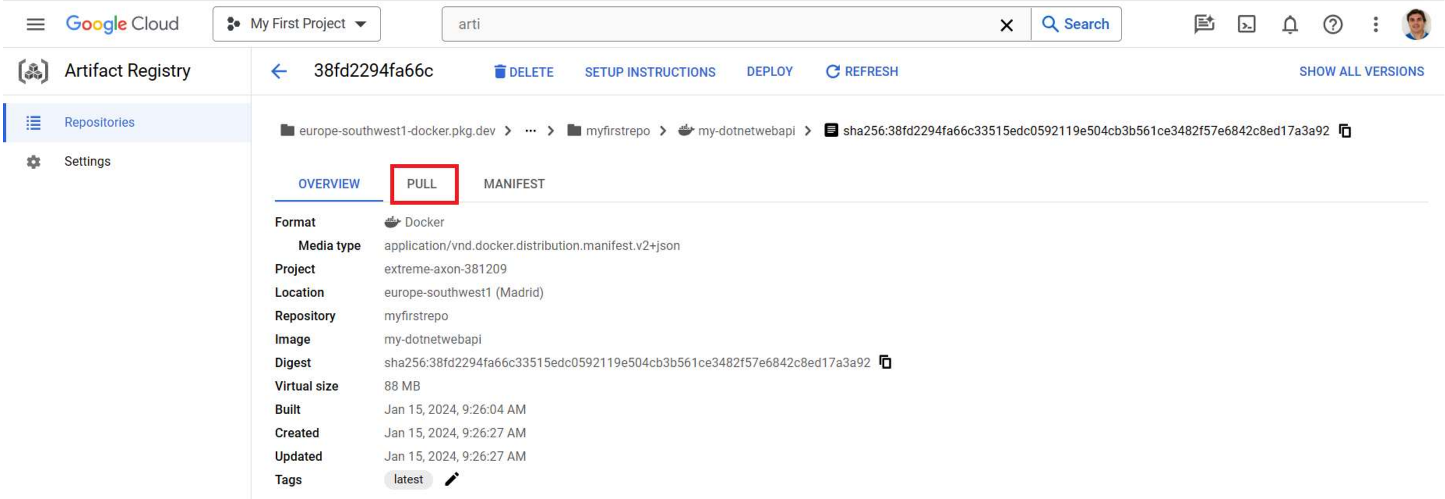
Turn on vulnerability scanning

Your registry is not being monitored for known vulnerabilities. GCP offers automatic vulnerability monitoring of all images pushed or pulled within the last 30 days at a cost of $0.26 per image.

europe-southwest1-docker.pkg.dev > extreme-axon-381209 > myfirstrepo > my-dotnetwebapi

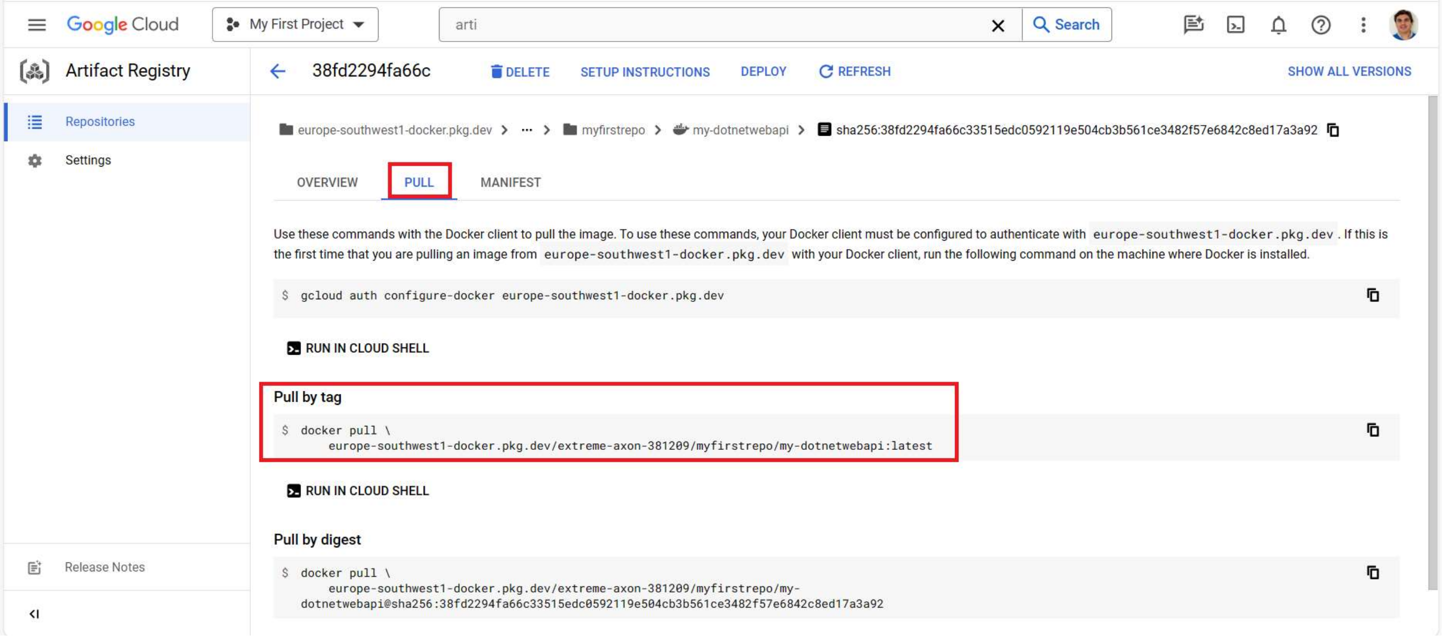
Filter Enter property name or value

□	Name	Description	Tags	Created	Updated ↓
□	38fd2294fa66		latest	1 minute ago	1 minute ago



europe-southwest1-docker.pkg.dev > ··· > myfirstrepo > my-dotnetwebapi > sha256:38fd2294fa66c33515edc0592119e504cb3b561ce3482f57e6842c8ed17a3a92

Format	Docker
Media type	application/vnd.docker.distribution.manifest.v2+json
Project	extreme-axon-381209
Location	europe-southwest1 (Madrid)
Repository	myfirstrepo
Image	my-dotnetwebapi
Digest	sha256:38fd2294fa66c33515edc0592119e504cb3b561ce3482f57e6842c8ed17a3a92
Virtual size	88 MB
Built	Jan 15, 2024, 9:26:04 AM
Created	Jan 15, 2024, 9:26:27 AM
Updated	Jan 15, 2024, 9:26:27 AM
Tags	latest



europe-southwest1-docker.pkg.dev > ··· > myfirstrepo > my-dotnetwebapi > sha256:38fd2294fa66c33515edc0592119e504cb3b561ce3482f57e6842c8ed17a3a92

Use these commands with the Docker client to pull the image. To use these commands, your Docker client must be configured to authenticate with `europe-southwest1-docker.pkg.dev`. If this is the first time that you are pulling an image from `europe-southwest1-docker.pkg.dev` with your Docker client, run the following command on the machine where Docker is installed.

```
$ gcloud auth configure-docker europe-southwest1-docker.pkg.dev
```

Pull by tag

```
$ docker pull \
    europe-southwest1-docker.pkg.dev/extreme-axon-381209/myfirstrepo/my-dotnetwebapi:latest
```

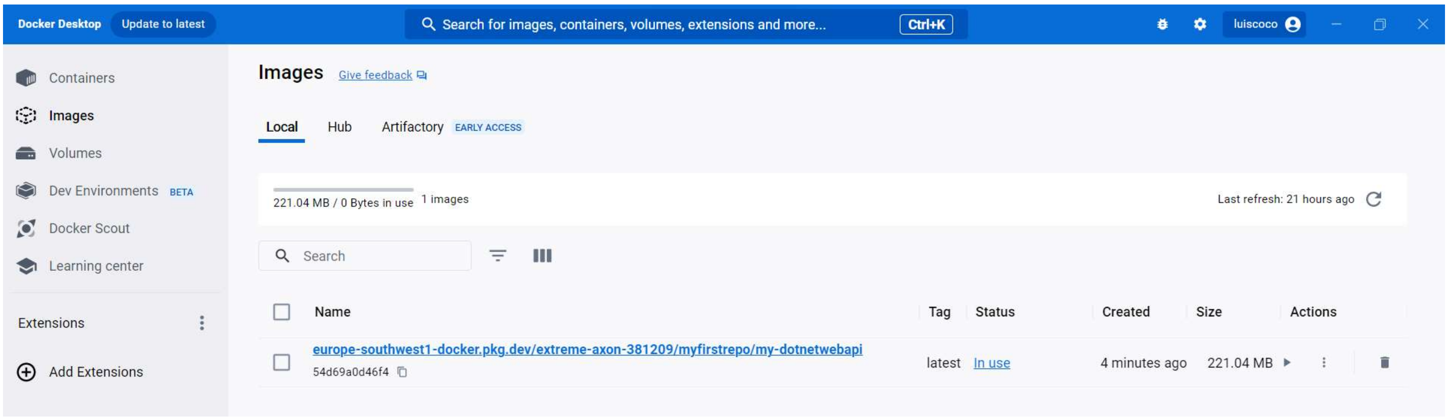
Pull by digest

```
$ docker pull \
    europe-southwest1-docker.pkg.dev/extreme-axon-381209/myfirstrepo/my-
dotnetwebapi@sha256:38fd2294fa66c33515edc0592119e504cb3b561ce3482f57e6842c8ed17a3a92
```

We run this command to pull the image and to

```
docker pull europe-southwest1-docker.pkg.dev/extreme-axon-381209/myfirstrepo/my-dotnetwebapi:l
```

We verified the downloaded image in Docker Desktop

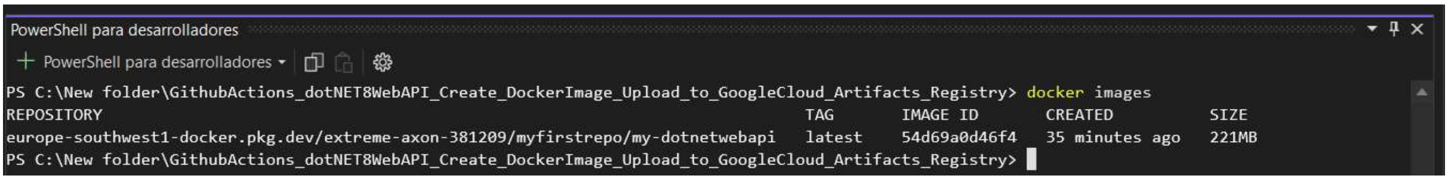


221.04 MB / 0 Bytes in use 1 images Last refresh: 21 hours ago

Name	Tag	Status	Created	Size	Actions
europe-southwest1-docker.pkg.dev/extreme-axon-381209/myfirstrepo/my-dotnetwebapi 54d69a0d46f4	latest	In use	4 minutes ago	221.04 MB	

Also we can see the image with the command

```
docker images
```


PowerShell para desarrolladores
PowerShell para desarrolladores

```
PS C:\New folder\GithubActions_dotNET8WebAPI_Create_DockerImage_Upload_to_GoogleCloud_Artifacts_Registry> docker images
REPOSITORY                                                              TAG       IMAGE ID       CREATED          SIZE
europe-southwest1-docker.pkg.dev/extreme-axon-381209/myfirstrepo/my-dotnetwebapi   latest    54d69a0d46f4   35 minutes ago   221MB
PS C:\New folder\GithubActions_dotNET8WebAPI_Create_DockerImage_Upload_to_GoogleCloud_Artifacts_Registry>
```

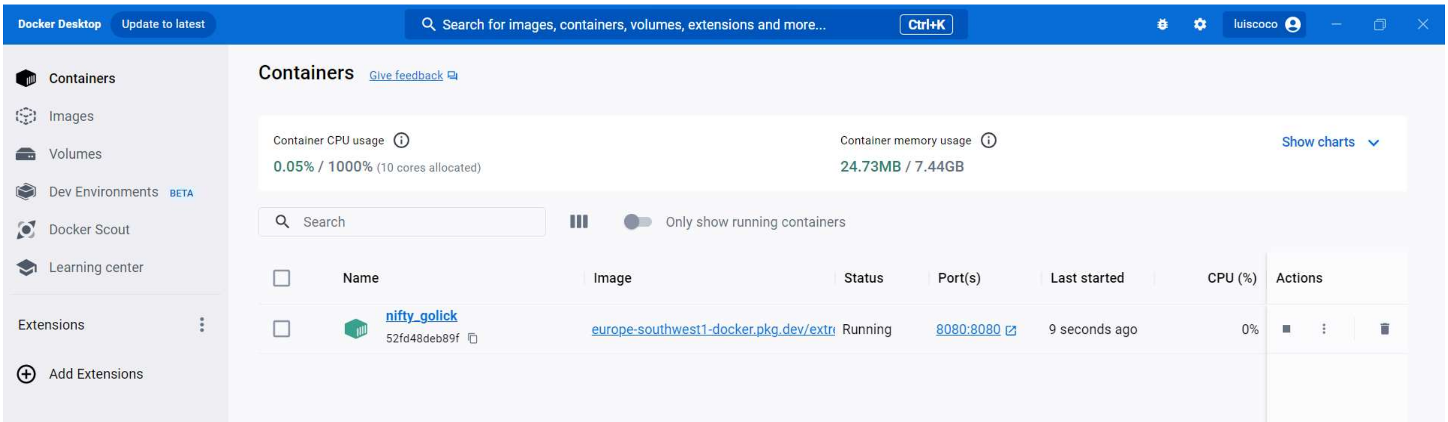
We run the image in our Docker Desktop

```
docker run -p 8080:8080 europe-southwest1-docker.pkg.dev/extreme-axon-381209/myfirstrepo/my-do
```

We see the running image with the command

```
docker ps
```

And also we can see the image in Docker Desktop



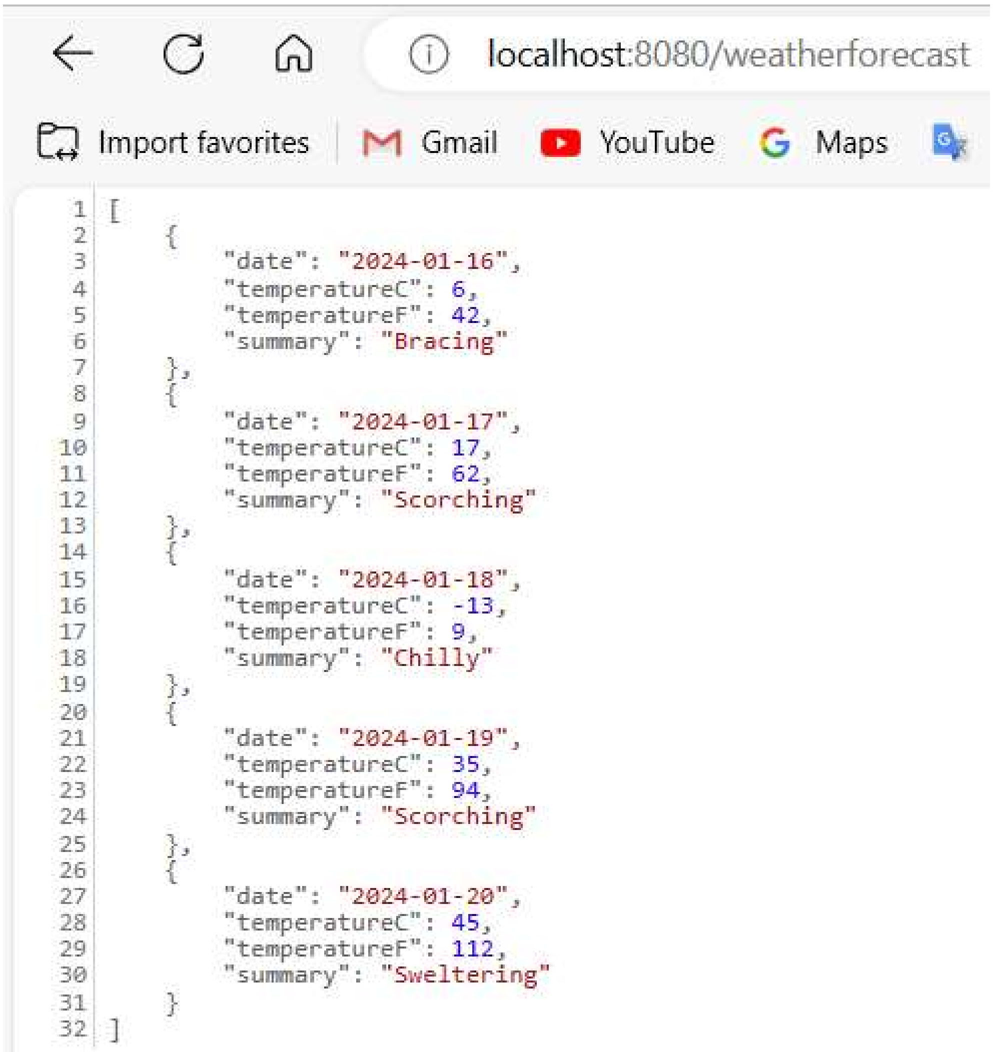
Containers

Container CPU usage 0.05% / 1000% (10 cores allocated)

Container memory usage 24.73MB / 7.44GB

□	Name	Image	Status	Port(s)	Last started	CPU (%)	Actions
□	nifty_golick 52fd48deb89f	europe-southwest1-docker.pkg.dev/extr	Running	8080:8080	9 seconds ago	0%	■ ⋮ 🗑

We can verify the running Docker container



```
[
    {
        "date": "2024-01-16",
        "temperatureC": 6,
        "temperatureF": 42,
        "summary": "Bracing"
    },
    {
        "date": "2024-01-17",
        "temperatureC": 17,
        "temperatureF": 62,
        "summary": "Scorching"
    },
    {
        "date": "2024-01-18",
        "temperatureC": -13,
        "temperatureF": 9,
        "summary": "Chilly"
    },
    {
        "date": "2024-01-19",
        "temperatureC": 35,
        "temperatureF": 94,
        "summary": "Scorching"
    },
    {
        "date": "2024-01-20",
        "temperatureC": 45,
        "temperatureF": 112,
        "summary": "Sweltering"
    }
]
```