

Cloud Run Lab

Lab Learning Outcomes:

At the end of this lab you should be able to:

- Create a serverless container service on the GCP
- Access that service using a secure HTTP endpoint
- Acknowledge how it works differently from Cloud Functions
- Recognise why you'd use this

Useful Links:

<https://www.youtube.com/watch?v=nhwYc4StHlc>

Prerequisites:

This lab assumes you've completed the previous labs for Enterprise Application Architecture. Therefore, you have a working valid Google Cloud Platform project. Familiarity with Docker from previous labs as well, so this assumes you have the Container Registry API enabled.

Step 1: Preparing the Container

We are going to be using a simple HTTP microservice to demonstrate the ability of Google Cloud Run.

The microservice is a simple PDF conversion using old Libreoffice Windows binaries, with a layer of Python and HTML on top of it. It is a perfect candidate to show off the capabilities of Google Cloud Run and a bit more exciting than a simple Hello World application, from a technological perspective.

Use the following command to clone the repository of the microservice:

```
1. git clone https://github.com/as-a-service/pdf
```

Then to go into the folder that we just cloned:

```
1. cd pdf
```

Now we need to build the container from the provided Dockerfile:

```
1. docker build .
```

As a reminder, this builds the Docker container image from the contents of the Dockerfile. Remember, the Dockerfile is a schematic of sorts, a template for what must be built. Once you run the command, you'll notice a lot of dependencies being downloaded and put into the image. You'll even notice some of the files ending with .exe... Remember, even though up until this part of the course we have only used typical Unix and Python programs for Docker.. Docker is capable of much, much more.

Once the container is built (this may genuinely take a while!) we need to tag it to push it to our Container Repository. Use the container ID from the end of the output, and the project ID. You can get this from going to the Google Cloud dashboard.

Run the following command:

```
1. docker tag <YOUR_CONTAINER_ID>  
   eu.gcr.io/<YOUR_PROJECT_ID>/cloudrunlab:latest
```

Then push the container to the repo:

```
1. docker push eu.gcr.io/<YOUR_PROJECT_ID>/cloudrunlab:latest
```

There we go, provided we haven't run into any errors, we are prepared for the rest of the lab! We should be prepared for the rest of the lab.

Step 2: Deploying to Cloud Run

Deploying the container image to Cloud Run is a simple enough process. It can be done through the Cloud Shell, but we will use the dashboard.

Open the navigation menu and go to the Compute section. You will find Cloud Run there. Click on it, and create a new service.

Google Cloud Platform EPA2020

Cloud Run Create service

1 Service settings

A service exposes a unique endpoint and automatically scales the underlying infrastructure to handle incoming requests. Deployment platform and service name cannot be changed.

Deployment platform

☒ Cloud Run (fully managed)

Region *
us-central1 (Iowa)

[How to pick a region?](#)

☐ Cloud Run for Anthos

Service name *

Service name

NEXT

You should see something like this. This is what we will use to configure and deploy our container to Cloud Run. We will pick europe-west2 (London) as it's the closest datacentre to Ireland, and for a lack of latency. Call the service cloudrundeploy.

Google Cloud Platform EPA2020 Search products and resources

Cloud Run Create service

2 Configure the service's first revision

A service can have multiple revisions. The configurations of each revision are immutable.

☒ Deploy one revision from an existing container image

Container image URL * SELECT

E.g. `us-docker.pkg.dev/cloudrun/container/hello`. Should listen for HTTP requests on PORT and not rely on local state. [How to build a container?](#)

☐ Continuously deploy new revisions from a source repository

Advanced settings

NEXT

3 Configure how this service is triggered

CANCEL

Select container image

CONTAINER REGISTRY ARTIFACT REGISTRY

Project: epa2020-290310 CHANGE

- Demo containers
 - eu.gcr.io/epa2020-290310/ca1
 - eu.gcr.io/epa2020-290310/careless-banking
 - eu.gcr.io/epa2020-290310/cloudrunlab
 - 7b08d14498 latest 15 minutes ago
 - us.gcr.io/epa2020-290310/gcf

SELECT CANCEL



After setting the location and name, you can see you can deploy the container from your Container Registry. Providing you successfully pushed the container to your Container Registry; you should see it when picking the single revision option. Although, you should take notice – Google Cloud Run has the potential for Continuous Deployment of serverless containers with Cloud Build, if a repo is provided. Leave the advanced settings as they are but look at them. You can change the number of vCPUs and memory, as well as the maximum amount of instances

Hit next – You will see HTTP settings.

3

Configure how this service is triggered

A service can be invoked via HTTP or Events. Click "Add Trigger" to create a new Event based trigger. [Learn more](#)

 HTTP 

Ingress BETA

☒ Allow all traffic

☐ Allow internal traffic and traffic from Cloud Load Balancing

☐ Allow internal traffic only

Authentication *

☒ Allow unauthenticated invocations

☐ Require authentication

Check this if you are creating a public API or website.

Manage authorized users with Cloud IAM.

+ ADD TRIGGER

CREATE

CANCEL

Make sure your settings match those in the image. Why? Well, we are only using this service as a test case and not deploying this in an internal network. We also need to see the output of the container in an internet browser. Hit create to deploy the container. You should be able to see it created successfully, after a moment.

Cloud Run

Service details



EDIT & DEPLOY NEW REVISION

SET UP CONTINUOUS DEPLOYMENT

cloudrundeploy

Region: europe-west2

URL [https://cloudrundeploy-00001-gan.run.app](#)



METRICS

REVISIONS

LOGS

TRIGGERS

DETAILS

YAML

PERMISSIONS

Revisions

MANAGE TRAFFIC

Filter revisions


Name


Traffic

Deployed

Revision URL(s) (tags)

Actions





cloudrundeploy-00001-gan

100% (to latest)

Just now

cloudrundeploy-00001-gan

Deployed by ryandeering1@gmail.com using Cloud Console

CONTAINER

VARIABLES

CONNECTIONS

YAML

General

Image URL

[eu.gcr.io/epac2020-290310/cloudrunlab@sha256:b87...](#)

Build

(no build information available)

Source

(no source information available)

Port

8080

Command and args

(container endpoint)

Service account

[668115442556-compute@developer.gserviceaccount.com](#)

Capacity

CPU allocated

1

Memory allocated

256Mi

Concurrency

80

Request timeout

300 seconds

Autoscaling

Max instances

100

After successfully creating the image, you should see the URL for the service created. Click on it and you should see this:

Convert .doc to .pdf

This service converts Word documents to PDF documents.

Upload your file

Choose File

No file chosen

Upload

Pro tip: You can also provide the URL of a document to convert with the `url` query parameter.
For example `?url=http://homepages.inf.ed.ac.uk/neilly/TestWordDoc.doc`
Made by [@stereon](#) - Sources on [GitHub](#)

If you see it, congratulations, you just deployed a serverless container image to a secure HTTPS endpoint. Try using it by uploading this lab document to it, it will convert to a PDF.

That's as simple as deploying to Google Run gets!

Step 3: Tearing Down

To stop the service from running more, go into the Cloud Run menu, select your service and delete it. This will stop it from running and incurring any more potential costs.

The screenshot shows the Google Cloud Run 'Services' page. At the top, there are tabs for 'Cloud Run' and 'Services', along with buttons for 'CREATE SERVICE', 'MANAGE CUSTOM DOMAINS', 'COPY', and 'DELETE'. Below the tabs, a message states: 'Each Cloud Run service has a unique endpoint and autoscales deployed containers. [Learn more](#)'. A 'Filter services' section is present above a table of services.

| <input checked="" type="checkbox"/> | <input type="radio"/> | Name ↑ | Req/sec ? | Region | Authentication ? | Ingress ? | Last deployed | Deployed by |
|-------------------------------------|----------------------------------|--------------------------------|-----------|--------------|-----------------------|-----------|---------------------------|------------------------|
| <input checked="" type="checkbox"/> | <input checked="" type="radio"/> | cloudrundeploy | 0 | europe-west2 | Allow unauthenticated | All | Dec 19, 2020, 11:38:44 PM | ryandeering1@gmail.com |

A confirmation dialog box is displayed in the bottom right corner with the text 'Delete cloudrundeploy?' and two buttons: 'CANCEL' and 'DELETE'.