



Big Data Processing

Keepcoding Bootcamp

Andrés B. Aldaz

October 27, 2022

Contents

1	Introduction	2
1.1	Google Compute Instance Setup Guide	2
1.2	Execute Apache Kafka	3
1.3	Create Systemd Unit Files for Kafka and Zookeeper	4
1.4	CMAK	4
2	Attachments	5
2.1	Google Cloud Console Code Configuration	5
2.2	Docker Installation Confirmation	5
2.3	Zookeeper.service	6
2.4	Kafka.service	6

1 Introduction

1.1 Google Compute Instance Setup Guide

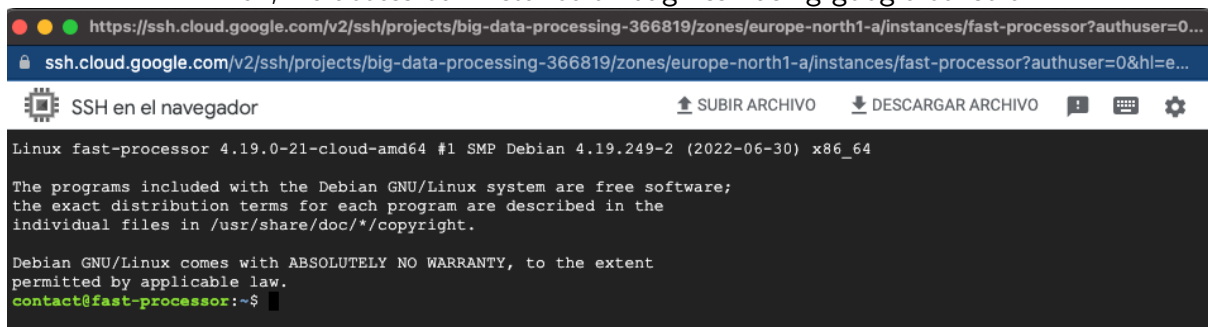
1. Access [Google Compute Engine](#) Console
2. Create new Project: **"big-data-processing"**
3. Enable Compute Engine API
 - (a) Instance Name: **fast-processor**
 - (b) Zone: Europe-north1(Finland)¹
 - (c) e2-standard-4
 - (d) Debian GNU/Linux 10: **100GB Persistent Disk**
 - (e) Enable HTTP & HTTPS connections

Figure 1: Created Instance

<input type="checkbox"/>	Estado	Nombre ↑	Zona	Recomendaciones	En uso por	IP interna	IP externa	Conectar
<input type="checkbox"/>	✓	fast-processor	europe-north1-a			10.166.0.2 (nic0)	35.228.45.255 (nic0)	SSH ▾ ⋮

Figure 2: SSH Environment

Then, we access our instance through ssh using google console



```
https://ssh.cloud.google.com/v2/ssh/projects/big-data-processing-366819/zones/europe-north1-a/instances/fast-processor?authuser=0...
ssh.cloud.google.com/v2/ssh/projects/big-data-processing-366819/zones/europe-north1-a/instances/fast-processor?authuser=0&hl=e...
SSH en el navegador SUBIR ARCHIVO DESCARGAR ARCHIVO ! 🗂 ⚙
Linux fast-processor 4.19.0-21-cloud-amd64 #1 SMP Debian 4.19.249-2 (2022-06-30) x86_64
The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
contact@fast-processor:~$
```

4. Configuration Debian Instance

- (a) Enable root operations
 - `sudo -i`
- (b) Update repositories
 - `apt-get update -y`
- (c) Install Java® and **wget**
 - `apt-get install default-jre wget -y`
- (d) Check Java® version
 - `java -version` ²

¹Personal preference since I live in Sweden.

²Running JDK 11.0.16

(e) Add user "kafka"

- `adduser kafka`
- User Name: daibeal
- Password: 3637

(f) Add created user to sudo group

- `adduser kafka sudo`

(g) Log in as a Kafka user and download the latest version of Apache Kafka

- `su - kafka`
- `wget https://archive.apache.org/dist/kafka/2.7.2/kafka-2.7.2-src.tgz`

(h) Extract the downloaded file

- `tar -xvzf kafka-2.7.2-src.tgz`
- `mv kafka-2.7.2-src kafka`

(i) Install docker

- `sudo apt-get update`
- Add Docker Official GPG key
 - `sudo mkdir -p /etc/apt/keyrings`
 - `curl -fsSL https://download.docker.com/linux/debian/gpg | sudo gpg --`
- Set up the repository

(j) Docker Engine

- `sudo apt-get install docker-ce docker-ce-cli containerd.io docker-compose`
- Verify installation
 - `sudo docker run hello-world`

(k) Exit kafka user

- `exit`

1.2 Execute Apache Kafka

▪ Access kafka

- `cd kafka`
- Install Gradle
 - * `./gradlew jar -PscalaVersion=2.13.3`
- Set proper ownership
 - * `chown -R kafka:kafka /home/kafka/kafka`

2 Attachments

2.1 Google Cloud Console Code Configuration

```
gcloud compute instances create fast-processor
--project=big-data-processing-366819
--zone=europe-north1-a
--machine-type=e2-standard-4
--network-interface=network-tier=PREMIUM,subnet=default
--maintenance-policy=MIGRATE --provisioning-model=STANDARD
--service-account=906672027712-compute@developer.gserviceaccount.com
--scopes=https://www.googleapis.com/auth/devstorage.read_only,https://www.googleapis.com/auth/monitoring.write,https://www.googleapis.com/auth/trace.append --tags=http-server,https-s
image=projects/debian-cloud/global/images/debian-10-buster-v20220920,mo
--no-shielded-secure-boot --shielded-vtpm --shielded-integrity-monitori
```

2.2 Docker Installation Confirmation

```
kafka@fast-processor:~$ sudo docker run hello-world
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
2db29710123e: Pull complete
Digest: sha256:e18f0a777aefabe047a671ab3ec3eed05414477c951ab1a6f352a069
Status: Downloaded newer image for hello-world:latest
```

Hello from Docker!

This message shows that your installation appears to be working correct

To generate this message, Docker took the following steps:

1. The Docker client contacted the Docker daemon.
2. The Docker daemon pulled the "hello-world" image from the Docker Hub (amd64)
3. The Docker daemon created a new container from that image which run executable that produces the output you are currently reading.
4. The Docker daemon streamed that output to the Docker client, which to your terminal.

To try something more ambitious, you can run an Ubuntu container with:

```
$ docker run -it ubuntu bash
```

Share images, automate workflows, and more with a free Docker ID:

<https://hub.docker.com/>

For more examples and ideas, visit:

`https://docs.docker.com/get-started/`

2.3 Zookeeper.service

```
[Unit]
Requires=network.target remote-fs.target
After=network.target remote-fs.target

[Service]
Type=simple
User=kafka
ExecStart=/home/kafka/kafka/bin/zookeeper-server-start.sh /home/kafka/k
ExecStop=/home/kafka/kafka/bin/zookeeper-server-stop.sh
Restart=on-abnormal

[Install]
WantedBy=multi-user.target
```

2.4 Kafka.service

```
[Unit]
Requires=zookeeper.service
After=zookeeper.service

[Service]
Type=simple
User=kafka
ExecStart=/bin/sh -c '/home/kafka/kafka/bin/kafka-server-start.sh /home
ExecStop=/home/kafka/kafka/bin/kafka-server-stop.sh
Restart=on-abnormal

[Install]
WantedBy=multi-user.target
```