Deploy Hortonworks Sandbox on Google Cloud:

The following steps need to be done only for the first time:

Open this PDF with Chrome or Adobe Reader - do not use Apple Preview.

1. Login to your Google Cloud account (https://cloud.google.com) and open a Google Cloud Shell. Make sure you are in the intended project.
2. Run the following script on **Google Cloud Shell** to launch a new instance named ambari. Copy this command to an editor of your choice and make sure to replace <PROJECT-ID> with your project-id (it can be found by clicking on the project name, under ID):

```
gcloud compute --project=<PROJECT-ID> instances create ambari \
--zone=us-east1-b --machine-type=e2-highmem-4 \
--subnet=default --network-tier=PREMIUM --no-restart-on-failure \
--maintenance-policy=MIGRATE --image=cos-77-12371-1109-0 \
--image-project=cos-cloud --boot-disk-size=50GB \
--boot-disk-type=pd-balanced --boot-disk-device-name=ambari
```

   - This will give you a machine with 50 GB HHD and 32GB Memory (e2-highmem-4) and cost about ¢18/hour (as of Sep 2024).
3. Once the instance is ready (External IP address will be populated), you should see the green checkmark next to its name under the Compute Engine section. From there, you can click on the SSH button to SSH into your machine.
   - For the first time, it may take a while until the SSH button works. If that is the case from Cloud Shell run this command:

```
gcloud compute --project "< PROJECT-ID >" ssh --zone "us-east1-b" "ambari"
```

4. Log in as the root user: `sudo su - root`
5. From the **SSH window** download the HDP2.5 image into your instance (~1 minute):
   - `wget https://storage.googleapis.com/is843/HDP_2.5_docker.tar.gz -P /tmp/`
6. Load this image to docker (~ 2 minutes):

```
docker load < /tmp/HDP_2.5_docker.tar.gz
```

7. Start this docker image by running the following command:

```
docker run -v hadoop:/hadoop --name sandbox --hostname \
"sandbox.hortonworks.com" --privileged -d -p 6080:6080 -p 9090:9090 \
-p 9000:9000  -p 8000:8000 -p 8020:8020 -p 42111:42111 -p 10500:10500 \
-p 16030:16030 -p 8042:8042 -p 8040:8040 -p 2100:2100 -p 4200:4200 \
-p 4040:4040 -p 8050:8050 -p 9996:9996 -p 9995:9995 -p 8080:8080 -p 8088:8088 \
-p 8886:8886 -p 8889:8889 -p 8443:8443 -p 8744:8744 -p 8888:8888 -p 8188:8188 \
-p 8983:8983 -p 1000:1000 -p 1100:1100 -p 11000:11000 -p 10001:10001 \
-p 15000:15000 -p 10000:10000 -p 8993:8993 -p 1988:1988 -p 5007:5007 \
-p 50070:50070 -p 19888:19888 -p 16010:16010 -p 50111:50111 -p 50075:50075 \
-p 50095:50095 -p 18080:18080 -p 60000:60000 -p 8090:8090 -p 8091:8091 \
-p 8005:8005 -p 8086:8086 -p 8082:8082 -p 60080:60080 -p 8765:8765 \
-p 5011:5011 -p 6001:6001 -p 6003:6003 -p 6008:6008 -p 1220:1220 \
-p 21000:21000 -p 6188:6188 -p 61888:61888 -p 2181:2181 -p 2222:22 \
sandbox /usr/sbin/sshd -D
```

   - The container should be running now. You can confirm this by `docker ps`; it should return a non-empty list of running containers.

- If you are able to see a new docker image called sandbox running you can remove the zip file that we downloaded earlier to open up 11GB space in our disk:

`rm /tmp/HDP_2.5_docker.tar.gz`

8. Start sandbox services, Ambari and HDP services do not start automatically when you start the Docker container. You need to start the processes with a script:
   - Let's SSH into the "sandbox" docker we just created:
     `ssh -p 2222 root@localhost`
     - If it's your first time you will be asked to change the password. The password for the root is "hadoop". Remember it and write it down somewhere.
   - Start the services (~ 2 minutes):

     `/etc/init.d/startup_script start`
9. While you are here let's set a password for the Ambari "admin" account:
   `ambari-admin-password-reset`
10. When you see all the services are OK and have been started, we are done with the Ambari setup. If you are using the SSH window, you can close it now. If you were using Cloud Shell, type `exit` three times to get to the Cloud Shell home page.
11. From **Cloud Shell** create a new firewall rule that allows INGRESS tcp:8080 with VMs containing the tag "allow-tcp-8080":

```
gcloud compute firewall-rules create rule-allow-tcp-8080 \
--source-ranges 0.0.0.0/0 --target-tags allow-tcp-8080 --allow tcp:8080
```

    - This needs to be done once per project. If you have done it before, you can ignore it.
12. Set the zone that the instance is located at
    `gcloud config set compute/zone us-east1-b`
13. Add the "allow-tcp-8080" tag to your new VM:
    `gcloud compute instances add-tags ambari --tags allow-tcp-8080`

Important note (optional): Last three steps (11-13) make our instance accessible by exposing its port 8080. This is not a secure way to connect to a cloud instance/cluster and was mentioned here for convenience. One secure way to do this is to set up an SSH tunnel:

- First, Google Cloud SDK has to be installed and initialized: https://cloud.google.com/sdk/docs/downloads-interactive
- If you are a Windows user and the above step didn't help you can find more info here: https://cloud.google.com/sdk/docs/quickstart-windows
- Use this command to forward port 8080 and access the instance securely: `gcloud compute ssh ambari --zone us-east1-b -- -L 8080:localhost:8080`
- From the browser go to `localhost:8080`

14. From the Google Compute Engine page, write down the external IP address of instance "ambari". With this, you can log in to Ambari in your local browser by:
    - <External-IP-Address>:8080
    - Use maria_dev for username and password!
15. IMPORTANT: Stop the instance at the end of the session. You are paying ¢18/hour to use this instance, and if you forget to stop it at the end of each session, your $50 credit will last for a little over a week.

Restarting the Stopped Instance

1. From Compute Engine, select your "ambari" instance, and from the top menu, click Start.
2. SSH into the machine (step 3 from deployment instructions) and log in as root
    a. sudo su - root
3. Run the following so your container starts again
    docker restart sandbox
4. Start sandbox services, Ambari and HDP services do not start automatically when you start the Docker container. You need to start the processes with a script:
    a. SSH into the docker container and use your "root" password that you set while setting up this instance:
    ssh -p 2222 root@localhost
    b. Start the startup script:
    /etc/init.d/startup_script start
5. Because we stopped the instance the IP address has changed. Look up the new external IP address from Compute Engine and log in to Ambari from your local browser:
    a. <External-IP-Address>:8080

Resetting password for maria_dev

1. Log in to Ambari with the "admin" username and the password you set in the previous step.
2. Go to Manage Ambari > Users
3. Reset the password for maria_dev to something secure.

It is recommended that the other users be deactivated or their passwords reset.

Clean up

To make sure we won't be charged for any of the resources delete the instance:

- From the Compute Engine page select the instance and click on the DELETE button.

Note: You won't be charged for the instance while it's stopped. But you will get charged for the disk, which is about $4/month for our 100GB configuration.