Create VM template and Automate deployment

**Introduction**

You're an IT Administrator for your company and you're assigned to work on a project that requires you to deploy eight virtual machines (VMs) as web servers. Each of them should have the same configuration. You'll create a VM, set up an auto-enabled service, and make it a template. Then you'll use the template to create seven more VMs.

What you'll do

* Create a VM using GCP web UI and make a template out of it
* Use a command-line interface to interact with VMs
* Learn how to configure an auto-enabled service
* Learn to use gcloud to deploy VMs with a template

**Setup**

What you need

To complete this lab, you need:

* Access to a standard internet browser (the Chrome browser is recommended)
* Time to complete the lab

**Note:** If you already have your own personal GCP account or project, please don't use it for this lab.

In this lab, you will be using **gcloud command-line interface**, which is a tool that provides the primary CLI to Google Cloud Platform, to interact with VMs. To use this, you should install the Google Cloud SDK, initialize it, and run core gcloud commands from the command line on your local computer.

To install Google Cloud SDK follow the instructions given below based on your device's operating system:

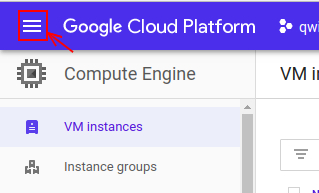
* [Windows](https://cloud.google.com/sdk/docs/install-sdk#windows)
* [Linux](https://cloud.google.com/sdk/docs/install-sdk#linux)
* [Debian and Ubuntu](https://cloud.google.com/sdk/docs/install-sdk#deb)
* [Red Hat and Cento](https://cloud.google.com/sdk/docs/install-sdk#rpm)
* [macOS](https://cloud.google.com/sdk/docs/install-sdk#mac)

lab.

**Create a VM instance from the Cloud Console**

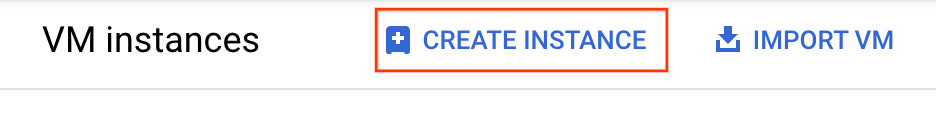
In this section, you'll learn how to create new, predefined machine types with Google Compute Engine from the Cloud Console.

In the GCP Console, on the top left of the screen, select **Navigation menu** > **Compute Engine** > **VM instances**:



This may take a moment to initialize for the first time.

To create a new instance, click **Create instance**.

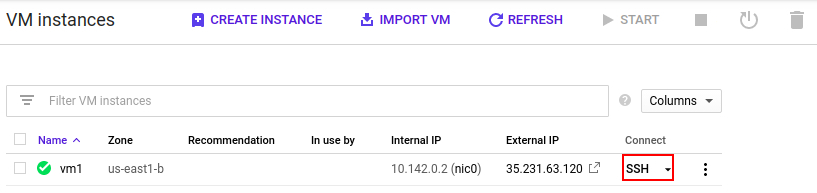


There are lots of parameters you can configure when creating a new instance. Use the following for this lab:

|  |  |  |
| --- | --- | --- |
| **Field** | **Value** | **Additional Information** |
| Name | vm1 | Name for the VM instance |
| Region | us-west4 | Learn more about regions in [Regions & Zones documentation](https://cloud.google.com/compute/docs/zones). |
| Zone | us-west4-a | Learn more about regions in [Regions & Zones documentation](https://cloud.google.com/compute/docs/zones). |
| Series | E2 | The E2 machine series is Compute Engine's second generation general-purpose machine series. |
| Machine Type | e2-medium | **Note**: A new project has a default [resource quota](https://cloud.google.com/compute/docs/resource-quotas), which may limit the number of CPU cores. You can request more when you work on projects outside of this lab. |
| Boot Disk | Ubuntu 22.04 LTS | Click on the **change** button, click on the **Operating system** and select Ubuntu then for version, select Ubuntu 22.04 LTS. Learn more about boot disk check out this [link](https://wiki.ubuntu.com/BionicBeaver/ReleaseNotes). |
| Boot disk type | standard persistent disk | Learn more about standard persistent disk check out this [link.](https://cloud.google.com/compute/docs/disks/) |
| Firewall | allow HTTP and HTTPS traffic | Learn more about firewall check out this [link.](https://cloud.google.com/vpc/docs/firewalls) |

Leave all the other configurations set to their defaults.

After entering the above parameters, click on the **Create** button to create your VM.



SSH into vm1 by clicking on the SSH button, as shown in the image above.

Git clone

Use Git to clone the repository by using the following command:

git clone https://www.github.com/google/it-cert-automation-practice.git

Output:



File operation

Once you have the repository successfully cloned, navigate to the Lab3/directory.

cd ~/it-cert-automation-practice/Course5/Lab3

To list the files in the working directory Lab3/ use the **list** command.

ls

Output:



In order to enable hello\_cloud.py to run on boot, copy the file hello\_cloud.py to the /usr/local/bin/ location.

sudo cp hello\_cloud.py /usr/local/bin/

Also copy hello\_cloud.service to the /etc/systemd/system/ location.

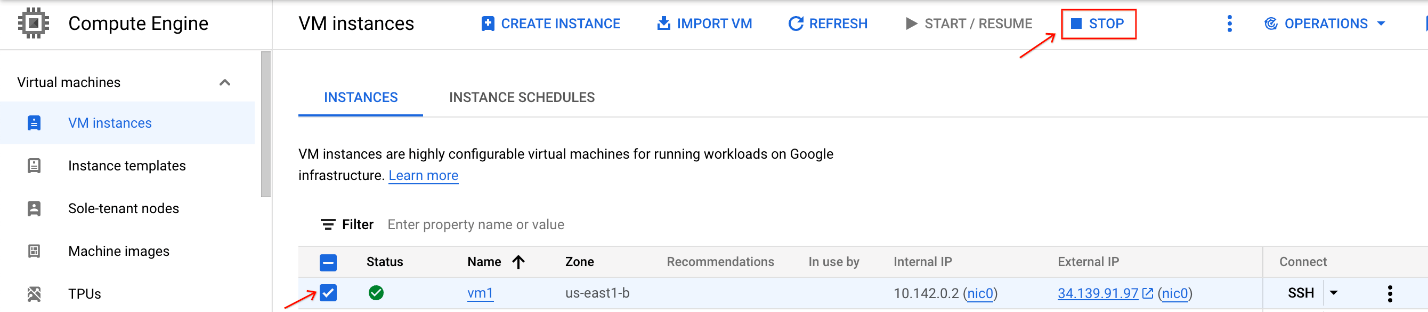
sudo cp hello\_cloud.service /etc/systemd/system

Now, use the systemctl command to enable the service **hello\_cloud**.

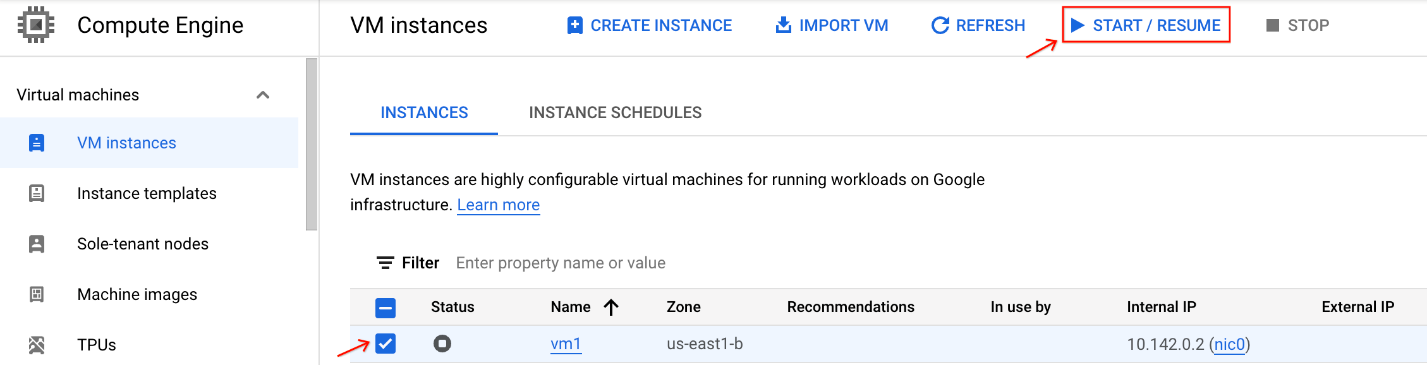
sudo systemctl enable hello\_cloud.service

Restart the VM

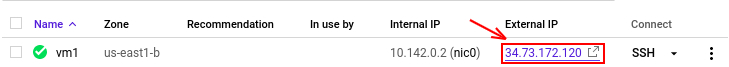
After enabling the **hello\_cloud** service, reboot the VM to ensure that the service is up. To reboot the VM instance **vm1** go to the **Compute Engine** > **VM instance** and stop the VM instance vm1 by selecting the VM instance vm1 and clicking on the **Stop** button at the top. Again, click on the **Stop** button in the popup.



The start method restarts an instance in a TERMINATED state. To start the VM instance **vm1**, select it first by tick marking it, then click on the **Start/Resume** button at the top. Again, click on the **Start** button in the popup. You can this in the image below.

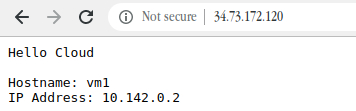


After restarting the VM instance **vm1,** visit the External IP link of the vm1 that's shown in the image below:



**Note:**If you are getting any error then click on the url and use http://EXTERNAL-IP.

Output:



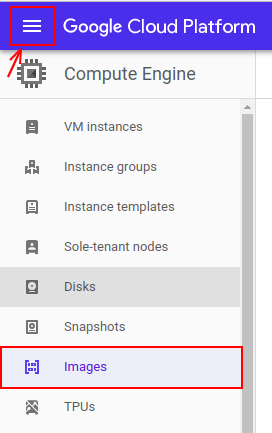
**Create VMs using a template**

You'll now create a template for vm1.

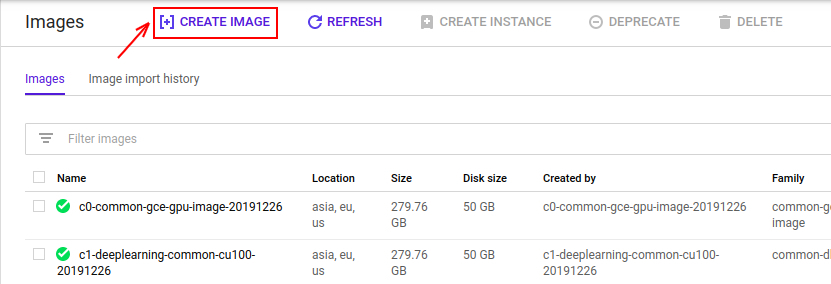
First, **shut down** the VM instance **vm1** by going to the **Compute Engine** > **VM instance**, selecting the VM instance vm1, and clicking on the **stop** button at the top.

Now, create an image named vm-image based on the **vm1** disk by following the steps below:

In the GCP Console, on the top left of the screen, select **Navigation menu > Compute Engine > Images:**



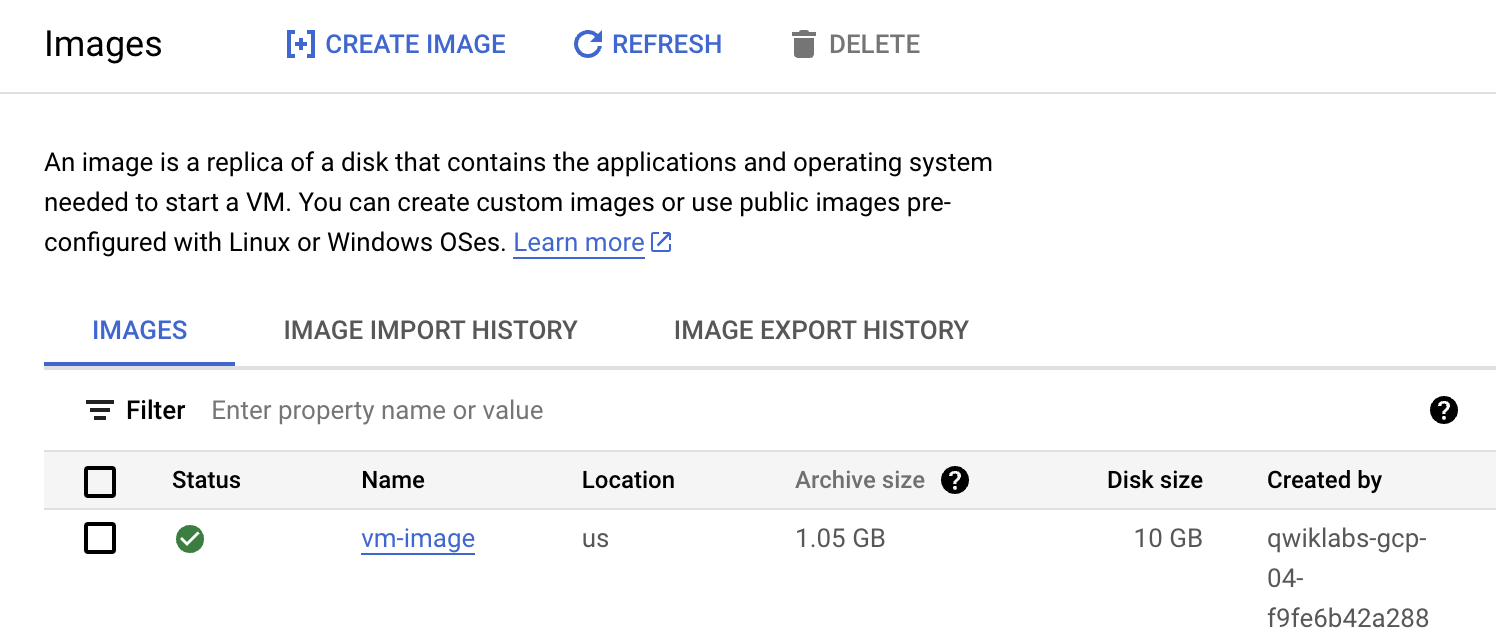
Click on the **CREATE IMAGE** button below.



Then, create an image based on the **vm1's** disk, using the following parameters:

|  |  |
| --- | --- |
| **Field** | **Value** |
| Name | vm-image |
| Source | Disk |
| Source Disk | vm1 |

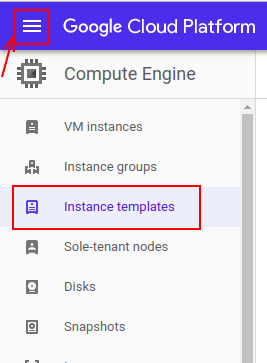
Leave all of the other values set to their default settings. Click on the **create** button to create your image.



Now, create an instance template using vm-image for the boot disk you just created.

To create a instance template, follow the instructions below:

In the GCP Console, on the top left of the screen, select **Navigation menu > Compute Engine > Instance templates:**

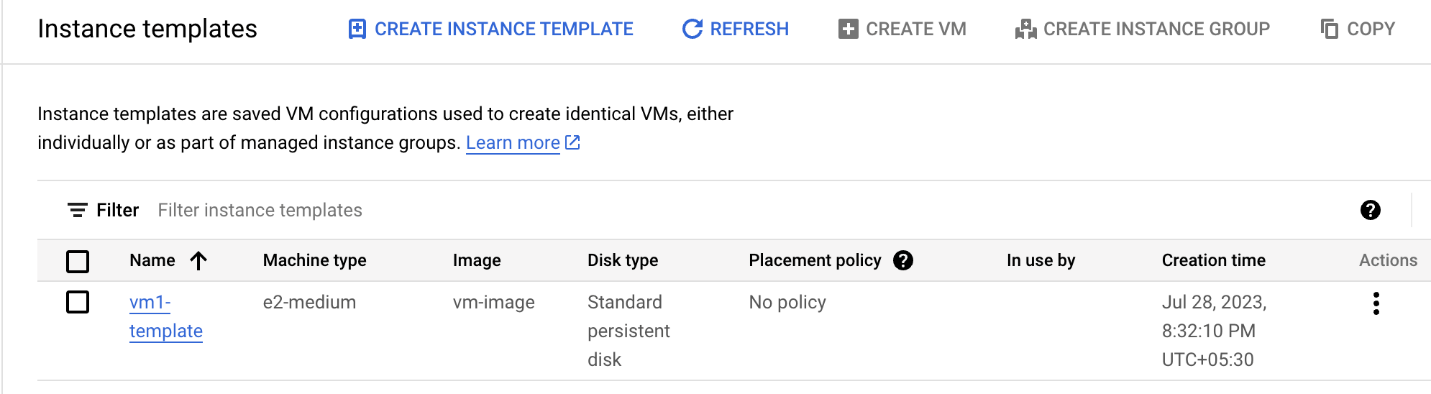


Now, click on **Create instance template** to create a new template.

There are lots of parameters that you can configure when creating a new instance. Use the following for this lab:

|  |  |  |
| --- | --- | --- |
| **Field** | **Value** | **Additional information** |
| Name | vm1-template | Name for the VM instance template |
| Series | E2 | The E2 machine series is Compute Engine's second generation general-purpose machine series. |
| Machine Type | e2-medium | **Note**: A new project has a default [resource quota](https://cloud.google.com/compute/docs/resource-quotas), which may limit the number of CPU cores. You can request more when you work on projects outside of this lab. |
| Boot Disk | vm-image | Click on the **change** button, click on the **custom images** section. Now, select **vm-image** by selecting the project you are working on. |
| Boot disk type | standard persistent disk | Learn more about standard persistent disk check out this [link.](https://cloud.google.com/compute/docs/disks/) |
| Firewall | allow HTTP and HTTPS traffic | Learn more about firewall check out this [link.](https://cloud.google.com/vpc/docs/firewalls) |

Leave the rest of the values set to their default settings. Click on the **create** button to create the instance template vm1-template.



Now, you'll create new VM instances with the template named vm1-template from your local computer using **gcloud command-line interface**. To do this, return back to the command line interface on your local computer, and enter the following command:

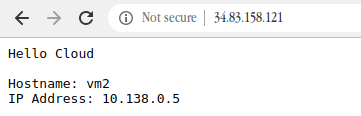
gcloud compute instances create --zone us-west4-a --source-instance-template vm1-template vm2 vm3 vm4 vm5 vm6 vm7 vm8

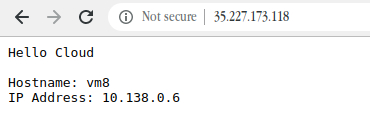
Wait for the command to finish. Once it's done, you can view the instances through the Console or by using the following gcloud command on your local terminal:

gcloud compute instances list

Now, open the external links for vm2 and vm8 to check if all the configuration set up properly as vm1.

Output:





**Congratulations!**

Nice work! You've successfully deployed eight VMs as web servers, each with the same configuration.