GCP Certification Series: Section 4: Ensuring the successful operation of a cloud solution 4.1: 4.1 Managing Compute Engine resources

Prashanta Paudel

Nov 6, 2018 · 25 min read

Today we are going to start section 4. This section basically deals with various tasks that you should be able to do in GCP for the Associate role.

Many of the theoretical portions are already done so we will directly dig into how to complete that tasks.

Managing a single VM instance (e.g., start, stop, edit the configuration, or delete an instance)

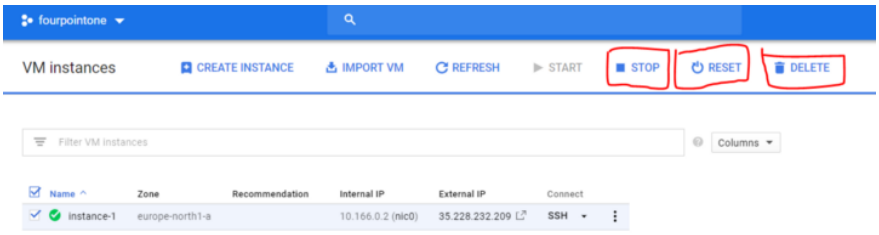
Once the VM is UP and running, you can perform several operations in it like start, stop, edit and delete.

Initially, when we spin up VM for the first time it will start running as soon as it is ready.

Steps are pretty simple and straightforward. We will do in both console and Shell.

> *In console*

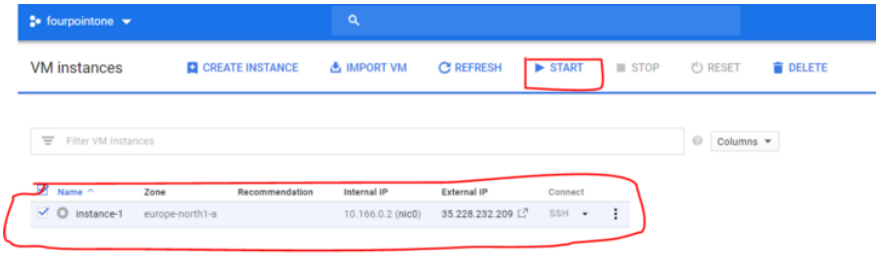
Goto VM instance. When shown ready it will be running so you can STOP or DELETE VM at this time.



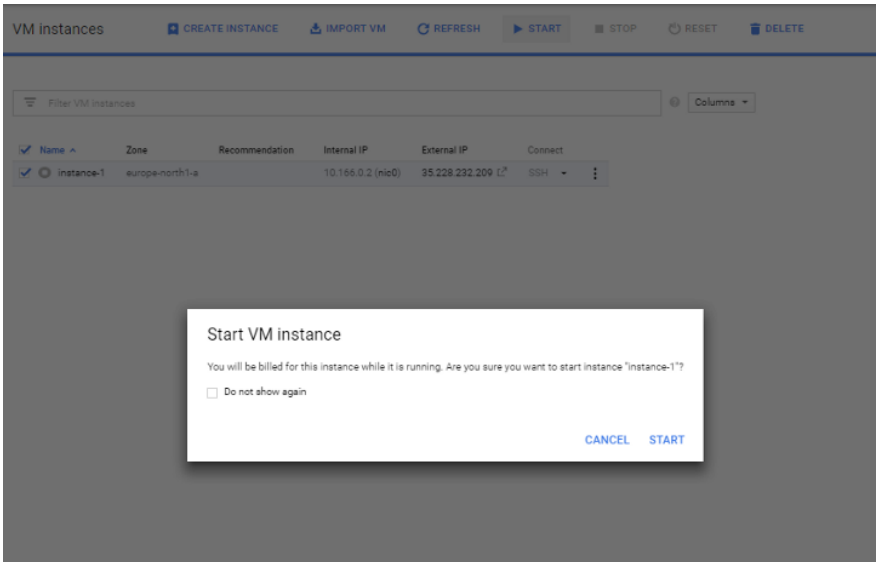
VM

when you STOP the VM

then you will see the START option enabled.



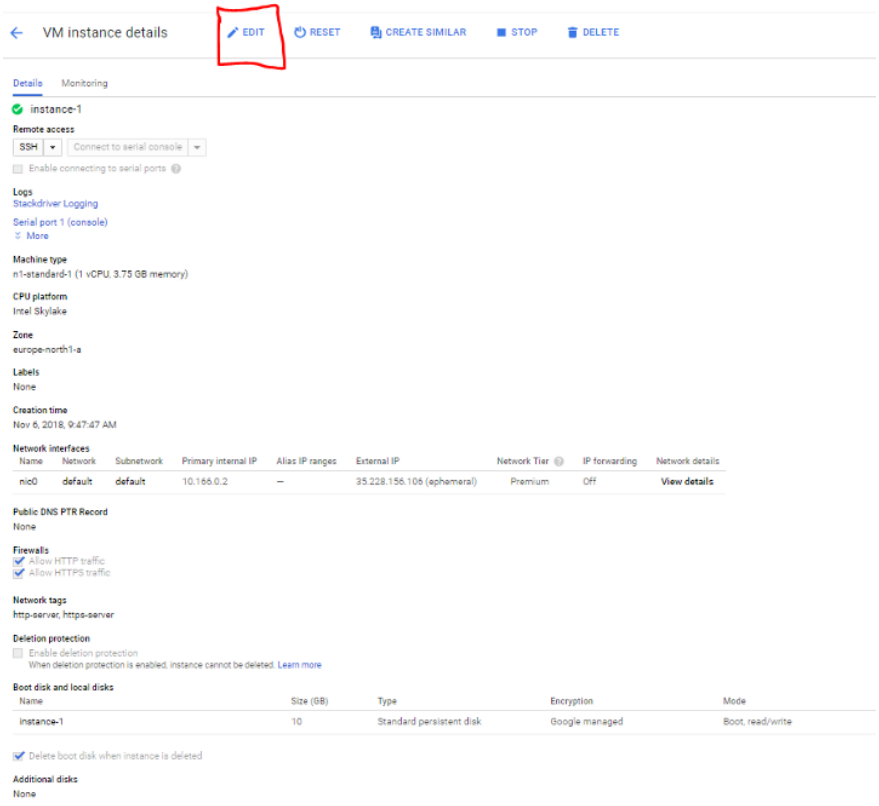
start enabled



Start VM

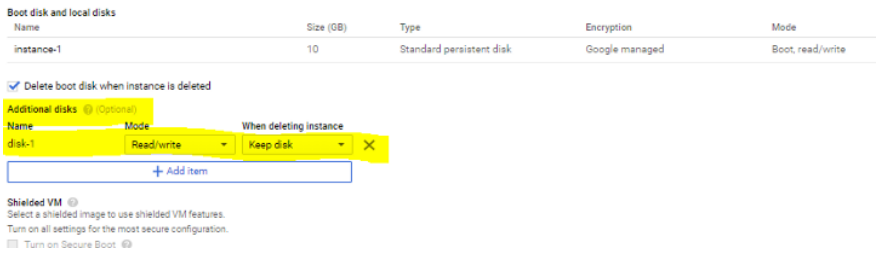
When you want to delete the VM then just select the VM and click on DELETE at the top of the page.

If you want to edit the configuration then click on VM name then you will see



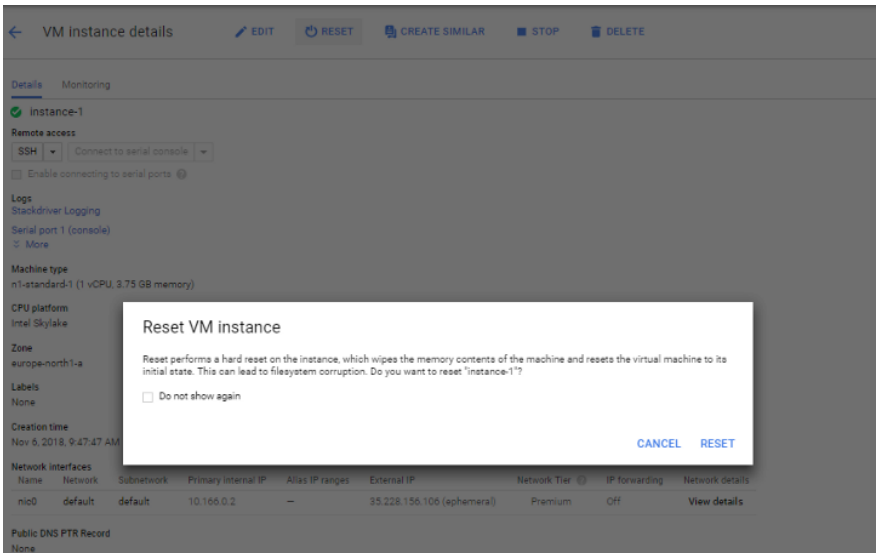
Edit VM

Now just click on Edit and you can change configuration settings, attach a Hard disk to VM, upgrade memory and so on.



attach disk

There is an option to reset the VM, this will erase any data and revert to default settings.



VM reset.

> *In Shell*

```
prashantagcppaudel@cloudshell:~ (fourpointone-221707)$
gcloud compute instances create vm1
Did you mean zone [europe-west4-a] for instance: [vm1]
(Y/n)?  y

Created
[https://www.googleapis.com/compute/v1/projects/fourpointone
-221707/zones/europe-west4-a/instances/vm1].
NAME  ZONE            MACHINE_TYPE   PREEMPTIBLE
INTERNAL_IP  EXTERNAL_IP   STATUS
vm1   europe-west4-a  n1-standard-1              10.164.0.2
35.204.71.54  RUNNING


prashantagcppaudel@cloudshell:~ (fourpointone-221707)$
gcloud compute instances stop instance-1 --zone=europe-
north1-a
Stopping instance(s) instance-1...done.
Updated
[https://www.googleapis.com/compute/v1/projects/fourpointone
-221707/zones/europe-north1-a/instances/instance-1].


prashantagcppaudel@cloudshell:~ (fourpointone-221707)$
gcloud compute instances start instance-1 --zone=europe-
north1-a
Starting instance(s) instance-1...done.
Updated
[https://www.googleapis.com/compute/v1/projects/fourpointone
-221707/zones/europe-north1-a/instances/instance-1].
prashantagcppaudel@cloudshell:~ (fourpointone-221707)$


prashantagcppaudel@cloudshell:~ (fourpointone-221707)$
gcloud compute instances delete instance-1 --zone=europe-
```

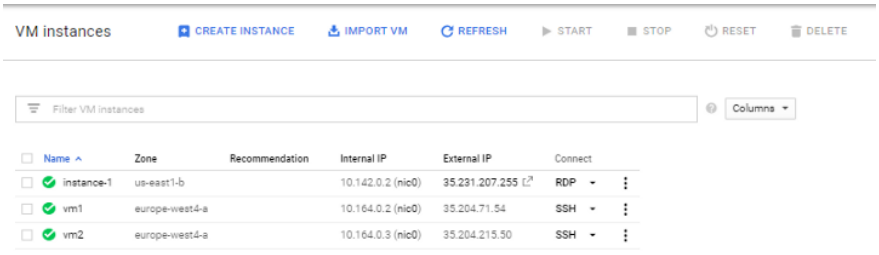
```
north1-a
The following instances will be deleted. Any attached disks
configured
 to be auto-deleted will be deleted unless they are attached
to any
other instances or the `--keep-disks` flag is given and
specifies them
 for keeping. Deleting a disk is irreversible and any data
on the disk
 will be lost.
 - [instance-1] in [europe-north1-a]

Do you want to continue (Y/n)?  y

Deleted
[https://www.googleapis.com/compute/v1/projects/fourpointone
-221707/zones/europe-north1-a/instances/instance-1].
prashantagcppaudel@cloudshell:~ (fourpointone-221707)$
```

==========

SSH/RDP to the instance

SSH is the most frequent and commonly used access medium to login to the VM in linux platform. RDP is commonly used for windows machine

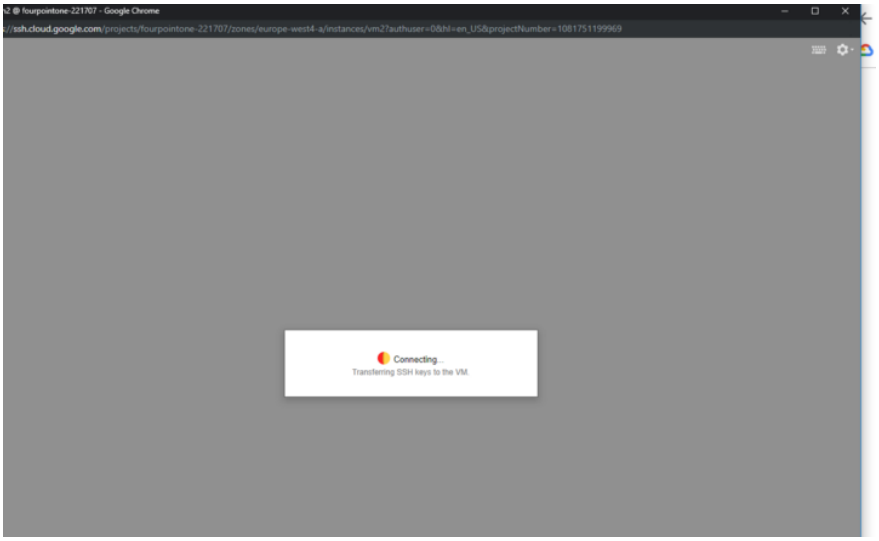
Once the VM is created you will see SSH or RDP on rightmost column depending on whether you have windows or Linux VM created.



VM and access type

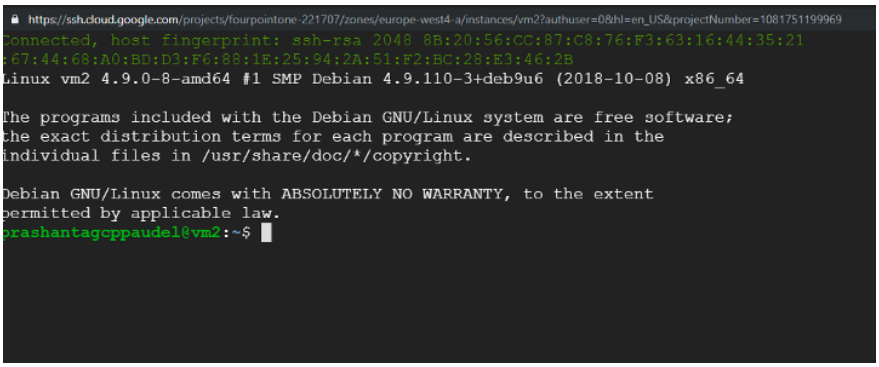
The easiest way to access VM is just clicking on SSH and wait for the connection to establish.

It will take a while for SSH to establish depending on the speed of your connection



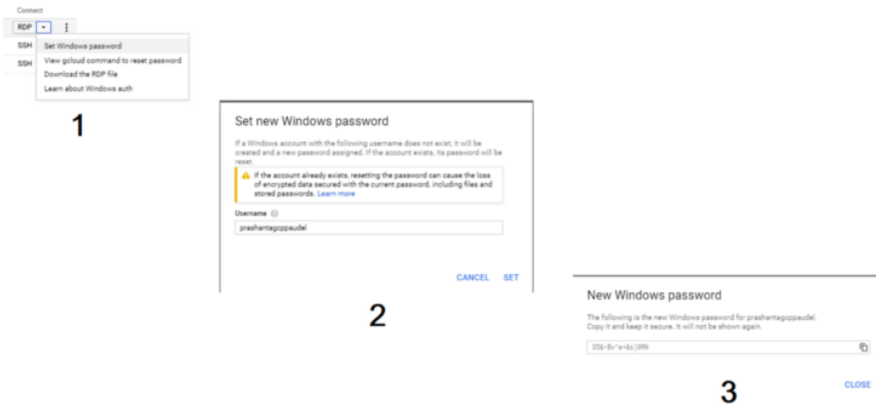
SSH connecting

Once finished you will see the VM's CLI prompt.

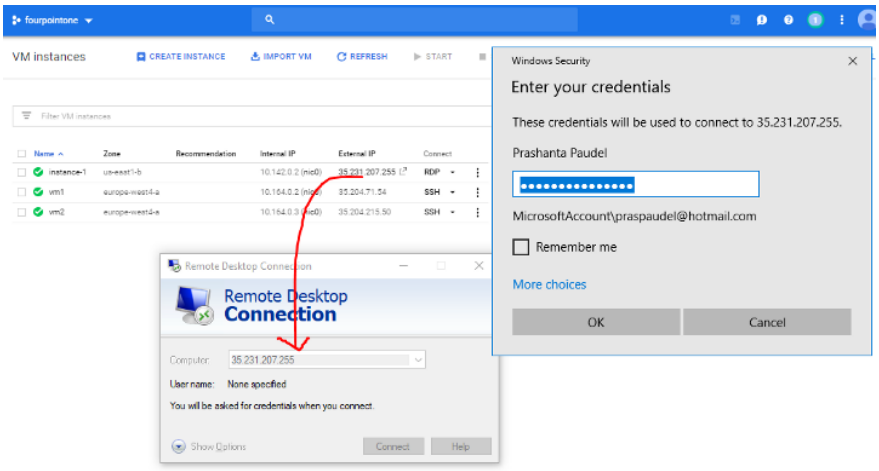


VM CLI

Whereas in RDP the process is a bit different. You have to reset the password first before accessing the server.



Now use the password and username to access the server remotely using RDP.



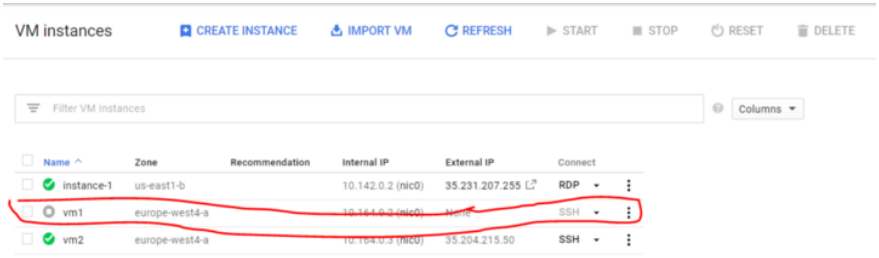
RDP desktop

==========

Attaching a GPU to a new instance and installing CUDA libraries

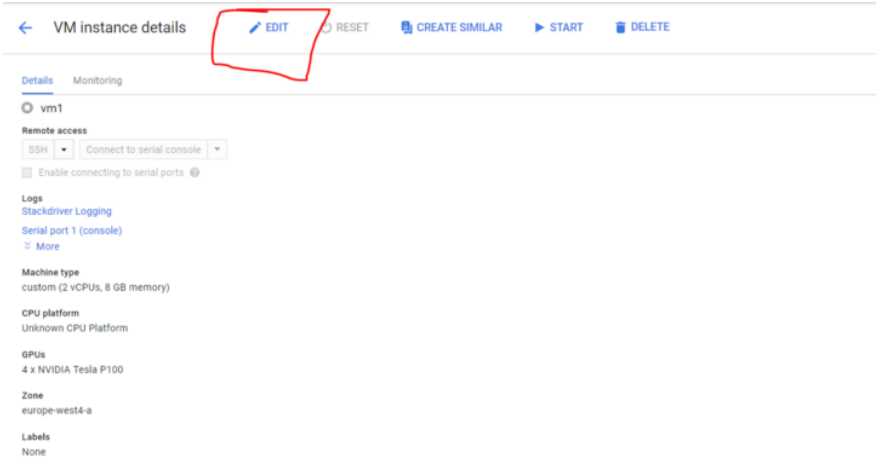
We can install GPU in two ways either you select while creating a VM or install later after VM is installed.

To install GPU afterward you need to STOP the VM and edit the configurations to add GPU.



Stop the VM

Then edit the VM to add GPU



Adding GPU

Add GPU according to the choice and click save. Then only START the VM.

For adding CUDA libraries you need to have NVIDIA account and download the repo from the website.

GCP instance should have Nvidia either K80 or P100 enabled which is not enabled in the free account.

Now, start by downloading the repository from Nvidia.

```
$ curl -O
http://developer.download.nvidia.com/compute/cuda/repos/ubun
tu1604/x86_64/cuda-repo-ubuntu1604_8.0.61-1_amd64.deb
```

When you have completed downloading, install with dpkg and update the repository. Next, clean up after ourselves.

```
$ sudo dpkg -i cuda-repo-ubuntu1604_8.0.61-1_amd64.deb
$ sudo apt-get update
$ rm cuda-repo-ubuntu1604_8.0.61-1_amd64.deb
```

==========

Viewing current running VM Inventory

Checking Instance Status

When you first create an instance, you should check the instance status to see if it is running before you can expect it to respond to requests. It can take a couple of seconds before your instance is fully up and running after the initial request. You can also check the status of an instance at any time after instance creation.

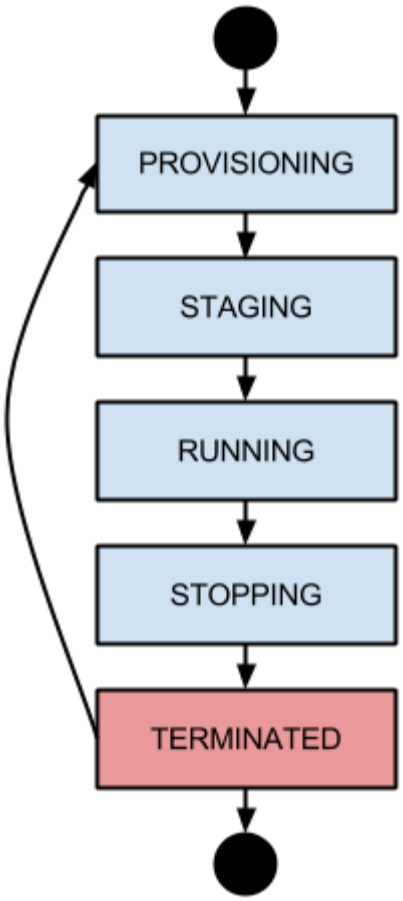
Instance statuses

Instances can be in the following states:

- `PROVISIONING` - Resources are being reserved for the instance. The instance isn't running yet.
- `STAGING` - Resources have been acquired and the instance is being prepared for launch.
- `RUNNING` - The instance is booting up or running. You should be able to ssh into the instance soon, though not immediately after it enters this state.
- `STOPPING` - The instance is being stopped either due to a failure, or the instance is shut down. This is a temporary status and the instance will move to `TERMINATED`.

- `TERMINATED` - The instance was shut down or encountered a failure, either through the API or from inside the guest. You can choose to restart the instance or delete it.
- **Warning:** `TERMINATED` instances with local SSDs attached cannot be restarted at this time. We plan to support this functionality in the future.

The following diagram describes the progression of these statuses.



STATUS

After an instance is `RUNNING`, you can try to connect to it. If you cannot connect to the instance initially, try connecting again in a few minutes, as the operating system might still be booting up.

Windows instances experience a longer startup time because of the sysprep process, so you must run an additional check to verify that the Windows instance has started.

Checking an instance's status

List all instances and their status:

```
gcloud compute instances list
```

Describe the status of a single instance:

```
gcloud compute instances describe example-instance
```

==========

Working with snapshots (e.g., create a snapshot from a VM, view snapshots, delete a snapshot)

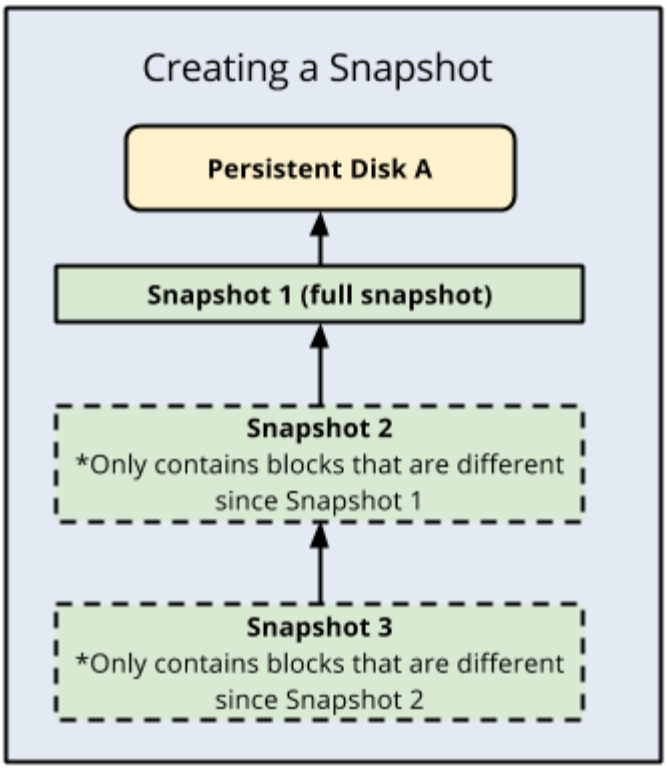
Snapshots are different from public images and custom images, which are used primarily to create instances or configure instance templates. Snapshots are useful for periodic backup of the data on your persistent disks, and you can use snapshots to create a custom image when needed. You can create snapshots from persistent disks even while they are attached to running instances.

Snapshots are incremental and automatically compressed, so you can create regular snapshots on a persistent disk faster and at a much lower cost than if you regularly created a full image of the disk. Incremental snapshots work in the following manner:

- The first successful snapshot of a persistent disk is a full snapshot that contains all the data on the persistent disk.
- The second snapshot only contains any new data or modified data since the first snapshot. Data that hasn't changed since snapshot 1 isn't included. Instead, snapshot 2 contains references to snapshot 1 for any unchanged data.
- Snapshot 3 contains any new or changed data since snapshot 2 but won't contain any unchanged data from snapshot 1 or 2. Instead,

snapshot 3 contains references to blocks in snapshot 1 and snapshot 2 for any unchanged data.

This repeats for all subsequent snapshots of the persistent disk. Snapshots are always created based on the last successful snapshot taken.



Compute Engine stores multiple copies of each snapshot redundantly across multiple locations with automatic checksums to ensure the integrity of your data. Use IAM roles to share snapshots across projects.

To see a list of snapshots available to a project, use the `gcloud compute snapshots list` command:

```
gcloud compute snapshots list
```

To list information about a particular snapshot, use the `gcloud compute snapshots describe` command:

```
gcloud compute snapshots describe example-snapshot
```

Understanding snapshot best practices

You can create as many snapshots as you need at any time, but you can create snapshots more quickly and with greater reliability, if you use the following best practices:

Prepare your persistent disk for the best snapshot consistency

In most situations, you can create a snapshot from persistent disks even while your applications are writing data to them and still expect the snapshot to have good consistency. The quality of the snapshot depends on the ability of your applications to recover from snapshots that you create during heavy write workloads.

If your applications require strict consistency, you can take one or more steps to ensure that a snapshot is consistent with the desired state of the persistent disk.

- Pause applications or operating system processes that write data to that persistent disk. Then, flush the disk buffers before you create the snapshot.
- Unmount the persistent disk completely to ensure that no data is written to it while you create the snapshot. This is usually unnecessary, but it does improve the consistency of the snapshot.
- If your applications require consistency between multiple persistent disks, you must freeze or unmount all of the file systems on each disk and complete all of the snapshots for those disks before you resume your applications. Compute Engine does not guarantee consistency between simultaneous snapshots running on multiple persistent disks.
- Use journaling file systems like `ext4` to reduce the risk that data is cached without actually being written to the persistent disk.
- For persistent disks that are attached to Windows Server instances, use VSS Snapshots to help preserve data integrity.

Use existing snapshots as a baseline for subsequent snapshots

If you have existing snapshots on a persistent disk, the system automatically uses them as a baseline for any subsequent snapshots that you create from that same disk.

- Create a new snapshot from a persistent disk before you delete the previous snapshot from that same persistent disk. The system can create the new snapshot more quickly if it can use the previous snapshot and reads only the new or changed data from the persistent disk.
- Wait for new snapshots to finish before you take subsequent snapshots from the same persistent disk. If you run two snapshots simultaneously on the same persistent disk, they will both start from the same baseline and duplicate effort. If you wait for the new snapshot to finish, any subsequent snapshots will run more quickly because they need only to obtain the data that has changed since the last snapshot finished.

Schedule snapshots during off-peak hours

If you schedule regular snapshots for your persistent disks, you can reduce the time that it takes to complete each snapshot by creating them during off-peak hours when possible.

- Schedule automated snapshots during the business day in the zone where your persistent disk is located. Snapshot creation typically peaks at the end of the business day.
- Schedule automated snapshots early in the morning in the zone where your persistent disk is located rather than immediately at midnight. Snapshot creation typically peaks at midnight.

Organize your data on separate persistent disks

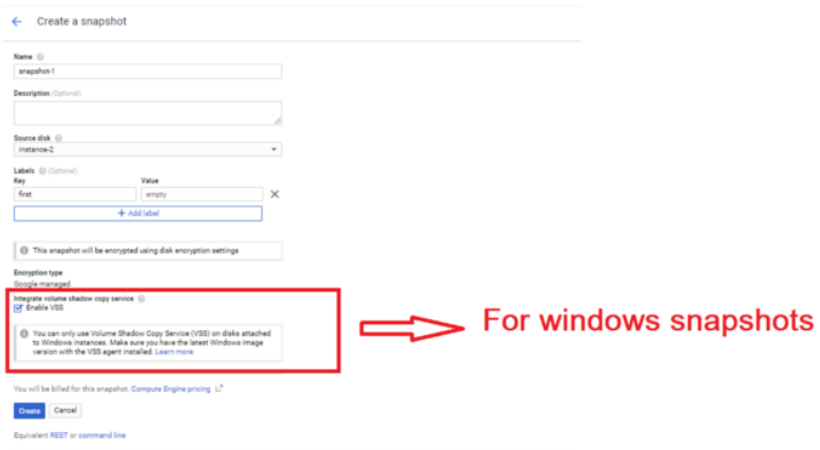
If you create a snapshot of a persistent disk, any data that you store on the disk will be included in the snapshot. Larger amounts of data create larger snapshots, which cost more and take longer to create. To ensure that you are snapshotting only the data that you need, organize your data on separate persistent disks.

- Store critical data on a secondary persistent disk rather than your boot disk. This allows you to snapshot your boot disks only when necessary or on a less frequent schedule.
- If you do create snapshots of your boot disks, store swap partitions, pagefiles, cache files, and non-critical logs on a separate persistent disk. These files and partitions change frequently and the snapshot process is likely to identify them as changed data that must be included in an incremental snapshot.
- Reduce the number of snapshots that you need to create by keeping similar data together on one persistent disk. You do want to keep your operating system and volatile data separate from the data that you want to snapshot, but there is no need to distribute your critical data across multiple persistent disks like you would for a physical machine. One large persistent disk is able to achieve the same performance as multiple smaller persistent disks of the same total size.

Enable the `discard` option or run `fstrim` on your persistent disk

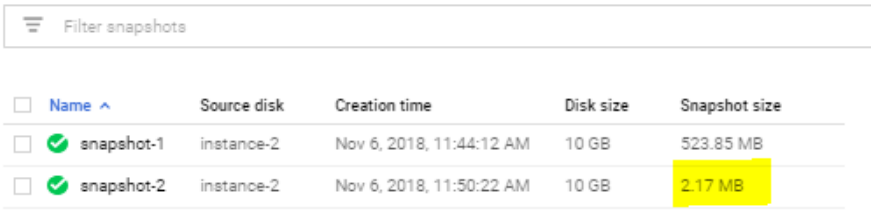
On Linux instances, if you did not format and mount your persistent disk with the `discard` option, run the `fstrim` command on the instance before you create a snapshot. The command removes blocks that the file system no longer needs so that the system can create the snapshot more quickly and with a smaller size. See formatting and mounting a persistent disk to learn how to configure the `discard` option on your persistent disks.

To create a snapshot for a VM, goto Compute engine>snapshots>create Snapshot



create snapshot

The updated snapshot will have updated data only in new snapshot



updated snapshot

Delete a snapshot

```
$ gcloud compute snapshots delete snapshot-2
The following snapshots will be deleted:
 - [snapshot-2]

Do you want to continue (Y/n)?  y

Deleted
[https://www.googleapis.com/compute/v1/projects/fourpointone
-221707/global/snapshots/snapshot-2].
```

Working with Images (e.g., create an image from a VM or a snapshot, view images, delete an image)

Images

Use operating system images to create boot disks for your instances. You can use one of the following image types:

- **Public images** are provided and maintained by Google, open-source communities, and third-party vendors. By default, all projects have access to these images and can use them to create instances.
- **Custom images** are available only to your project. You can create a custom image from boot disks and other images. Then, use the custom image to create an instance.

You can use most public images at no additional cost, but there are some premium images that do add additional cost to your instances. Custom images that you import to Compute Engine add no cost to your instances, but do incur an image storage charge while you keep your custom image in your project.

Some images are capable of running containers on Compute Engine.

Public images

Compute Engine offers many preconfigured public images that have compatible Linux and Windows operating systems. Use these operating system images to create and start instances. Compute Engine uses your selected image to create a persistent boot disk for each instance. By default, the boot disk for an instance is the same size as the image that you selected. If your instance requires a larger persistent boot disk than the image size, resize the boot disk.

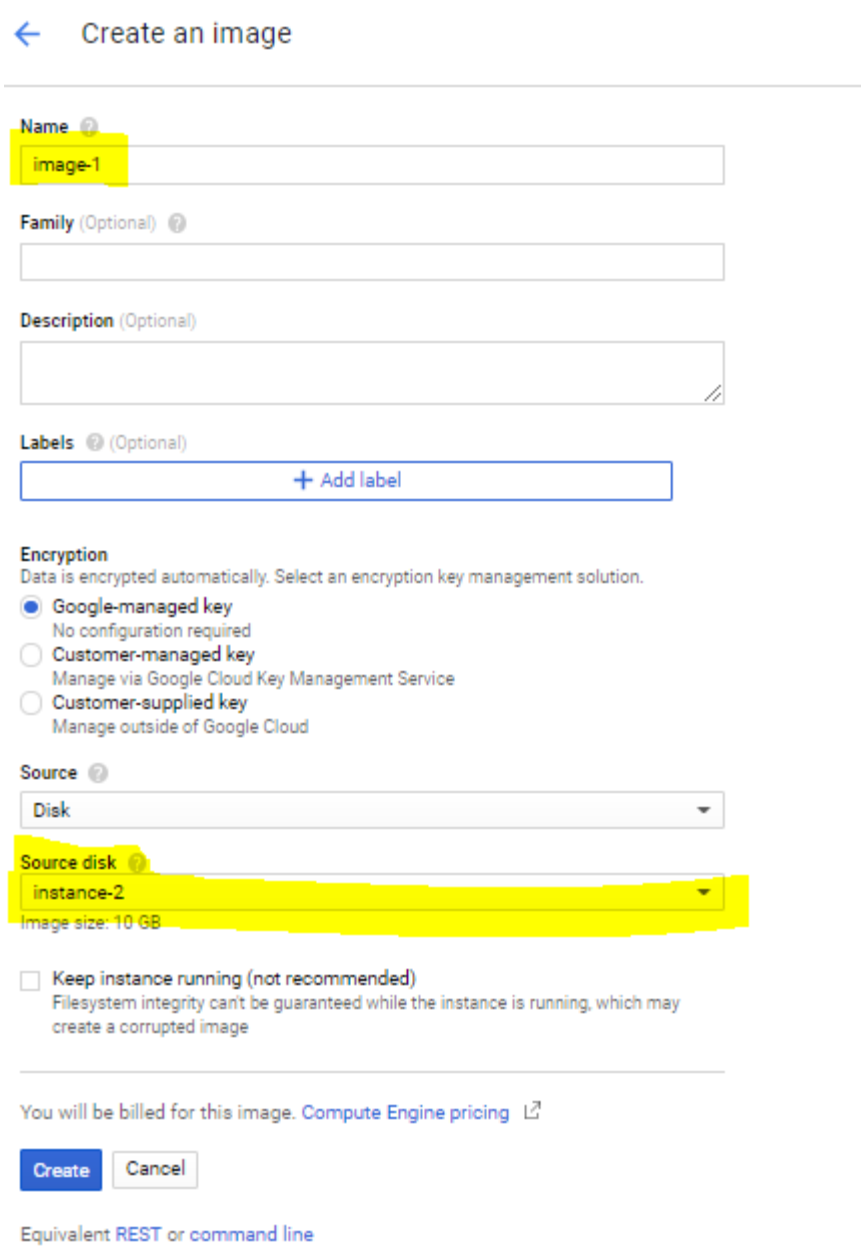
To see the full list of public images with their image names, versions numbers, and image sizes, go to the Images page in the console. Google updates public images regularly, or when a patch for a critical impact CVE is available. Subscribe to GCE-image-notifications to receive notifications for update releases.

Custom images

A custom image is a boot disk image that you own and control access to. Use custom images for the following tasks:

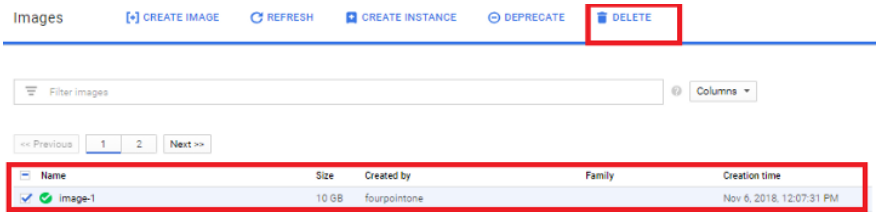
- Import a boot disk image to Compute Engine from your on-prem environment or import virtual disks from VMs that are running on your local workstation or on another cloud platform.
- **Note:** If you are planning to migrate several VMs to Compute Engine, consider using the VM migration service.
- Create an image from the boot disks of your existing Compute Engine instances. Then use that image to create new boot disks for your instances. This process allows you to create new instances that are preconfigured with the applications that you need without having to configure a public image from scratch.
- Copy one image to another image using either the `gcloud` tool or the API. Use the same process that you use to create an image, but specify another image as the image source. You can also create an image from a custom image in a different project.

Create an image from VM



create an image from VM

To delete an image



delete an image

To create an image from a snapshot

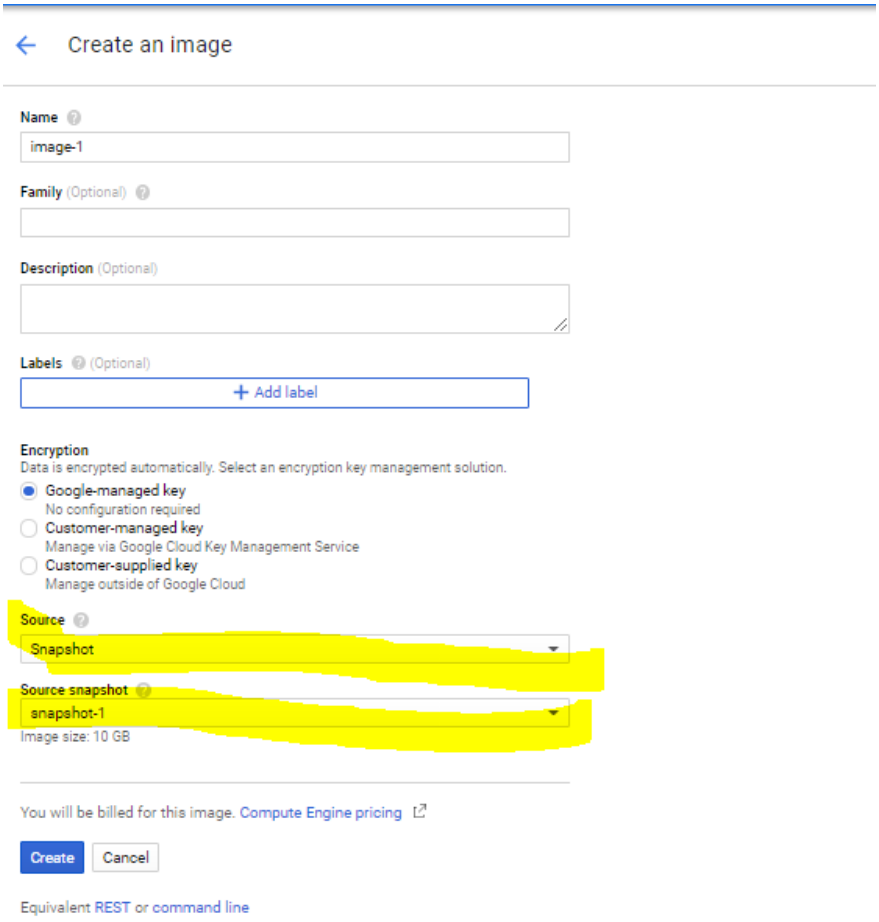


image from snapshot

==========

Working with Instance Groups (e.g., set auto scaling parameters, assign instance template, create an instance template, remove instance group)

The starting point in auto-scaling is determining template of the machine, then build an instance group and finally define load balancing.

Auto-scaling and load-balancing are the fulfillment of each other.

Autoscaling Groups of Instances

Managed instance groups offer autoscaling capabilities that allow you to automatically add or delete instances from a managed instance group based on increases or decreases in load. Autoscaling helps your applications gracefully handle increases in traffic and reduces cost when the need for resources is lower. You just define the autoscaling policy and the autoscaler performs automatic scaling based on the measured load.

Autoscaling works by adding more instances to your instance group when there is more load (upscaling), and deleting instances when the need for instances is lowered (downscaling).

Fundamentals

Autoscaling uses the following fundamental concepts and services.

Managed instance groups

Autoscaling is a feature of managed instance groups. A managed instance group is a pool of homogeneous instances, created from a common instance template. An autoscaler adds or deletes instances from a managed instance group. Although Compute Engine has both managed and unmanaged instance groups, only managed instance groups can be used with autoscaler.

To understand the difference between a managed instance group and unmanaged instance group, see the Instance Groups documentation.

Autoscaling policy and target utilization

To create an autoscaler, you must specify the autoscaling policy and a target utilization level that the autoscaler uses to determine when to scale the group. You can choose to scale using the following policies:

- Average CPU utilization
- HTTP load balancing serving capacity, which can be based on either utilization or requests per second.
- Stackdriver Monitoring metrics

The autoscaler will collect information based on the policy, compare it to your desired target utilization, and determine if it needs to perform scaling.

The target utilization level is the level at which you want to maintain your virtual machine instances. For example, if you scale based on CPU utilization, you can set your target utilization level at 75% and the autoscaler will maintain the CPU utilization of the specified group of instances at or close to 75%. The utilization level for each metric is interpreted differently based on the autoscaling policy.

For a brief summary of each policy, see Autoscaling policies in **Overviews**. For a detailed discussion of each policy, see:

- Scaling Based on CPU or Load Balancing Serving Capacity
- Scaling Based on Stackdriver Monitoring Metrics

Instance Groups

You can create and manage groups of virtual machine (VM) instances so that you don't have to individually control each instance in your project. Compute Engine offers two different types of instance groups: **managed** and **unmanaged instance** groups.

Managed instance groups

A managed instance group uses an instance template to create a group of identical instances. You control a managed instance group as a single entity. If you wanted to make changes to instances that are part of a managed instance group, you would make the change to the whole instance group. Because managed instance groups contain identical instances, they offer the following features:

- When your applications require additional compute resources, managed instance groups can automatically scale the number of instances in the group.
- Managed instance groups work with load balancing services to distribute traffic to all of the instances in the group.
- If an instance in the group stops, crashes, or is deleted by an action other than the instance groups commands, the managed instance

group automatically recreates the instance so it can resume its processing tasks. The recreated instance uses the same name and the same instance template as the previous instance, even if the group references a different instance template.

- Managed instance groups can automatically identify and recreate unhealthy instances in a group to ensure that all of the instances are running optimally.

Types of managed instance groups

You can create two types of managed instance groups:

- A zonal managed instance group, which contains instances from the same zone.
- A regional managed instance group, which contains instances from multiple zones across the same region.

Regional managed instance groups are generally recommended over zonal managed instance groups because they allow you to spread application load across multiple zones, rather than confining your application to a single zone or having to manage multiple instance groups across different zones. This replication protects against zonal failures and unforeseen scenarios where an entire group of instances in a single zone malfunctions. If that happens, your application can continue serving traffic from instances running in another zone in the same region.

Choose zonal managed instance groups if you want to avoid the slightly higher latency incurred by cross-zone communication or if you need fine-grained control of the sizes of your groups in each zone.

Managed instance groups and the network

By default, instances in the group will be placed in the `default` network and randomly assigned IP addresses from the regional range. Alternatively, you can restrict the IP range of the group by creating a custom mode VPC network and subnet that uses a smaller IP range, then specifying this subnet in the instance template. This can simplify the creation of firewall rules.

After you create a managed instance group, the new instances start in the group as soon as the system can provide them. This process can take a significant amount of time depending on the number of instances in the group. Verify the status of instances in your managed instance group.

Unmanaged instance groups

Unmanaged instance groups are groups of dissimilar instances that you can arbitrarily add and remove from the group. Unmanaged instance groups do not offer autoscaling, rolling update support, or the use of instance templates so Google recommends creating managed instance groups whenever possible. Use unmanaged instance groups only if you need to apply load balancing to your pre-existing configurations or to groups of dissimilar instances.

If you must create a group of dissimilar instances that do not follow an instance template, see Unmanaged Instance Groups.

Instance groups and load balancing

All of the load balancing configurations available on Google Cloud Platform require that you specify instance groups or target pools that can serve traffic distributed from the load balancer.

For HTTP(S), internal, and SSL load balancing, you must assign an instance group to a backend service. A backend service is a centralized service for managing backends, which in turn manages instances that handle user requests for your load balancer. Each backend service contains one or more backends, and each backend contains one instance group. The backend service knows which instances it can use, how much traffic they can handle, and how much traffic they are currently handling. You can assign either a managed or unmanaged instance group to a backend service.

For Network load balancing, you must add individual VM instances to a target pool or assign one or more managed instance groups to a target pool, which causes the server to add all instances that are part of the instance group to the specified target pool.

For more information on different load balancing configurations, see the load balancing documentation.

Managed instance groups and autoscaling

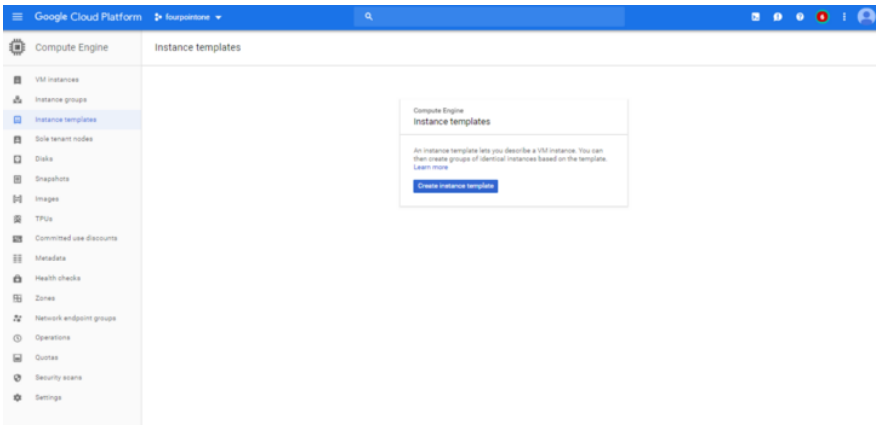
Managed instance groups support autoscaling so you can dynamically add or remove instances from a managed instance group in response to increases or decreases in load. You enable autoscaling and choose an autoscaling policy to determine how you want to scale. Applicable autoscaling policies include scaling based on CPU utilization, load balancing capacity, Stackdriver monitoring metrics, or by a queue-based workload like Google Cloud Pub/Sub.

Because autoscaling requires adding and removing instances from a group, you can only use autoscaling with managed instance groups so the autoscaler can maintain identical instances. Autoscaling does not work on unmanaged instance groups, which can contain heterogeneous instances.

For more information, read Autoscaling Groups of Instances.

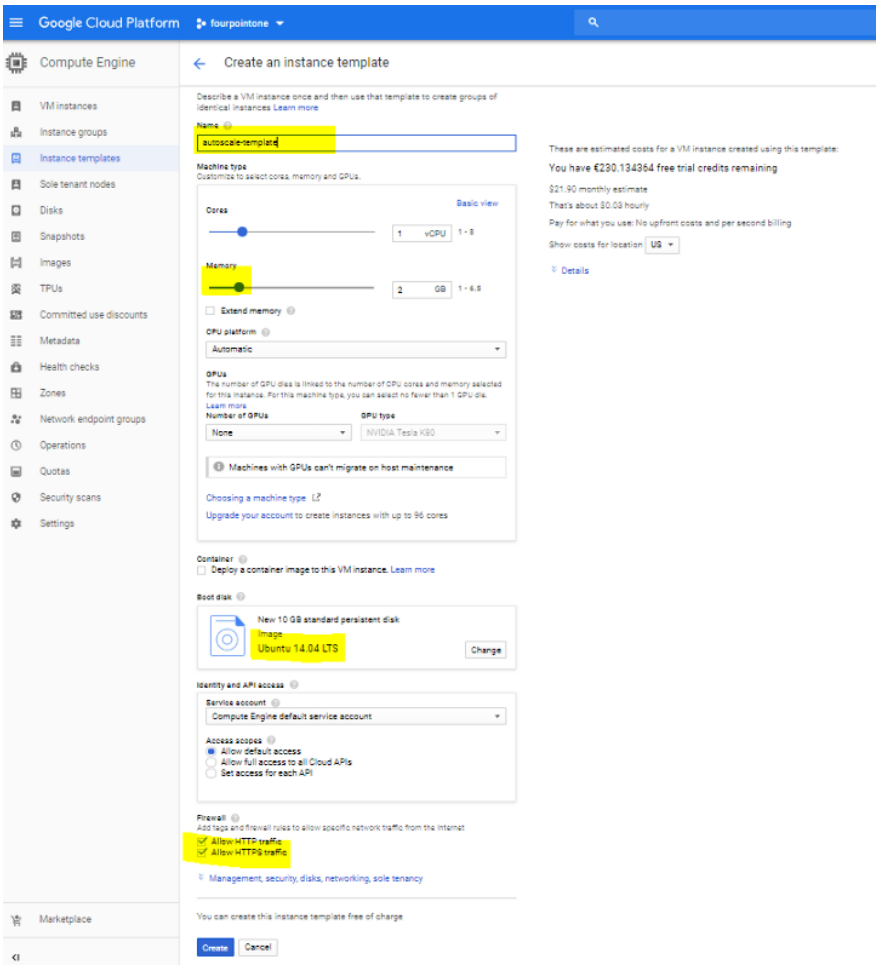
Create Instance Template

To create instance template goto compute engine and instance template



create instance template

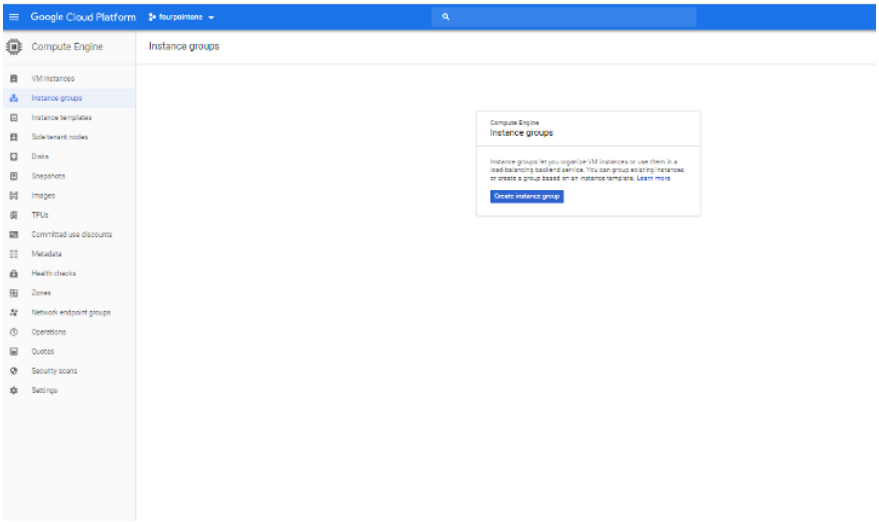
After clicking create you will be presented with the same kind of page as creating a VM. This template will be replicated to several machines while autoscaling.



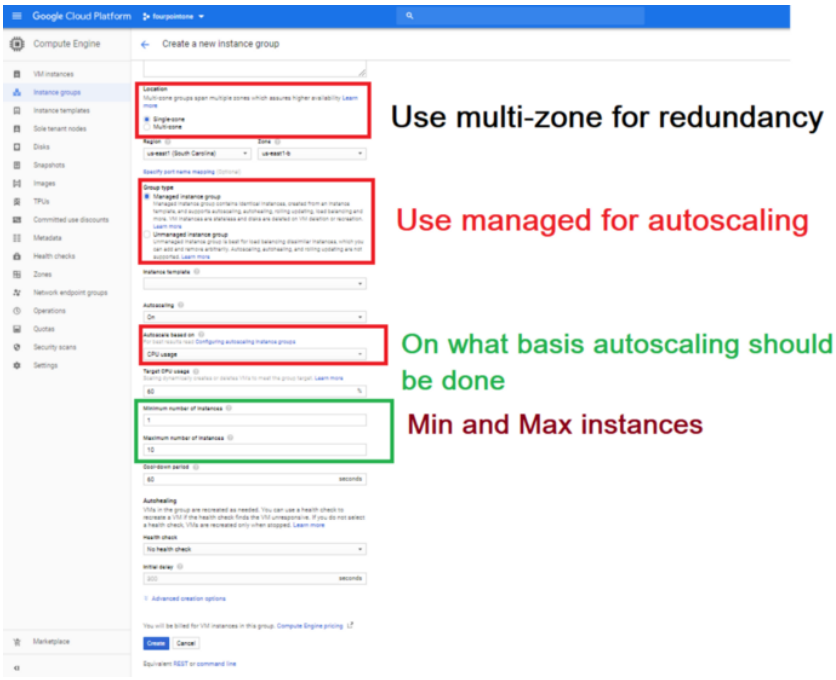
template

Create an instance group

click on the compute engine>Instance group

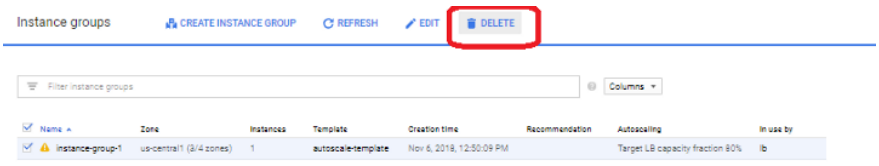


Instance group



Create a group

To remove instance group



delete instance group

IN shell

```
$ gcloud compute instance-groups managed delete instance-
group-1 --region=us-central1
The following region instance group managers will be
deleted:
 - [instance-group-1] in [us-central1]

Do you want to continue (Y/n)?  y

Deleting autoscaler...⠛Deleted
[https://www.googleapis.com/compute/v1/projects/fourpointone
-221707/regions/us-central1/autoscalers/instance-group-1].
Deleting autoscaler...done.
Deleting Managed Instance Group...done.
```

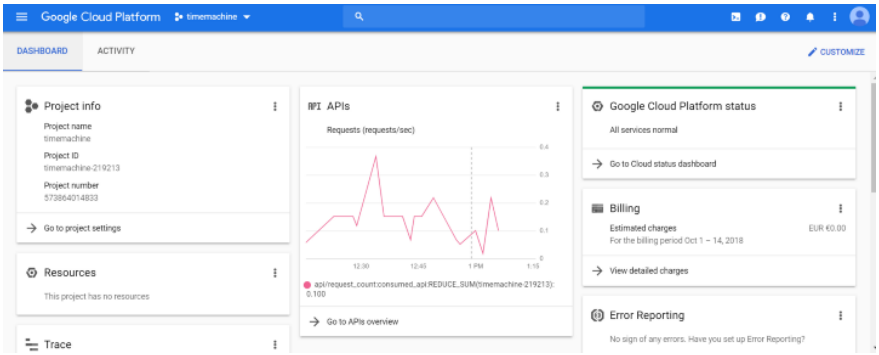
==========

Working with management interfaces (e.g., Cloud Console, Cloud Shell, GCloud SDK)

Google cloud platform can be used in various ways through API, cloud shell and cloud console. Most common method of getting your hands dirty while learning the Google cloud platform is Cloud Console while someone with scripting or programming knowledge may wish to use through the shell.

Cloud console

Cloud Console is the visual way of working with the cloud. It has almost all the features that cloud SDK as a virtual machine gives to the admin. Console in the gateway to all the features of Google cloud and is very handy in a way it is organized visually.

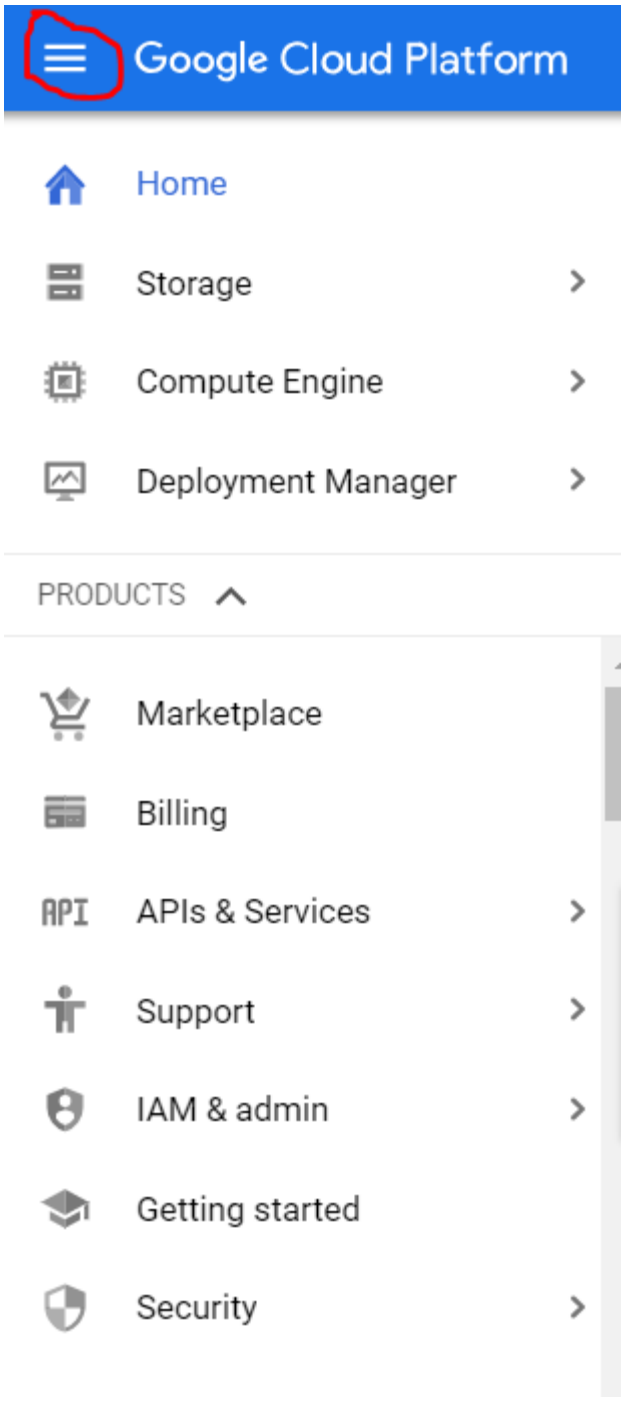


GCP Console

The top left portion of the console has a menu icon which when clicked will slide a list of menus from left. Even though the interface of GCP has been changing frequently the place for menu icon and personal profile, notifications, cloud shell has more or less remained in some spots.

When you click on the menu icon(three small lines), you will get a list of sub-menus. by default, you will have a home button and products. Home button will show the dashboard of your GCP project selected besides Top menu.

project selector

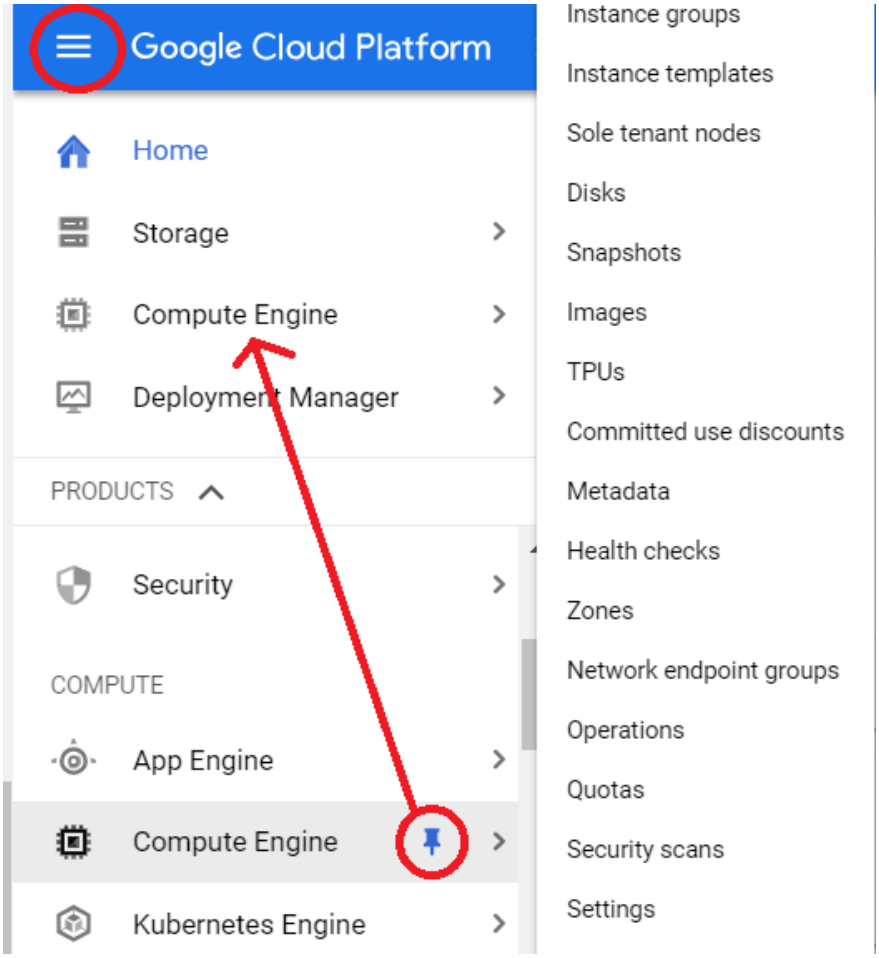


submenus

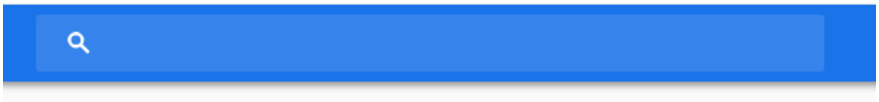
Sub-menu consists of all the features in GCP. This list is the most changing list on the whole console page. Some of the features are in

beta phase and some merged in another menu while some appear in the main list.

In the main list, there is an option to pin the menu below home so that you don't need to scroll down all the time.



Search option in the middle of the top menu bar gives easy access to the search function for the users. You can search for any project, deployment, marketplace etc from this place. If you are fast typer then this could be much easier than going through the main menu.



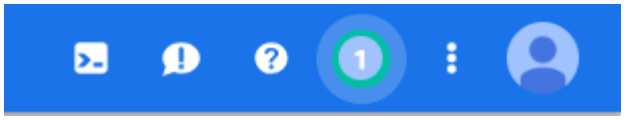
Search

The top right items are profile, settings, notification, help, feedback and cloud console.

Top-right menu

All other buttons are general except notification and cloud shell.

Notification is like the log of the activities you perform in GCP. It will change its appearance when GCP is doing something.



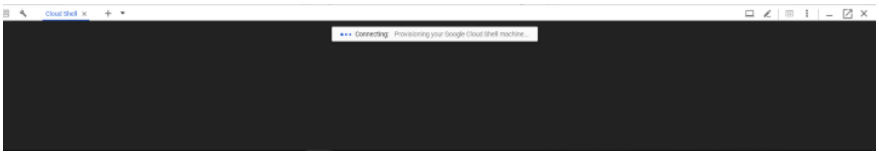
creating a project notification

Cloud shell

It is another way of using GCP. The important thing to note while using the shell is that you must have prior experience working with the shell. Although shell has help and manuals for understanding its commands, it is time-consuming to go through all the options and to apply properly structured command if the same thing can be carried out from GUI easily.

You can access cloud shell from the top menu bar by clicking the command prompt like icon.

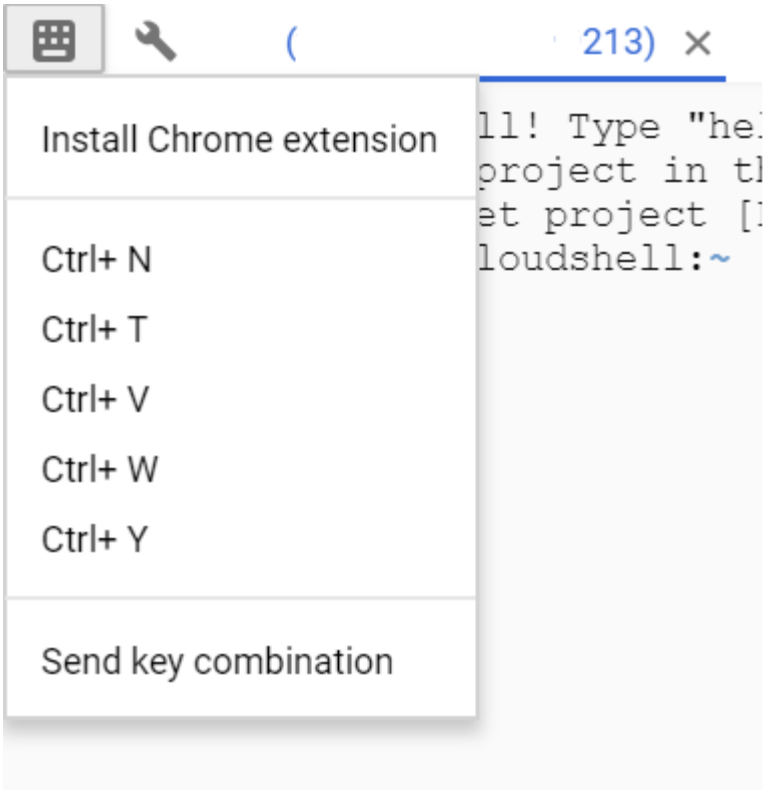
After clicking the icon a new virtual machine with cloud SDK will be initialized and loaded into the platform.



Cloud shell provides command line environment to access cloud resources directly from the browser. It is equivalent to downloading, installing and connecting to Google Cloud SDK in your computer.

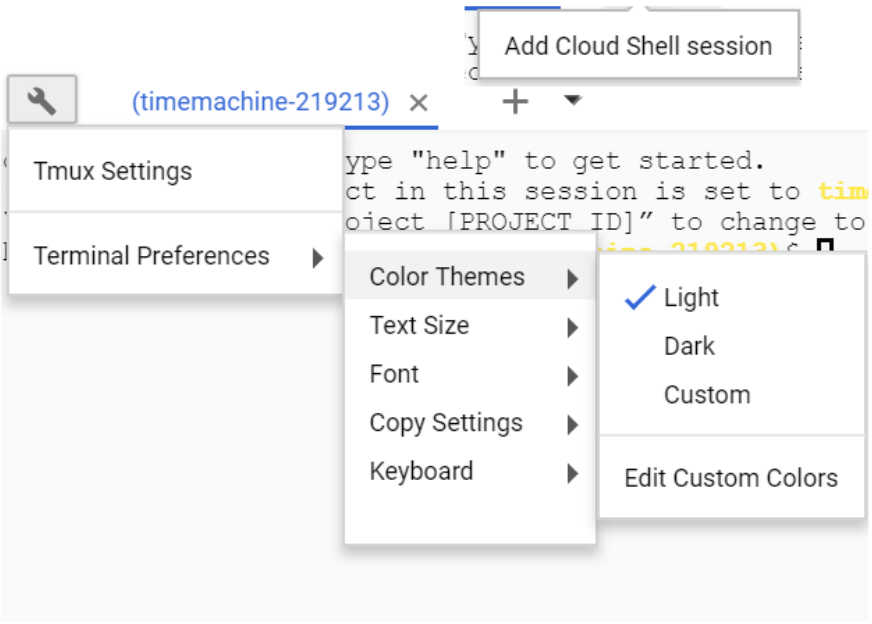
Cloud shell also has options at the top menu bar.

The first option will let you install chrome extension for sending the key combination



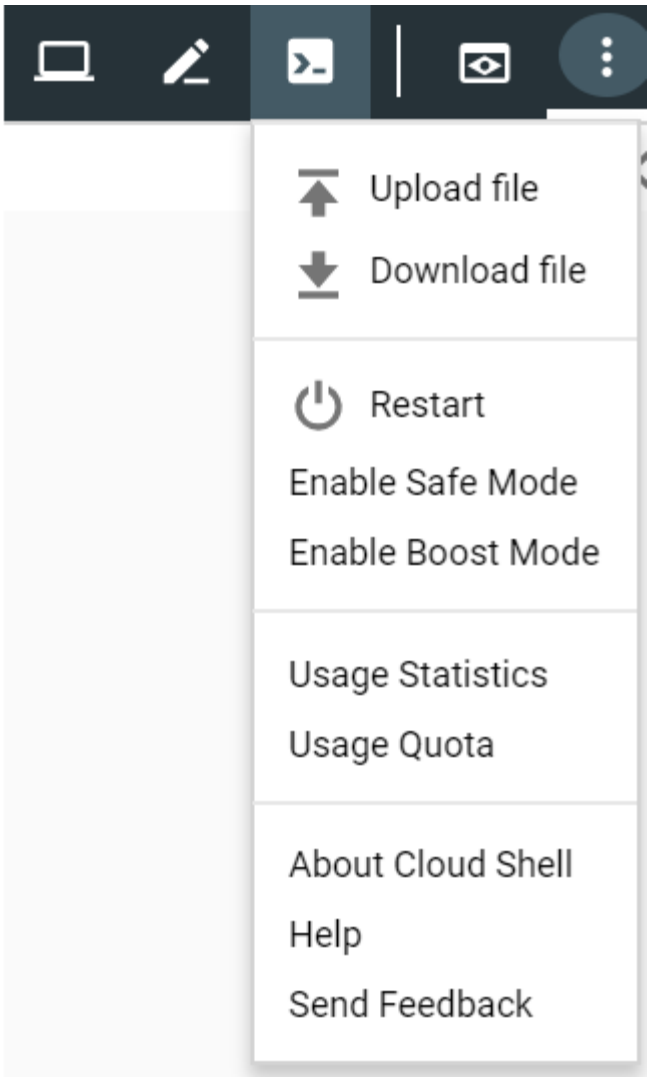
First option

The second icon will give an option for changing the appearance of the shell as well as copy and keyboard settings



The third option will let you add another session to the shell.

The last icon with three dots when pressed has several functions like uploading the files, restarting shell, checking statistics etc.



last option

Things to be careful

Cloud Shell gives 5GB of persistent disk storage in your home directory.

You must load the project in the cloud shell to start using resources in it. We do it using the command

```
$ gcloud config set project Project_ID
```

if we apply commands for different project things may really go wrong.

If you use root privileges the project selection should be confirmed again.

Always check which project are you working on by seeing the project ID after username@cloudshell~(PROJECT _ID)$.

GCLoud SDK in Linux

First we need to add repository info into linux then use that info to download cloud-sdk. After downloading initialize it throgh browser or console and setup region and project.

```
[prashantapaudel@localhost ~]$ sudo tee -a
/etc/yum.repos.d/google-cloud-sdk.repo << EOM
> [google-cloud-sdk]
> name=Google Cloud SDK
> baseurl=https://packages.cloud.google.com/yum/repos/cloud-sdk-
el7-x86_64
> enabled=1
> gpgcheck=1
> repo_gpgcheck=1
> gpgkey=https://packages.cloud.google.com/yum/doc/yum-
key.gpg
> https://packages.cloud.google.com/yum/doc/rpm-package-key.gpg
> EOM
[google-cloud-sdk]
name=Google Cloud SDK
baseurl=https://packages.cloud.google.com/yum/repos/cloud-sdk-
el7-x86_64
enabled=1
gpgcheck=1
repo_gpgcheck=1
gpgkey=https://packages.cloud.google.com/yum/doc/yum-key.gpg
 https://packages.cloud.google.com/yum/doc/rpm-package-key.gpg
[prashantapaudel@localhost ~]$ sudo yum install google-cloud-sdk
Loaded plugins: fastestmirror, langpacks
google-cloud-sdk/signature
 | 454 B 00:00:00
Retrieving key from
https://packages.cloud.google.com/yum/doc/yum-key.gpg
Importing GPG key 0xA7317B0F:
 Userid : "Google Cloud Packages Automatic Signing Key <gc-
```

```
team@google.com>”
 Fingerprint: d0bc 747f d8ca f711 7500 d6fa 3746 c208 a731 7b0f
 From : https://packages.cloud.google.com/yum/doc/yum-key.gpg
Is this ok [y/N]: y
Retrieving key from
https://packages.cloud.google.com/yum/doc/rpm-package-key.gpg
google-cloud-sdk/signature
 | 1.4 kB 00:00:03 !!!
google-cloud-sdk/primary
 | 59 kB 00:00:01
Loading mirror speeds from cached hostfile
google-cloud-sdk
 380/380
Resolving Dependencies
 → Running transaction check
—-> Package google-cloud-sdk.noarch 0:223.0.0–1.el7 will be
installed
 → Finished Dependency Resolution

Dependencies Resolved

================================================
================================================
================================================
====
 Package Arch Version
 Repository Size
================================================
================================================
================================================
====
Installing:
 google-cloud-sdk noarch 223.0.0–1.el7
 google-cloud-sdk 29 M

Transaction Summary
================================================
================================================
================================================
====
Install 1 Package
```

```
Total download size: 29 M
Installed size: 130 M
Is this ok [y/d/N]: y
Downloading packages:
warning: /var/cache/yum/x86_64/7/google-cloud-
sdk/packages/441c59a170ea8367941da520d486218a043fb308e4e5
d7f64c44c47a64603a13-google-cloud-sdk-223.0.0–1.el7.noarch.rpm:
Header V4 RSA/SHA1 Signature, key ID 3e1ba8d5: NOKEY
Public key for
441c59a170ea8367941da520d486218a043fb308e4e5d7f64c44c47a6
4603a13-google-cloud-sdk-223.0.0–1.el7.noarch.rpm
is not installed
441c59a170ea8367941da520d486218a043fb308e4e5d7f64c44c47a6
4603a13-google-cloud-sdk-223.0.
| 29 MB 00:00:09
Retrieving key from
https://packages.cloud.google.com/yum/doc/yum-key.gpg
Importing GPG key 0xA7317B0F:
 Userid : "Google Cloud Packages Automatic Signing Key <gc-
team@google.com>"
 Fingerprint: d0bc 747f d8ca f711 7500 d6fa 3746 c208 a731 7b0f
 From : https://packages.cloud.google.com/yum/doc/yum-key.gpg
Is this ok [y/N]: y
Retrieving key from
https://packages.cloud.google.com/yum/doc/rpm-package-key.gpg
Importing GPG key 0x3E1BA8D5:
 Userid : "Google Cloud Packages RPM Signing Key <gc-
team@google.com>"
 Fingerprint: 3749 e1ba 95a8 6ce0 5454 6ed2 f09c 394c 3e1b a8d5
 From : https://packages.cloud.google.com/yum/doc/rpm-package-
key.gpg
Is this ok [y/N]: y
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
 Installing : google-cloud-sdk-223.0.0–1.el7.noarch
 1/1
 Verifying : google-cloud-sdk-223.0.0–1.el7.noarch
 1/1
```

```
Installed:
 google-cloud-sdk.noarch 0:223.0.0–1.el7

Complete!
[prashantapaudel@localhost ~]$ gcloud init
Welcome! This command will take you through the configuration of
gcloud.

Your current configuration has been set to: [default]

You can skip diagnostics next time by using the following flag:
 gcloud init—skip-diagnostics

Network diagnostic detects and fixes local network connection issues.
Checking network connection…done.
Reachability Check passed.
Network diagnostic (1/1 checks) passed.

You must log in to continue. Would you like to log in (Y/n)? y

Your browser has been opened to visit:

https://accounts.google.com/o/oauth2/auth?
redirect_uri=http%3A%2F%2Flocalhost%3A8085%2F&prompt=selec
t_account&response_type=code&client_id=32555940559.apps.googl
eusercontent.com&scope=https%3A%2F%2Fwww.googleapis.com%2
Fauth%2Fuserinfo.email+https%3A%2F%2Fwww.googleapis.com%2
Fauth%2Fcloud-
platform+https%3A%2F%2Fwww.googleapis.com%2Fauth%2Fappen
gine.admin+https%3A%2F%2Fwww.googleapis.com%2Fauth%2Fco
mpute+https%3A%2F%2Fwww.googleapis.com%2Fauth%2Faccounts
.reauth&access_type=offline

(process:3530): GLib-CRITICAL **: g_slice_set_config: assertion
`sys_page_size == 0' failed
You are logged in as: [prashantagcppaudel@gmail.com].

Pick cloud project to use:
 [1] fourpointone-221707
 [2] Create a new project
Please enter numeric choice or text value (must exactly match list
item): 1
```

Your current project has been set to: [fourpointone-221707].

Do you want to configure a default Compute Region and Zone? (Y/n)?
y

Which Google Compute Engine zone would you like to use as project default?
If you do not specify a zone via a command line flag while working with Compute Engine resources, the default is assumed.
[1] us-east1-b
[2] us-east1-c
[3] us-east1-d
[4] us-east4-c
[5] us-east4-b
[6] us-east4-a
[7] us-central1-c
[8] us-central1-a
[9] us-central1-f
[10] us-central1-b
[11] us-west1-b
[12] us-west1-c
[13] us-west1-a
[14] europe-west4-a
[15] europe-west4-b
[16] europe-west4-c
[17] europe-west1-b
[18] europe-west1-d
[19] europe-west1-c
[20] europe-west3-b
[21] europe-west3-c
[22] europe-west3-a
[23] europe-west2-c
[24] europe-west2-b
[25] europe-west2-a
[26] asia-east1-b
[27] asia-east1-a
[28] asia-east1-c
[29] asia-southeast1-b
[30] asia-southeast1-a
[31] asia-southeast1-c
[32] asia-northeast1-b

```
 [33] asia-northeast1-c
 [34] asia-northeast1-a
 [35] asia-south1-c
 [36] asia-south1-b
 [37] asia-south1-a
 [38] australia-southeast1-b
 [39] australia-southeast1-c
 [40] australia-southeast1-a
 [41] southamerica-east1-b
 [42] southamerica-east1-c
 [43] southamerica-east1-a
 [44] asia-east2-a
 [45] asia-east2-b
 [46] asia-east2-c
 [47] europe-north1-a
 [48] europe-north1-b
 [49] europe-north1-c
 [50] northamerica-northeast1-a
Did not print [6] options.
Too many options [56]. Enter "list" at prompt to print choices fully.
Please enter numeric choice or text value (must exactly match list
item): 10

Your project default Compute Engine zone has been set to [us-central1-
b].
You can change it by running [gcloud config set compute/zone NAME].

Your project default Compute Engine region has been set to [us-
central1].
You can change it by running [gcloud config set compute/region
NAME].

Created a default .boto configuration file at
[/home/prashantapaudel/.boto]. See this file and
[https://cloud.google.com/storage/docs/gsutil/commands/config] for
more
information about configuring Google Cloud Storage.
Your Google Cloud SDK is configured and ready to use!

* Commands that require authentication will use
prashantagcppaudel@gmail.com by default
```

```
* Commands will reference project `fourpointone-221707` by default
* Compute Engine commands will use region `us-central1` by default
* Compute Engine commands will use zone `us-central1-b` by default

Run `gcloud help config` to learn how to change individual settings

This gcloud configuration is called [default]. You can create
additional configurations if you work with multiple accounts and/or
projects.
Run `gcloud topic configurations` to learn more.

Some things to try next:

* Run `gcloud—help` to see the Cloud Platform services you can
interact with. And run `gcloud help COMMAND` to get help on any
gcloud
command.
* Run `gcloud topic -h` to learn about advanced features of the SDK
like arg files and output formatting
[prashantapaudel@localhost ~]$
```