

# Google Cloud Platform(GCP)

## Certification Series: 1.3 Installing and configuring the command line interface (CLI), specifically the Cloud SDK.



Prashanta Paudel

Oct 15, 2018 · 7 min read

We had a brief discussion about GCP command line interface in our blog some time ago. Please go through it before jumping directly into the topic.

### GCP Learning Series: Cloud Console and Cloud Shell

Google cloud platform can be used in various ways through API, cloud shell and cloud console. Most...  
medium.com



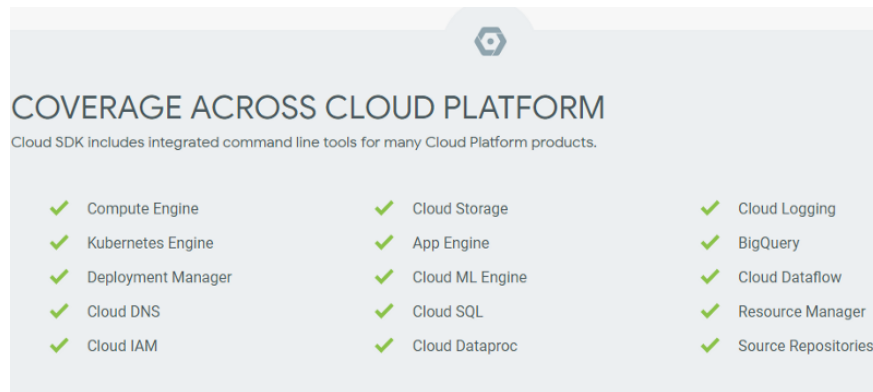
In this blog, we will go through the installation of Cloud SDK in local computer/laptop. We will also see how you can install cloud SDK in compute engine inside the Google Cloud Platform.

Cloud SDK is a set of tools in GCP for managing the cloud. SDK basically contains gcloud, gsutil and bq commands that can be used to access and manage the compute engine, cloud storage, big query, and other product and services from command line environment. You can also use automated scripts for managing these instances.

Managing virtual machines from gcloud is the easiest way to perform various tasks in VM.

gcloud can also be used to manage networks, firewalls, storage, and more without having to use the console. With gcloud, managing configurations for your Compute Engine environment are just a few keystrokes away.

Cloud SDK covers a wide range of services in GCP.



coverage

You can use various commands to manage all the products in GCP.

### **gcloud Tool**

gcloud manages authentication, local configuration, developer workflow, and interactions with the Cloud Platform APIs.

### **gsutil Tool**

gsutil provides command line access to manage Cloud Storage buckets and objects.

### **Powershell cmdlets (Windows)**

Google Cloud Tools for PowerShell is a collection of Windows PowerShell cmdlets for managing Google Cloud Platform resources within the Windows PowerShell environment.

### **bq Tool**

bq allows you to run queries, manipulate datasets, tables, and entities in BigQuery through the command line.

### **kubectl Tool**

kubectl orchestrates the deployment and management of Kubernetes container clusters on gcloud.

---

You can install cloud SDK in the following systems

- a. Linux
- b. Mac OS
- c. Windows

**Along with this, you may also use browser-based SDK which gives you 5 GB of persistent disk.**

In all cases, the main motive is to use commands available in SDK from different platforms.

Let us install Cloud SDK in Linux.

## Linux [Red Hat]

You don't need to download manually and install cloud SDK in Linux.

You should install repository prior to installing SDK.

The command for installing repo is :

```
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
```

After that command for installing the cloud SDK is

```
#yum install google-cloud-sdk
```

Additional components that contain dependencies for various commands are also available in the SDK, but not installed by default.

ID	Name	Description
app-engine-python	gcloud App Python Extensions	Google App Engine SDK packages for Python. Installed when you use the <code>gcloud app deploy</code> command to deploy a Python application to App Engine.
app-engine-java	gcloud App Java Extensions	Google App Engine SDK packages for Java. Installed when you use the <code>gcloud app deploy</code> command to deploy a Java application to App Engine.
app-engine-go	gcloud App Go Extensions	Google App Engine SDK packages for Go. Installed when you use the <code>gcloud app deploy</code> command to deploy a Go application to App Engine.
gcd-emulator	Cloud Datastore Emulator	Emulator for Google Cloud Datastore. Installed when you use commands in the <code>gcloud beta emulators datastore</code> command group.
pubsub-emulator	Cloud Pub/Sub Emulator	Emulator for Google Cloud Pub/Sub. Installed when you use commands in the <code>gcloud beta emulators pubsub</code> command group.
cbt	Cloud Bigtable CLI	Google Cloud Bigtable command line tool
cloud-build-local	Cloud Build Local Builder	Google Cloud Build Local Builder
bigtable	Cloud Bigtable Emulator	Emulator for Google Cloud Bigtable. Installed when you use commands in the <code>gcloud beta emulators bigtable</code> command group.
kubectl	kubectl	Kubernetes command-line tool. When you run commands in the <code>gcloud container</code> command group, you are prompted to install this component. <code>kubectl</code> is not required to create container clusters, but is required to run them.
docker-credential-gcr	docker-credential-gcr	Docker credential helper for Google Container Registry. After installing, enable the credential helper by running <code>docker-credential-gcr configure-docker</code> . For more information, <a href="#">see the project on GitHub</a> .
datalab	datalab	Cloud Datalab Command Line Tool.
cloud-build-local	Google Cloud Build Local	Local builder for Cloud Build. After installing, you can run builds on

additional component

While using other package managers Cloud SDK can be packaged with the additional element

Cloud SDK Component	APT Package	YUM Package
app-engine-go	google-cloud-sdk-app-engine-go	google-cloud-sdk-app-engine-go
app-engine-java	google-cloud-sdk-app-engine-java	google-cloud-sdk-app-engine-java
app-engine-python	google-cloud-sdk-app-engine-python	google-cloud-sdk-app-engine-python
bigtable	google-cloud-sdk-bigtable-emulator	google-cloud-sdk-bigtable-emulator
cbt	google-cloud-sdk-cbt	google-cloud-sdk-cbt
cloud-build-local	google-cloud-sdk-cloud-build-local	google-cloud-sdk-cloud-build-local
datalab	google-cloud-sdk-datalab	google-cloud-sdk-datalab
docker-credential-gcr	Not yet available.	Not yet available.
gcd-emulator	google-cloud-sdk-datastore-emulator	google-cloud-sdk-datastore-emulator
kubect1	kubect1	kubect1
pubsub-emulator	google-cloud-sdk-pubsub-emulator	google-cloud-sdk-pubsub-emulator

other components

For example, the `google-cloud-sdk-app-engine-java` component can be installed as follows:

```
#yum install google-cloud-sdk-app-engine-java
```

## Linux[Debian]

The process to install Cloud SDK in Debian based Linux is similar to the red hat. This package contains the `gcloud` , `gcloud alpha` , `gcloud beta` , `gsutil` , and `bq` commands only.

***It does not include `kubect1` or the App Engine extensions required to deploy an application using `gc loud` commands***

1. Create an environment variable for the correct distribution:

```
export CLOUD_SDK_REPO="cloud-sdk-$(lsb_release -c -s)"
```

2. Add the Cloud SDK distribution URI as a package source:

```
echo "deb http://packages.cloud.google.com/apt
$CLOUD_SDK_REPO main" | sudo tee -a
/etc/apt/sources.list.d/google-cloud-sdk.list
```

**Note:** If you have `apt-transport-https` installed, you can use “https” instead of “HTTP” in this step.

3. Import the Google Cloud public key:

- `curl https://packages.cloud.google.com/apt/doc/apt-key.gpg | sudo apt-key add -`

**Troubleshooting Tip:** If you are unable to get the latest updates due to an expired key, [obtain the latest apt-get.gpg key file](#).

4. Update and install the Cloud SDK:

- `sudo apt-get update && sudo apt-get install google-cloud-sdk`

**Note:** For additional options, such as disabling prompts or dry runs, refer to the [apt-get man pages](#).

5. Optionally, install any of these [additional components](#):

- `google-cloud-sdk-app-engine-python`
- `google-cloud-sdk-app-engine-python-extras`
- `google-cloud-sdk-app-engine-java`
- `google-cloud-sdk-app-engine-go`
- `google-cloud-sdk-datalab`
- `google-cloud-sdk-datastore-emulator`
- `google-cloud-sdk-pubsub-emulator`
- `google-cloud-sdk-cbt`
- `google-cloud-sdk-cloud-build-local`
- `google-cloud-sdk-bigtable-emulator`
- `kubectl`

6. For example, the `google-cloud-sdk-app-engine-java` component can be installed as follows:

- `sudo apt-get install google-cloud-sdk-app-engine-java`

7. Run `gcloud init` to get started:

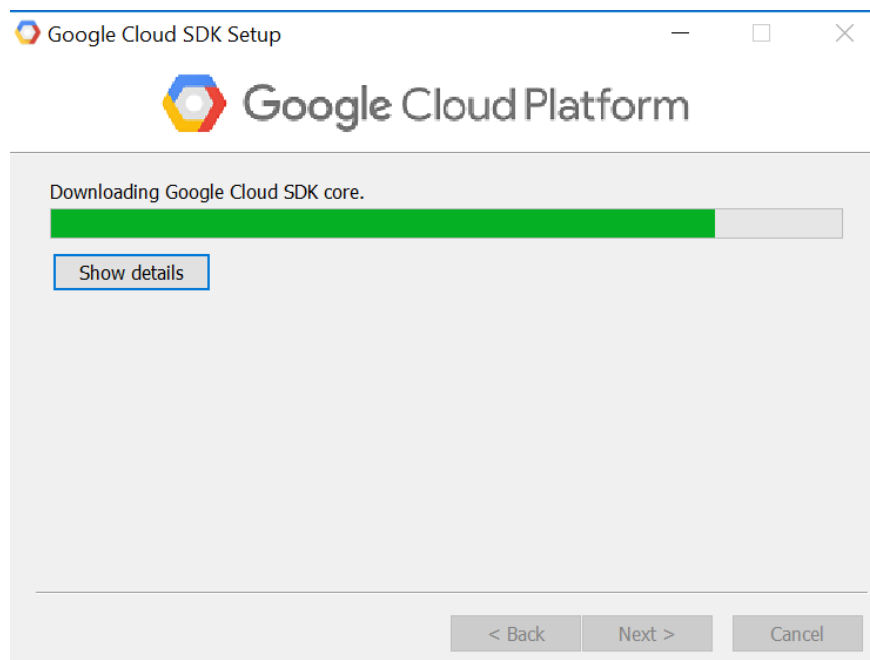
- `gcloud init`

## Windows

There are two approaches to installing cloud SDK on windows

1. Download the [Cloud SDK installer](#). The installer is signed by Google Inc.
2. Launch the installer and follow the prompts.

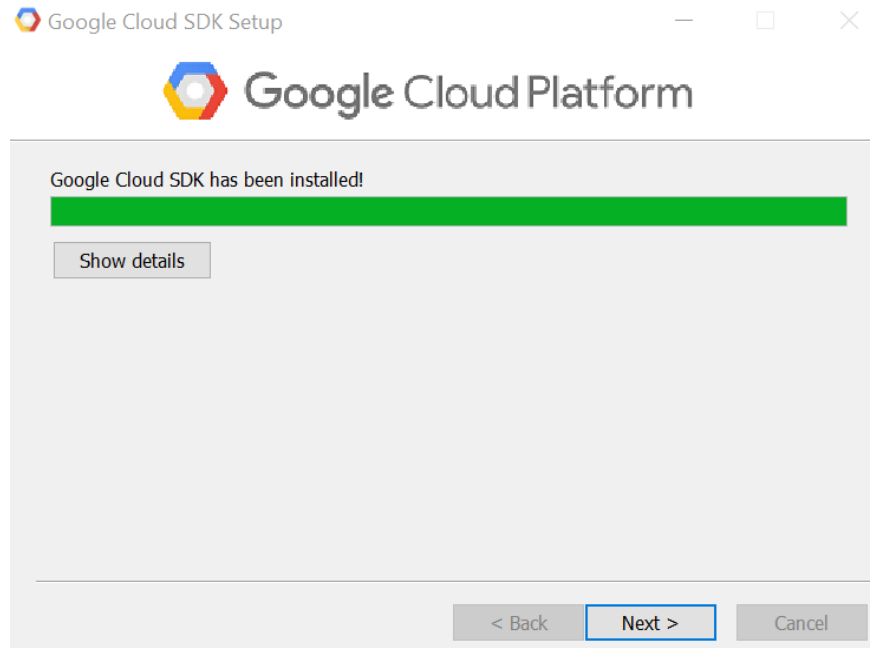
Cloud SDK requires Python 2 with a release version of Python 2.7.9 or later. The installer will install all necessary dependencies, including the needed Python version, by default. If you already have Python 2.x.y installed and want to use the existing installation, you can uncheck the option to **install Bundled Python**.



1. After installation has completed, accept the following options:
- **Start Cloud SDK Shell**

- **Run gcloud init**

2. The default installation does not include the App Engine extensions required to deploy an application using `gcloud` commands.



You can also install the latest version from a downloaded `.zip` file:

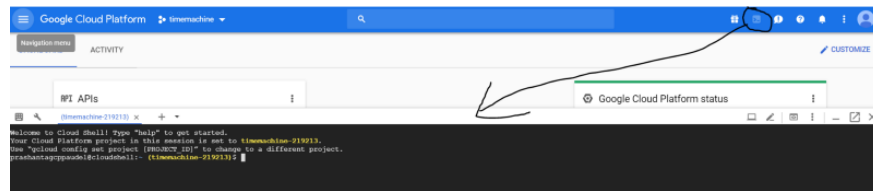
1. Download [google-cloud-sdk.zip](#) and extract its contents. (Right click on the downloaded file and select **Extract All**.)
2. Launch the `google-cloud-sdk\install.bat` script and follow the installation prompts.
3. When the installation finishes, restart the command prompt (`cmd.exe`).
4. Run `gcloud init` :

C:\> gcloud init

### Web-Browser Shell

After you are logged into the cloud console click on the command prompt icon in the right top of the page which will launch cloud shell with cloud SDK pre-installed. This is the best way to use google cloud SDK.





Sometimes it may take a while to launch depending on speed and browser but once loaded you can use all the tools like `gcloud`, `gsutil`, `bq` etc

Of all the tools and methods browser-based SDK seems to be the fastest and easiest way to work in GCP.

Now let's look at the commands in SDK.

Google says:

## Release levels

`gcloud` commands have the following release levels:

Release level | Label | Description | General Availability | None | Commands are considered fully stable and available for production use. Advance warnings will be made for commands that break current functionality and documented in the [release notes](#). **Beta** | `beta` | Commands are functionally complete, but may still have some outstanding issues. Breaking changes to these commands may be made without notice. **Alpha** | `alpha` | Commands are in early release and may change without notice. **Preview** | `preview` | Commands may be unstable and may change without notice.

The `alpha` and `beta` components are not installed by default when you install the SDK. You must [install these](#) separately using the `gcloud components install` command. If you try to run an alpha or beta command and the corresponding component is not installed, `gcloud` will prompt you to install it.

gcloud and gsutil commands are listed below.

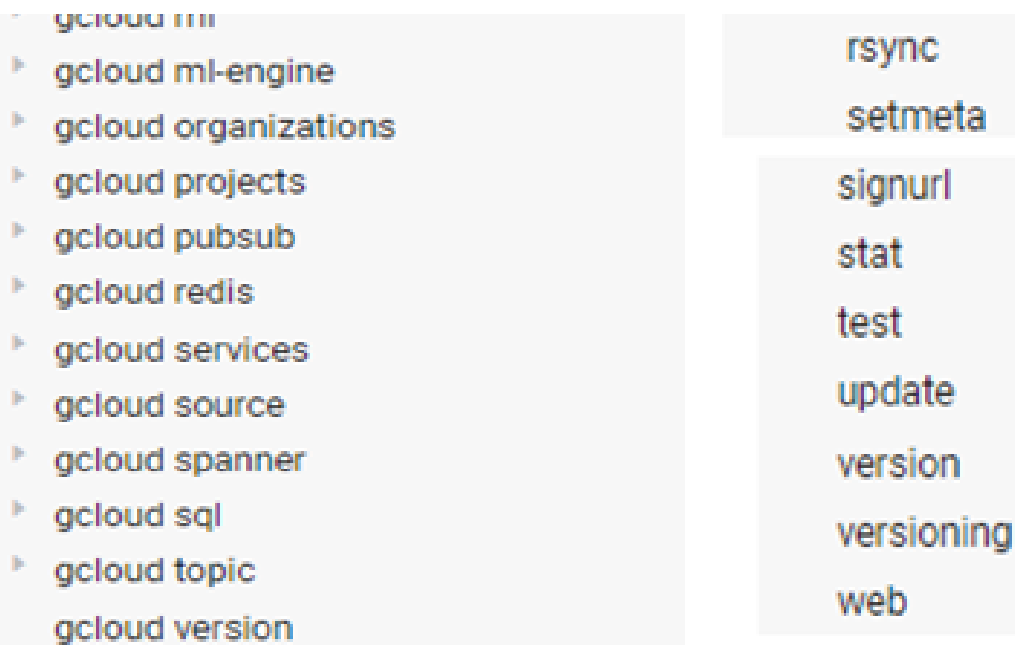
## gcloud Reference

### Overview

- gcloud alpha
- gcloud app
- gcloud auth
- gcloud beta
- gcloud bigtable
- gcloud builds
- gcloud components
- gcloud composer
- gcloud compute
- gcloud config
- gcloud container
- gcloud dataflow
- gcloud dataproc
- gcloud datastore
- gcloud debug
- gcloud deployment-manager
- gcloud dns
- gcloud docker
- gcloud domains
- gcloud endpoints
- gcloud feedback
- gcloud firebase
- gcloud functions
- gcloud help
- gcloud iam
- gcloud info
- gcloud init
- gcloud iot
- gcloud kms
- gcloud logging
- gcloud ml

## gsutil Commands

acl  
cat  
compose  
config  
cors  
cp  
defacl  
defstorageclass  
du  
hash  
help  
iam  
kms  
label  
lifecycle  
logging  
ls  
mb  
mv  
notification  
perfdiag  
rb  
requesterpays  
rewrite  
rm



```
gcloud ml
gcloud ml-engine
gcloud organizations
gcloud projects
gcloud pubsub
gcloud redis
gcloud services
gcloud source
gcloud spanner
gcloud sql
gcloud topic
gcloud version

rsync
setmeta

signurl
stat
test
update
version
versioning
web
```

gcloud commands

We will not go into deep in all the commands mentioned above but list some of them which are used frequently.

```
#gcloud --help -----> help page
#gcloud -h -----> Help for structuring command
#gcloud project create PROJECT_ID -----> create new
project
#gcloud project describe PROJECT_ID
#gcloud projects list -----> list projects
#gcloud project delete PROJECT_ID
#gcloud config set project PROJECT_ID -----> set project in
shell
```

You can see from the commands that google has structured commands in a way it is easier to do rather than remembering whole commands. There is always possible to find next portion of command by typing -h.

### GCP's New interactive CLI

To install interactive CLI in your default web browser shell.

```
$ gcloud components install alpha
$ gcloud alpha interactive
```

```
$ gcloud auth
  alpha
  app
  auth
  beta

The gcloud auth command group lets you grant and revoke authorization to Cloud
SDK (gcloud) to access Google Cloud Platform. Typically, when scripting Cloud
SDK tools for use on multiple machines, using gcloud auth
activate-service-account is recommended....

SYNOPSIS
  gcloud auth GROUP | COMMAND [--trace-token=TRACE_TOKEN]
  [G_CLOUD_WIDE_FLAG ...]
```

In this way we installed , configure and use various commands using google cloud SDK in CLI mode.

