**Week-6**

Installing and Understanding Eucalyptus cloud –II

1. Understand the concepts of Walrus Storage controller and Storage Controller (CLC) and in backend Node Controller (NC)

#### Ans- Walrus

Walrus is also called “WS3” and is the storage service provided by Eucalyptus. The Storage Service provides simple storage functionality, which is exposed by ReSTful and Soap APIs. Walrus takes care of storing the virtual machine images, storing the snapshots and serving Files. As with all other public facing Services in Eucalyptus, these Services are based on the Amazon Web Services API.  
Containers in Walrus Storage are called „Buckets“ and they have to be unique across accounts, just like it is with Amazon Web Services (AWS). Some naming restrictions are:

* Containers can contain lowercase letters, numbers, periods (.), underscores (\_), and dashes (-)
* Container Names must start with a number or letter
* The Length of a Name must be between 3 and 255 characters long
* It is not allowed to use an IP-Address as Name (e.g., 265.255.5.4)

The Maximum File Size in a Walrus Container is 5 Terabytes and Files can either be public or private. If the Container should be deleted, a container must be empty, which means that all files have to be deleted prior to deleting the container. Files are identified via unique Keys represented by Uniform Resource Identifiers (URIs).  
Common Actions performed on the Walrus storage are the creation of containers, store data in containers, download data, and grant or deny permissions. These actions can be performed via the eSTful or SOAP Interfaces. The Walrus Storage distinguishes two major read options: consistent read or eventually consistent read. The later one is faster but might server inconsistent data whereas the first one might have higher latency but data is always consistent.

#### Storage Controller

The Storage Controller is comparable to the Elastic Block Storage (EBS) for Amazon Web Services. Elastic Block Storage is a fast storage for virtual Image Files. The Storage Controller takes care of the creation of persistent EBS devices. Block Storage Devices are typically provided over over the ATAoverEthernet or iSCSI protocol to the instances.

**Week-7**

Create and Manage AWS Users and Groups, And Use Permissions to Allow and Deny their Access to AWS Resources by using IAM. Learn and Practice: AWS MFA for extra layer security on user

authentication**.**

# **Ans:- Creating an IAM user in your AWS account**

**Important**

**Topics**

* [Creating IAM users (console)](https://docs.aws.amazon.com/IAM/latest/UserGuide/id_users_create.html#id_users_create_console)
* [Creating IAM users (AWS CLI)](https://docs.aws.amazon.com/IAM/latest/UserGuide/id_users_create.html#id_users_create_cliwpsapi)
* [Creating IAM users (AWS API)](https://docs.aws.amazon.com/IAM/latest/UserGuide/id_users_create.html#id_users_create_api)

The process of creating a user and enabling that user to perform work tasks consists of the following steps:

1. Create the user in the AWS Management Console, the AWS CLI, Tools for Windows PowerShell, or using an AWS API operation. If you create the user in the AWS Management Console, then steps 1–4 are handled automatically, based on your choices. If you create the users programmatically, then you must perform each of those steps individually.

Create credentials for the user, depending on the type of access the user requires:

* + **Programmatic access:** The IAM user might need to make API calls, use the AWS CLI, or use the Tools for Windows PowerShell. In that case, create an access key (access key ID and a secret access key) for that user.
  + **AWS Management Console access**: If the user needs to access the AWS Management Console, [create a password for the user](https://docs.aws.amazon.com/IAM/latest/UserGuide/id_credentials_passwords_admin-change-user.html). Disabling console access for a user prevents them from signing in to the AWS Management Console using their user name and password. It does not change their permissions or prevent them from accessing the console using an assumed role.

As a best practice, create only the credentials that the user needs. For example, for a user who requires access only through the AWS Management Console, do not create access keys.

1. Give the user permissions to perform the required tasks by adding the user to one or more groups. You can also grant permissions by attaching permissions policies directly to the user. However, we recommend instead that you put your users in groups and manage permissions through policies that are attached to those groups. You can also use a [permissions boundary](https://docs.aws.amazon.com/IAM/latest/UserGuide/access_policies_boundaries.html) to limit the permissions that a user can have, though this is not common.
2. (Optional) Add metadata to the user by attaching tags. For more information about using tags in IAM, see [Tagging IAM resources](https://docs.aws.amazon.com/IAM/latest/UserGuide/id_tags.html).
3. Provide the user with the necessary sign-in information. This includes the password and the console URL for the account sign-in page where the user provides those credentials. For more information, see [How IAM users sign in to AWS](https://docs.aws.amazon.com/IAM/latest/UserGuide/id_users_sign-in.html).
4. (Optional) Configure [multi-factor authentication (MFA)](https://docs.aws.amazon.com/IAM/latest/UserGuide/id_credentials_mfa.html) for the user. MFA requires the user to provide a one-time-use code each time he or she signs into the AWS Management Console.
5. (Optional) Give users permissions to manage their own security credentials. (By default, users do not have permissions to manage their own credentials.) For more information, see [Permitting IAM users to change their own passwords](https://docs.aws.amazon.com/IAM/latest/UserGuide/id_credentials_passwords_enable-user-change.html).

For information about the permissions that you need in order to create a user, see [Permissions required to access IAM resources](https://docs.aws.amazon.com/IAM/latest/UserGuide/access_permissions-required.html).

## Creating IAM users (console)

You can use the AWS Management Console to create IAM users.

**To create one or more IAM users (console)**

1. Sign in to the AWS Management Console and open the IAM console at <https://console.aws.amazon.com/iam/>.
2. In the navigation pane, choose **Users** and then choose **Add users**.
3. Type the user name for the new user. This is the sign-in name for AWS. If you want to add multiple users, choose **Add another user** for each additional user and type their user names. You can add up to 10 users at one time.

**Note**

The number and size of IAM resources in an AWS account are limited. For more information, see [IAM and AWS STS quotas](https://docs.aws.amazon.com/IAM/latest/UserGuide/reference_iam-quotas.html). User names can be a combination of up to 64 letters, digits, and these characters: plus (+), equal (=), comma (,), period (.), at sign (@), underscore (\_), and hyphen (-). Names must be unique within an account. They are not distinguished by case. For example, you cannot create two users named TESTUSER and testuser.

1. Select the type of access this set of users will have. You can select programmatic access, access to the AWS Management Console, or both.
   * Select **Programmatic access** if the users require access to the API, AWS CLI, or Tools for Windows PowerShell. This creates an access key for each new user. You can view or download the access keys when you get to the **Final** page.
   * Select **AWS Management Console access** if the users require access to the AWS Management Console. This creates a password for each new user.
   * For **Console password**, choose one of the following:
     + **Autogenerated password**. Each user gets a randomly generated password that meets the [account password policy](https://docs.aws.amazon.com/IAM/latest/UserGuide/id_credentials_passwords_account-policy.html). You can view or download the passwords when you get to the **Final** page.
     + **Custom password**. Each user is assigned the password that you type in the box.
   * (Optional) We recommend that you select **Require password reset** to ensure that users are forced to change their password the first time they sign in.

**Note**

If an administrator has enabled [the **Allow users to change their own password** account password policy setting](https://console.aws.amazon.com/iam/home?#/account_settings), then this check box does nothing. Otherwise, it automatically attaches an AWS managed policy named [IAMUserChangePassword](https://console.aws.amazon.com/iam/home" \l "policies/arn:aws:iam::aws:policy/IAMUserChangePassword" \t "_blank) to the new users. The policy grants them permission to change their own passwords.

1. Choose **Next: Permissions**.
2. On the **Set permissions** page, specify how you want to assign permissions to this set of new users. Choose one of the following three options:
   * **Add user to group**. Choose this option if you want to assign the users to one or more groups that already have permissions policies. IAM displays a list of the groups in your account, along with their attached policies. You can select one or more existing groups, or choose **Create group** to create a new group. For more information, see [Changing permissions for an IAM user](https://docs.aws.amazon.com/IAM/latest/UserGuide/id_users_change-permissions.html).
   * **Copy permissions from existing user**. Choose this option to copy all of the group memberships, attached managed policies, embedded inline policies, and any existing [permissions boundaries](https://docs.aws.amazon.com/IAM/latest/UserGuide/access_policies_boundaries.html) from an existing user to the new users. IAM displays a list of the users in your account. Select the one whose permissions most closely match the needs of your new users.
   * **Attach existing policies directly**. Choose this option to see a list of the AWS managed and customer managed policies in your account. Select the policies that you want to attach to the new users or choose **Create policy** to open a new browser tab and create a new policy from scratch. For more information, see step 4 in the procedure [Creating IAM policies](https://docs.aws.amazon.com/IAM/latest/UserGuide/access_policies_create-console.html#access_policies_create-start). After you create the policy, close that tab and return to your original tab to add the policy to the new user. As a best practice, we recommend that you instead attach your policies to a group and then make users members of the appropriate groups.
3. (Optional) Set a [permissions boundary](https://docs.aws.amazon.com/IAM/latest/UserGuide/access_policies_boundaries.html). This is an advanced feature.

Open the **Set permissions boundary** section and choose **Use a permissions boundary to control the maximum user permissions**. IAM displays a list of the AWS managed and customer managed policies in your account. Select the policy to use for the permissions boundary or choose **Create policy** to open a new browser tab and create a new policy from scratch. For more information, see step 4 in the procedure [Creating IAM policies](https://docs.aws.amazon.com/IAM/latest/UserGuide/access_policies_create-console.html#access_policies_create-start). After you create the policy, close that tab and return to your original tab to select the policy to use for the permissions boundary.

1. Choose **Next: Tags**.
2. (Optional) Add metadata to the user by attaching tags as key-value pairs. For more information about using tags in IAM, see [Tagging IAM resources](https://docs.aws.amazon.com/IAM/latest/UserGuide/id_tags.html).
3. Choose **Next: Review** to see all of the choices you made up to this point. When you are ready to proceed, choose **Create user**.
4. To view the users' access keys (access key IDs and secret access keys), choose **Show** next to each password and access key that you want to see. To save the access keys, choose **Download .csv** and then save the file to a safe location.

**Important**

This is your only opportunity to view or download the secret access keys, and you must provide this information to your users before they can use the AWS API. Save the user's new access key ID and secret access key in a safe and secure place. **You will not have access to the secret keys again after this step.**

1. Provide each user with his or her credentials. On the final page you can choose **Send email** next to each user. Your local mail client opens with a draft that you can customize and send. The email template includes the following details to each user:
   * User name
   * URL to the account sign-in page. Use the following example, substituting the correct account ID number or account alias:

https://*AWS-account-ID or alias*.signin.aws.amazon.com/console

1. For more information, see [How IAM users sign in to AWS](https://docs.aws.amazon.com/IAM/latest/UserGuide/id_users_sign-in.html).
2. **Important**
3. The user's password is **not** included in the generated email. You must provide them to the customer in a way that complies with your organization's security guidelines.

## Creating IAM users (AWS CLI)

You can use the AWS CLI to create an IAM user.

**To create an IAM user (AWS CLI)**

1. Create a user.
   * [aws iam create-user](https://docs.aws.amazon.com/cli/latest/reference/iam/create-user.html)
2. (Optional) Give the user access to the AWS Management Console. This requires a password. You must also give the user the [URL of your account's sign-in page.](https://docs.aws.amazon.com/IAM/latest/UserGuide/id_users_sign-in.html)
   * [aws iam create-login-profile](https://docs.aws.amazon.com/cli/latest/reference/iam/create-login-profile.html)
3. (Optional) Give the user programmatic access. This requires access keys.
   * [aws iam create-access-key](https://docs.aws.amazon.com/cli/latest/reference/iam/create-access-key.html)
   * Tools for Windows PowerShell: [New-IAMAccessKey](https://docs.aws.amazon.com/powershell/latest/reference/Index.html?page=New-IAMAccessKey.html&tocid=New-IAMAccessKey)
   * IAM API: [CreateAccessKey](https://docs.aws.amazon.com/IAM/latest/APIReference/API_CreateAccessKey.html)

**Important**

This is your only opportunity to view or download the secret access keys, and you must provide this information to your users before they can use the AWS API. Save the user's new access key ID and secret access key in a safe and secure place. **You will not have access to the secret keys again after this step.**

1. Add the user to one or more groups. The groups that you specify should have attached policies that grant the appropriate permissions for the user.
   * [aws iam add-user-to-group](https://docs.aws.amazon.com/cli/latest/reference/iam/add-user-to-group.html)
2. (Optional) Attach a policy to the user that defines the user's permissions. **Note:** We recommend that you manage user permissions by adding the user to a group and attaching a policy to the group instead of attaching directly to a user.
   * [aws iam attach-user-policy](https://docs.aws.amazon.com/cli/latest/reference/iam/attach-user-policy.html)
3. (Optional) Add custom attributes to the user by attaching tags. For more information, see [Managing tags on IAM users (AWS CLI or AWS API)](https://docs.aws.amazon.com/IAM/latest/UserGuide/id_tags_users.html#id_tags_users_procs-cli-api).
4. (Optional) Give the user permission to manage their own security credentials. For more information, see [AWS: Allows MFA-authenticated IAM users to manage their own credentials on the My Security Credentials page](https://docs.aws.amazon.com/IAM/latest/UserGuide/reference_policies_examples_aws_my-sec-creds-self-manage.html).

## Creating IAM users (AWS API)

You can use the AWS API to create an IAM user.

**To create an IAM user from the (AWS API)**

1. Create a user.
   * [CreateUser](https://docs.aws.amazon.com/IAM/latest/APIReference/API_CreateUser.html)
2. (Optional) Give the user access to the AWS Management Console. This requires a password. You must also give the user the [URL of your account's sign-in page.](https://docs.aws.amazon.com/IAM/latest/UserGuide/id_users_sign-in.html)
   * [CreateLoginProfile](https://docs.aws.amazon.com/IAM/latest/APIReference/API_CreateLoginProfile.html)
3. (Optional) Give the user programmatic access. This requires access keys.
   * [CreateAccessKey](https://docs.aws.amazon.com/IAM/latest/APIReference/API_CreateAccessKey.html)

**Important**

This is your only opportunity to view or download the secret access keys, and you must provide this information to your users before they can use the AWS API. Save the user's new access key ID and secret access key in a safe and secure place. **You will not have access to the secret keys again after this step.**

1. Add the user to one or more groups. The groups that you specify should have attached policies that grant the appropriate permissions for the user.
   * [AddUserToGroup](https://docs.aws.amazon.com/IAM/latest/APIReference/API_AddUserToGroup.html)
2. (Optional) Attach a policy to the user that defines the user's permissions. **Note:** We recommend that you manage user permissions by adding the user to a group and attaching a policy to the group instead of attaching directly to a user.
   * [AttachUserPolicy](https://docs.aws.amazon.com/IAM/latest/APIReference/API_AttachUserPolicy.html)
3. (Optional) Add custom attributes to the user by attaching tags. For more information, see [Managing tags on IAM users (AWS CLI or AWS API)](https://docs.aws.amazon.com/IAM/latest/UserGuide/id_tags_users.html#id_tags_users_procs-cli-api).
4. (Optional) Give the user permission to manage their own security credentials. For more information, see [AWS: Allows MFA-authenticated IAM users to manage their own credentials on the My Security Credentials page](https://docs.aws.amazon.com/IAM/latest/UserGuide/reference_policies_examples_aws_my-sec-creds-self-manage.html).

**Week-8**

Write Case study on Amazon S3.

1. Discuss technical description and organizational usage with its impact.

### Ans- What is Amazon S3?

Amazon Simple Storage Service (Amazon S3) is a scalable, high-speed, web-based cloud storage service. The service is designed for online [backup](https://www.techtarget.com/searchdatabackup/definition/backup) and [archiving](https://www.techtarget.com/searchstorage/definition/archive) of data and [applications](https://www.techtarget.com/searchsoftwarequality/definition/application) on Amazon Web Services (AWS). Amazon S3 was designed with a minimal feature set and created to make web-scale computing easier for developers.

### Amazon S3 features

S3 provides 99.999999999% durability for objects stored in the service and supports multiple security and compliance certifications. An administrator can also link S3 to other AWS security and monitoring services, including [CloudTrail](https://searchcloudsecurity.techtarget.com/definition/AWS-CloudTrail), [CloudWatch](https://www.techtarget.com/searchaws/definition/CloudWatch) and [Macie](https://www.techtarget.com/searchaws/definition/Amazon-Macie). There's also an extensive partner network of vendors that link their services directly to S3.

Data can be transferred to S3 over the public internet via access to S3 application programming interfaces ([APIs](https://www.techtarget.com/searchapparchitecture/definition/application-program-interface-API)). There's also Amazon S3 Transfer Acceleration for faster movement over long distances, as well as AWS Direct Connect for a private, consistent connection between S3 and an enterprise's own data center. An administrator can also use AWS Snowball, a physical transfer device, to ship large amounts of data from an enterprise data center directly to AWS, which will then upload it to S3.

In addition, users can integrate other AWS services with S3. For example, an analyst can query data directly on S3 either with [Amazon Athena](https://www.techtarget.com/searchaws/definition/Amazon-Athena) for ad hoc queries or with [Amazon Redshift Spectrum](https://www.techtarget.com/searchaws/definition/Amazon-Redshift-Spectrum) for more complex analyses.

### Amazon S3 use cases

Amazon S3 can be used by organizations ranging in size from small businesses to large enterprises. S3's scalability, availability, security and performance capabilities make it suitable for a variety of data storage use cases. Common use cases for S3 include the following:

* data storage;
* data archiving;
* application [hosting](https://www.techtarget.com/searchstorage/definition/cloud-hosting) for deployment, installation and management of web apps;
* software delivery;
* data backup;
* disaster recovery ([DR](https://www.techtarget.com/searchdisasterrecovery/definition/disaster-recovery));
* running [big data analytics](https://www.techtarget.com/searchbusinessanalytics/definition/big-data-analytics) tools on stored data;
* [data lakes](https://www.techtarget.com/searchaws/definition/data-lake);
* mobile applications;
* internet of things ([IoT](https://internetofthingsagenda.techtarget.com/definition/Internet-of-Things-IoT)) devices;
* media hosting for images, videos and music files; and
* website hosting -- particularly well suited to work with [Amazon CloudFront](https://www.techtarget.com/searchaws/definition/Amazon-CloudFront) for content delivery.

### How Amazon S3 works

Amazon S3 is an [object storage](https://www.techtarget.com/searchstorage/definition/object-storage) service, which differs from other types of cloud computing storage types, such as [block](https://www.techtarget.com/searchstorage/definition/block-storage) and [file](https://www.techtarget.com/searchstorage/definition/cloud-file-storage) storage. Each object is stored as a file with its metadata included. The object is also given an ID number. Applications use this ID number to access objects. This is unlike file and block cloud storage, where a developer can access an object via a [representational state transfer (REST) API](https://www.techtarget.com/searchapparchitecture/definition/RESTful-API).

The S3 object storage cloud service gives a subscriber access to the same systems that Amazon uses to run its own websites. S3 enables customers to upload, store and download practically any file or object that is up to 5 terabytes (TB) in size -- with the largest single upload capped at 5 gigabytes ([GB](https://www.techtarget.com/searchstorage/definition/gigabyte)).

### Amazon S3 storage classes

Amazon S3 comes in seven storage classes:

1. **S3 Standard** is suitable for frequently accessed data that needs to be delivered with low latency and high throughput. S3 Standard targets applications, dynamic websites, content distribution and big data workloads.
2. **S3 Intelligent-Tiering**is most suitable for data with access needs that are either changing or unknown. S3 Intelligent-Tiering has four different access tiers: Frequent Access, Infrequent Access (IA), Archive and Deep Archive. Data is automatically moved to the most inexpensive storage tier according to customer access patterns.
3. **S3 Standard-IA** offers a lower storage price for data that is needed less often but that must be quickly accessible. This tier can be used for backups, DR and long-term data storage.
4. **S3 One Zone-IA** is designed for data that is used infrequently but requires rapid access on the occasions that it is needed. Use of S3 One Zone-IA is indicated for infrequently accessed data without high resilience or availability needs, data that is able to be recreated and backing up on-premises data.
5. **S3 Glacier** is the least expensive storage option in S3, but it is strictly designed for archival storage because it takes longer to access the data. Glacier offers variable retrieval rates that range from minutes to hours.
6. **S3 Glacier Deep Archive**has the lowest price option for S3 storage. S3 Glacier Deep Archive is designed to retain data that only needs to be accessed once or twice a year.
7. **S3 Outposts**adds S3 object storage features and APIs to an on-premises [AWS Outposts](https://www.techtarget.com/searchcloudcomputing/tip/What-you-need-to-know-to-deploy-an-AWS-hybrid-cloud-with-Outposts) environment. S3 Outposts is best used when performance needs call for data to be stored near on-premises applications or to satisfy specific data residency requirements.

A user can also implement lifecycle management policies to curate data and move it to the most appropriate tier over time.

### Working with S3 buckets

Amazon does not impose a limit on the number of items that a subscriber can store; however, there are limits to Amazon [S3 bucket](https://www.techtarget.com/searchaws/definition/AWS-bucket) quantities. Each AWS account allows up to 100 buckets to be created; limits can be increased to 1,000 with service limit increases. An Amazon S3 bucket exists within a particular region of the cloud. An AWS customer can use an Amazon S3 API to upload objects to a particular bucket. Customers can configure and manage S3 buckets.

### Protecting your S3 data

User data is stored on [redundant](https://www.techtarget.com/searchstorage/definition/redundant) servers in multiple data centers. S3 uses a simple web-based interface -- the Amazon S3 console – and [encryption](https://www.techtarget.com/searchsecurity/definition/encryption) for user [authentication](https://www.techtarget.com/searchsecurity/definition/authentication).

S3 buckets are kept private from public access by default, but an administrator can choose to make them publicly accessible. A user can also [encrypt data prior to storage](https://www.techtarget.com/searchstorage/answer/What-are-3-encryption-for-cloud-storage-best-practices). Rights may be specified for individual users, who will then need approved AWS credentials to download or access a file in S3.

When a user stores data in S3, Amazon tracks the usage for billing purposes, but it does not otherwise access the data unless required to do so by law.

### Competitor services

Competitor services to Amazon S3 include other object storage software tool services. Major cloud service providers such as Google, Microsoft, IBM and Alibaba offer [comparable object storage services](https://www.techtarget.com/searchstorage/feature/AWS-vs-Azure-vs-Google-pricing-and-features-compared). Main competitor services to Amazon S3 include the following:

* Google Cloud Storage
* Azure Blob storage
* IBM Cloud Object Storage
* DigitalOcean Spaces
* Alibaba Cloud Object Storage Service (OSS)
* Cloudian
* Zadara Storage
* Oracle Cloud Infrastructure Object Storage

**Week-9**

Installation and Understanding the features of Google app engine.

* 1. Understand how Google App Engine is to be installed and work with Python.

Ans- Before begin

1. Create a Google Cloud Platform project, if you don't have one already.

2. Make sure that Python 2.7 is installed on your system:

python -V

Note: As of Cloud SDK version 206.0.0, the gcloud CLI has experimental support for

running using a Python 3.4+ interpreter (run gcloud topic startup for exclusions and more

information on configuring your Python interpreter). All other Cloud SDK tools still require

a Python 2.7 interpreter.

3. Download the archive file best suited to your operating system. Most machines will run

the 64-bit package. If you'd like to check, run uname -m to verify if you're running a 64-bit

system.

Platform Package Size SHA256 Checksum

Linux

64-bit

(x86\_64)

google-cloud-sdk229.0.0-linuxx86\_64.tar.gz

25.6

MB

b1c87fc9451598a76cf66978dd8aa06482bfced639b56cf31559dc2c7f8b7b90

Linux

32-bit

(x86)

google-cloud-sdk229.0.0-linuxx86.tar.gz

25.2

MB

ee8c45f8018d0fee92b07c32cc6d8c891241da0b88bfe289d4e58e6746c3f668

Alternatively, to download the Linux 64-bit archive file from your command-line, run:

curl -O https://dl.google.com/dl/cloudsdk/channels/rapid/downloads/google-cloud-sdk229.0.0-linux-x86\_64.tar.gz

For the 32-bit archive file, run:

curl -O https://dl.google.com/dl/cloudsdk/channels/rapid/downloads/google-cloud-sdk229.0.0-linux-x86.tar.gz

4. Extract the archive to any location on your file system; preferably, your Home folder. On

Linux, you can extract the archive file by running this command:

tar zxvf [ARCHIVE\_FILE] google-cloud-sdk

5. If you're having trouble getting the gcloud command to work, ensure your $PATH is defined

appropriately. Use the install script to add Cloud SDK tools to your path. You will also be

able to opt-in to command-completion for your bash shell and usage statistics collection

during the installation process. Run the script using this command:

./google-cloud-sdk/install.sh

Restart your terminal for the changes to take effect.

Alternatively, you can call Cloud SDK after extracting the downloaded archive by invoking

its executables via the full path.

Initialize the SDK

Use the gcloud init command to perform several common SDK setup tasks. These include

authorizing the SDK tools to access Google Cloud Platform using your user account credentials and

setting up the default SDK configuration.

To initialize the SDK:

1. Run the following at a command prompt:

gcloud init

Note: To prevent the command from launching a web browser, use gcloud init --consoleonly instead. To authorize without a web browser and non-interactively, create a service

account with the appropriate scopes using the Google Cloud Platform Console and use

gcloud auth activate-service-account with the corresponding JSON key file.

2. Accept the option to log in using your Google user account:

To continue, you must log in. Would you like to log in (Y/n)? Y

3. In your browser, log in to your Google user account when prompted and click Allow to grant

permission to access Google Cloud Platform resources.

4. At the command prompt, select a Cloud Platform project from the list of those where you

have Owner, Editor or Viewer permissions:

Pick cloud project to use:

[1] [my-project-1]

[2] [my-project-2]

...

Please enter your numeric choice:

If you only have one project, gcloud init selects it for you.

5. If you have the Google Compute Engine API enabled, gcloud init allows you to choose a

default Compute Engine zone:

Which compute zone would you like to use as project default?

[1] [asia-east1-a]

[2] [asia-east1-b]

...

[14] Do not use default zone

Please enter your numeric choice:

gcloud init confirms that you have complete the setup steps successfully:

gcloud has now been configured!

You can use [gcloud config] to change more gcloud settings.

Your active configuration is: [default]

Run core gcloud commands

Run these gcloud commands to view information about your SDK installation:

1. To list accounts whose credentials are stored on the local system:

gcloud auth list

gcloud displays a list of credentialed accounts:

Credentialed Accounts

ACTIVE ACCOUNT

\* example-user-1@gmail.com

example-user-2@gmail.com

2. To list the properties in your active SDK configuration:

gcloud config list

gcloud displays the list of properties:

[core]

account = example-user-1@gmail.com

disable\_usage\_reporting = False

project = example-project

3. To view information about your Cloud SDK installation and the active SDK configuration:

gcloud info

gcloud displays a summary of information about your Cloud SDK installation. This includes

information about your system, the installed SDK components, the active user account and

current project, and the properties in the active SDK configuration.

4. To view information about gcloud commands and other topics from the command line:

gcloud help

For example, to view the help for gcloud compute instances create:

gcloud help compute instances create

gcloud displays a help topic that contains a description of the command, a list of command

flags and arguments, and examples of how to use it.

How to Run Program:

Now as we have finished installing app engine, now it’s time to create

and upload an app. In this case we will be taking example of a

“HELLO WORLD” app in python.

1. As we already have made sure that we have python installed in our system, It will be easier for us

to clone existing code and deploy it rather than creating our own so we will use python docssample. Run the command “git clone https://github.com/GoogleCloudPlatform/python-docssamples”.

2. cd python docs- samples/appengine/standard/hello\_world

3. dev\_appserver.py app.yaml

1. **Laboratory Equipment/Software/Tools Required**

* A computer System with Windows / Ubuntu Operating System.
* Technology / Tools Used – Virtualbox, Cloud Foundry, Open Stack, Eucalyptus Cloud AWS Amazon and google app engine.