**Lab Manual- SysOps with AWS CLI**

**Prepared for**:

**Date:** 18th Nov 2018

**Prepared by:**

Document Name: Lab Manual

**Document Number** SysOpsLab311

**Contributor:**

Sultan Talib

Table of Contents

[1 OBJECTIVE 3](#_Toc9015960)

[2 PRE-REQUISISTE 3](#_Toc9015961)

[3 Setup Up AWS CLI 3](#_Toc9015962)

[3.1 **Connect to AWS** 3](#_Toc9015963)

[3.2 **AWS EC2 Comamnd** 5](#_Toc9015964)

[3.3 **AWS EC2 Launch Instance** 10](#_Toc9015965)

[3.4 **AWS S3 Comamnd** 11](#_Toc9015966)

[3.5 **AWS IAM Command** 13](#_Toc9015967)

[3.6 **AWS EC2 Jason Script** 15](#_Toc9015968)

# OBJECTIVE

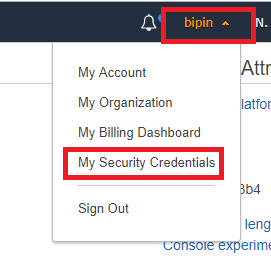
AWS CLI is a tool that pulls all the AWS services together in one central console, giving you easy control of multiple AWS services with a single tool. The acronym stands for Amazon Web Services Command Line Interface because, as its name suggests, users operate it from the command line. With it, you can control services manually or automate them with powerful scripts.In This Lab will cover the basics of configuring AWS cli.

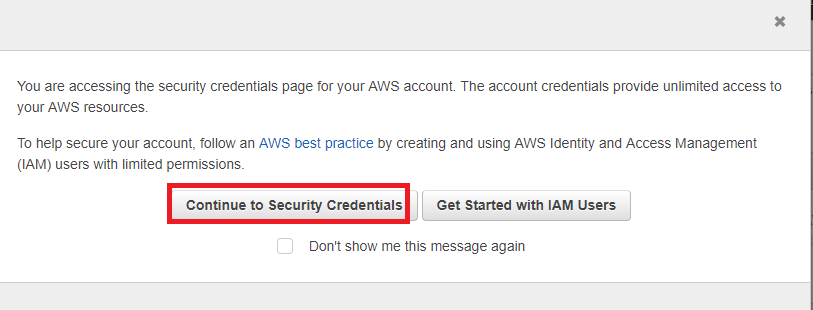
# PRE-REQUISISTE

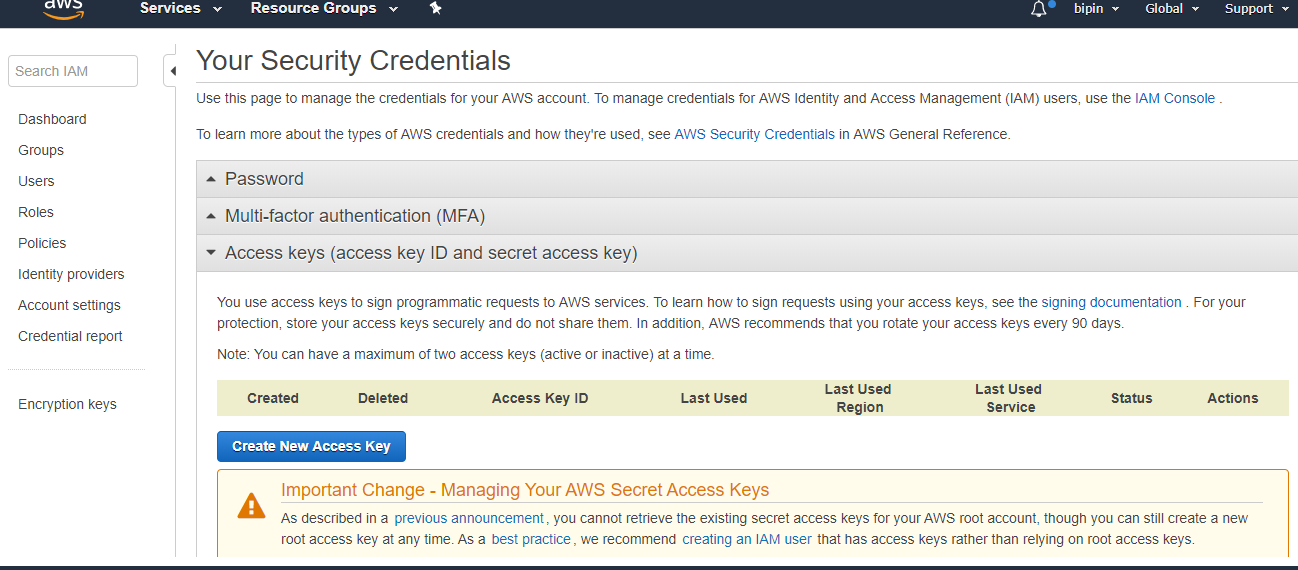
* Accounts in AWS
* A local Computer with 4 CPU, 16 GB RAM, 200 GB disk space

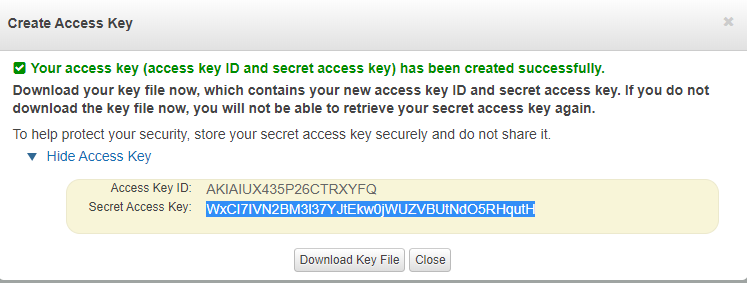
# Setup Up AWS CLI

## **Connect to AWS**

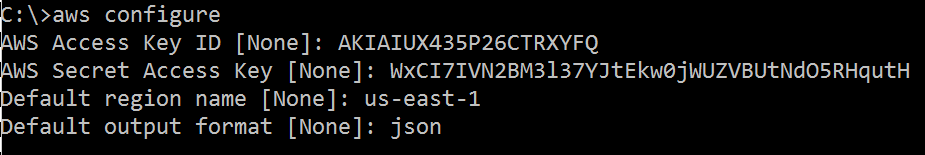






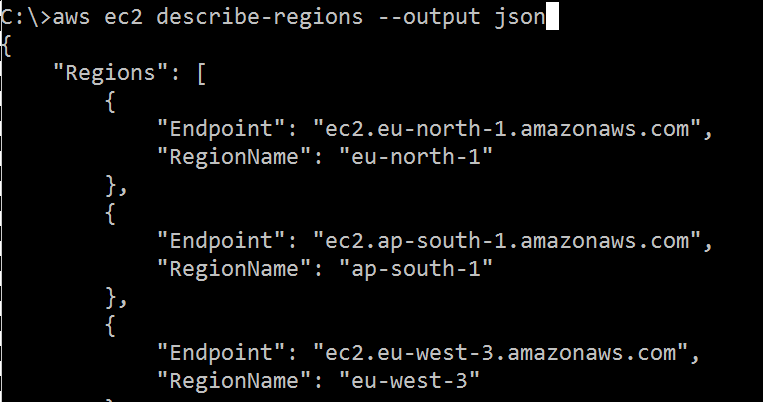


aws configure

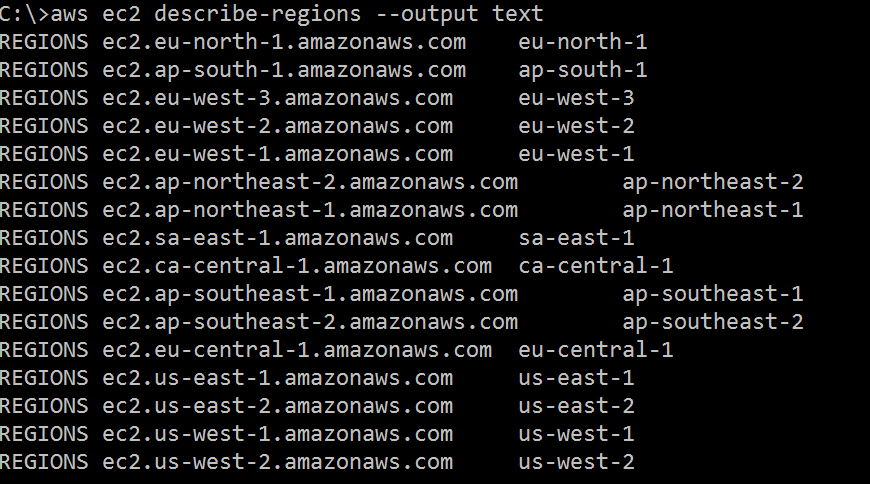


## **AWS EC2 Comamnd**

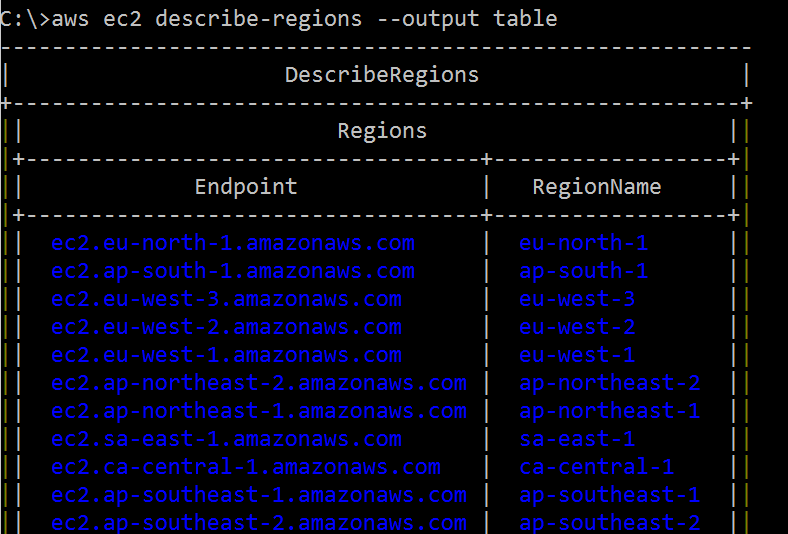
aws ec2 describe-regions --output json



aws ec2 describe-regions --output text



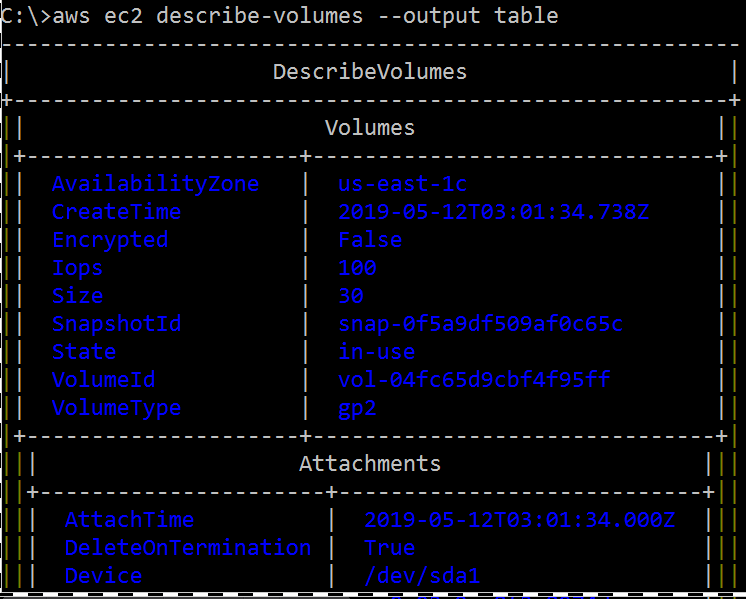
aws ec2 describe-regions --output table



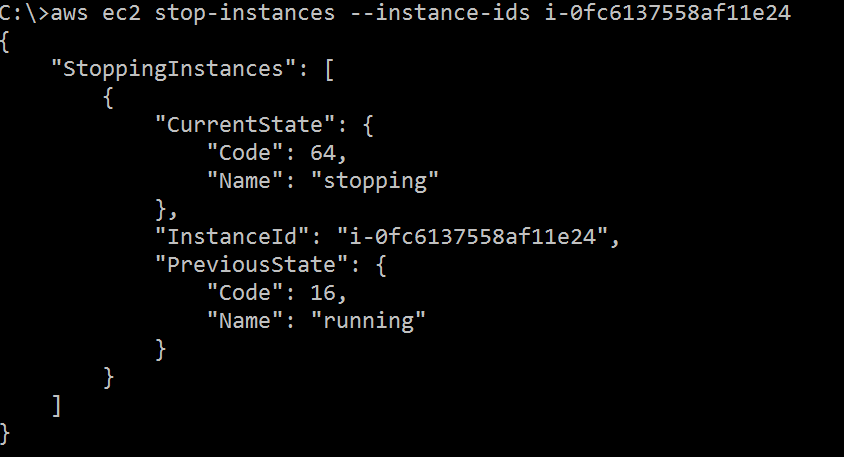
aws ec2 create-key-pair --key-name mykey



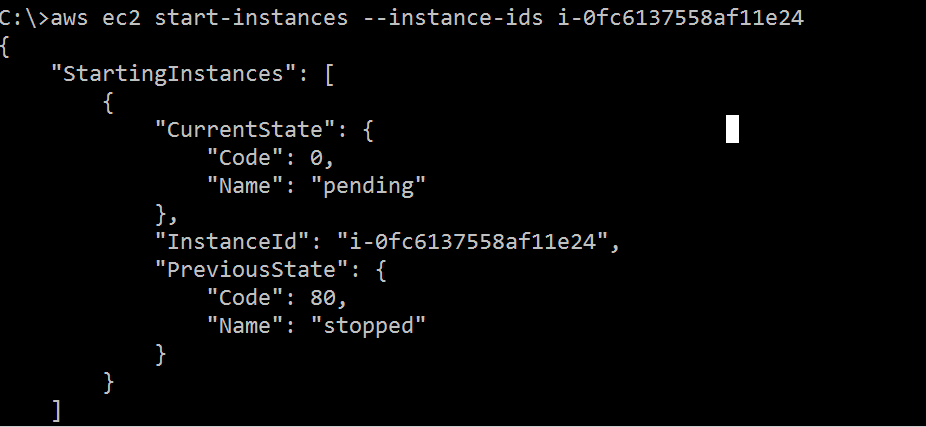
aws ec2 describe-volumes –output table



aws ec2 stop-instances --instance-ids i-0fc6137558af11e24



aws ec2 start-instances --instance-ids i-0fc6137558af11e24



if you want to start multiple instances using a single command, provide all the instance ids at the end as shown below.

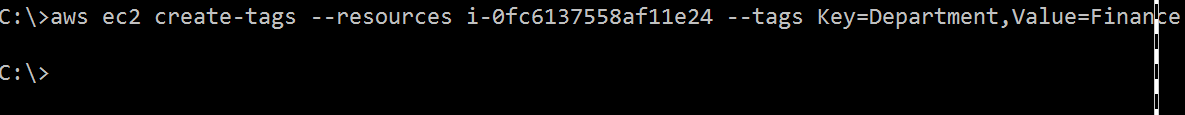
aws ec2 start-instances --instance-ids i-0fc6137558af11e24 i-0fc6137558af11e29

The following are the possible state name and state code for an instance:

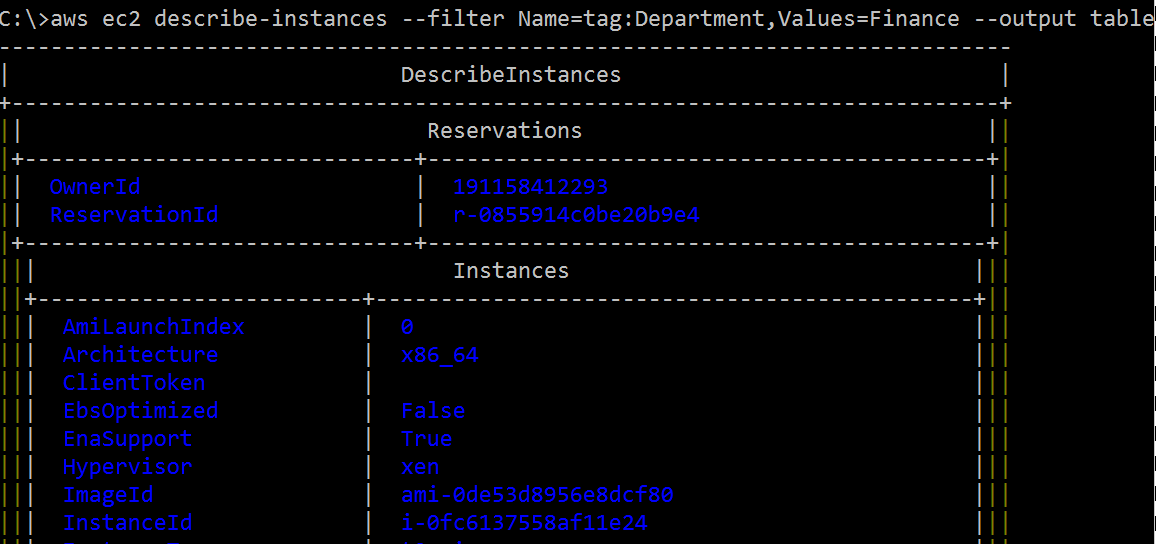
* 0 is for pending
* 16 is for running
* 32 is for shutting-down
* 48 is for terminated
* 64 is for stopping
* 80 is for stopped

The following “aws ec2 terminate-instances” command will terminate the instance that is specified in the –instance-ids field.

aws ec2 create-tags --resources i-0fc6137558af11e24 --tags Key=Department,Value=Finance

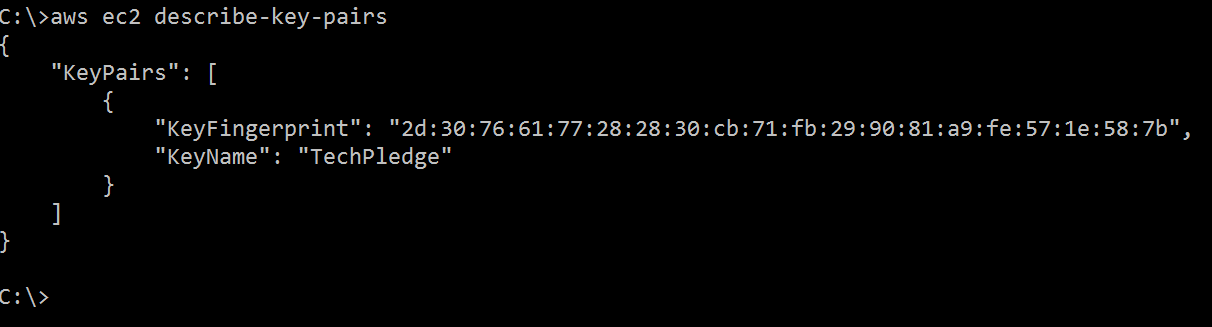


aws ec2 describe-instances --filter Name=tag:Department,Values=Finance –-output table

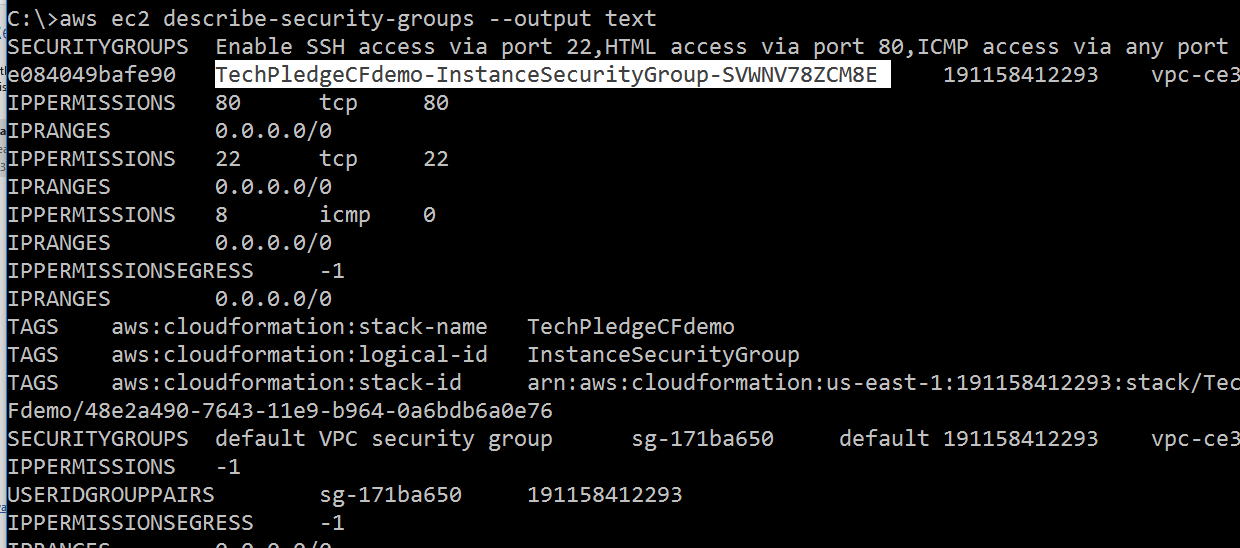


## **AWS EC2 Launch Instance**

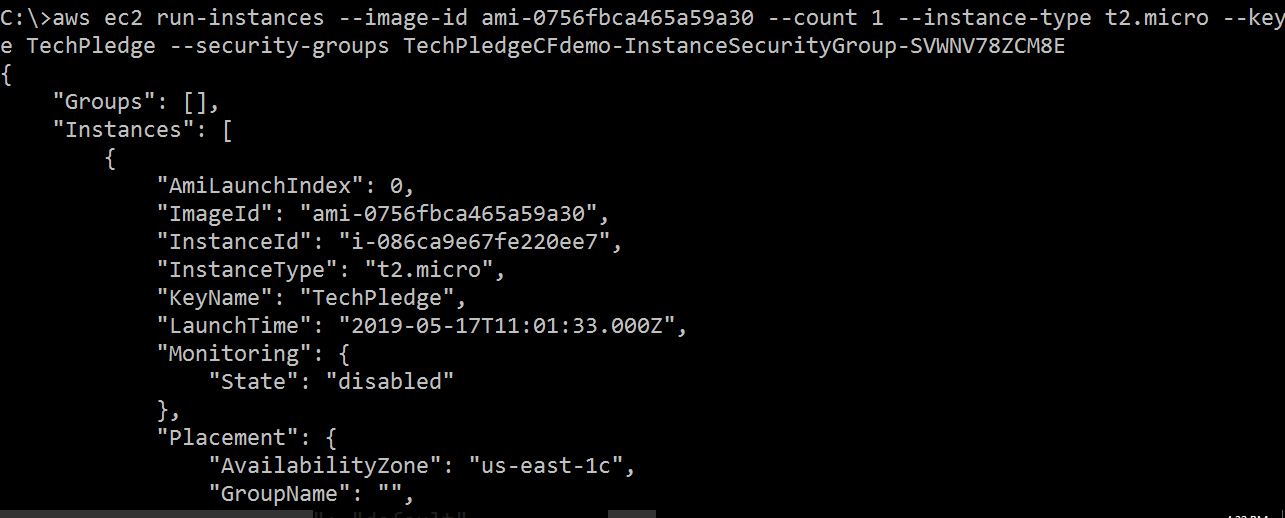
aws ec2 describe-key-pairs



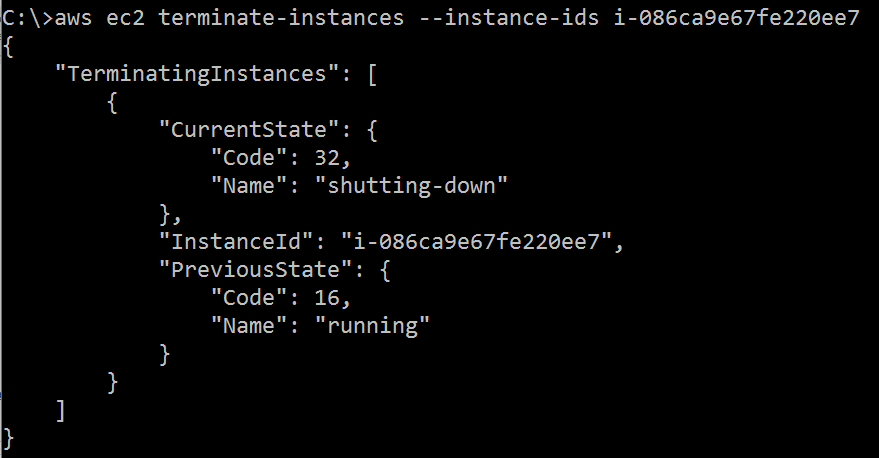
aws ec2 describe-security-groups --output text



aws ec2 run-instances --image-id ami-0756fbca465a59a30 --count 1 --instance-type t2.micro --key-name TechPledge --security-groups CarrerSG

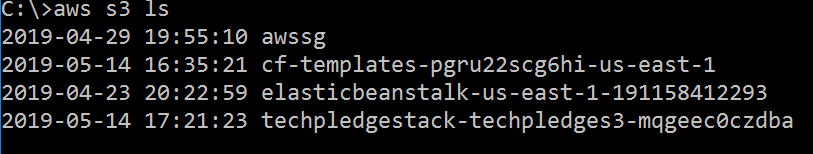


aws ec2 terminate-instances --instance-ids i-086ca9e67fe220ee7



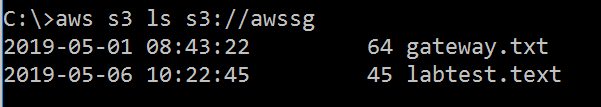
## **AWS S3 Comamnd**

aws s3 ls

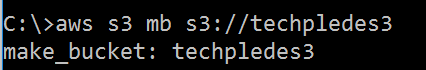


the following command lists all objects and folders in a bucket.

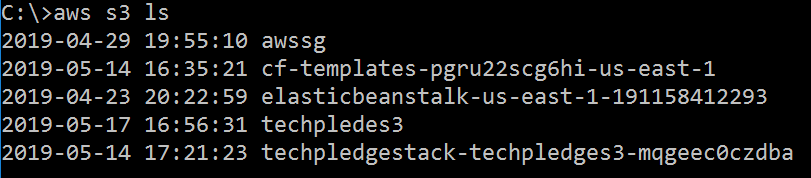
aws s3 ls s3://awssg

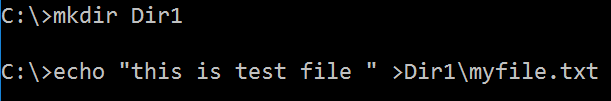


aws s3 mb s3://bipeen8989



aws s3 ls





aws s3 cp Dir1\myfile.txt s3://techpledes3



aws s3 ls s3://techpledes3



aws s3 rm s3://techpledes3/myfile.txt

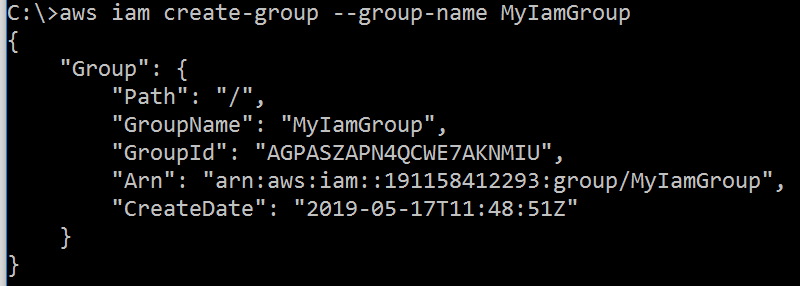


aws s3 rb s3://techpledes3/

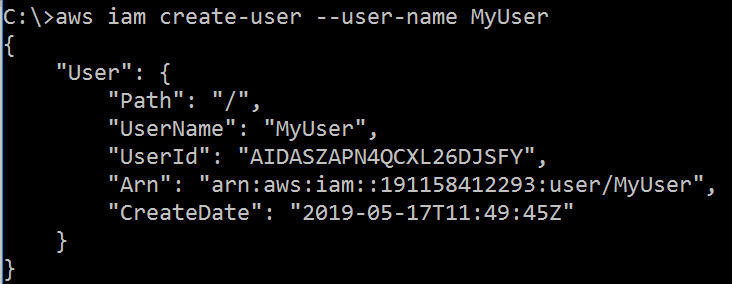


## **AWS IAM Command**

aws iam create-group --group-name MyIamGroup



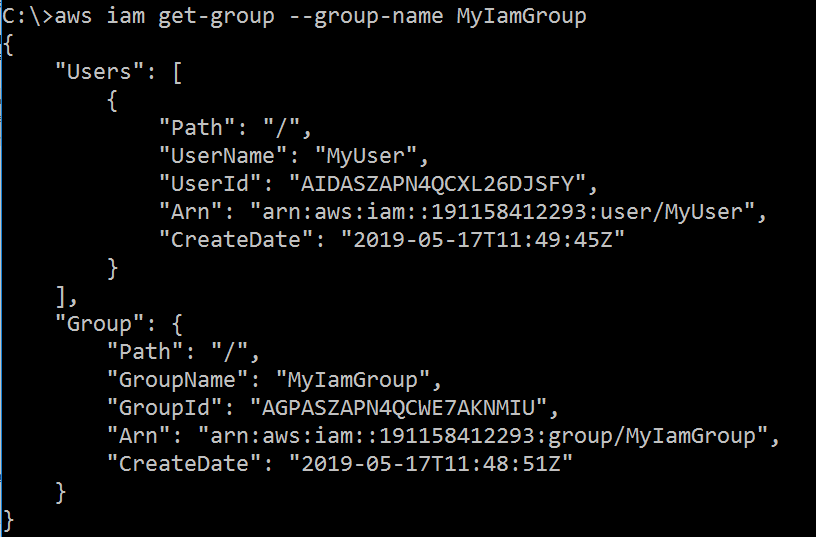
aws iam create-user --user-name MyUser



aws iam add-user-to-group --user-name MyUser --group-name MyIamGroup



aws iam get-group --group-name MyIamGroup



## **AWS EC2 Jason Script**

aws ec2 run-instances --generate-cli-skeleton >Dir1\test.json



aws ec2 run-instances --cli-input-json file://test2.json