



**Ganpat
University**

॥ विद्यया समाजोत्कर्षः ॥

**Institute of
Computer
Technology**

Name: Tushar Panchal

En.No: 21162101014

Sub: CCE(Cloud Computing Essentials)

Branch: CBA

Batch:71

PRACTICAL 13

Use AWS CLI and configure it.

Services to be configured through CLI:

- 1) S3
- 2) EC2
- 3) IAM
- 4) VPC

<https://awscli.amazonaws.com/AWSCLIV2.msi>

First you have to install the aws-cli using the above link.

And check for the version.

```
tushar@ROG in ~  
└─$ sudo snap install aws-cli --classic  
[sudo] password for tushar:  
Warning: /var/lib/snapd/snap/bin was not found in your $PATH. If you've not restarted your session  
since you installed snapd, try doing that. Please see https://forum.snapcraft.io/t/9469  
for more details.  
  
aws-cli (v2/stable) 2.22.2 from Amazon Web Services (aws**) installed  
  
tushar@ROG in ~ as 🐧 took 37s  
└─$ aws --version  
aws-cli/2.22.2 Python/3.12.6 Linux/6.11.9-zen1-1-zen exe/x86_64.garuda
```

After that go to security credentials and create Access key

The screenshot shows the AWS IAM console interface. On the left is the navigation menu with 'Identity and Access Management (IAM)' selected. The main content area is titled 'My security credentials' and includes sections for 'Account details', 'Multi-factor authentication (MFA) (1)', and 'Access keys (0)'. A green box highlights the 'Security credentials' link in the top right menu. Another green box highlights the 'Create access key' button in the 'Access keys (0)' section.

Hit the checkbox and hit create access key button

The screenshot shows the 'Create access key' page in the AWS IAM console. It includes a warning box about root user access keys and a checkbox labeled 'I understand creating a root access key is not a best practice, but I still want to create one.' A green box highlights the 'Create access key' button at the bottom right.

Now copy that access key and secret key to use it in CMD

The screenshot shows the 'Access key' page in the AWS IAM console. It displays the generated 'Access key' (AKIAQEIP3FOFT4NPBYKB) and 'Secret access key' (a masked string). A green box highlights both the access key and secret access key fields.

Now run below command to configure

```
tushar@ROG in ~ took 0s
λ aws configure
AWS Access Key ID [*****BYKB]: AKIAQEIP3FOFT4NPBYKB
AWS Secret Access Key [*****iW0e]: lw0/h2Gh/v6J1WHeRmRhJnmd+sDDxtqzCaLiW0e
Default region name [None]: ap-south-1
Default output format [None]: json
```

- **S3 :**

Run below command to create S3 Bucket:

```
tushar@ROG in ~ on ○ (ap-south-1) took 3m22s
λ aws s3 mb s3://tushar-s3-bucket-cli
make_bucket: tushar-s3-bucket-cli

tushar@ROG in ~ on ○ (ap-south-1) took 3s
λ aws s3 ls
2024-11-21 09:40:36 tushar-s3-bucket-cli
2024-11-14 05:43:46 user-code-storage
```

Now let's upload an image file

```
tushar@ROG in ~ on ○ (ap-south-1) took 1s
λ cd Pictures/

tushar@ROG in ~/Pictures on ○ (ap-south-1)
λ ls
.rw-r--r-- 115k tushar 19 Nov 00:11 2024-11-19_00-11.png
.rw-r--r-- 15k tushar 20 Nov 10:37 2024-11-20_10-37.png
.rw-r--r-- 1.9M tushar 21 Nov 09:44 ghost-call-of-duty-modern-warfare-2-2022-4k-wallpaper-uhdpaper.com-447@10h.jpg
```

Now run below command to upload your file

```
tushar@ROG in ~/Pictures on ○ (ap-south-1) took 0s
λ ls
.rw-r--r-- 1.9M tushar 21 Nov 09:44 ghost.jpg

tushar@ROG in ~/Pictures on ○ (ap-south-1) took 0s
λ aws s3 cp ghost.jpg s3://tushar-s3-bucket-cli
upload: ./ghost.jpg to s3://tushar-s3-bucket-cli/ghost.jpg
```

Now run below command to download file which we upload to our bucket in above command

```

tushar@ROG in ~/Pictures on ○ (ap-south-1) took 0s
λ cd images/

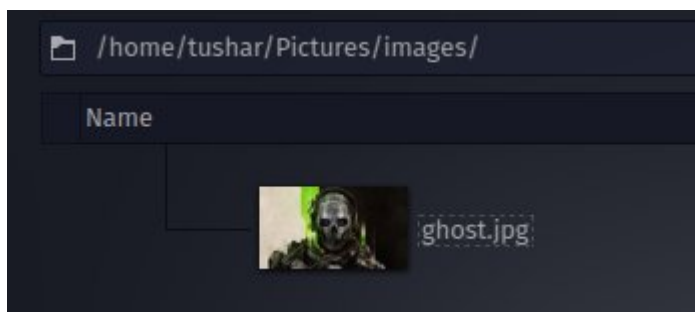
tushar@ROG in ~/Pictures/images on ○ (ap-south-1)
λ ls

tushar@ROG in ~/Pictures/images on ○ (ap-south-1) took 0s
λ aws s3 cp s3://tushar-s3-bucket-cli/ghost.jpg .
download: s3://tushar-s3-bucket-cli/ghost.jpg to ./ghost.jpg

tushar@ROG in ~/Pictures/images on ○ (ap-south-1) took 3s
λ ls
.rw-r--r-- 1.9M tushar 21 Nov 09:58 📄 ghost.jpg

```

Here you can see that the file is copy to the other location



Now lets list our objects located in bucket using below script

```

tushar@ROG in ~/Pictures/images on ○ (ap-south-1) took 0s
λ aws s3 ls s3://tushar-s3-bucket-cli
2024-11-21 09:58:38      1923113 ghost.jpg

```

To remove object run below command

```

tushar@ROG in ~/Pictures/images on ○ (ap-south-1) took 2s
λ aws s3 rm s3://tushar-s3-bucket-cli/ghost.jpg
delete: s3://tushar-s3-bucket-cli/ghost.jpg

tushar@ROG in ~/Pictures/images on ○ (ap-south-1) took 1s
λ aws s3 ls s3://tushar-s3-bucket-cli

```

As you can see above object is removed successfully.

To remove bucket run below command

```
tushar@ROG in ~/Pictures/images on ^ (ap-south-1) took 1s
λ aws s3 rb s3://tushar-s3-bucket-cli
remove_bucket: tushar-s3-bucket-cli

tushar@ROG in ~/Pictures/images on ^ (ap-south-1) took 2s
λ aws s3 ls
2024-11-14 05:43:46 user-code-storage
```

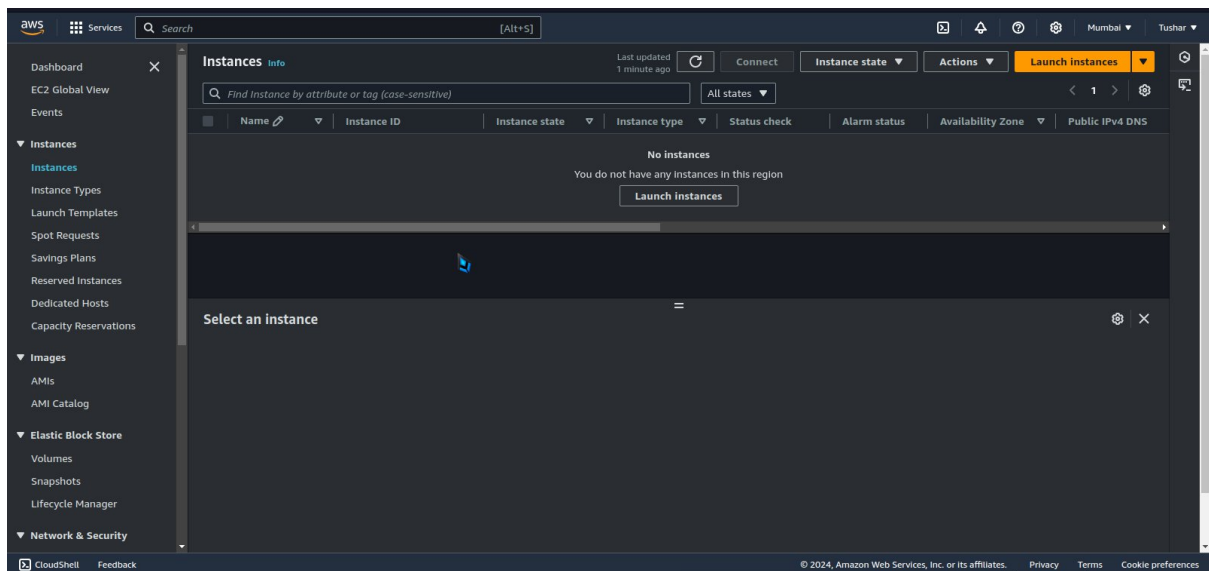
As you can see our bucket has been removed successfully.

- **EC2:**

To list instances located in our aws run below command

```
tushar@ROG in ~ on ^ (ap-south-1) took 0s
λ aws ec2 describe-instances
{
  "Reservations": []
}
```

I don't have any instances running so above showing empty



This command is use to Create the instance take the image id, key name, security group from Lanuch instance and subents from vpc

Here's the general syntax

```
aws ec2 run-instances \
  --image-id ami-xxxxxxxxxxxxxxxx \
  --count 1 \
  --instance-type t2.micro \
  --key-name your-key-name \
  --security-group-ids sg-xxxxxxxxxxxxxxxx \
  --subnet-id subnet-xxxxxxxxxxxxxxxx
```

```
tushar@ROG in ~ on ~ (ap-south-1) took 1s
[✗] * aws ec2 run-instances --image-id ami-0aebec83a182ea7ea --count 1 --instance-type t3.micro --key-name P13 --security-group-ids sg-0540331144081c6c4 --subnet-id subnet-0f5740ab48e51f44f
{
  "ReservationId": "r-0e45c863e3a791f14",
  "OwnerId": "009160043403",
  "Groups": [],
  "Instances": [
    {
      "Architecture": "x86_64",
      "BlockDeviceMappings": [],
      "ClientToken": "c95fa6fe-fc8b-4f37-ba17-9c0c2debafa5",
      "EbsOptimized": false,
      "EnaSupport": true,
      "Hypervisor": "xen",
      "NetworkInterfaces": [
        {
          "Attachment": {
            "AttachTime": "2024-11-21T05:15:50+00:00",
            "AttachmentId": "eni-attach-0bdf87ea6c3338e00",
            "DeleteOnTermination": true,
            "DeviceIndex": 0,
            "Status": "attaching",
            "NetworkCardIndex": 0
          },
          "Description": "",
          "Groups": [
            {
              "GroupId": "sg-0540331144081c6c4",
              "GroupName": "launch-wizard-1"
            }
          ],
          "Ipv6Addresses": [],
          "MacAddress": "06:51:b0:de:92:bd",
          "NetworkInterfaceId": "eni-0efda540905b752f6",
          "OwnerId": "009160043403",
          "PrivateDnsName": "ip-172-30-2-161.ap-south-1.compute.internal",
          "PrivateIpAddress": "172.30.2.161",
          "PrivateIpAddresses": [
            {
              "Primary": true,
              "PrivateDnsName": "ip-172-30-2-161.ap-south-1.compute.internal",

```

As you can see below running successfully

```
aws ec2 describe-instances --query
"Reservations[*].Instances[*].[InstanceId,InstanceType,State.Name,PublicIpAddress,PrivateIpAddress,Tags[?Key=='Name'].Value | [0]]" --output table
```

```
tushar@ROG in ~ on ~ (ap-south-1) took 0s
λ aws ec2 describe-instances --query "Reservations[*].Instances[*].[InstanceId,InstanceType,State.Name,PublicIpAddress,PrivateIpAddress,Tags[?Key=='Name'].Value | [0]]" --output table
```

DescribeInstances					
i-0e1cddbdf799176dd	t3.micro	running	3.110.99.205	172.30.2.38	Practical-13
i-094485b90d65d9854	t3.micro	running	3.110.63.203	172.30.2.161	None

Here you can see the instance is created

	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS
<input checked="" type="checkbox"/>	Practical-13	i-0e1cdbddf799176dd	Running	t3.micro	3/3 checks passed	View alarms	ap-south-1c	ec2-3-110-99-205.ap
<input type="checkbox"/>		i-094485b90d65d9854	Running	t3.micro	3/3 checks passed	View alarms	ap-south-1c	ec2-3-110-63-203.ap

To stop an instance run below command

```
tushar@ROG in ~ on ○ (ap-south-1) took 1s
λ aws ec2 stop-instances --instance-ids i-094485b90d65d9854 --output table
```

StopInstances	
StoppingInstances	
InstanceId	
i-094485b90d65d9854	
CurrentState	
Code	Name
64	stopping
PreviousState	
Code	Name
16	running

To start again that instance run below command

```
tushar@ROG in ~ on ○ (ap-south-1) took 0s
λ aws ec2 start-instances --instance-ids i-094485b90d65d9854 --output table
```

StartInstances	
StartingInstances	
+-----+	
InstanceId	
+-----+	
i-094485b90d65d9854	
+-----+	
CurrentState	
+-----+	
Code	Name
+-----+	
0	pending
+-----+	
PreviousState	
+-----+	
Code	Name
+-----+	
80	stopped
+-----+	

To terminate the instance run below command

```
tushar@ROG in ~ on ○ (ap-south-1) took 0s
λ aws ec2 terminate-instances --instance-ids i-094485b90d65d9854 --output table
```

TerminateInstances	
TerminatingInstances	
+-----+	
InstanceId	
+-----+	
i-094485b90d65d9854	
+-----+	
CurrentState	
+-----+	
Code	Name
+-----+	
32	shutting-down
+-----+	
PreviousState	
+-----+	
Code	Name
+-----+	
16	running
+-----+	

As you can see below instance has been terminated

```
tushar@ROG in ~ on ○ (ap-south-1) took 0s
λ aws ec2 describe-instances --query "Reservations[*].Instances[*].[InstanceId,InstanceType,State.Name,PublicIpAddress,PrivateIpAddress,Tags[?Key='Name'].Value | [0]]" --output table
```

DescribeInstances					
i-0e1cddbdf799176dd	t3.micro	running	3.110.99.205	172.30.2.38	Practical-13
i-094485b90d65d9854	t3.micro	terminated	None	None	None

- **IAM :**

First of all let's list our iam users by running below command

```
tushar@ROG in ~ on ○ (ap-south-1) took 0s
λ aws iam list-users --output table

|ListUsers|
+-----+
```

To create the IAM user run below command

```
tushar@ROG in ~ on ○ (ap-south-1) took 6s
λ aws iam create-user --user-name Tushar --output table

|CreateUser|
+-----+
|User|
+-----+
|Arn|arn:aws:iam::009160043403:user/Tushar|
|CreateDate|2024-11-21T06:30:12+00:00|
|Path|/|
|UserId|AIDAQEIP3FOFWG6GVGHXJ|
|UserName|Tushar|
+-----+
```

As you can see below IAM user Tushar is created successfully

```
tushar@ROG in ~ on ○ (ap-south-1) took 2s
λ aws iam list-users --output table

|ListUsers|
+-----+
|Users|
+-----+
|Arn|arn:aws:iam::009160043403:user/Tushar|
|CreateDate|2024-11-21T06:30:12+00:00|
|Path|/|
|UserId|AIDAQEIP3FOFWG6GVGHXJ|
|UserName|Tushar|
+-----+
```

To delete IAM user run below command

```
tushar@ROG in ~ on ○ (ap-south-1) took 0s
λ aws iam delete-user --user-name Tushar --output table

tushar@ROG in ~ on ○ (ap-south-1) took 3s
λ aws iam list-users --output table

|ListUsers|
+-----+
```

- **VPC :**

To see VPC located in aws account run below command

```
aws ec2 describe-vpcs \
  --query "Vpcs[*].[VpcId,CidrBlock,State,IsDefault,Tags[?Key=='Name'].Value | [0]]" \
  --output table
```

```
tushar@ROG in ~ on ○ (ap-south-1) took 2s
λ aws ec2 describe-vpcs \
  --query "Vpcs[*].[VpcId,CidrBlock,State,IsDefault,Tags[?Key=='Name'].Value | [0]]" \
  --output table
```

DescribeVpcs				
vpc-0bdfb69bd8cc491fd	172.30.0.0/16	available	False	None
vpc-05611eb2495f86cee	10.0.0.0/16	available	False	project-vpc