

School of Computer Science, Engineering and Applications (SCSEA)
B.C.A. TY (CCSA)
Subject: Containers & Orchestration

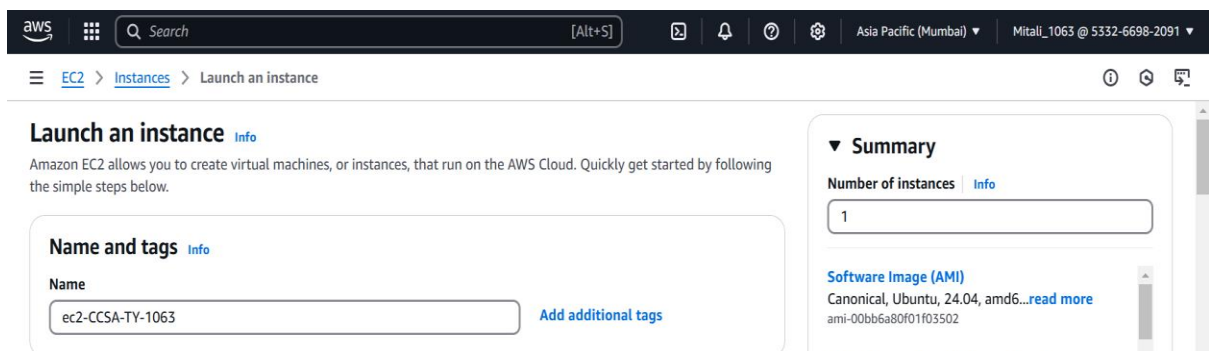
Name of the Student: Mitali Bhattad

PRN: 20220801063

Title of Practical: Configure Docker Community Latest version edition on cloud based instance verify the docker version

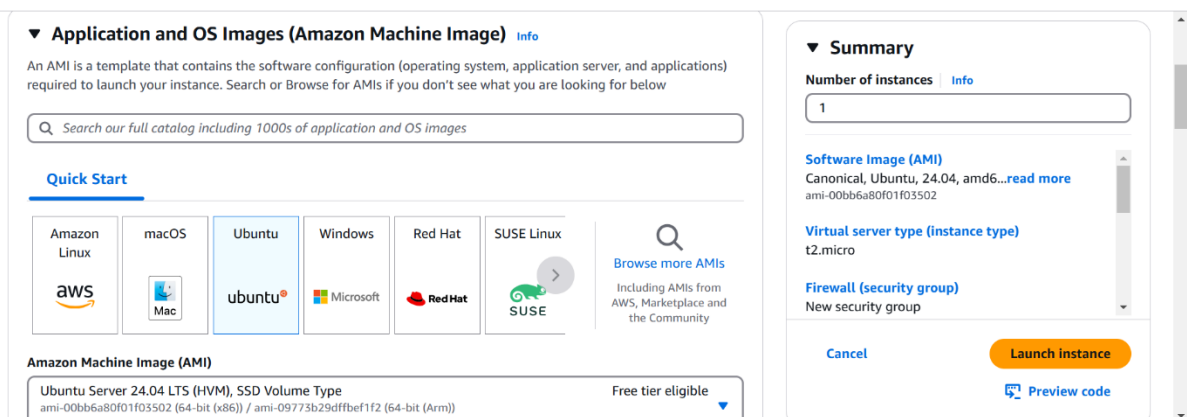
Step1: Launch an EC2 Instance

- Name the Instance



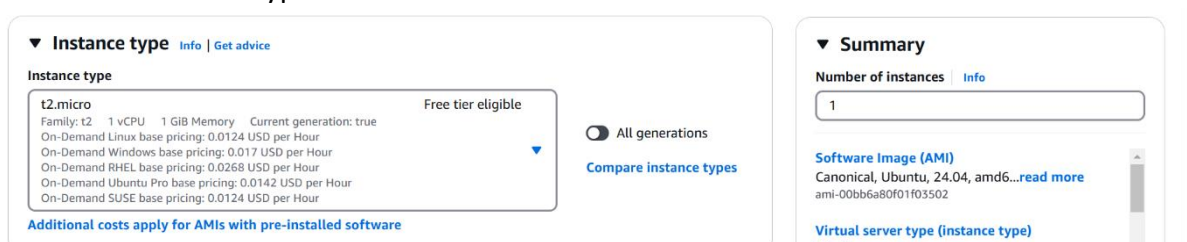
The screenshot shows the AWS Management Console 'Launch an instance' page. The 'Name and tags' section has a text input field containing 'ec2-CCSA-TY-1063'. The 'Summary' section on the right shows 'Number of instances' as 1 and 'Software Image (AMI)' as 'Canonical, Ubuntu, 24.04, amd64...read more' with the ID 'ami-00bb6a80f01f03502'.

- Choose AMI: Ubuntu



The screenshot shows the 'Choose AMI' page. Under 'Quick Start', the 'Ubuntu' tile is selected. The 'Amazon Machine Image (AMI)' section shows 'Ubuntu Server 24.04 LTS (HVM), SSD Volume Type' with ID 'ami-00bb6a80f01f03502'. The 'Summary' section on the right shows 'Number of instances' as 1, 'Software Image (AMI)' as 'Canonical, Ubuntu, 24.04, amd64...read more' with ID 'ami-00bb6a80f01f03502', 'Virtual server type (instance type)' as 't2.micro', and 'Firewall (security group)' as 'New security group'. At the bottom right, there are 'Cancel', 'Launch instance', and 'Preview code' buttons.

- Select the instance type: t2 micro



The screenshot shows the 'Select instance type' page. The 'Instance type' dropdown is set to 't2.micro', which is marked as 'Free tier eligible'. The 'Summary' section on the right shows 'Number of instances' as 1, 'Software Image (AMI)' as 'Canonical, Ubuntu, 24.04, amd64...read more' with ID 'ami-00bb6a80f01f03502', and 'Virtual server type (instance type)' as 't2.micro'.

School of Computer Science, Engineering and Applications (SCSEA)

B.C.A. TY (CCSA)

Subject: Containers & Orchestration

Name of the Student: Mitali Bhattad

PRN: 20220801063

Title of Practical: Configure Docker Community Latest version edition on cloud based instance verify the docker version

- Create a Key pair

Key pair (login) [Info](#)

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name - *required*

KP-1063 [Create new key pair](#)

Summary

Number of instances [Info](#)

1

Software Image (AMI)

Canonical, Ubuntu, 24.04, amd64...[read more](#)

- Security Group: Allow all the traffic

[Additional charges apply](#) when outside of [free tier allowance](#)

Firewall (security groups) [Info](#)

A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

☒ Create security group ☐ Select existing security group

We'll create a new security group called 'launch-wizard-1' with the following rules:

☒ Allow SSH traffic from

☒ Allow HTTPS traffic from the internet

To set up an endpoint, for example when creating a web server

☒ Allow HTTP traffic from the internet

To set up an endpoint, for example when creating a web server

Number of instances [Info](#)

1

Software Image (AMI)

Canonical, Ubuntu, 24.04, amd64...[read more](#)

ami-00bb6a80f01f03502

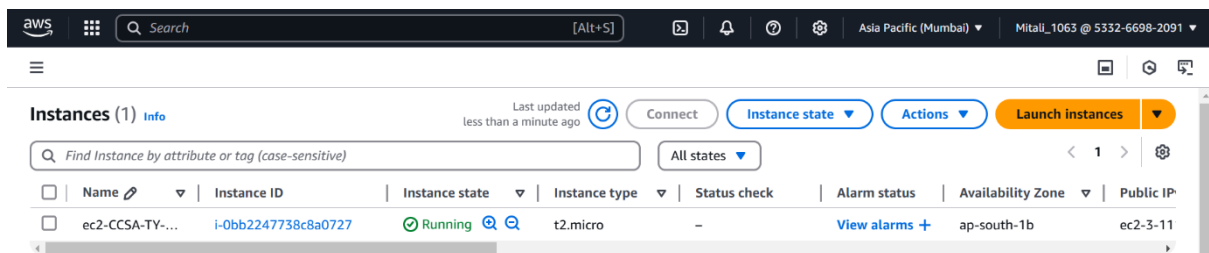
Virtual server type (instance type)

t2.micro

Firewall (security group)

New security group

- Launch the instance



The screenshot shows the AWS Management Console interface. At the top, there's a search bar and navigation icons. Below that, the 'Instances' section is active, showing a table with one instance: 'ec2-CCSA-TY-...' with ID 'i-0bb2247738c8a0727', state 'Running', type 't2.micro', and availability zone 'ap-south-1b'. The 'Launch instances' button is visible in the top right of the console area.

	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IP
<input type="checkbox"/>	ec2-CCSA-TY-...	i-0bb2247738c8a0727	Running	t2.micro	-	View alarms	ap-south-1b	ec2-3-11

Step 2: Connect the EC2 Instance and run the following commands:

1. Set up Docker's apt repository.

-sudo apt-get update -y

```
ubuntu@ip-172-31-3-152:~$ sudo apt-get update -y
```

i-0bb2247738c8a0727 (ec2-CCSA-TY-1063)

PublicIPs: 3.111.55.38 PrivateIPs: 172.31.3.152

-sudo apt-get install ca-certificates curl -y

```
Fetches 32.2 MB in 12s (2622 kB/s)
Reading package lists... Done
ubuntu@ip-172-31-3-152:~$ sudo apt-get install ca-certificates curl -y
```

i-0bb2247738c8a0727 (ec2-CCSA-TY-1063)

PublicIPs: 3.111.55.38 PrivateIPs: 172.31.3.152

School of Computer Science, Engineering and Applications (SCSEA)

B.C.A. TY (CCSA)

Subject: Containers & Orchestration

Name of the Student: Mitali Bhattad

PRN: 20220801063

Title of Practical: Configure Docker Community Latest version edition on cloud based instance verify the docker version

```
-sudo install -m 0755 -d /etc/apt/keyrings
```

```
-sudo curl -fsSL https://download.docker.com/linux/ubuntu/gpg -o  
/etc/apt/keyrings/docker.asc
```

```
-sudo chmod a+r /etc/apt/keyrings/docker.asc
```

```
-echo \  
"deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.asc]  
https://download.docker.com/linux/ubuntu \  
$(. /etc/os-release && echo "$VERSION_CODENAME") stable" | \  
sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
```

```
ubuntu@ip-172-31-3-152:~$ sudo install -m 0755 -d /etc/apt/keyrings  
ubuntu@ip-172-31-3-152:~$ sudo curl -fsSL https://download.docker.com/linux/ubuntu/gpg -o /etc/apt/keyrings/docker.asc  
ubuntu@ip-172-31-3-152:~$ sudo chmod a+r /etc/apt/keyrings/docker.asc  
ubuntu@ip-172-31-3-152:~$ echo \  
"deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.asc] https://download.docker.com/linux/ubuntu \  
$(. /etc/os-release && echo "${UBUNTU_CODENAME:-$VERSION_CODENAME}") stable" | \  
sudo tee /etc/apt/sources.list.d/docker.list > /dev/null  
ubuntu@ip-172-31-3-152:~$
```

i-0bb2247738c8a0727 (ec2-CCSA-TY-1063)

PublicIPs: 3.111.55.38 PrivateIPs: 172.31.3.152

```
-sudo apt-get update
```

```
ubuntu@ip-172-31-3-152:~$ sudo apt-get update  
Hit:1 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble InRelease  
Hit:2 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease  
Hit:3 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease  
Get:4 https://download.docker.com/linux/ubuntu noble InRelease [48.8 kB]  
Hit:5 http://security.ubuntu.com/ubuntu noble-security InRelease  
Get:6 https://download.docker.com/linux/ubuntu noble/stable amd64 Packages [18.9 kB]  
Fetched 67.7 kB in 1s (88.2 kB/s)  
Reading package lists... Done  
ubuntu@ip-172-31-3-152:~$
```

i-0bb2247738c8a0727 (ec2-CCSA-TY-1063)

PublicIPs: 3.111.55.38 PrivateIPs: 172.31.3.152



School of Computer Science, Engineering and Applications (SCSEA)

B.C.A. TY (CCSA)

Subject: Containers & Orchestration

Name of the Student: Mitali Bhattad

PRN: 20220801063

Title of Practical: Configure Docker Community Latest version edition on cloud based instance verify the docker version

2. Install the Docker packages.

-sudo apt-get install docker-ce docker-ce-cli containerd.io docker-buildx-plugin
docker-compose-plugin

```
ubuntu@ip-172-31-3-152:~$ sudo apt-get install docker-ce docker-ce-cli containerd.io docker-buildx-plugin docker-compose-plugin
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
```

3. To verify the docker installation:

Run the following commands:

- docker -v

```
ubuntu@ip-172-31-3-152:~$ docker -v
Docker version 27.5.1, build 9f9e405
ubuntu@ip-172-31-3-152:~$
```

i-0bb2247738c8a0727 (ec2-CCSA-TY-1063)

PublicIPs: 3.111.55.38 PrivateIPs: 172.31.3.152