

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

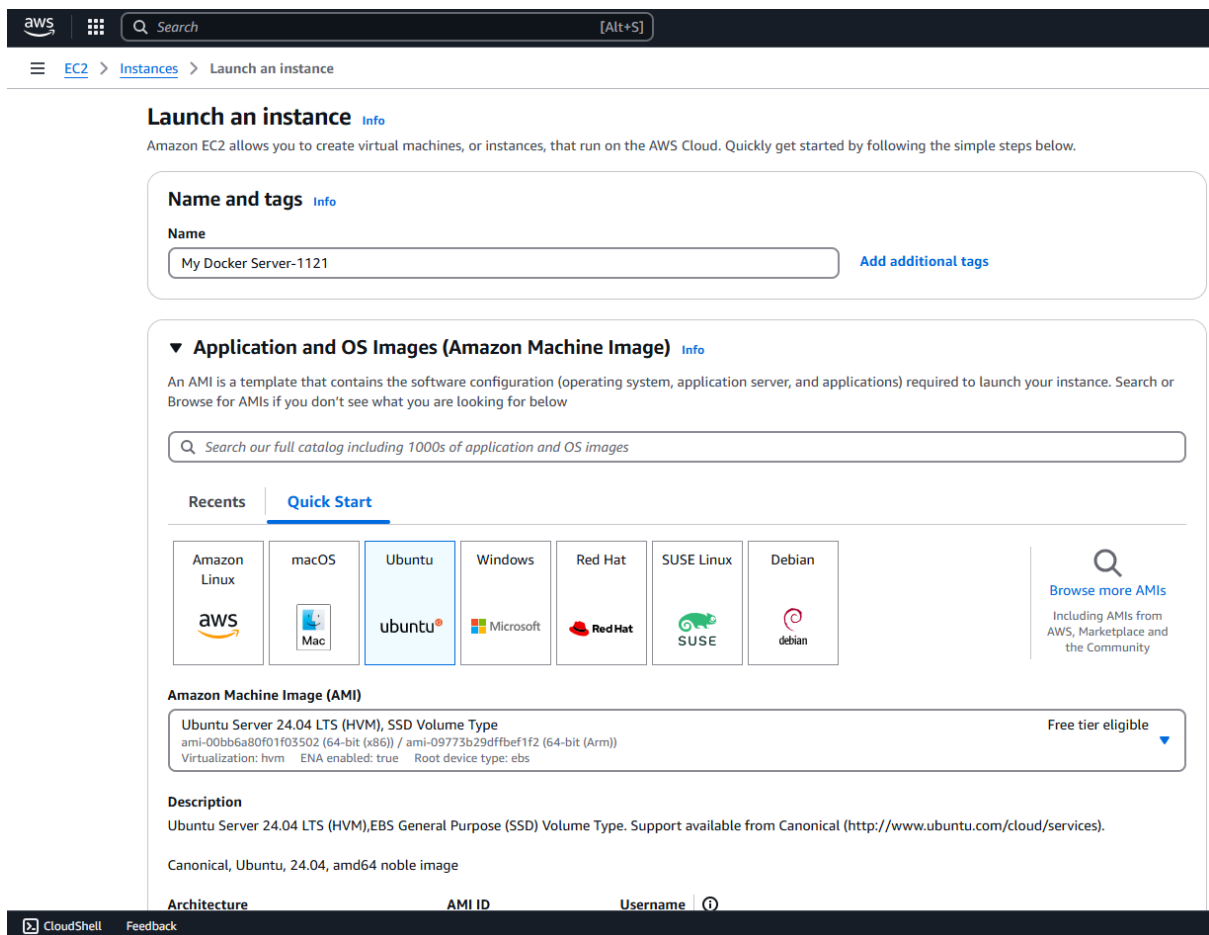
Name of the Student: **Prakhar Anil Sharma**

PRN: 20220801121

Title of Practical : **Deploying and Securing Docker Community Edition on
AWS Linux EC2 Instance**

Step 1] Create an EC2 instance :

- Name your instance- My Docker Server-1121 .
- Select AMI- Ubuntu .



Launch an instance [Info](#)

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

Name and tags [Info](#)

Name

My Docker Server-1121 [Add additional tags](#)

▼ Application and OS Images (Amazon Machine Image) [Info](#)

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below

Recents **Quick Start**

Amazon Linux macOS **Ubuntu** Windows Red Hat SUSE Linux Debian

aws Mac ubuntu® Microsoft Red Hat SUSE debian

Amazon Machine Image (AMI)

Ubuntu Server 24.04 LTS (HVM), SSD Volume Type
ami-00bb6a80f01f03502 (64-bit (x86)) / ami-09773b29dffbf1f2 (64-bit (Arm))
Virtualization: hvm ENA enabled: true Root device type: ebs [Free tier eligible](#)

Description

Ubuntu Server 24.04 LTS (HVM),EBS General Purpose (SSD) Volume Type. Support available from Canonical (<http://www.ubuntu.com/cloud/services>).

Canonical, Ubuntu, 24.04, amd64 noble image

Architecture **AMI ID** **Username** ⓘ

CloudShell Feedback

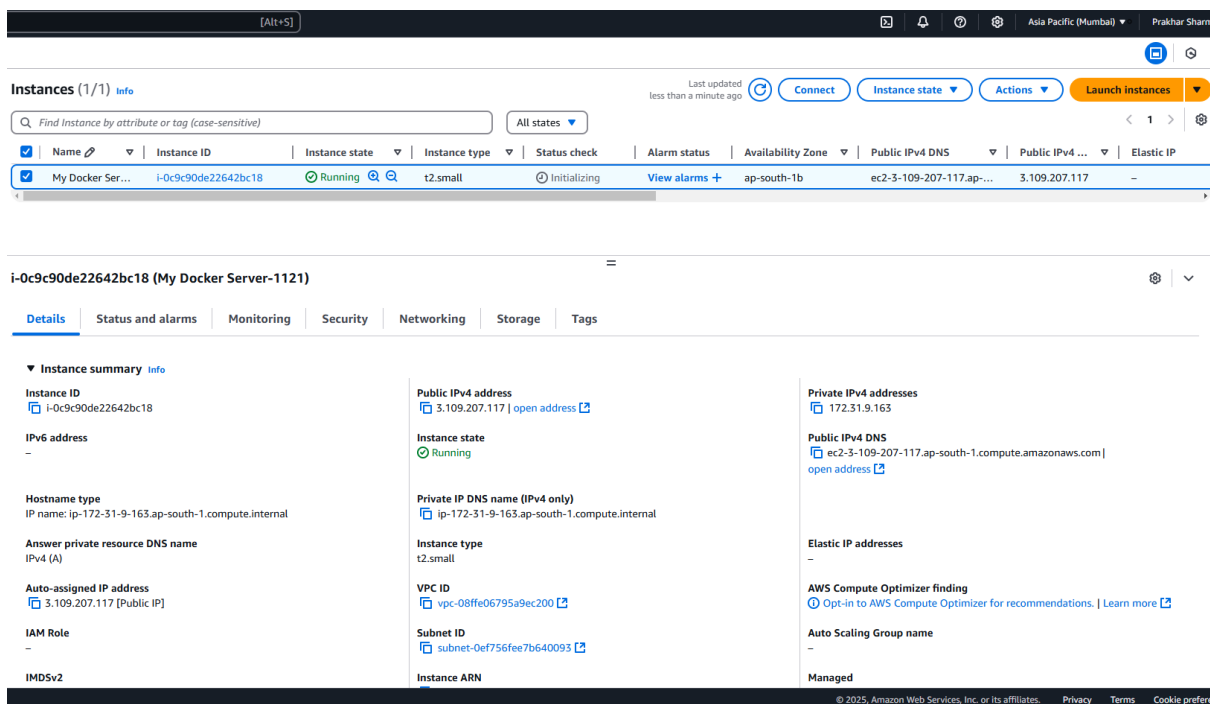
- Select instance type– t2.small .
- Create a key pair- My Key 007 .

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

Name of the Student: **Prakhar Anil Sharma**

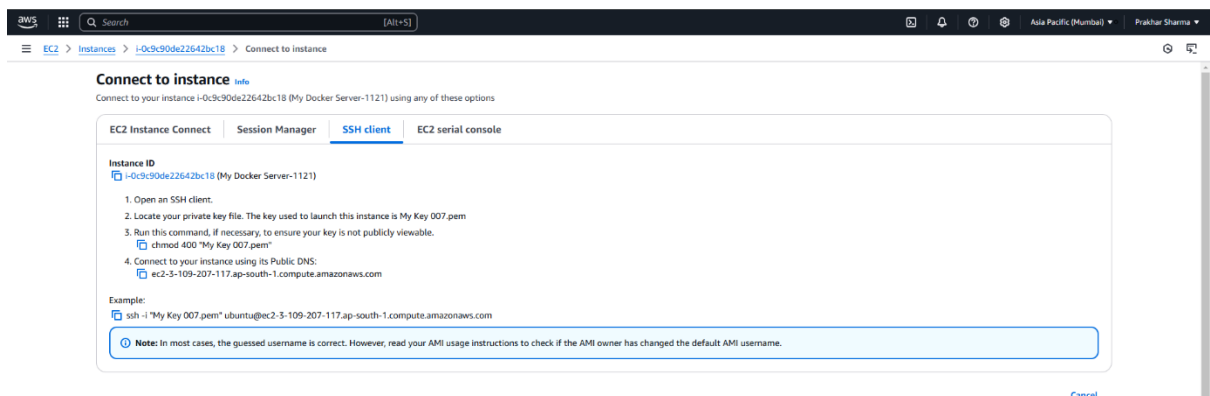
PRN: 20220801121

Title of Practical : **Deploying and Securing Docker Community Edition on
AWS Linux EC2 Instance**



The screenshot shows the AWS Management Console interface. At the top, there's a navigation bar with the AWS logo, search bar, and user profile. Below it, the 'Instances' page is displayed, showing a table with one instance: 'My Docker Ser...' with ID 'i-0c9c90de22642bc18', state 'Running', type 't2.small', and public IP '3.109.207.117'. The instance details page for 'i-0c9c90de22642bc18 (My Docker Server-1121)' is shown below the table. It includes tabs for Details, Status and alarms, Monitoring, Security, Networking, Storage, and Tags. The 'Details' tab is active, showing instance summary, public and private IP addresses, DNS names, VPC ID, Subnet ID, and Instance ARN.

- Connect the instance using SSH .
- Copy the example .



The screenshot shows the 'Connect to instance' page in the AWS Management Console. It provides instructions on how to connect to the EC2 instance 'i-0c9c90de22642bc18 (My Docker Server-1121)' using the SSH client. The instructions include: 1. Open an SSH client. 2. Locate your private key file. 3. Run the command 'chmod 400 "My Key 007.pem"'. 4. Connect to your instance using its Public DNS: 'ec2-3-109-207-117.ap-south-1.compute.amazonaws.com'. An example command is provided: 'ssh -i "My Key 007.pem" ubuntu@ec2-3-109-207-117.ap-south-1.compute.amazonaws.com'. A note states: 'Note: In most cases, the guessed username is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI username.'

Step 3] Installing Docker :

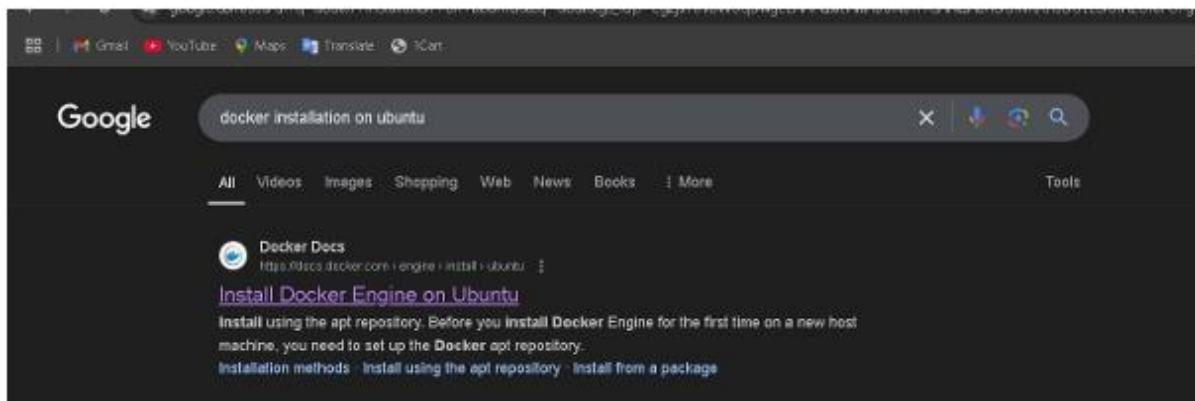
School of Computer Science, Engineering and Applications(SCSEA)
B.C.A. TY (CCSA)
Subject : Containers and Orchestration (P)

Name of the Student: **Prakhar Anil Sharma**

PRN: **20220801121**

Title of Practical : **Deploying and Securing Docker Community Edition on AWS Linux EC2 Instance**

- Search docker installation on ubuntu on google .
- Click on the first link .



Step 4] Open the Command Prompt :

- Go to the destination folder where the key is downloaded, using cd command following the destination folder name

Then,

Enter the following commands:

- i) sudo-i .
- ii)sudo apt-get update-y .

School of Computer Science, Engineering and Applications(SCSEA)

B.C.A. TY (CCSA)

Subject : Containers and Orchestration (P)

Name of the Student: **Prakhar Anil Sharma**

PRN: 20220801121

Title of Practical : **Deploying and Securing Docker Community Edition on AWS Linux EC2 Instance**

```
root@ip-172-31-9-163: ~  
The list of available updates is more than a week old.  
To check for new updates run: sudo apt update  
  
The programs included with the Ubuntu system are free software;  
the exact distribution terms for each program are described in the  
individual files in /usr/share/doc/*/copyright.  
  
Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by  
applicable law.  
  
To run a command as administrator (user "root"), use "sudo <command>".  
See "man sudo_root" for details.  
  
ubuntu@ip-172-31-9-163:~$  
ubuntu@ip-172-31-9-163:~$ sudo -i  
root@ip-172-31-9-163:~# sudo apt-get update -y  
Hit:1 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble InRelease  
Get:2 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease [126 kB]  
Get:3 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease [126 kB]  
Get:4 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 Packages [15.0 MB]  
Get:5 http://security.ubuntu.com/ubuntu noble-security InRelease [126 kB]  
Get:6 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble/universe Translation-en [5982 kB]  
Get:7 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 Components [3871 kB]  
Get:8 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 c-n-f Metadata [301 kB]  
Get:9 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse amd64 Packages [269 kB]
```

iii)sudo apt-get install ca-certificates curl-y .

iv)sudo install-m 0755-d /etc/apt/keyrings .

v)sudo curl-fsSL <https://download.docker.com/linux/ubuntu/gpg-o/etc/apt/keyrings/docker.asc> .

vi)sudo chmod a+r /etc/apt/keyrings/docker.asc .

vii)echo \ "deb [arch=\$(dpkg--print-architecture) signed
by=/etc/apt/keyrings/docker.asc]

<https://download.docker.com/linux/ubuntu> \ \$(. /etc/os-release &&

School of Computer Science, Engineering and Applications(SCSEA)

B.C.A. TY (CCSA)

Subject : Containers and Orchestration (P)

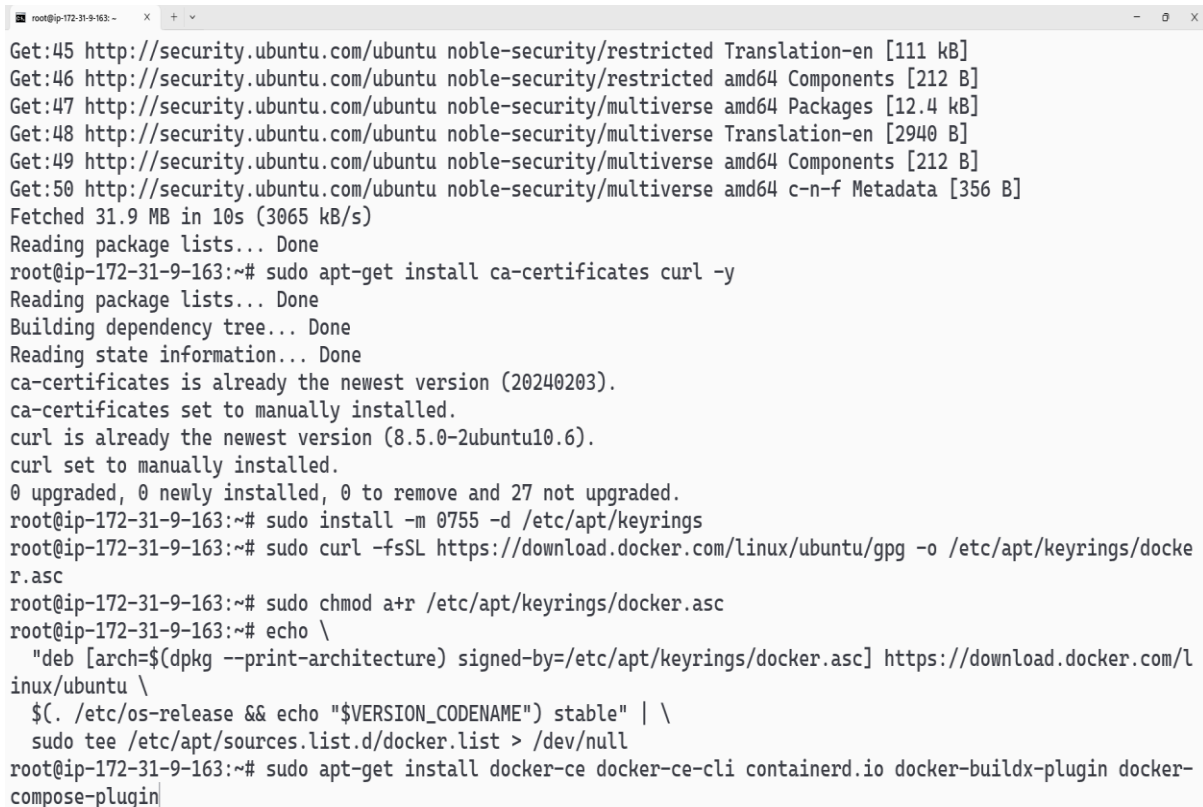
Name of the Student: **Prakhar Anil Sharma**

PRN: 20220801121

Title of Practical : **Deploying and Securing Docker Community Edition on
AWS Linux EC2 Instance**

```
echo "$VERSION_CODENAME") stable" | \ sudo tee  
/etc/apt/sources.list.d/docker.list > /dev/null .
```

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



```
Get:45 http://security.ubuntu.com/ubuntu noble-security/restricted Translation-en [111 kB]  
Get:46 http://security.ubuntu.com/ubuntu noble-security/restricted amd64 Components [212 B]  
Get:47 http://security.ubuntu.com/ubuntu noble-security/multiverse amd64 Packages [12.4 kB]  
Get:48 http://security.ubuntu.com/ubuntu noble-security/multiverse Translation-en [2940 B]  
Get:49 http://security.ubuntu.com/ubuntu noble-security/multiverse amd64 Components [212 B]  
Get:50 http://security.ubuntu.com/ubuntu noble-security/multiverse amd64 c-n-f Metadata [356 B]  
Fetched 31.9 MB in 10s (3065 kB/s)  
Reading package lists... Done  
root@ip-172-31-9-163:~# sudo apt-get install ca-certificates curl -y  
Reading package lists... Done  
Building dependency tree... Done  
Reading state information... Done  
ca-certificates is already the newest version (20240203).  
ca-certificates set to manually installed.  
curl is already the newest version (8.5.0-2ubuntu10.6).  
curl set to manually installed.  
0 upgraded, 0 newly installed, 0 to remove and 27 not upgraded.  
root@ip-172-31-9-163:~# sudo install -m 0755 -d /etc/apt/keyrings  
root@ip-172-31-9-163:~# sudo curl -fsSL https://download.docker.com/linux/ubuntu/gpg -o /etc/apt/keyrings/docker.asc  
root@ip-172-31-9-163:~# sudo chmod a+r /etc/apt/keyrings/docker.asc  
root@ip-172-31-9-163:~# 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  
root@ip-172-31-9-163:~# sudo apt-get install docker-ce docker-ce-cli containerd.io docker-buildx-plugin docker-compose-plugin
```

Step 4] Verify the docker version :

To verify the docker version enter the following command:

- **docker-v or docker —version**

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

Name of the Student: **Prakhar Anil Sharma**

PRN: 20220801121

Title of Practical : **Deploying and Securing Docker Community Edition on AWS Linux EC2 Instance**

```
root@ip-172-31-9-163: ~
Setting up docker-ce-cli (5:27.5.1-1~ubuntu.24.04~noble) ...
Setting up libslirp0:amd64 (4.7.0-1ubuntu3) ...
Setting up pigz (2.8-1) ...
Setting up docker-ce-rootless-extras (5:27.5.1-1~ubuntu.24.04~noble) ...
Setting up slirp4netns (1.2.1-1build2) ...
Setting up docker-ce (5:27.5.1-1~ubuntu.24.04~noble) ...
Created symlink /etc/systemd/system/multi-user.target.wants/docker.service → /usr/lib/systemd/system/docker.service.
Created symlink /etc/systemd/system/sockets.target.wants/docker.socket → /usr/lib/systemd/system/docker.socket.
Processing triggers for man-db (2.12.0-4build2) ...
Processing triggers for libc-bin (2.39-0ubuntu8.3) ...
Scanning processes...
Scanning linux images...

Running kernel seems to be up-to-date.

No services need to be restarted.

No containers need to be restarted.

No user sessions are running outdated binaries.

No VM guests are running outdated hypervisor (qemu) binaries on this host.
root@ip-172-31-9-163:~# docker -v
Docker version 27.5.1, build 9f9e405
root@ip-172-31-9-163:~# docker --version
Docker version 27.5.1, build 9f9e405
root@ip-172-31-9-163:~#
```