

Assignment–6

Module-9: Containerization, Docker, and Docker Hub

Submitted by : Shaik Junaid Adil

Date of Submission: 26-06-2024

Submitted to: Vikul

L1 - Create Docker file and build the docker container Application Image for the application build in Jenkins Module

Step 1: Create an Instance

The screenshot shows the AWS EC2 console interface for launching a new instance. The 'Name' field is set to 'Docker-Test'. In the 'Application and OS Images (Amazon Machine Image)' section, the 'Ubuntu' icon is selected. The 'Virtual server type (instance type)' is set to 't2.micro'. The 'Launch instance' button is highlighted in orange at the bottom right.

The screenshot shows the AWS EC2 console interface for launching a new instance. The 'Amazon Machine Image (AMI)' dropdown is set to 'Ubuntu Server 22.04 LTS (HVM), SSD Volume Type'. The 'Instance type' dropdown is set to 't2.micro'. The 'Launch instance' button is highlighted in orange at the bottom right.

ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#LaunchInstances:

Services Search [Alt+S] Mumbai Junaid Adil

EC2 Instances Launch an instance

Launching instance
Creating security group rules 21%

▶ Details

Please wait while we launch your instance.
Do not close your browser while this is loading.

Activate Windows
Go to Settings to activate Windows.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#Instances:instanceId=i-06c497142a314286c

Services Search [Alt+S] Mumbai Junaid Adil

EC2 Dashboard EC2 Global View Events

Instances Instances Instance Types Launch Templates Spot Requests Savings Plans Reserved Instances Dedicated Hosts Capacity Reservations

Images AMIs AMI Catalog

Elastic Block Store Volumes

Instances (1/1) Info Connect Instance state Actions Launch instances

Find Instance by attribute or tag (case-sensitive) All states

Instance ID i-06c497142a314286c X Clear filters

Docker-Test i-06c497142a314286c Running t2.micro Initializing View alarms ap-south-1

i-06c497142a314286c (Docker-Test)

Details Status and alarms Monitoring Security Networking Storage Tags

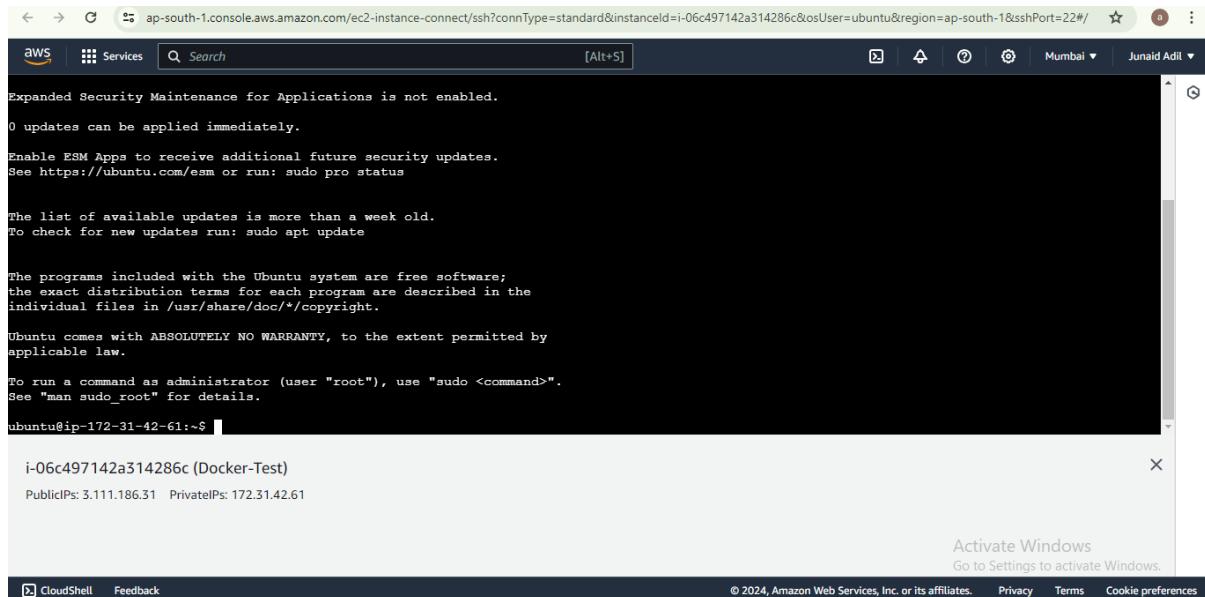
Instance summary Info

Instance ID i-06c497142a314286c (Docker-Test)	Public IPv4 address 3.111.186.31 open address	Private IPv4 addresses 172.31.42.61
IPv6 address -	Instance state Running	Public IPv4 DNS ec2-3-111-186-31.ap-south-1

Activate Windows
Go to Settings to activate Windows.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Step 2: Connect to the Instance

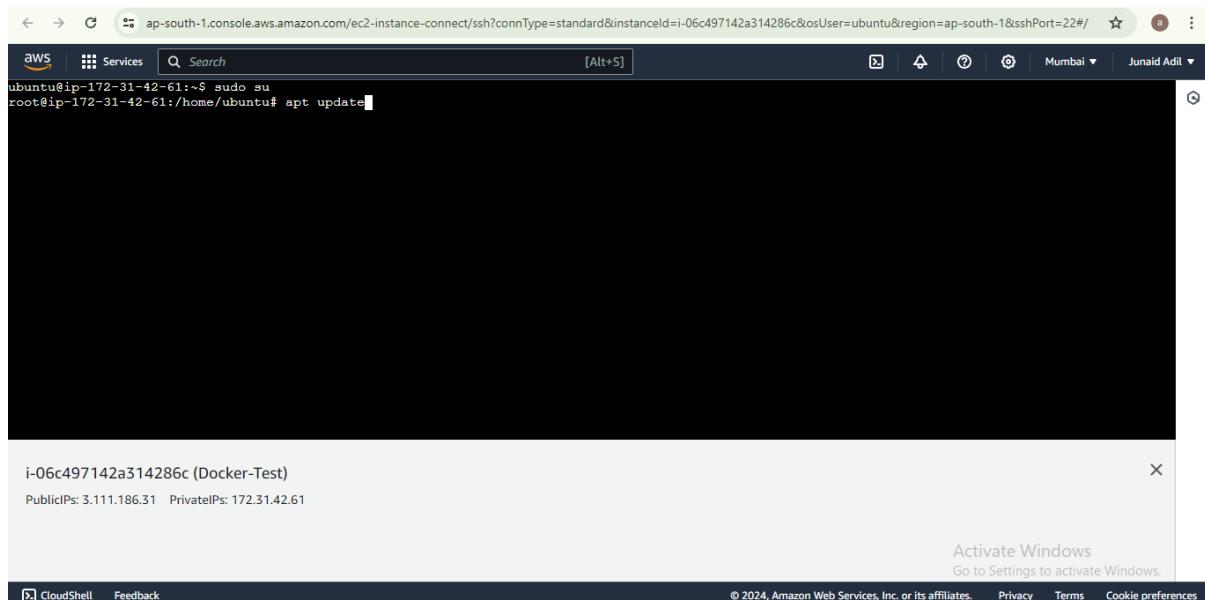


The screenshot shows a terminal window within the AWS CloudShell interface. The URL in the address bar is `ap-south-1.console.aws.amazon.com/ec2-instance-connect/ssh?connType=standard&instanceId=i-06c497142a314286c&osUser=ubuntu®ion=ap-south-1&sshPort=22#/`. The terminal output shows several system status messages:

```
Expanded Security Maintenance for Applications is not enabled.  
0 updates can be applied immediately.  
Enable ESM Apps to receive additional future security updates.  
See https://ubuntu.com/esm or run: sudo pro status  
  
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-172-31-42-61:~$
```

Below the terminal, the status bar indicates the instance ID as `i-06c497142a314286c (Docker-Test)`, and its PublicIPs and PrivateIPs as `3.111.186.31` and `172.31.42.61`. A watermark for "Activate Windows" is visible in the bottom right corner.

Step 3: Become root user and run the command to update packages “apt update”



The screenshot shows a terminal window within the AWS CloudShell interface. The URL in the address bar is `ap-south-1.console.aws.amazon.com/ec2-instance-connect/ssh?connType=standard&instanceId=i-06c497142a314286c&osUser=ubuntu®ion=ap-south-1&sshPort=22#/`. The terminal output shows the user running the `sudo su` command to become root, followed by the `apt update` command:

```
ubuntu@ip-172-31-42-61:~$ sudo su  
root@ip-172-31-42-61:/home/ubuntu# apt update
```

Below the terminal, the status bar indicates the instance ID as `i-06c497142a314286c (Docker-Test)`, and its PublicIPs and PrivateIPs as `3.111.186.31` and `172.31.42.61`. A watermark for "Activate Windows" is visible in the bottom right corner.

```
Get:21 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/main amd64 Packages [67.1 kB]
Get:22 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/main Translation-en [11.0 kB]
Get:23 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/main amd64 c-n-f Metadata [388 B]
Get:24 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/restricted amd64 c-n-f Metadata [116 B]
Get:25 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/universe amd64 Packages [27.2 kB]
Get:26 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/universe Translation-en [16.3 kB]
Get:27 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/universe amd64 c-n-f Metadata [644 B]
Get:28 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/multiverse amd64 c-n-f Metadata [116 B]
Get:29 http://security.ubuntu.com/ubuntu jammy-security/main amd64 Packages [1526 kB]
Get:30 http://security.ubuntu.com/ubuntu jammy-security/main Translation-en [261 kB]
Get:31 http://security.ubuntu.com/ubuntu jammy-security/restricted amd64 Packages [1937 kB]
Get:32 http://security.ubuntu.com/ubuntu jammy-security/restricted Translation-en [330 kB]
Get:33 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 Packages [860 kB]
Get:34 http://security.ubuntu.com/ubuntu jammy-security/universe Translation-en [167 kB]
Get:35 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 c-n-f Metadata [16.8 kB]
Get:36 http://security.ubuntu.com/ubuntu jammy-security/multiverse amd64 Packages [37.2 kB]
Get:37 http://security.ubuntu.com/ubuntu jammy-security/multiverse Translation-en [7588 B]
Get:38 http://security.ubuntu.com/ubuntu jammy-security/multiverse amd64 c-n-f Metadata [260 B]
Fetched 31.8 MB in 6s (5395 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
49 packages can be upgraded. Run 'apt list --upgradable' to see them.
root@ip-172-31-42-61:/home/ubuntu#
```

i-06c497142a314286c (Docker-Test)

PublicIPs: 3.111.186.31 PrivateIPs: 172.31.42.61

Activate Windows
Go to Settings to activate Windows.

CloudShell Feedback

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Step 4: Install Docker using command “`apt install docker.io -y`”

```
Get:21 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/main amd64 Packages [67.1 kB]
Get:22 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/main Translation-en [11.0 kB]
Get:23 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/main amd64 c-n-f Metadata [388 B]
Get:24 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/restricted amd64 c-n-f Metadata [116 B]
Get:25 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/universe amd64 Packages [27.2 kB]
Get:26 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/universe Translation-en [16.3 kB]
Get:27 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/universe amd64 c-n-f Metadata [644 B]
Get:28 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/multiverse amd64 c-n-f Metadata [116 B]
Get:29 http://security.ubuntu.com/ubuntu jammy-security/main amd64 Packages [1526 kB]
Get:30 http://security.ubuntu.com/ubuntu jammy-security/main Translation-en [261 kB]
Get:31 http://security.ubuntu.com/ubuntu jammy-security/restricted amd64 Packages [1937 kB]
Get:32 http://security.ubuntu.com/ubuntu jammy-security/restricted Translation-en [330 kB]
Get:33 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 Packages [860 kB]
Get:34 http://security.ubuntu.com/ubuntu jammy-security/universe Translation-en [167 kB]
Get:35 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 c-n-f Metadata [16.8 kB]
Get:36 http://security.ubuntu.com/ubuntu jammy-security/multiverse amd64 Packages [37.2 kB]
Get:37 http://security.ubuntu.com/ubuntu jammy-security/multiverse Translation-en [7588 B]
Get:38 http://security.ubuntu.com/ubuntu jammy-security/multiverse amd64 c-n-f Metadata [260 B]
Fetched 31.8 MB in 6s (5395 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
49 packages can be upgraded. Run 'apt list --upgradable' to see them.
root@ip-172-31-42-61:/home/ubuntu# apt install docker.io -y
```

i-06c497142a314286c (Docker-Test)

PublicIPs: 3.111.186.31 PrivateIPs: 172.31.42.61

Activate Windows
Go to Settings to activate Windows.

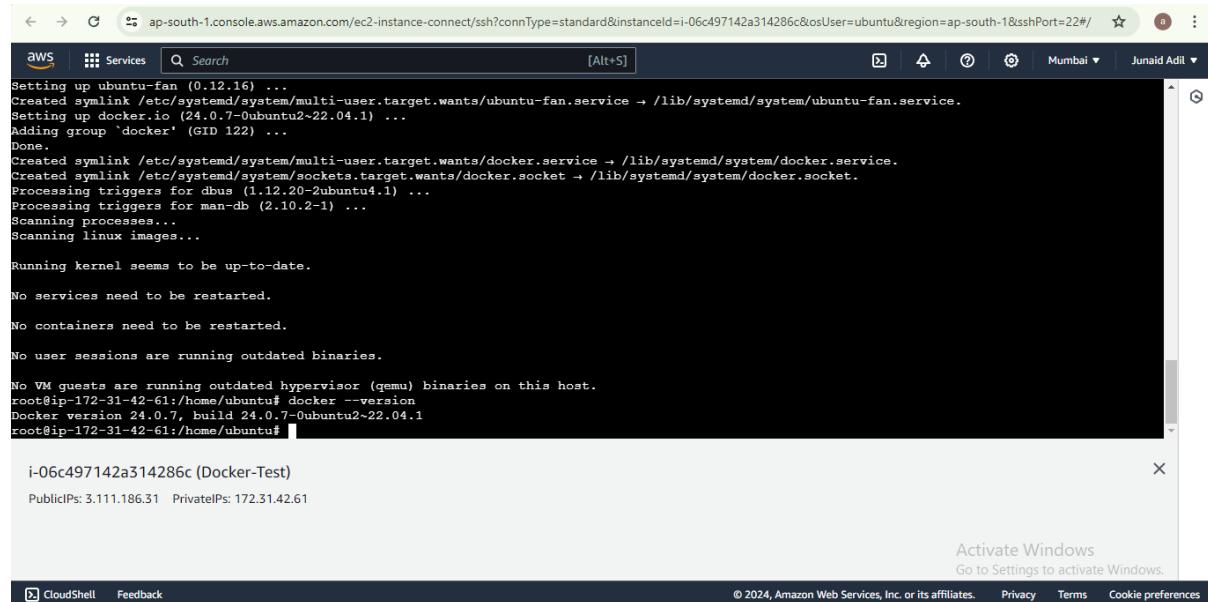
CloudShell Feedback

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

```
← → ⌂ ap-south-1.console.aws.amazon.com/ec2-instance-connect/ssh?connType=standard&instanceId=i-06c497142a314286c&osUser=ubuntu&region=ap-south-1&sshPort=22#/  
aws Services Search [Alt+S] Mumbai Junaid Adil  
Unpacking runc (1.1.12-0ubuntu2~22.04.1) ...  
Selecting previously unselected package containerd.  
Preparing to unpack .../3-containerd_1.7.12-0ubuntu2~22.04.1_amd64.deb ...  
Unpacking containerd (1.7.12-0ubuntu2~22.04.1) ...  
Selecting previously unselected package dns-root-data.  
Preparing to unpack .../4-dns-root-data_2023112702-ubuntu0.22.04.1_all.deb ...  
Unpacking dns-root-data (2023112702-ubuntu0.22.04.1) ...  
Selecting previously unselected package dnsmasq-base.  
Preparing to unpack .../5-dnsmasq-base_2.90-0ubuntu0.22.04.1_amd64.deb ...  
Unpacking dnsmasq-base (2.90-0ubuntu0.22.04.1) ...  
Selecting previously unselected package docker.io.  
Preparing to unpack .../6-docker.io_24.0.7-0ubuntu2~22.04.1_amd64.deb ...  
Unpacking docker.io (24.0.7-0ubuntu2~22.04.1) ...  
Selecting previously unselected package ubuntu-fan.  
Preparing to unpack .../7-ubuntu-fan_0.12.16_all.deb ...  
Unpacking ubuntu-fan (0.12.16) ...  
Setting up runc (1.1.12-0ubuntu2~22.04.1) ...  
Setting up dns-root-data (2023112702-ubuntu0.22.04.1) ...  
Setting up bridge-utils (1.7-1ubuntu3) ...  
Setting up pigz (2.6-1) ...  
Setting up containerd (1.7.12-0ubuntu2~22.04.1) ...  
[=====]  
i-06c497142a314286c (Docker-Test)  
PublicIPs: 3.111.186.31 PrivateIPs: 172.31.42.61  
Activate Windows  
Go to Settings to activate Windows.  
CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences
```

```
← → ⌂ ap-south-1.console.aws.amazon.com/ec2-instance-connect/ssh?connType=standard&instanceId=i-06c497142a314286c&osUser=ubuntu&region=ap-south-1&sshPort=22#/  
aws Services Search [Alt+S] Mumbai Junaid Adil  
Setting up containerd (1.7.12-0ubuntu2~22.04.1) ...  
Created symlink /etc/systemd/system/multi-user.target.wants/containerd.service → /lib/systemd/system/containerd.service.  
Setting up ubuntu-fan (0.12.16) ...  
Created symlink /etc/systemd/system/multi-user.target.wants/ubuntu-fan.service → /lib/systemd/system/ubuntu-fan.service.  
Setting up docker.io (24.0.7-0ubuntu2~22.04.1) ...  
Adding group 'docker' (GID 122) ...  
Done.  
Created symlink /etc/systemd/system/multi-user.target.wants/docker.service → /lib/systemd/system/docker.service.  
Created symlink /etc/systemd/system/sockets.target.wants/docker.socket → /lib/systemd/system/docker.socket.  
Processing triggers for dbus (1.12.20-2ubuntu4.1) ...  
Processing triggers for man-db (2.10.2-1) ...  
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-172-31-42-61:/home/ubuntu#  
i-06c497142a314286c (Docker-Test)  
PublicIPs: 3.111.186.31 PrivateIPs: 172.31.42.61  
Activate Windows  
Go to Settings to activate Windows.  
CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences
```

Step 5: we can check the docker version using command “**docker --version**”



```
Setting up ubuntu-fan (0.12.16) ...
Created symlink /etc/systemd/system/multi-user.target.wants/ubuntu-fan.service → /lib/systemd/system/ubuntu-fan.service.
Setting up docker.io (24.0.7-0ubuntu2~22.04.1) ...
Adding group 'docker' (GID 122) ...
Done.
Created symlink /etc/systemd/system/multi-user.target.wants/docker.service → /lib/systemd/system/docker.service.
Created symlink /etc/systemd/system/sockets.target.wants/docker.socket → /lib/systemd/system/docker.socket.
Processing triggers for dbus (1.12.20-2ubuntu4.1) ...
Processing triggers for man-db (2.10.2-1) ...
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.

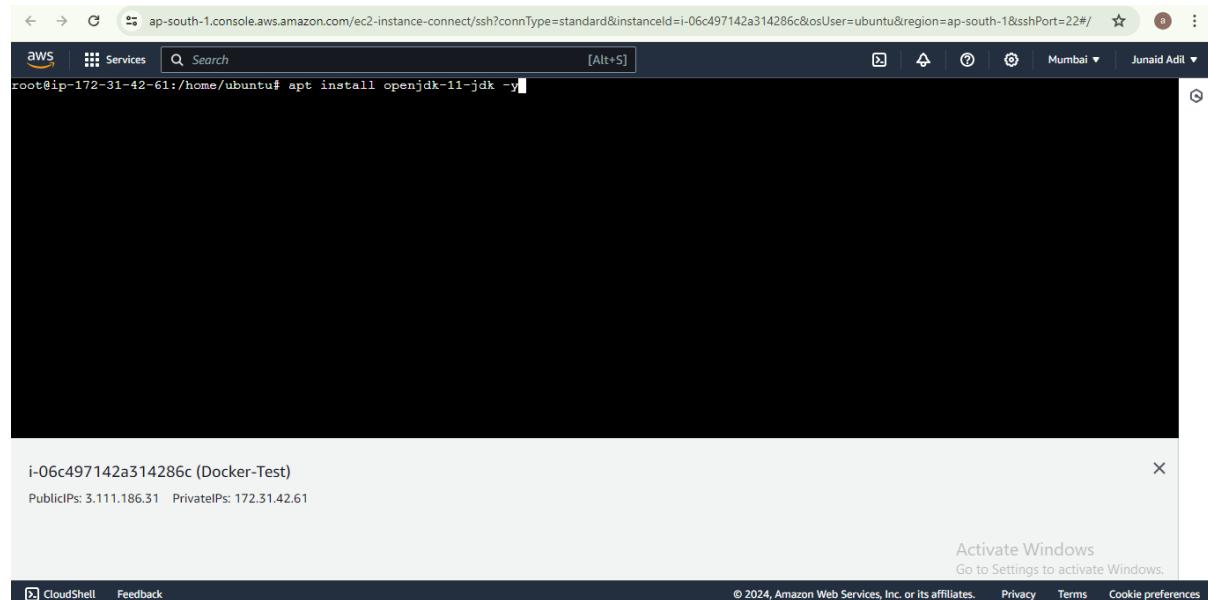
No user sessions are running outdated binaries.

No VM guests are running outdated hypervisor (qemu) binaries on this host.
root@ip-172-31-42-61:/home/ubuntu# docker --version
Docker version 24.0.7, build 24.0.7-0ubuntu2~22.04.1
root@ip-172-31-42-61:/home/ubuntu#
```

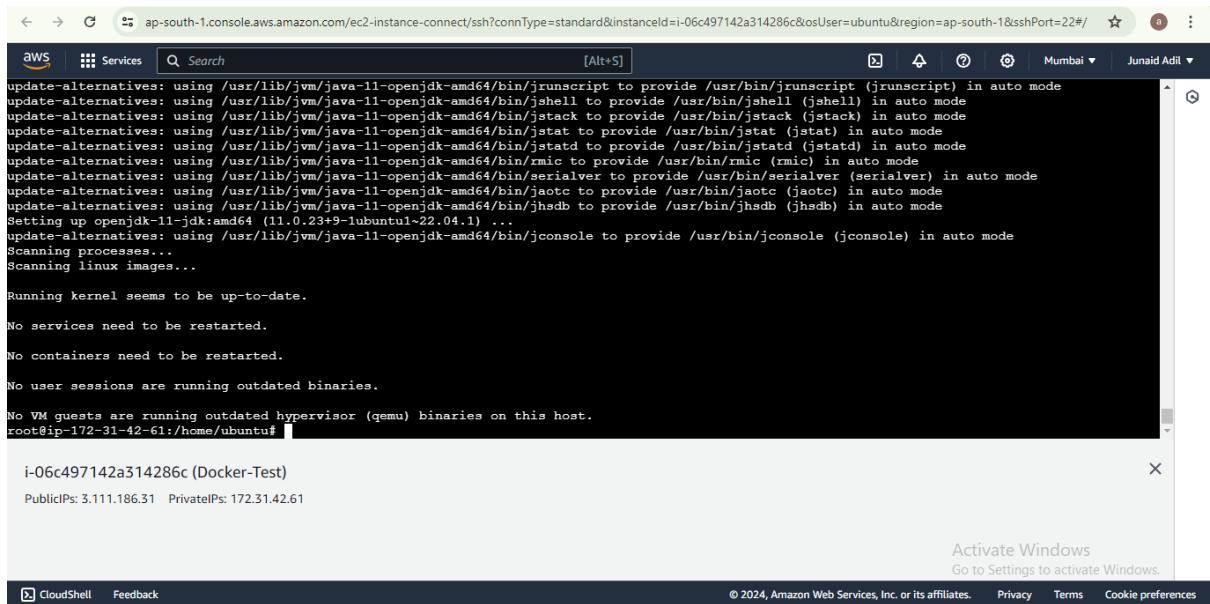
Step 6: Install Java, Maven using command

“Java : **apt install openjdk-11-jdk -y**”

“Maven : **apt install maven -y**”



```
root@ip-172-31-42-61:/home/ubuntu# apt install openjdk-11-jdk -y
```



```
update-alternatives: using /usr/lib/jvm/java-11-openjdk-amd64/bin/jrunscript to provide /usr/bin/jrunscript (jrunscript) in auto mode
update-alternatives: using /usr/lib/jvm/java-11-openjdk-amd64/bin/jshell to provide /usr/bin/jshell (jshell) in auto mode
update-alternatives: using /usr/lib/jvm/java-11-openjdk-amd64/bin/jstack to provide /usr/bin/jstack (jstack) in auto mode
update-alternatives: using /usr/lib/jvm/java-11-openjdk-amd64/bin/jstat to provide /usr/bin/jstat (jstat) in auto mode
update-alternatives: using /usr/lib/jvm/java-11-openjdk-amd64/bin/jstalg to provide /usr/bin/jstalg (jstalg) in auto mode
update-alternatives: using /usr/lib/jvm/java-11-openjdk-amd64/bin/rmic to provide /usr/bin/rmic (rmic) in auto mode
update-alternatives: using /usr/lib/jvm/java-11-openjdk-amd64/bin/serialver to provide /usr/bin/serialver (serialver) in auto mode
update-alternatives: using /usr/lib/jvm/java-11-openjdk-amd64/bin/jaotc to provide /usr/bin/jaotc (jaotc) in auto mode
update-alternatives: using /usr/lib/jvm/java-11-openjdk-amd64/bin/jhsdb to provide /usr/bin/jhsdb (jhsdb) in auto mode
Setting up openjdk-11-jdk:amd64 (11.0.23+9-lubuntu1~22.04.1) ...
update-alternatives: using /usr/lib/jvm/java-11-openjdk-amd64/bin/jconsole to provide /usr/bin/jconsole (jconsole) in auto mode
Scanning processes...
Scanning linux images...

Running kernel seems to be up-to-date.

No services need to be restarted.

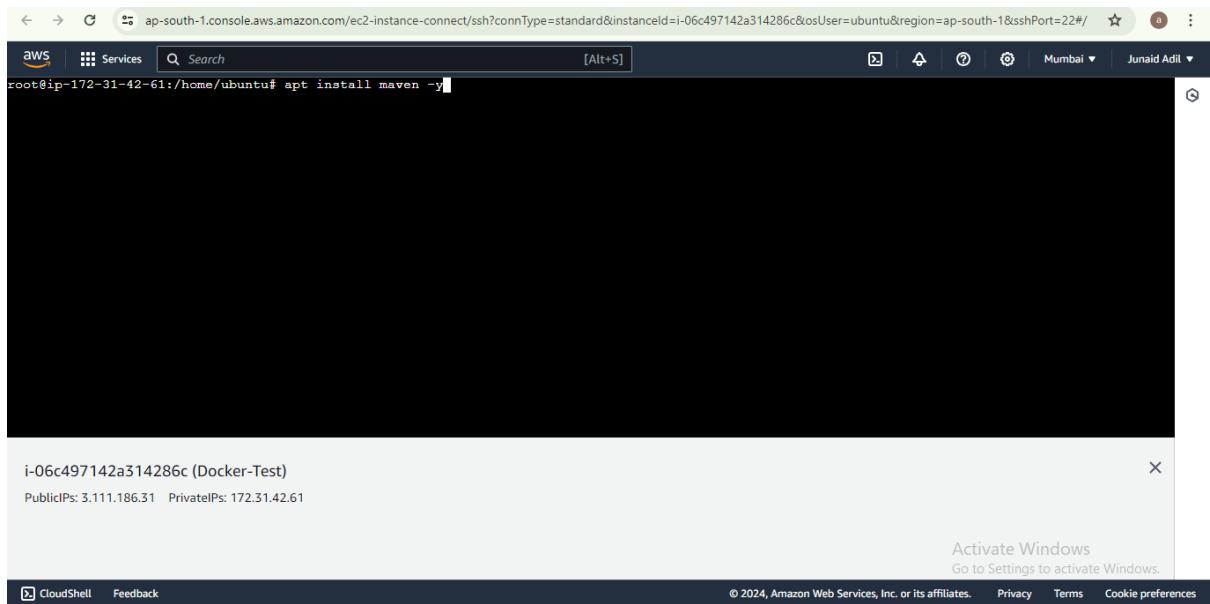
No containers need to be restarted.

No user sessions are running outdated binaries.

No VM guests are running outdated hypervisor (qemu) binaries on this host.
root@ip-172-31-42-61:/home/ubuntu#
```

i-06c497142a314286c (Docker-Test)
Public IPs: 3.111.186.31 Private IPs: 172.31.42.61

Maven Installation:



```
root@ip-172-31-42-61:/home/ubuntu# apt install maven -y
```

i-06c497142a314286c (Docker-Test)
Public IPs: 3.111.186.31 Private IPs: 172.31.42.61

```
← → G ap-south-1.console.aws.amazon.com/ec2-instance-connect/ssh?connType=standard&instanceId=i-06c497142a314286c&osUser=ubuntu&region=ap-south-1&sshPort=22#/ Junaid Adil ▾

aws Services Search [Alt+S] Mumbai ▾ Junaid Adil ▾

Preparing to unpack .../10-libguava-java_29.0-6_all.deb ...
Unpacking libguava-java (29.0-6) ...
Selecting previously unselected package libaopalliance-java.
Preparing to unpack .../11-libaopalliance-java_20070526-6_all.deb ...
Unpacking libaopalliance-java (20070526-6) ...
Selecting previously unselected package libguice-java.
Preparing to unpack .../12-libguice-java_4.2.3-2_all.deb ...
Unpacking libguice-java (4.2.3-2) ...
Selecting previously unselected package libhawtjni-runtime-java.
Preparing to unpack .../13-libhawtjni-runtime-java_1.17-1_all.deb ...
Unpacking libhawtjni-runtime-java (1.17-1) ...
Selecting previously unselected package libjansi-native-java.
Preparing to unpack .../14-libjansi-native-java_1.8-1_all.deb ...
Unpacking libjansi-native-java (1.8-1) ...
Selecting previously unselected package libjansi-java.
Preparing to unpack .../15-libjansi-java_1.18-1_all.deb ...
Unpacking libjansi-java (1.18-1) ...
Selecting previously unselected package libmaven-parent-java.
Preparing to unpack .../16-libmaven-parent-java_31-2_all.deb ...
Unpacking libmaven-parent-java (31-2) ...
Selecting previously unselected package libplexus-utils2-java.
Preparing to unpack .../17-libplexus-utils2-java_3.3.0-1_all.deb ...
[Progress: [ 26% [#####
i-06c497142a314286c (Docker-Test)
PublicIPs: 3.111.186.31 PrivateIPs: 172.31.42.61

Activate Windows
Go to Settings to activate Windows.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences
```

```
← → G ap-south-1.console.aws.amazon.com/ec2-instance-connect/ssh?connType=standard&instanceId=i-06c497142a314286c&osUser=ubuntu&region=ap-south-1&sshPort=22#/ Junaid Adil ▾

aws Services Search [Alt+S] Mumbai ▾ Junaid Adil ▾

Setting up libguava-java (29.0-6) ...
Setting up libcommons-lang3-java (3.11-1) ...
Setting up libjansi-native-java (1.8-1) ...
Setting up libwagon-file-java (3.3.4-1) ...
Setting up libcommons-io-java (2.11.0-2) ...
Setting up libguice-java (4.2.3-2) ...
Setting up libjansi-java (1.18-1) ...
Setting up libmaven-shared-utils-java (3.3.0-1ubuntu0.22.04.1) ...
Setting up libmaven3-core-java (3.6.3-5) ...
Setting up maven (3.6.3-5) ...
update-alternatives: using /usr/share/maven/bin/mvn to provide /usr/bin/mvn (mvn) in auto mode
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-172-31-42-61:/home/ubuntu# []

i-06c497142a314286c (Docker-Test)
PublicIPs: 3.111.186.31 PrivateIPs: 172.31.42.61

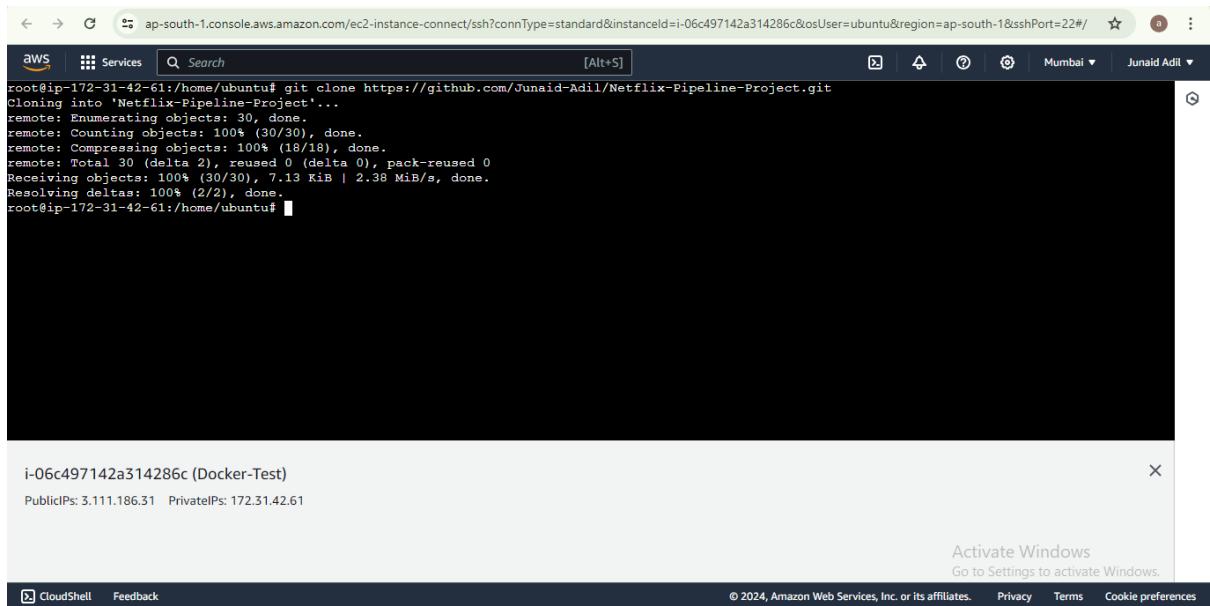
Activate Windows
Go to Settings to activate Windows.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences
```

Step 7: Clone the Repository using command “`git clone <HTTPS code>`”. Where the Artifact is built and saved in a Repository “Netflix-Pipeline-Project”.

The screenshot shows the GitHub repository page for 'Junaid-Adil / Netflix-Pipeline-Project'. The repository is public and contains one branch ('main') and one file ('pom.xml'). The 'Code' tab is selected. On the right, there's a 'Clone' section with options for Local, Codespaces, HTTPS (selected), SSH, and GitHub CLI. The HTTPS URL is highlighted with a red box: `https://github.com/Junaid-Adil/Netflix-Pipeline`. Other sections visible include 'About' (no description, website, or topics provided), 'Activity' (0 stars, 1 watching, 0 forks), 'Releases' (no releases published), and 'Packages' (no packages published). A 'README' section is present with a 'Add a README' button.

The screenshot shows an AWS CloudShell terminal window. The user is running the command `git clone https://github.com/Junaid-Adil/Netflix-Pipeline-Project.git`. The terminal output shows the repository has been cloned to the current directory. Below the terminal, a status bar displays the instance ID (i-06c497142a314286c) and IP addresses (PublicIPs: 3.111.186.31, PrivateIPs: 172.31.42.61). At the bottom, there are links for CloudShell, Feedback, and a footer with copyright information for Amazon Web Services.



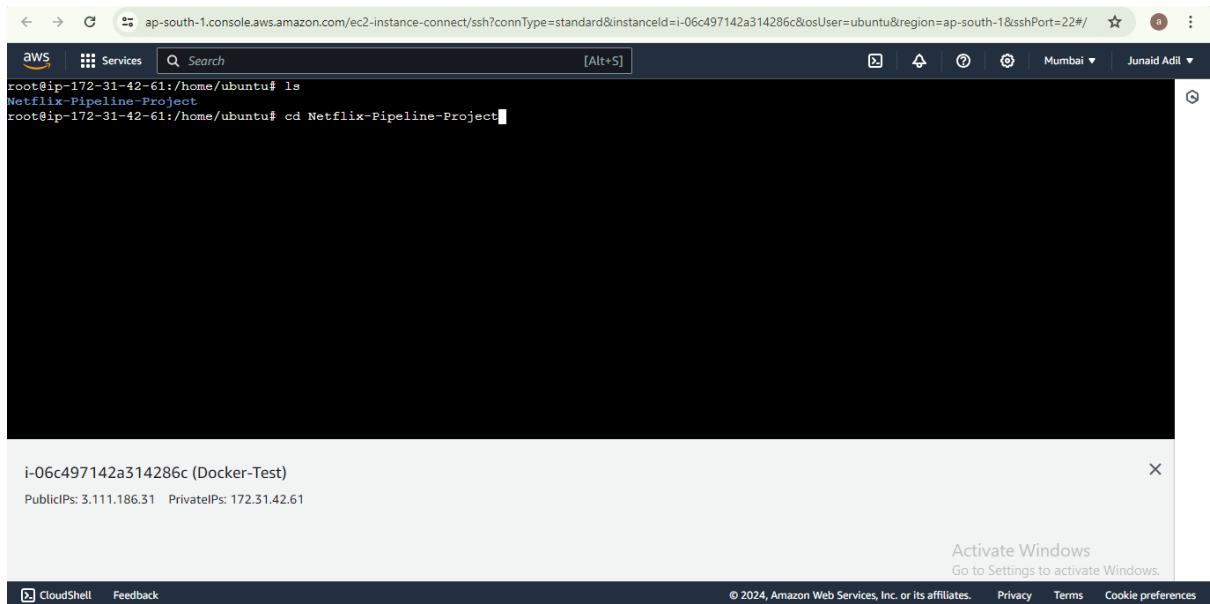
```
aws Services Search [Alt+S] Mumbai Junaid Adil
root@ip-172-31-42-61:/home/ubuntu# git clone https://github.com/Junaid-Adil/Netflix-Pipeline-Project.git
Cloning into 'Netflix-Pipeline-Project'...
remote: Enumerating objects: 30, done.
remote: Counting objects: 100% (30/30), done.
remote: Compressing objects: 100% (18/18), done.
remote: Total 30 (delta 2), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (30/30), 7.13 KiB | 2.38 MiB/s, done.
Resolving deltas: 100% (2/2), done.
root@ip-172-31-42-61:/home/ubuntu#
```

i-06c497142a314286c (Docker-Test)
PublicIPs: 3.111.186.31 PrivateIPs: 172.31.42.61

Activate Windows
Go to Settings to activate Windows.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Step 8: Go to project directory



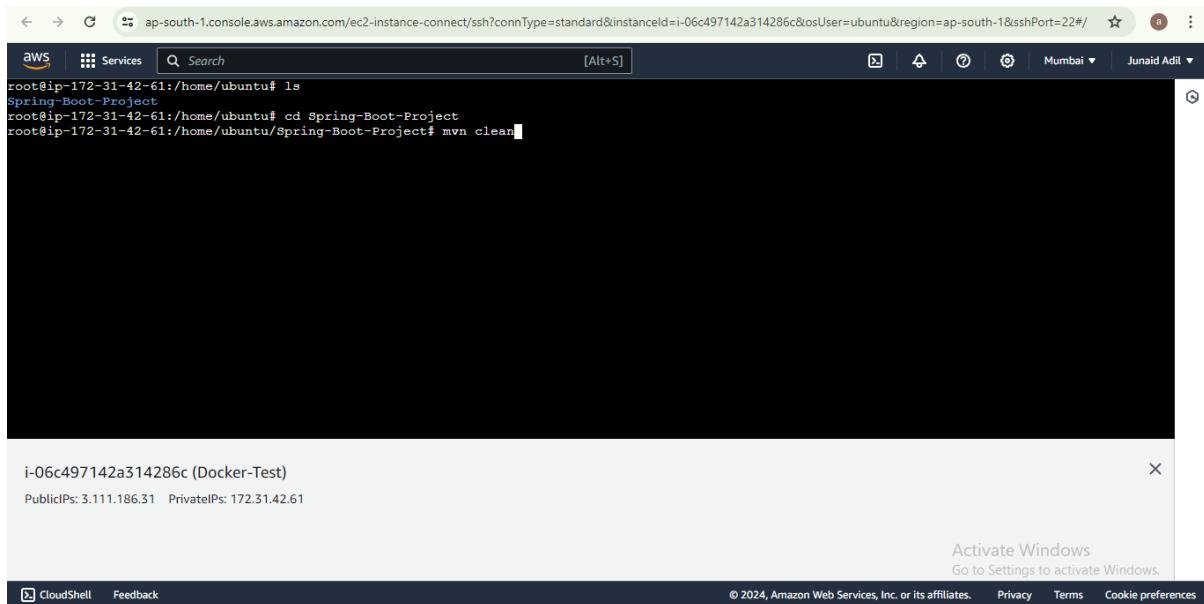
```
aws Services Search [Alt+S] Mumbai Junaid Adil
root@ip-172-31-42-61:/home/ubuntu# ls
Netflix-Pipeline-Project
root@ip-172-31-42-61:/home/ubuntu# cd Netflix-Pipeline-Project
```

i-06c497142a314286c (Docker-Test)
PublicIPs: 3.111.186.31 PrivateIPs: 172.31.42.61

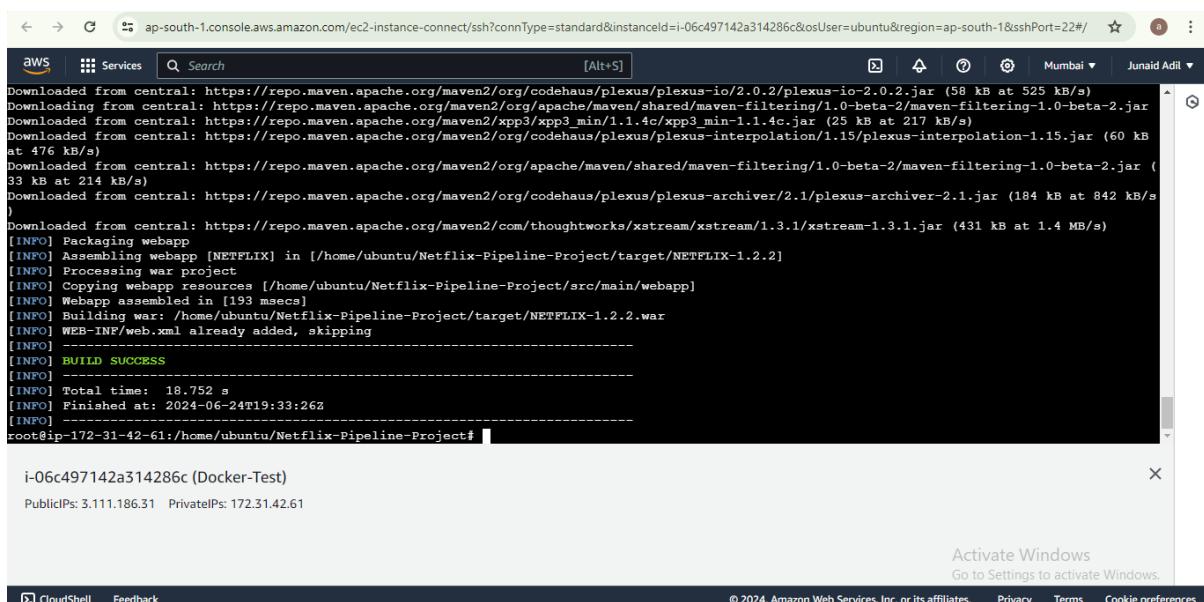
Activate Windows
Go to Settings to activate Windows.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Step 9: Run command “mvn clean” and then execute “mvn package”

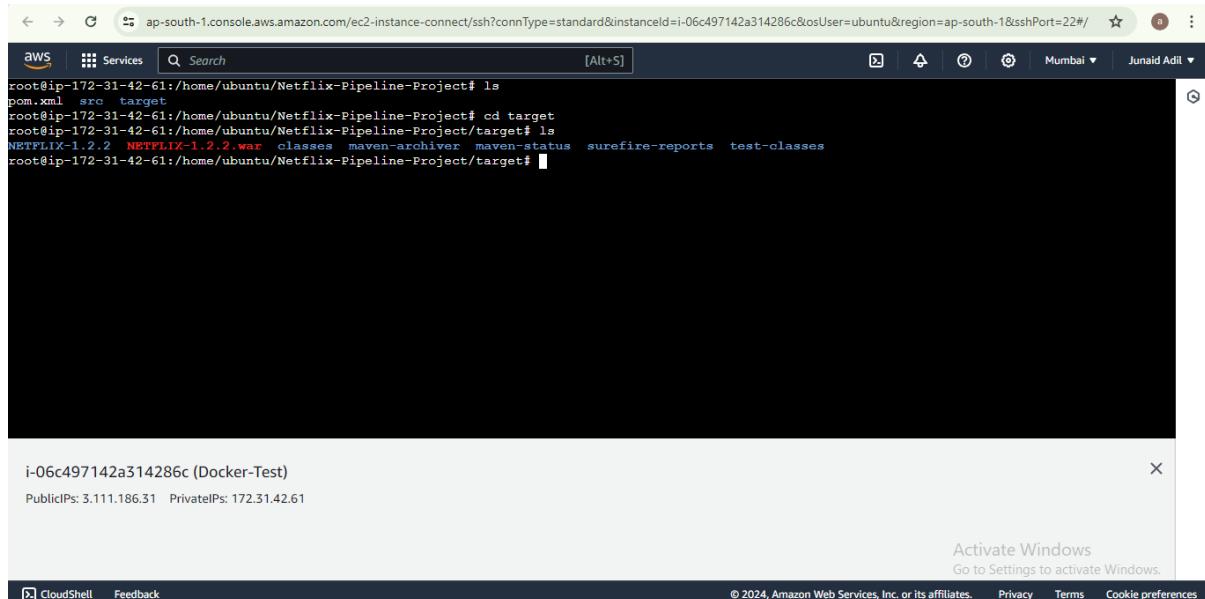


```
aws Services Search [Alt+S] Junaid Adil Mumbai
root@ip-172-31-42-61:/home/ubuntu# ls
Spring-Boot-Project
root@ip-172-31-42-61:/home/ubuntu# cd Spring-Boot-Project
root@ip-172-31-42-61:/home/ubuntu/Spring-Boot-Project# mvn clean
i-06c497142a314286c (Docker-Test)
PublicIPs: 3.111.186.31 PrivateIPs: 172.31.42.61
Activate Windows
Go to Settings to activate Windows.
CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences
```



```
aws Services Search [Alt+S] Junaid Adil Mumbai
Downloaded from central: https://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-io/2.0.2/plexus-io-2.0.2.jar (58 kB at 525 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/shared/maven-filtering/1.0-beta-2/maven-filtering-1.0-beta-2.jar
Downloaded from central: https://repo.maven.apache.org/maven2/xpp3/xpp3_min/1.1.4c/xpp3_min-1.1.4c.jar (25 kB at 217 kB/s)
Downloaded from central: https://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-interpolation/1.15/plexus-interpolation-1.15.jar (60 kB at 476 kB/s)
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/shared/maven-filtering/1.0-beta-2/maven-filtering-1.0-beta-2.jar (33 kB at 214 kB/s)
Downloaded from central: https://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-archiver/2.1/plexus-archiver-2.1.jar (184 kB at 842 kB/s)
Downloaded from central: https://repo.maven.apache.org/maven2/com/thoughtworks/xstream/xstream/1.3.1/xstream-1.3.1.jar (431 kB at 1.4 MB/s)
[INFO] Packaging webapp
[INFO] Assembling webapp [NETFLIX] in [/home/ubuntu/Netflix-Pipeline-Project/target/NETFLIX-1.2.2]
[INFO] Processing war project
[INFO] Copying webapp resources [/home/ubuntu/Netflix-Pipeline-Project/src/main/webapp]
[INFO] Webapp assembled in [193 mssecs]
[INFO] Building war: /home/ubuntu/Netflix-Pipeline-Project/target/NETFLIX-1.2.2.war
[INFO] WEB-INF/web.xml already added, skipping
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 18.752 s
[INFO] Finished at: 2024-06-24T19:33:26Z
[INFO] -----
root@ip-172-31-42-61:/home/ubuntu/Netflix-Pipeline-Project# 
i-06c497142a314286c (Docker-Test)
PublicIPs: 3.111.186.31 PrivateIPs: 172.31.42.61
Activate Windows
Go to Settings to activate Windows.
CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences
```

Step 10: we can check the .war file in Target folder. Execute command: “ls”



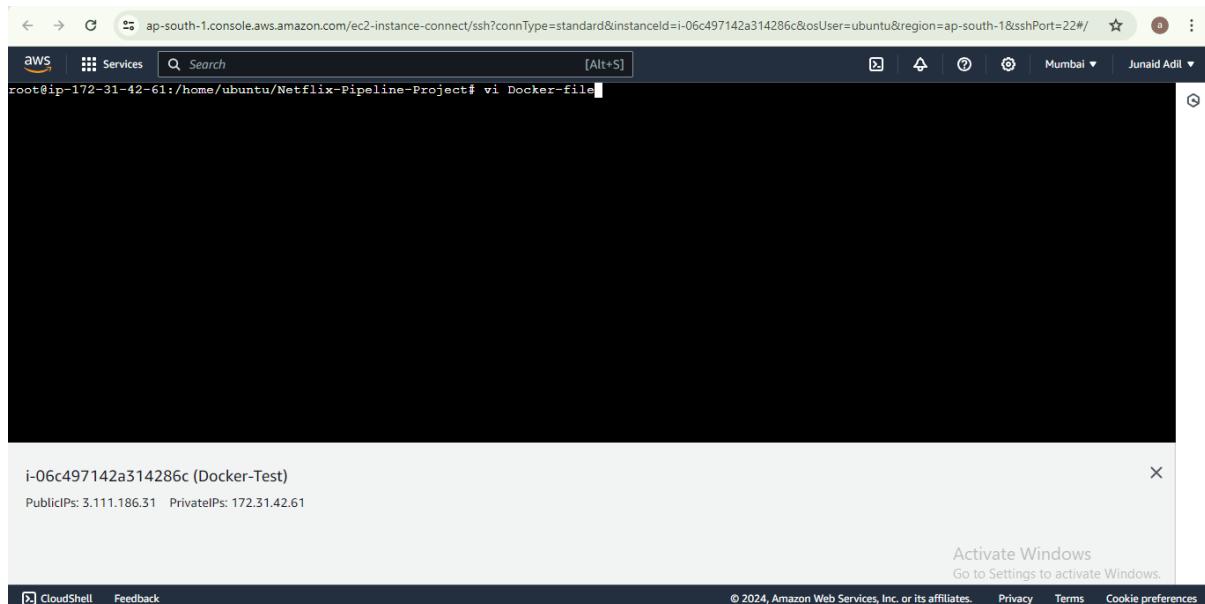
```
root@ip-172-31-42-61:/home/ubuntu/Netflix-Pipeline-Project# ls
pom.xml  src  target
root@ip-172-31-42-61:/home/ubuntu/Netflix-Pipeline-Project# cd target
root@ip-172-31-42-61:/home/ubuntu/Netflix-Pipeline-Project/target# ls
NETFLIX-1.2.2  NETFLIX-1.2.2.war  classes  maven-archiver  maven-status  surefire-reports  test-classes
root@ip-172-31-42-61:/home/ubuntu/Netflix-Pipeline-Project/target#
```

i-06c497142a314286c (Docker-Test)
Public IPs: 3.111.186.31 Private IPs: 172.31.42.61

Activate Windows
Go to Settings to activate Windows.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Step 11: Create a Docker file using command “vi Docker-file”



```
root@ip-172-31-42-61:/home/ubuntu/Netflix-Pipeline-Project# vi Docker-file
```

i-06c497142a314286c (Docker-Test)
Public IPs: 3.111.186.31 Private IPs: 172.31.42.61

Activate Windows
Go to Settings to activate Windows.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Add Instructions in the file.

The screenshot shows a CloudShell terminal window. The URL in the address bar is `ap-south-1.console.aws.amazon.com/ec2-instance-connect/ssh?connType=standard&instanceId=i-06c497142a314286c&osUser=ubuntu®ion=ap-south-1&sshPort=22#/`. The terminal content displays a Dockerfile:

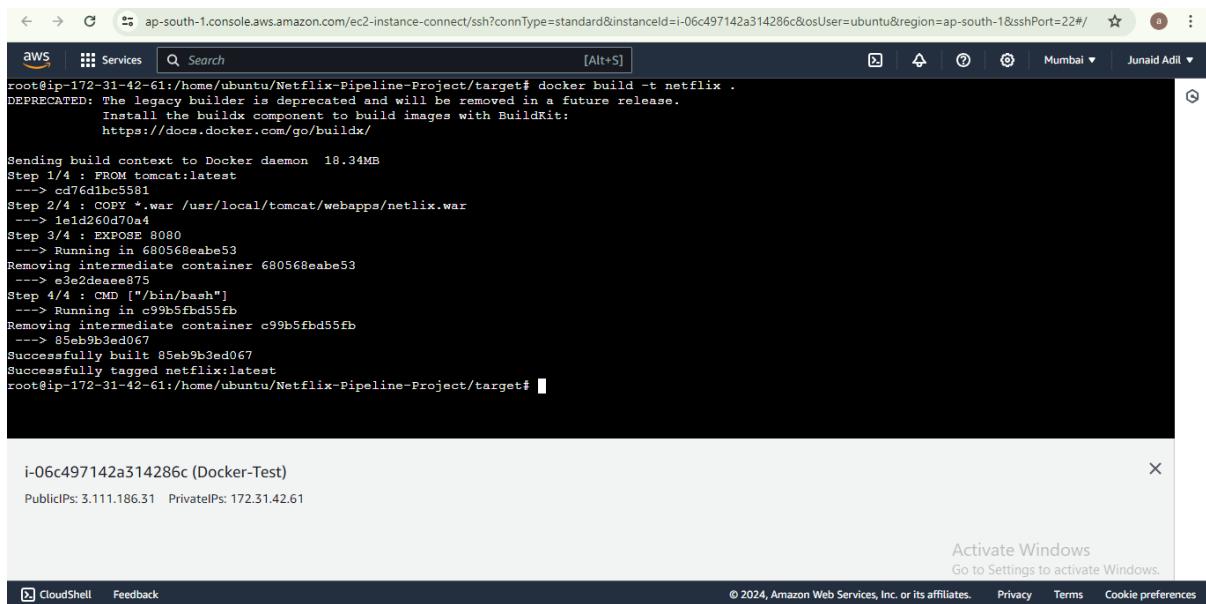
```
PROMT tomcat:latest
COPY *.war /usr/local/tomcat/webapps/netflix.war
EXPOSE 8080
CMD ["/bin/bash", "run"]
```

Below the terminal, a message box shows the instance details: `i-06c497142a314286c (Docker-Test)`, Public IPs: 3.111.186.31, Private IPs: 172.31.42.61. At the bottom right, there's an "Activate Windows" link.

Step 12: Build the Image from the created Dockerfile.

Use command “**docker build -t netflix .**”

The screenshot shows the same CloudShell terminal window. The command `root@ip-172-31-42-61:/home/ubuntu/Netflix-Pipeline-Project# docker build -t netflix .` has been entered and is being processed. The terminal is currently blacked out during the build process. Below the terminal, the instance details are shown again: `i-06c497142a314286c (Docker-Test)`, Public IPs: 3.111.186.31, Private IPs: 172.31.42.61. At the bottom right, there's an "Activate Windows" link.



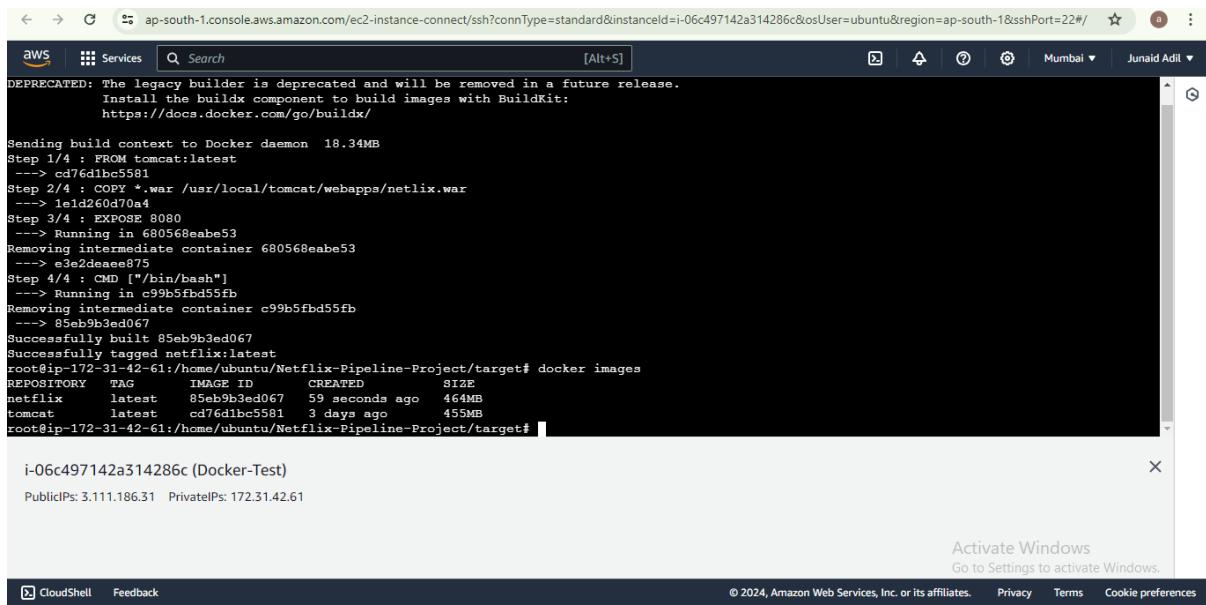
```
aws Services Search [Alt+S] Mumbai Junaid Adil
root@ip-172-31-42-61:/home/ubuntu/Netflix-Pipeline-Project/target# docker build -t netflix .
DEPRECATED: The legacy builder is deprecated and will be removed in a future release.
Install the buildx component to build images with BuildKit:
https://docs.docker.com/go/buildx/
Sending build context to Docker daemon 18.34MB
Step 1/4 : FROM tomcat:latest
--> cd76d1bc5581
Step 2/4 : COPY *.war /usr/local/tomcat/webapps/netflix.war
--> 1e1d260d70a4
Step 3/4 : EXPOSE 8080
--> Running in 680568eabe53
Removing intermediate container 680568eabe53
--> e3e2deee875
Step 4/4 : CMD ["/bin/bash"]
--> Running in c99b5fb55fb
Removing intermediate container c99b5fb55fb
--> 85eb9b3ed067
Successfully built 85eb9b3ed067
Successfully tagged netflix:latest
root@ip-172-31-42-61:/home/ubuntu/Netflix-Pipeline-Project/target#
```

i-06c497142a314286c (Docker-Test)
PublicIPs: 3.111.186.31 PrivateIPs: 172.31.42.61

Activate Windows
Go to Settings to activate Windows.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Step 13: Execute command “**docker images**” to check the images created



```
aws Services Search [Alt+S] Mumbai Junaid Adil
DEPRECATED: The legacy builder is deprecated and will be removed in a future release.
Install the buildx component to build images with BuildKit:
https://docs.docker.com/go/buildx/
Sending build context to Docker daemon 18.34MB
Step 1/4 : FROM tomcat:latest
--> cd76d1bc5581
Step 2/4 : COPY *.war /usr/local/tomcat/webapps/netflix.war
--> 1e1d260d70a4
Step 3/4 : EXPOSE 8080
--> Running in 680568eabe53
Removing intermediate container 680568eabe53
--> e3e2deee875
Step 4/4 : CMD ["/bin/bash"]
--> Running in c99b5fb55fb
Removing intermediate container c99b5fb55fb
--> 85eb9b3ed067
Successfully built 85eb9b3ed067
Successfully tagged netflix:latest
root@ip-172-31-42-61:/home/ubuntu/Netflix-Pipeline-Project/target# docker images
REPOSITORY TAG IMAGE ID CREATED SIZE
netflix latest 85eb9b3ed067 59 seconds ago 464MB
tomcat latest cd76d1bc5581 3 days ago 455MB
root@ip-172-31-42-61:/home/ubuntu/Netflix-Pipeline-Project/target#
```

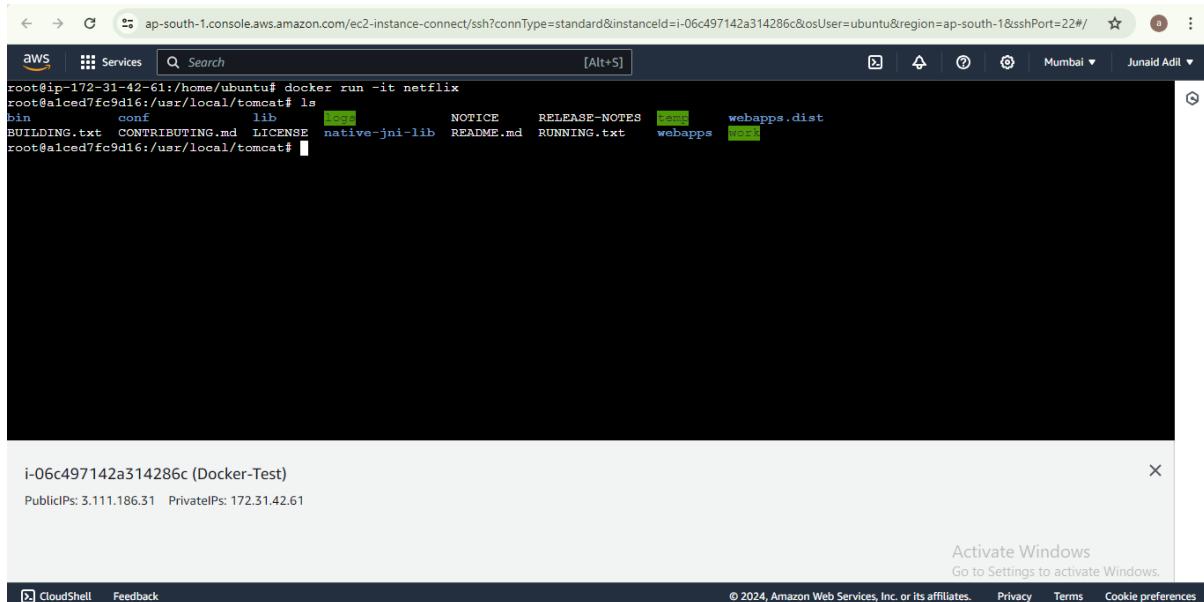
i-06c497142a314286c (Docker-Test)
PublicIPs: 3.111.186.31 PrivateIPs: 172.31.42.61

Activate Windows
Go to Settings to activate Windows.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

We can see the Images with the name Netflix has been created along with the Tomcat Image.

Step 14: Execute command “**docker run -it netflix**” to run the container.



The screenshot shows a terminal window in the AWS CloudShell interface. The user has run the command `docker run -it netflix`. The output shows the contents of the `netflix` Docker image, including files like `bin`, `conf`, `lib`, `log`, `NOTICE`, `RELEASE-NOTES`, `temp`, `webapps`, and `webapps.dist`. The terminal prompt is `root@ip-172-31-42-61:/home/ubuntu$`.

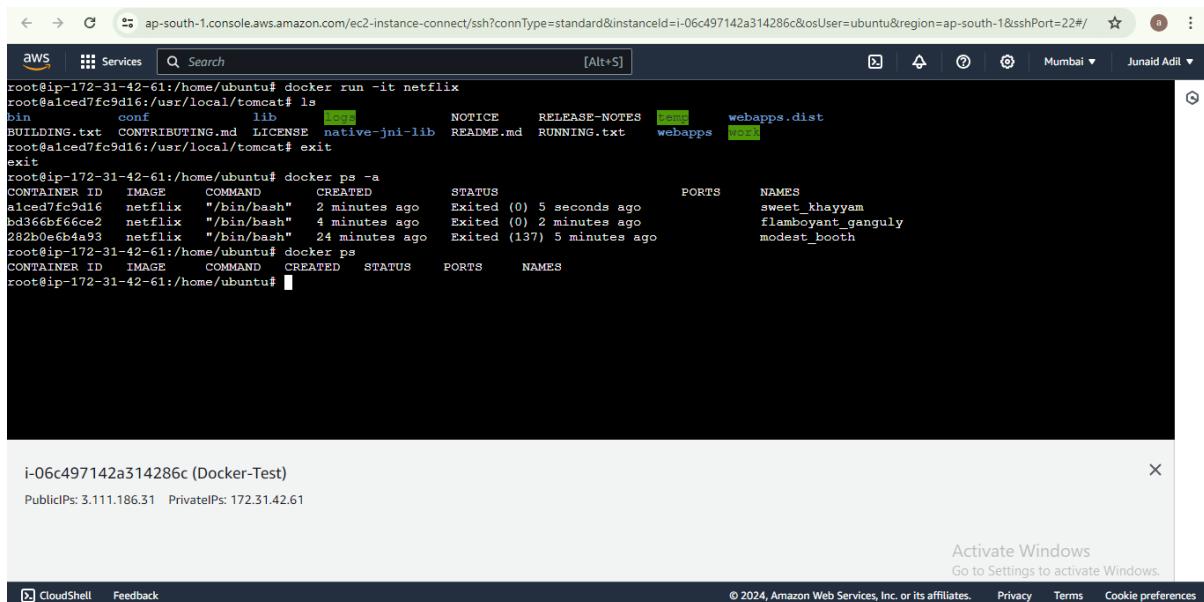
i-06c497142a314286c (Docker-Test)
PublicIPs: 3.111.186.31 PrivateIPs: 172.31.42.61

Activate Windows
Go to Settings to activate Windows.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

We can see the container has been started working and as it is running in interactive mode, we are inside the container

Step 15: Use command “**exit**” to exit from the container



The screenshot shows the continuation of the terminal session. After running `docker run -it netflix`, the user types `exit` to leave the container. The terminal then lists all running Docker containers with the command `docker ps -a`. It shows three containers:

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
alced7fc9d16	netflix	"bin/bash"	2 minutes ago	Exited (0) 5 seconds ago		sweet_khayyam
bd36bf66ce2	netflix	"bin/bash"	4 minutes ago	Exited (0) 2 minutes ago		flamboyant_ganguly
282b0e6b4a93	netflix	"bin/bash"	24 minutes ago	Exited (137) 5 minutes ago		modest_booth

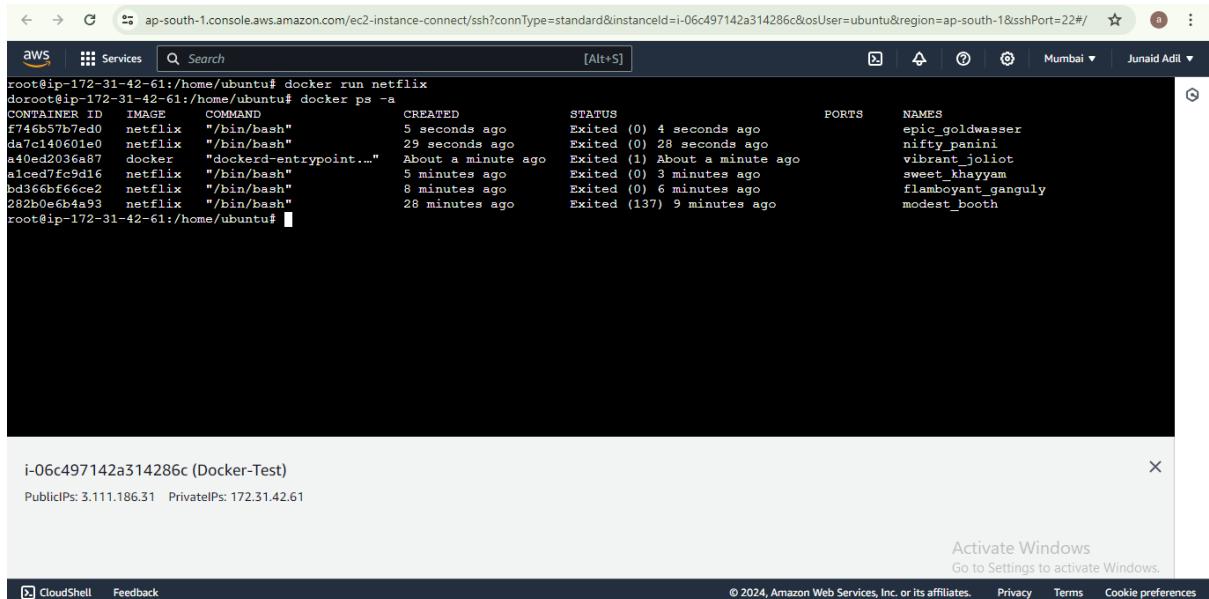
root@ip-172-31-42-61:/home/ubuntu\$ docker ps -a

i-06c497142a314286c (Docker-Test)
PublicIPs: 3.111.186.31 PrivateIPs: 172.31.42.61

Activate Windows
Go to Settings to activate Windows.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Step 16: To run the container we can use command “**docker run <application name>**”



The screenshot shows a terminal session in AWS CloudShell. The user has run the command `docker run netflix`, which creates a new container named `t746b57b7ed0`. The user then runs `docker ps -a` to list all containers, showing the newly created container along with others like `nifty_panini` and `vibrant_joliot`.

```
root@ip-172-31-42-61:/home/ubuntu# docker run netflix
root@ip-172-31-42-61:/home/ubuntu# docker ps -a
CONTAINER ID        IMAGE               COMMAND             CREATED            STATUS              PORTS               NAMES
t746b57b7ed0        netflix            "/bin/bash"         5 seconds ago      Exited (0) 4 seconds ago
da7c140601e0        netflix            "/bin/bash"         29 seconds ago     Exited (0) 28 seconds ago
a40ed2036a87        docker              "dockerd-entrypoint...."   About a minute ago   Exited (1) About a minute ago
alced7fc9d16        netflix            "/bin/bash"         5 minutes ago      Exited (0) 3 minutes ago
bd36bf66ce2         netflix            "/bin/bash"         8 minutes ago      Exited (0) 6 minutes ago
282b0e6b4a93        netflix            "/bin/bash"         28 minutes ago     Exited (137) 9 minutes ago
root@ip-172-31-42-61:/home/ubuntu#
```

Below the terminal, the instance details are shown:

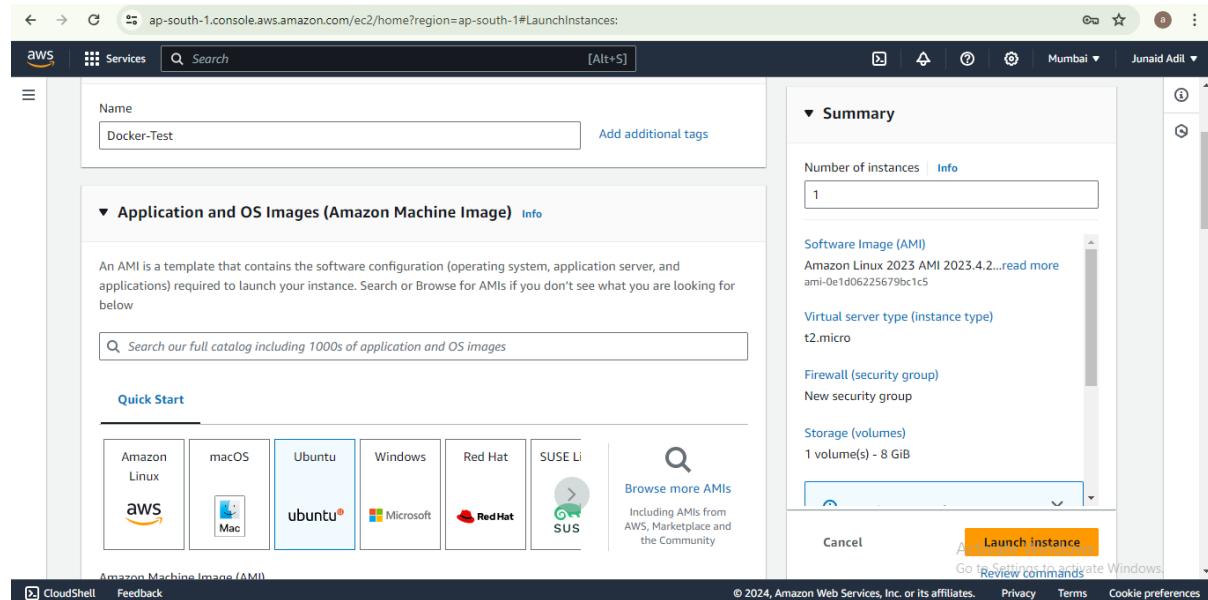
i-06c497142a314286c (Docker-Test)
PublicIPs: 3.111.186.31 PrivateIPs: 172.31.42.61

At the bottom right, there are links for "Activate Windows", "CloudShell", "Feedback", "Privacy", "Terms", and "Cookie preferences".

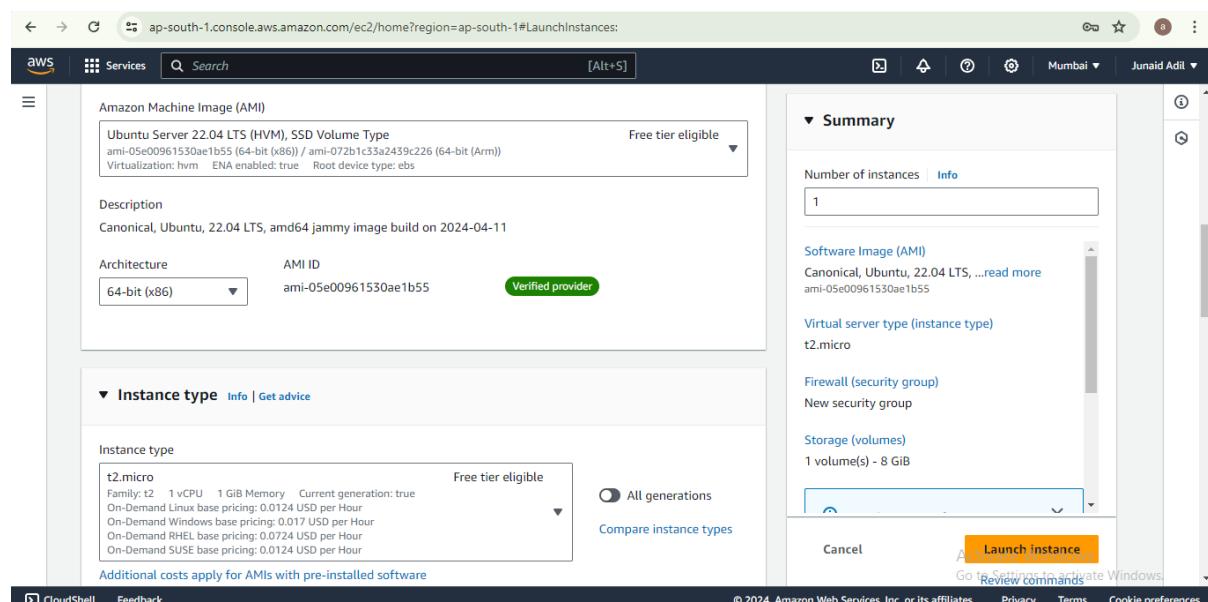
We can see the container is working.

L2 - Create the Container using the same Application Image and run the application in a Web Browser using container port mapping

Step 1: Create an Instance



The screenshot shows the AWS EC2 Instances creation page. In the 'Quick Start' section, the 'Ubuntu' option is selected. Other options shown include Amazon Linux, macOS, Windows, Red Hat, and SUSE. The 'Launch instance' button is highlighted in orange at the bottom right.



The screenshot shows the 'Amazon Machine Image (AMI)' selection step. The 'Ubuntu Server 22.04 LTS (HVM, SSD Volume Type)' AMI is selected, showing its details: ami-05e00961530ae1b55 (64-bit (x86)) / ami-072b1c33a2439c226 (64-bit (Arm)). The 'Launch instance' button is highlighted in orange at the bottom right.

ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#LaunchInstances:

aws Services Search [Alt+S] Mumbai Junaid Adil

EC2 Instances Launch an instance

Launching instance
Creating security group rules 21%

▶ Details

Please wait while we launch your instance.
Do not close your browser while this is loading.

Activate Windows
Go to Settings to activate Windows.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#Instances:instanceId=i-06c497142a314286c

aws Services Search [Alt+S] Mumbai Junaid Adil

EC2 Dashboard EC2 Global View Events

Instances Instances Instance Types Launch Templates Spot Requests Savings Plans Reserved Instances Dedicated Hosts Capacity Reservations

Images AMIs AMI Catalog

Elastic Block Store Volumes

Instances (1/1) Info Connect Instance state Actions Launch instances

Find Instance by attribute or tag (case-sensitive) All states

Instance ID = i-06c497142a314286c Clear filters

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability
Docker-Test	i-06c497142a314286c	Running	t2.micro	Initializing	View alarms +	ap-south-1

i-06c497142a314286c (Docker-Test)

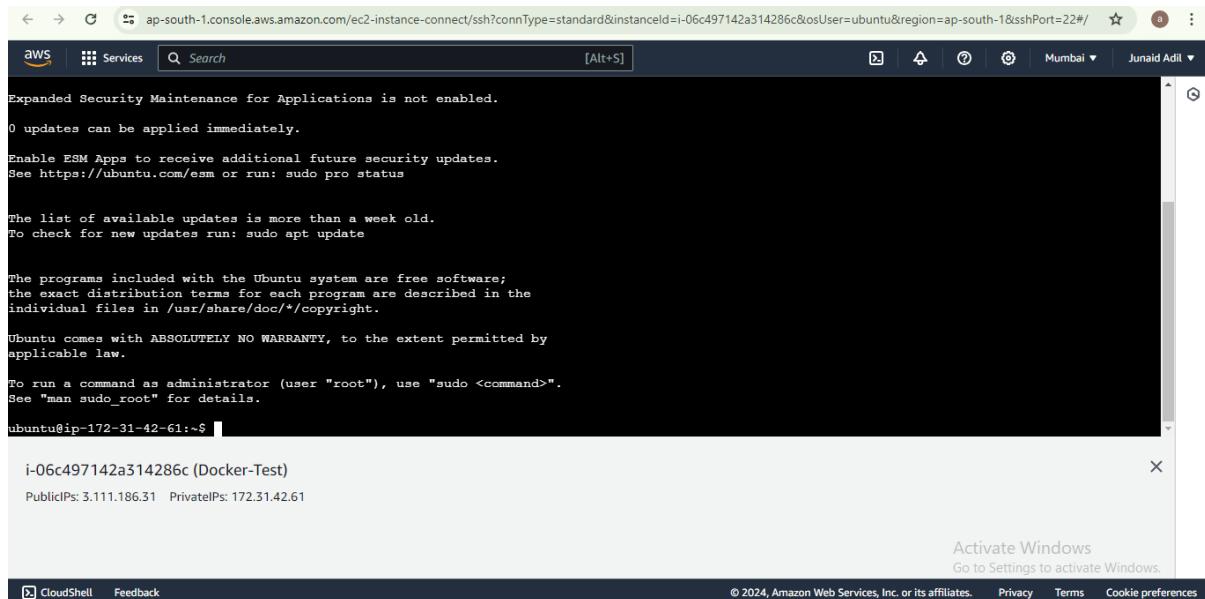
Details Status and alarms Monitoring Security Networking Storage Tags

Instance summary Info

Instance ID	Public IPv4 address	Private IPv4 addresses
i-06c497142a314286c (Docker-Test)	3.111.186.31 open address	172.31.42.61
IPv6 address	Instance state	Public IPv4 DNS
-	Running	Activate Windows ec2-3-111-186-31.ap-south-1... Go to Settings to activate Windows.

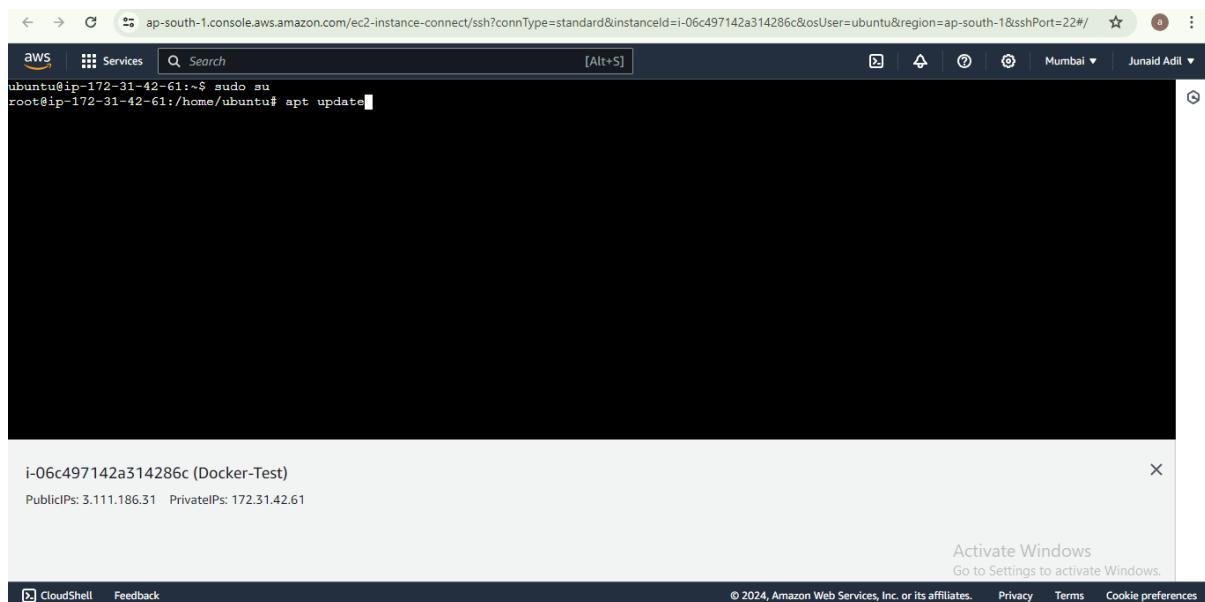
CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Step 2: Connect to the Instance

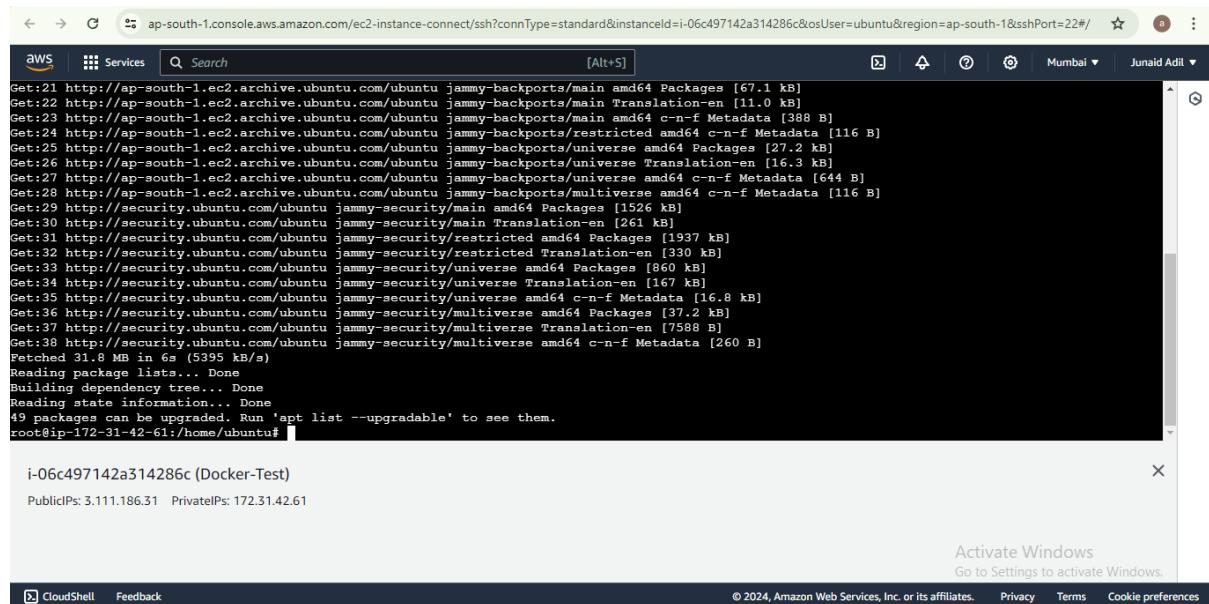


```
← → ⌂ ap-south-1.console.aws.amazon.com/ec2-instance-connect/ssh?connType=standard&instanceId=i-06c497142a314286c&osUser=ubuntu&region=ap-south-1&sshPort=22#/  
AWS Services Search [Alt+S] Mumbai Junaid Adil  
  
Expanded Security Maintenance for Applications is not enabled.  
0 updates can be applied immediately.  
Enable ESM Apps to receive additional future security updates.  
See https://ubuntu.com/esm or run: sudo pro status  
  
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-172-31-42-61:~$  
  
i-06c497142a314286c (Docker-Test)  
Public IPs: 3.111.186.31 Private IPs: 172.31.42.61  
  
Activate Windows  
Go to Settings to activate Windows.  
CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences
```

Step 3: Become root user and run the command to update packages “apt update”



```
← → ⌂ ap-south-1.console.aws.amazon.com/ec2-instance-connect/ssh?connType=standard&instanceId=i-06c497142a314286c&osUser=ubuntu&region=ap-south-1&sshPort=22#/  
AWS Services Search [Alt+S] Mumbai Junaid Adil  
ubuntu@ip-172-31-42-61:~$ sudo su  
root@ip-172-31-42-61:/home/ubuntu# apt update  
  
i-06c497142a314286c (Docker-Test)  
Public IPs: 3.111.186.31 Private IPs: 172.31.42.61  
  
Activate Windows  
Go to Settings to activate Windows.  
CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences
```

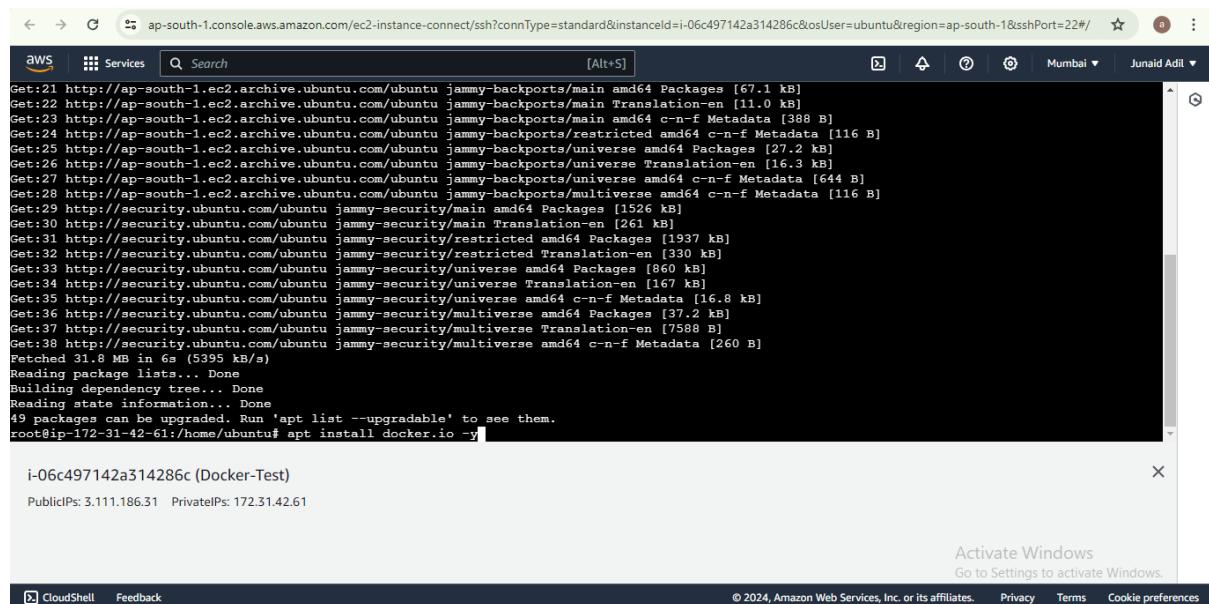


```
Get:21 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/main amd64 Packages [67.1 kB]
Get:22 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/main Translation-en [11.0 kB]
Get:23 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/main amd64 c-n-f Metadata [388 B]
Get:24 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/restricted amd64 c-n-f Metadata [116 B]
Get:25 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/universe amd64 Packages [27.2 kB]
Get:26 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/universe Translation-en [16.3 kB]
Get:27 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/universe amd64 c-n-f Metadata [644 B]
Get:28 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/multiverse amd64 c-n-f Metadata [116 B]
Get:29 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-security/main amd64 Packages [1526 kB]
Get:30 http://security.ubuntu.com/ubuntu jammy-security/main Translation-en [261 kB]
Get:31 http://security.ubuntu.com/ubuntu jammy-security/restricted amd64 Packages [1937 kB]
Get:32 http://security.ubuntu.com/ubuntu jammy-security/restricted Translation-en [330 kB]
Get:33 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 Packages [860 kB]
Get:34 http://security.ubuntu.com/ubuntu jammy-security/universe Translation-en [167 kB]
Get:35 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 c-n-f Metadata [16.8 kB]
Get:36 http://security.ubuntu.com/ubuntu jammy-security/multiverse amd64 Packages [37.2 kB]
Get:37 http://security.ubuntu.com/ubuntu jammy-security/multiverse Translation-en [7588 B]
Get:38 http://security.ubuntu.com/ubuntu jammy-security/multiverse amd64 c-n-f Metadata [260 B]
Fetched 31.8 MB in 6s (5395 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
49 packages can be upgraded. Run 'apt list --upgradable' to see them.
root@ip-172-31-42-61:/home/ubuntu#
```

i-06c497142a314286c (Docker-Test)
PublicIPs: 3.111.186.31 PrivateIPs: 172.31.42.61

Activate Windows
Go to Settings to activate Windows.

Step 4: Install Docker using command “**apt install docker.io -y**”



```
Get:21 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/main amd64 Packages [67.1 kB]
Get:22 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/main Translation-en [11.0 kB]
Get:23 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/main amd64 c-n-f Metadata [388 B]
Get:24 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/restricted amd64 c-n-f Metadata [116 B]
Get:25 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/universe amd64 Packages [27.2 kB]
Get:26 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/universe Translation-en [16.3 kB]
Get:27 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/universe amd64 c-n-f Metadata [644 B]
Get:28 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/multiverse amd64 c-n-f Metadata [116 B]
Get:29 http://security.ubuntu.com/ubuntu jammy-security/main amd64 Packages [1526 kB]
Get:30 http://security.ubuntu.com/ubuntu jammy-security/main Translation-en [261 kB]
Get:31 http://security.ubuntu.com/ubuntu jammy-security/restricted amd64 Packages [1937 kB]
Get:32 http://security.ubuntu.com/ubuntu jammy-security/restricted Translation-en [330 kB]
Get:33 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 Packages [860 kB]
Get:34 http://security.ubuntu.com/ubuntu jammy-security/universe Translation-en [167 kB]
Get:35 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 c-n-f Metadata [16.8 kB]
Get:36 http://security.ubuntu.com/ubuntu jammy-security/multiverse amd64 Packages [37.2 kB]
Get:37 http://security.ubuntu.com/ubuntu jammy-security/multiverse Translation-en [7588 B]
Get:38 http://security.ubuntu.com/ubuntu jammy-security/multiverse amd64 c-n-f Metadata [260 B]
Fetched 31.8 MB in 6s (5395 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
49 packages can be upgraded. Run 'apt list --upgradable' to see them.
root@ip-172-31-42-61:/home/ubuntu# apt install docker.io -y
```

i-06c497142a314286c (Docker-Test)
PublicIPs: 3.111.186.31 PrivateIPs: 172.31.42.61

Activate Windows
Go to Settings to activate Windows.

```
← → ⚙ ap-south-1.console.aws.amazon.com/ec2-instance-connect/ssh?connType=standard&instanceId=i-06c497142a314286c&osUser=ubuntu&region=ap-south-1&sshPort=22#/
aws Services Search [Alt+S] Mumbai Junaid Adil ▾
Unpacking runc (1.1.12-0ubuntu2~22.04.1) ...
Selecting previously unselected package containerd.
Preparing to unpack .../3-containerd_1.7.12-0ubuntu2~22.04.1_amd64.deb ...
Unpacking containerd (1.7.12-0ubuntu2~22.04.1) ...
Selecting previously unselected package dns-root-data.
Preparing to unpack .../4-dns-root-data_2023112702-ubuntu0.22.04.1_all.deb ...
Unpacking dns-root-data (2023112702-ubuntu0.22.04.1) ...
Selecting previously unselected package dnsmasq-base.
Preparing to unpack .../5-dnsmasq-base_2.90-0ubuntu0.22.04.1_amd64.deb ...
Unpacking dnsmasq-base (2.90-0ubuntu0.22.04.1) ...
Selecting previously unselected package docker.io.
Preparing to unpack .../6-docker.io_24.0.7-0ubuntu2~22.04.1_amd64.deb ...
Unpacking docker.io (24.0.7-0ubuntu2~22.04.1) ...
Selecting previously unselected package ubuntu-fan.
Preparing to unpack .../7-ubuntu-fan_0.12.16_all.deb ...
Unpacking ubuntu-fan (0.12.16) ...
Setting up dnsmasq-base (2.90-0ubuntu0.22.04.1) ...
Setting up runc (1.1.12-0ubuntu2~22.04.1) ...
Setting up dns-root-data (2023112702-ubuntu0.22.04.1) ...
Setting up bridge-utils (1.7-1ubuntu3) ...
Setting up pigz (2.6-1) ...
Setting up containerd (1.7.12-0ubuntu2~22.04.1) ...

[########################################] 100% Done

i-06c497142a314286c (Docker-Test)
PublicIPs: 3.111.186.31 PrivateIPs: 172.31.42.61

Activate Windows
Go to Settings to activate Windows.

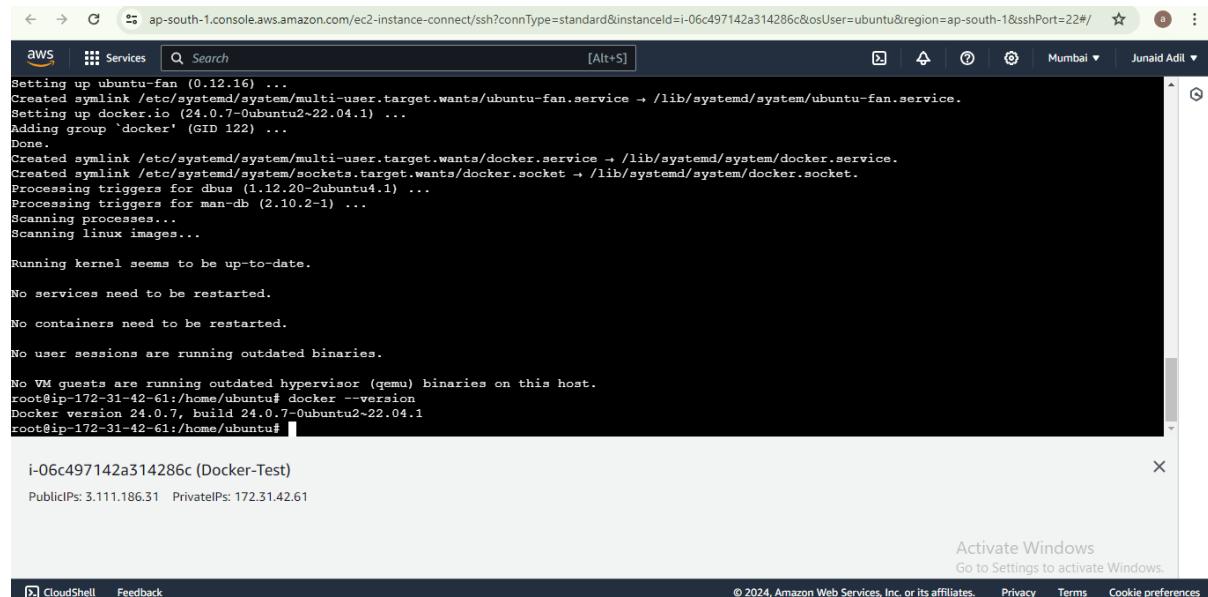
CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences
```

```
← → ⚙ ap-south-1.console.aws.amazon.com/ec2-instance-connect/ssh?connType=standard&instanceId=i-06c497142a314286c&osUser=ubuntu&region=ap-south-1&sshPort=22#/
aws Services Search [Alt+S] Mumbai Junaid Adil ▾
Setting up containerd (1.7.12-0ubuntu2~22.04.1) ...
Created symlink /etc/systemd/system/multi-user.target.wants/containerd.service → /lib/systemd/system/containerd.service.
Setting up ubuntu-fan (0.12.16) ...
Created symlink /etc/systemd/system/multi-user.target.wants/ubuntu-fan.service → /lib/systemd/system/ubuntu-fan.service.
Setting up docker.io (24.0.7-0ubuntu2~22.04.1) ...
Adding group 'docker' (GID 122) ...
Done.
Created symlink /etc/systemd/system/multi-user.target.wants/docker.service → /lib/systemd/system/docker.service.
Created symlink /etc/systemd/system/sockets.target.wants/docker.socket → /lib/systemd/system/docker.socket.
Processing triggers for dbus (1.12.20-2ubuntu4.1) ...
Processing triggers for man-db (2.10.2-1) ...
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-172-31-42-61:/home/ubuntu# 
```

Step 5: We can check the docker version using command “**docker --version**”



```
Setting up ubuntu-fan (0.12.16) ...
Created symlink /etc/systemd/system/multi-user.target.wants/ubuntu-fan.service → /lib/systemd/system/ubuntu-fan.service.
Setting up docker.io (24.0.7-0ubuntu2~22.04.1) ...
Adding group 'docker' (GID 122) ...
Done.
Created symlink /etc/systemd/system/multi-user.target.wants/docker.service → /lib/systemd/system/docker.service.
Created symlink /etc/systemd/system/sockets.target.wants/docker.socket → /lib/systemd/system/docker.socket.
Processing triggers for dbus (1.12.20-2ubuntu4.1) ...
Processing triggers for man-db (2.10.2-1) ...
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.

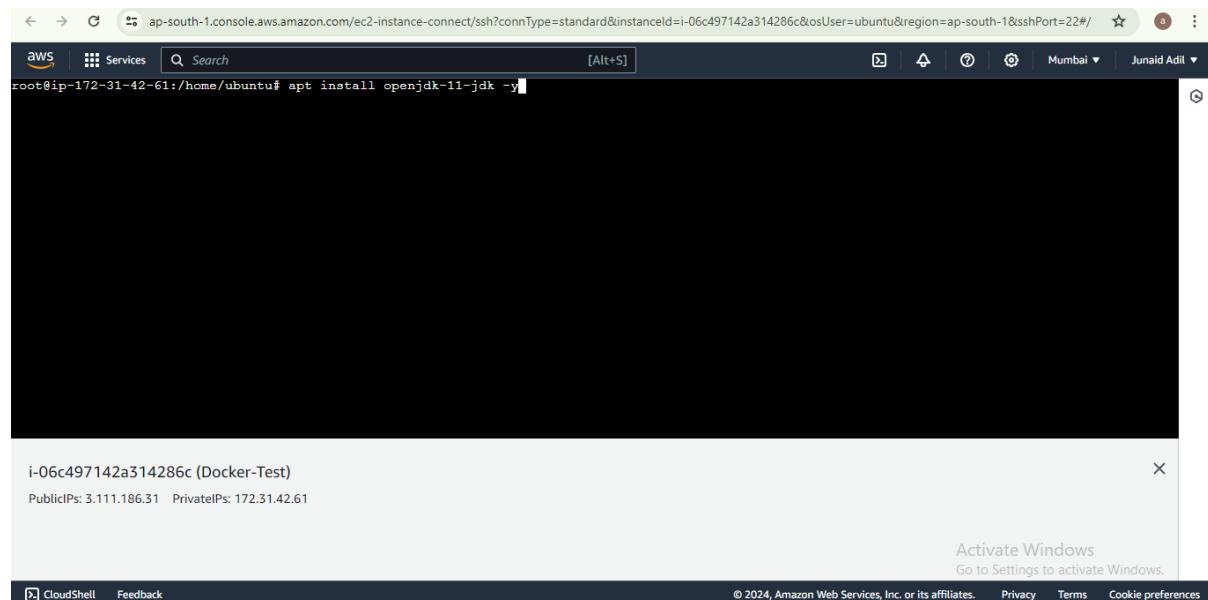
No VM guests are running outdated hypervisor (qemu) binaries on this host.

root@ip-172-31-42-61:/home/ubuntu# docker --version
Docker version 24.0.7, build 24.0.7-0ubuntu2~22.04.1
root@ip-172-31-42-61:/home/ubuntu#
```

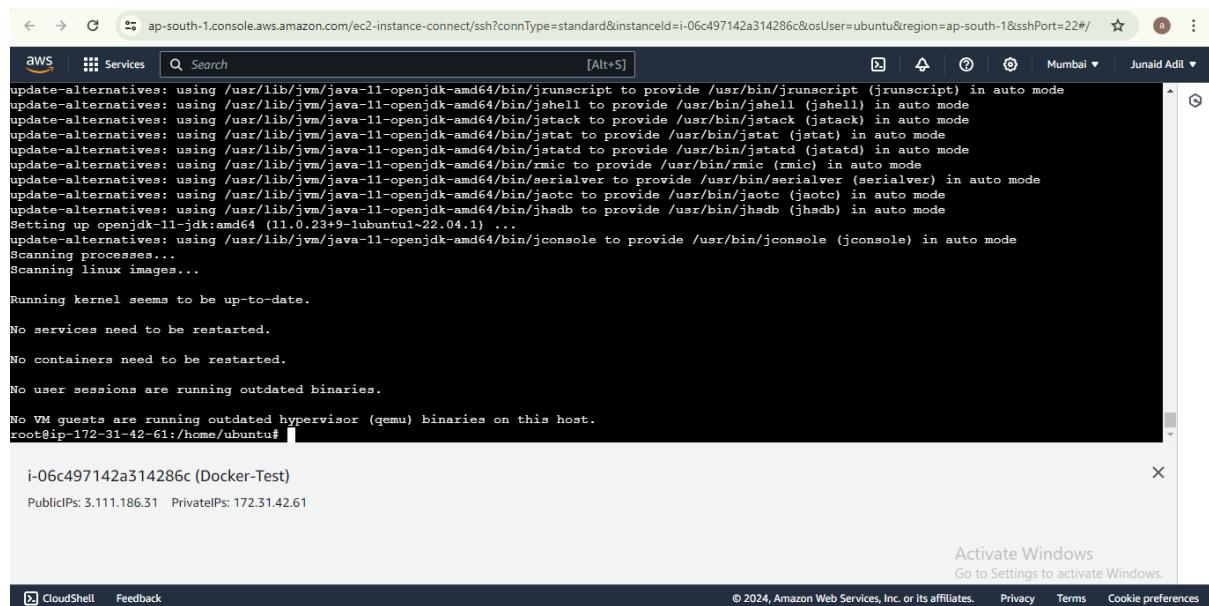
Step 6: Install Java, Maven using command

Java – “**apt install openjdk-11-jdk -y**”

Maven – “**apt install maven -y**”



```
root@ip-172-31-42-61:/home/ubuntu# apt install openjdk-11-jdk -y
```



```
aws Services Search [Alt+S] Mumbai ▾ Junaid Adil ▾
update-alternatives: using /usr/lib/jvm/java-11-openjdk-amd64/bin/jrunscript to provide /usr/bin/jrunscript (jrunscript) in auto mode
update-alternatives: using /usr/lib/jvm/java-11-openjdk-amd64/bin/jshell to provide /usr/bin/jshell (jshell) in auto mode
update-alternatives: using /usr/lib/jvm/java-11-openjdk-amd64/bin/jstack to provide /usr/bin/jstack (jstack) in auto mode
update-alternatives: using /usr/lib/jvm/java-11-openjdk-amd64/bin/jstat to provide /usr/bin/jstat (jstat) in auto mode
update-alternatives: using /usr/lib/jvm/java-11-openjdk-amd64/bin/jstatd to provide /usr/bin/jstatd (jstatd) in auto mode
update-alternatives: using /usr/lib/jvm/java-11-openjdk-amd64/bin/rmic to provide /usr/bin/rmic (rmic) in auto mode
update-alternatives: using /usr/lib/jvm/java-11-openjdk-amd64/bin/serialver to provide /usr/bin/serialver (serialver) in auto mode
update-alternatives: using /usr/lib/jvm/java-11-openjdk-amd64/bin/jactc to provide /usr/bin/jactc (jactc) in auto mode
update-alternatives: using /usr/lib/jvm/java-11-openjdk-amd64/bin/jhsdb to provide /usr/bin/jhsdb (jhsdb) in auto mode
Setting up openjdk-11-jdk-amd64 (11.0.23+9-lubuntu1-22.04.1) ...
update-alternatives: using /usr/lib/jvm/java-11-openjdk-amd64/bin/jconsole to provide /usr/bin/jconsole (jconsole) in auto mode
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.

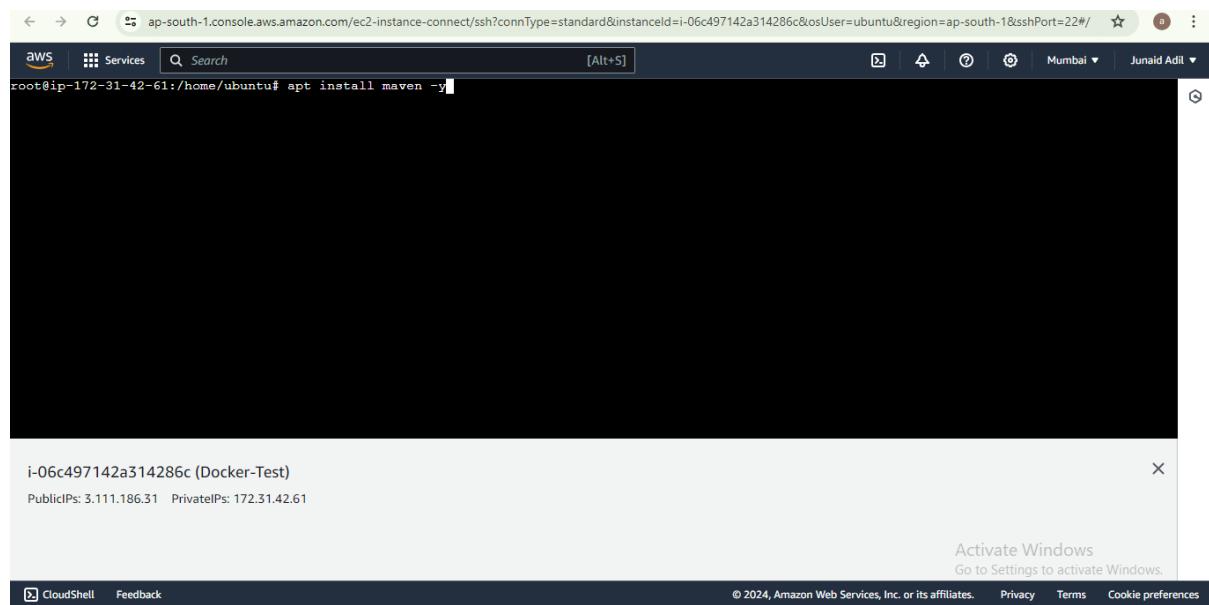
No user sessions are running outdated binaries.

No VM guests are running outdated hypervisor (qemu) binaries on this host.
root@ip-172-31-42-61:/home/ubuntu#
```

i-06c497142a314286c (Docker-Test)
Public IPs: 3.111.186.31 Private IPs: 172.31.42.61

Activate Windows
Go to Settings to activate Windows.

Maven Installation:



```
aws Services Search [Alt+S] Mumbai ▾ Junaid Adil ▾
root@ip-172-31-42-61:/home/ubuntu# apt install maven -y
```

i-06c497142a314286c (Docker-Test)
Public IPs: 3.111.186.31 Private IPs: 172.31.42.61

Activate Windows
Go to Settings to activate Windows.

```
aws Services Search [Alt+S] Mumbai ▾ Junaid Adil ▾
← → ⌂ ap-south-1.console.aws.amazon.com/ec2-instance-connect/ssh?connType=standard&instanceId=i-06c497142a314286c&osUser=ubuntu&region=ap-south-1&sshPort=22#/
Preparing to unpack .../10-libguava-java_29.0-6_all.deb ...
Unpacking libguava-java (29.0-6) ...
Selecting previously unselected package libcapalliance-java.
Preparing to unpack .../11-libcapalliance-java_20070526-6_all.deb ...
Unpacking libcapalliance-java (20070526-6) ...
Selecting previously unselected package libguice-java.
Preparing to unpack .../12-libguice-java_4.2.3-2_all.deb ...
Unpacking libguice-java (4.2.3-2) ...
Selecting previously unselected package libhawtjni-runtime-java.
Preparing to unpack .../13-libhawtjni-runtime-java_1.17-1_all.deb ...
Unpacking libhawtjni-runtime-java (1.17-1) ...
Selecting previously unselected package libjansi-native-java.
Preparing to unpack .../14-libjansi-native-java_1.8-1_all.deb ...
Unpacking libjansi-native-java (1.8-1) ...
Selecting previously unselected package libjansi-java.
Preparing to unpack .../15-libjansi-java_1.18-1_all.deb ...
Unpacking libjansi-java (1.18-1) ...
Selecting previously unselected package libmaven-parent-java.
Preparing to unpack .../16-libmaven-parent-java_31-2_all.deb ...
Unpacking libmaven-parent-java (31-2) ...
Selecting previously unselected package libplexus-utils2-java.
Preparing to unpack .../17-libplexus-utils2-java_3.3.0-1_all.deb ...

Progress: [====] 100% [.....] .
```

```
aws Services Search [Alt+S] Mumbai Junaid Adil

Setting up libguava-java (29.0-6) ...
Setting up libcommons-lang3-java (3.11-1) ...
Setting up libjansi-native-java (1.8-1) ...
Setting up libwagon-file-java (3.3.4-1) ...
Setting up libcommons-io-java (2.11.0-2) ...
Setting up libguice-java (4.2.3-2) ...
Setting up libjansi-java (1.18-1) ...
Setting up libmaven-shared-utils-java (3.3.0-1ubuntu0.22.04.1) ...
Setting up libmaven3-core-java (3.6.3-5) ...
Setting up maven (3.6.3-5) ...
update-alternatives: using /usr/share/maven/bin/mvn to provide /usr/bin/mvn (mvn) in auto mode
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.

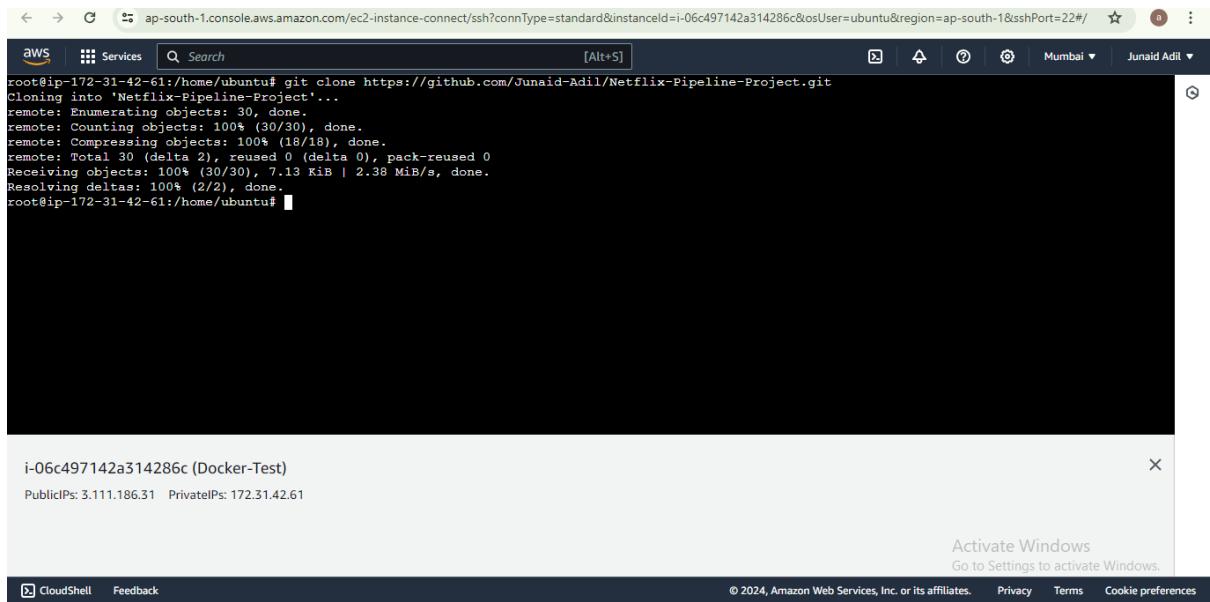
No user sessions are running outdated binaries.

No VM guests are running outdated hypervisor (qemu) binaries on this host.
root@ip-172-31-42-61:/home/ubuntu#
```

Step 7: Clone the Repository using command “`git clone <HTTPS code>`”. Where the Artifacts is built and saved in a Repository “Netflix-Pipeline-Project”.

The screenshot shows the GitHub repository page for 'Junaid-Adil / Netflix-Pipeline-Project'. The repository is public and contains one branch ('main') and one tag ('1 Branch'). The main file listed is 'pom.xml', which has been updated by 'Junaid-Adil'. A 'README' file is present but empty. On the right side, there's a 'Clone' section with options for Local, Codespaces, HTTPS (selected), SSH, and GitHub CLI. The HTTPS URL is highlighted with a red box. Other sections include 'About' (no description), 'Activity' (0 stars, 1 watching, 0 forks), 'Releases' (no releases), and 'Packages' (no packages). A 'Copy url to clipboard' button is also visible.

The screenshot shows an AWS CloudShell terminal window. The user is running the command `git clone https://github.com/Junaid-Adil/Netflix-Pipeline-Project.git`. The terminal output shows the repository being cloned. Below the terminal, a message box displays the instance ID (i-06c497142a314286c) and Docker Test status. At the bottom, there are links for CloudShell, Feedback, and a copyright notice for Amazon Web Services.



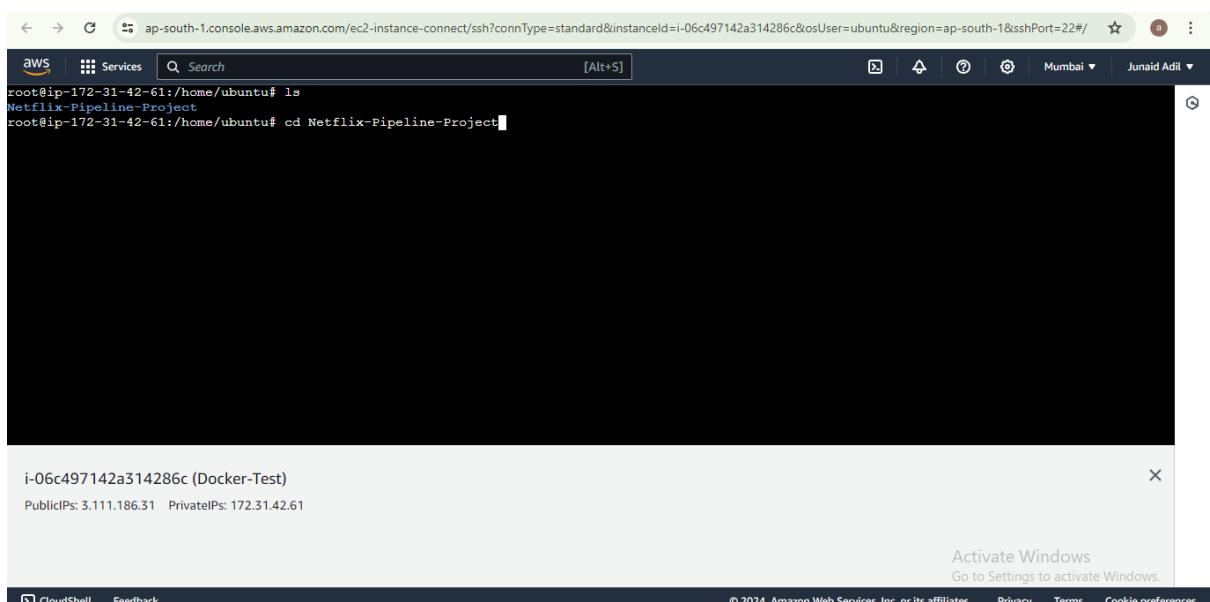
```
aws Services Search [Alt+S] Mumbai Junaid Adil
root@ip-172-31-42-61:/home/ubuntu# git clone https://github.com/Junaid-Adil/Netflix-Pipeline-Project.git
Cloning into 'Netflix-Pipeline-Project'...
remote: Enumerating objects: 30, done.
remote: Counting objects: 100% (30/30), done.
remote: Compressing objects: 100% (18/18), done.
remote: Total 30 (delta 2), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (30/30), 7.13 KiB | 2.38 MiB/s, done.
Resolving deltas: 100% (2/2), done.
root@ip-172-31-42-61:/home/ubuntu#
```

i-06c497142a314286c (Docker-Test)
PublicIPs: 3.111.186.31 PrivateIPs: 172.31.42.61

Activate Windows
Go to Settings to activate Windows.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Step 8: Go to the project directory



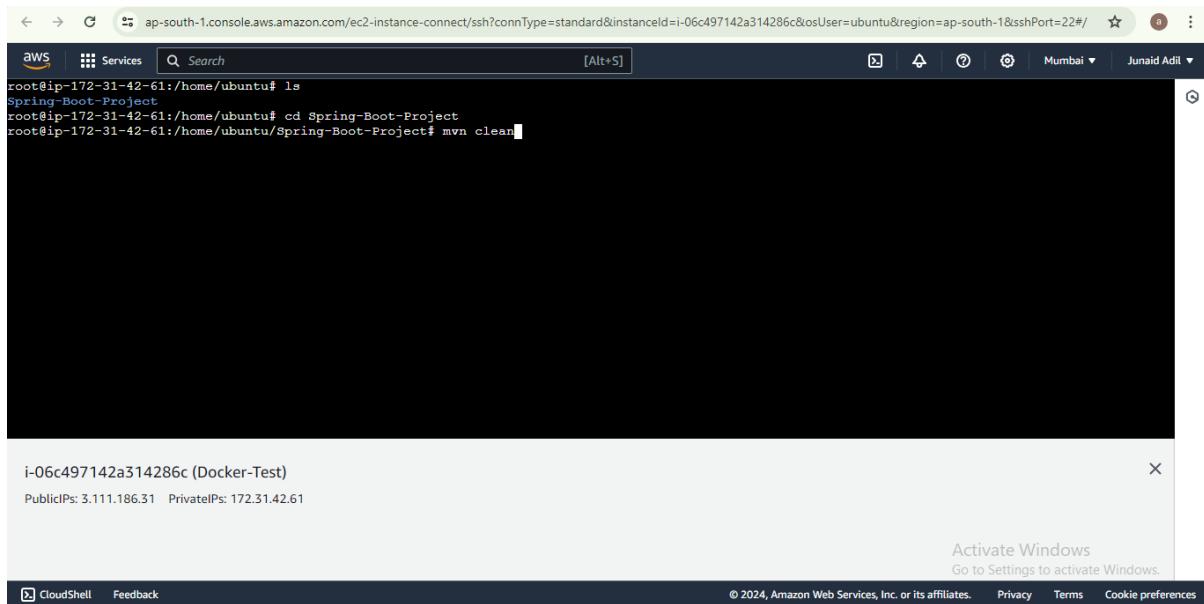
```
aws Services Search [Alt+S] Mumbai Junaid Adil
root@ip-172-31-42-61:/home/ubuntu# ls
Netflix-Pipeline-Project
root@ip-172-31-42-61:/home/ubuntu# cd Netflix-Pipeline-Project
```

i-06c497142a314286c (Docker-Test)
PublicIPs: 3.111.186.31 PrivateIPs: 172.31.42.61

Activate Windows
Go to Settings to activate Windows.

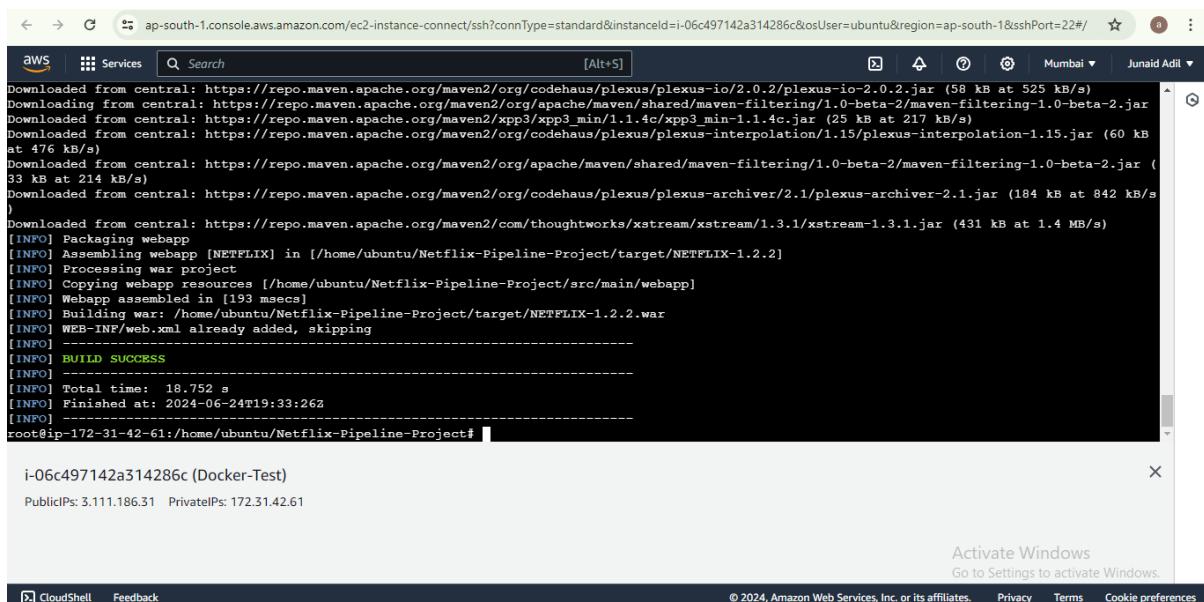
CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Step 9: Run command “mvn clean” and then execute “mvn package”



```
aws Services Search [Alt+S] Junaid Adil Mumbai i-06c497142a314286c (Docker-Test) PublicIPs: 3.111.186.31 PrivateIPs: 172.31.42.61 Activate Windows Go to Settings to activate Windows. © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences
```

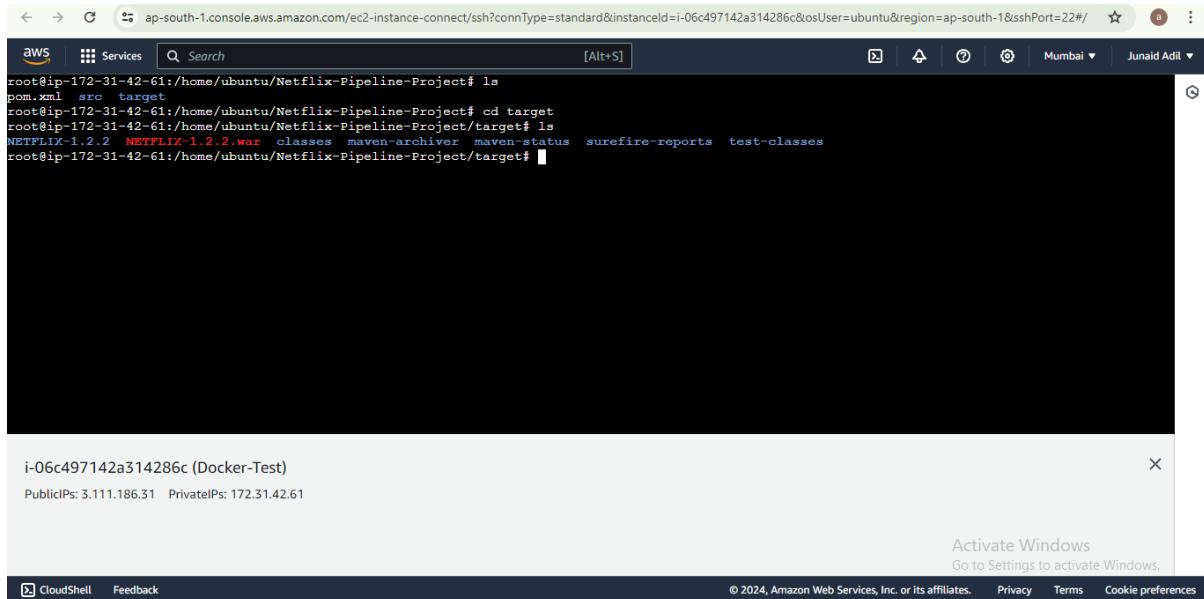
```
root@ip-172-31-42-61:/home/ubuntu# ls
Spring-Boot-Project
root@ip-172-31-42-61:/home/ubuntu# cd Spring-Boot-Project
root@ip-172-31-42-61:/home/ubuntu/Spring-Boot-Project# mvn clean
```



```
aws Services Search [Alt+S] Junaid Adil Mumbai i-06c497142a314286c (Docker-Test) PublicIPs: 3.111.186.31 PrivateIPs: 172.31.42.61 Activate Windows Go to Settings to activate Windows. © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences
```

```
Downloaded from central: https://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-io/2.0.2/plexus-io-2.0.2.jar (58 kB at 525 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/shared/maven-filtering/1.0-beta-2/maven-filtering-1.0-beta-2.jar
Downloaded from central: https://repo.maven.apache.org/maven2/xpp3/xpp3_min/1.1.4c/xpp3_min-1.1.4c.jar (25 kB at 217 kB/s)
Downloaded from central: https://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-interpolation/1.15/plexus-interpolation-1.15.jar (60 kB at 476 kB/s)
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/shared/maven-filtering/1.0-beta-2/maven-filtering-1.0-beta-2.jar (33 kB at 214 kB/s)
Downloaded from central: https://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-archiver/2.1/plexus-archiver-2.1.jar (184 kB at 842 kB/s)
Downloaded from central: https://repo.maven.apache.org/maven2/com/thoughtworks/xstream/xstream/1.3.1/xstream-1.3.1.jar (431 kB at 1.4 MB/s)
[INFO] Packaging webapp
[INFO] Assembling webapp [NETFLIX] in [/home/ubuntu/Netflix-Pipeline-Project/target/NETFLIX-1.2.2]
[INFO] Processing war project
[INFO] Copying webapp resources [/home/ubuntu/Netflix-Pipeline-Project/src/main/webapp]
[INFO] Webapp assembled in [193 msecs]
[INFO] Building war: /home/ubuntu/Netflix-Pipeline-Project/target/NETFLIX-1.2.2.war
[INFO] WEB-INF/web.xml already added, skipping
[INFO]
[INFO] BUILD SUCCESS
[INFO]
[INFO] Total time: 18.752 s
[INFO] Finished at: 2024-06-24T19:33:26Z
[INFO]
```

Step 10: We can check the .war file in Target folder



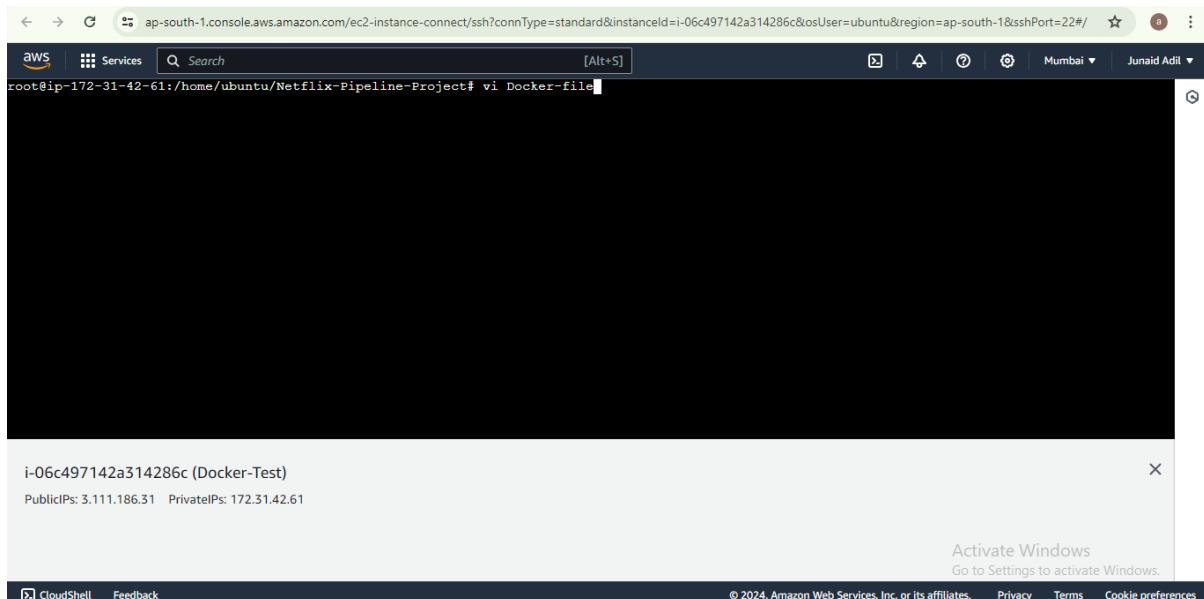
```
root@ip-172-31-42-61:/home/ubuntu/Netflix-Pipeline-Project# ls
pom.xml  src  target
root@ip-172-31-42-61:/home/ubuntu/Netflix-Pipeline-Project# cd target
root@ip-172-31-42-61:/home/ubuntu/Netflix-Pipeline-Project/target# ls
NETFLIX-1.2.2  classes  maven-archiver  maven-status  surefire-reports  test-classes
root@ip-172-31-42-61:/home/ubuntu/Netflix-Pipeline-Project/target#
```

i-06c497142a314286c (Docker-Test)
PublicIPs: 3.111.186.31 PrivateIPs: 172.31.42.61

Activate Windows
Go to Settings to activate Windows.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Step 11: Create a Docker file using command “vi Docker-file”



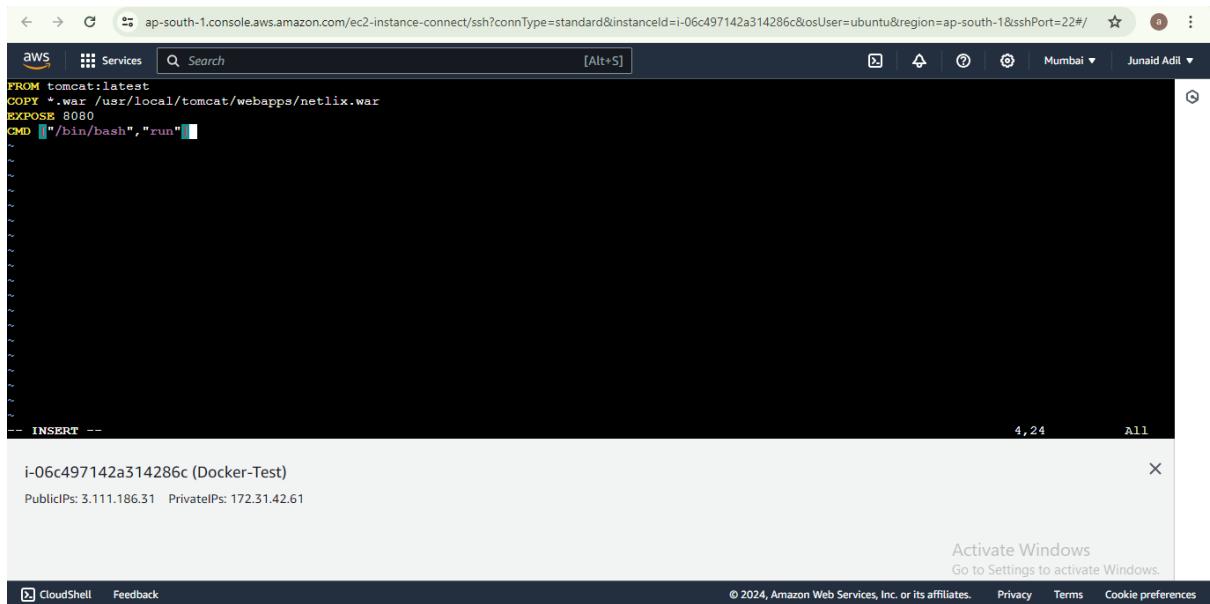
```
root@ip-172-31-42-61:/home/ubuntu/Netflix-Pipeline-Project# vi Docker-file
```

i-06c497142a314286c (Docker-Test)
PublicIPs: 3.111.186.31 PrivateIPs: 172.31.42.61

Activate Windows
Go to Settings to activate Windows.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Add Instructions in the file.



```
FROM tomcat:latest
COPY *.war /usr/local/tomcat/webapps/netflix.war
EXPOSE 8080
CMD ["/bin/bash", "run"]
```

-- INSERT --

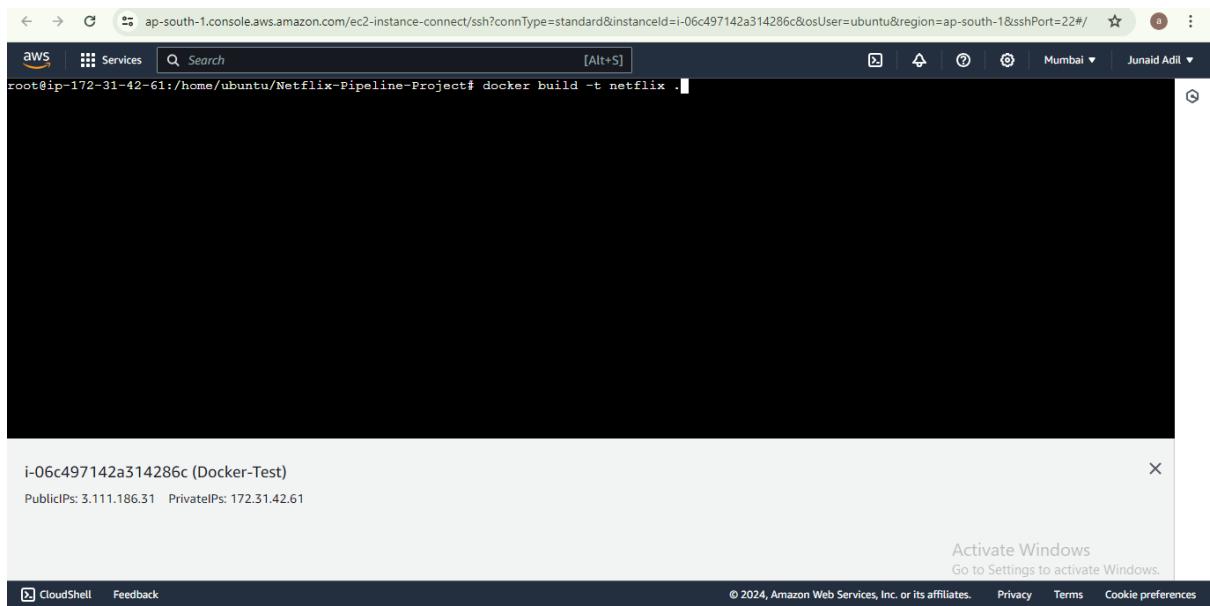
i-06c497142a314286c (Docker-Test)
PublicIPs: 3.111.186.31 PrivateIPs: 172.31.42.61

Activate Windows
Go to Settings to activate Windows.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Step 12: Build the Image from the created Dockerfile.

Use command “**docker build -t netflix .**”



```
root@ip-172-31-42-61:/home/ubuntu/Netflix-Pipeline-Project# docker build -t netflix .
```

i-06c497142a314286c (Docker-Test)
PublicIPs: 3.111.186.31 PrivateIPs: 172.31.42.61

Activate Windows
Go to Settings to activate Windows.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

```
aws Services Search [Alt+S] Mumbai Junaid Adil
root@ip-172-31-42-61:/home/ubuntu/Netflix-Pipeline-Project/target# docker build -t netflix .
DEPRECATED: The legacy builder is deprecated and will be removed in a future release.
Install the buildx component to build images with BuildKit:
https://docs.docker.com/go/buildx/
Sending build context to Docker daemon 18.34MB
Step 1/4 : FROM tomcat:latest
--> cd76d1bc5581
Step 2/4 : COPY *.war /usr/local/tomcat/webapps/netflix.war
--> 1e1d260d70a4
Step 3/4 : EXPOSE 8080
--> Running in 680568eabe53
Removing intermediate container 680568eabe53
--> e3e2deeee875
Step 4/4 : CMD ["/bin/bash"]
--> Running in c99b5fb55fb
Removing intermediate container c99b5fb55fb
--> 85eb9b3ed067
Successfully built 85eb9b3ed067
Successfully tagged netflix:latest
root@ip-172-31-42-61:/home/ubuntu/Netflix-Pipeline-Project/target#
```

i-06c497142a314286c (Docker-Test)
PublicIPs: 3.111.186.31 PrivateIPs: 172.31.42.61

Activate Windows
Go to Settings to activate Windows.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Step 13: Execute command “`docker images`” to check the images created

```
aws Services Search [Alt+S] Mumbai Junaid Adil
DEPRECATED: The legacy builder is deprecated and will be removed in a future release.
Install the buildx component to build images with BuildKit:
https://docs.docker.com/go/buildx/
Sending build context to Docker daemon 18.34MB
Step 1/4 : FROM tomcat:latest
--> cd76d1bc5581
Step 2/4 : COPY *.war /usr/local/tomcat/webapps/netflix.war
--> 1e1d260d70a4
Step 3/4 : EXPOSE 8080
--> Running in 680568eabe53
Removing intermediate container 680568eabe53
--> e3e2deeee875
Step 4/4 : CMD ["/bin/bash"]
--> Running in c99b5fb55fb
Removing intermediate container c99b5fb55fb
--> 85eb9b3ed067
Successfully built 85eb9b3ed067
Successfully tagged netflix:latest
root@ip-172-31-42-61:/home/ubuntu/Netflix-Pipeline-Project/target# docker images
REPOSITORY TAG IMAGE ID CREATED SIZE
netflix latest 85eb9b3ed067 59 seconds ago 464MB
tomcat latest cd76d1bc5581 3 days ago 455MB
root@ip-172-31-42-61:/home/ubuntu/Netflix-Pipeline-Project/target#
```

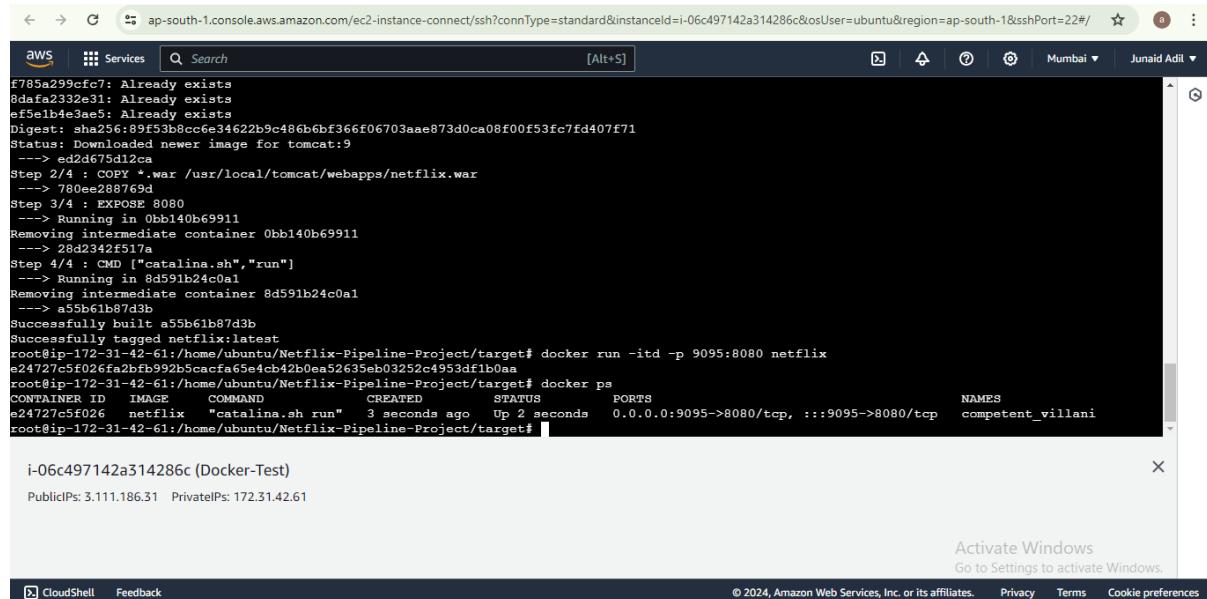
i-06c497142a314286c (Docker-Test)
PublicIPs: 3.111.186.31 PrivateIPs: 172.31.42.61

Activate Windows
Go to Settings to activate Windows.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

We can see the Images with the name Netflix has been created along with the Tomcat Image.

Step 14: For container Port Mapping execute command “**docker run -itd -p 9095:8080 Netflix**” to run the container.

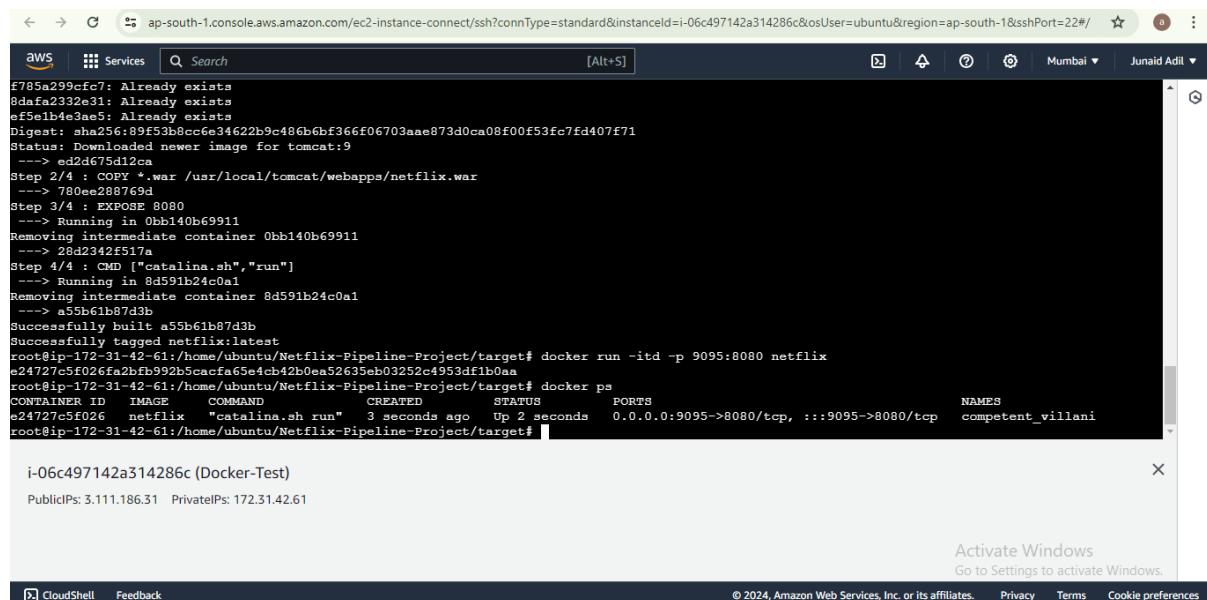


```
f785a299cf7: Already exists
8dfa2332e31: Already exists
ef5e1b4e3ae5: Already exists
Digest: sha256:89f53b8cc6e34622b9c486b6bf366f06703aae873d0ca08f00f53fc7fd407f71
Status: Downloaded newer image for tomcat:9
--> ed2d675d12ca
Step 2/4 : COPY *.war /usr/local/tomcat/webapps/netflix.war
--> 780ee288769d
Step 3/4 : EXPOSE 8080
--> Running in 0bb140b69911
Removing intermediate container 0bb140b69911
--> 28d2342f517a
Step 4/4 : CMD ["catalina.sh","run"]
--> Running in 8d591b24c0a1
Removing intermediate container 8d591b24c0a1
--> a55b61b87d3b
Successfully built a55b61b87d3b
Successfully tagged netflix:latest
root@ip-172-31-42-61:/home/ubuntu/Netflix-Pipeline-Project/target# docker run -itd -p 9095:8080 netflix
e24727c5f026fa2bfb992b5cacfa65e4cb42b0ea52635eb03252c4953df1b0aa
root@ip-172-31-42-61:/home/ubuntu/Netflix-Pipeline-Project/target# docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
e24727c5f026 netflix "catalina.sh run" 3 seconds ago Up 2 seconds 0.0.0.0:9095->8080/tcp, :::9095->8080/tcp competent_villani
root@ip-172-31-42-61:/home/ubuntu/Netflix-Pipeline-Project/target#
```

i-06c497142a314286c (Docker-Test)
PublicIPs: 3.111.186.31 PrivateIPs: 172.31.42.61

Activate Windows
Go to Settings to activate Windows.

Step 15: Execute “**docker ps**” to check the status of container



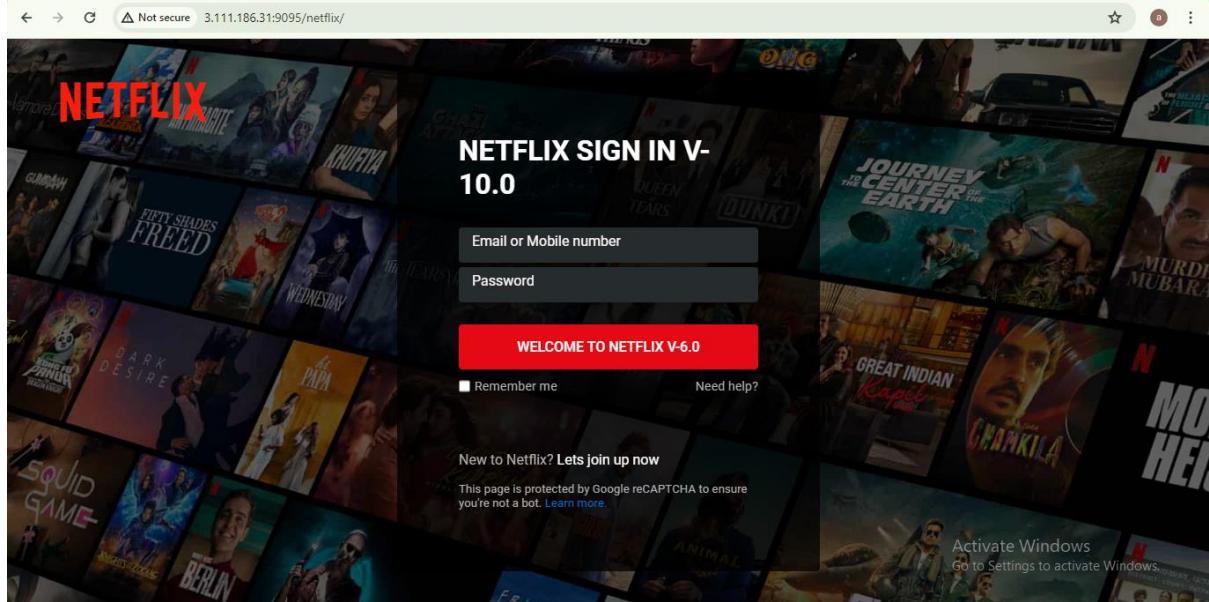
```
f785a299cf7: Already exists
8dfa2332e31: Already exists
ef5e1b4e3ae5: Already exists
Digest: sha256:89f53b8cc6e34622b9c486b6bf366f06703aae873d0ca08f00f53fc7fd407f71
Status: Downloaded newer image for tomcat:9
--> ed2d675d12ca
Step 2/4 : COPY *.war /usr/local/tomcat/webapps/netflix.war
--> 780ee288769d
Step 3/4 : EXPOSE 8080
--> Running in 0bb140b69911
Removing intermediate container 0bb140b69911
--> 28d2342f517a
Step 4/4 : CMD ["catalina.sh","run"]
--> Running in 8d591b24c0a1
Removing intermediate container 8d591b24c0a1
--> a55b61b87d3b
Successfully built a55b61b87d3b
Successfully tagged netflix:latest
root@ip-172-31-42-61:/home/ubuntu/Netflix-Pipeline-Project/target# docker run -itd -p 9095:8080 netflix
e24727c5f026fa2bfb992b5cacfa65e4cb42b0ea52635eb03252c4953df1b0aa
root@ip-172-31-42-61:/home/ubuntu/Netflix-Pipeline-Project/target# docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
e24727c5f026 netflix "catalina.sh run" 3 seconds ago Up 2 seconds 0.0.0.0:9095->8080/tcp, :::9095->8080/tcp competent_villani
root@ip-172-31-42-61:/home/ubuntu/Netflix-Pipeline-Project/target#
```

i-06c497142a314286c (Docker-Test)
PublicIPs: 3.111.186.31 PrivateIPs: 172.31.42.61

Activate Windows
Go to Settings to activate Windows.

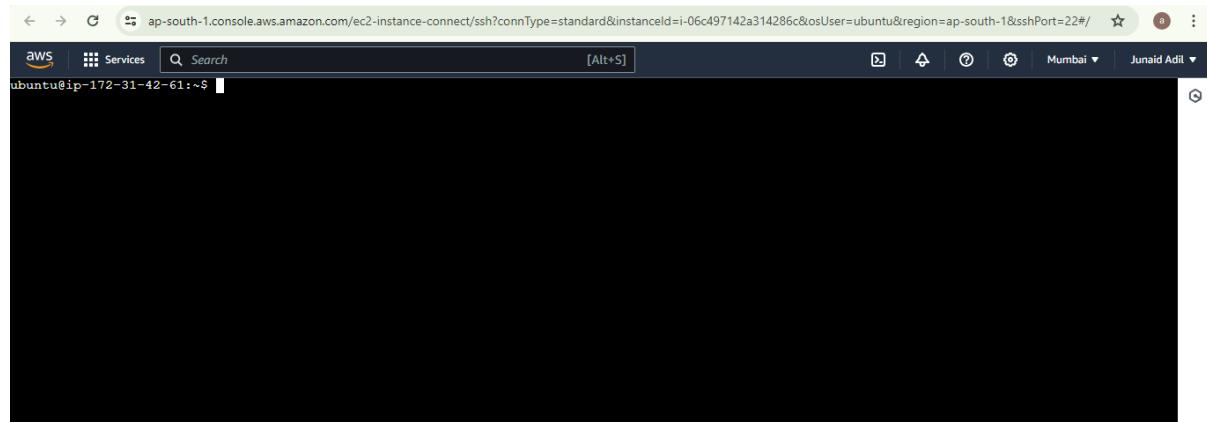
We can see the Container is running.

Step 16: To run the application in a Web Browser, copy public IP of Instance and run in browser along with the port number “**3.111.186.31:9095/Netflix/**”



L3 - Demonstrate Docker Compose using the Application Image and MySQL Image to start and stop all container services

Step 1: Create an Instance, connect and become root user

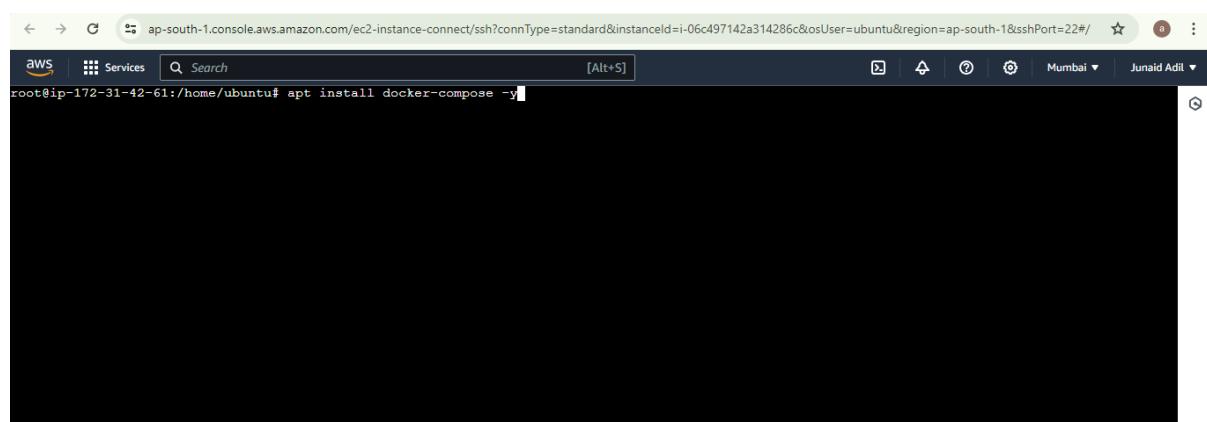


```
i-06c497142a314286c (Docker-Test)
Public IPs: 15.206.68.94 Private IPs: 172.31.42.61
```

Activate Windows
Go to Settings to activate Windows.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Step 2: Install docker compose using command “**apt install docker-compose -y**”



```
root@ip-172-31-42-61:/home/ubuntu# apt install docker-compose -y
```

```
i-06c497142a314286c (Docker-Test)
Public IPs: 15.206.68.94 Private IPs: 172.31.42.61
```

Activate Windows
Go to Settings to activate Windows.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

```
Processing triggers for man-db (2.10.2-1) ...
Scanning processes...
Scanning candidates...
Scanning linux images...

Restarting services...
systemctl restart acpid.service chrony.service containerd.service cron.service multipathd.service packagekit.service polkit.service rsyslog.service serial-getty@tty0.service snapd.service ssh.service systemd-journald.service systemd-networkd.service systemd-resolved.service systemd-unitservice
Service restarts being deferred:
/etc/needrestart/restart.d/dbus.service
systemctl restart docker.service
systemctl restart getty@tty1.service
systemctl restart networkd-dispatcher.service
systemctl restart systemd-logind.service
systemctl restart unattended-upgrades.service
systemctl restart user@1000.service

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-172-31-42-61:/home/ubuntu#
```

i-06c497142a314286c (Docker-Test)
PublicIPs: 15.206.68.94 PrivateIPs: 172.31.42.61

Activate Windows
Go to Settings to activate Windows.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Step 3: Check docker-compose version using command “**docker-compose --version**”

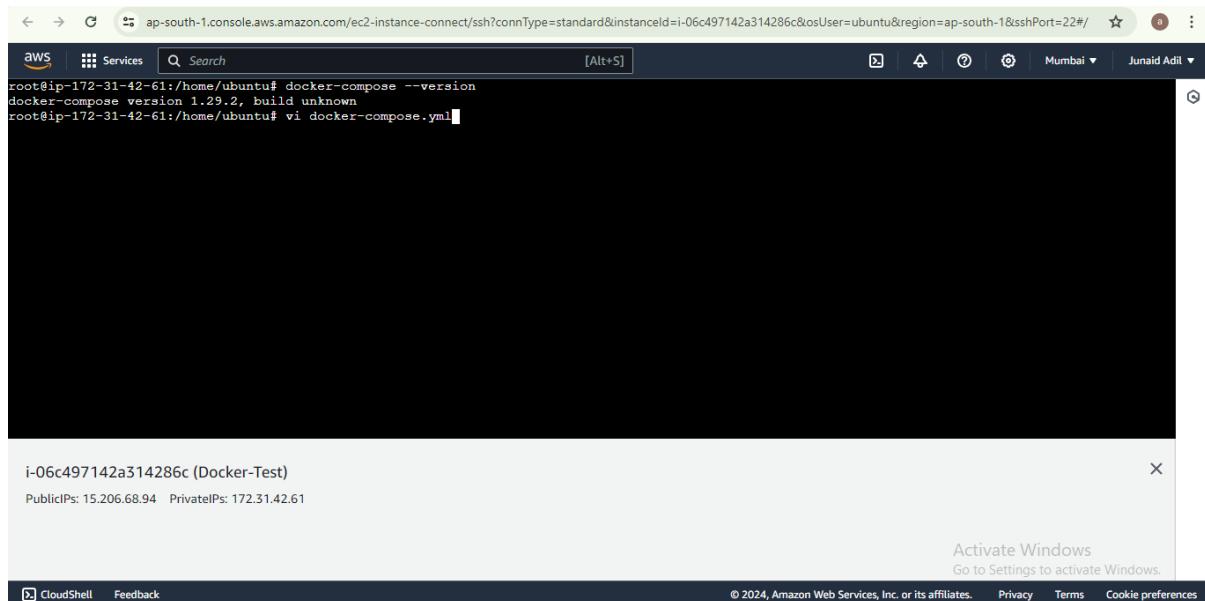
```
root@ip-172-31-42-61:/home/ubuntu# docker-compose --version
docker-compose version 1.29.2, build unknown
root@ip-172-31-42-61:/home/ubuntu#
```

i-06c497142a314286c (Docker-Test)
PublicIPs: 15.206.68.94 PrivateIPs: 172.31.42.61

Activate Windows
Go to Settings to activate Windows.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Step 4: Create a file “docker-compose.yml”



A screenshot of the AWS CloudShell interface. The terminal window shows the command `docker-compose --version` being run, which outputs "docker-compose version 1.29.2, build unknown". Below this, the command `vi docker-compose.yml` is run to open the configuration file. The status bar at the bottom indicates the instance ID (i-06c497142a314286c) and IP addresses (PublicIPs: 15.206.68.94, PrivateIPs: 172.31.42.61).

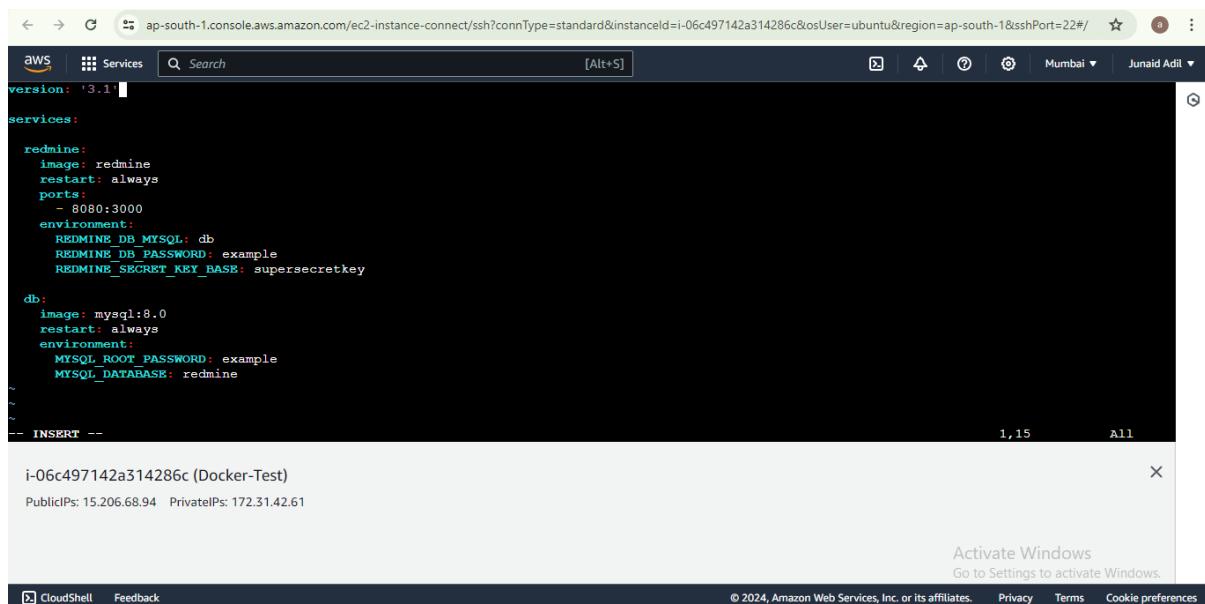
```
root@ip-172-31-42-61:/home/ubuntu# docker-compose --version
docker-compose version 1.29.2, build unknown
root@ip-172-31-42-61:/home/ubuntu# vi docker-compose.yml
```

i-06c497142a314286c (Docker-Test)
PublicIPs: 15.206.68.94 PrivateIPs: 172.31.42.61

Activate Windows
Go to Settings to activate Windows.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

In docker-compose.yml file we define the services, networks, volumes, and configurations for the multi-container Docker applications.



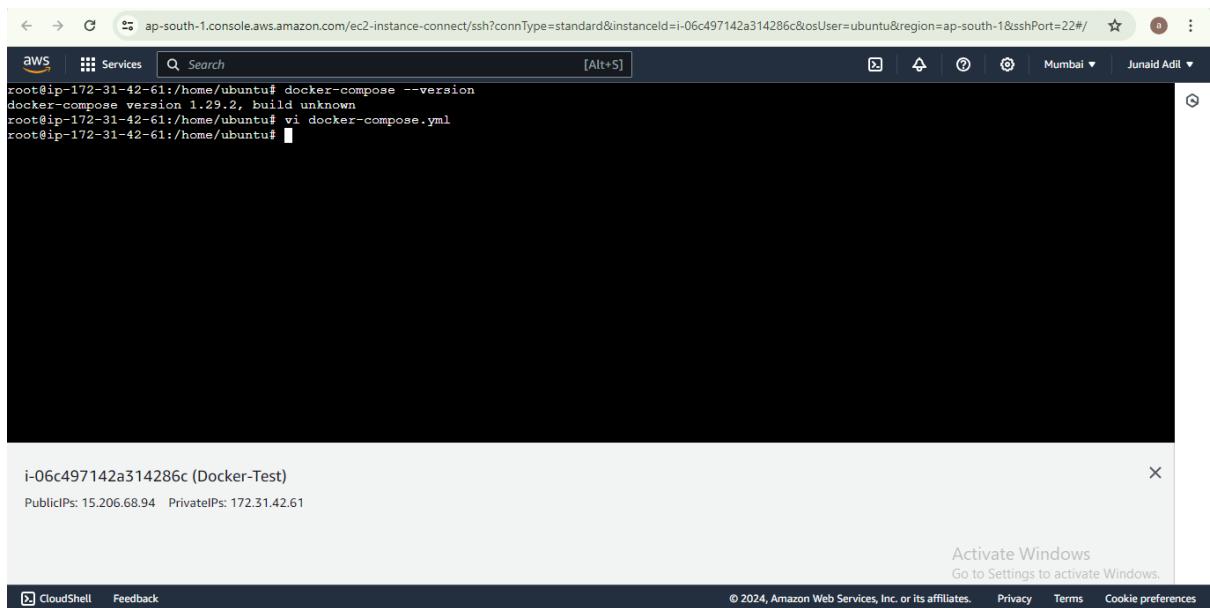
A screenshot of the AWS CloudShell interface showing the contents of a `docker-compose.yml` file. The file defines two services: `redmine` and `db`. The `redmine` service uses the `redmine` image, restarts always, and maps port 8080 to 3000. It has environment variables `REDMINE_DB_MYSQL`, `REDMINE_DB_PASSWORD`, and `REDMINE_SECRET_KEY_BASE` set to `db`, `example`, and `supersecretkey` respectively. The `db` service uses the `mysql:8.0` image, restarts always, and has environment variables `MYSQL_ROOT_PASSWORD` and `MYSQL_DATABASE` set to `example` and `redmine` respectively. The status bar at the bottom indicates the instance ID (i-06c497142a314286c) and IP addresses (PublicIPs: 15.206.68.94, PrivateIPs: 172.31.42.61).

```
version: '3.1'
services:
  redmine:
    image: redmine
    restart: always
    ports:
      - 8080:3000
    environment:
      REDMINE_DB_MYSQL: db
      REDMINE_DB_PASSWORD: example
      REDMINE_SECRET_KEY_BASE: supersecretkey
  db:
    image: mysql:8.0
    restart: always
    environment:
      MYSQL_ROOT_PASSWORD: example
      MYSQL_DATABASE: redmine
  ~
  ~
  ~
-- INSERT --
```

i-06c497142a314286c (Docker-Test)
PublicIPs: 15.206.68.94 PrivateIPs: 172.31.42.61

Activate Windows
Go to Settings to activate Windows.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences



A screenshot of the AWS CloudShell interface. The terminal window shows the command `docker-compose --version` being run, which outputs:

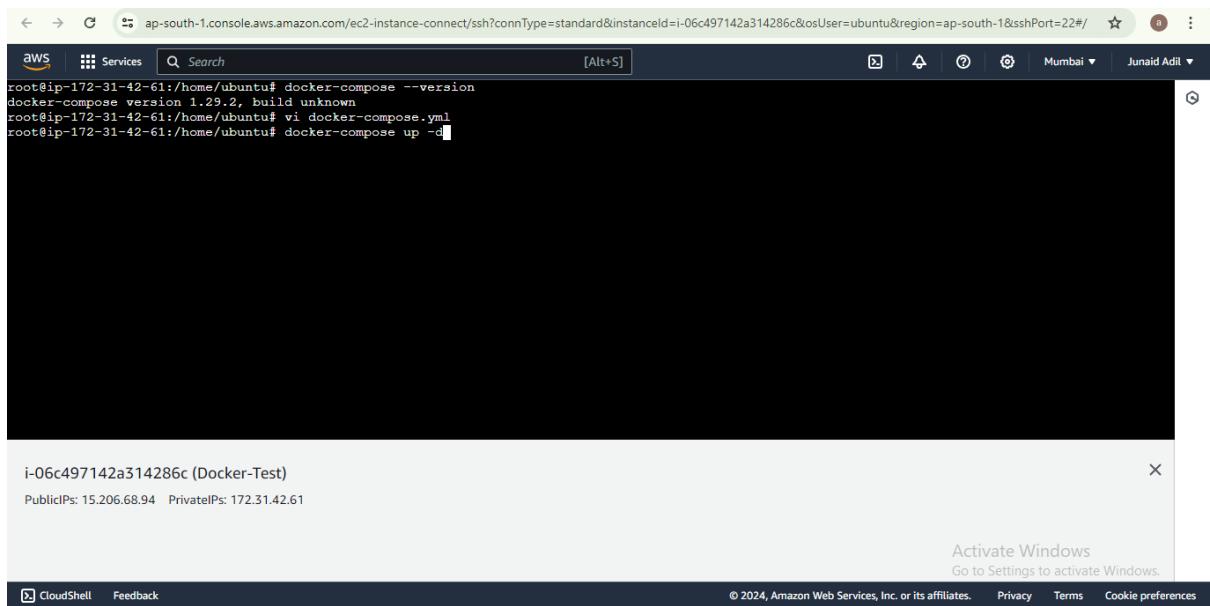
```
root@ip-172-31-42-61:/home/ubuntu# docker-compose --version
docker-compose version 1.29.2, build unknown
root@ip-172-31-42-61:/home/ubuntu# vi docker-compose.yml
root@ip-172-31-42-61:/home/ubuntu#
```

The AWS navigation bar at the top includes "aws", "Services", "Search", "[Alt+S]", "Mumbai", and "Junaid Adil". Below the terminal, the instance details are shown:

i-06c497142a314286c (Docker-Test)
PublicIPs: 15.206.68.94 PrivateIPs: 172.31.42.61

At the bottom right, there is an "Activate Windows" message with a link to "Go to Settings to activate Windows". The footer contains links for "CloudShell", "Feedback", "© 2024, Amazon Web Services, Inc. or its affiliates.", "Privacy", "Terms", and "Cookie preferences".

Step 5: Start the container using command “**docker-compose up -d**”



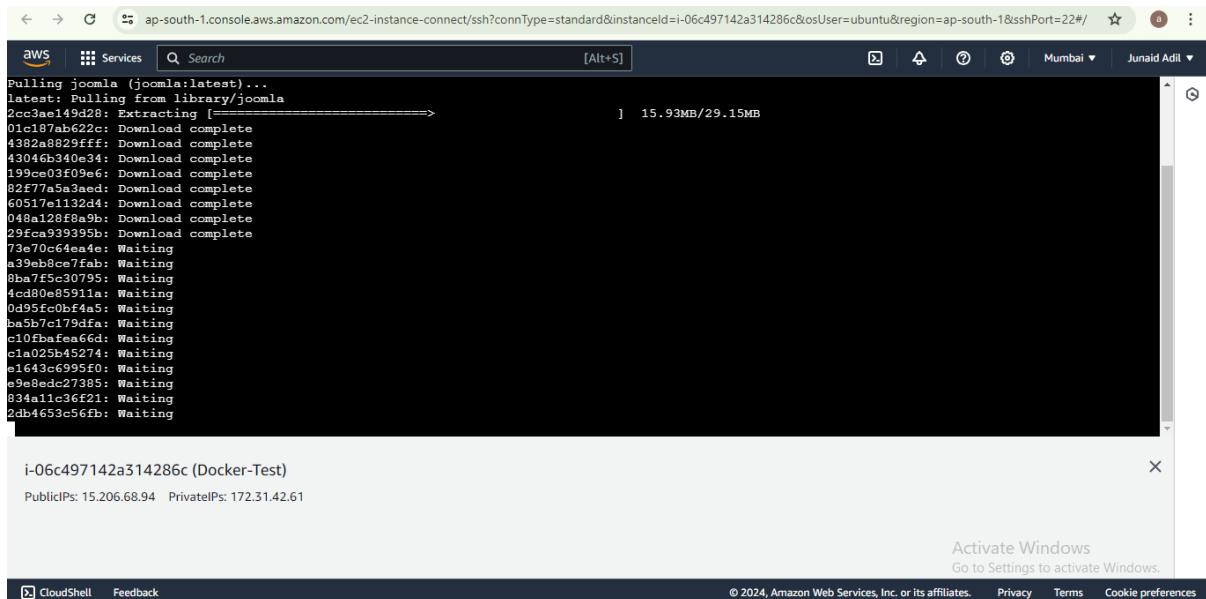
A screenshot of the AWS CloudShell interface. The terminal window shows the command `docker-compose up -d` being run.

```
root@ip-172-31-42-61:/home/ubuntu# docker-compose --version
docker-compose version 1.29.2, build unknown
root@ip-172-31-42-61:/home/ubuntu# vi docker-compose.yml
root@ip-172-31-42-61:/home/ubuntu# docker-compose up -d
```

The AWS navigation bar at the top includes "aws", "Services", "Search", "[Alt+S]", "Mumbai", and "Junaid Adil". Below the terminal, the instance details are shown:

i-06c497142a314286c (Docker-Test)
PublicIPs: 15.206.68.94 PrivateIPs: 172.31.42.61

At the bottom right, there is an "Activate Windows" message with a link to "Go to Settings to activate Windows". The footer contains links for "CloudShell", "Feedback", "© 2024, Amazon Web Services, Inc. or its affiliates.", "Privacy", "Terms", and "Cookie preferences".

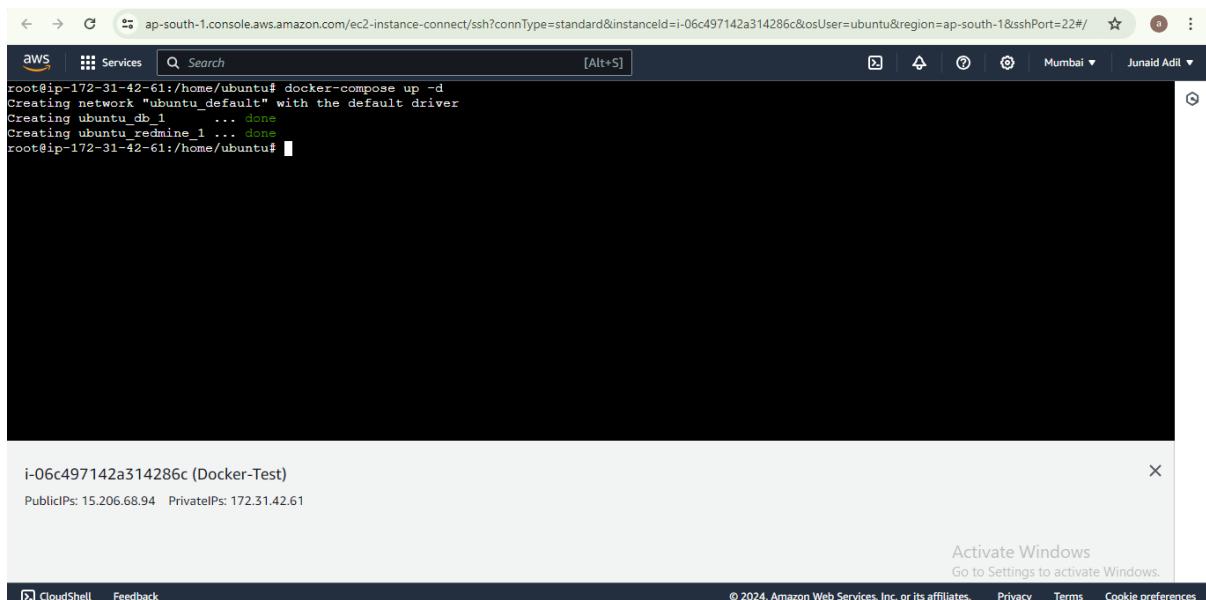


Pulling joomla (joomla:latest)...
latest: Pulling from library/joomla
2cc3ae149d28: Extracting [=====] 15.93MB/29.15MB
01c187ab622e: Download complete
4382a8829fff: Download complete
43046b340e34: Download complete
199ce03f09e6: Download complete
82f7a5a3aed: Download complete
60517e1132d4: Download complete
048a128f8a9b: Download complete
29fca939395b: Download complete
73e70c64ea4e: Waiting
a39eb3cce7fab: Waiting
9ba7f5c30795: Waiting
4cd80e85911a: Waiting
0d95fc0bf4a5: Waiting
ba5b7c179dfa: Waiting
c10fbfaea66d: Waiting
cla025b45274: Waiting
e1643c6995f0: Waiting
e9ebcd27385: Waiting
834a11c36f21: Waiting
2db4653c56fb: Waiting

i-06c497142a314286c (Docker-Test)
PublicIPs: 15.206.68.94 PrivateIPs: 172.31.42.61

Activate Windows
Go to Settings to activate Windows.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences



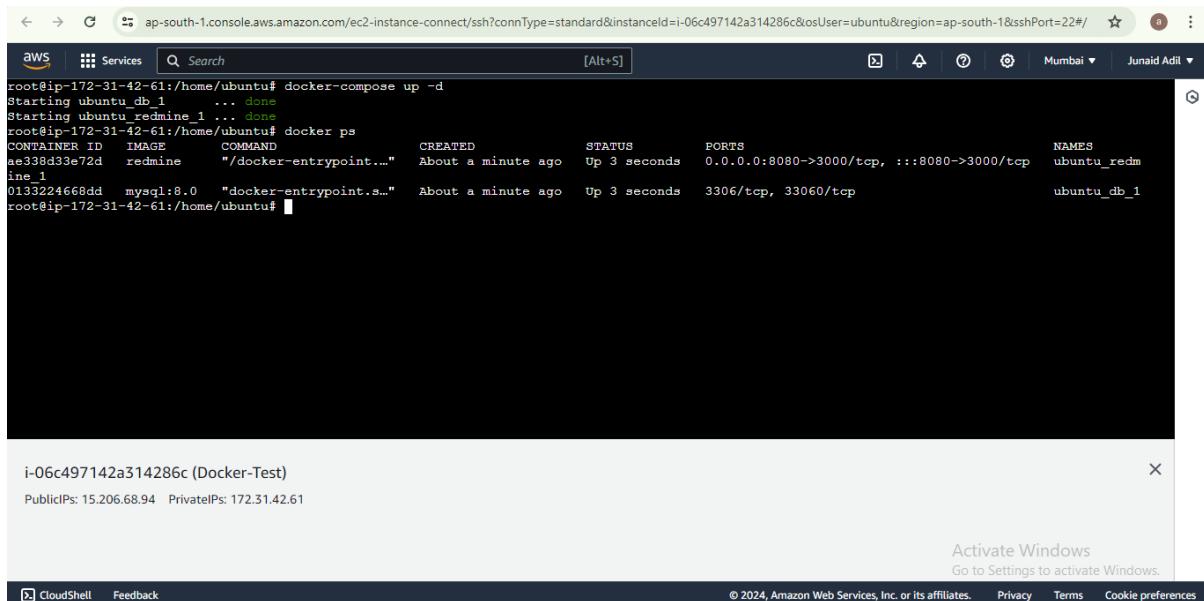
```
root@ip-172-31-42-61:/home/ubuntu# docker-compose up -d
Creating network "ubuntu_default" with the default driver
Creating ubuntu_db_1 ... done
Creating ubuntu_redmine_1 ... done
root@ip-172-31-42-61:/home/ubuntu#
```

i-06c497142a314286c (Docker-Test)
PublicIPs: 15.206.68.94 PrivateIPs: 172.31.42.61

Activate Windows
Go to Settings to activate Windows.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Step 6: Run “docker ps” to check the status of container



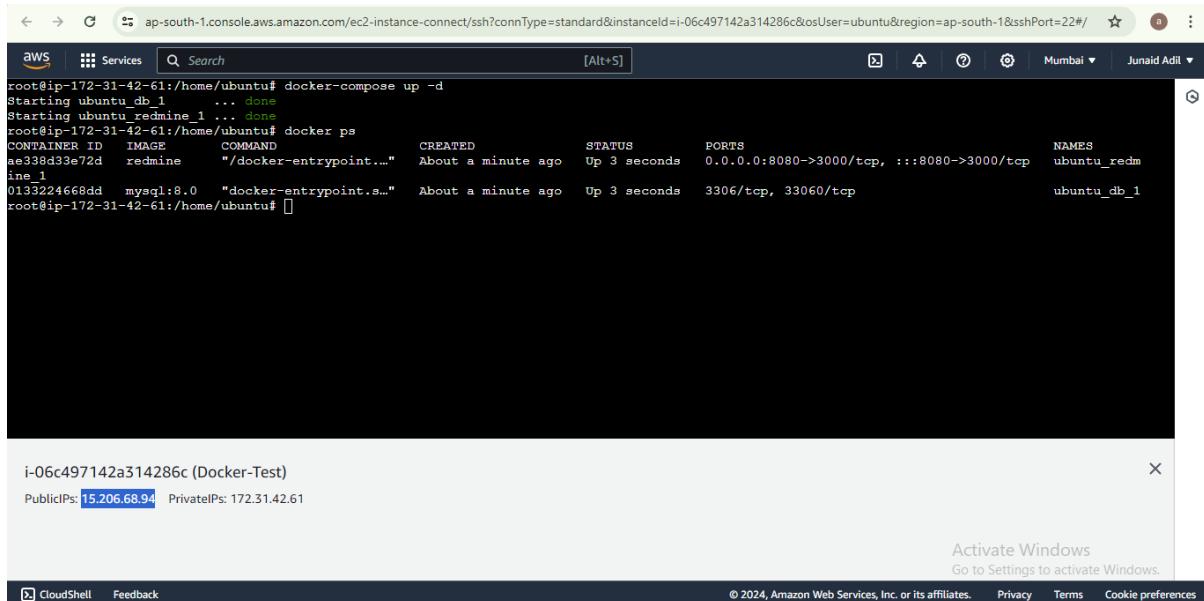
```
root@ip-172-31-42-61:/home/ubuntu# docker-compose up -d
Starting ubuntu_db_1 ... done
Starting ubuntu_redmine_1 ... done
root@ip-172-31-42-61:/home/ubuntu# docker ps
CONTAINER ID   IMAGE          COMMAND       CREATED      STATUS      PORTS          NAMES
ae338d33e72d   redmine        "/docker-entrypoint..."   About a minute ago   Up 3 seconds   0.0.0.0:8080->3000/tcp, :::8080->3000/tcp   ubuntu_redm
ine_1          mysql:8.0     "docker-entrypoint.s..."   About a minute ago   Up 3 seconds   3306/tcp, 33060/tcp   ubuntu_db_1
root@ip-172-31-42-61:/home/ubuntu#
```

i-06c497142a314286c (Docker-Test)
PublicIPs: 15.206.68.94 PrivateIPs: 172.31.42.61

Activate Windows
Go to Settings to activate Windows.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Step 7: Go to web-browser and run https://localhost:8080, copy Public IP of instance along with the port number:8080. Command “**15.206.68.94:8080**”



```
root@ip-172-31-42-61:/home/ubuntu# docker-compose up -d
Starting ubuntu_db_1 ... done
Starting ubuntu_redmine_1 ... done
root@ip-172-31-42-61:/home/ubuntu# docker ps
CONTAINER ID   IMAGE          COMMAND       CREATED      STATUS      PORTS          NAMES
ae338d33e72d   redmine        "/docker-entrypoint..."   About a minute ago   Up 3 seconds   0.0.0.0:8080->3000/tcp, :::8080->3000/tcp   ubuntu_redm
ine_1          mysql:8.0     "docker-entrypoint.s..."   About a minute ago   Up 3 seconds   3306/tcp, 33060/tcp   ubuntu_db_1
root@ip-172-31-42-61:/home/ubuntu#
```

i-06c497142a314286c (Docker-Test)
PublicIPs: **15.206.68.94** PrivateIPs: 172.31.42.61

Activate Windows
Go to Settings to activate Windows.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Redmine

Home Projects Help

Search: Jump to a project...

Sign in Register

Projects Activity

Projects

Filters Status is active Add filter

Apply Clear

No data to display

Also available in: Atom

Activate Windows
Go to Settings to activate Windows.

Powered by Redmine © 2006-2024 Jean-Philippe Lang

Step 8: Execute command “**docker-compose down**” to stop all running services defined in docker-copmose.yml file or Container.

```
aws Services Q Search [Alt+S] Mumbai Junaid Adil
```

```
root@ip-172-31-42-61:/home/ubuntu# docker-compose up -d
Starting ubuntu_db_1 ... done
Starting ubuntu_redmine_1 ... done
root@ip-172-31-42-61:/home/ubuntu# docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
ae38d33e72d redmine "/dock...r-entrypoint..." About a minute ago Up 3 seconds 0.0.0.0:8080->3000/tcp, :::8080->3000/tcp ubuntu_redmine_1
0133224668dd mysql:8.0 "dock...r-entrypoint.s..." About a minute ago Up 3 seconds 3306/tcp, 33060/tcp ubuntu_db_1
root@ip-172-31-42-61:/home/ubuntu# docker-compose stop
```

i-06c497142a314286c (Docker-Test)

PublicIPs: 15.206.68.94 PrivateIPs: 172.31.42.61

Activate Windows
Go to Settings to activate Windows.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

The screenshot shows a terminal session in AWS CloudShell. The user has run several commands to manage Docker containers:

```
root@ip-172-31-42-61:/home/ubuntu# docker-compose up -d
Starting ubuntu_db_1 ... done
Starting ubuntu_redmine_1 ... done
root@ip-172-31-42-61:/home/ubuntu# docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
ae338d33e72d redmine "/docker-entrypoint..." About a minute ago Up 3 seconds 0.0.0.0:8080->3000/tcp, :::8080->3000/tcp ubuntu_redmine_1
0133224668dd mysql:8.0 "docker-entrypoint.s..." About a minute ago Up 3 seconds 3306/tcp, 33060/tcp ubuntu_db_1
root@ip-172-31-42-61:/home/ubuntu# docker-compose stop
Stopping ubuntu_redmine_1 ... done
Stopping ubuntu_db_1 ... done
root@ip-172-31-42-61:/home/ubuntu#
```

Below the terminal, the session summary shows:

i-06c497142a314286c (Docker-Test)
PublicIPs: 15.206.68.94 PrivateIPs: 172.31.42.61

At the bottom, there are links for CloudShell and Feedback, and copyright information for Amazon Web Services.

We can see the container has been stopped

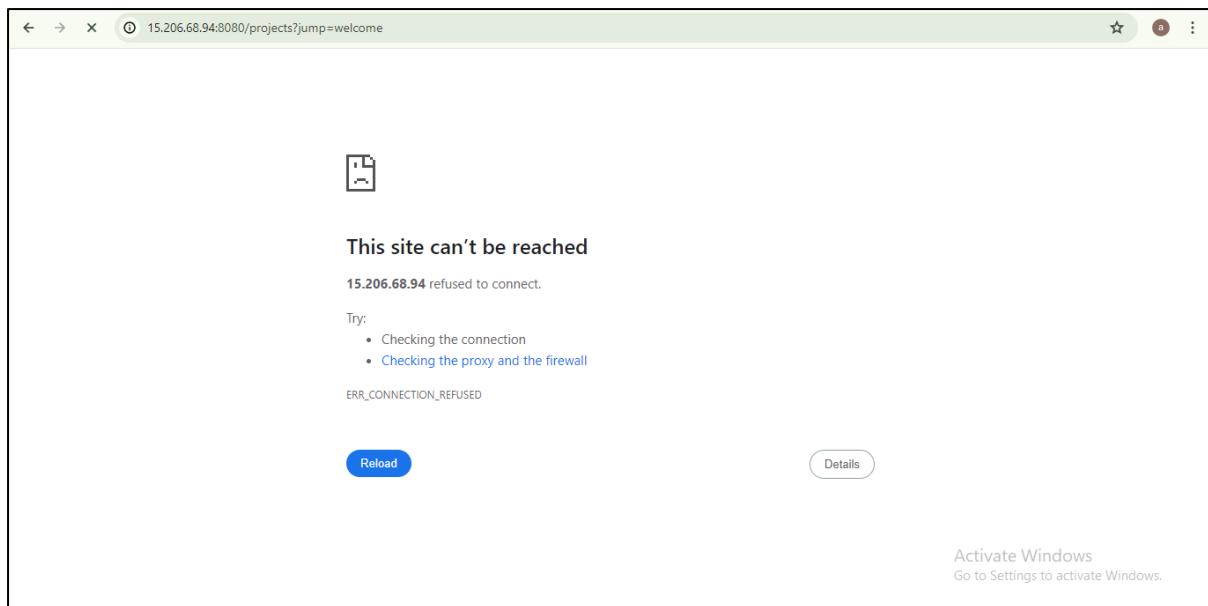
The screenshot shows a terminal session in AWS CloudShell. The user has run several commands to manage Docker containers:

```
root@ip-172-31-42-61:/home/ubuntu# docker-compose up -d
Starting ubuntu_db_1 ... done
Starting ubuntu_redmine_1 ... done
root@ip-172-31-42-61:/home/ubuntu# docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
ae338d33e72d redmine "/docker-entrypoint..." About a minute ago Up 3 seconds 0.0.0.0:8080->3000/tcp, :::8080->3000/tcp ubuntu_redmine_1
0133224668dd mysql:8.0 "docker-entrypoint.s..." About a minute ago Up 3 seconds 3306/tcp, 33060/tcp ubuntu_db_1
root@ip-172-31-42-61:/home/ubuntu# docker-compose stop
Stopping ubuntu_redmine_1 ... done
Stopping ubuntu_db_1 ... done
root@ip-172-31-42-61:/home/ubuntu# docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
root@ip-172-31-42-61:/home/ubuntu#
```

Below the terminal, the session summary shows:

i-06c497142a314286c (Docker-Test)
PublicIPs: 15.206.68.94 PrivateIPs: 172.31.42.61

At the bottom, there are links for CloudShell and Feedback, and copyright information for Amazon Web Services.

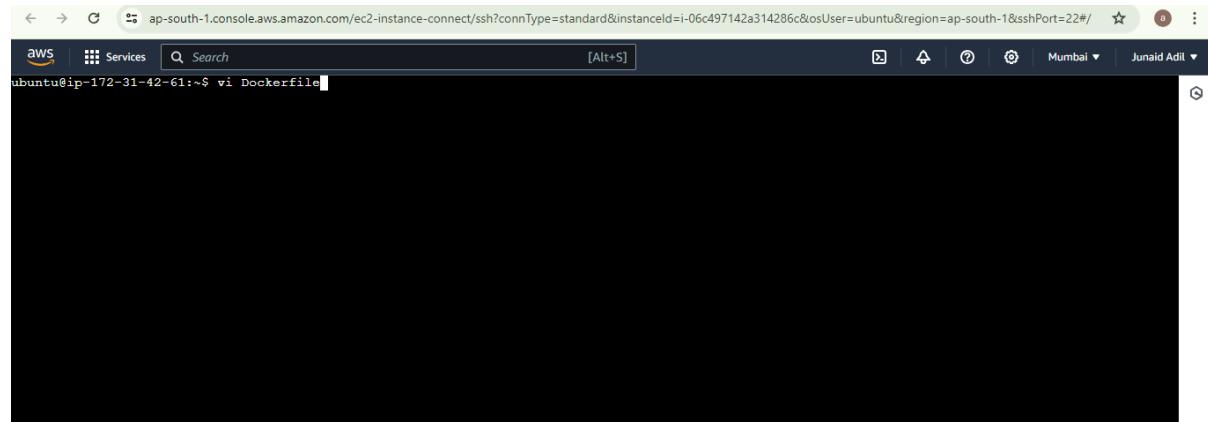


Site can't be Reached.

Additional:

Create Docker container image based on an existing image and push the Image to Docker Hub Registry:

Step 1: Connect to an Instance and create a Dockerfile



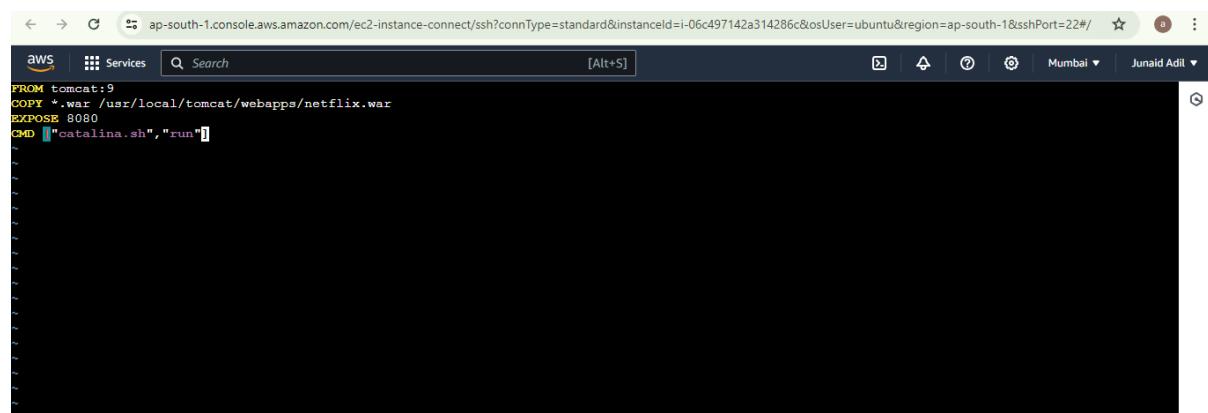
```
ubuntu@ip-172-31-42-61:~$ vi Dockerfile
```

i-06c497142a314286c (Docker-Test)
PublicIPs: 15.206.68.94 PrivateIPs: 172.31.42.61

Activate Windows
Go to Settings to activate Windows.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Enter the Instructions based upon the Base Image in Dockerfile. Where the Artifacts already built.



```
FROM tomcat:9
COPY *.war /usr/local/tomcat/webapps/netflix.war
EXPOSE 8080
CMD ["catalina.sh","run"]
```

"Dockerfile" 4L, 101B

4, 25 All

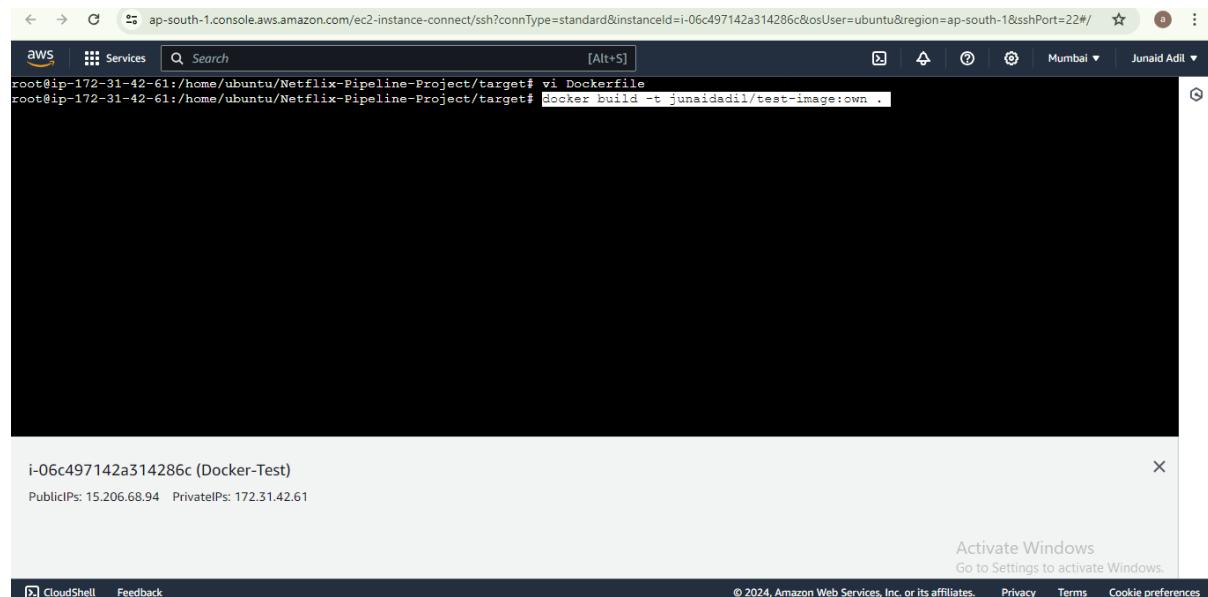
i-06c497142a314286c (Docker-Test)
PublicIPs: 15.206.68.94 PrivateIPs: 172.31.42.61

Activate Windows
Go to Settings to activate Windows.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Step 2: Using this command “**docker build -t yourusername/your-image-name:tag .**” we can Build the Docker Image.

Execute command “**docker build -t junaidadil/test-image:own .**” to Build the Docker Image.

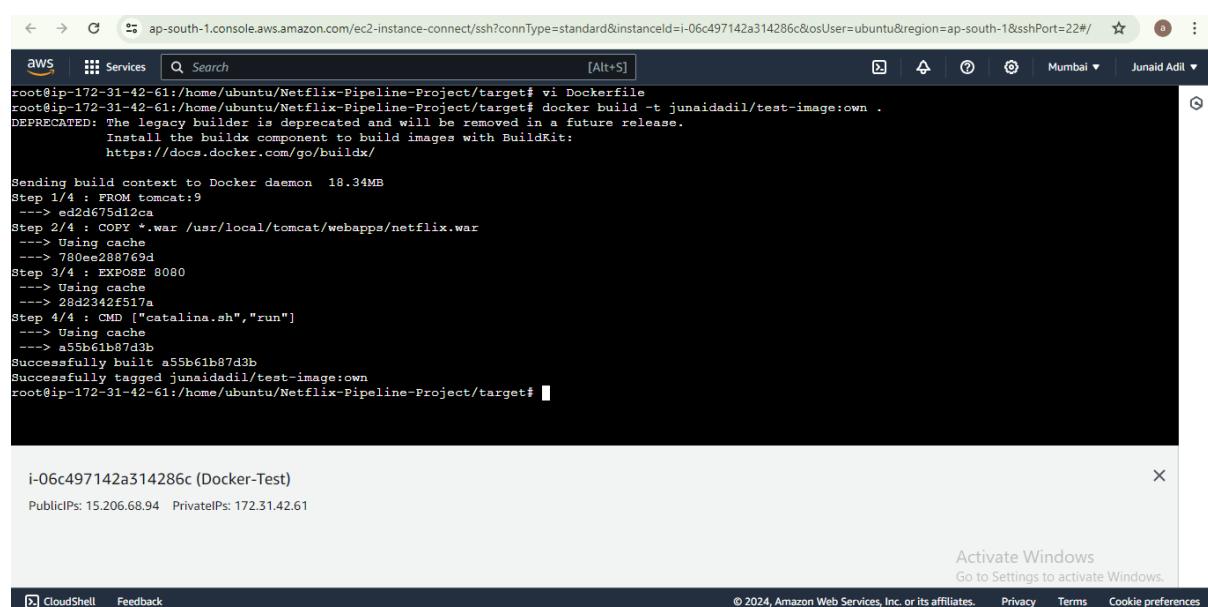


```
root@ip-172-31-42-61:/home/ubuntu/Netflix-Pipeline-Project/target# vi Dockerfile
root@ip-172-31-42-61:/home/ubuntu/Netflix-Pipeline-Project/target# docker build -t junaidadil/test-image:own .
```

i-06c497142a314286c (Docker-Test)
PublicIPs: 15.206.68.94 PrivateIPs: 172.31.42.61

Activate Windows
Go to Settings to activate Windows.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences



```
root@ip-172-31-42-61:/home/ubuntu/Netflix-Pipeline-Project/target# vi Dockerfile
root@ip-172-31-42-61:/home/ubuntu/Netflix-Pipeline-Project/target# docker build -t junaidadil/test-image:own .
DEPRECATED: The legacy builder is deprecated and will be removed in a future release.
Install the buildx component to build images with BuildKit.
https://docs.docker.com/go/buildx/
```

```
Sending build context to Docker daemon 18.34MB
Step 1/4 : FROM tomcat:9
--> ed2d675d12ca
Step 2/4 : COPY *.war /usr/local/tomcat/webapps/netflix.war
--> Using cache
--> 780ee288769d
Step 3/4 : EXPOSE 8080
--> Using cache
--> 28d2342f517a
Step 4/4 : CMD ["catalina.sh", "run"]
--> Using cache
--> a55b61b87d3b
Successfully built a55b61b87d3b
Successfully tagged junaidadil/test-image:own
root@ip-172-31-42-61:/home/ubuntu/Netflix-Pipeline-Project/target#
```

i-06c497142a314286c (Docker-Test)
PublicIPs: 15.206.68.94 PrivateIPs: 172.31.42.61

Activate Windows
Go to Settings to activate Windows.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Step 3: Login to Docker Hub Account using command “docker login ”

The screenshot shows a terminal session in AWS CloudShell. The user has run the command `vi Dockerfile` to view the Dockerfile content:

```
root@ip-172-31-42-61:/home/ubuntu/Netflix-Pipeline-Project/target# vi Dockerfile
```

Then, the user runs `docker build -t junaidadil/test-image:own .`. A warning message is displayed:

```
DEPRECATED: The legacy builder is deprecated and will be removed in a future release.  
Install the buildx component to build images with BuildKit:  
https://docs.docker.com/go/buildx/
```

The build process starts, sending context to the Docker daemon (18.34MB). The steps are as follows:

- Step 1/4 : FROM tomcat:9
- > ed2d675d12ca
- Step 2/4 : COPY *.war /usr/local/tomcat/webapps/netflix.war
- > Using cache
- > 780ee288769d
- Step 3/4 : EXPOSE 8080
- > Using cache
- > 20d2342f517a
- Step 4/4 : CMD ["catalina.sh","run"]
- > Using cache
- > a55b61b87d3b

The image is successfully built and tagged:

```
Successfully built a55b61b87d3b  
Successfully tagged junaidadil/test-image:own
```

Finally, the user runs `docker login`:

```
root@ip-172-31-42-61:/home/ubuntu/Netflix-Pipeline-Project/target# docker login
```

A modal window displays the Docker ID and IP information:

i-06c497142a314286c (Docker-Test)
Public IPs: 15.206.68.94 Private IPs: 172.31.42.61

Activate Windows
Go to Settings to activate Windows.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Enter username

The screenshot shows a terminal session in AWS CloudShell. The user has run the command `docker login`:

```
root@ip-172-31-42-61:/home/ubuntu/Netflix-Pipeline-Project/target# docker login
```

A message from Docker Hub is displayed:

```
Install the buildx component to build images with BuildKit:  
https://docs.docker.com/go/buildx/
```

The build process for the Docker image continues, showing the same steps as before:

- Step 1/4 : FROM tomcat:9
- > ed2d675d12ca
- Step 2/4 : COPY *.war /usr/local/tomcat/webapps/netflix.war
- > Using cache
- > 780ee288769d
- Step 3/4 : EXPOSE 8080
- > Using cache
- > 20d2342f517a
- Step 4/4 : CMD ["catalina.sh","run"]
- > Using cache
- > a55b61b87d3b

The image is successfully built and tagged:

```
Successfully built a55b61b87d3b  
Successfully tagged junaidadil/test-image:own
```

Log in with your Docker ID or email address to push and pull images from Docker Hub. If you don't have a Docker ID, head over to <https://hub.docker.com/> to create one. You can log in with your password or a Personal Access Token (PAT). Using a limited-scope PAT grants better security and is required for organizations using SSO. Learn more at <https://docs.docker.com/go/access-tokens/>

Username: junaidadil

i-06c497142a314286c (Docker-Test)
Public IPs: 15.206.68.94 Private IPs: 172.31.42.61

Activate Windows
Go to Settings to activate Windows.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Enter password

Sending build context to Docker daemon 18.34MB
Step 1/4 : FROM tomcat:9
--> ed2d675d12ca
Step 2/4 : COPY *.war /usr/local/tomcat/webapps/netflix.war
--> Using cache
--> 780ee288769d
Step 3/4 : EXPOSE 8080
--> Using cache
--> 28d2342ff517a
Step 4/4 : CMD ["catalina.sh","run"]
--> Using cache
--> a55b61b87d3b
Successfully built a55b61b87d3b
Successfully tagged junaidadil/test-image:own
root@ip-172-31-42-61:/home/ubuntu/Netflix-Pipeline-Project/target# docker login
Log in with your Docker ID or email address to push and pull images from Docker Hub. If you don't have a Docker ID, head over to https://hub.docker.com/ to create one.
You can log in with your password or a Personal Access Token (PAT). Using a limited-scope PAT grants better security and is required for organizations using SSO. Learn more at https://docs.docker.com/go/access-tokens/
Username: junaidadil
Password: [REDACTED]

i-06c497142a314286c (Docker-Test)
PublicIPs: 15.206.68.94 PrivateIPs: 172.31.42.61

Activate Windows
Go to Settings to activate Windows.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Logged in successfully

--> Using cache
--> 780ee288769d
Step 3/4 : EXPOSE 8080
--> Using cache
--> 28d2342ff517a
Step 4/4 : CMD ["catalina.sh","run"]
--> Using cache
--> a55b61b87d3b
Successfully built a55b61b87d3b
Successfully tagged junaidadil/test-image:own
root@ip-172-31-42-61:/home/ubuntu/Netflix-Pipeline-Project/target# docker login
Log in with your Docker ID or email address to push and pull images from Docker Hub. If you don't have a Docker ID, head over to https://hub.docker.com/ to create one.
You can log in with your password or a Personal Access Token (PAT). Using a limited-scope PAT grants better security and is required for organizations using SSO. Learn more at https://docs.docker.com/go/access-tokens/
Username: junaidadil
Password:
WARNING! Your password will be stored unencrypted in /root/.docker/config.json.
Configure a credential helper to remove this warning. See
<https://docs.docker.com/engine/reference/commandline/login/#credentials-store>
Login Succeeded
root@ip-172-31-42-61:/home/ubuntu/Netflix-Pipeline-Project/target# [REDACTED]

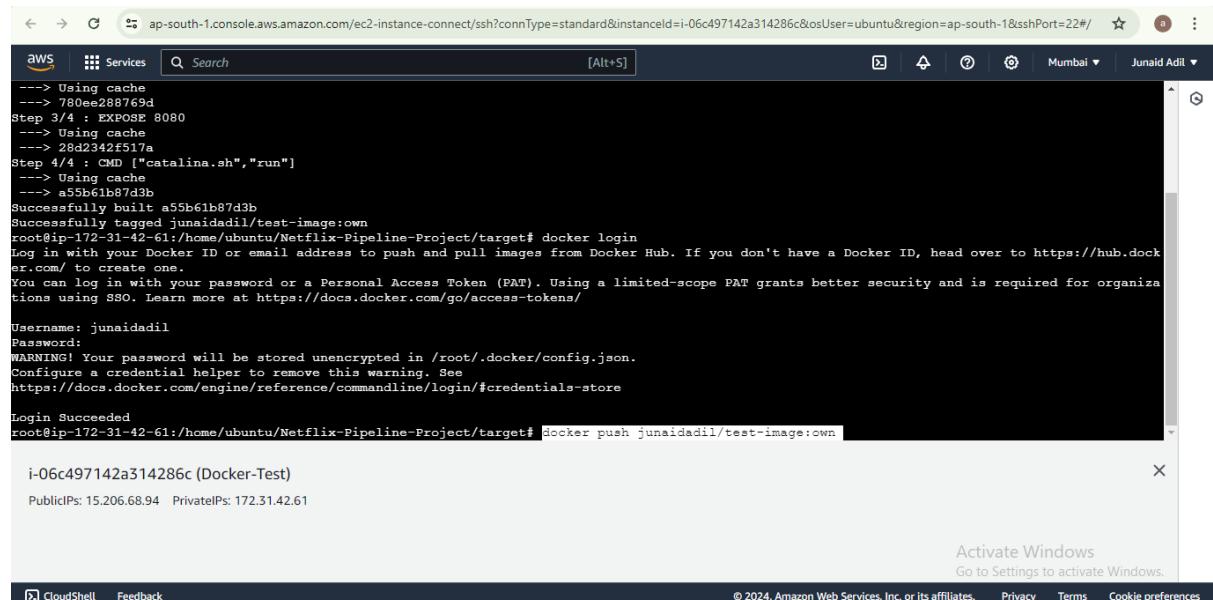
i-06c497142a314286c (Docker-Test)
PublicIPs: 15.206.68.94 PrivateIPs: 172.31.42.61

Activate Windows
Go to Settings to activate Windows.

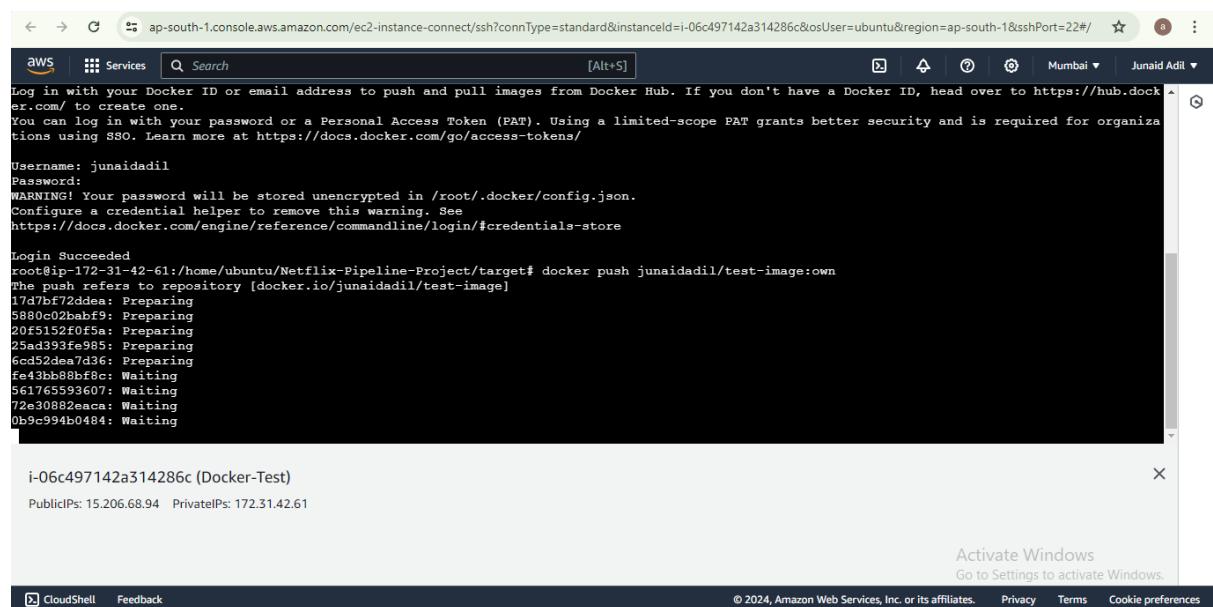
CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Step 4: Run command “**docker push yourusername/your-image-name:tag**” to push the created image into docker registry.

Execute command “ **docker push junaidadil/test-image:own** ”

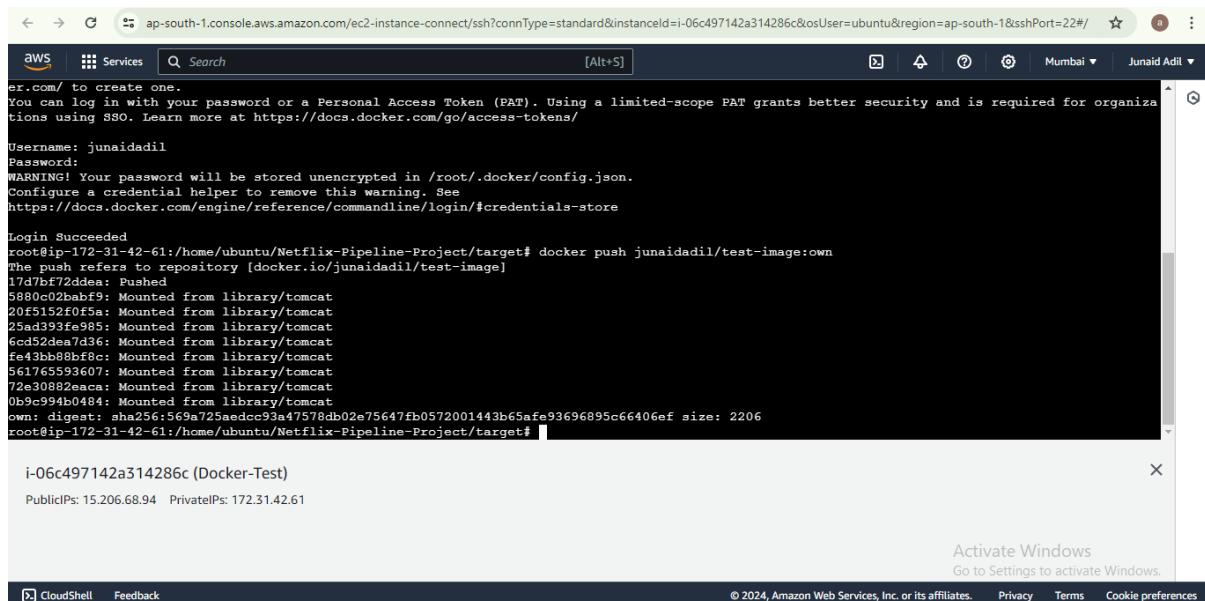


```
aws Services Search [Alt+S] Mumbai Junaid Adil
--> Using cache
--> 780ee288769d
Step 3/4 : EXPOSE 8080
--> Using cache
--> 28d2342f517a
Step 4/4 : CMD ["catalina.sh","run"]
--> Using cache
--> a55b61b87d3b
Successfully built a55b61b87d3b
Successfully tagged junaidadil/test-image:own
root@ip-172-31-42-61:/home/ubuntu/Netflix-Pipeline-Project/target# docker login
Log in with your Docker ID or email address to push and pull images from Docker Hub. If you don't have a Docker ID, head over to https://hub.docker.com/ to create one.
You can log in with your password or a Personal Access Token (PAT). Using a limited-scope PAT grants better security and is required for organizations using SSO. Learn more at https://docs.docker.com/go/access-tokens/
Username: junaidadil
Password:
WARNING! Your password will be stored unencrypted in /root/.docker/config.json.
Configure a credential helper to remove this warning. See
https://docs.docker.com/engine/reference/commandline/login/#credentials-store
Login Succeeded
root@ip-172-31-42-61:/home/ubuntu/Netflix-Pipeline-Project/target# docker push junaidadil/test-image:own
i-06c497142a314286c (Docker-Test)
PublicIPs: 15.206.68.94 PrivateIPs: 172.31.42.61
X
Activate Windows
Go to Settings to activate Windows.
CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences
```



```
aws Services Search [Alt+S] Mumbai Junaid Adil
Log in with your Docker ID or email address to push and pull images from Docker Hub. If you don't have a Docker ID, head over to https://hub.docker.com/ to create one.
You can log in with your password or a Personal Access Token (PAT). Using a limited-scope PAT grants better security and is required for organizations using SSO. Learn more at https://docs.docker.com/go/access-tokens/
Username: junaidadil
Password:
WARNING! Your password will be stored unencrypted in /root/.docker/config.json.
Configure a credential helper to remove this warning. See
https://docs.docker.com/engine/reference/commandline/login/#credentials-store
Login Succeeded
root@ip-172-31-42-61:/home/ubuntu/Netflix-Pipeline-Project/target# docker push junaidadil/test-image:own
The push refers to repository [docker.io/junaidadil/test-image]
17d7bf72dea: Preparing
5880c02habf9: Preparing
20f5152f0f5a: Preparing
25ad393fe985: Preparing
6cd52dea7d36: Preparing
fe4bb88b8fb8: Waiting
561765593607: Waiting
72e30882eaca: Waiting
0b9c994b0484: Waiting
i-06c497142a314286c (Docker-Test)
PublicIPs: 15.206.68.94 PrivateIPs: 172.31.42.61
X
Activate Windows
Go to Settings to activate Windows.
CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences
```

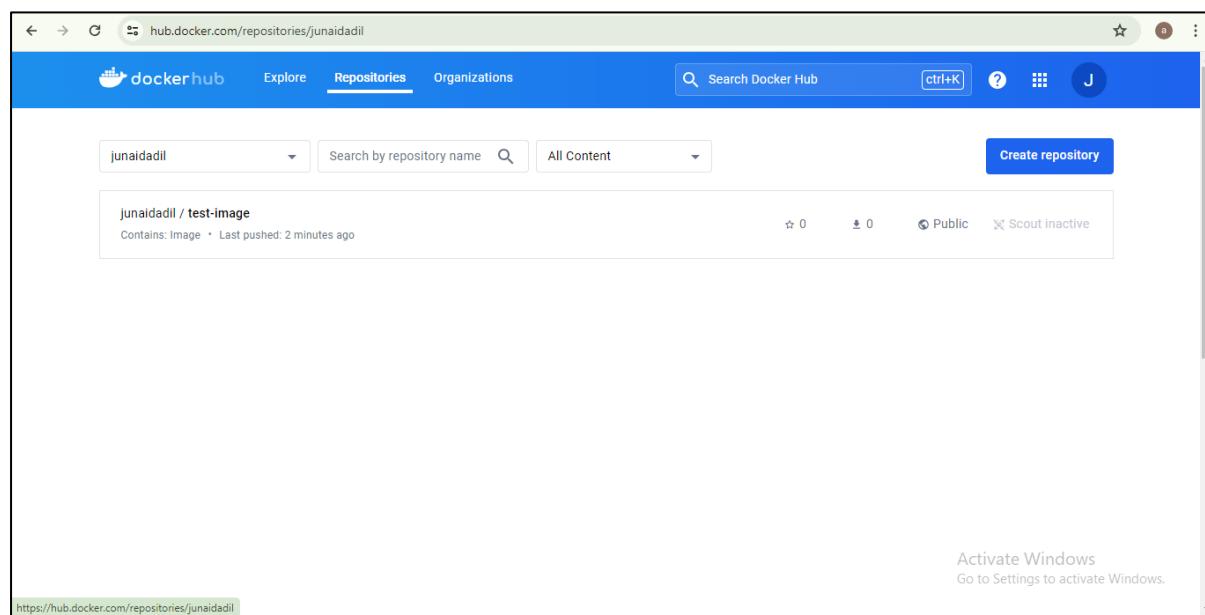
We can see the image has been pushed to docker registry



```
Username: junaidadil
Password:
WARNING! Your password will be stored unencrypted in /root/.docker/config.json.
Configure a credential helper to remove this warning. See
https://docs.docker.com/engine/reference/commandline/login/#credentials-store

Login Succeeded
root@ip-172-31-42-61:/home/ubuntu/Netflix-Pipeline-Project/target# docker push junaidadil/test-image:own
The push refers to repository [docker.io/junaidadil/test-image]
17d7bf72de: Pushed
5880c02babf9: Mounted from library/tomcat
20f5152ef0f5: Mounted from library/tomcat
25ad393fe985: Mounted from library/tomcat
6cd52dea7d36: Mounted from library/tomcat
fe43bb89bf8c: Mounted from library/tomcat
561765593607: Mounted from library/tomcat
72e30882eaca: Mounted from library/tomcat
0b9c994b0484: Mounted from library/tomcat
own: digest: sha256:569a725aedcc93a47578db02e75647fb0572001443b65afe93696895c66406ef size: 2206
root@ip-172-31-42-61:/home/ubuntu/Netflix-Pipeline-Project/target# 
```

Step 5: Login to Docker Hub account and verify the created Image



The screenshot shows the Docker Hub interface for the repository 'junaidadil/test-image'. The 'Tags' tab is selected. A single tag named 'own' is listed, which was pushed 2 minutes ago by the user 'junaidadil'. The tag details show it's a digest, based on 'linux/amd64', and was last pulled 2 minutes ago. A 'Copy' button is available for the tag name. The URL in the address bar is <https://hub.docker.com/repository/docker/junaidadil/test-image/tags>.

We can see the Image has been successfully pushed to Docker registry

Step 6: Execute command “`docker logout`” to logout of the Docker hub account

The screenshot shows the AWS CloudShell interface. The user is executing commands on an EC2 instance connected via SSH. They have run 'docker push junaidadil/test-image:own' and now run 'docker logout'. The output shows the session is logged out. The URL in the address bar is <https://ap-south-1.console.aws.amazon.com/ec2-instance-connect/ssh?connType=standard&instanceId=i-06c497142a314286c&osUser=ubuntu®ion=ap-south-1&sshPort=22#/>.

Successfully logged out of the Dockerhub account

The screenshot shows a terminal window within the AWS CloudShell interface. The URL in the address bar is `ap-south-1.console.aws.amazon.com/ec2-instance-connect/ssh?connType=standard&instanceId=i-06c497142a314286c&osUser=ubuntu®ion=ap-south-1&sshPort=22#`. The terminal output is as follows:

```
Username: junaidadil
Password:
WARNING! Your password will be stored unencrypted in /root/.docker/config.json.
Configure a credential helper to remove this warning. See
https://docs.docker.com/engine/reference/commandline/login/#credentials-store

Login Succeeded
root@ip-172-31-42-61:/home/ubuntu/Netflix-Pipeline-Project/target$ docker push junaidadil/test-image:own
The push refers to repository [docker.io/junaidadil/test-image]
17dbf72ddae: Pushed
5880c02babff9: Mounted from library/tomcat
20f5152f0f5a: Mounted from library/tomcat
25ad393fe985: Mounted from library/tomcat
6cd52deea7d36: Mounted from library/tomcat
fe43bb88bf8c: Mounted from library/tomcat
561765593607: Mounted from library/tomcat
72e30882eacc: Mounted from library/tomcat
0b9c994b0484: Mounted from library/tomcat
own: digest: sha256:569a725aedcc93a47578db02e75647fb0572001443b65afe93696895c66406ef size: 2206
root@ip-172-31-42-61:/home/ubuntu/Netflix-Pipeline-Project/target$ docker logout
Removing login credentials for https://index.docker.io/v1/
root@ip-172-31-42-61:/home/ubuntu/Netflix-Pipeline-Project/target$ 
```

Below the terminal, the message "Login Succeeded" is displayed. At the bottom of the terminal window, it says "Public IPs: 15.206.68.94 Private IPs: 172.31.42.61".

At the bottom right of the screen, there is an "Activate Windows" message with a link to "Go to Settings to activate Windows".

The footer of the page includes links for "CloudShell", "Feedback", "© 2024, Amazon Web Services, Inc. or its affiliates.", "Privacy", "Terms", and "Cookie preferences".