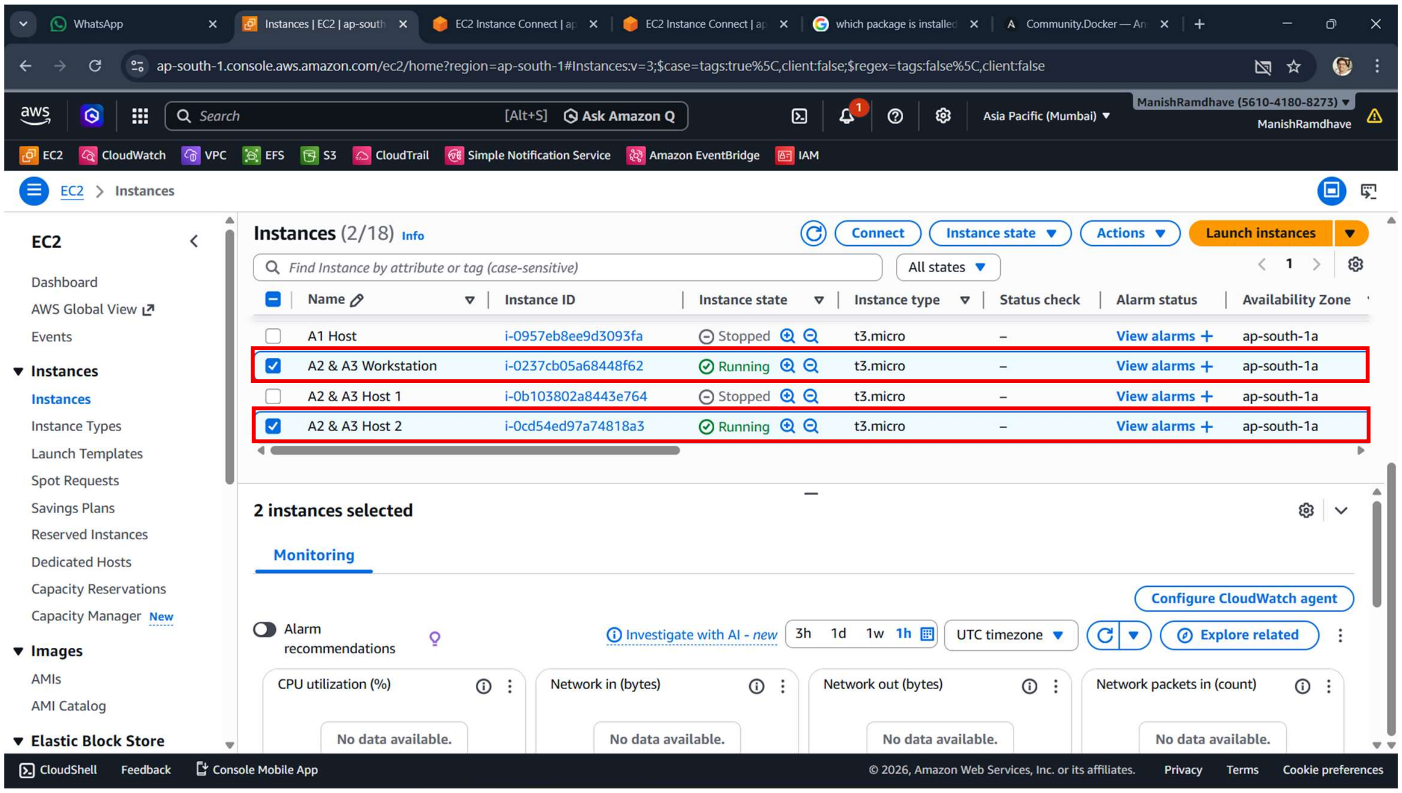
Ansible Assignment 3 (Playbook2)

Step 1: Launched two EC2 Instances for our **Ansible Workstation** and **Host2:**

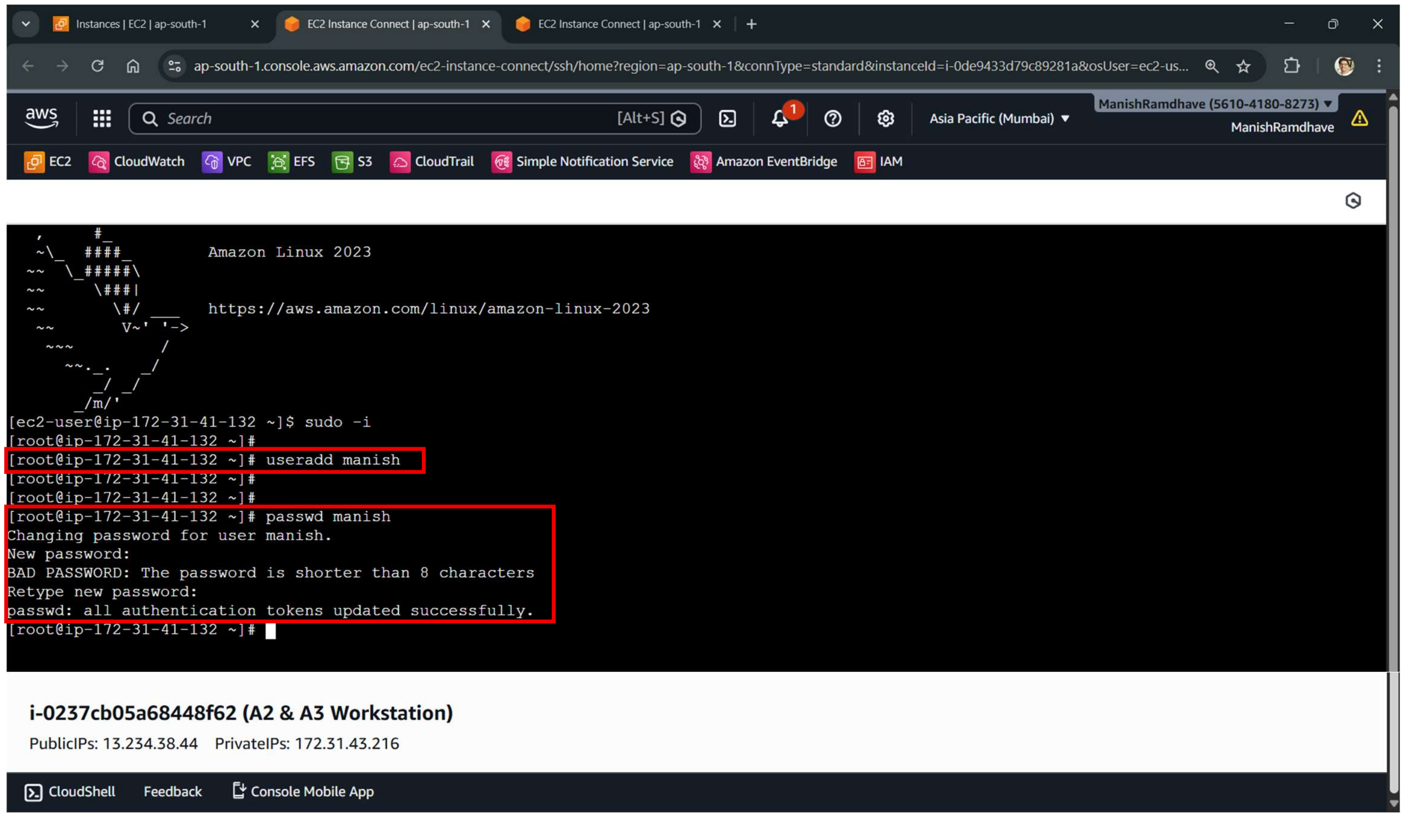


Instances (2/18)

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone
A1 Host	i-0957eb8ee9d3093fa	Stopped	t3.micro	–	View alarms	ap-south-1a
A2 & A3 Workstation	i-0237cb05a68448f62	Running	t3.micro	–	View alarms	ap-south-1a
A2 & A3 Host 1	i-0b103802a8443e764	Stopped	t3.micro	–	View alarms	ap-south-1a
A2 & A3 Host 2	i-0cd54ed97a74818a3	Running	t3.micro	–	View alarms	ap-south-1a

2 instances selected

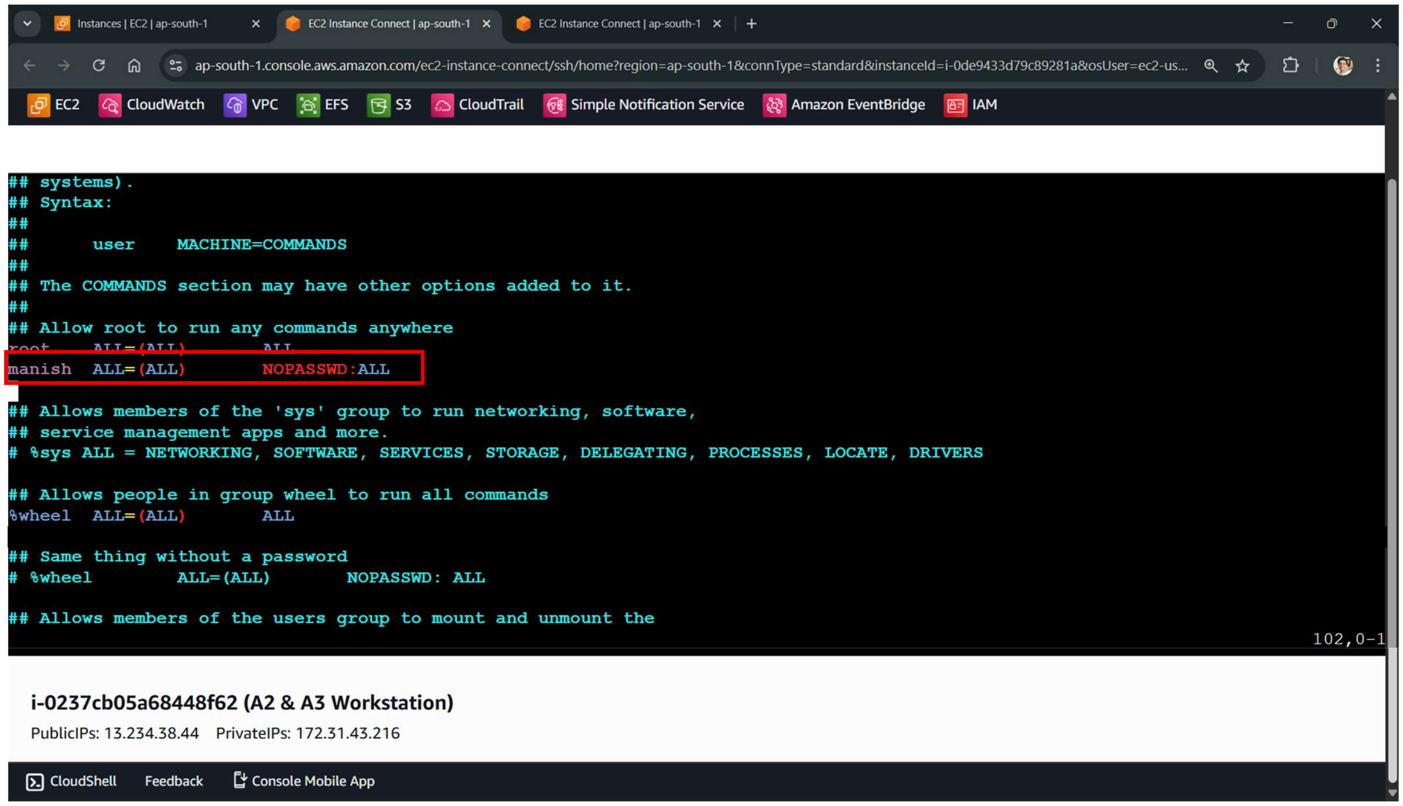
Step 2: Created an user having **'manish'** and also applied a **password** to it **on both 'Workstation' as well as 'Host2'**:



```
   ,     #_
   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\
  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
[ec2-user@ip-172-31-41-132 ~]$ sudo -i
[root@ip-172-31-41-132 ~]#
[root@ip-172-31-41-132 ~]# useradd manish
[root@ip-172-31-41-132 ~]#
[root@ip-172-31-41-132 ~]#
[root@ip-172-31-41-132 ~]# passwd manish
Changing password for user manish.
New password:
BAD PASSWORD: The password is shorter than 8 characters
Retype new password:
passwd: all authentication tokens updated successfully.
[root@ip-172-31-41-132 ~]#
```

i-0237cb05a68448f62 (A2 & A3 Workstation)
PublicIPs: 13.234.38.44 PrivateIPs: 172.31.43.216

Step 3: Updated the **‘sudoers’** file in /etc for assigning **superuser privileges** to **‘manish’** user on both the instances:

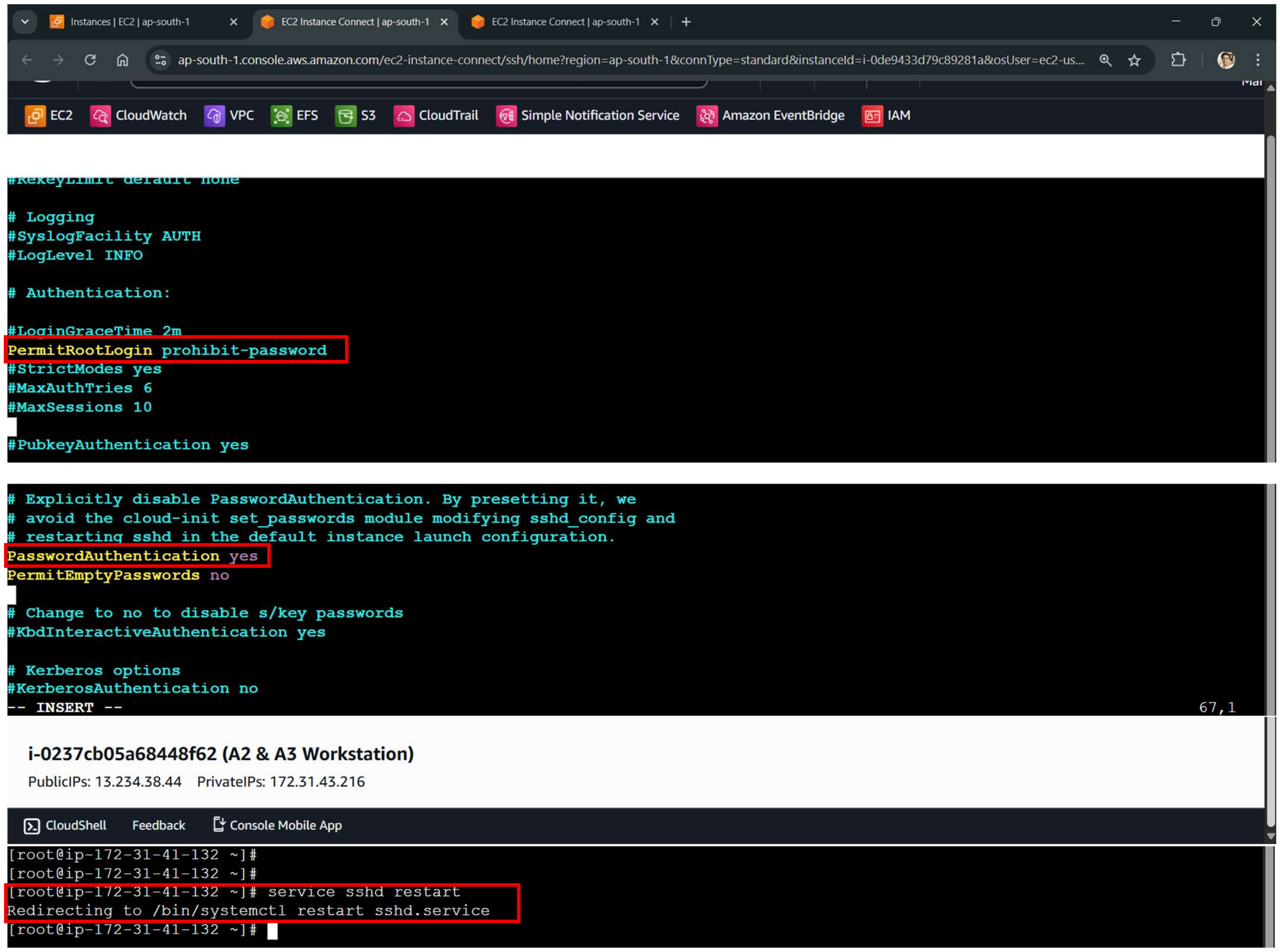

Instances | EC2 | ap-south-1
EC2 Instance Connect | ap-south-1
EC2 Instance Connect | ap-south-1
ap-south-1.console.aws.amazon.com/ec2-instance-connect/ssh/home?region=ap-south-1&connType=standard&instanceId=i-0de9433d79c89281a&osUser=ec2-us...
EC2
CloudWatch
VPC
EFS
S3
CloudTrail
Simple Notification Service
Amazon EventBridge
IAM
CloudShell
Feedback
Console Mobile App

```
## systems).
## Syntax:
##
##      user    MACHINE=COMMANDS
##
## The COMMANDS section may have other options added to it.
##
## Allow root to run any commands anywhere
root    ALL=(ALL)       ALL
manish  ALL=(ALL)       NOPASSWD:ALL

## Allows members of the 'sys' group to run networking, software,
## service management apps and more.
# %sys ALL = NETWORKING, SOFTWARE, SERVICES, STORAGE, DELEGATING, PROCESSES, LOCATE, DRIVERS

## Allows people in group wheel to run all commands
%wheel  ALL=(ALL)       ALL

## Same thing without a password
# %wheel        ALL=(ALL)       NOPASSWD: ALL

## Allows members of the users group to mount and unmount the
                                                                                          102,0-1
```

i-0237cb05a68448f62 (A2 & A3 Workstation)

PublicIPs: 13.234.38.44   PrivateIPs: 172.31.43.216

Step 4: In /etc/ssh/**sshd_config** file, made following changes and **restarted the ‘sshd service’**:


Instances | EC2 | ap-south-1
EC2 Instance Connect | ap-south-1
EC2 Instance Connect | ap-south-1
ap-south-1.console.aws.amazon.com/ec2-instance-connect/ssh/home?region=ap-south-1&connType=standard&instanceId=i-0de9433d79c89281a&osUser=ec2-us...
EC2
CloudWatch
VPC
EFS
S3
CloudTrail
Simple Notification Service
Amazon EventBridge
IAM
CloudShell
Feedback
Console Mobile App

```
#RekeyLimit default none

# Logging
#SyslogFacility AUTH
#LogLevel INFO

# Authentication:

#LoginGraceTime 2m
PermitRootLogin prohibit-password
#StrictModes yes
#MaxAuthTries 6
#MaxSessions 10

#PubkeyAuthentication yes
```

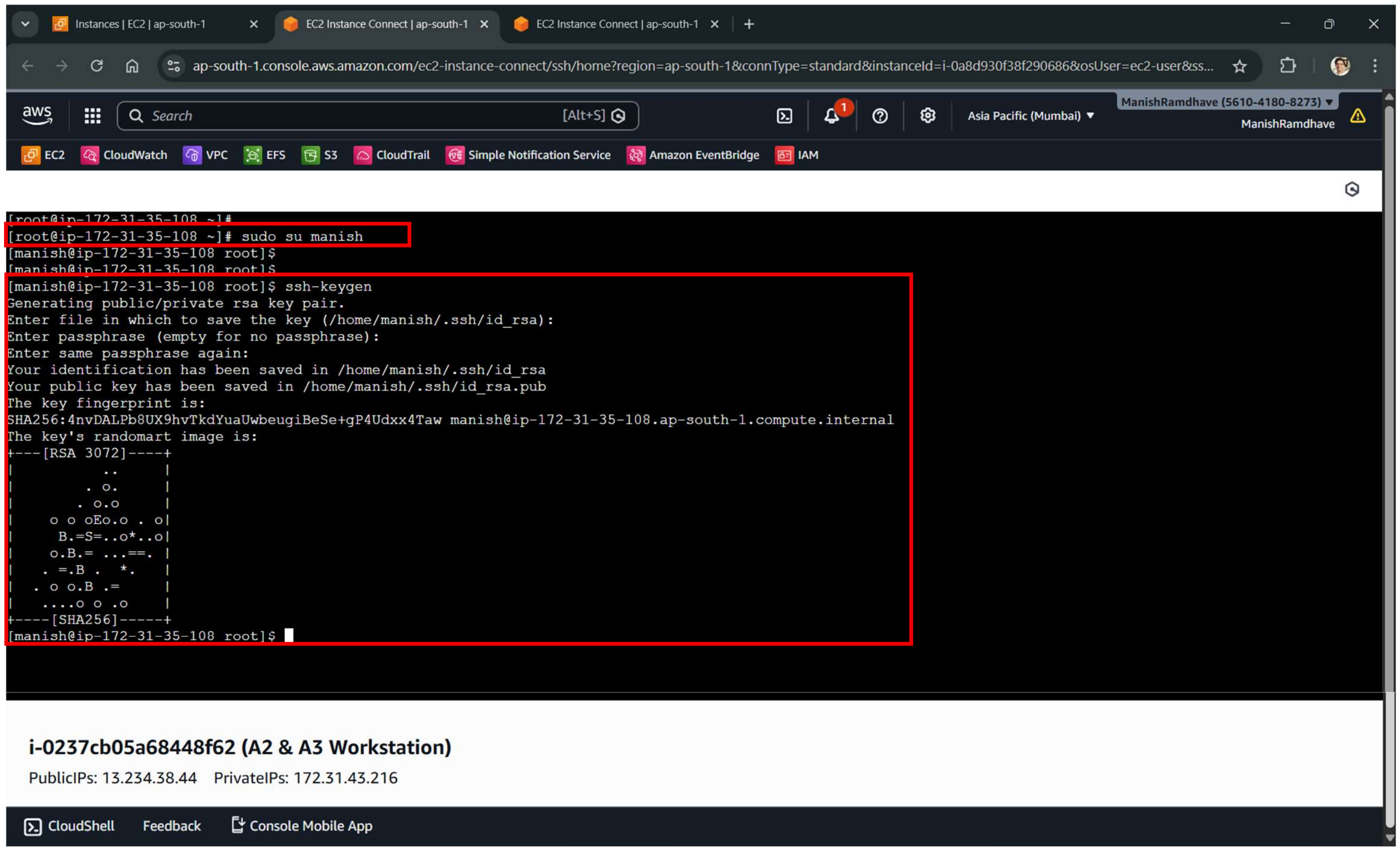
```
# Explicitly disable PasswordAuthentication. By presetting it, we
# avoid the cloud-init set_passwords module modifying sshd_config and
# restarting sshd in the default instance launch configuration.
PasswordAuthentication yes
PermitEmptyPasswords no

# Change to no to disable s/key passwords
#KbdInteractiveAuthentication yes

# Kerberos options
#KerberosAuthentication no
-- INSERT --                                                                              67,1
```

i-0237cb05a68448f62 (A2 & A3 Workstation)

PublicIPs: 13.234.38.44   PrivateIPs: 172.31.43.216

```
[root@ip-172-31-41-132 ~]#
[root@ip-172-31-41-132 ~]#
[root@ip-172-31-41-132 ~]# service sshd restart
Redirecting to /bin/systemctl restart sshd.service
[root@ip-172-31-41-132 ~]#
```

Step 5: Logged in to the **'manish'** user and generated **SSH private and public keypairs**:

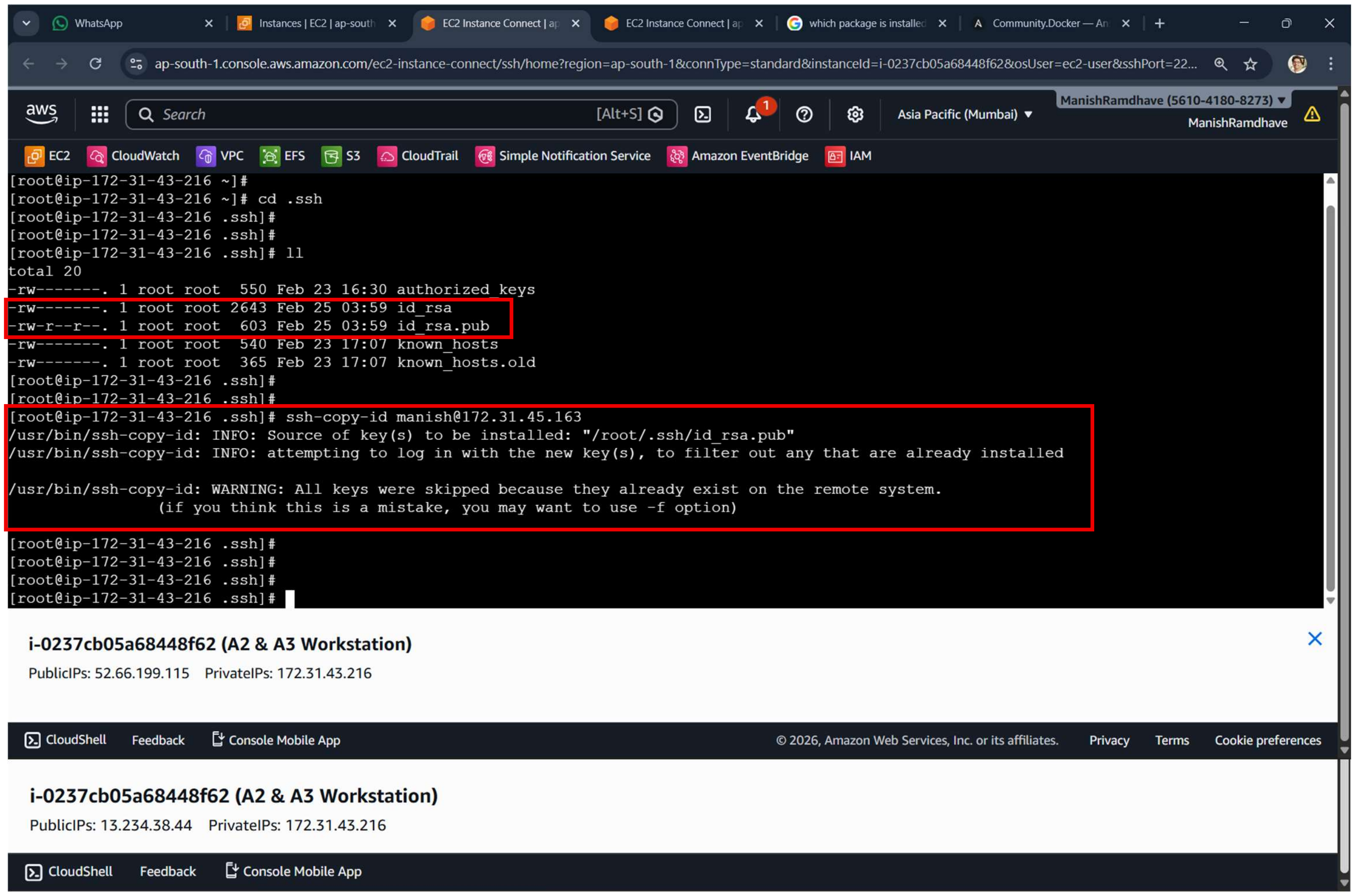

Instances | EC2 | ap-south-1
EC2 Instance Connect | ap-south-1
EC2 Instance Connect | ap-south-1
ap-south-1.console.aws.amazon.com/ec2-instance-connect/ssh/home?region=ap-south-1&connType=standard&instanceId=i-0a8d930f38f290686&osUser=ec2-user&ss...
Search
[Alt+S]
Asia Pacific (Mumbai)
ManishRamdhave (5610-4180-8273)
ManishRamdhave
EC2
CloudWatch
VPC
EFS
S3
CloudTrail
Simple Notification Service
Amazon EventBridge
IAM

```
[root@ip-172-31-35-108 ~]#
[root@ip-172-31-35-108 ~]# sudo su manish
[manish@ip-172-31-35-108 root]$
[manish@ip-172-31-35-108 root]$
[manish@ip-172-31-35-108 root]$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/manish/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/manish/.ssh/id_rsa
Your public key has been saved in /home/manish/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:4nvDALPb8UX9hvTkdYuaUwbeugiBeSe+gP4Udxx4Taw manish@ip-172-31-35-108.ap-south-1.compute.internal
The key's randomart image is:
+---[RSA 3072]----+
|          ..     |
|        . o.     |
|       . o.o     |
|    o o oEo.o . o|
|     B.=S=..o*..o|
|    o.B.= ...==. |
|   . =.B .  *.   |
|  . o o.B .=     |
|   ....o o .o    |
+----[SHA256]-----+
[manish@ip-172-31-35-108 root]$
```

i-0237cb05a68448f62 (A2 & A3 Workstation)

PublicIPs: 13.234.38.44 PrivateIPs: 172.31.43.216

Step 6: Generated both **Private (id_rsa) and Public (id_rsa.pub) key pairs** and **copied them to the Host2 instance** using its private IP:


WhatsApp
Instances | EC2 | ap-south
EC2 Instance Connect | ap
EC2 Instance Connect | ap
which package is installed
Community.Docker — An
ap-south-1.console.aws.amazon.com/ec2-instance-connect/ssh/home?region=ap-south-1&connType=standard&instanceId=i-0237cb05a68448f62&osUser=ec2-user&sshPort=22...
Search
[Alt+S]
Asia Pacific (Mumbai)
ManishRamdhave (5610-4180-8273)
ManishRamdhave
EC2
CloudWatch
VPC
EFS
S3
CloudTrail
Simple Notification Service
Amazon EventBridge
IAM

```
[root@ip-172-31-43-216 ~]#
[root@ip-172-31-43-216 ~]# cd .ssh
[root@ip-172-31-43-216 .ssh]#
[root@ip-172-31-43-216 .ssh]#
[root@ip-172-31-43-216 .ssh]# ll
total 20
-rw-------. 1 root root  550 Feb 23 16:30 authorized_keys
-rw-------. 1 root root 2643 Feb 25 03:59 id_rsa
-rw-r--r--. 1 root root  603 Feb 25 03:59 id_rsa.pub
-rw-------. 1 root root  540 Feb 23 17:07 known_hosts
-rw-------. 1 root root  365 Feb 23 17:07 known_hosts.old
[root@ip-172-31-43-216 .ssh]#
[root@ip-172-31-43-216 .ssh]#
[root@ip-172-31-43-216 .ssh]# ssh-copy-id manish@172.31.45.163
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/root/.ssh/id_rsa.pub"
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed

/usr/bin/ssh-copy-id: WARNING: All keys were skipped because they already exist on the remote system.
                (if you think this is a mistake, you may want to use -f option)

[root@ip-172-31-43-216 .ssh]#
[root@ip-172-31-43-216 .ssh]#
[root@ip-172-31-43-216 .ssh]#
[root@ip-172-31-43-216 .ssh]#
```

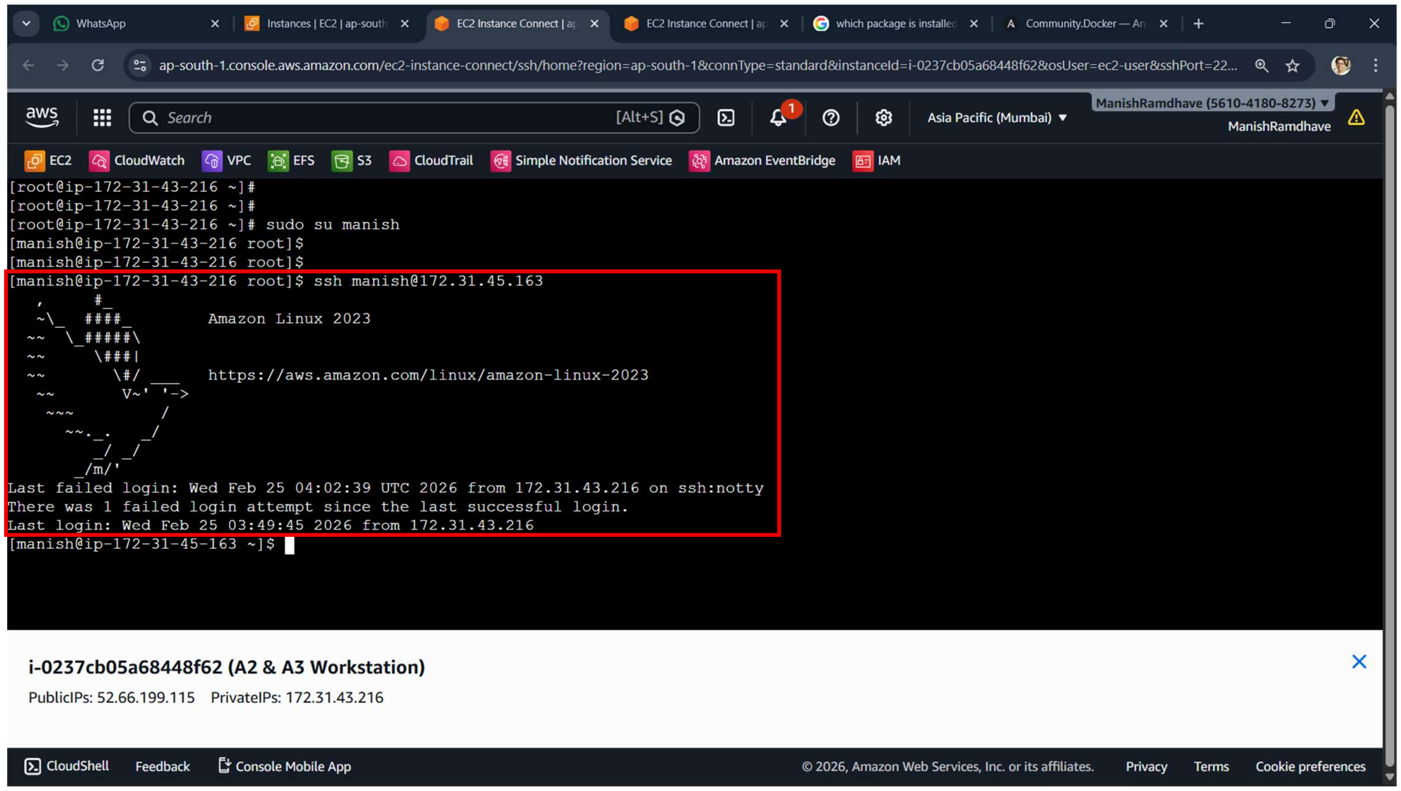
i-0237cb05a68448f62 (A2 & A3 Workstation)

PublicIPs: 52.66.199.115 PrivateIPs: 172.31.43.216

i-0237cb05a68448f62 (A2 & A3 Workstation)

PublicIPs: 13.234.38.44 PrivateIPs: 172.31.43.216

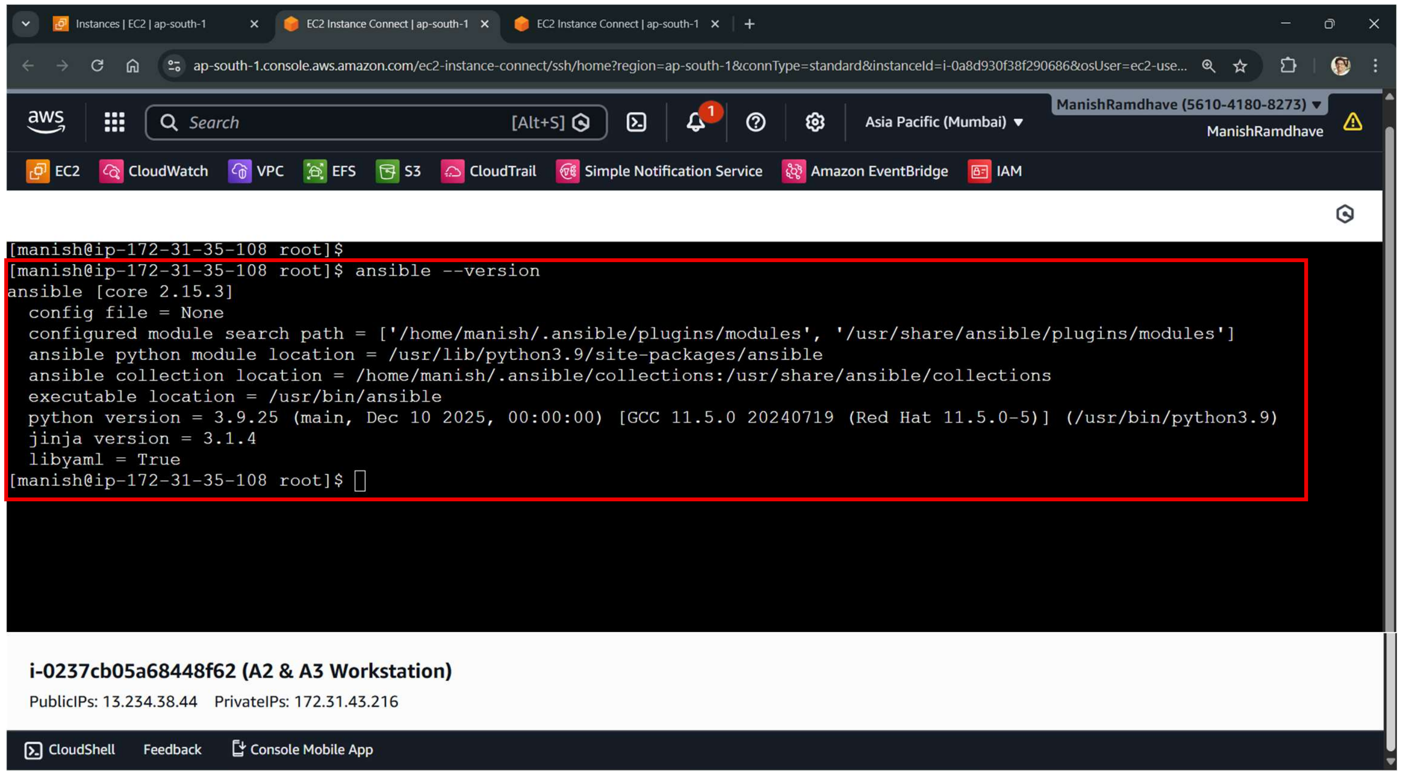
Step 7: Successfully logged in to the **Host2** from **Workstation** using SSH and Private IP of Host:



```
[root@ip-172-31-43-216 ~]#
[root@ip-172-31-43-216 ~]#
[root@ip-172-31-43-216 ~]# sudo su manish
[manish@ip-172-31-43-216 root]$
[manish@ip-172-31-43-216 root]$
[manish@ip-172-31-43-216 root]$ ssh manish@172.31.45.163
   ,     #_
   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\
  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
Last failed login: Wed Feb 25 04:02:39 UTC 2026 from 172.31.43.216 on ssh:notty
There was 1 failed login attempt since the last successful login.
Last login: Wed Feb 25 03:49:45 2026 from 172.31.43.216
[manish@ip-172-31-45-163 ~]$
```

i-0237cb05a68448f62 (A2 & A3 Workstation)
PublicIPs: 52.66.199.115 PrivateIPs: 172.31.43.216

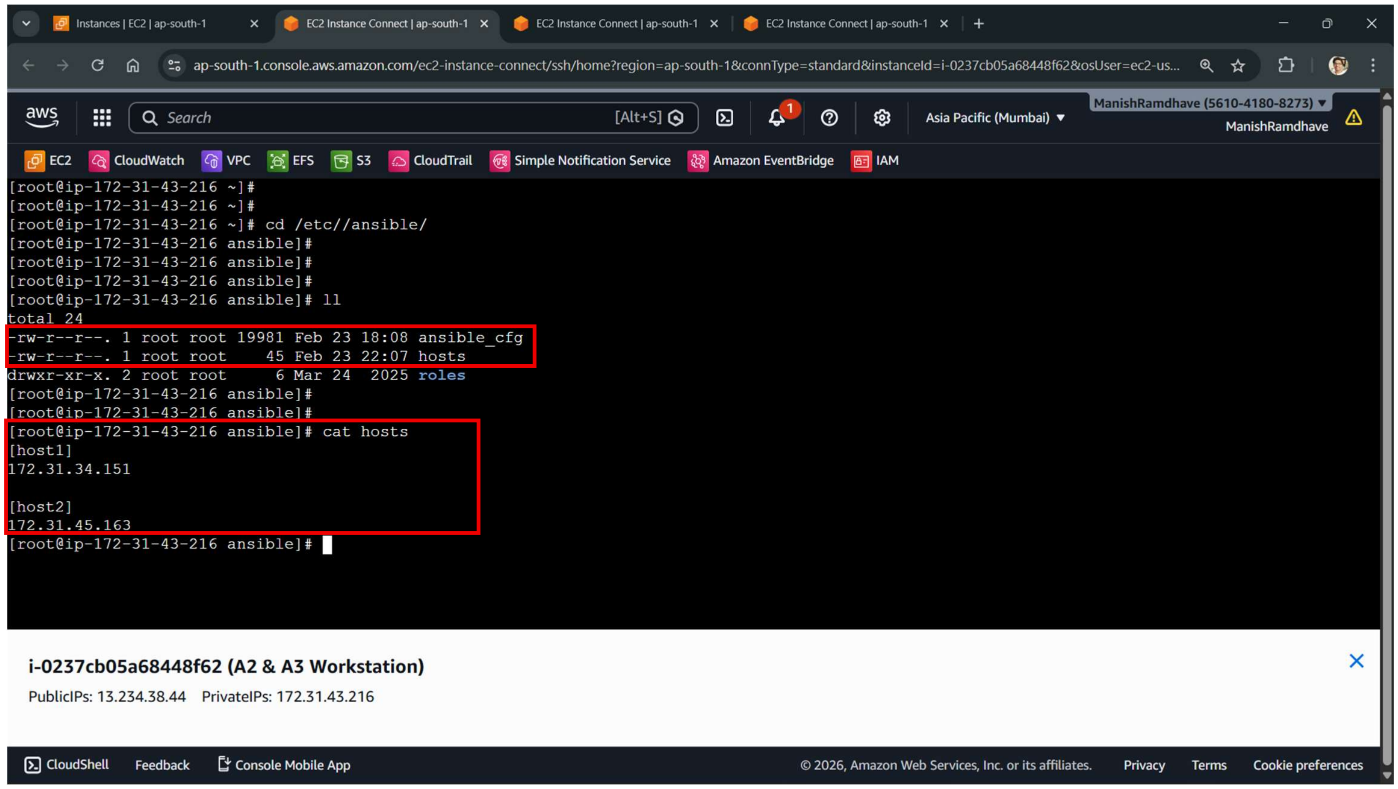
Step 8: Installed **Ansible** on the **Workstation**:



```
[manish@ip-172-31-35-108 root]$
[manish@ip-172-31-35-108 root]$ ansible --version
ansible [core 2.15.3]
  config file = None
  configured module search path = ['/home/manish/.ansible/plugins/modules', '/usr/share/ansible/plugins/modules']
  ansible python module location = /usr/lib/python3.9/site-packages/ansible
  ansible collection location = /home/manish/.ansible/collections:/usr/share/ansible/collections
  executable location = /usr/bin/ansible
  python version = 3.9.25 (main, Dec 10 2025, 00:00:00) [GCC 11.5.0 20240719 (Red Hat 11.5.0-5)] (/usr/bin/python3.9)
  jinja version = 3.1.4
  libyaml = True
[manish@ip-172-31-35-108 root]$
```

i-0237cb05a68448f62 (A2 & A3 Workstation)
PublicIPs: 13.234.38.44 PrivateIPs: 172.31.43.216

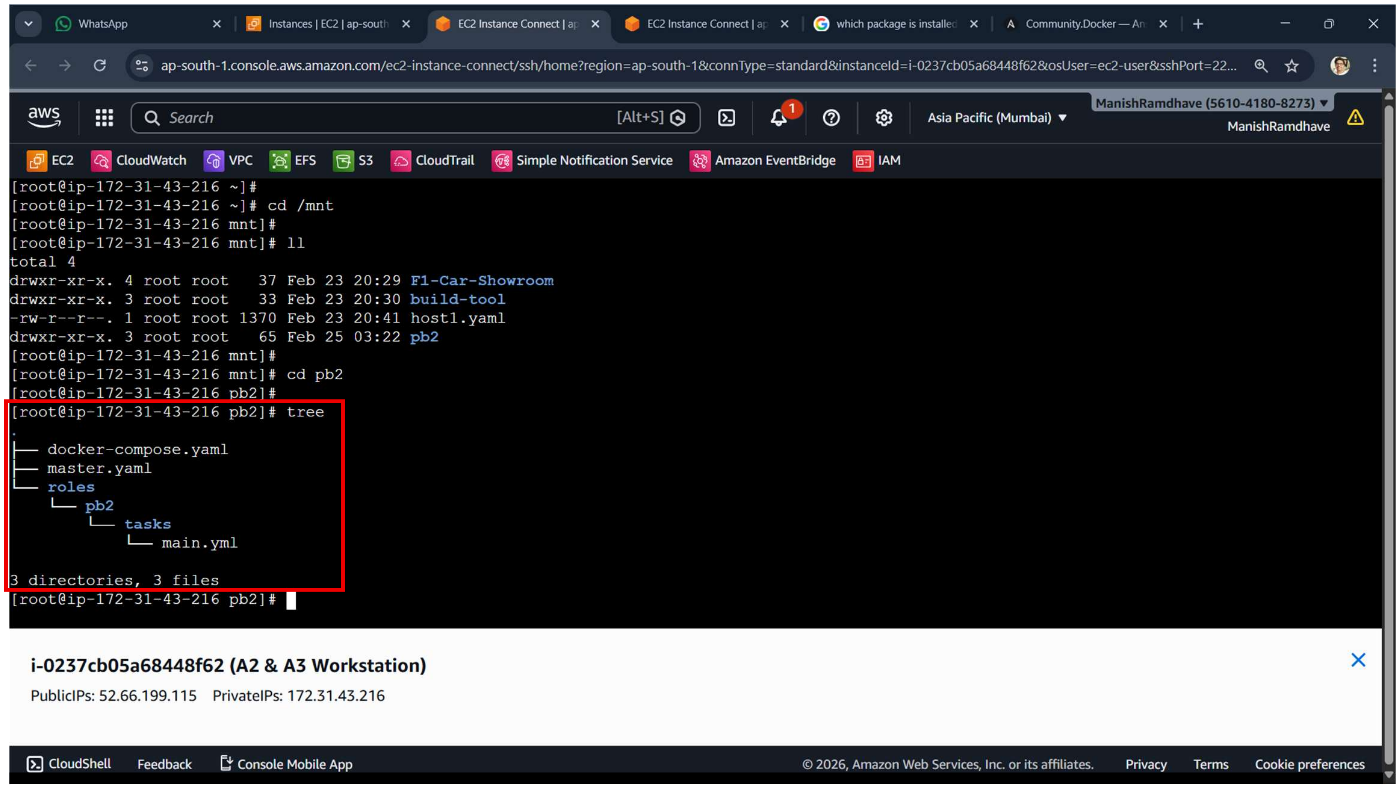
Step 9: Created two files in **/etc/ansible**/ path, **‘hosts’** and **‘ansible.cfg’.** The ‘hosts’ consists of the host instances’ private IPs:



```
[root@ip-172-31-43-216 ~]#
[root@ip-172-31-43-216 ~]#
[root@ip-172-31-43-216 ~]# cd /etc//ansible/
[root@ip-172-31-43-216 ansible]#
[root@ip-172-31-43-216 ansible]#
[root@ip-172-31-43-216 ansible]#
[root@ip-172-31-43-216 ansible]# ll
total 24
-rw-r--r--. 1 root root 19981 Feb 23 18:08 ansible_cfg
-rw-r--r--. 1 root root    45 Feb 23 22:07 hosts
drwxr-xr-x. 2 root root     6 Mar 24  2025 roles
[root@ip-172-31-43-216 ansible]#
[root@ip-172-31-43-216 ansible]#
[root@ip-172-31-43-216 ansible]# cat hosts
[host1]
172.31.34.151

[host2]
172.31.45.163
[root@ip-172-31-43-216 ansible]#
```

i-0237cb05a68448f62 (A2 & A3 Workstation)
PublicIPs: 13.234.38.44 PrivateIPs: 172.31.43.216

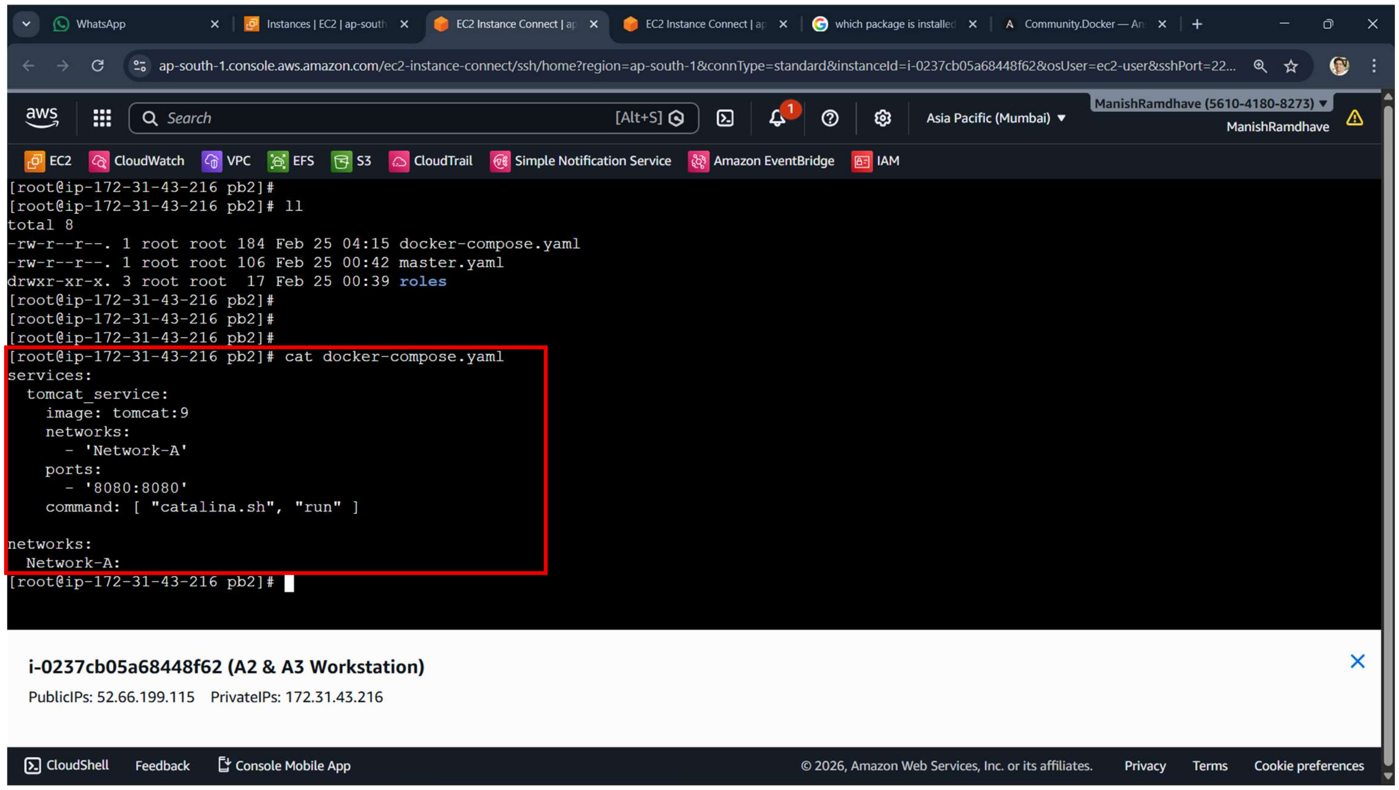
Step 10: Created those **files and folders** as shown below in the **tree** syntax:



```
[root@ip-172-31-43-216 ~]#
[root@ip-172-31-43-216 ~]# cd /mnt
[root@ip-172-31-43-216 mnt]#
[root@ip-172-31-43-216 mnt]# ll
total 4
drwxr-xr-x. 4 root root   37 Feb 23 20:29 F1-Car-Showroom
drwxr-xr-x. 3 root root   33 Feb 23 20:30 build-tool
-rw-r--r--. 1 root root 1370 Feb 23 20:41 host1.yaml
drwxr-xr-x. 3 root root   65 Feb 25 03:22 pb2
[root@ip-172-31-43-216 mnt]#
[root@ip-172-31-43-216 mnt]# cd pb2
[root@ip-172-31-43-216 pb2]#
[root@ip-172-31-43-216 pb2]# tree
.
├── docker-compose.yaml
├── master.yaml
└── roles
    └── pb2
        └── tasks
            └── main.yml

3 directories, 3 files
[root@ip-172-31-43-216 pb2]#
```

i-0237cb05a68448f62 (A2 & A3 Workstation)
PublicIPs: 52.66.199.115 PrivateIPs: 172.31.43.216

Step 11: Added the following script in thef **'docker-compose.yaml'** file of **'/mnt/pb2/'** path:

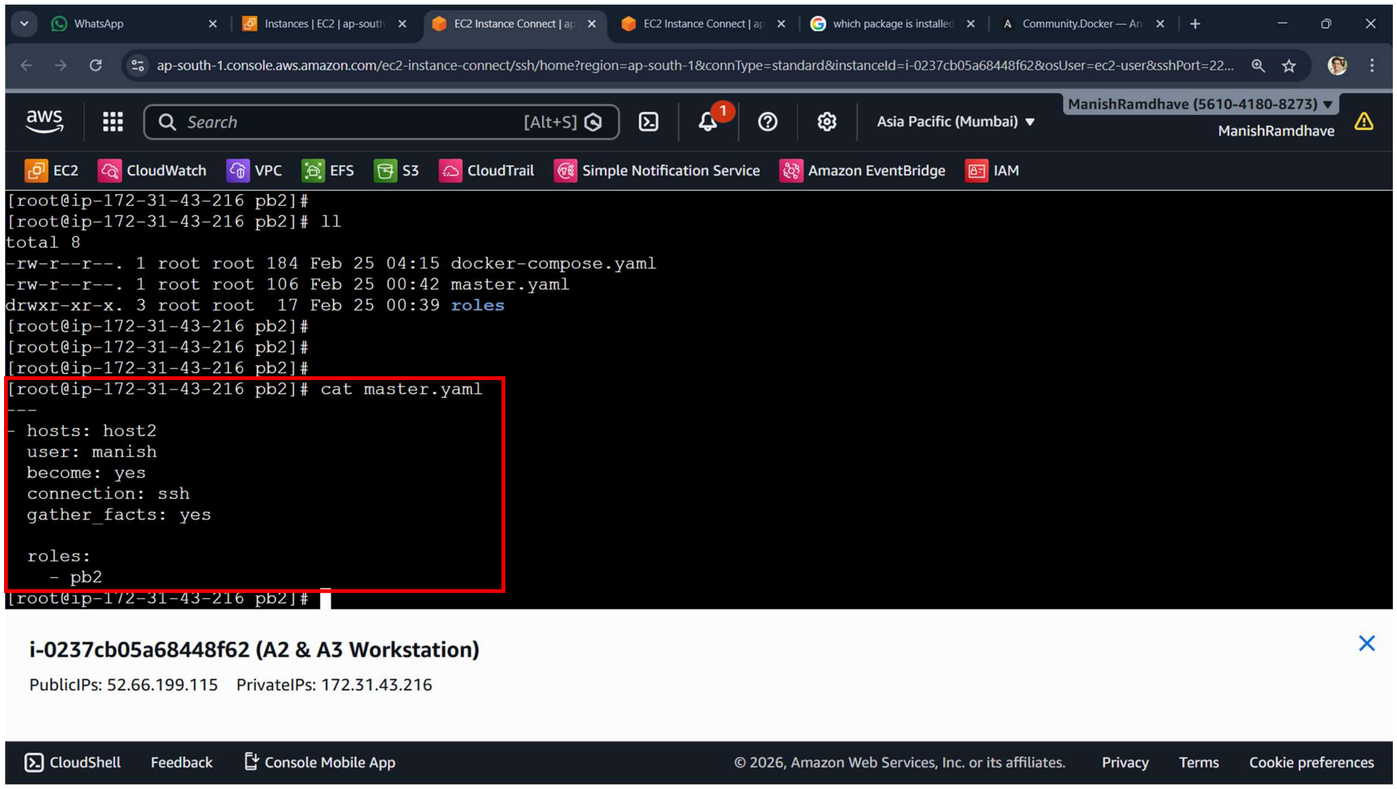

WhatsApp
Instances | EC2 | ap-south
EC2 Instance Connect | ap
EC2 Instance Connect | ap
which package is installed
Community.Docker — An
ap-south-1.console.aws.amazon.com/ec2-instance-connect/ssh/home?region=ap-south-1&connType=standard&instanceId=i-0237cb05a68448f62&osUser=ec2-user&sshPort=22...
Search
[Alt+S]
Asia Pacific (Mumbai)
ManishRamdhave (5610-4180-8273)
ManishRamdhave
EC2
CloudWatch
VPC
EFS
S3
CloudTrail
Simple Notification Service
Amazon EventBridge
IAM

```
[root@ip-172-31-43-216 pb2]#
[root@ip-172-31-43-216 pb2]# ll
total 8
-rw-r--r--. 1 root root 184 Feb 25 04:15 docker-compose.yaml
-rw-r--r--. 1 root root 106 Feb 25 00:42 master.yaml
drwxr-xr-x. 3 root root  17 Feb 25 00:39 roles
[root@ip-172-31-43-216 pb2]#
[root@ip-172-31-43-216 pb2]#
[root@ip-172-31-43-216 pb2]#
[root@ip-172-31-43-216 pb2]# cat docker-compose.yaml
services:
  tomcat_service:
    image: tomcat:9
    networks:
      - 'Network-A'
    ports:
      - '8080:8080'
    command: [ "catalina.sh", "run" ]

networks:
  Network-A:
[root@ip-172-31-43-216 pb2]#
```

i-0237cb05a68448f62 (A2 & A3 Workstation)

PublicIPs: 52.66.199.115 PrivateIPs: 172.31.43.216

Step 12: Added the following script in thef **'docker-compose.yaml'** file of **'/mnt/pb2/'** path:

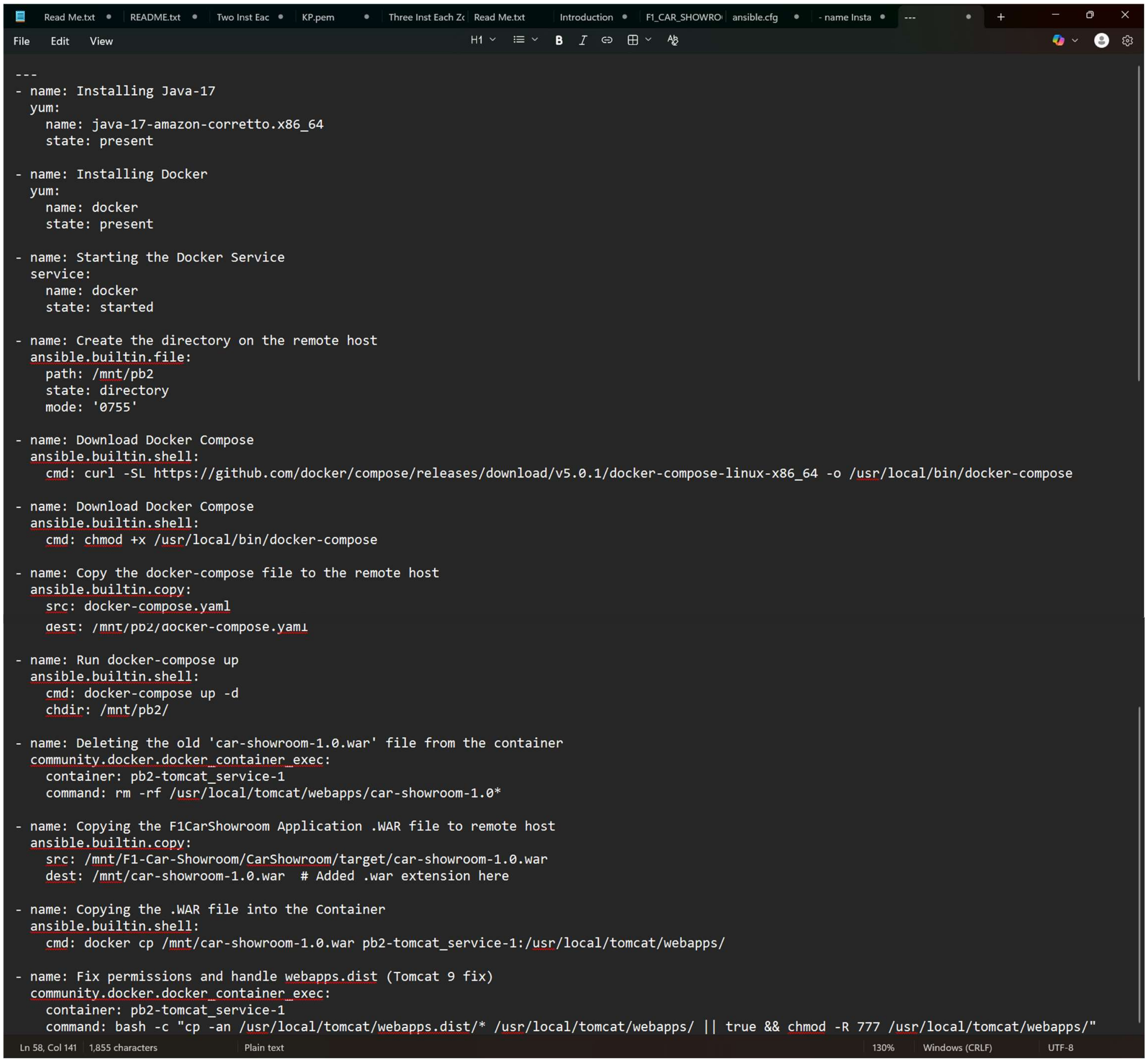

WhatsApp
Instances | EC2 | ap-south
EC2 Instance Connect | ap
EC2 Instance Connect | ap
which package is installed
Community.Docker — An
ap-south-1.console.aws.amazon.com/ec2-instance-connect/ssh/home?region=ap-south-1&connType=standard&instanceId=i-0237cb05a68448f62&osUser=ec2-user&sshPort=22...
Search
[Alt+S]
Asia Pacific (Mumbai)
ManishRamdhave (5610-4180-8273)
ManishRamdhave
EC2
CloudWatch
VPC
EFS
S3
CloudTrail
Simple Notification Service
Amazon EventBridge
IAM

```
[root@ip-172-31-43-216 pb2]#
[root@ip-172-31-43-216 pb2]# ll
total 8
-rw-r--r--. 1 root root 184 Feb 25 04:15 docker-compose.yaml
-rw-r--r--. 1 root root 106 Feb 25 00:42 master.yaml
drwxr-xr-x. 3 root root  17 Feb 25 00:39 roles
[root@ip-172-31-43-216 pb2]#
[root@ip-172-31-43-216 pb2]#
[root@ip-172-31-43-216 pb2]#
[root@ip-172-31-43-216 pb2]# cat master.yaml
---
- hosts: host2
  user: manish
  become: yes
  connection: ssh
  gather_facts: yes

  roles:
    - pb2
[root@ip-172-31-43-216 pb2]#
```

i-0237cb05a68448f62 (A2 & A3 Workstation)

PublicIPs: 52.66.199.115 PrivateIPs: 172.31.43.216

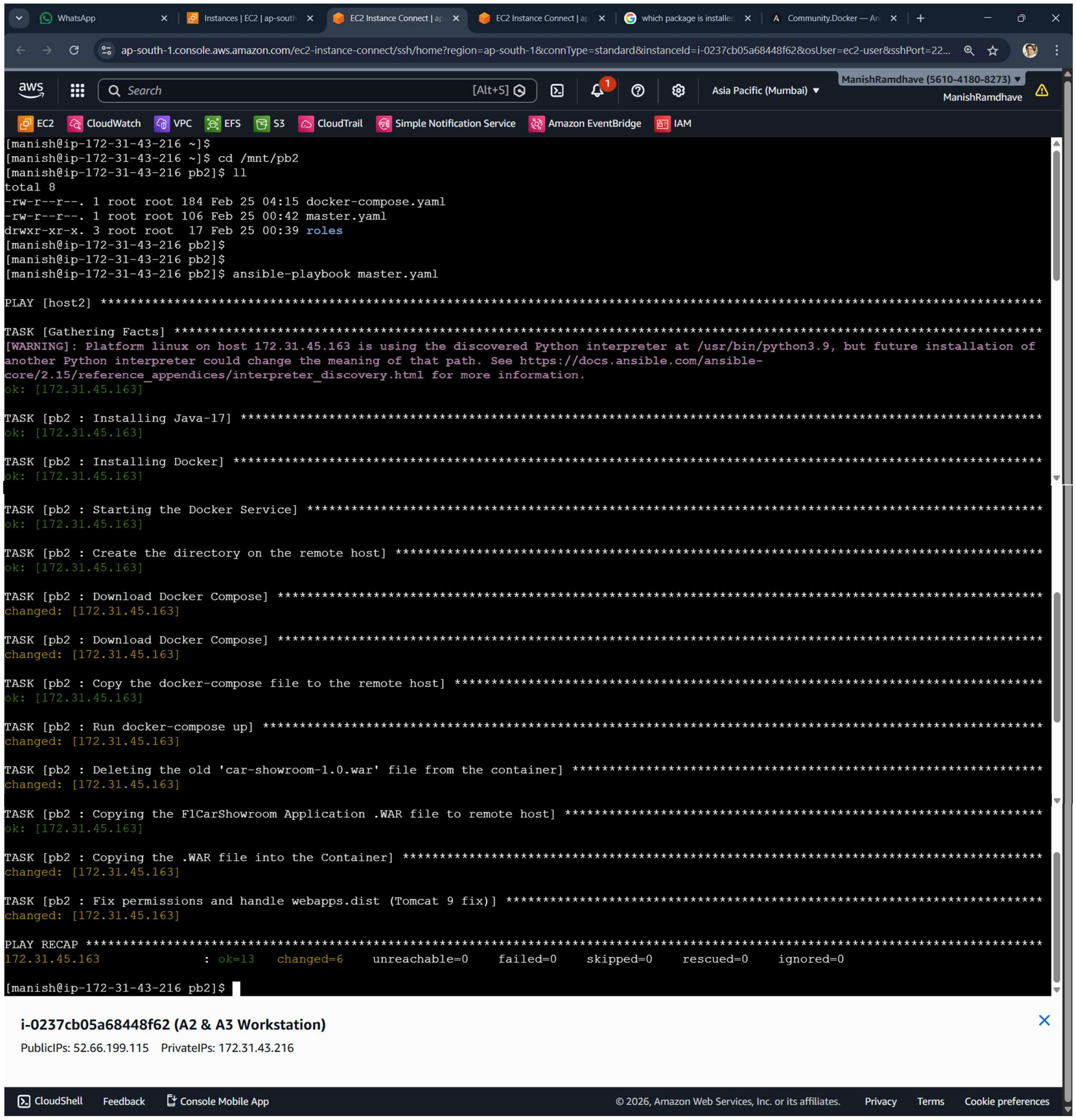
Step 13: Added the following script in the **‘main.yaml’** file of **‘/mnt/pb2/roles/pb2/tasks’** path:



```
---
- name: Installing Java-17
  yum:
    name: java-17-amazon-corretto.x86_64
    state: present

- name: Installing Docker
  yum:
    name: docker
    state: present

- name: Starting the Docker Service
  service:
    name: docker
    state: started

- name: Create the directory on the remote host
  ansible.builtin.file:
    path: /mnt/pb2
    state: directory
    mode: '0755'

- name: Download Docker Compose
  ansible.builtin.shell:
    cmd: curl -SL https://github.com/docker/compose/releases/download/v5.0.1/docker-compose-linux-x86_64 -o /usr/local/bin/docker-compose

- name: Download Docker Compose
  ansible.builtin.shell:
    cmd: chmod +x /usr/local/bin/docker-compose

- name: Copy the docker-compose file to the remote host
  ansible.builtin.copy:
    src: docker-compose.yaml
    dest: /mnt/pb2/docker-compose.yaml

- name: Run docker-compose up
  ansible.builtin.shell:
    cmd: docker-compose up -d
    chdir: /mnt/pb2/

- name: Deleting the old 'car-showroom-1.0.war' file from the container
  community.docker.docker_container_exec:
    container: pb2-tomcat_service-1
    command: rm -rf /usr/local/tomcat/webapps/car-showroom-1.0*

- name: Copying the F1CarShowroom Application .WAR file to remote host
  ansible.builtin.copy:
    src: /mnt/F1-Car-Showroom/CarShowroom/target/car-showroom-1.0.war
    dest: /mnt/car-showroom-1.0.war  # Added .war extension here

- name: Copying the .WAR file into the Container
  ansible.builtin.shell:
    cmd: docker cp /mnt/car-showroom-1.0.war pb2-tomcat_service-1:/usr/local/tomcat/webapps/

- name: Fix permissions and handle webapps.dist (Tomcat 9 fix)
  community.docker.docker_container_exec:
    container: pb2-tomcat_service-1
    command: bash -c "cp -an /usr/local/tomcat/webapps.dist/* /usr/local/tomcat/webapps/ || true && chmod -R 777 /usr/local/tomcat/webapps/"
```

\

Step 14: Executed the **Playbook2 Script using ansible command** and all these commands are executed on **'Host2 Instance'**:

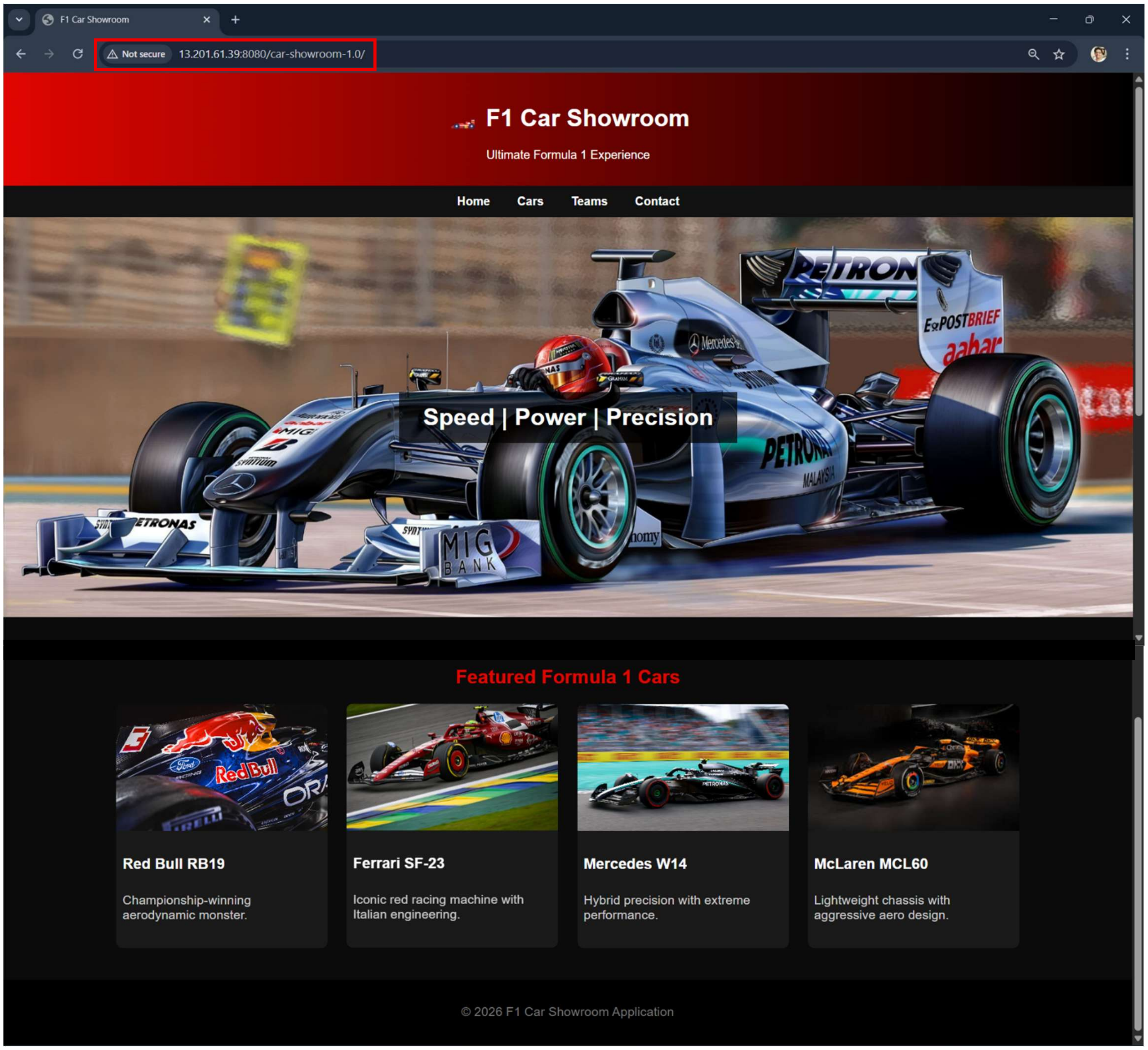

WhatsApp
Instances | EC2 | ap-south
EC2 Instance Connect | ap
EC2 Instance Connect | ap
which package is installed
Community.Docker — An
ap-south-1.console.aws.amazon.com/ec2-instance-connect/ssh/home?region=ap-south-1&connType=standard&instanceId=i-0237cb05a68448f62&osUser=ec2-user&sshPort=22...
Search [Alt+S]
Asia Pacific (Mumbai)
ManishRamdhave (5610-4180-8273)
ManishRamdhave
EC2 CloudWatch VPC EFS S3 CloudTrail Simple Notification Service Amazon EventBridge IAM

```
[manish@ip-172-31-43-216 ~]$
[manish@ip-172-31-43-216 ~]$ cd /mnt/pb2
[manish@ip-172-31-43-216 pb2]$ ll
total 8
-rw-r--r--. 1 root root 184 Feb 25 04:15 docker-compose.yaml
-rw-r--r--. 1 root root 106 Feb 25 00:42 master.yaml
drwxr-xr-x. 3 root root  17 Feb 25 00:39 roles
[manish@ip-172-31-43-216 pb2]$
[manish@ip-172-31-43-216 pb2]$
[manish@ip-172-31-43-216 pb2]$ ansible-playbook master.yaml

PLAY [host2] *************************************************************************************************************

TASK [Gathering Facts] ***************************************************************************************************
[WARNING]: Platform linux on host 172.31.45.163 is using the discovered Python interpreter at /usr/bin/python3.9, but future installation of
another Python interpreter could change the meaning of that path. See https://docs.ansible.com/ansible-
core/2.15/reference_appendices/interpreter_discovery.html for more information.
ok: [172.31.45.163]

TASK [pb2 : Installing Java-17] ******************************************************************************************
ok: [172.31.45.163]

TASK [pb2 : Installing Docker] *******************************************************************************************
ok: [172.31.45.163]

TASK [pb2 : Starting the Docker Service] *********************************************************************************
ok: [172.31.45.163]

TASK [pb2 : Create the directory on the remote host] *********************************************************************
ok: [172.31.45.163]

TASK [pb2 : Download Docker Compose] *************************************************************************************
changed: [172.31.45.163]

TASK [pb2 : Download Docker Compose] *************************************************************************************
changed: [172.31.45.163]

TASK [pb2 : Copy the docker-compose file to the remote host] *************************************************************
ok: [172.31.45.163]

TASK [pb2 : Run docker-compose up] ***************************************************************************************
changed: [172.31.45.163]

TASK [pb2 : Deleting the old 'car-showroom-1.0.war' file from the container] *********************************************
changed: [172.31.45.163]

TASK [pb2 : Copying the FlCarShowroom Application .WAR file to remote host] **********************************************
ok: [172.31.45.163]

TASK [pb2 : Copying the .WAR file into the Container] ********************************************************************
changed: [172.31.45.163]

TASK [pb2 : Fix permissions and handle webapps.dist (Tomcat 9 fix)] ******************************************************
changed: [172.31.45.163]

PLAY RECAP ***************************************************************************************************************
172.31.45.163              : ok=13   changed=6    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

[manish@ip-172-31-43-216 pb2]$
```

i-0237cb05a68448f62 (A2 & A3 Workstation)
PublicIPs: 52.66.199.115   PrivateIPs: 172.31.43.216

Results:

We have successfully deployed the **‘car-showroom-1.0’ application** using **‘docker-compose.yaml’ file & ‘ansible adhoc command with modules’** and hosted the same application using **Host2 Public IP on Tomcat Server Port No.8080:**


F1 Car Showroom
Not secure
13.201.61.39:8080/car-showroom-1.0/
F1 Car Showroom
Ultimate Formula 1 Experience
Home
Cars
Teams
Contact
Speed | Power | Precision
Featured Formula 1 Cars
Red Bull RB19
Championship-winning aerodynamic monster.
Ferrari SF-23
Iconic red racing machine with Italian engineering.
Mercedes W14
Hybrid precision with extreme performance.
McLaren MCL60
Lightweight chassis with aggressive aero design.
© 2026 F1 Car Showroom Application