

Ansible Assignment 1

Step 1: Launched instances for our Jenkins Master and Slave:

The screenshot shows the AWS Management Console for the 'Instances' page. The left sidebar contains navigation options like Dashboard, AWS Global View, Events, Instances, Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity Reservations, Capacity Manager, Images, and Elastic Block Store. The main content area shows a list of instances with columns for Name, Instance ID, Instance state, Instance type, Status check, Alarm status, Availability Zone, and Public IPv. Two instances are highlighted with a red box: 'A1 Workstation' (Instance ID: i-0a8d930f38f290686, State: Running, Type: t3.micro, Availability Zone: ap-south-1a, Public IPv: ec2-13-12) and 'A1 Host' (Instance ID: i-0957eb8ee9d3093fa, State: Running, Type: t3.micro, Availability Zone: ap-south-1a, Public IPv: ec2-3-110). Below the list, the 'Monitoring' section shows various metrics like CPU utilization, network in/out, and network packets.

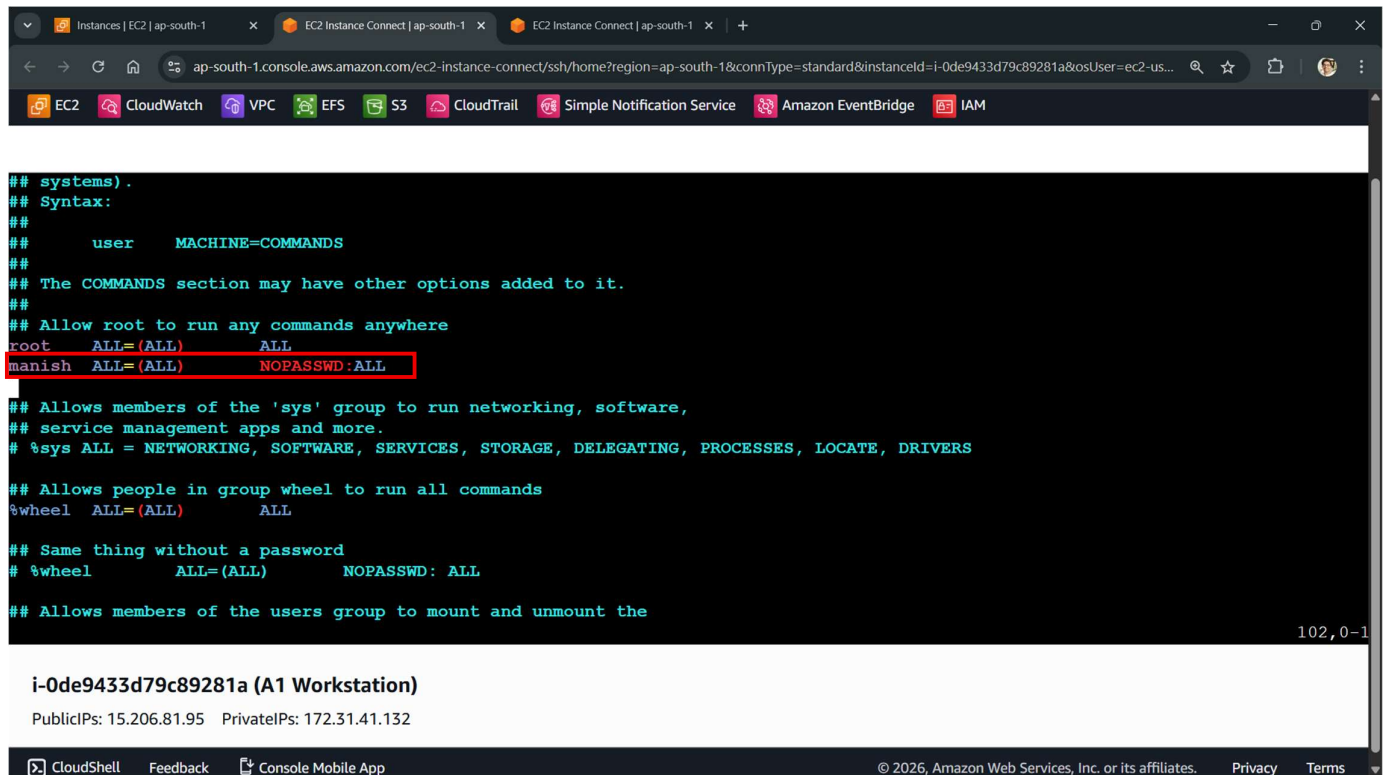
Step 2: Created an user having ‘manish’ and also applied a password to it on both ‘Workstation’ as well as ‘Host’:

The screenshot shows the AWS Management Console for the 'EC2 Instance Connect' page. The terminal output of the 'useradd' and 'passwd' commands on the 'A1 Workstation' instance is displayed. The terminal shows the creation of the 'manish' user and the setting of a password. The terminal output is as follows:

```
Amazon Linux 2023
https://aws.amazon.com/linux/amazon-linux-2023

[ec2-user@ip-172-31-41-132 ~]$ sudo -i
[root@ip-172-31-41-132 ~]# useradd manish
[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 ~]#
```

Step 3: Updated the 'sudoers' file in /etc for assigning superuser privileges to 'manish' user:



```
## 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

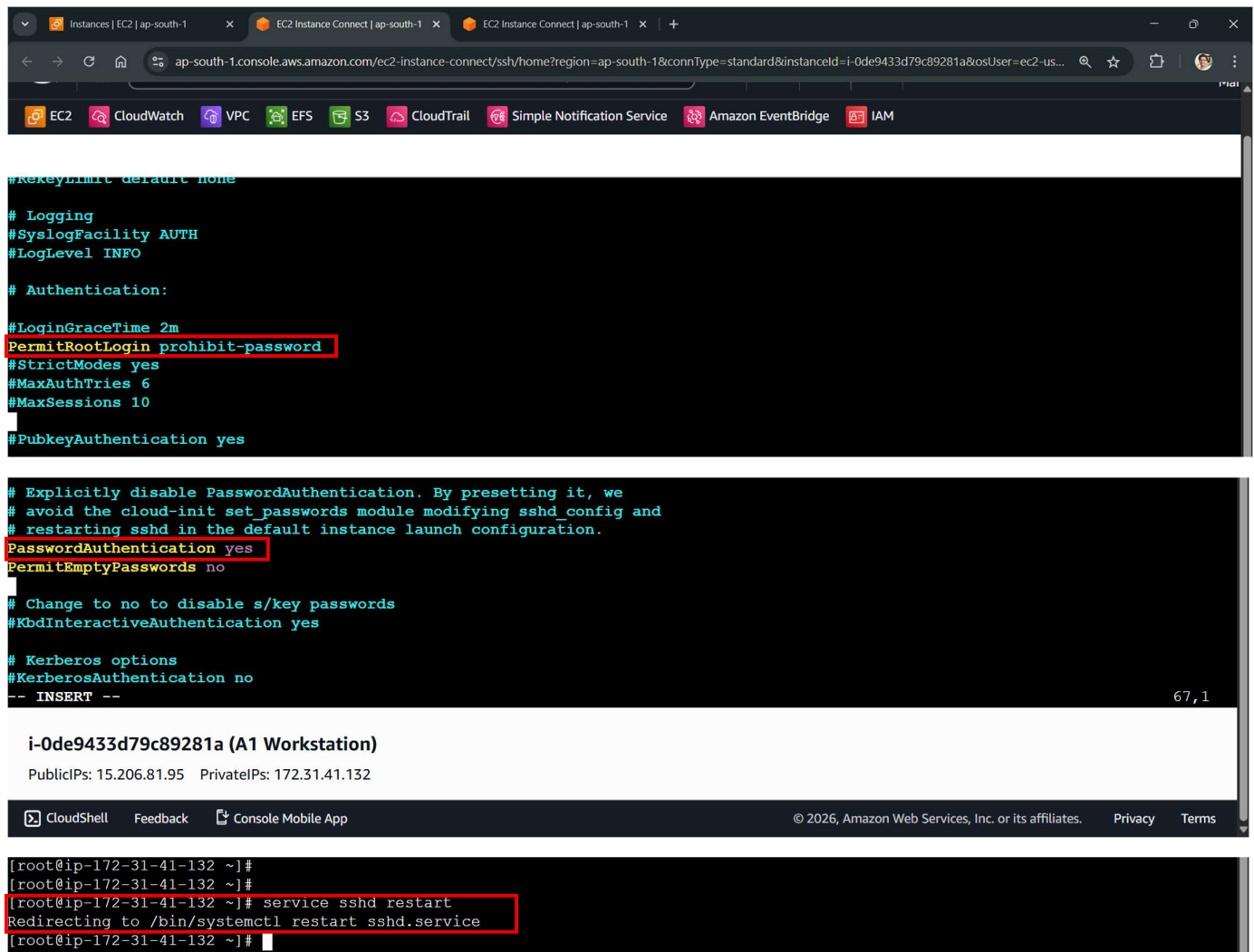
## Same thing without a password
# %wheel    ALL=(ALL)    NOPASSWD: ALL

## Allows members of the users group to mount and unmount the
```

i-0de9433d79c89281a (A1 Workstation)
PublicIPs: 15.206.81.95 PrivateIPs: 172.31.41.132

CloudShell Feedback Console Mobile App © 2026, Amazon Web Services, Inc. or its affiliates. Privacy Terms

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



```
#Kerberos options
#KerberosAuthentication no
-- INSERT --

# 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

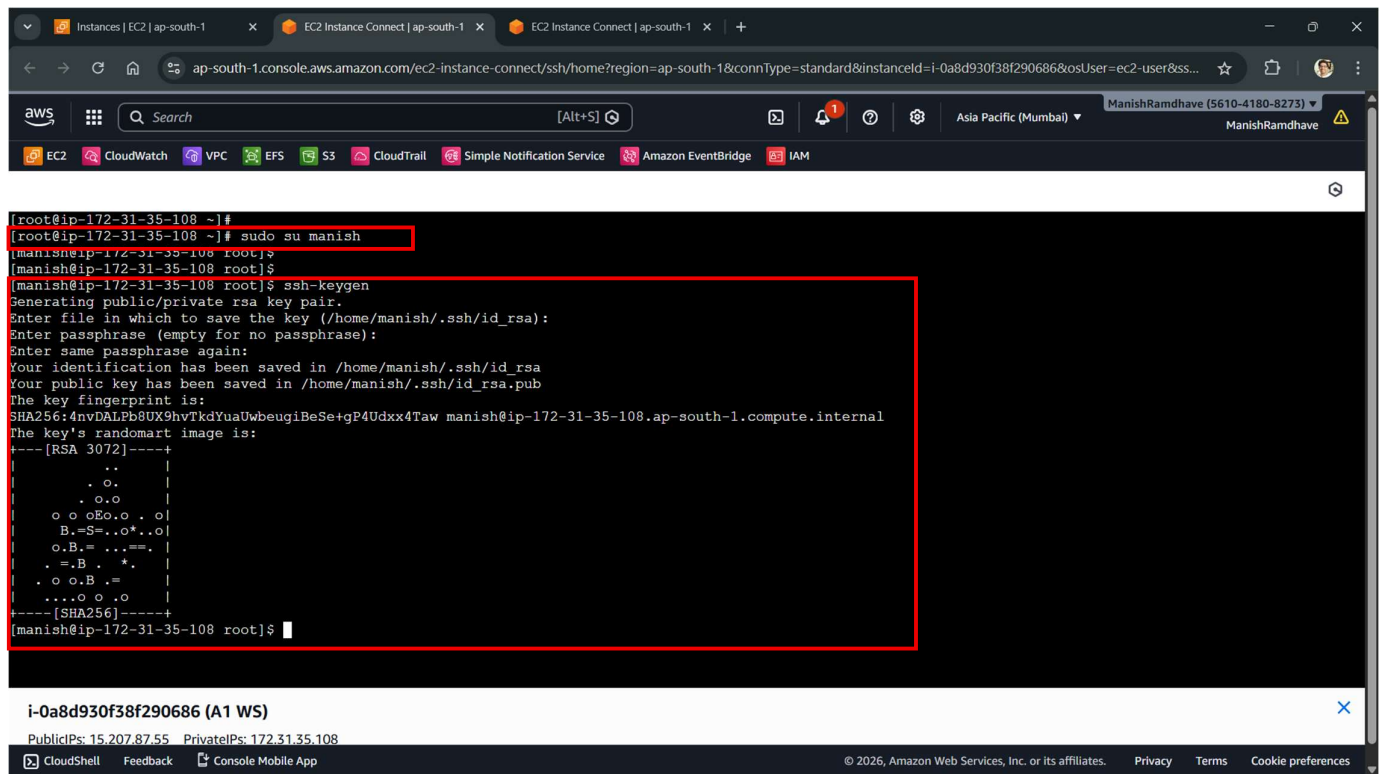
# Kerberos options
#KerberosAuthentication no
-- INSERT --
```

i-0de9433d79c89281a (A1 Workstation)
PublicIPs: 15.206.81.95 PrivateIPs: 172.31.41.132

CloudShell Feedback Console Mobile App © 2026, Amazon Web Services, Inc. or its affiliates. Privacy Terms

```
[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:

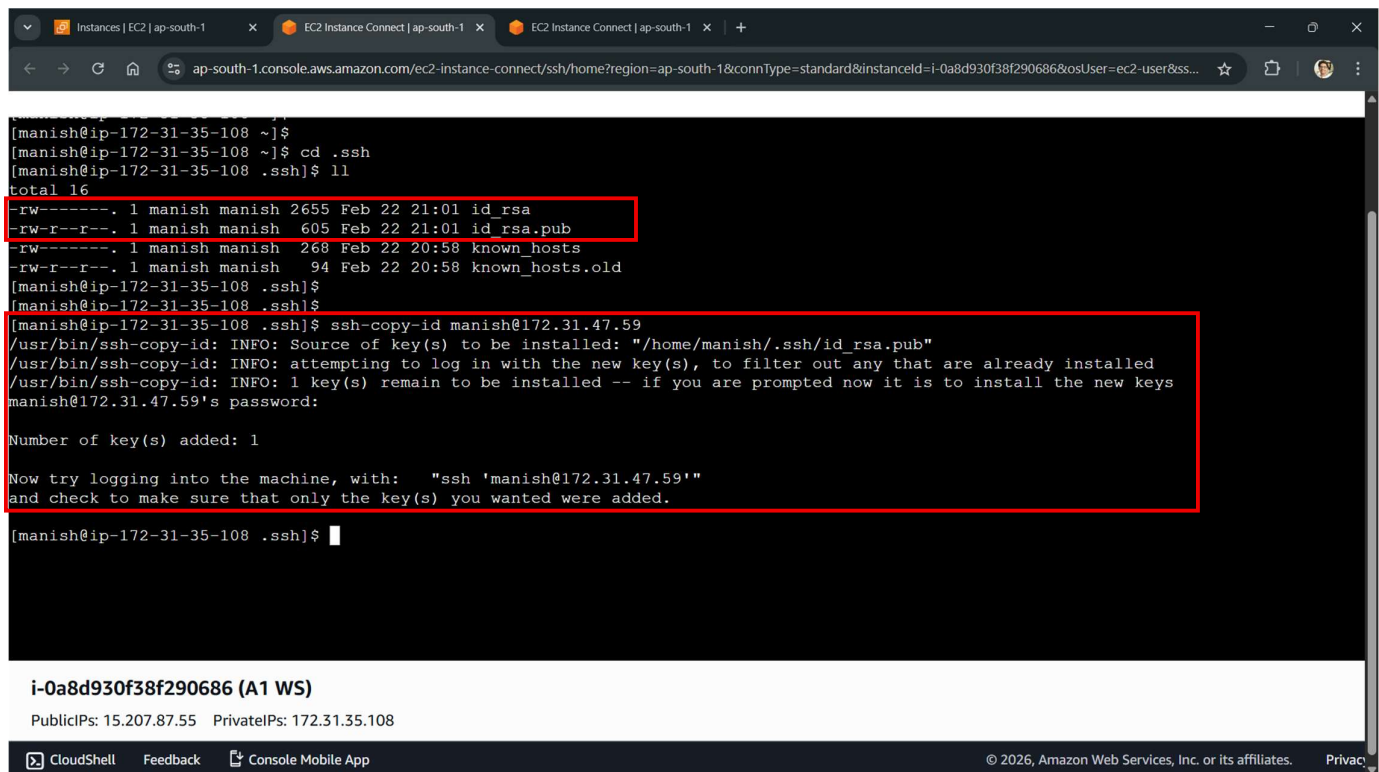


The screenshot shows the AWS Management Console interface with a terminal window open. The terminal output shows the user switching to 'manish' and generating an SSH key pair. The key pair is named 'id_rsa' and is saved in the directory '/home/manish/.ssh/'. The terminal output is as follows:

```
[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]$ 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.oEo.o.o|
|B.=S=..o*..o|
|.B.=...==|
|. =B. *|
|.o.o.B =|
|...o.o.o|
+---[SHA256]-----+
[manish@ip-172-31-35-108 root]$
```

The terminal window is titled 'i-0a8d930f38f290686 (A1 WS)' and shows the public IP as 15.207.87.55 and the private IP as 172.31.35.108. The console footer shows the AWS logo, 'CloudShell', 'Feedback', 'Console Mobile App', and copyright information for Amazon Web Services, Inc. or its affiliates.

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



The screenshot shows the AWS Management Console interface with a terminal window open. The terminal output shows the user copying the SSH key pairs to the host instance. The terminal output is as follows:

```
[manish@ip-172-31-35-108 ~]$
[manish@ip-172-31-35-108 ~]$ cd .ssh
[manish@ip-172-31-35-108 .ssh]$ ll
total 16
-rw----- 1 manish manish 2655 Feb 22 21:01 id_rsa
-rw-r--r-- 1 manish manish 605 Feb 22 21:01 id_rsa.pub
-rw----- 1 manish manish 268 Feb 22 20:58 known_hosts
-rw-r--r-- 1 manish manish 94 Feb 22 20:58 known_hosts.old
[manish@ip-172-31-35-108 .ssh]$
[manish@ip-172-31-35-108 .ssh]$ ssh-copy-id manish@172.31.47.59
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/manish/.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: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to install the new keys
manish@172.31.47.59's password:

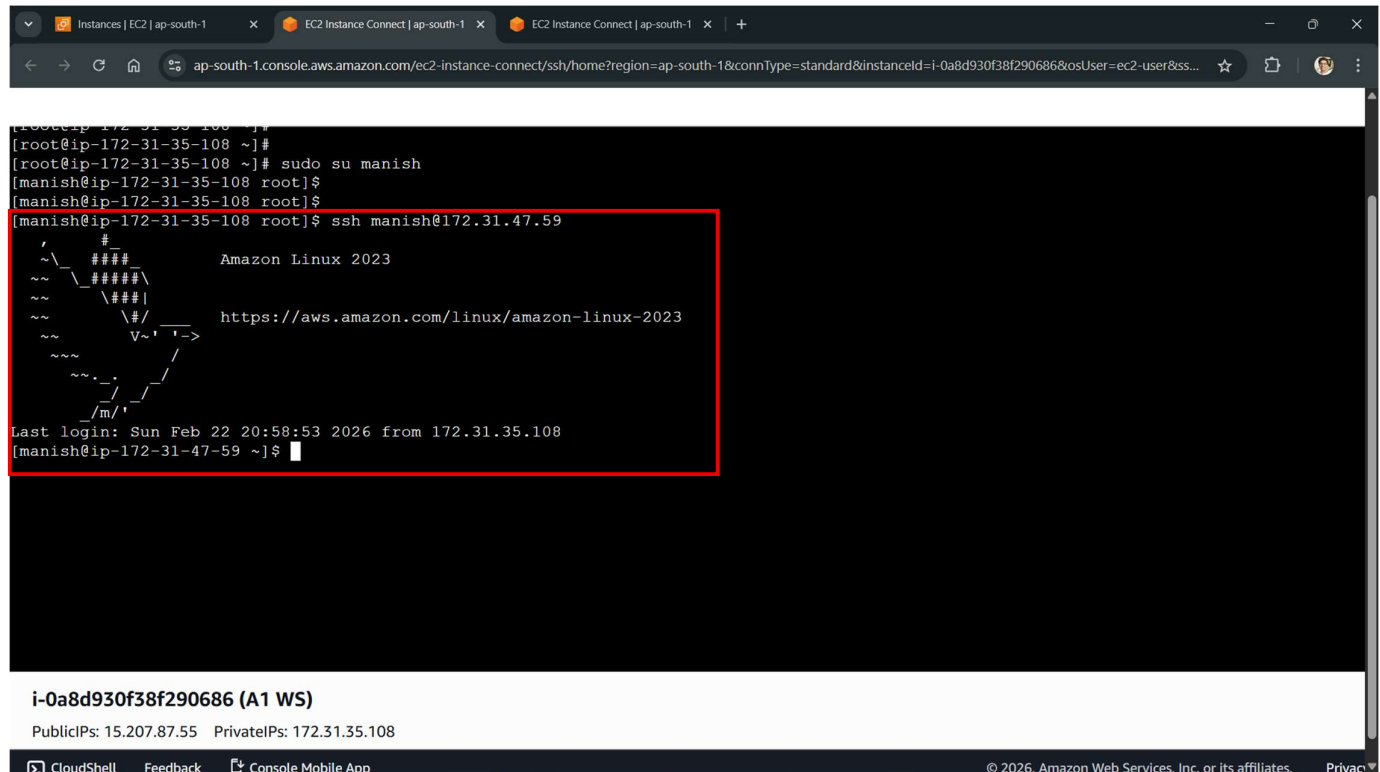
Number of key(s) added: 1

Now try logging into the machine, with: "ssh 'manish@172.31.47.59'"
and check to make sure that only the key(s) you wanted were added.

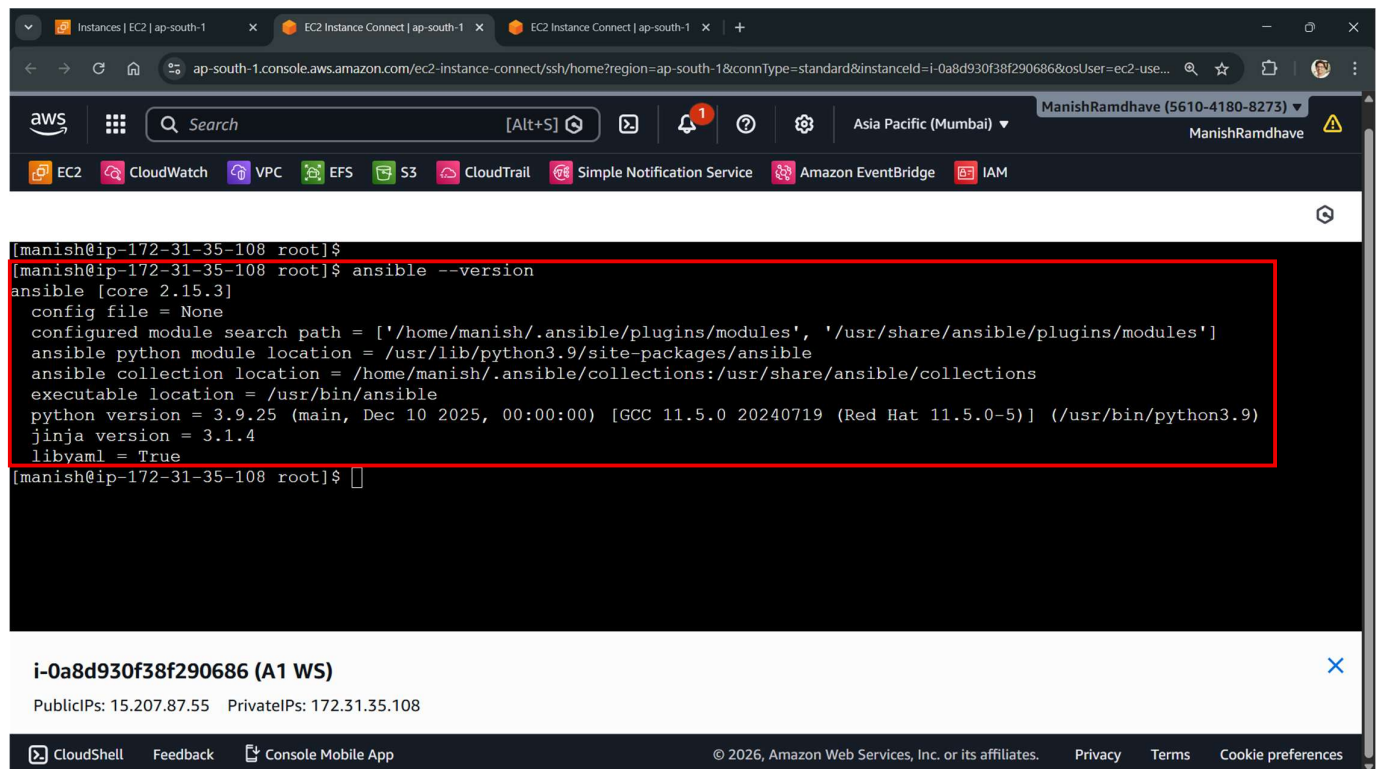
[manish@ip-172-31-35-108 .ssh]$
```

The terminal window is titled 'i-0a8d930f38f290686 (A1 WS)' and shows the public IP as 15.207.87.55 and the private IP as 172.31.35.108. The console footer shows the AWS logo, 'CloudShell', 'Feedback', 'Console Mobile App', and copyright information for Amazon Web Services, Inc. or its affiliates.

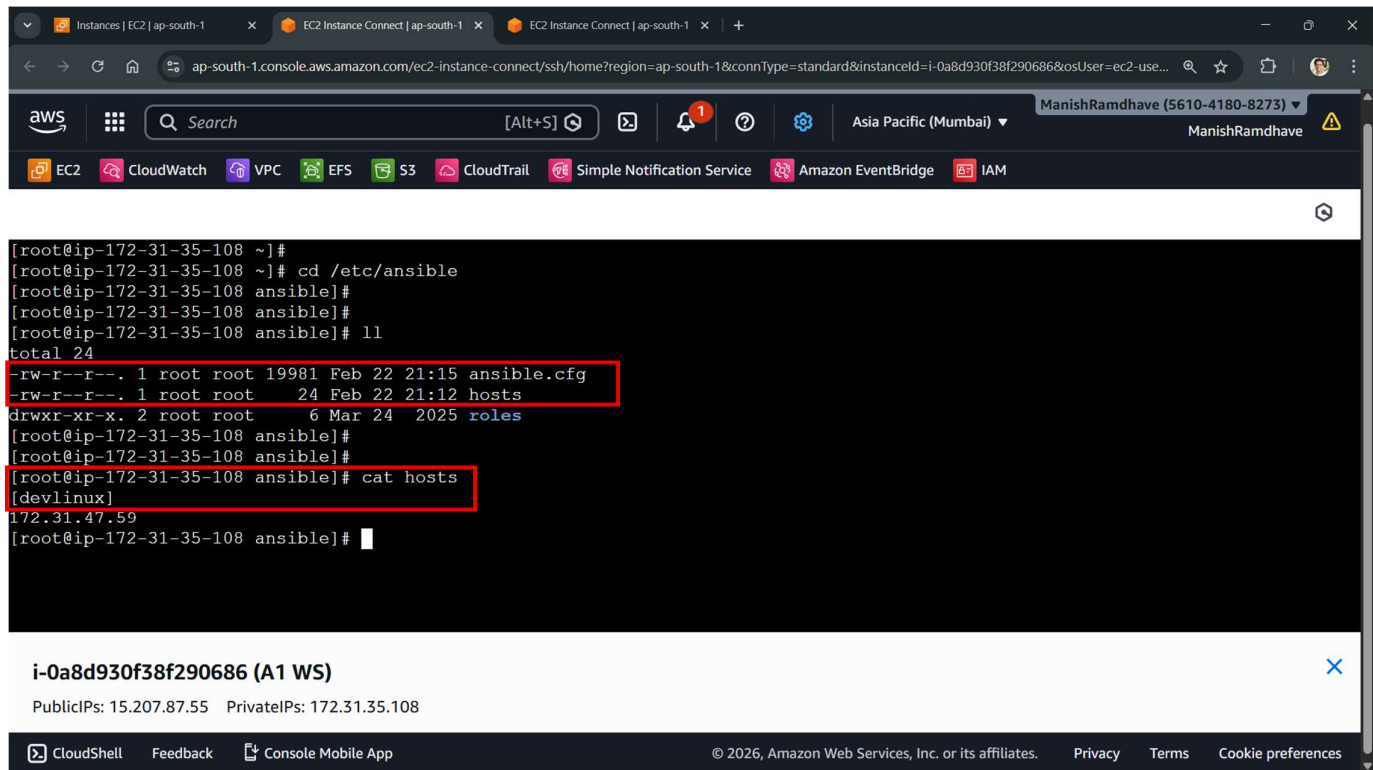
Step 7: Successfully logged in to the **Host** from **Workstation** using SSH and Private IP of Host:



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



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

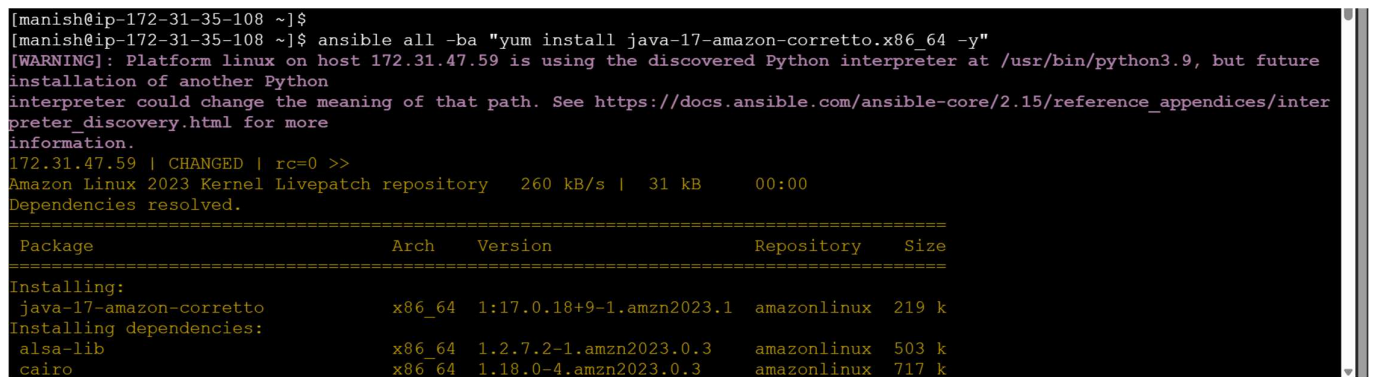


The screenshot shows the AWS Management Console interface with a terminal window open. The terminal output shows the user navigating to `/etc/ansible` and listing files. Two files, `ansible.cfg` and `hosts`, are highlighted with red boxes. The `hosts` file content is also shown, listing the private IP `172.31.47.59`.

```
[root@ip-172-31-35-108 ~]#
[root@ip-172-31-35-108 ~]# cd /etc/ansible
[root@ip-172-31-35-108 ansible]#
[root@ip-172-31-35-108 ansible]#
[root@ip-172-31-35-108 ansible]# ll
total 24
-rw-r--r--. 1 root root 19981 Feb 22 21:15 ansible.cfg
-rw-r--r--. 1 root root 24 Feb 22 21:12 hosts
drwxr-xr-x. 2 root root 6 Mar 24 2025 roles
[root@ip-172-31-35-108 ansible]#
[root@ip-172-31-35-108 ansible]#
[root@ip-172-31-35-108 ansible]# cat hosts
[devlinux]
172.31.47.59
[root@ip-172-31-35-108 ansible]#
```

i-0a8d930f38f290686 (A1 WS)
PublicIPs: 15.207.87.55 PrivateIPs: 172.31.35.108

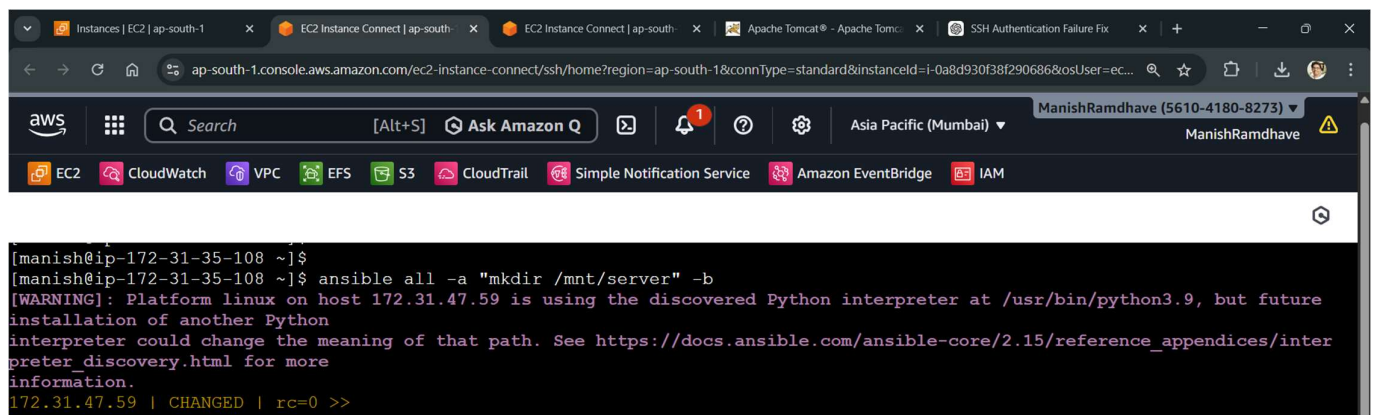
Step 10: Installed **Java-17** on **Host** instance by **Workstation** instance using **adhoc** commands:



The screenshot shows the AWS Management Console interface with a terminal window open. The terminal output shows the user running an Ansible command to install Java 17. The output includes a warning about the Python interpreter and a table of installed packages.

```
[manish@ip-172-31-35-108 ~]$
[manish@ip-172-31-35-108 ~]$ ansible all -ba "yum install java-17-amazon-corretto.x86_64 -y"
[WARNING]: Platform linux on host 172.31.47.59 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/inter
preter_discovery.html for more
information.
172.31.47.59 | CHANGED | rc=0 >>
Amazon Linux 2023 Kernel Livepatch repository 260 kB/s | 31 kB 00:00
Dependencies resolved.
=====
Package Arch Version Repository Size
=====
Installing:
java-17-amazon-corretto x86_64 1:17.0.18+9-1.amzn2023.1 amazonlinux 219 k
Installing dependencies:
alsa-lib x86_64 1.2.7.2-1.amzn2023.0.3 amazonlinux 503 k
cairo x86_64 1.18.0-4.amzn2023.0.3 amazonlinux 717 k
```

Step 11: Made a `'server'` folder in `/mnt/` path using `'adhoc'` commands which will run on `'Host'`:



The screenshot shows the AWS Management Console interface with a terminal window open. The terminal output shows the user running an Ansible command to create a directory `/mnt/server`. The output includes a warning about the Python interpreter and a confirmation message.

```
[manish@ip-172-31-35-108 ~]$
[manish@ip-172-31-35-108 ~]$ ansible all -a "mkdir /mnt/server" -b
[WARNING]: Platform linux on host 172.31.47.59 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/inter
preter_discovery.html for more
information.
172.31.47.59 | CHANGED | rc=0 >>
```

Step 12: Downloaded the Apache-Tomcat-10.1.52 in '/mnt/server/' path:

```
[manish@ip-172-31-35-108 ~]$  
[manish@ip-172-31-35-108 ~]$ ansible all -m shell -a "cd /mnt/server && wget https://dlcdn.apache.org/tomcat/tomcat-10/v10.1.52/bin/apache-tomcat-10.1.52.zip" -b  
[WARNING]: Platform linux on host 172.31.47.59 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.  
172.31.47.59 | CHANGED | rc=0 >>  
--2026-02-22 21:58:33-- https://dlcdn.apache.org/tomcat/tomcat-10/v10.1.52/bin/apache-tomcat-10.1.52.zip  
Resolving dlcdn.apache.org (dlcdn.apache.org)... 151.101.2.132, 2a04:4e42::644  
Connecting to dlcdn.apache.org (dlcdn.apache.org)|151.101.2.132|:443... connected.  
HTTP request sent, awaiting response... 200 OK  
Length: 14869727 (14M) [application/zip]  
Saving to: 'apache-tomcat-10.1.52.zip'
```

Step 13: Extracted the 'apache-tomcat-10.1.52.zip' file in the same path:

```
[manish@ip-172-31-35-108 ~]$  
[manish@ip-172-31-35-108 ~]$ ansible all -m shell -a "cd /mnt/server && unzip apache-tomcat-10.1.52.zip" -b  
[WARNING]: Platform linux on host 172.31.47.59 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.  
172.31.47.59 | CHANGED | rc=0 >>  
Archive:  apache-tomcat-10.1.52.zip  
  creating: apache-tomcat-10.1.52/  
    creating: apache-tomcat-10.1.52/bin/  
    creating: apache-tomcat-10.1.52/conf/  
    creating: apache-tomcat-10.1.52/lib/  
    creating: apache-tomcat-10.1.52/logs/
```

Step 14: After Extraction, deleted the apache-tomcat-10.1.52.zip file:

```
[manish@ip-172-31-35-108 ~]$  
[manish@ip-172-31-35-108 ~]$ ansible all -m shell -a "cd /mnt/server && rm -rf apache-tomcat-10.1.52.zip" -b  
[WARNING]: Platform linux on host 172.31.47.59 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.  
172.31.47.59 | CHANGED | rc=0 >>
```

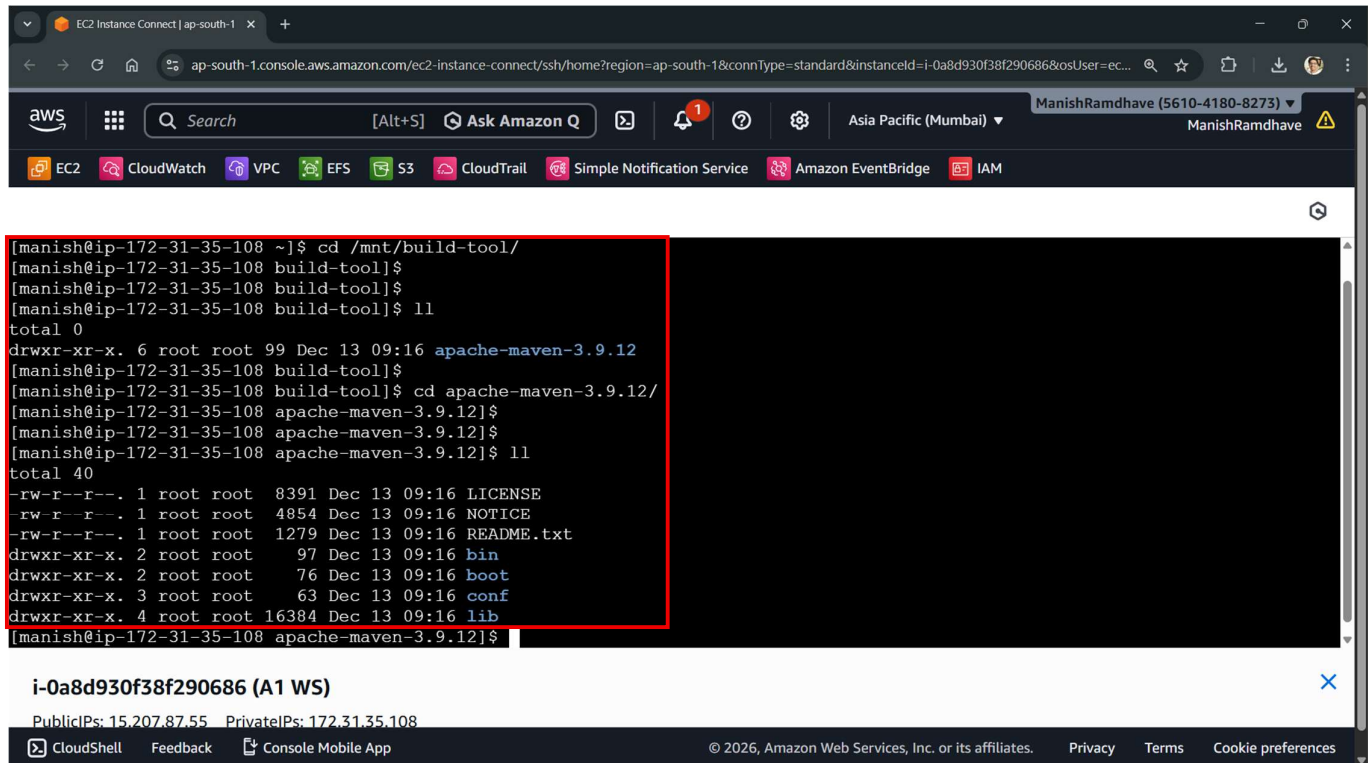
Step 15: Assigned full permission to the 'apache-tomcat-10.1.52' folder (i.e.Tomcat Home):

```
[manish@ip-172-31-35-108 ~]$  
[manish@ip-172-31-35-108 ~]$ ansible all -m shell -a "cd /mnt/server && chmod -R 777 apache-tomcat-10.1.52" -b  
[WARNING]: Platform linux on host 172.31.47.59 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.  
172.31.47.59 | CHANGED | rc=0 >>
```

Step 16: Started the apache-tomcat-10.1.52 using '/bin/startup.sh' script file:

```
[manish@ip-172-31-35-108 ~]$  
[manish@ip-172-31-35-108 ~]$ ansible all -m shell -ba "cd /mnt/server/apache-tomcat-10.1.52/bin && ./startup.sh"  
[WARNING]: Platform linux on host 172.31.47.59 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.  
172.31.47.59 | CHANGED | rc=0 >>  
Using CATALINA_BASE:   /mnt/server/apache-tomcat-10.1.52  
Using CATALINA_HOME:   /mnt/server/apache-tomcat-10.1.52  
Using CATALINA_TMPDIR: /mnt/server/apache-tomcat-10.1.52/temp  
Using JRE_HOME:        /usr  
Using CLASSPATH:        /mnt/server/apache-tomcat-10.1.52/bin/bootstrap.jar:/mnt/server/apache-tomcat-10.1.52/bin/tomcat-juli.jar  
Using CATALINA_OPTS:  
Tomcat started.
```

Step 17: Created a 'build-tool' folder in /mnt/ on Workstation and Downloaded the Apache-Maven-3.9.12 in it:



```
[manish@ip-172-31-35-108 ~]$ cd /mnt/build-tool/
[manish@ip-172-31-35-108 build-tool]$
[manish@ip-172-31-35-108 build-tool]$ ll
total 0
drwxr-xr-x. 6 root root 99 Dec 13 09:16 apache-maven-3.9.12
[manish@ip-172-31-35-108 build-tool]$ cd apache-maven-3.9.12/
[manish@ip-172-31-35-108 apache-maven-3.9.12]$
[manish@ip-172-31-35-108 apache-maven-3.9.12]$ ll
total 40
-rw-r--r--. 1 root root 8391 Dec 13 09:16 LICENSE
-rw-r--r--. 1 root root 4854 Dec 13 09:16 NOTICE
-rw-r--r--. 1 root root 1279 Dec 13 09:16 README.txt
drwxr-xr-x. 2 root root 97 Dec 13 09:16 bin
drwxr-xr-x. 2 root root 76 Dec 13 09:16 boot
drwxr-xr-x. 3 root root 63 Dec 13 09:16 conf
drwxr-xr-x. 4 root root 16384 Dec 13 09:16 lib
[manish@ip-172-31-35-108 apache-maven-3.9.12]$
```

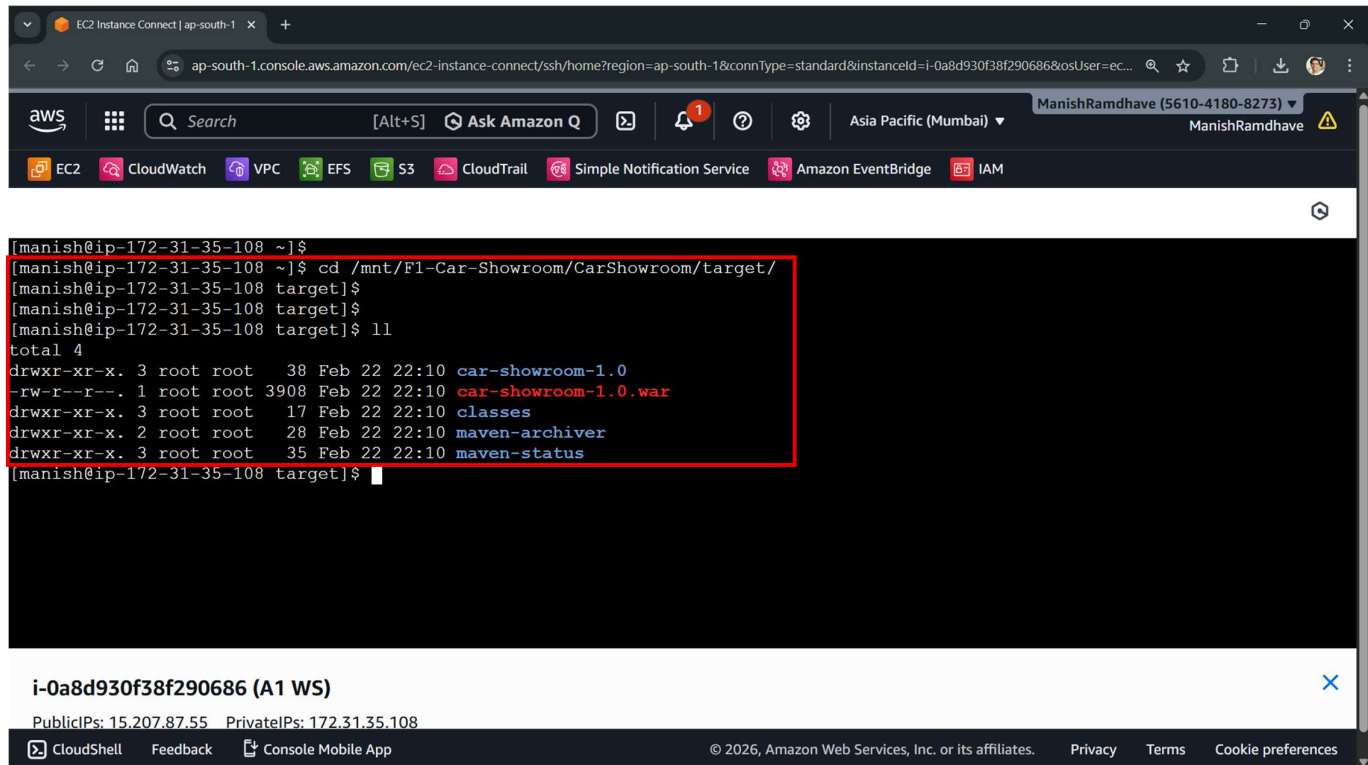
i-Oa8d930f38f290686 (A1 WS)

PublicIPs: 15.207.87.55 PrivateIPs: 172.31.35.108

CloudShell Feedback Console Mobile App

© 2026, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Step 18: Compiled and Packaged the 'car-showroom-1.0.war' file in the target folder:



```
[manish@ip-172-31-35-108 ~]$
[manish@ip-172-31-35-108 ~]$ cd /mnt/F1-Car-Showroom/CarShowroom/target/
[manish@ip-172-31-35-108 target]$
[manish@ip-172-31-35-108 target]$ ll
total 4
drwxr-xr-x. 3 root root 38 Feb 22 22:10 car-showroom-1.0
-rw-r--r--. 1 root root 3908 Feb 22 22:10 car-showroom-1.0.war
drwxr-xr-x. 3 root root 17 Feb 22 22:10 classes
drwxr-xr-x. 2 root root 28 Feb 22 22:10 maven-archiver
drwxr-xr-x. 3 root root 35 Feb 22 22:10 maven-status
[manish@ip-172-31-35-108 target]$
```

i-Oa8d930f38f290686 (A1 WS)

PublicIPs: 15.207.87.55 PrivateIPs: 172.31.35.108

CloudShell Feedback Console Mobile App

© 2026, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Step 19: Copied the ‘car-showroom-1.0’ application to Tomcat Home’s /webapps folder and it gets deployed as we have kept tomcat server started:

```
[manish@ip-172-31-35-108 ~]$
[manish@ip-172-31-35-108 ~]$ ansible all -m copy -a "src=/mnt/ansible all -m copy -a "src=/mnt/F1-Car-Showroom/CarShowroom/target/car-showroom-1.0.war dest=/mnt/server/apache-tomcat-10.1.52/webapps/" -b
[WARNING]: Platform linux on host 172.31.47.59 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.
172.31.47.59 | CHANGED => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3.9"
  },
  "changed": true,
  "checksum": "66ebd081e9b3531f3dd042a71b41ad1e852f129a",
  "dest": "/mnt/server/apache-tomcat-10.1.52/webapps/car-showroom-1.0.war",
  "gid": 1001,
  "group": "manish",
  "md5sum": "d5128f2bc9a0f60f1e015de8e4546dd7",
  "mode": "0664",
```

Results:

We have successfully deployed the ‘car-showroom-1.0’ application using ansible ‘ad hoc command’ and hosted the same application using Host Public IP and Tomcat Server Port No.8080:

