

Module 5: Ansible

Assignment



ANSIBLE

Module 5: Ansible Assignment - 1

Tasks To Be Performed:

1. Setup Ansible cluster with 3 nodes
2. On slave 1 install Java
3. On slave 2 install MySQL server

Do the above tasks using Ansible Playbooks

aws

Search [Alt+S]

EC2 EC2 Image Builder S3 IAM VPC CloudFormation Lambda Cognito API Gateway CloudWatch

Launch an instance

Launch an instance

Launch an instance

Info

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

Name and tags

Info

Name

Ansible

Add additional tags

Application and OS Images (Amazon Machine Image)

Info

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

Search our full catalog including 1000s of application and OS images

Recents

My AMIs

Quick Start

Amazon Linux

aws

macOS

Mac

Ubuntu

ubuntu

Windows

Microsoft

Red Hat

Red Hat

SUSE Linux

SUSE

Debian

debian

Browse more AMIs

Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)

Free tier eligible

Ubuntu Server 22.04 LTS (HVM), SSD Volume Type

ami-09b0a86a2c84101e1 (64-bit (x86)) / ami-0a87daabd88e93b1f (64-bit (Arm))

Virtualization: hvm ENA enabled: true Root device type: ebs

Summary

Number of instances

Info

3

When launching more than 1 instance, consider EC2 Auto Scaling

Software image (AMI)

Canonical, Ubuntu, 22.04 LTS, ...read more

ami-09b0a86a2c84101e1

Virtual server type (instance type)

t2.micro

Firewall (security group)

default

Storage (volumes)

1 volume(s) - 8 GiB

Free tier:

In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 750 hours of public IPv4 address usage per month, 30 GiB of EBS storage, 2 million ...

Cancel

Launch instance

Preview code



3 Instance 1- master & 2 slave











The screenshot displays the AWS Management Console interface for the 'Instances' page. The top navigation bar shows the AWS logo, a search bar, and various service icons. The left sidebar contains navigation links for Dashboard, EC2 Global View, Events, and a list of services including EC2, EC2 Image Builder, S3, IAM, VPC, CloudFormation, Lambda, Cognito, API Gateway, and CloudWatch. The main content area is titled 'Instances (2/3)' and shows a table of three EC2 instances:

Name	Instance ID	Instance state	Instance type	Status check	Alarm status
Ansible-Master	i-0545735bd53b33955	Running	t2.micro	Initializing	View alarms +
Ansible-Slave1	i-04e8dfdf40c13bff	Running	t2.micro	Initializing	View alarms +
Ansible-Slave2	i-0f28093ff91a518a8	Running	t2.micro	Initializing	View alarms +

Below the table, the 'Monitoring' section is visible, showing four graphs for CPU utilization, network in/out, and network packets. The graphs are currently displaying 'No data available' and 'No unit'.

Ansible Master Slave

  [Alt+S]

 EC2  EC2 Image Builder  S3  IAM  VPC  CloudFormation  Lambda  Cognito  API Gateway  CloudWatch

⌕

```
* Documentation:  https://help.ubuntu.com
* Management:    https://landscape.canonical.com
* Support:        https://ubuntu.com/pro

System information as of Fri Jan 10 02:36:06 UTC 2025

System load:  0.1          Processes:            108
Usage of /:   21.1% of 7.57GB Users logged in:        0
Memory usage: 21%          IPv4 address for eth0: 172.31.12.86
Swap usage:   0%

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-172-31-12-86:~$
```

i-0545735bd53b33955 (Ansible-Master)

PublicIPs: 13.233.144.190 PrivateIPs: 172.31.12.86

Ansible Slave-1

  [Alt+S]     Asia Pacific (Mumbai) ▾ prakash24 ▾

EC2 EC2 Image Builder S3 IAM VPC CloudFormation Lambda Cognito API Gateway CloudWatch

```
* Documentation:  https://help.ubuntu.com
* Management:    https://landscape.canonical.com
* Support:        https://ubuntu.com/pro

System information as of Fri Jan 10 02:36:12 UTC 2025

System load:  0.27          Processes:            107
Usage of /:   21.1% of 7.57GB Users logged in:          0
Memory usage: 22%          IPv4 address for eth0: 172.31.2.175
Swap usage:   0%

Expanded Security Maintenance for Applications is not enabled.

) 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-172-31-2-175:~$
```

i-04e8dfdff40c13bff (Ansible-Slave1)
PublicIPs: 43.204.219.17 PrivateIPs: 172.31.2.175

Ansible Slave-2

  [Alt+S]

 EC2  EC2 Image Builder  S3  IAM  VPC  CloudFormation  Lambda  Cognito  API Gateway  CloudWatch

⌕

```
* Documentation:  https://help.ubuntu.com
* Management:    https://landscape.canonical.com
* Support:        https://ubuntu.com/pro

System information as of Fri Jan 10 02:36:18 UTC 2025

System load:  0.05          Processes:            109
Usage of /:   21.1% of 7.57GB Users logged in:          0
Memory usage: 22%          IPv4 address for eth0: 172.31.3.113
Swap usage:   0%

Expanded Security Maintenance for Applications is not enabled.

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.

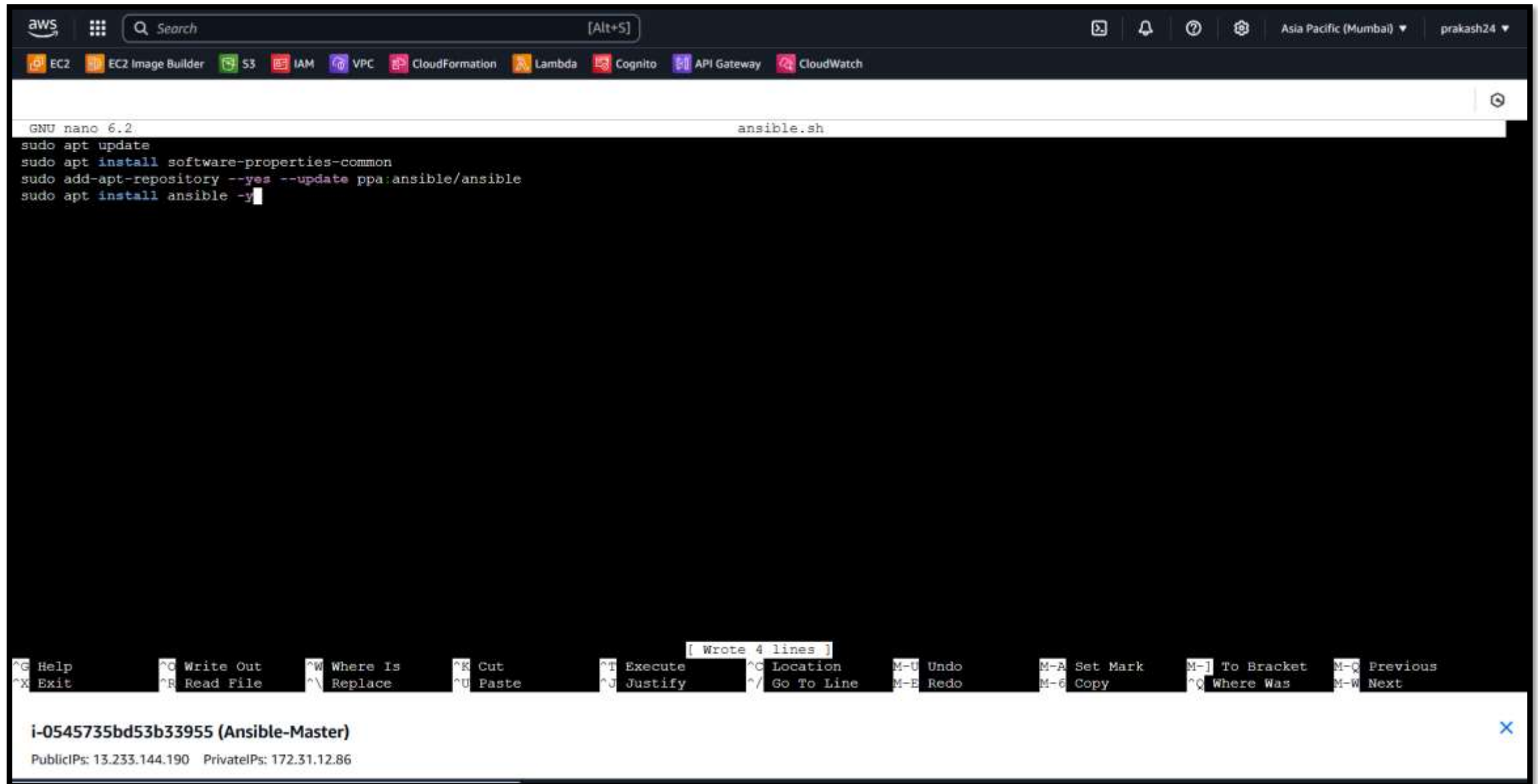
ubuntu@ip-172-31-3-113:~$
```

✕

i-Of28093ff91a518a8 (Ansible-Slave2)

PublicIPs: 13.235.81.228 PrivateIPs: 172.31.3.113

Installing ansible in master instance



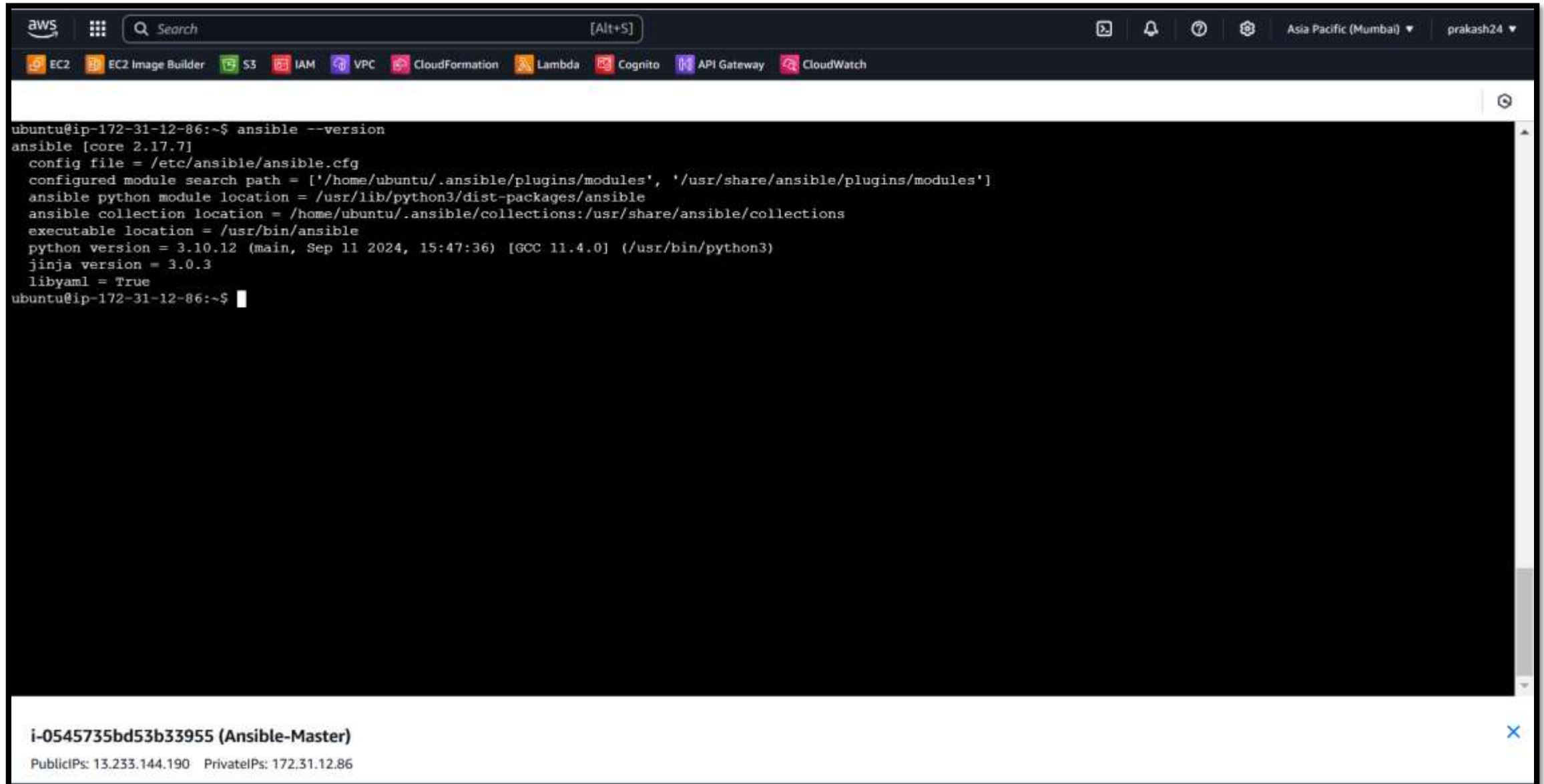
The screenshot shows the AWS Management Console interface. At the top, there's a navigation bar with the AWS logo, a search bar, and various service icons like EC2, EC2 Image Builder, S3, IAM, VPC, CloudFormation, Lambda, Cognito, API Gateway, and CloudWatch. The region is set to Asia Pacific (Mumbai) and the user is prakash24.

The main content area displays a terminal window for an EC2 instance named **i-0545735bd53b33955 (Ansible-Master)**. The terminal shows the following commands being executed in a nano editor:

```
GNU nano 6.2 ansible.sh
sudo apt update
sudo apt install software-properties-common
sudo add-apt-repository --yes --update ppa:ansible/ansible
sudo apt install ansible -y
```

Below the terminal window, there's a status bar showing the instance's public and private IP addresses: PublicIPs: 13.233.144.190 PrivateIPs: 172.31.12.86.

Validate ansible installation in master instance

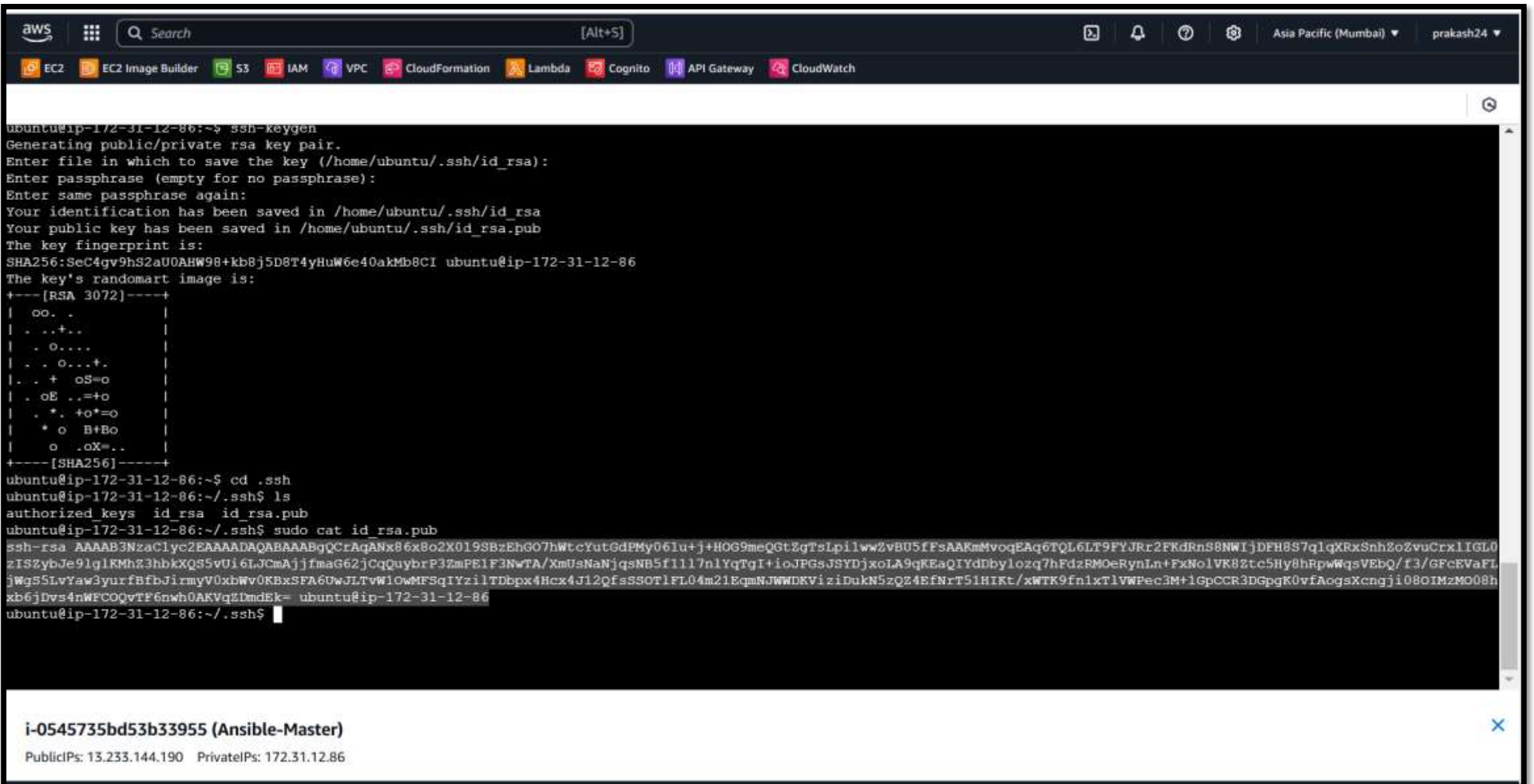


The screenshot shows an AWS CloudShell interface. At the top, there's a navigation bar with the AWS logo, a search bar, and a list of services: EC2, EC2 Image Builder, S3, IAM, VPC, CloudFormation, Lambda, Cognito, API Gateway, and CloudWatch. The region is set to Asia Pacific (Mumbai) and the user is prakash24. The terminal window shows the command `ansible --version` being executed on an Ubuntu instance. The output displays the Ansible core version (2.17.7) and various configuration details including the config file path, module search paths, Python module location, collection locations, executable location, Python version (3.10.12), Jinja version (3.0.3), and the libyaml status (True).

```
ubuntu@ip-172-31-12-86:~$ ansible --version
ansible [core 2.17.7]
  config file = /etc/ansible/ansible.cfg
  configured module search path = ['/home/ubuntu/.ansible/plugins/modules', '/usr/share/ansible/plugins/modules']
  ansible python module location = /usr/lib/python3/dist-packages/ansible
  ansible collection location = /home/ubuntu/.ansible/collections:/usr/share/ansible/collections
  executable location = /usr/bin/ansible
  python version = 3.10.12 (main, Sep 11 2024, 15:47:36) [GCC 11.4.0] (/usr/bin/python3)
  jinja version = 3.0.3
  libyaml = True
ubuntu@ip-172-31-12-86:~$
```

i-0545735bd53b33955 (Ansible-Master)
PublicIPs: 13.233.144.190 PrivateIPs: 172.31.12.86

Generating SSH Key for connect slave nodes



The screenshot shows an AWS console terminal window with the following content:

```
aws
Search [Alt+S]
EC2 EC2 Image Builder S3 IAM VPC CloudFormation Lambda Cognito API Gateway CloudWatch

ubuntu@ip-172-31-12-86:~$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/ubuntu/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/ubuntu/.ssh/id_rsa
Your public key has been saved in /home/ubuntu/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:SeC4gv9hS2aU0AHW98+kb8j5D8T4yHuW6e40akMb8CI ubuntu@ip-172-31-12-86
The key's randomart image is:
+---[RSA 3072]-----+
|  oo. .                |
| . ..+..              |
| . o....              |
| . . O...+.           |
|... + oS=o            |
| . oE ..=+o           |
| . *. +o*=o           |
| * o B+Bo             |
| o .oX=..             |
+---[SHA256]-----+
ubuntu@ip-172-31-12-86:~$ cd .ssh
ubuntu@ip-172-31-12-86:~/.ssh$ ls
authorized_keys id_rsa id_rsa.pub
ubuntu@ip-172-31-12-86:~/.ssh$ sudo cat id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQGCrAqANx86x8o2X019SBzEhGO7hWtcYutGdPMY06lu+j+HOG9meQGTZgTsLpilwwZvBU5IFsAAKmMvoqEAq6TQL6LT9FYJRr2FKdRnS8NWIjDPH8S7qlqXRxSnhZoZvuCrxlIGL0
zISZybJe9lg1KMhZ3hbKXQS5vUi6LJCmAjjfmaG62jCqQuybrP3ZmPE1F3NwTA/XmUsNaNjqsnB5fll17nlYqTgI+ioJPGsJSYDjxoLA9qKEaQIYdDbylozq7hFdZRMoeRynLn+FxNo1VK8Ztc5Hy8hRpwWqsVEbQ/f3/GFcEvaFL
jWgS5LvYaw3yurfBfbJirmYV0xbWv0KBxSFA6UwJLTvW1owMFSqIYzilTDbpX4Hcx4Jl2QfsSSOTlFL04m21EqmNJWWDKviziDukN5zQZ4EfNrT51HIKt/xWTK9fn1xTlVWPec3M+lGpCCR3DGpgK0vfAogsXcngji08oIMzMO08h
xb6jDvs4nWFCOQvTF6nwh0AKVqZDmdEk= ubuntu@ip-172-31-12-86
ubuntu@ip-172-31-12-86:~/.ssh$
```

i-0545735bd53b33955 (Ansible-Master)
PublicIPs: 13.233.144.190 PrivateIPs: 172.31.12.86

Mapping SSH Key from master node to Slave1 node

aws

Search

[Alt+S]

Asia Pacific (Mumbai)

prakash24

EC2

EC2 Image Builder

S3

IAM

VPC

CloudFormation

Lambda

Cognito

API Gateway

CloudWatch

GNU nano 6.2

.ssh/authorized keys

ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQDJ0KFUzadAgyQocDiG0PHC5SB681dXDnVDOy2NLSlCelIAdi/GdiLiEbQlqvMEyx1xjBKRAN9MEkvnd1X2SsPuxmsisFqhYbsvz1St4tb+RGcQPCerqcxBC0AElphSqAvXuPnp>

ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQgQCrAqANx86x8o2X019SBzEhGO7hWtcYutGdPMY061u+j+HOG9meQGT2gTsLpilwwZvBU5fFsAAKmMvoqEAq6TQL6LT9FYJRr2FKdRnS8NWIjDFH8S7q1qXRxSnhZoZvuCrXlIGL>

Read 2 lines

Help Write Out Where Is Cut Execute Location Undo Set Mark To Bracket Previous

Exit Read File Replace Paste Justify Go To Line Redo Copy Where Was Next

i-04e8dfdff40c13bff (Ansible-Slave1)

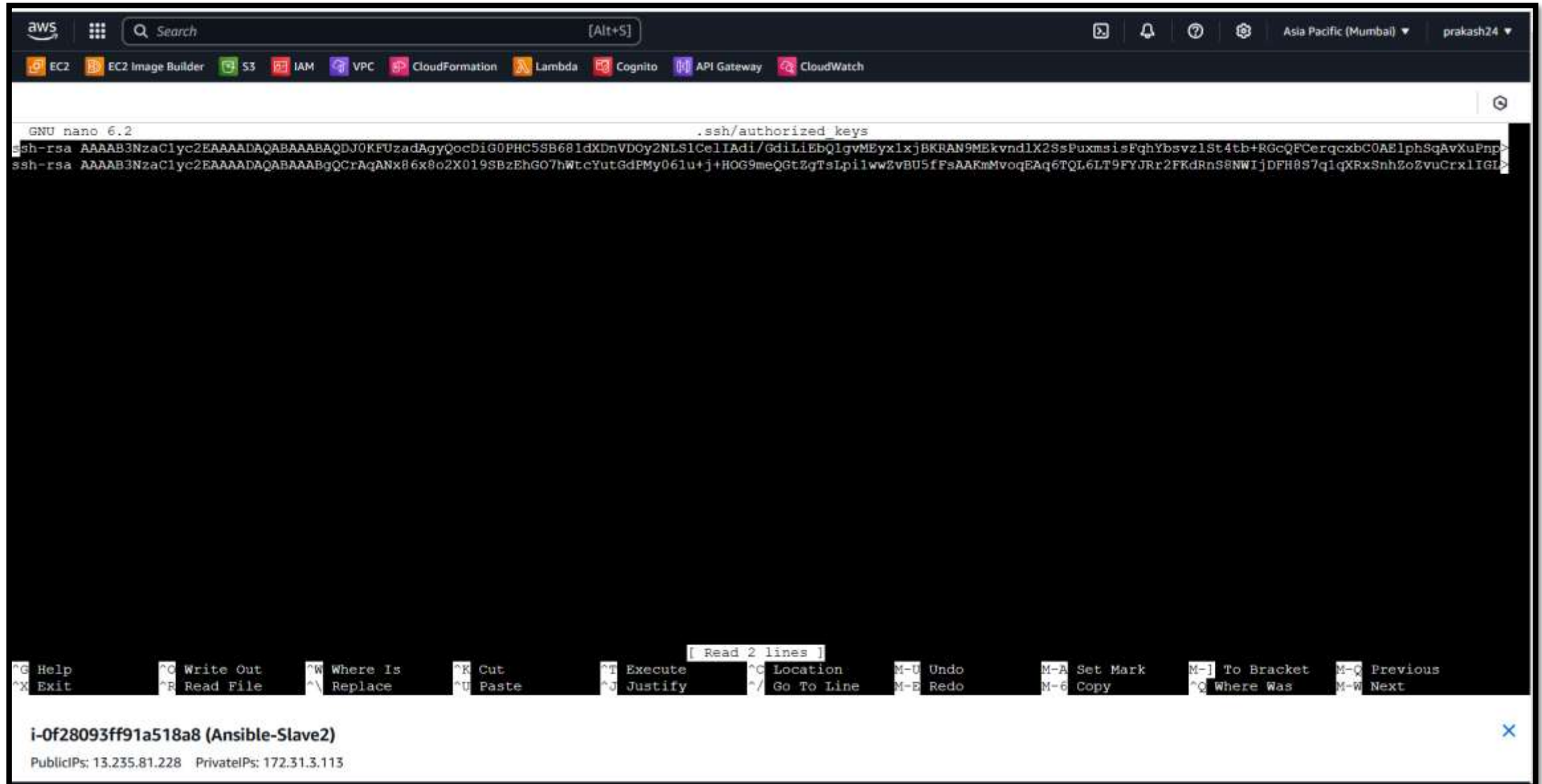
PublicIPs: 43.204.219.17 PrivateIPs: 172.31.2.175

CloudShell

Feedback

© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Mapping SSH Key from master node to Slave2 node

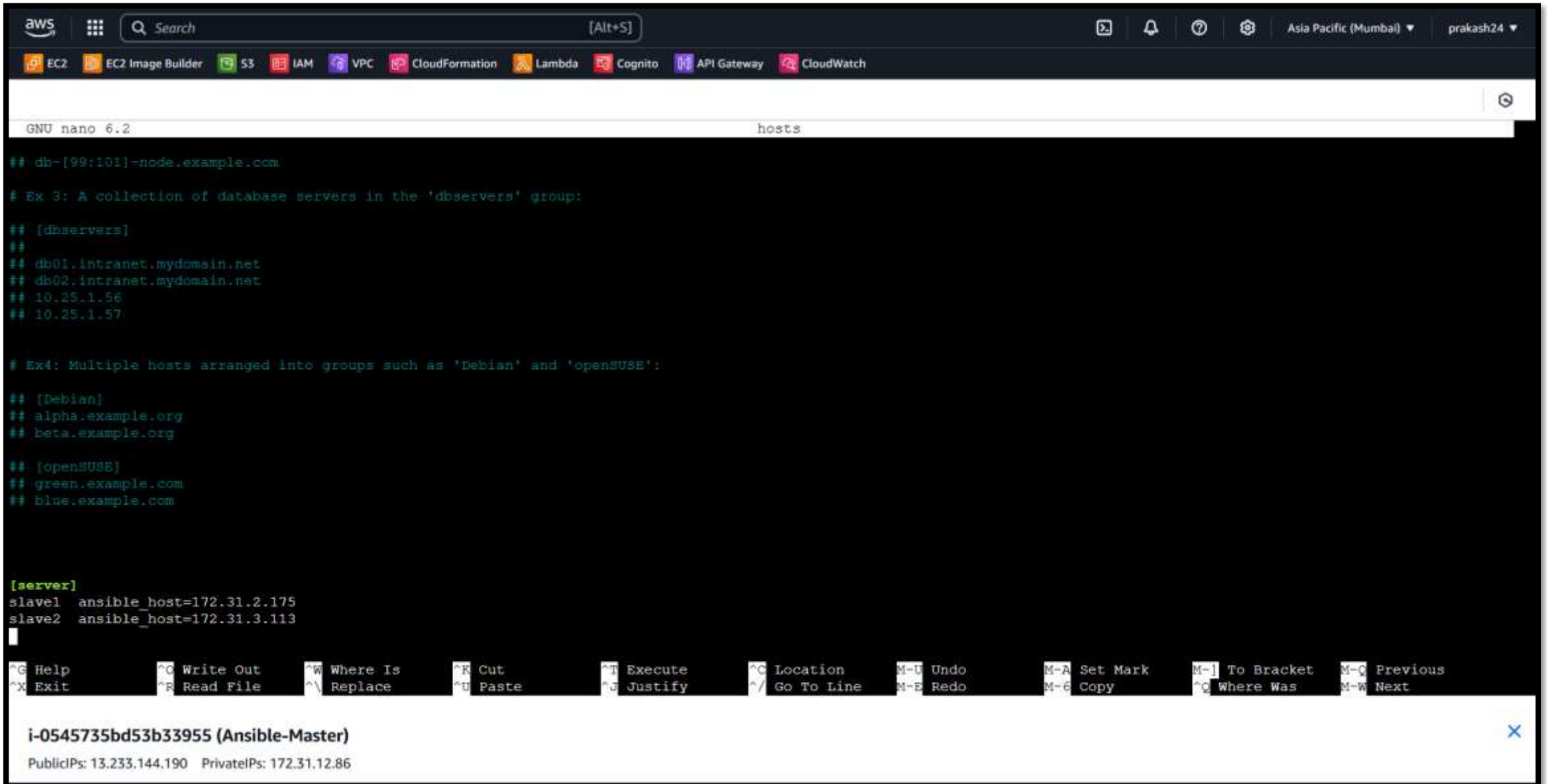


The screenshot shows the AWS Management Console interface. At the top, there's a search bar and navigation icons. Below that, a row of service icons includes EC2, EC2 Image Builder, S3, IAM, VPC, CloudFormation, Lambda, Cognito, API Gateway, and CloudWatch. The main area displays a terminal window for an EC2 instance named **i-0f28093ff91a518a8 (Ansible-Slave2)**. The terminal is running the `cat` command to view the contents of the `.ssh/authorized_keys` file. The output shows two SSH public keys, both starting with `ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQD`. The terminal window has a status bar at the bottom with various keyboard shortcuts like `^G Help`, `^O Write Out`, etc. The instance details at the bottom indicate it's in the **Asia Pacific (Mumbai)** region with Public IPs: 13.235.81.228 and Private IPs: 172.31.3.113.

```
GNU nano 6.2 .ssh/authorized_keys
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQDJOKEFuzadAgyQocDiG0PHC5SB681dXDnVDOy2NLS1CellIAdi/GdiLiEbQlgvMEyx1xjBKRA9MEkvnd1X2SsPuxmsisFqhYbsvz1St4tb+RGcQFCerqcxBC0AE1phSqAvXuPng>
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQgQCraQANx86x8o2X019SBzEhG07hWtcYutGdPMY061u+j+HOG9meQGtZgTsLpilwwZvBU5fFsAAKmMvoqEAq6TQL6LT9FYJRr2FKdRnS8NWIjDFH8S7q1qXRxSnhZoZvuCrxlIGL>
```

i-0f28093ff91a518a8 (Ansible-Slave2)
PublicIPs: 13.235.81.228 PrivateIPs: 172.31.3.113

Binding slave1 & Slave2 on hosts using master node



The screenshot shows the AWS Management Console interface. At the top, there's a navigation bar with the AWS logo, a search bar, and various service icons (EC2, EC2 Image Builder, S3, IAM, VPC, CloudFormation, Lambda, Cognito, API Gateway, CloudWatch). The user is logged in as 'prakash24' in the 'Asia Pacific (Mumbai)' region. Below the navigation bar, a terminal window titled 'hosts' is open, showing the GNU nano 6.2 editor. The editor contains the following text:

```
## db-[99:101]-node.example.com

# Ex 3: A collection of database servers in the 'dbservers' group:

## [dbservers]
##
## db01.intranet.mydomain.net
## db02.intranet.mydomain.net
## 10.25.1.56
## 10.25.1.57

# Ex4: Multiple hosts arranged into groups such as 'Debian' and 'openSUSE':

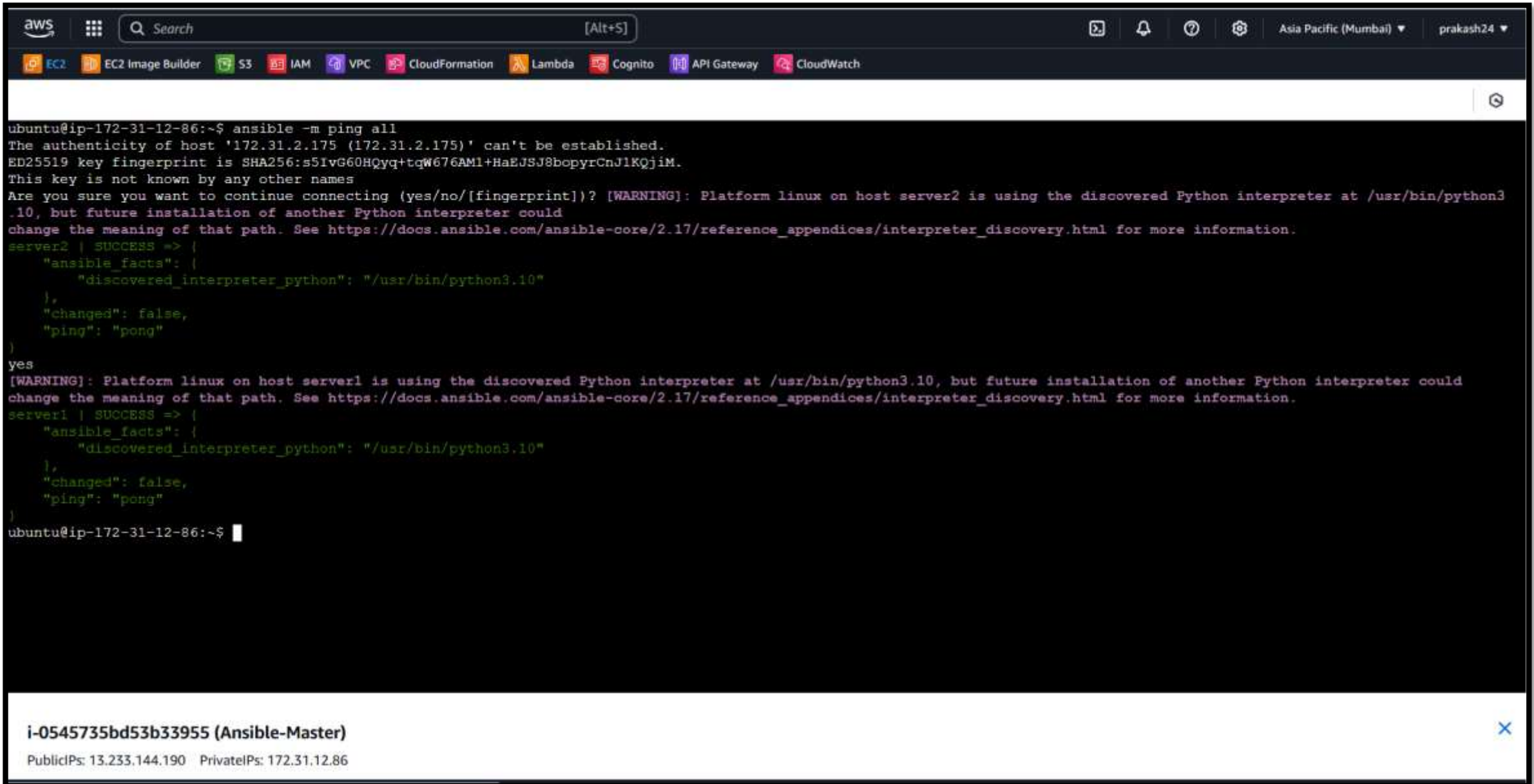
## [Debian]
## alpha.example.org
## beta.example.org

## [openSUSE]
## green.example.com
## blue.example.com

[server]
slave1 ansible_host=172.31.2.175
slave2 ansible_host=172.31.3.113
```

At the bottom of the terminal window, there's a status bar with various keyboard shortcuts for nano editor functions. Below the terminal window, there's a notification box for the instance 'i-0545735bd53b33955 (Ansible-Master)' with public IP 13.233.144.190 and private IP 172.31.12.86.

Validating slave1 & Slave2 using “ansible –m ping all” command , master node able to reach slave node or not



```
aws
[Alt+S]
EC2 EC2 Image Builder S3 IAM VPC CloudFormation Lambda Cognito API Gateway CloudWatch
Asia Pacific (Mumbai) prakash24

ubuntu@ip-172-31-12-86:~$ ansible -m ping all
The authenticity of host '172.31.2.175 (172.31.2.175)' can't be established.
ED25519 key fingerprint is SHA256:s5IvG60HQyq+tgW676AM1+HaEJSJ8bopyrCnJlKQjim.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? [WARNING]: Platform linux on host server2 is using the discovered Python interpreter at /usr/bin/python3
.10, but future installation of another Python interpreter could
change the meaning of that path. See https://docs.ansible.com/ansible-core/2.17/reference_appendices/interpreter_discovery.html for more information.
server2 | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3.10"
  },
  "changed": false,
  "ping": "pong"
}
yes
[WARNING]: Platform linux on host server1 is using the discovered Python interpreter at /usr/bin/python3.10, but future installation of another Python interpreter could
change the meaning of that path. See https://docs.ansible.com/ansible-core/2.17/reference_appendices/interpreter_discovery.html for more information.
server1 | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3.10"
  },
  "changed": false,
  "ping": "pong"
}
ubuntu@ip-172-31-12-86:~$
```

i-0545735bd53b33955 (Ansible-Master)
PublicIPs: 13.233.144.190 PrivateIPs: 172.31.12.86

Creating a playbook yml file to install Java & MySQL Server

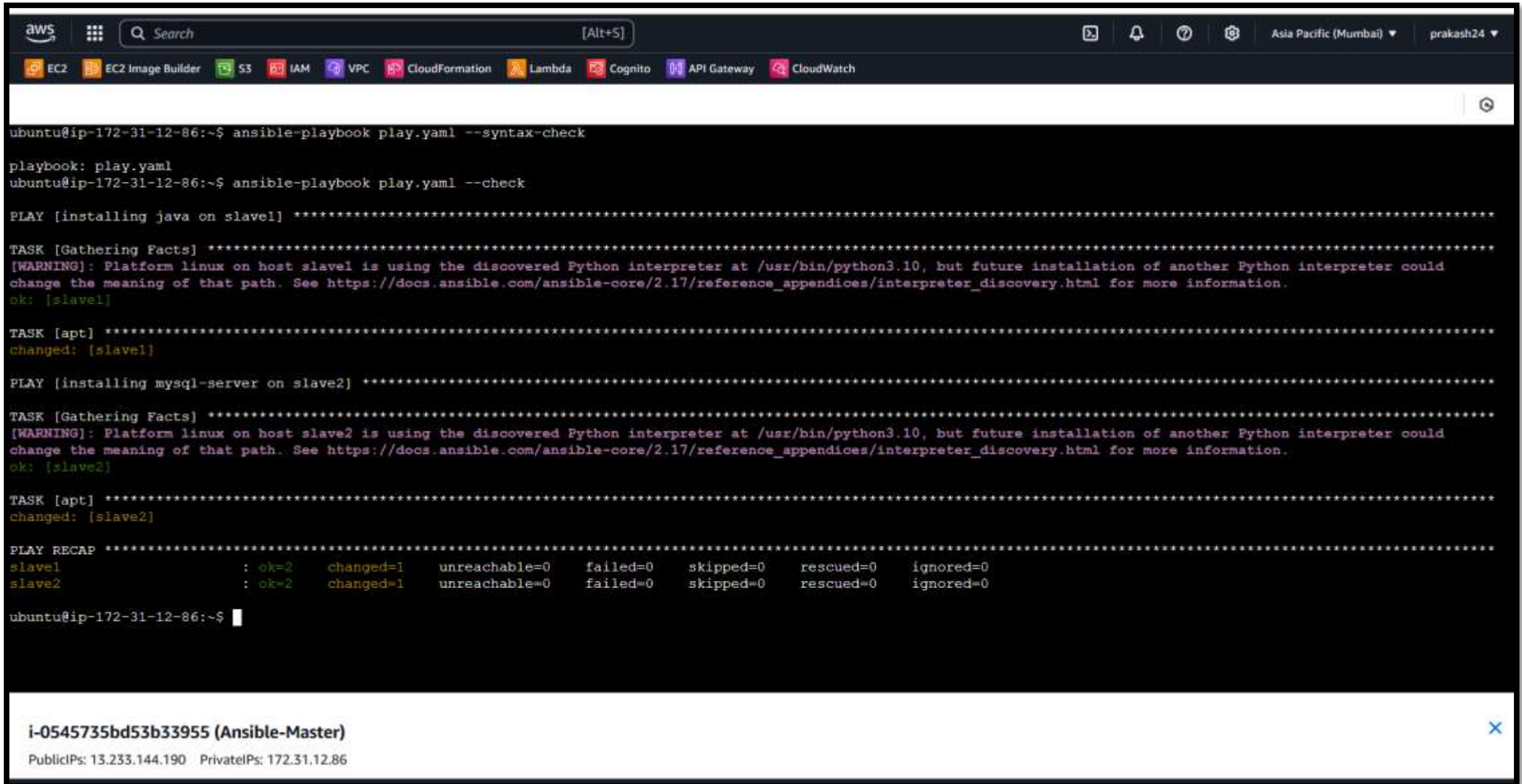
The screenshot shows an AWS console terminal window with the following components:

- Top Bar:** AWS logo, search bar, and navigation icons for EC2, EC2 Image Builder, S3, IAM, VPC, CloudFormation, Lambda, Cognito, API Gateway, and CloudWatch. The region is set to Asia Pacific (Mumbai) and the user is prakash24.
- Terminal Window:** Titled 'play.yaml', it shows the GNU nano 6.2 editor with the following Ansible playbook content:

```
--
- name: installing java on slave1
  hosts: slave1
  become: true
  tasks:
    - apt: name=openjdk-17-jdk state=latest
- name: installing mysql-server on slave2
  hosts: slave2
  become: true
  tasks:
    - apt: name=mysql-server state=latest
```
- Error Message:** A red banner across the terminal reads: **[File 'play.yaml' is unwritable]**
- Keyboard Shortcuts:** A list of shortcuts is shown at the bottom of the terminal:

^G Help	^O Write Out	^W Where Is	^K Cut	^T Execute	^C Location	M-U Undo	M-A Set Mark	M-] To Bracket	M-Q Previous
^X Exit	^R Read File	^_ Replace	^U Paste	^J Justify	^/_ Go To Line	M-E Redo	M-6 Copy	^Q Where Was	M-W Next
- Terminal Title Bar:** Shows the instance ID **i-0545735bd53b33955 (Ansible-Master)** and IP addresses: PublicIPs: 13.233.144.190, PrivateIPs: 172.31.12.86.

Checking syntax of playbook file & running dry run command



The screenshot shows an AWS CloudShell terminal window. The top navigation bar includes the AWS logo, a search bar, and various service icons (EC2, EC2 Image Builder, S3, IAM, VPC, CloudFormation, Lambda, Cognito, API Gateway, CloudWatch). The user is logged in as 'prakash24' in the 'Asia Pacific (Mumbai)' region.

```
ubuntu@ip-172-31-12-86:~$ ansible-playbook play.yaml --syntax-check
playbook: play.yaml
ubuntu@ip-172-31-12-86:~$ ansible-playbook play.yaml --check

PLAY [installing java on slavel] *****

TASK [Gathering Facts] *****
[WARNING]: Platform linux on host slavel is using the discovered Python interpreter at /usr/bin/python3.10, but future installation of another Python interpreter could change the meaning of that path. See https://docs.ansible.com/ansible-core/2.17/reference_appendices/interpreter_discovery.html for more information.
ok: [slavel]

TASK [apt] *****
changed: [slavel]

PLAY [installing mysql-server on slave2] *****

TASK [Gathering Facts] *****
[WARNING]: Platform linux on host slave2 is using the discovered Python interpreter at /usr/bin/python3.10, but future installation of another Python interpreter could change the meaning of that path. See https://docs.ansible.com/ansible-core/2.17/reference_appendices/interpreter_discovery.html for more information.
ok: [slave2]

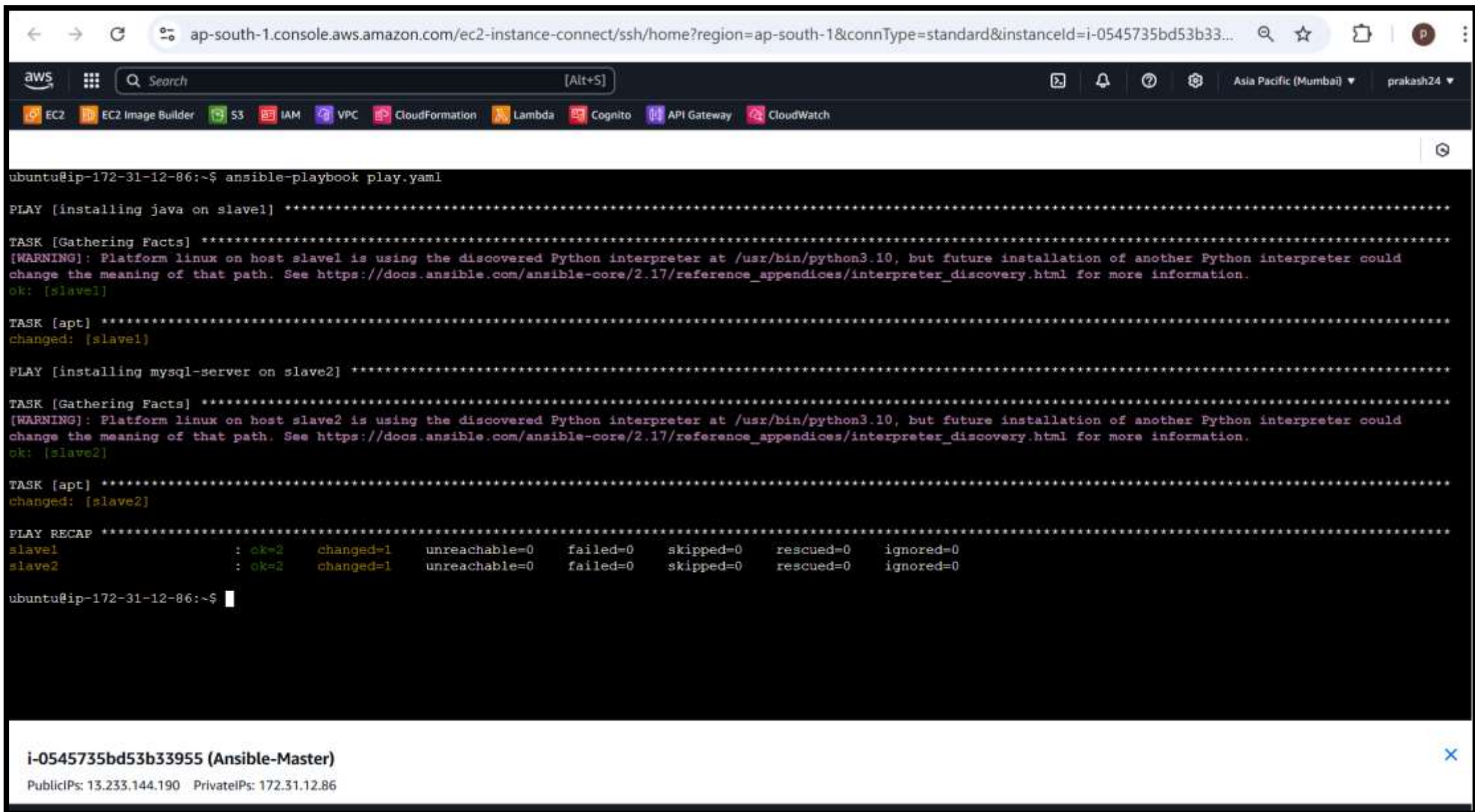
TASK [apt] *****
changed: [slave2]

PLAY RECAP *****
slavel          : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
slave2          : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

ubuntu@ip-172-31-12-86:~$
```

At the bottom of the terminal window, there is a summary box for the instance **i-0545735bd53b33955 (Ansible-Master)**. It lists the Public IPs as 13.233.144.190 and Private IPs as 172.31.12.86. A close button (X) is located in the top right corner of this box.

Running playbook file for installation



The screenshot shows the AWS Management Console interface for an EC2 instance named 'i-0545735bd53b33955 (Ansible-Master)'. The terminal window displays the output of an Ansible playbook execution. The playbook is named 'play.yaml' and is being run on a host named 'slave1'. The output shows the following steps:

```
ubuntu@ip-172-31-12-86:~$ ansible-playbook play.yaml

PLAY [installing java on slave1] *****

TASK [Gathering Facts] *****
[WARNING]: Platform linux on host slave1 is using the discovered Python interpreter at /usr/bin/python3.10, but future installation of another Python interpreter could change the meaning of that path. See https://docs.ansible.com/ansible-core/2.17/reference_appendices/interpreter_discovery.html for more information.
ok: [slave1]

TASK [apt] *****
changed: [slave1]

PLAY [installing mysql-server on slave2] *****

TASK [Gathering Facts] *****
[WARNING]: Platform linux on host slave2 is using the discovered Python interpreter at /usr/bin/python3.10, but future installation of another Python interpreter could change the meaning of that path. See https://docs.ansible.com/ansible-core/2.17/reference_appendices/interpreter_discovery.html for more information.
ok: [slave2]

TASK [apt] *****
changed: [slave2]

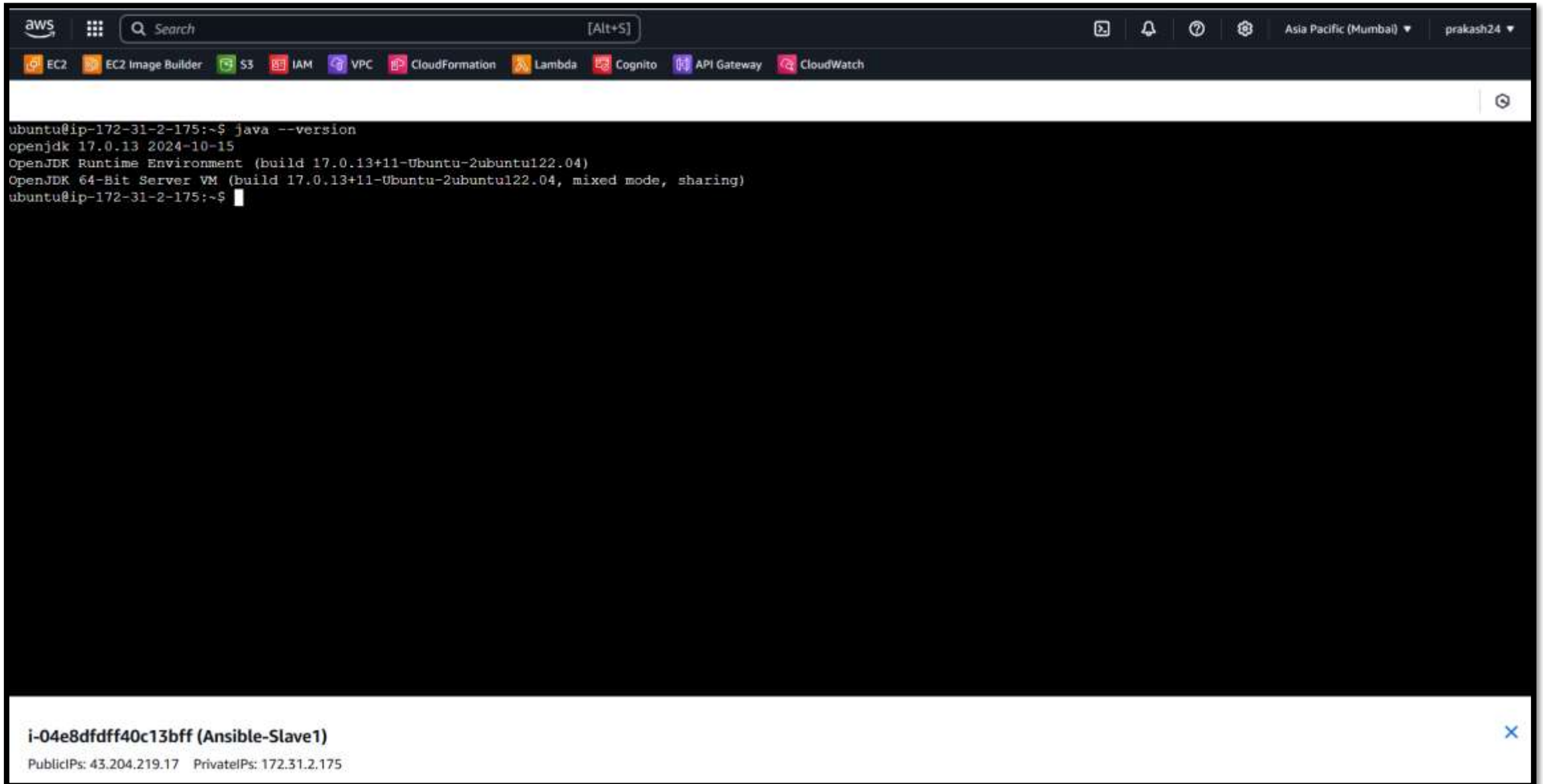
PLAY RECAP *****
slave1      : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
slave2      : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

ubuntu@ip-172-31-12-86:~$
```

At the bottom of the console, there is a summary box for the instance 'i-0545735bd53b33955 (Ansible-Master)' with the following details:

- Public IPs: 13.233.144.190
- Private IPs: 172.31.12.86

Validate java installation in slave 1

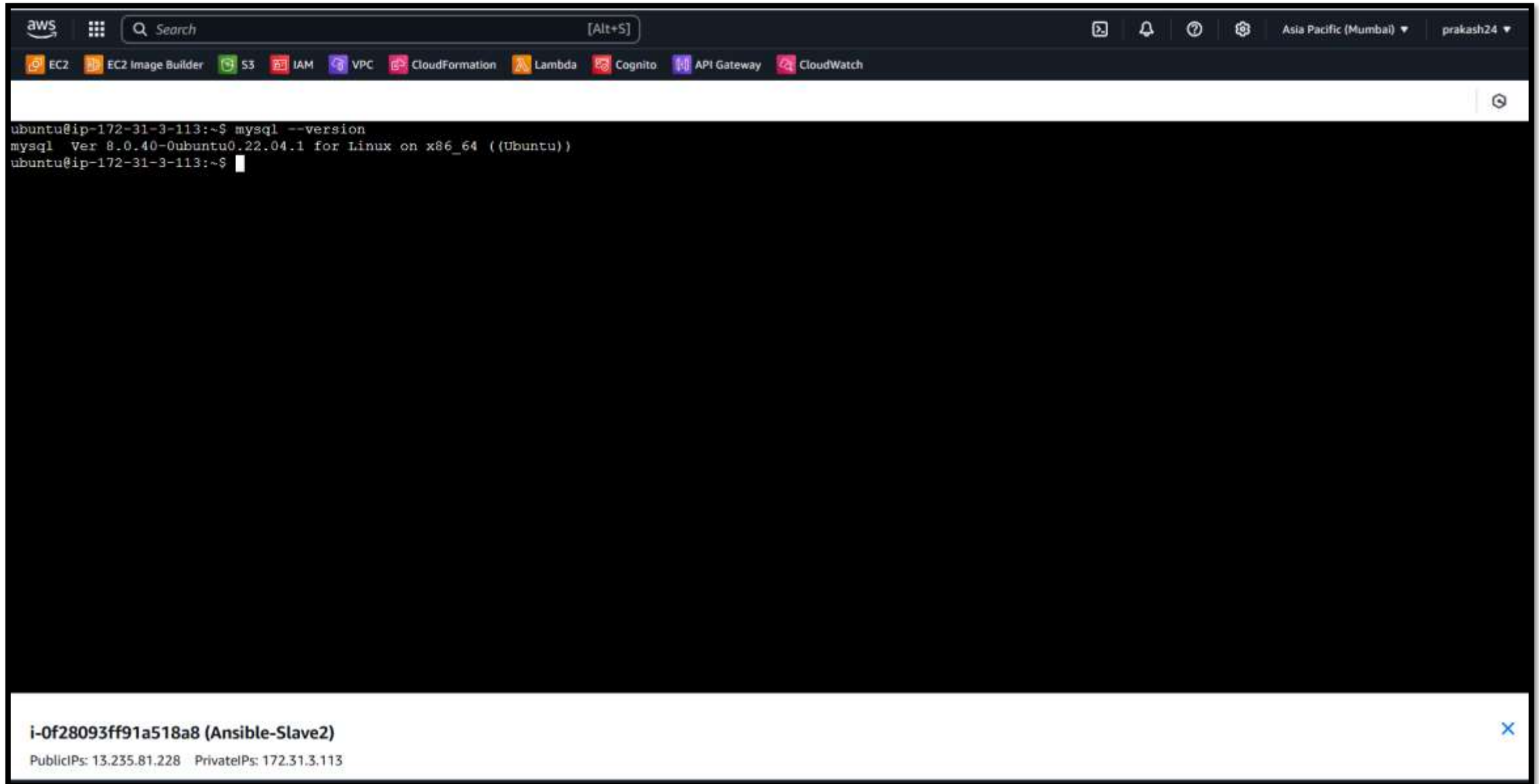


The screenshot shows the AWS CloudShell interface. At the top, there's a navigation bar with the AWS logo, a search bar, and various service icons (EC2, EC2 Image Builder, S3, IAM, VPC, CloudFormation, Lambda, Cognito, API Gateway, CloudWatch). The region is set to Asia Pacific (Mumbai) and the user is prakash24. The terminal window shows the command `java --version` being executed on an Ubuntu instance. The output displays the OpenJDK version (17.0.13), the build date (2024-10-15), the runtime environment details, and the VM architecture (64-bit server VM).

```
ubuntu@ip-172-31-2-175:~$ java --version
openjdk 17.0.13 2024-10-15
OpenJDK Runtime Environment (build 17.0.13+11-Ubuntu-2ubuntu122.04)
OpenJDK 64-Bit Server VM (build 17.0.13+11-Ubuntu-2ubuntu122.04, mixed mode, sharing)
ubuntu@ip-172-31-2-175:~$
```

i-04e8dfdff40c13bff (Ansible-Slave1)
PublicIPs: 43.204.219.17 PrivateIPs: 172.31.2.175

Validate MySQL installation in slave 2



The screenshot shows the AWS CloudShell interface. At the top, there's a navigation bar with the AWS logo, a search bar, and various service icons (EC2, EC2 Image Builder, S3, IAM, VPC, CloudFormation, Lambda, Cognito, API Gateway, CloudWatch). The right side of the bar shows the region 'Asia Pacific (Mumbai)' and the user 'prakash24'. Below the navigation bar is a terminal window with a black background and white text. The terminal shows the command 'mysql --version' being executed, resulting in the output 'mysql Ver 8.0.40-0ubuntu0.22.04.1 for Linux on x86_64 ((Ubuntu))'. At the bottom of the terminal window, there's a white status bar with the instance ID 'i-0f28093ff91a518a8 (Ansible-Slave2)' and its public and private IP addresses: 'PublicIPs: 13.235.81.228 PrivateIPs: 172.31.3.113'.

```
aws
[Alt+S]
EC2 EC2 Image Builder S3 IAM VPC CloudFormation Lambda Cognito API Gateway CloudWatch
ubuntu@ip-172-31-3-113:~$ mysql --version
mysql Ver 8.0.40-0ubuntu0.22.04.1 for Linux on x86_64 ((Ubuntu))
ubuntu@ip-172-31-3-113:~$
```

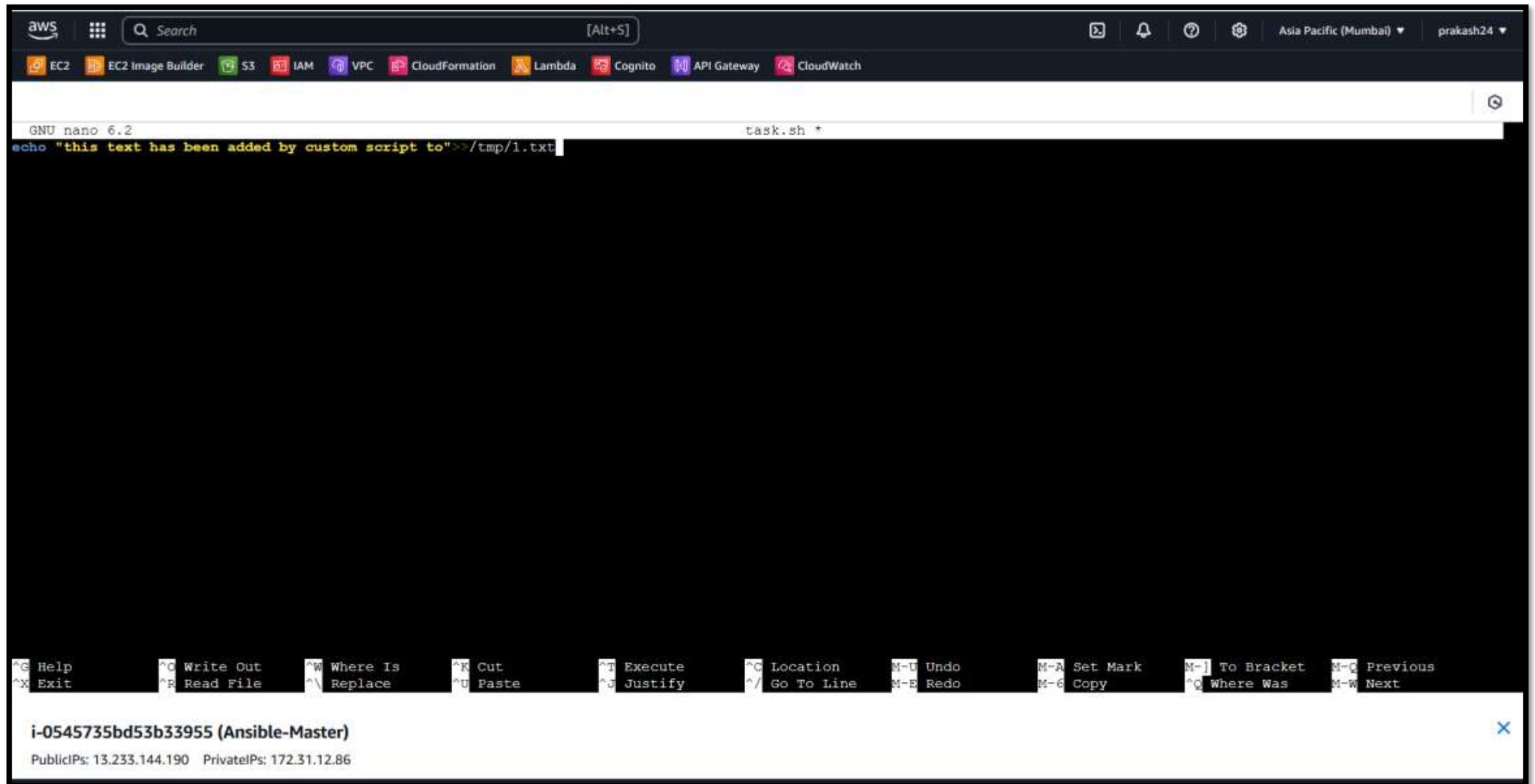
i-0f28093ff91a518a8 (Ansible-Slave2)
PublicIPs: 13.235.81.228 PrivateIPs: 172.31.3.113

Module 5: Ansible Assignment - 2

Tasks To Be Performed:

1. Create a script which can add text "This text has been added by custom script" to /tmp.1.txt
2. Run this script using Ansible on all the hosts

Creating a 1.txt file in /tmp/ location



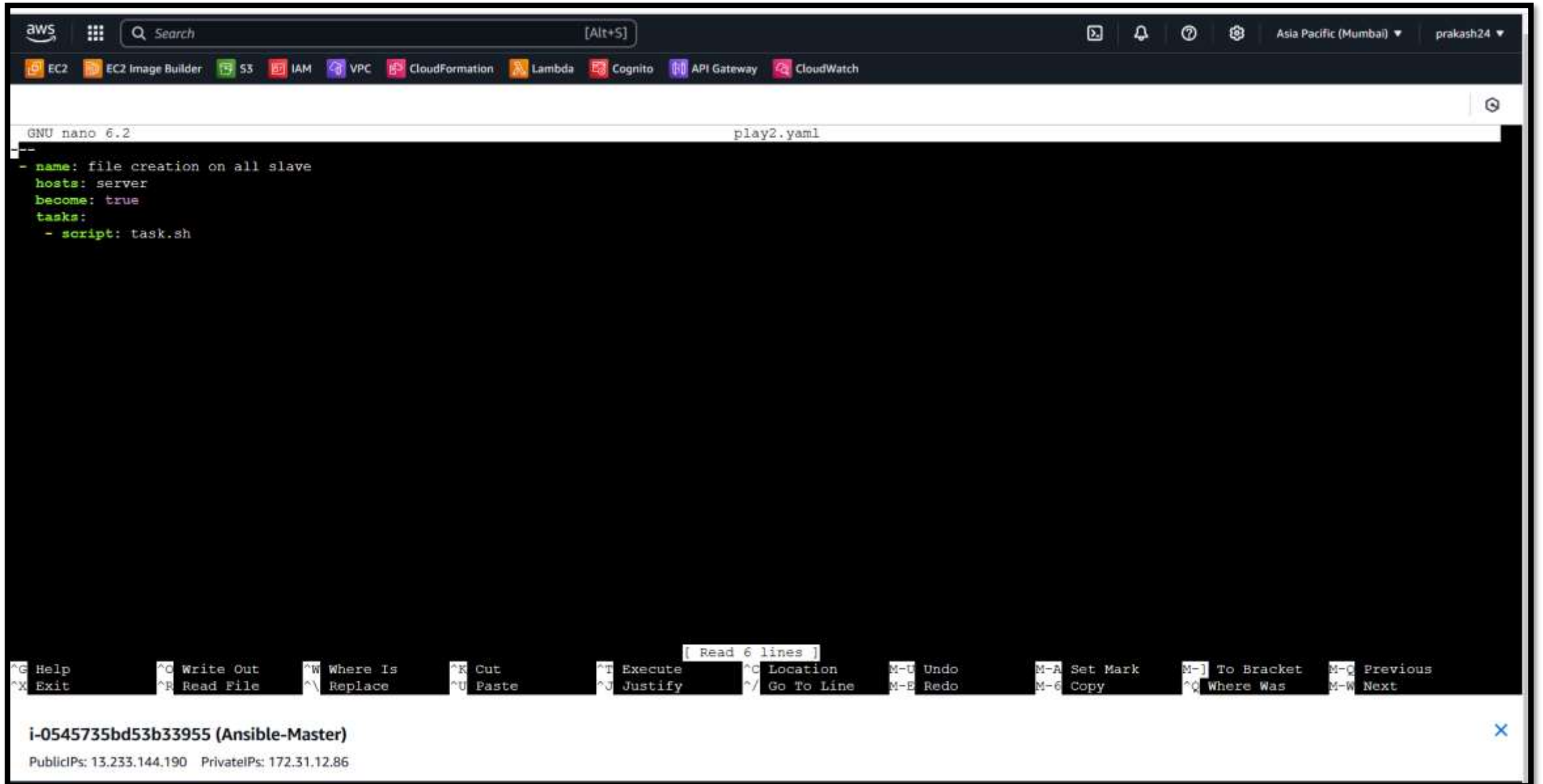
The screenshot shows an AWS console terminal window. At the top, the AWS logo and a search bar are visible. Below the search bar, a row of service icons includes EC2, EC2 Image Builder, S3, IAM, VPC, CloudFormation, Lambda, Cognito, API Gateway, and CloudWatch. The terminal itself has a title bar with 'task.sh' and a close button. The main area shows the GNU nano 6.2 editor with the command `echo "this text has been added by custom script to">>/tmp/1.txt` entered. The bottom of the terminal displays a list of nano editor shortcuts: `^G Help`, `^O Write Out`, `^W Where Is`, `^K Cut`, `^T Execute`, `^C Location`, `^M-U Undo`, `^M-A Set Mark`, `^M-J To Bracket`, `^M-Q Previous`, `^X Exit`, `^R Read File`, `^_ Replace`, `^U Paste`, `^J Justify`, `^/_ Go To Line`, `^M-E Redo`, `^M-6 Copy`, `^Q Where Was`, and `^M-W Next`. At the very bottom, a status bar shows the instance ID `i-0545735bd53b33955 (Ansible-Master)` and its IP addresses: `PublicIPs: 13.233.144.190` and `PrivateIPs: 172.31.12.86`.

```
GNU nano 6.2 task.sh *
echo "this text has been added by custom script to">>/tmp/1.txt
```

`^G Help` `^O Write Out` `^W Where Is` `^K Cut` `^T Execute` `^C Location` `^M-U Undo` `^M-A Set Mark` `^M-J To Bracket` `^M-Q Previous`
`^X Exit` `^R Read File` `^_ Replace` `^U Paste` `^J Justify` `^/_ Go To Line` `^M-E Redo` `^M-6 Copy` `^Q Where Was` `^M-W Next`

i-0545735bd53b33955 (Ansible-Master)
PublicIPs: 13.233.144.190 PrivateIPs: 172.31.12.86

Creating a playbook file to move 1.txt file using task.sh file



The screenshot shows the AWS Management Console interface. At the top, there's a navigation bar with the AWS logo, a search bar, and various service icons including EC2, EC2 Image Builder, S3, IAM, VPC, CloudFormation, Lambda, Cognito, API Gateway, and CloudWatch. The user is logged in as 'prakash24' in the 'Asia Pacific (Mumbai)' region.

The main content area displays a terminal window running GNU nano 6.2, editing a file named 'play2.yaml'. The content of the file is an Ansible playbook:

```
--  
- name: file creation on all slave  
  hosts: server  
  become: true  
  tasks:  
    - script: task.sh
```

At the bottom of the terminal window, there's a status bar showing the IP address 'i-0545735bd53b33955 (Ansible-Master)' and its public and private IP addresses: 'PublicIPs: 13.233.144.190' and 'PrivateIPs: 172.31.12.86'. A blue 'X' icon is visible in the bottom right corner of the terminal window.

Checking syntax of playbook file & running dry run command

```
aws [Search] [Alt+S] Asia Pacific (Mumbai) prakash24
EC2 EC2 Image Builder S3 IAM VPC CloudFormation Lambda Cognito API Gateway CloudWatch

ubuntu@ip-172-31-12-86:~$ sudo nano play2.yaml
ubuntu@ip-172-31-12-86:~$ ansible-playbook play2.yaml --syntax-check

playbook: play2.yaml
ubuntu@ip-172-31-12-86:~$ ansible-playbook play2.yaml --check

PLAY [file creation on all slave] *****

TASK [Gathering Facts] *****
[WARNING]: Platform linux on host slave1 is using the discovered Python interpreter at /usr/bin/python3.10, but future installation of another Python interpreter could change the meaning of that path. See https://docs.ansible.com/ansible-core/2.17/reference_appendices/interpreter_discovery.html for more information.
ok: [slave1]
[WARNING]: Platform linux on host slave2 is using the discovered Python interpreter at /usr/bin/python3.10, but future installation of another Python interpreter could change the meaning of that path. See https://docs.ansible.com/ansible-core/2.17/reference_appendices/interpreter_discovery.html for more information.
ok: [slave2]

TASK [script] *****
skipping: [slave1]
skipping: [slave2]

PLAY RECAP *****
slave1      : ok=1    changed=0    unreachable=0    failed=0    skipped=1    rescued=0    ignored=0
slave2      : ok=1    changed=0    unreachable=0    failed=0    skipped=1    rescued=0    ignored=0

ubuntu@ip-172-31-12-86:~$
```

i-0545735bd53b33955 (Ansible-Master)

PublicIPs: 13.233.144.190 PrivateIPs: 172.31.12.86

Running playbook file for move file from give location /tmp/

```
aws [Search] [Alt+S] Asia Pacific (Mumbai) prakash24
```

EC2 EC2 Image Builder S3 IAM VPC CloudFormation Lambda Cognito API Gateway CloudWatch

```
ubuntu@ip-172-31-12-86:~$ ansible-playbook play2.yaml
```

```
PLAY [file creation on all slave] *****
```

```
TASK [Gathering Facts] *****
```

```
[WARNING]: Platform linux on host slave1 is using the discovered Python interpreter at /usr/bin/python3.10, but future installation of another Python interpreter could change the meaning of that path. See https://docs.ansible.com/ansible-core/2.17/reference_appendices/interpreter_discovery.html for more information.
```

```
ok: [slave1]
```

```
[WARNING]: Platform linux on host slave2 is using the discovered Python interpreter at /usr/bin/python3.10, but future installation of another Python interpreter could change the meaning of that path. See https://docs.ansible.com/ansible-core/2.17/reference_appendices/interpreter_discovery.html for more information.
```

```
ok: [slave2]
```

```
TASK [script] *****
```

```
changed: [slave1]
```

```
changed: [slave2]
```

```
PLAY RECAP *****
```

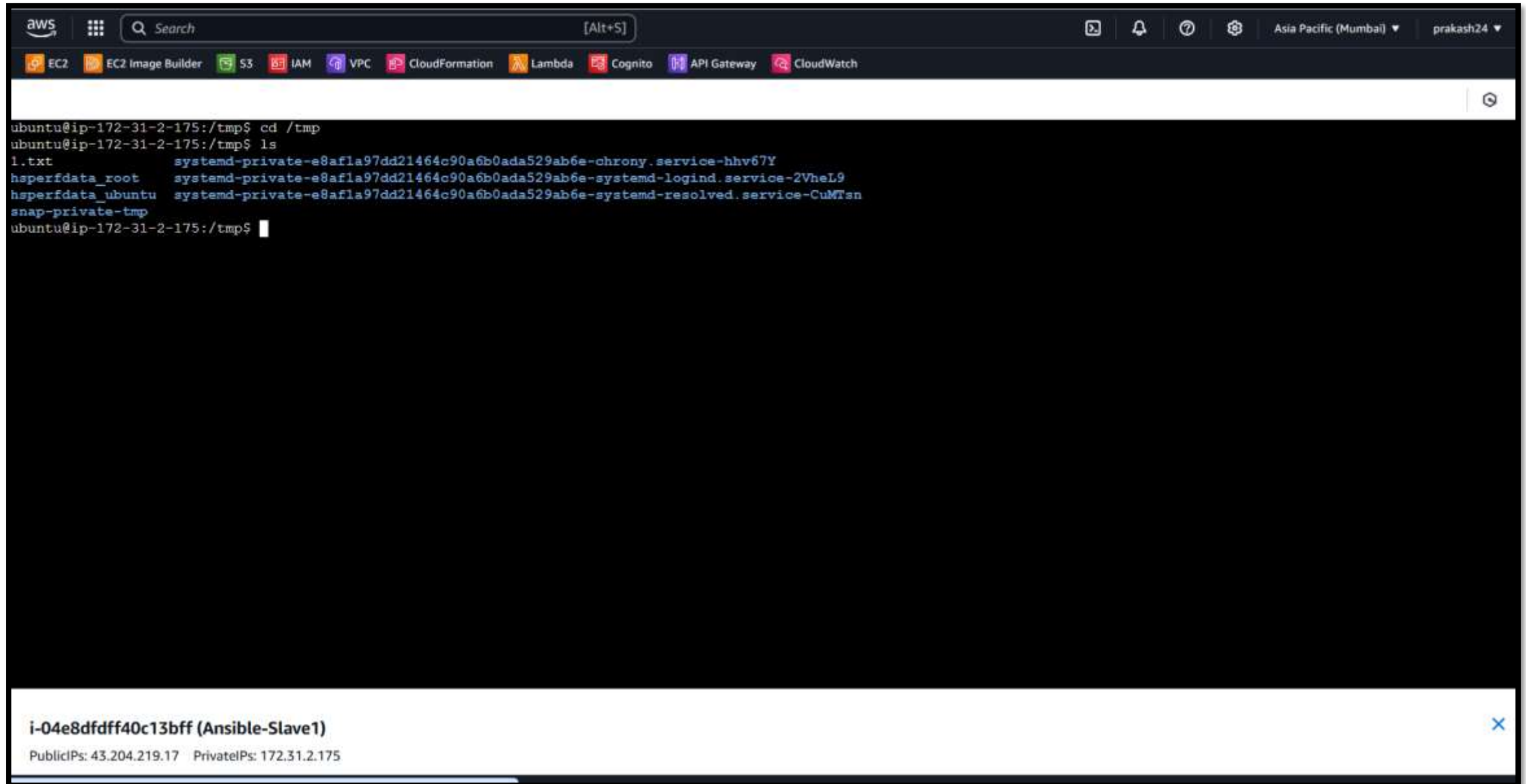
slave1	: ok=2	changed=1	unreachable=0	failed=0	skipped=0	rescued=0	ignored=0
slave2	: ok=2	changed=1	unreachable=0	failed=0	skipped=0	rescued=0	ignored=0

```
ubuntu@ip-172-31-12-86:~$
```

i-0545735bd53b33955 (Ansible-Master)

PublicIPs: 13.233.144.190 PrivateIPs: 172.31.12.86

Validate in slave1 we see 1.txt file in location /tmp/

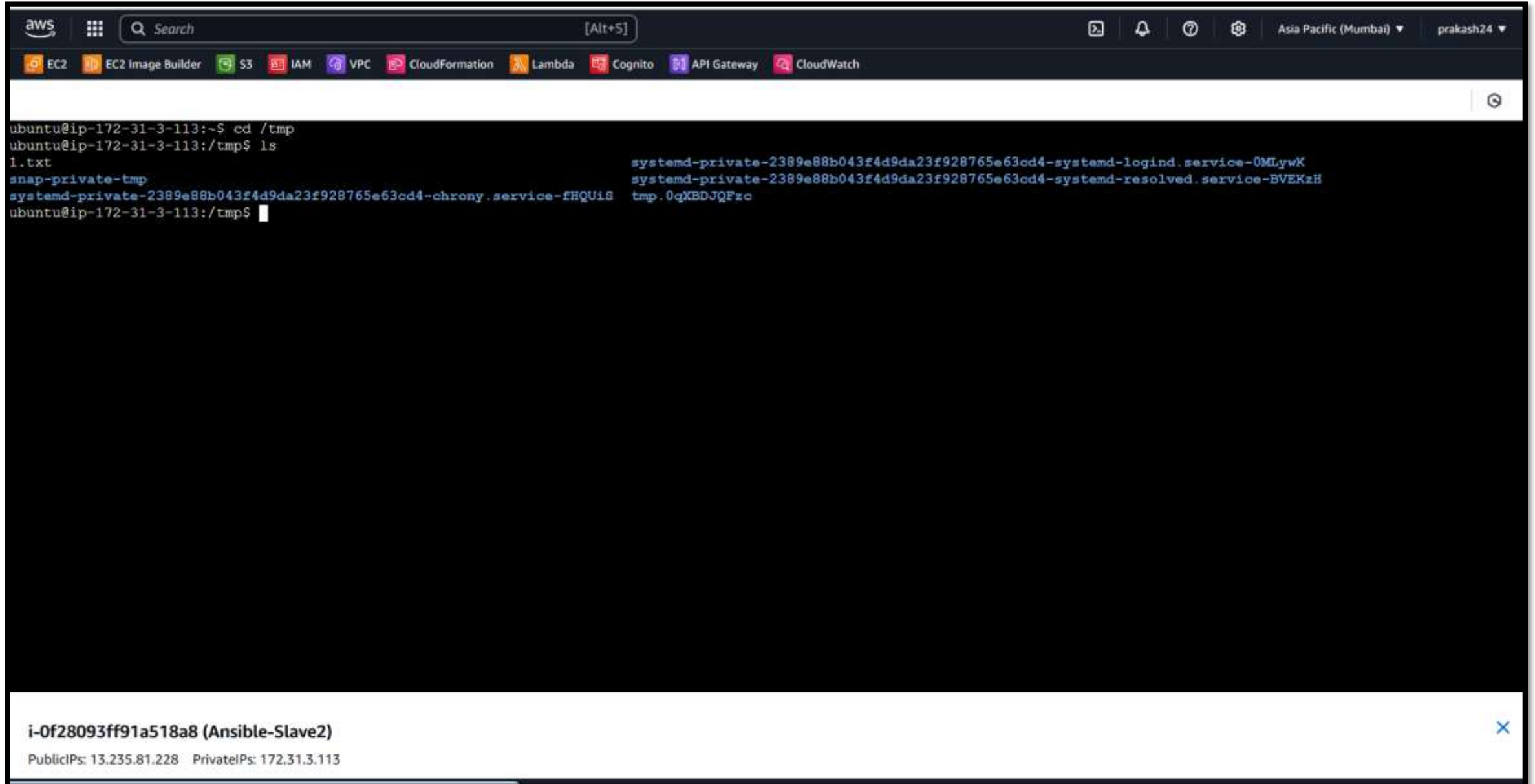


The screenshot shows the AWS CloudShell interface. At the top, there's a navigation bar with the AWS logo, a search bar, and various service icons (EC2, EC2 Image Builder, S3, IAM, VPC, CloudFormation, Lambda, Cognito, API Gateway, CloudWatch). The region is set to Asia Pacific (Mumbai) and the user is prakash24. The terminal window shows a series of commands and their output:

```
ubuntu@ip-172-31-2-175:/tmp$ cd /tmp
ubuntu@ip-172-31-2-175:/tmp$ ls
1.txt                systemd-private-e8afla97dd21464c90a6b0ada529ab6e-chrony.service-hhv67Y
hspcrfdata_root      systemd-private-e8afla97dd21464c90a6b0ada529ab6e-systemd-logind.service-2VheL9
hspcrfdata_ubuntu     systemd-private-e8afla97dd21464c90a6b0ada529ab6e-systemd-resolved.service-CuMTsn
snap-private-tmp
ubuntu@ip-172-31-2-175:/tmp$
```

At the bottom, a status bar identifies the instance as i-04e8dfdf40c13bff (Ansible-Slave1) and provides public and private IP addresses: PublicIPs: 43.204.219.17, PrivateIPs: 172.31.2.175.

Validate in slave2 we see 1.txt file in location /tmp/



The screenshot shows the AWS Management Console interface with a terminal window open. The terminal displays the following commands and output:

```
ubuntu@ip-172-31-3-113:~$ cd /tmp
ubuntu@ip-172-31-3-113:/tmp$ ls
1.txt
snap-private-tmp
systemd-private-2389e88b043f4d9da23f928765e63cd4-chrony.service-FHQUiS
systemd-private-2389e88b043f4d9da23f928765e63cd4-systemd-logind.service-0MLywK
systemd-private-2389e88b043f4d9da23f928765e63cd4-systemd-resolved.service-BVEKzH
tmp.0qXBDJQFzc
ubuntu@ip-172-31-3-113:/tmp$
```

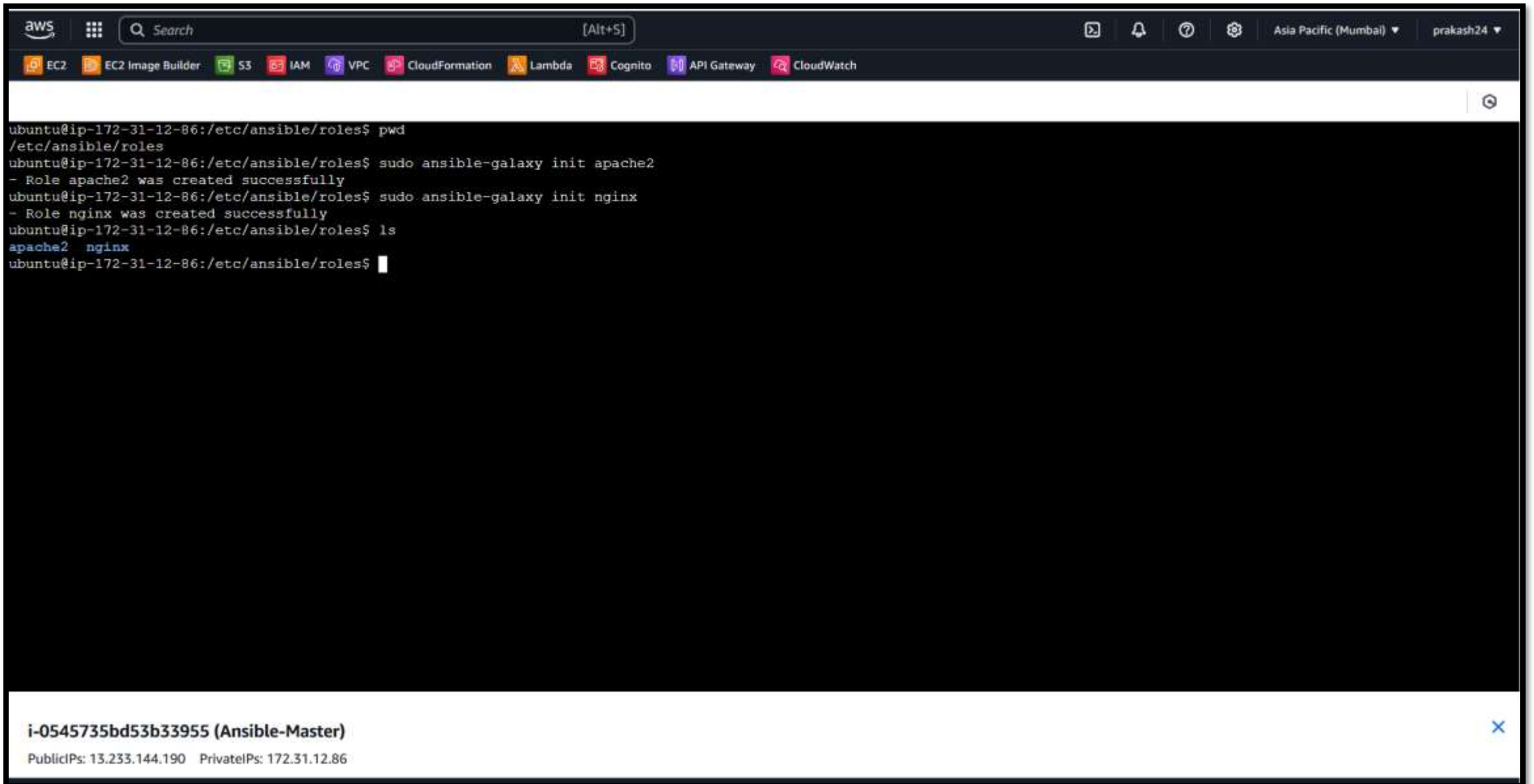
The terminal window is titled "i-Of28093ff91a518a8 (Ansible-Slave2)". At the bottom, it shows the instance's IP addresses: "PublicIPs: 13.235.81.228 PrivateIPs: 172.31.3.113".

Module 5: Ansible Assignment - 3

Tasks To Be Performed:

1. Create 2 Ansible roles
2. Install Apache2 on slave1 using one role and NGINX on slave2 using the other role
3. Above should be implemented using different Ansible roles

Creating two roles apache2 & nginx

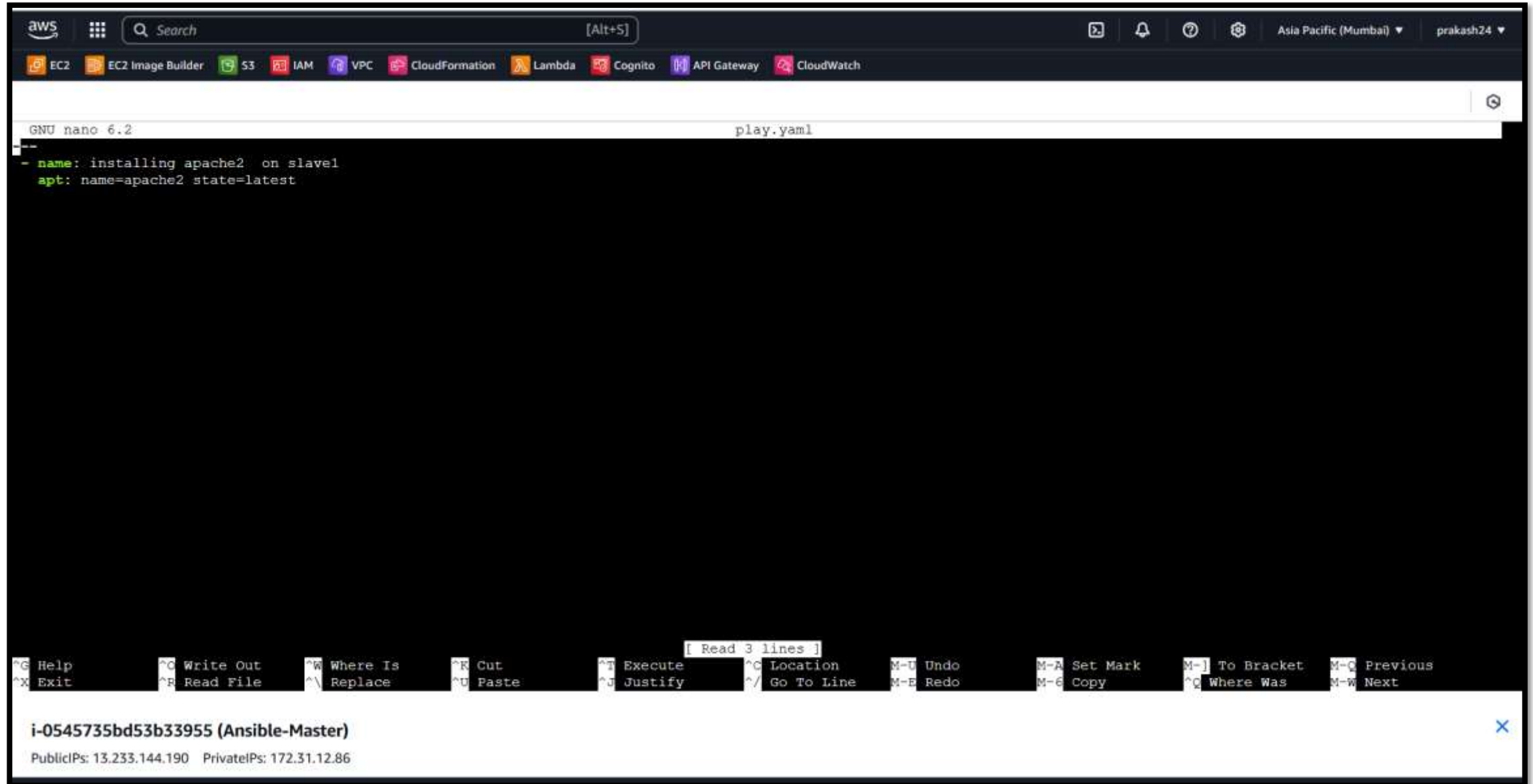


The screenshot shows an AWS console terminal window with a dark theme. The top bar includes the AWS logo, a search bar, and navigation icons. Below the bar, a row of service icons is visible: EC2, EC2 Image Builder, S3, IAM, VPC, CloudFormation, Lambda, Cognito, API Gateway, and CloudWatch. The terminal output shows a user running commands to create Ansible roles. The commands and their outputs are as follows:

```
ubuntu@ip-172-31-12-86:/etc/ansible/roles$ pwd
/etc/ansible/roles
ubuntu@ip-172-31-12-86:/etc/ansible/roles$ sudo ansible-galaxy init apache2
- Role apache2 was created successfully
ubuntu@ip-172-31-12-86:/etc/ansible/roles$ sudo ansible-galaxy init nginx
- Role nginx was created successfully
ubuntu@ip-172-31-12-86:/etc/ansible/roles$ ls
apache2  nginx
ubuntu@ip-172-31-12-86:/etc/ansible/roles$
```

At the bottom of the terminal window, a status bar displays the instance ID **i-0545735bd53b33955 (Ansible-Master)** and its IP addresses: Public IPs: 13.233.144.190 and Private IPs: 172.31.12.86.

Creating play.yaml file for ruing installation of apache2 inside role tasks folder



The screenshot shows an AWS console terminal window. The top navigation bar includes the AWS logo, a search bar, and various service icons (EC2, EC2 Image Builder, S3, IAM, VPC, CloudFormation, Lambda, Cognito, API Gateway, CloudWatch). The user is logged in as 'prakash24' in the 'Asia Pacific (Mumbai)' region. The terminal window title is 'play.yaml'. The editor shows the following content:

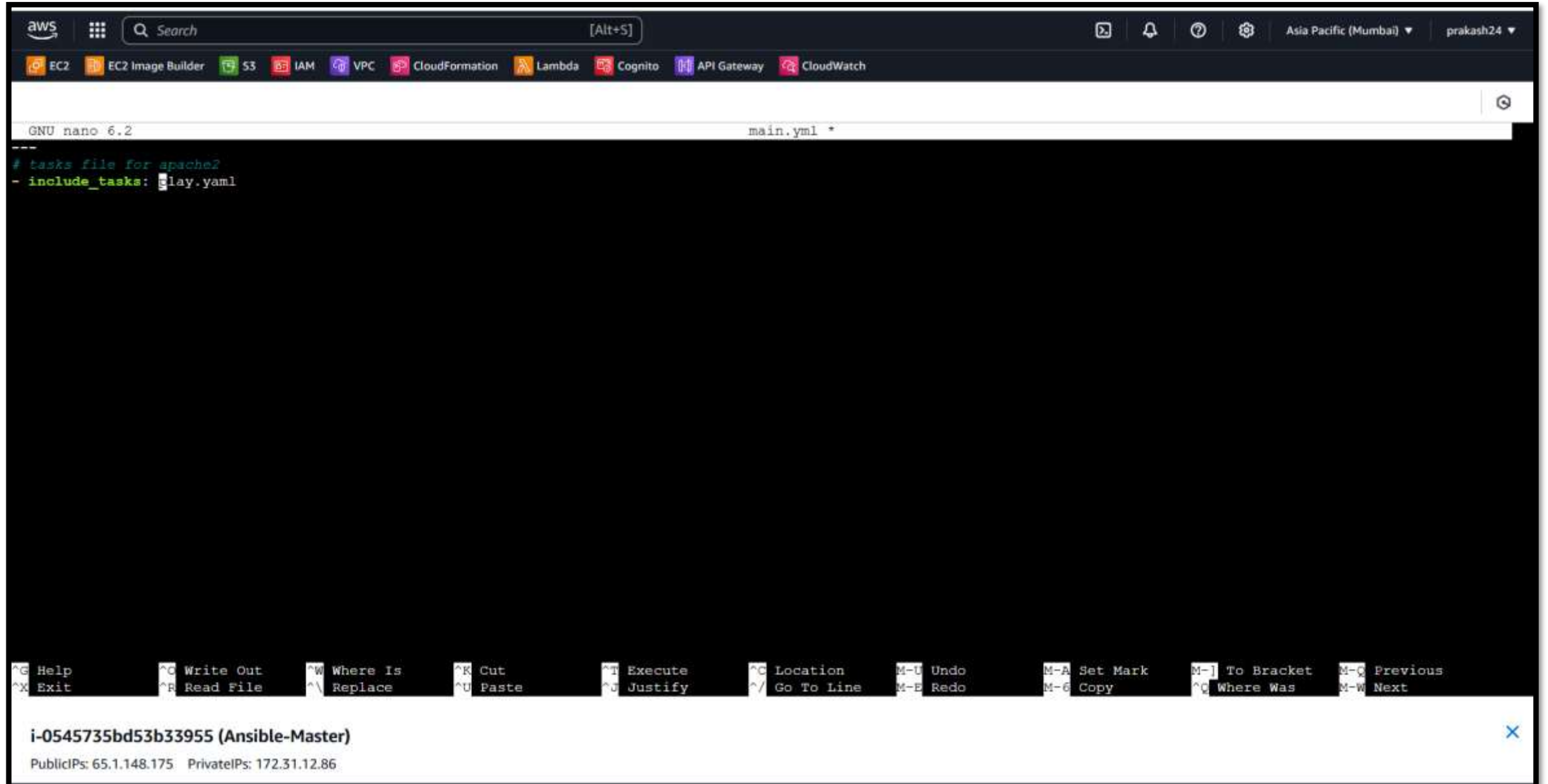
```
GNU nano 6.2 play.yaml
--
- name: installing apache2 on slave1
  apt: name=apache2 state=latest
```

The bottom of the terminal displays a list of nano editor shortcuts:

^G Help	^O Write Out	^W Where Is	^K Cut	^T Execute	^C Location	M-U Undo	M-A Set Mark	M-l To Bracket	M-Q Previous
^X Exit	^R Read File	^_ Replace	^U Paste	^J Justify	^/_ Go To Line	M-E Redo	M-6 Copy	^Q Where Was	M-W Next

At the bottom, a status bar shows the instance ID 'i-0545735bd53b33955 (Ansible-Master)' and IP addresses: 'PublicIPs: 13.233.144.190 PrivateIPs: 172.31.12.86'.

Including play.yml file in main.yml file



The screenshot shows a terminal window within the AWS console. The terminal title bar indicates 'GNU nano 6.2' and the file being edited is 'main.yml'. The content of the file is as follows:

```
---  
# tasks file for apache2  
- include_tasks: play.yml
```

The terminal window has a dark background with light-colored text. The top of the window shows the AWS console header with various service icons (EC2, EC2 Image Builder, S3, IAM, VPC, CloudFormation, Lambda, Cognito, API Gateway, CloudWatch) and a search bar. The bottom of the window displays a status bar with the instance ID 'i-0545735bd53b33955 (Ansible-Master)' and its IP addresses: 'PublicIPs: 65.1.148.175' and 'PrivateIPs: 172.31.12.86'. A help menu is visible at the bottom of the terminal, listing various keyboard shortcuts for nano editor operations.

Shortcut	Action
^G	Help
^C	Write Out
^W	Where Is
^K	Cut
^T	Execute
^C	Location
M-U	Undo
M-A	Set Mark
M-]	To Bracket
M-]	Where Was
M-Q	Previous
M-W	Next
^X	Exit
^R	Read File
^\	Replace
^U	Paste
^J	Justify
^/_	Go To Line
M-B	Redo
M-6	Copy

Creating play.yaml file for ruing installation of nginx inside role tasks folder

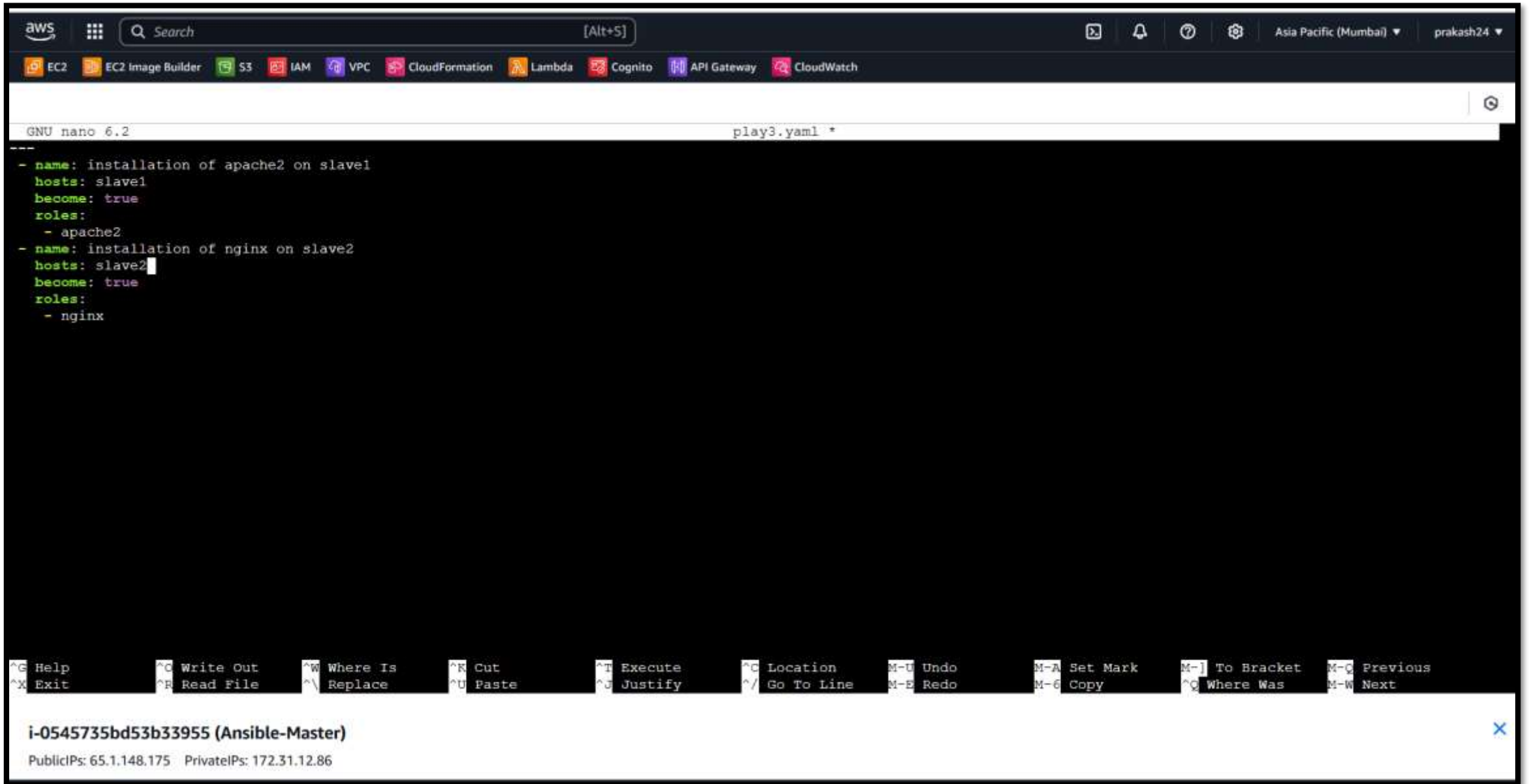


The screenshot shows the AWS Management Console interface. At the top, there's a navigation bar with the AWS logo, a search bar, and various service icons (EC2, EC2 Image Builder, S3, IAM, VPC, CloudFormation, Lambda, Cognito, API Gateway, CloudWatch). The user is logged in as 'prakash24' in the 'Asia Pacific (Mumbai)' region. The main content area displays a terminal window titled 'GNU nano 6.2' editing a file named 'play.yaml'. The file contains two lines of Ansible play syntax:

```
--  
- name: installing nginx server slave2  
- apt: name=nginx state=latest
```

At the bottom of the terminal window, there's a status bar showing various keyboard shortcuts for nano (e.g., ^G Help, ^O Write Out, ^W Where Is, etc.). Below the terminal window, a white box displays the instance ID 'i-0545735bd53b33955 (Ansible-Master)' and its IP addresses: 'PublicIPs: 65.1.148.175' and 'PrivateIPs: 172.31.12.86'.

Creating play3.yaml playbook file to run roles

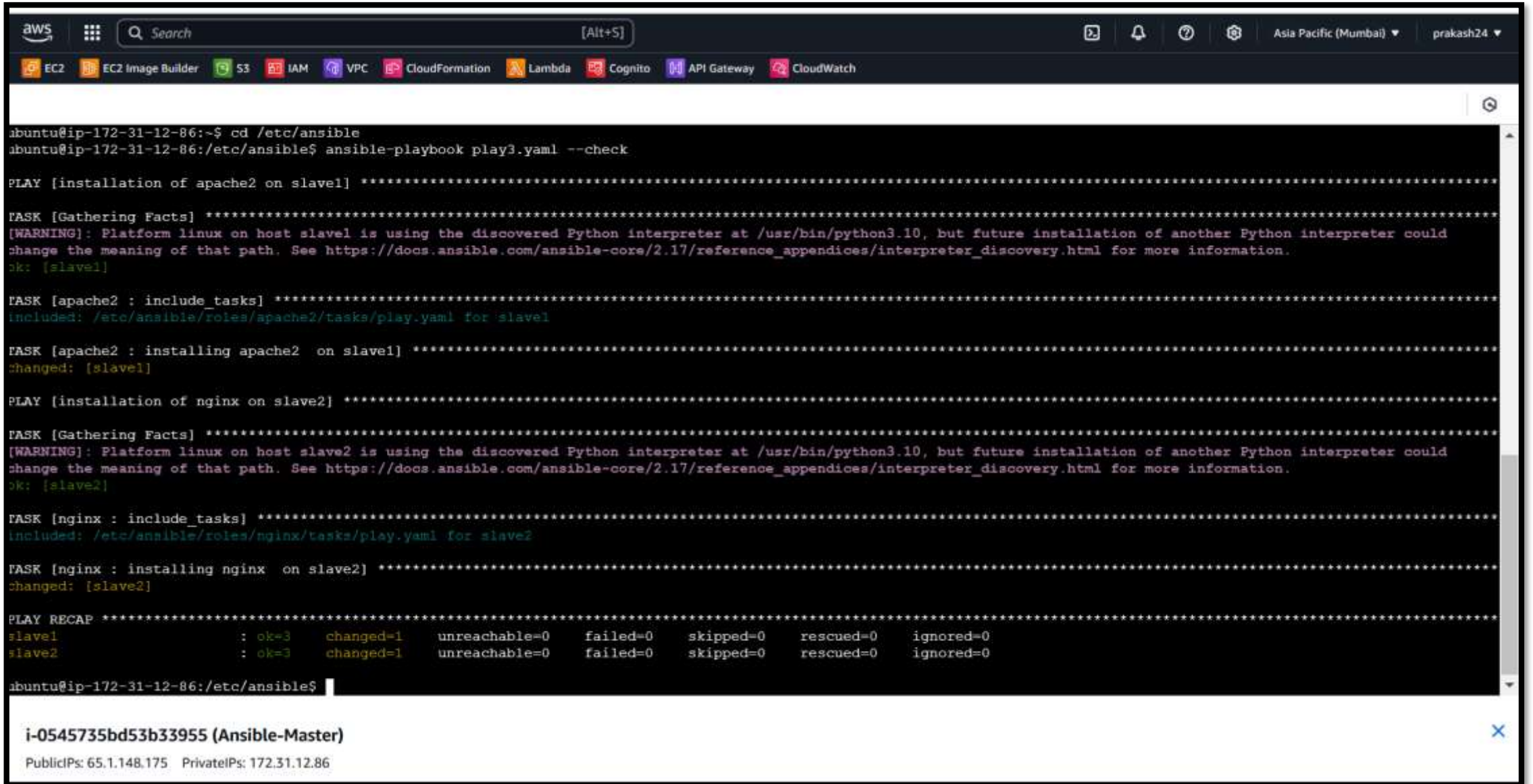


The screenshot shows the AWS Management Console interface. At the top, there's a navigation bar with the AWS logo, a search bar, and a list of services including EC2, EC2 Image Builder, S3, IAM, VPC, CloudFormation, Lambda, Cognito, API Gateway, and CloudWatch. The user is logged in as 'prakash24' in the 'Asia Pacific (Mumbai)' region. Below the navigation bar, a terminal window titled 'GNU nano 6.2' is open, editing a file named 'play3.yaml'. The file contains two playbooks:

```
---
- name: installation of apache2 on slave1
  hosts: slave1
  become: true
  roles:
    - apache2
- name: installation of nginx on slave2
  hosts: slave2
  become: true
  roles:
    - nginx
```

At the bottom of the terminal window, there's a status bar with various keyboard shortcuts for nano editor functions like Help, Exit, Write Out, Read File, Where Is, Replace, Cut, Paste, Execute, Justify, Location, Go To Line, Undo, Redo, Set Mark, Copy, To Bracket, Where Was, Previous, and Next. Below the terminal window, a system information bar shows the instance ID 'i-0545735bd53b33955 (Ansible-Master)' and its IP addresses: 'PublicIPs: 65.1.148.175' and 'PrivateIPs: 172.31.12.86'.

Running dry run commands for validation playbook file



The screenshot shows an AWS CloudShell terminal window. The top bar includes the AWS logo, a search bar, and navigation icons for various services like EC2, IAM, VPC, CloudFormation, Lambda, Cognito, API Gateway, and CloudWatch. The user is logged in as 'prakash24' in the 'Asia Pacific (Mumbai)' region.

```
abuntu@ip-172-31-12-86:~$ cd /etc/ansible
abuntu@ip-172-31-12-86:/etc/ansible$ ansible-playbook play3.yaml --check
```

The output of the command shows the dry run results for two hosts, slave1 and slave2.

```
PLAY [installation of apache2 on slave1] *****

TASK [Gathering Facts] *****
[WARNING]: Platform linux on host slave1 is using the discovered Python interpreter at /usr/bin/python3.10, but future installation of another Python interpreter could
change the meaning of that path. See https://docs.ansible.com/ansible-core/2.17/reference_appendices/interpreter_discovery.html for more information.
ok: [slave1]

TASK [apache2 : include tasks] *****
included: /etc/ansible/roles/apache2/tasks/play.yaml for slave1

TASK [apache2 : installing apache2 on slave1] *****
changed: [slave1]

PLAY [installation of nginx on slave2] *****

TASK [Gathering Facts] *****
[WARNING]: Platform linux on host slave2 is using the discovered Python interpreter at /usr/bin/python3.10, but future installation of another Python interpreter could
change the meaning of that path. See https://docs.ansible.com/ansible-core/2.17/reference_appendices/interpreter_discovery.html for more information.
ok: [slave2]

TASK [nginx : include tasks] *****
included: /etc/ansible/roles/nginx/tasks/play.yaml for slave2

TASK [nginx : installing nginx on slave2] *****
changed: [slave2]

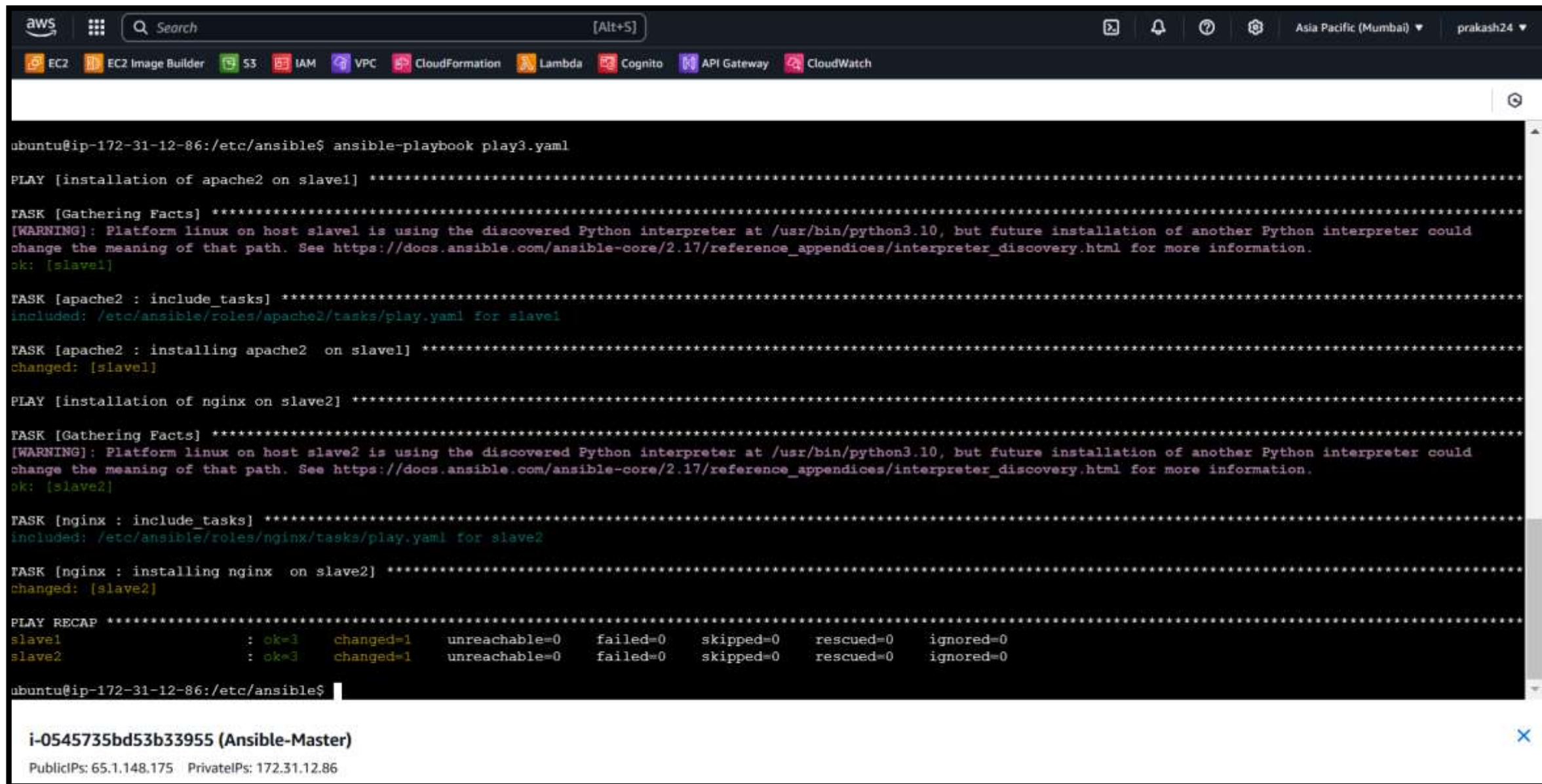
PLAY RECAP *****
slave1      : ok=3    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
slave2      : ok=3    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

abuntu@ip-172-31-12-86:/etc/ansible$
```

At the bottom of the terminal window, there is a box for the instance ID and IP addresses:

```
i-0545735bd53b33955 (Ansible-Master)
PublicIPs: 65.1.148.175 PrivateIPs: 172.31.12.86
```


Running playbook file for installation



The screenshot shows the AWS CloudShell interface with a terminal window. The terminal displays the execution of an Ansible playbook named 'play3.yaml'. The playbook consists of two plays: one for installing Apache on 'slave1' and another for installing Nginx on 'slave2'. Each play includes a 'Gathering Facts' task with a warning about the Python interpreter, followed by an 'include_tasks' task and an 'installing' task. The output shows successful execution on both nodes. Below the terminal, a metadata box for the instance 'i-0545735bd53b33955 (Ansible-Master)' is visible, showing public and private IP addresses.

```
aws
[Search] [Alt+S]
EC2 EC2 Image Builder S3 IAM VPC CloudFormation Lambda Cognito API Gateway CloudWatch

ubuntu@ip-172-31-12-86:/etc/ansible$ ansible-playbook play3.yaml

PLAY [installation of apache2 on slave1] *****

TASK [Gathering Facts] *****
[WARNING]: Platform linux on host slave1 is using the discovered Python interpreter at /usr/bin/python3.10, but future installation of another Python interpreter could change the meaning of that path. See https://docs.ansible.com/ansible-core/2.17/reference_appendices/interpreter_discovery.html for more information.
ok: [slave1]

TASK [apache2 : include_tasks] *****
included: /etc/ansible/roles/apache2/tasks/play.yaml for slave1

TASK [apache2 : installing apache2 on slave1] *****
changed: [slave1]

PLAY [installation of nginx on slave2] *****

TASK [Gathering Facts] *****
[WARNING]: Platform linux on host slave2 is using the discovered Python interpreter at /usr/bin/python3.10, but future installation of another Python interpreter could change the meaning of that path. See https://docs.ansible.com/ansible-core/2.17/reference_appendices/interpreter_discovery.html for more information.
ok: [slave2]

TASK [nginx : include_tasks] *****
included: /etc/ansible/roles/nginx/tasks/play.yaml for slave2







TASK [nginx : installing nginx on slave2] *****
changed: [slave2]











PLAY RECAP *****
slave1      : ok=3    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
slave2      : ok=3    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

ubuntu@ip-172-31-12-86:/etc/ansible$
```

i-0545735bd53b33955 (Ansible-Master)
PublicIPs: 65.1.148.175 PrivateIPs: 172.31.12.86


Validating Apache2 service in slave 1

  [Alt+S]     Asia Pacific (Mumbai) ▾ prakash24 ▾

 EC2  EC2 Image Builder  S3  IAM  VPC  CloudFormation  Lambda  Cognito  API Gateway  CloudWatch

```
ubuntu@ip-172-31-2-175:~$ sudo service apache2 status
• apache2.service - The Apache HTTP Server
  Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enabled)
  Active: active (running) since Tue 2025-01-14 13:24:18 UTC; 2min 49s ago
    Docs: https://httpd.apache.org/docs/2.4/
  Main PID: 2544 (apache2)
    Tasks: 55 (limit: 1130)
  Memory: 5.6M
    CPU: 37ms
  CGroup: /system.slice/apache2.service
          └─2544 /usr/sbin/apache2 -k start
            └─2546 /usr/sbin/apache2 -k start
              └─2547 /usr/sbin/apache2 -k start

Jan 14 13:24:18 ip-172-31-2-175 systemd[1]: Starting The Apache HTTP Server...
Jan 14 13:24:18 ip-172-31-2-175 systemd[1]: Started The Apache HTTP Server.
ubuntu@ip-172-31-2-175:~$
```

i-04e8dfdf40c13bff (Ansible-Slave1) 
PublicIPs: 3.110.183.86 PrivateIPs: 172.31.2.175

Validating nginx service in slave 2

[illegible]

Validating Apache2 default page in slave 1



← → ↻ 🏠 ⚠ Not secure 3.110.183.86 ☆ 📁 📄 🌐 ⋮



Apache2 Default Page

Ubuntu

It works!

This is the default welcome page used to test the correct operation of the Apache2 server after installation on Ubuntu systems. It is based on the equivalent page on Debian, from which the Ubuntu Apache packaging is derived. If you can read this page, it means that the Apache HTTP server installed at this site is working properly. You should **replace this file** (located at `/var/www/html/index.html`) before continuing to operate your HTTP server.

If you are a normal user of this web site and don't know what this page is about, this probably means that the site is currently unavailable due to maintenance. If the problem persists, please contact the site's administrator.

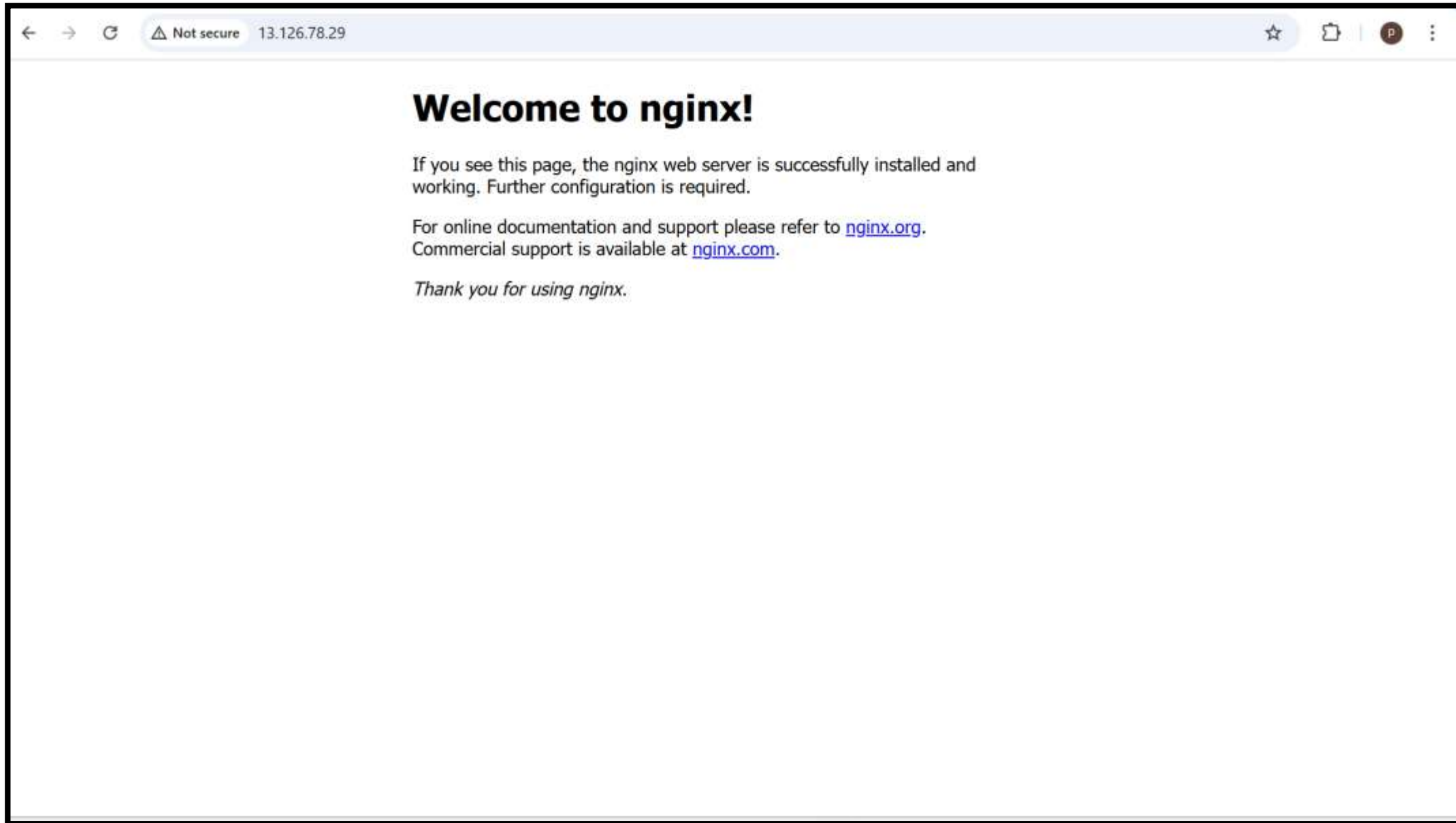
Configuration Overview

Ubuntu's Apache2 default configuration is different from the upstream default configuration, and split into several files optimized for interaction with Ubuntu tools. The configuration system is **fully documented in `/usr/share/doc/apache2/README.Debian.gz`**. Refer to this for the full documentation. Documentation for the web server itself can be found by accessing the **manual** if the `apache2-doc` package was installed on this server.

The configuration layout for an Apache2 web server installation on Ubuntu systems is as follows:

```
/etc/apache2/  
|-- apache2.conf  
|   |-- ports.conf  
|-- mods-enabled  
|   |-- *.Load  
|   |-- *.conf  
|-- conf-enabled  
|   |-- *.conf  
|-- sites-enabled
```

Validating nginx default page in slave 2



Module 5: Ansible Assignment - 4

Tasks To Be Performed:

1. Use the previous deployment of Ansible cluster
2. Configure the files folder in the role with index.html which should be replaced with the original index.html

All of the above should only happen on the slave which has NGINX installed using the role.

Creating a html page so we can replay this page using playbook

The screenshot shows an AWS CloudShell terminal window. At the top, there's a navigation bar with various AWS services like EC2, IAM, VPC, CloudFormation, Lambda, Cognito, API Gateway, and CloudWatch. Below this, the terminal window is open, showing the GNU nano 6.2 editor editing a file named index.html. The content of the file is an HTML document with a title 'Chaning default apache2 webpage' and a body containing 'This is new apache page'. The terminal window also displays a command prompt and a list of nano editor shortcuts. At the bottom, there's a status bar showing the instance ID 'i-0545735bd53b33955 (Ansible-Master)' and its public and private IP addresses. The footer of the CloudShell window includes the 'CloudShell' logo, a 'Feedback' link, and copyright information for Amazon Web Services, Inc. or its affiliates, along with links for 'Privacy', 'Terms', and 'Cookie preferences'.

Click to go back, hold to see history [Alt+S]

EC2 EC2 Image Builder S3 IAM VPC CloudFormation Lambda Cognito API Gateway CloudWatch

GNU nano 6.2 index.html

```
<html>
<title> Chaning default apache2 webpage</title>
<body> <h1> This is new apache page</h1></body>
</html>
```

[Read 4 lines]

^G Help ^O Write Out ^W Where Is ^K Cut ^T Execute ^C Location M-U Undo M-A Set Mark M-] To Bracket M-Q Previous
^X Exit ^F Read File ^_ Replace ^U Paste ^J Justify ^/_ Go To Line M-E Redo M-G Copy ^Q Where Was M-W Next

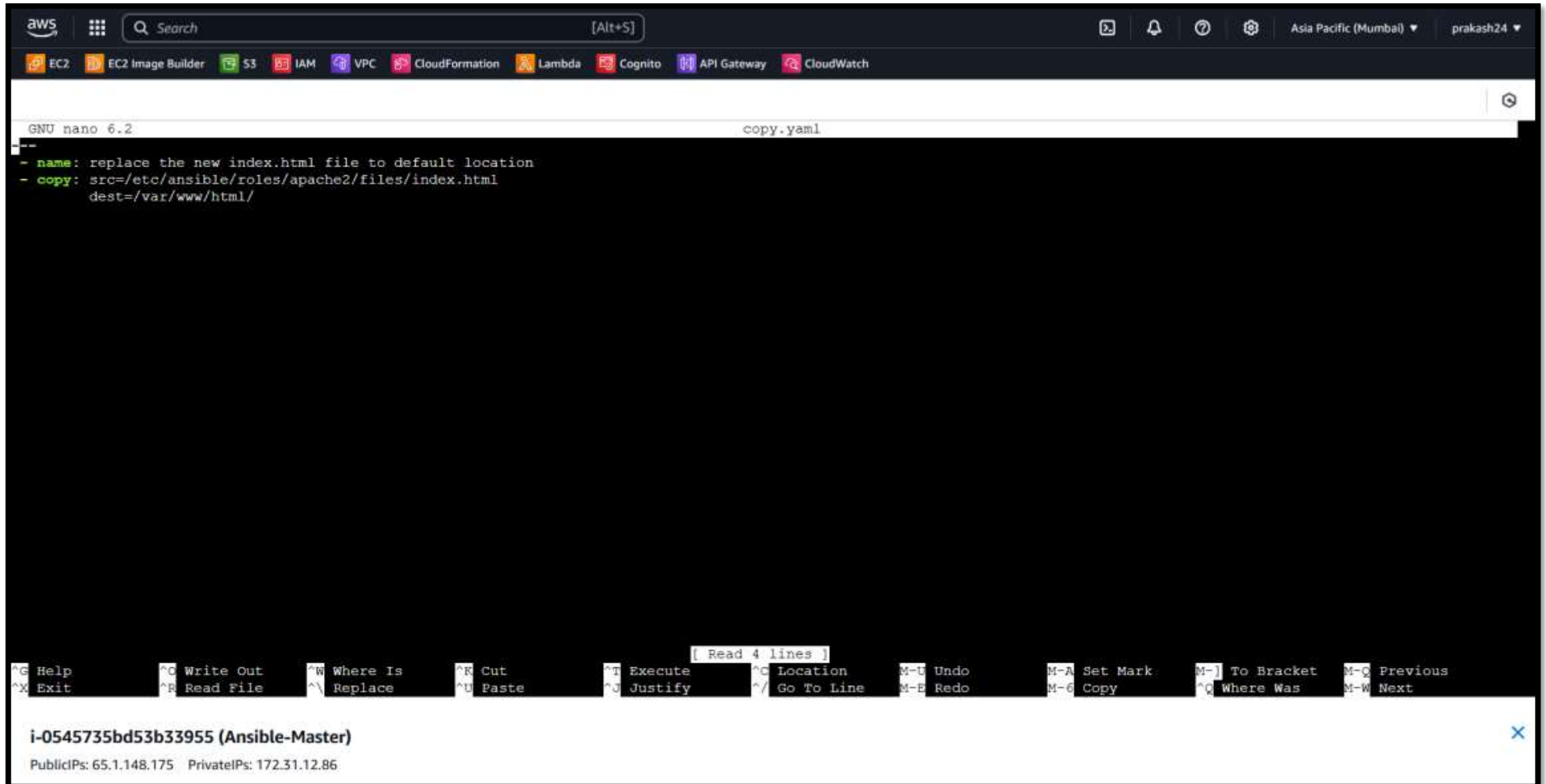
i-0545735bd53b33955 (Ansible-Master)

PublicIPs: 65.1.148.175 PrivateIPs: 172.31.12.86

CloudShell Feedback

© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Creating a playbook file for replace html file in apache2 role



The screenshot shows the AWS Management Console interface. At the top, there's a navigation bar with the AWS logo, a search bar, and a list of services including EC2, EC2 Image Builder, S3, IAM, VPC, CloudFormation, Lambda, Cognito, API Gateway, and CloudWatch. The user's profile 'prakash24' is visible in the top right corner.

The main area displays a terminal window running GNU nano 6.2. The file being edited is 'copy.yaml'. The content of the file is as follows:

```
--  
- name: replace the new index.html file to default location  
- copy: src=/etc/ansible/roles/apache2/files/index.html  
      dest=/var/www/html/
```

At the bottom of the terminal window, there's a status bar showing the instance ID 'i-0545735bd53b33955 (Ansible-Master)' and its IP addresses: PublicIPs: 65.1.148.175 and PrivateIPs: 172.31.12.86. A keyboard shortcuts menu is also visible, listing various commands like Help, Exit, Write Out, Read File, Where Is, Replace, Cut, Paste, Execute, Justify, Location, Go To Line, Undo, Redo, Set Mark, Copy, To Bracket, Where Was, Previous, and Next.



Asia Pacific (Mumbai) ▼

prakash24 ▼

EC2 EC2 Image Builder S3 IAM VPC CloudFormation Lambda Cognito API Gateway CloudWatch

GNU nano 6.2 main.yml

```
---
# tasks file for apache2
#- include_tasks: play.yaml
- include_tasks: copy.yaml
```

[Read 4 lines]

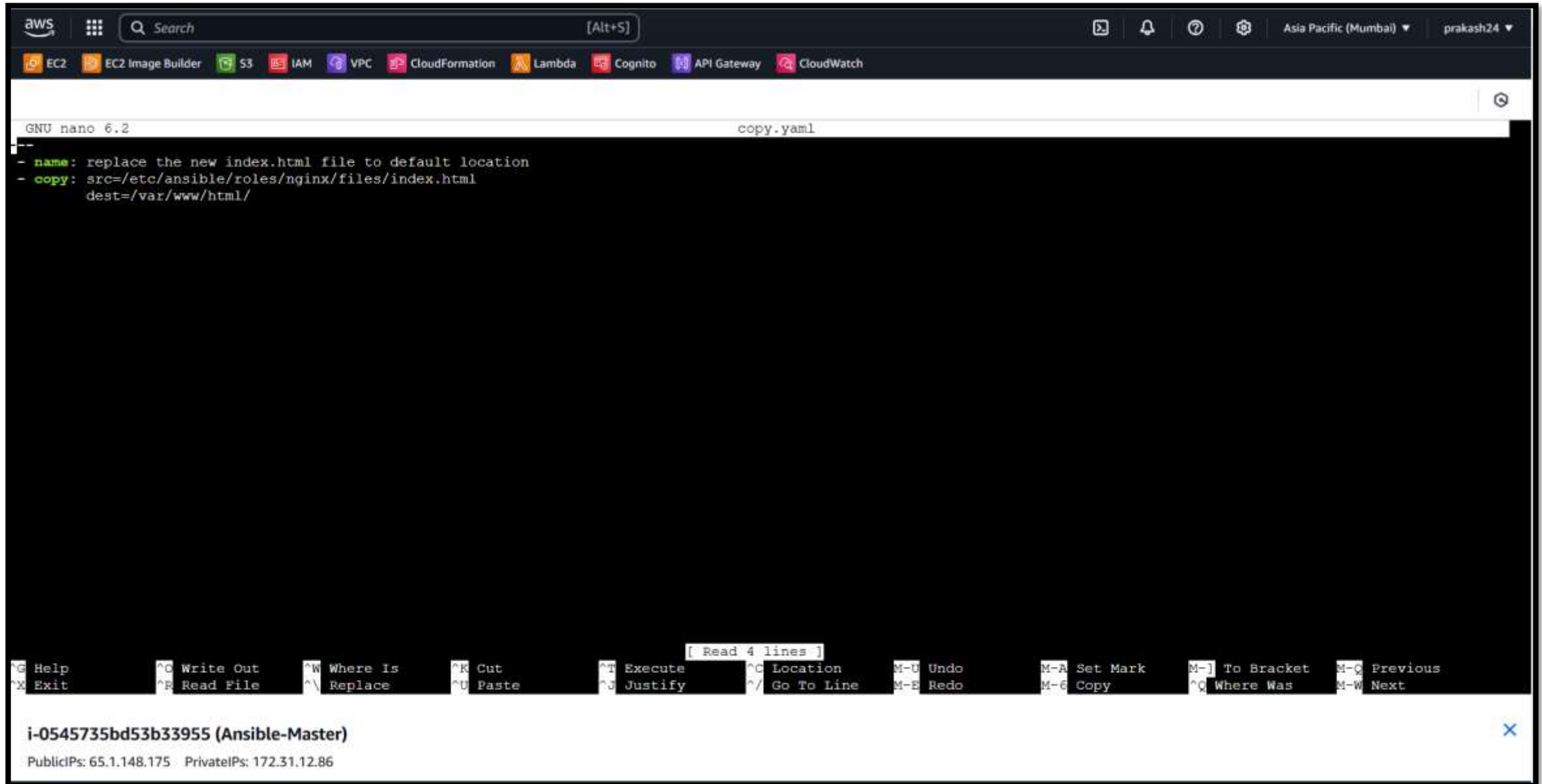
Help	Write Out	Where Is	Cut	Execute	Location	Undo	Set Mark	To Bracket	Previous
Exit	Read File	Replace	Paste	Justify	Go To Line	Redo	Copy	Where Was	Next

i-0545735bd53b33955 (Ansible-Master)

PublicIPs: 65.1.148.175 PrivateIPs: 172.31.12.86



Creating a playbook file for replace html file in nginx roles

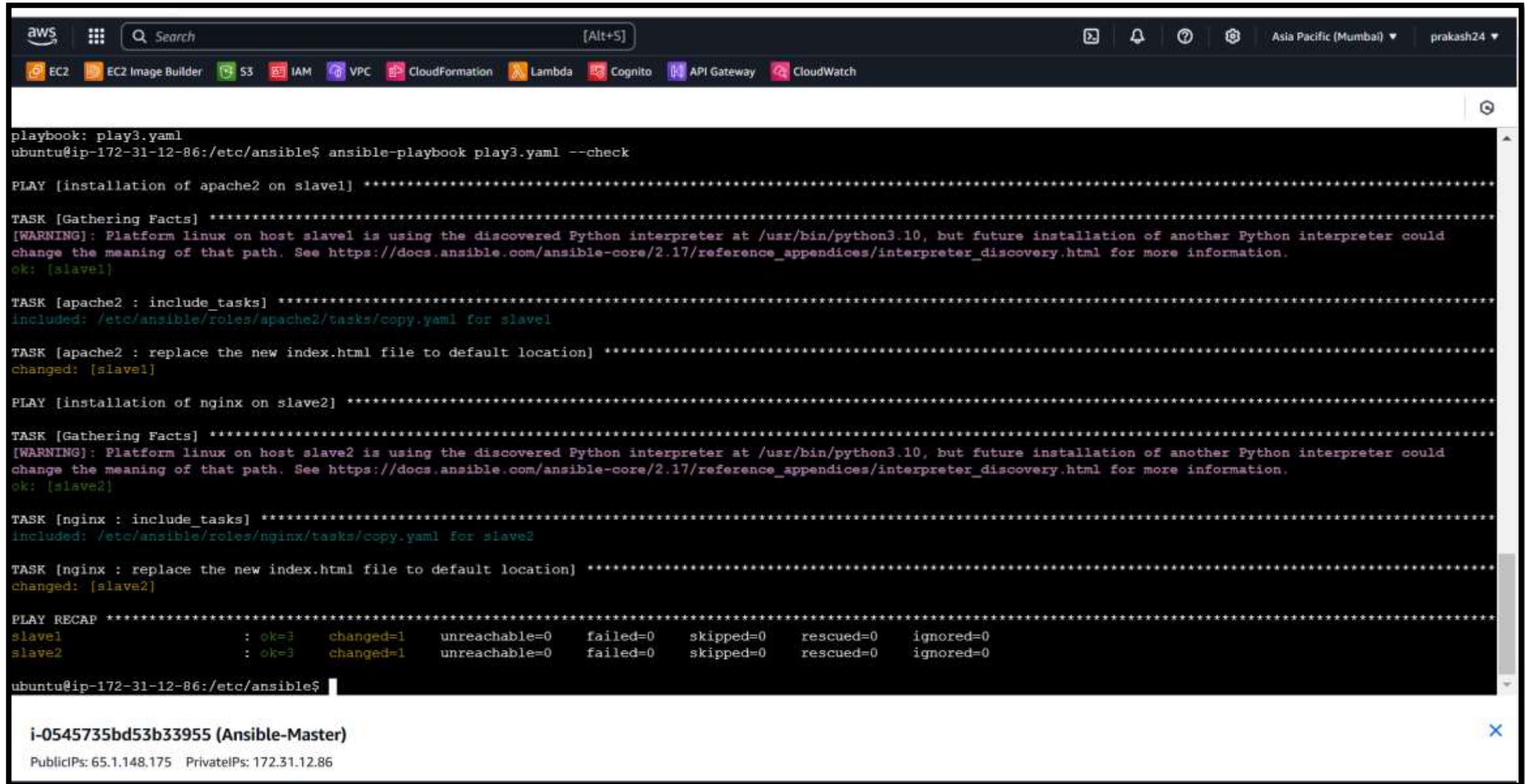


The screenshot shows the AWS Management Console interface. At the top, there's a navigation bar with the AWS logo, a search bar, and a list of services including EC2, EC2 Image Builder, S3, IAM, VPC, CloudFormation, Lambda, Cognito, API Gateway, and CloudWatch. The user's profile 'prakash24' is in the top right corner. Below the navigation bar, a terminal window is open, displaying the GNU nano 6.2 editor. The file being edited is 'copy.yaml'. The content of the file is as follows:

```
--  
- name: replace the new index.html file to default location  
- copy: src=/etc/ansible/roles/nginx/files/index.html  
      dest=/var/www/html/
```

At the bottom of the terminal window, there's a status bar showing the instance ID 'i-0545735bd53b33955 (Ansible-Master)' and its public and private IP addresses: 'PublicIPs: 65.1.148.175' and 'PrivateIPs: 172.31.12.86'. A keyboard shortcuts menu is also visible, listing various actions like Help, Exit, Write Out, Read File, Where Is, Replace, Cut, Paste, Execute, Justify, Location, Go To Line, Undo, Redo, Set Mark, Copy, To Bracket, Where Was, Previous, and Next.

Dry run of playbook & Execution of playbook



```
aws
[Alt+S]
Asia Pacific (Mumbai)
prakash24

EC2 EC2 Image Builder S3 IAM VPC CloudFormation Lambda Cognito API Gateway CloudWatch

playbook: play3.yaml
ubuntu@ip-172-31-12-86:/etc/ansible$ ansible-playbook play3.yaml --check

PLAY [installation of apache2 on slave1] *****

TASK [Gathering Facts] *****
[WARNING]: Platform linux on host slave1 is using the discovered Python interpreter at /usr/bin/python3.10, but future installation of another Python interpreter could change the meaning of that path. See https://docs.ansible.com/ansible-core/2.17/reference_appendices/interpreter_discovery.html for more information.
ok: [slave1]

TASK [apache2 : include_tasks] *****
included: /etc/ansible/roles/apache2/tasks/copy.yaml for slave1

TASK [apache2 : replace the new index.html file to default location] *****
changed: [slave1]

PLAY [installation of nginx on slave2] *****

TASK [Gathering Facts] *****
[WARNING]: Platform linux on host slave2 is using the discovered Python interpreter at /usr/bin/python3.10, but future installation of another Python interpreter could change the meaning of that path. See https://docs.ansible.com/ansible-core/2.17/reference_appendices/interpreter_discovery.html for more information.
ok: [slave2]

TASK [nginx : include_tasks] *****
included: /etc/ansible/roles/nginx/tasks/copy.yaml for slave2

TASK [nginx : replace the new index.html file to default location] *****
changed: [slave2]

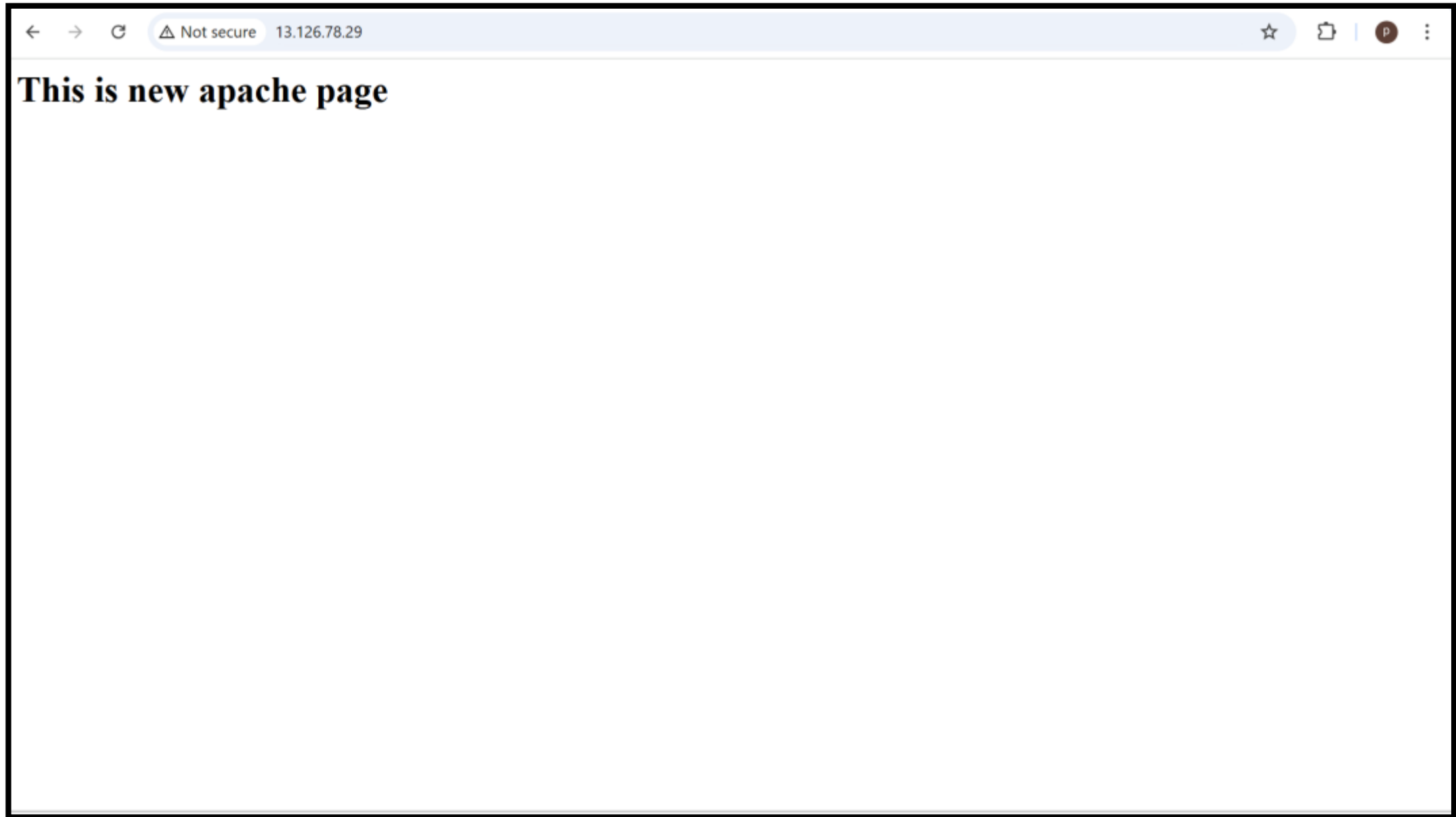
PLAY RECAP *****
slave1      : ok=3    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
slave2      : ok=3    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

ubuntu@ip-172-31-12-86:/etc/ansible$
```

i-0545735bd53b33955 (Ansible-Master)

PublicIPs: 65.1.148.175 PrivateIPs: 172.31.12.86

We Can see the Apache2 default index page got changed



Module 5: Ansible Assignment - 5

Tasks To Be Performed:

1. Create a new deployment of Ansible cluster of 5 nodes
2. Label 2 nodes as test and other 2 as prod
3. Install Java on test nodes
4. Install MySQL server on prod nodes

Use Ansible roles for the above and group the hosts under test and prod.

Launching 5 Ec2 Instance

aws

Search

[Alt+S]

EC2

EC2 Image Builder

S3

IAM

VPC

CloudFormation

Lambda

Cognito

API Gateway

CloudWatch

Asia Pacific (Mumbai)prakash24

EC2

Instances

Launch an instance

Launch an instance

Info

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

Name and tags

Info

Name

ansible-master

Add additional tags

Application and OS Images (Amazon Machine Image)

Info

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

Search our full catalog including 1000s of application and OS images

Recents

My AMIs

Quick Start

Amazon Linux

aws

macOS

Mac

Ubuntu

ubuntu

Windows

Microsoft

Red Hat

Red Hat

SUSE Linux

SUSE

Debian

debian

Browse more AMIs

Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)

Ubuntu Server 22.04 LTS (HVM), SSD Volume Type

ami-09b0a86a2c84101e1 (64-bit (x86)) / ami-0a87daabd88e93b1f (64-bit (Arm))

Virtualization: hvm ENA enabled: true Root device type: ebs

Free tier eligible

Summary

Number of instances

Info

5

When launching more than 1 instance, consider EC2 Auto Scaling

Software Image (AMI)

Canonical, Ubuntu, 22.04 LTS, ...read more

ami-09b0a86a2c84101e1

Virtual server type (instance type)

t2.micro

Firewall (security group)

default

Storage (volumes)

1 volume(s) - 8 GiB

Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 750 hours of public IPv4 address usage per month, 30 GiB of EBS storage, 2 million I/Os, 1 GB of enache, and 100 GB of bandwidth

Cancel

Launch instance

Preview code

5 Instance 1- master & 4 slave

The screenshot shows the AWS Management Console interface for the EC2 service. The main content area displays a list of 5 instances. The left sidebar contains a navigation menu with categories like Instances, Images, Elastic Block Store, and Network & Security. The top navigation bar includes the AWS logo, search bar, and various service icons.

Instances (5) Info

Last updated less than a minute ago

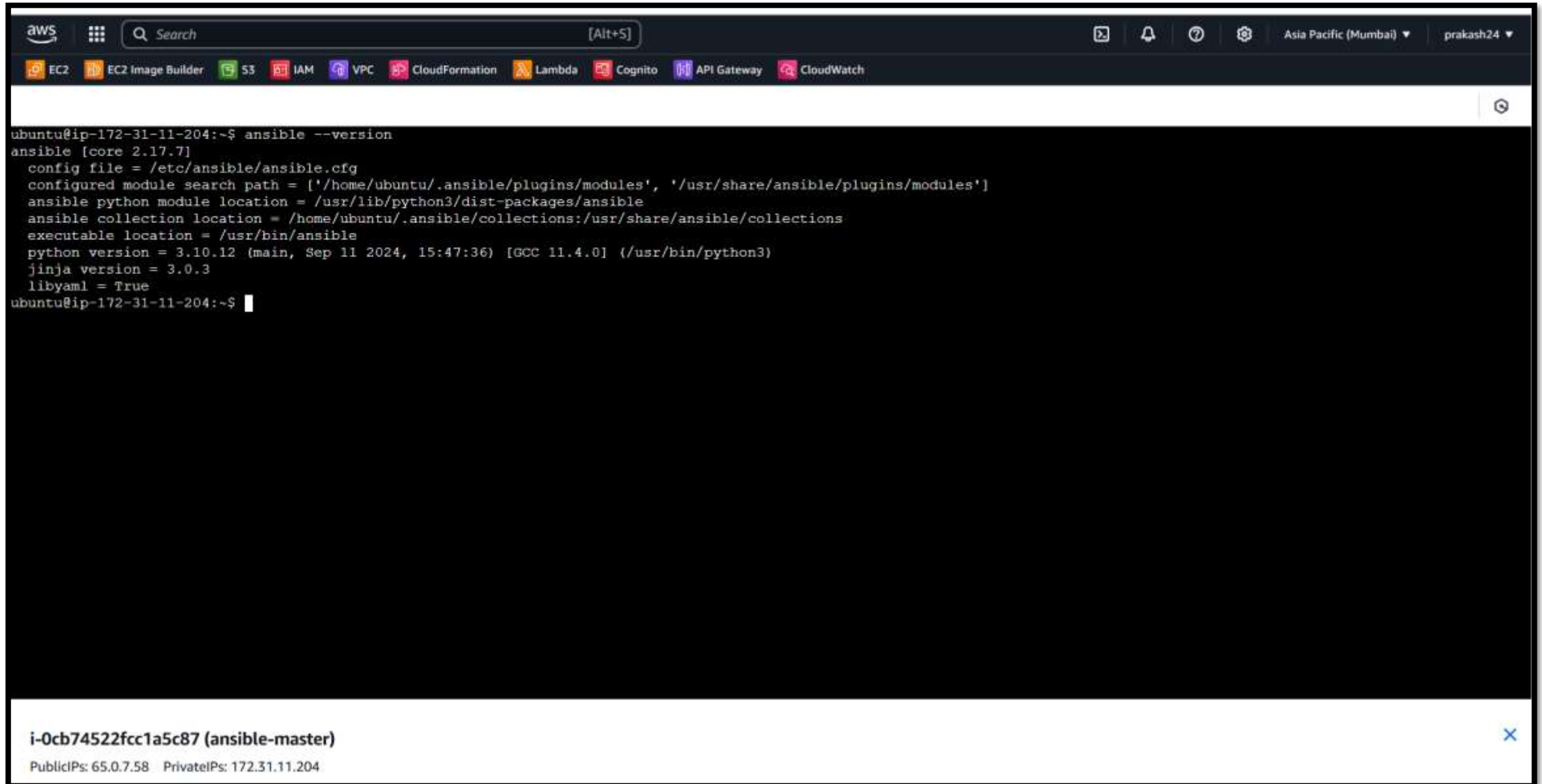
Connect Instance state Actions Launch instances

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

<input type="checkbox"/>	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Av
<input type="checkbox"/>	ansible-prod-slave2	i-0badc128f6c6c8413	Running	t2.micro	2/2 checks passed	View alarms +	ap
<input type="checkbox"/>	ansible-test-slave3	i-062db80751d6ee0e2	Running	t2.micro	2/2 checks passed	View alarms +	ap
<input type="checkbox"/>	ansible-test-slave4	i-05b4dd3a07c24d07d	Running	t2.micro	2/2 checks passed	View alarms +	ap
<input type="checkbox"/>	ansible-prod-slave1	i-0e93e84404926d369	Running	t2.micro	2/2 checks passed	View alarms +	ap
<input type="checkbox"/>	ansible-master	i-0cb74522fcc1a5c87	Running	t2.micro	2/2 checks passed	View alarms +	ap

Select an instance

Validate ansible installation in master instance



The screenshot shows the AWS Management Console interface with a terminal window open. The terminal displays the output of the command `ansible --version` executed on an Ubuntu instance. The output shows the Ansible core version is 2.17.7, the config file is `/etc/ansible/ansible.cfg`, and the configured module search path includes `/home/ubuntu/.ansible/plugins/modules` and `/usr/share/ansible/plugins/modules`. It also lists the Python module location, collection location, executable location, Python version (3.10.12), Jinja version (3.0.3), and `libyaml = True`.

```
aws
[Alt+S]
EC2 EC2 Image Builder S3 IAM VPC CloudFormation Lambda Cognito API Gateway CloudWatch

ubuntu@ip-172-31-11-204:~$ ansible --version
ansible [core 2.17.7]
  config file = /etc/ansible/ansible.cfg
  configured module search path = ['/home/ubuntu/.ansible/plugins/modules', '/usr/share/ansible/plugins/modules']
  ansible python module location = /usr/lib/python3/dist-packages/ansible
  ansible collection location = /home/ubuntu/.ansible/collections:/usr/share/ansible/collections
  executable location = /usr/bin/ansible
  python version = 3.10.12 (main, Sep 11 2024, 15:47:36) [GCC 11.4.0] (/usr/bin/python3)
  jinja version = 3.0.3
  libyaml = True
ubuntu@ip-172-31-11-204:~$
```

i-0cb74522fcc1a5c87 (ansible-master)
PublicIPs: 65.0.7.58 PrivateIPs: 172.31.11.204

Generating SSH Key for connect slave nodes

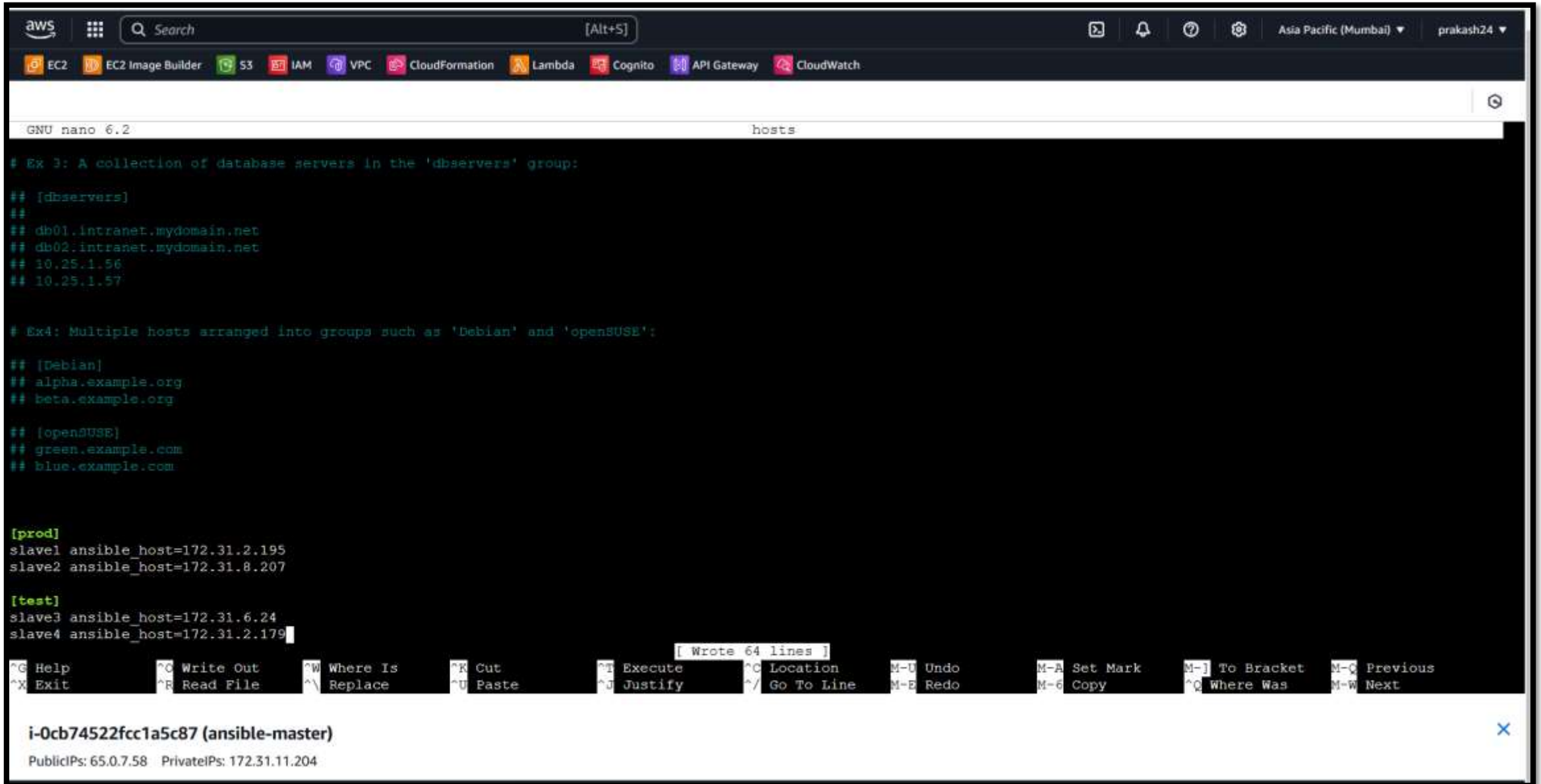
```
aws [Search] [Alt+S] Asia Pacific (Mumbai) prakash24
EC2 EC2 Image Builder S3 IAM VPC CloudFormation Lambda Cognito API Gateway CloudWatch

ubuntu@ip-172-31-11-204:~$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/ubuntu/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/ubuntu/.ssh/id_rsa
Your public key has been saved in /home/ubuntu/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:elgEzooFl9wBPstFf1Af4sH1xoVJAjpBhAVaQdFTynGQ ubuntu@ip-172-31-11-204
The key's randomart image is:
+---[RSA 3072]---+
|  .B@BE.o.+..o.|
|  .+o=O.+ = o.o|
|    o +*.* * .o|
|  . . + o..+ o .|
|  . o . S . . |
|  . o +       |
|  . o .       |
|  .           |
|  .           |
+-----[SHA256]-----+
ubuntu@ip-172-31-11-204:~$ cd .ssh
ubuntu@ip-172-31-11-204:~/.ssh$ ls
authorized_keys  id_rsa  id_rsa.pub
ubuntu@ip-172-31-11-204:~/.ssh$ sudo cat id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQGDWSTg/hLkbKesWep1MwmjRKEsmFSX990WY8a+q6EfCx4TGnnoKd+H79D+cwSp/DcdAOAKsHOZBVxu7AG4cU10kXLe8rb7wHBtK+uT3aQ1jC2HJ1SLJZdvf7fFx6XOOCV6S08AND
bV3WI5TWbOfzlc83+PyadR5EdW3ItZK5iWuM6drydbh9mQMA12znovHjsineNiiYoHIC77S5wLDYLnKeAdUN9fVe8DDEy7CPlaePFh7XT9EX5wcty8yzSv3Rbs8jH62utFh/gM4ona23xyDO8hwr9akcW29TCeBfLTmquxnQrtBxe
9As6+DqWnV6Xar4qRipxI+LMY1NunKK8ou4Ewc4nce1lWLorSs0BRFe76Jyf9PgWXEScAI5Tm/hMiJ5614UdhPXN1iB/hM+7sDhgKGRuP7A9BWdLacInYDnbpkXzPtivOWiAczh0oMqJwoQVSkd2q2KungXlvSQ6MIUyC9nBbQxKY
GmuXh98P/Xa3Z46b2C8s2msc5RItm7bM= ubuntu@ip-172-31-11-204
ubuntu@ip-172-31-11-204:~/.ssh$
```

i-0cb74522fcc1a5c87 (ansible-master)

PublicIPs: 65.0.7.58 PrivateIPs: 172.31.11.204

Binding slave1 & Slave2 as prod & slave3 & slave 4 as test on hosts using master node



The screenshot shows the AWS Management Console interface. At the top, there's a navigation bar with the AWS logo, a search bar, and various service icons like EC2, EC2 Image Builder, S3, IAM, VPC, CloudFormation, Lambda, Cognito, API Gateway, and CloudWatch. The region is set to Asia Pacific (Mumbai) and the user is prakash24.

The main content area displays a terminal window titled "GNU nano 6.2" editing a file named "hosts". The terminal content is as follows:

```
# Ex 3: A collection of database servers in the 'dbservers' group:

## [dbservers]
##
## db01.intranet.mydomain.net
## db02.intranet.mydomain.net
## 10.25.1.56
## 10.25.1.57

# Ex4: Multiple hosts arranged into groups such as 'Debian' and 'openSUSE':

## [Debian]
## alpha.example.org
## beta.example.org

## [openSUSE]
## green.example.com
## blue.example.com

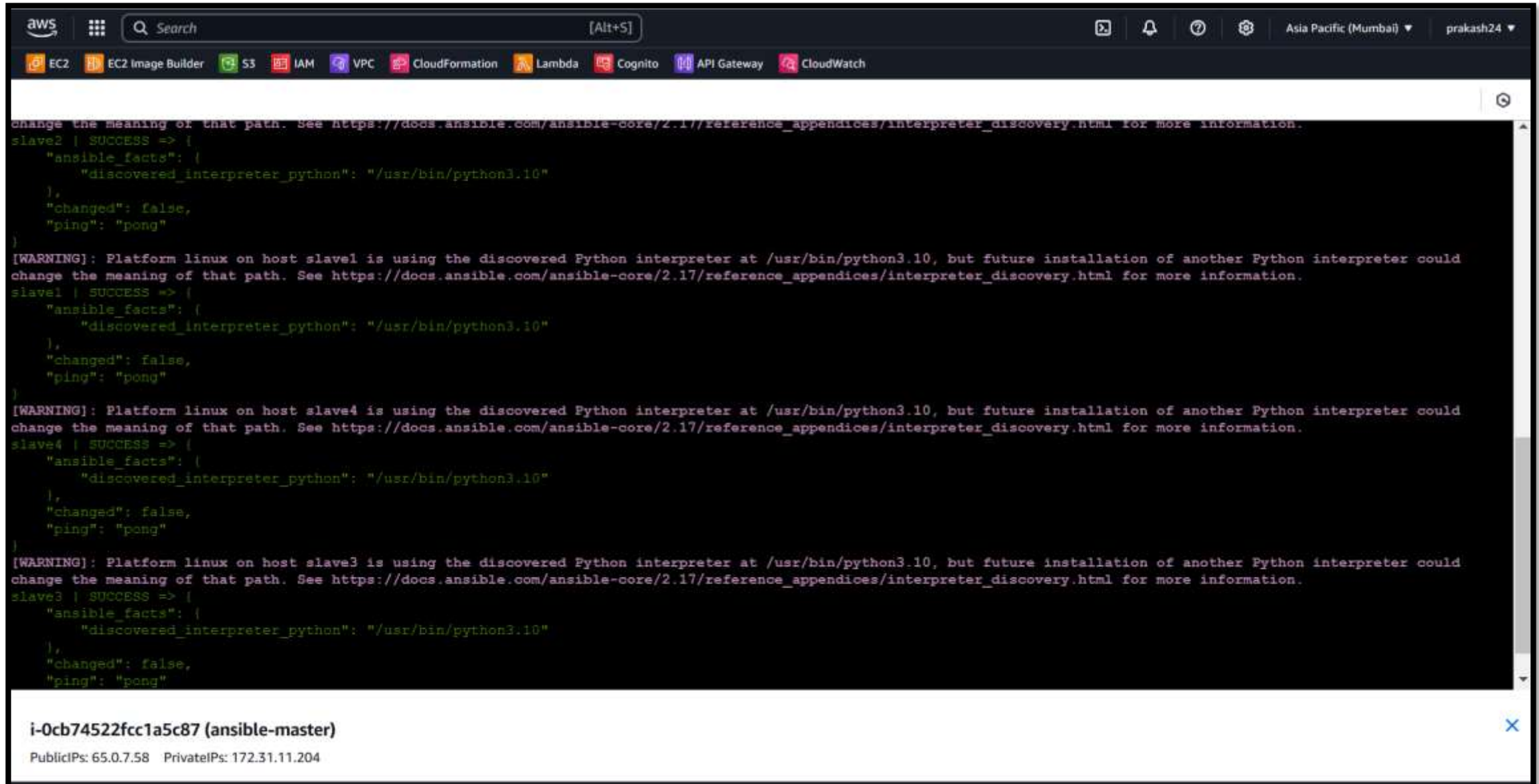
[prod]
slave1 ansible_host=172.31.2.195
slave2 ansible_host=172.31.8.207

[test]
slave3 ansible_host=172.31.6.24
slave4 ansible_host=172.31.2.179
```

Below the terminal content, there's a status bar indicating "[Wrote 64 lines]". At the bottom of the terminal window, there's a navigation bar with various keyboard shortcuts for nano editor commands.

At the bottom of the console, there's a notification for the instance "i-0cb74522fcc1a5c87 (ansible-master)" with its Public IP as 65.0.7.58 and Private IP as 172.31.11.204.

Validating all 4 slave using “ansible –m ping all” command , master node able to reach slave node or not



The screenshot shows the AWS CloudShell interface with a terminal window displaying the output of an Ansible command. The terminal output shows that all four slave nodes (slave2, slave1, slave4, and slave3) were successfully reached by the master node. Each node's output includes a success message and a warning about the Python interpreter path.

```
change the meaning of that path. See https://docs.ansible.com/ansible-core/2.17/reference_appendices/interpreter_discovery.html for more information.
slave2 | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3.10"
  },
  "changed": false,
  "ping": "pong"
}
[WARNING]: Platform linux on host slave1 is using the discovered Python interpreter at /usr/bin/python3.10, but future installation of another Python interpreter could
change the meaning of that path. See https://docs.ansible.com/ansible-core/2.17/reference_appendices/interpreter_discovery.html for more information.
slave1 | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3.10"
  },
  "changed": false,
  "ping": "pong"
}
[WARNING]: Platform linux on host slave4 is using the discovered Python interpreter at /usr/bin/python3.10, but future installation of another Python interpreter could
change the meaning of that path. See https://docs.ansible.com/ansible-core/2.17/reference_appendices/interpreter_discovery.html for more information.
slave4 | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3.10"
  },
  "changed": false,
  "ping": "pong"
}
[WARNING]: Platform linux on host slave3 is using the discovered Python interpreter at /usr/bin/python3.10, but future installation of another Python interpreter could
change the meaning of that path. See https://docs.ansible.com/ansible-core/2.17/reference_appendices/interpreter_discovery.html for more information.
slave3 | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3.10"
  },
  "changed": false,
  "ping": "pong"
}
```

i-0cb74522fcc1a5c87 (ansible-master)
PublicIPs: 65.0.7.58 PrivateIPs: 172.31.11.204

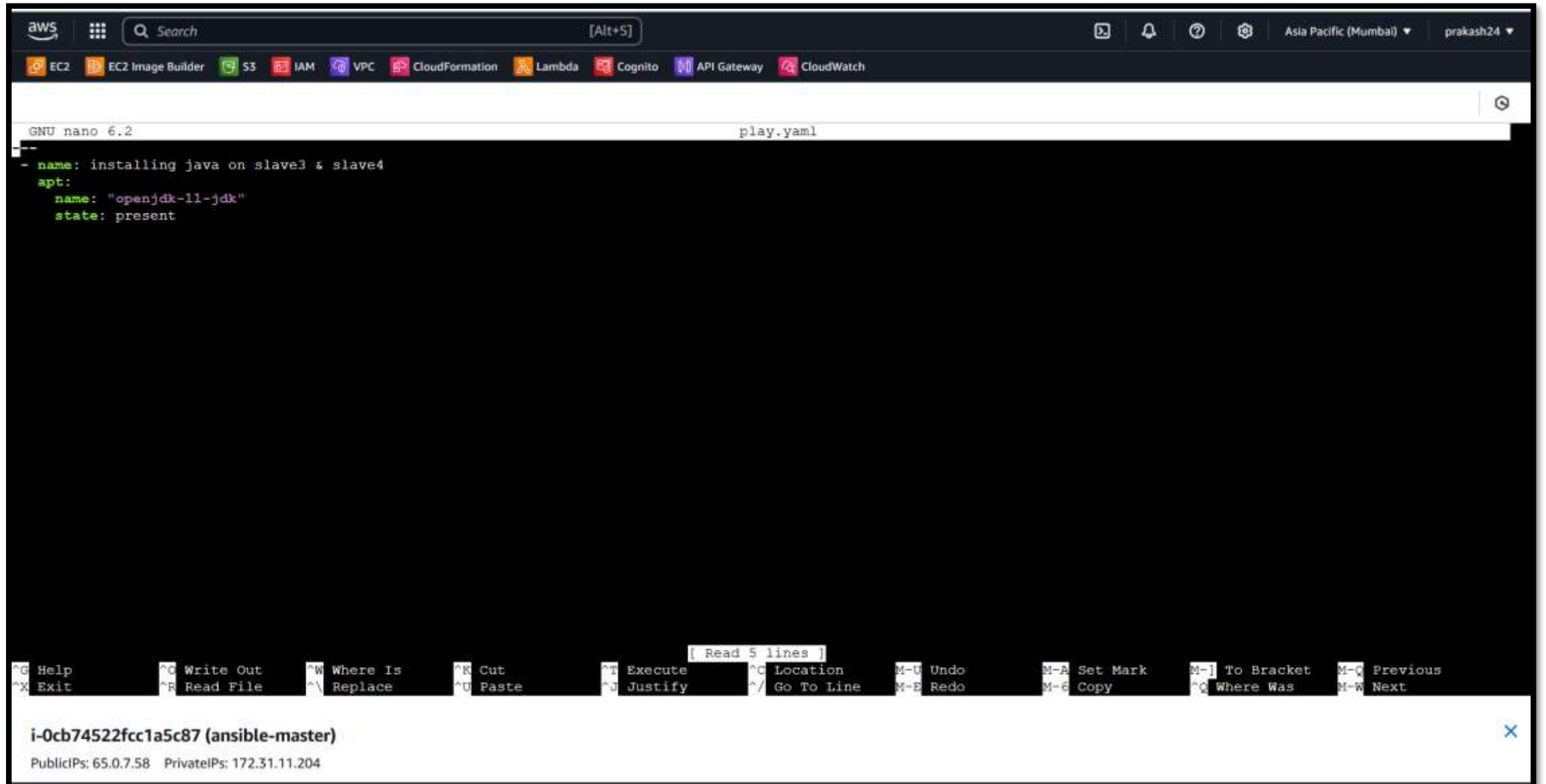
Created two roles Java & MySql

```
ubuntu@ip-172-31-11-204:~$ cd /etc/ansible
ubuntu@ip-172-31-11-204:/etc/ansible$ ls
ansible.cfg  hosts  roles
ubuntu@ip-172-31-11-204:/etc/ansible$ cd roles
ubuntu@ip-172-31-11-204:/etc/ansible/roles$ ls
ubuntu@ip-172-31-11-204:/etc/ansible/roles$ ls
ubuntu@ip-172-31-11-204:/etc/ansible/roles$ sudo ansible-galaxy init Java
- Role Java was created successfully
ubuntu@ip-172-31-11-204:/etc/ansible/roles$ sudo ansible-galaxy init MySql
- Role MySql was created successfully
ubuntu@ip-172-31-11-204:/etc/ansible/roles$ ls
Java  MySql
ubuntu@ip-172-31-11-204:/etc/ansible/roles$
```

i-0cb74522fcc1a5c87 (ansible-master)

PublicIPs: 65.0.7.58 PrivateIPs: 172.31.11.204

Created playbook file for installation of Java in test

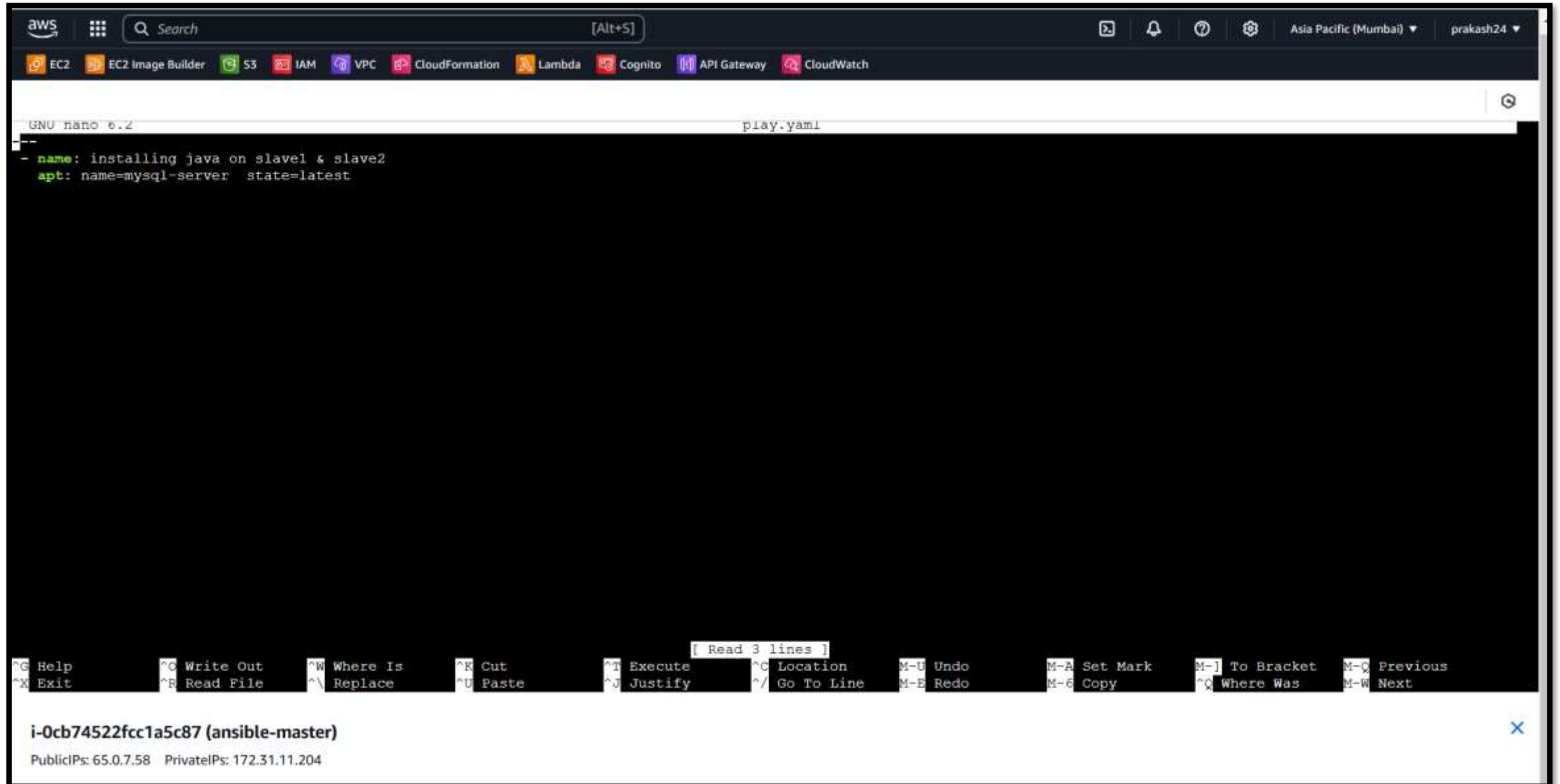


The screenshot shows the AWS Management Console interface. At the top, there's a navigation bar with the AWS logo, a search bar, and a list of services including EC2, EC2 Image Builder, S3, IAM, VPC, CloudFormation, Lambda, Cognito, API Gateway, and CloudWatch. The user is logged in as 'prakash24' in the 'Asia Pacific (Mumbai)' region. Below the navigation bar, a terminal window titled 'GNU nano 6.2' is open, editing a file named 'play.yaml'. The content of the file is an Ansible playbook with the following structure:

```
---
- name: installing java on slave3 & slave4
  apt:
    name: "openjdk-11-jdk"
    state: present
```

At the bottom of the terminal window, there's a status bar showing the instance ID 'i-0cb74522fcc1a5c87 (ansible-master)' and its IP addresses: 'PublicIPs: 65.0.7.58 PrivateIPs: 172.31.11.204'. A keyboard shortcuts menu is also visible, listing various commands like Help, Exit, Write Out, Read File, Where Is, Replace, Cut, Paste, Execute, Justify, Location, Go To Line, Undo, Redo, Set Mark, Copy, To Bracket, Where Was, Previous, and Next.

Created playbook file for installation of MySql in Prod



```
GNU nano 6.2 play.yaml
--
- name: installing java on slave1 & slave2
  apt: name=mysql-server state=latest
```

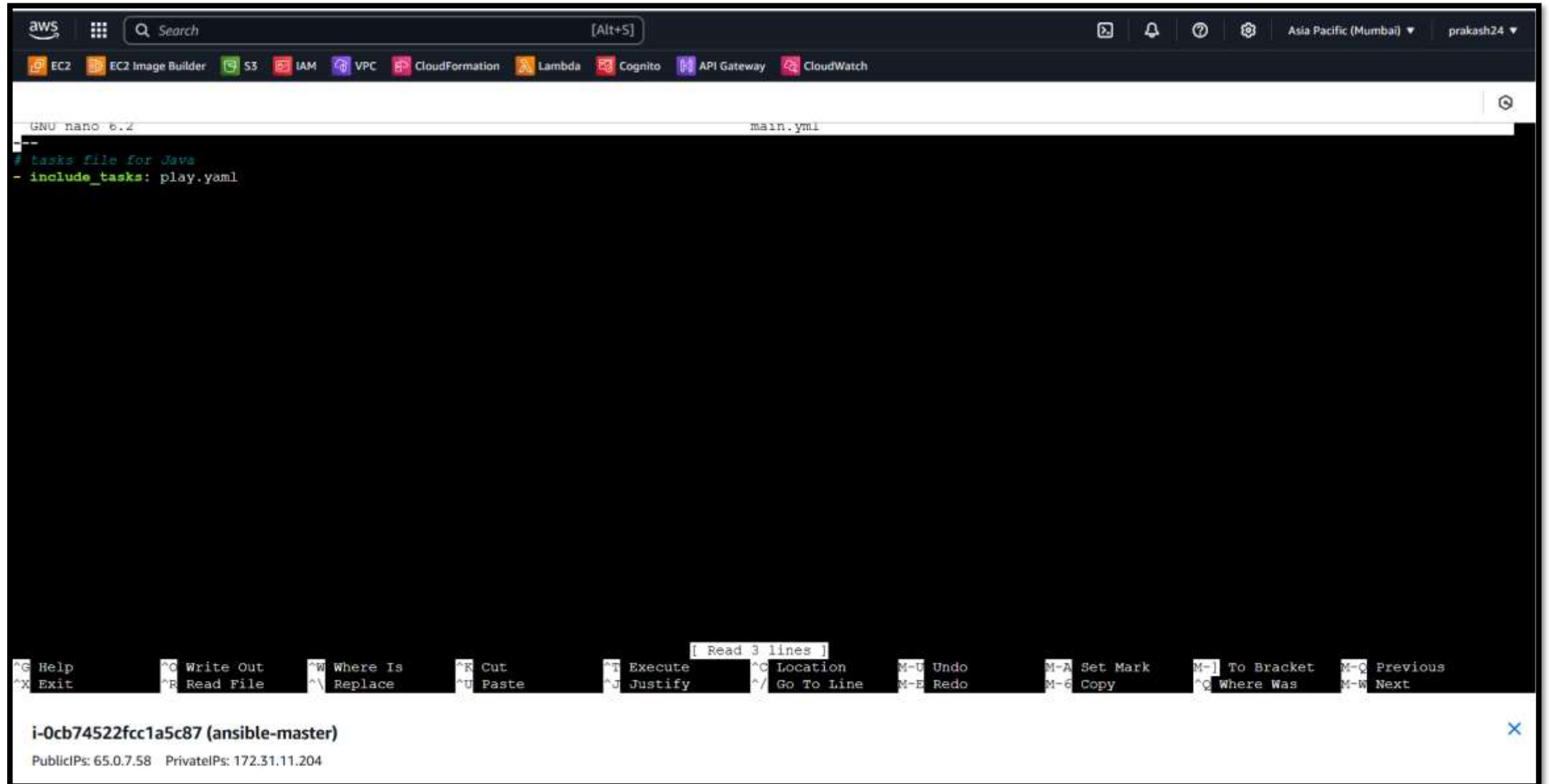
[Read 3 lines]

^G Help	^O Write Out	^W Where Is	^K Cut	^T Execute	^C Location	M-U Undo	M-A Set Mark	M-] To Bracket	M-Q Previous
^X Exit	^R Read File	^_\ Replace	^U Paste	^J Justify	^/ Go To Line	M-E Redo	M-G Copy	^Q Where Was	M-W Next

i-0cb74522fcc1a5c87 (ansible-master)

PublicIPs: 65.0.7.58 PrivateIPs: 172.31.11.204

Including play.yaml file in main.yaml file



```
aws [Alt+S] Asia Pacific (Mumbai) prakash24
```

EC2 EC2 Image Builder S3 IAM VPC CloudFormation Lambda Cognito API Gateway CloudWatch

GNU nano 6.2 main.yaml

```
---
# tasks file for Java
- include_tasks: play.yaml
```

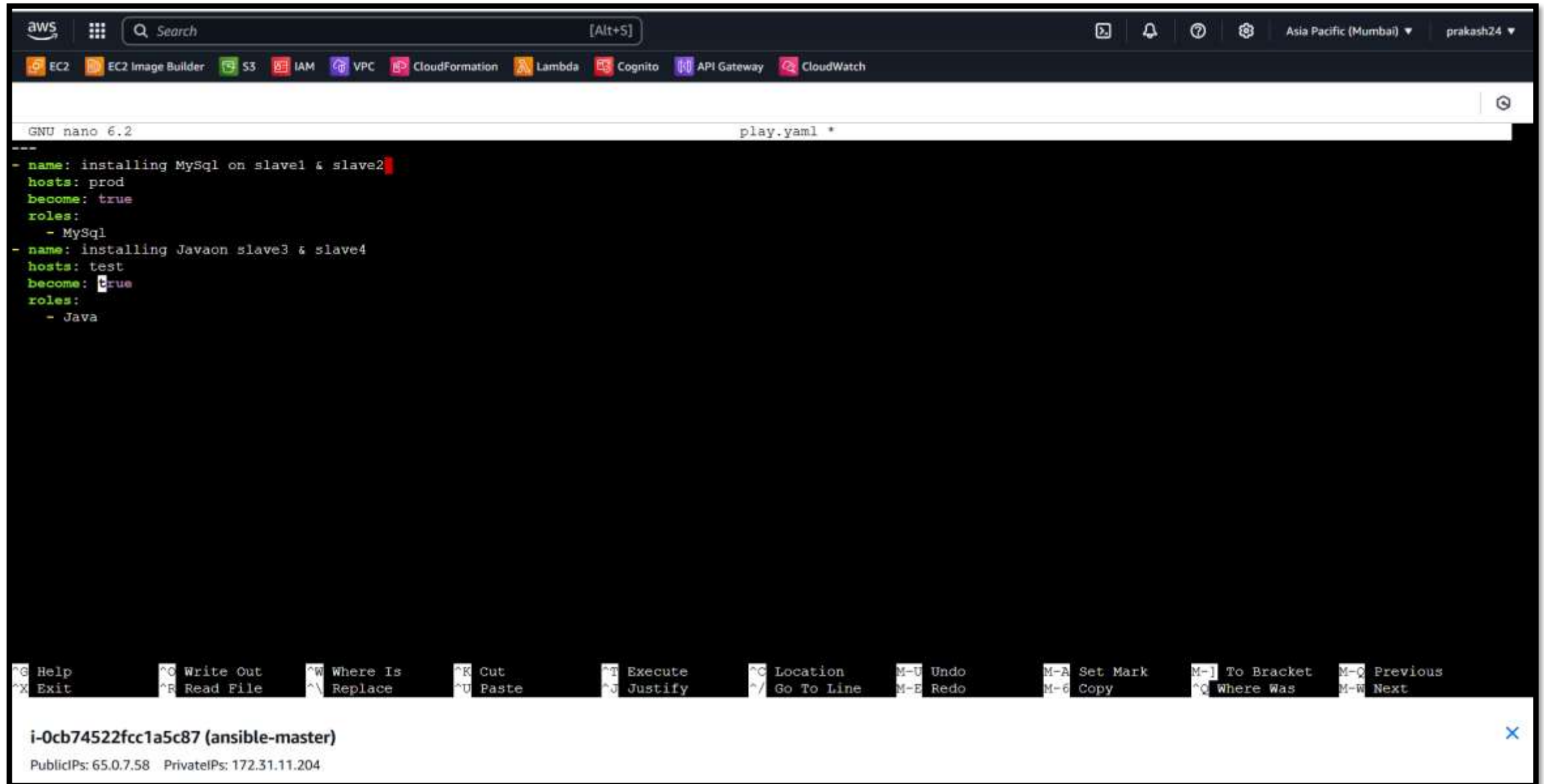
[Read 3 lines]

^G Help	^O Write Out	^W Where Is	^K Cut	^T Execute	^C Location	M-U Undo	M-A Set Mark	M-l To Bracket	M-Q Previous
^X Exit	^R Read File	^_ Replace	^U Paste	^J Justify	^/_ Go To Line	M-E Redo	M-6 Copy	^Q Where Was	M-W Next

i-0cb74522fcc1a5c87 (ansible-master)

PublicIPs: 65.0.7.58 PrivateIPs: 172.31.11.204

Creating a playbook file to run roles



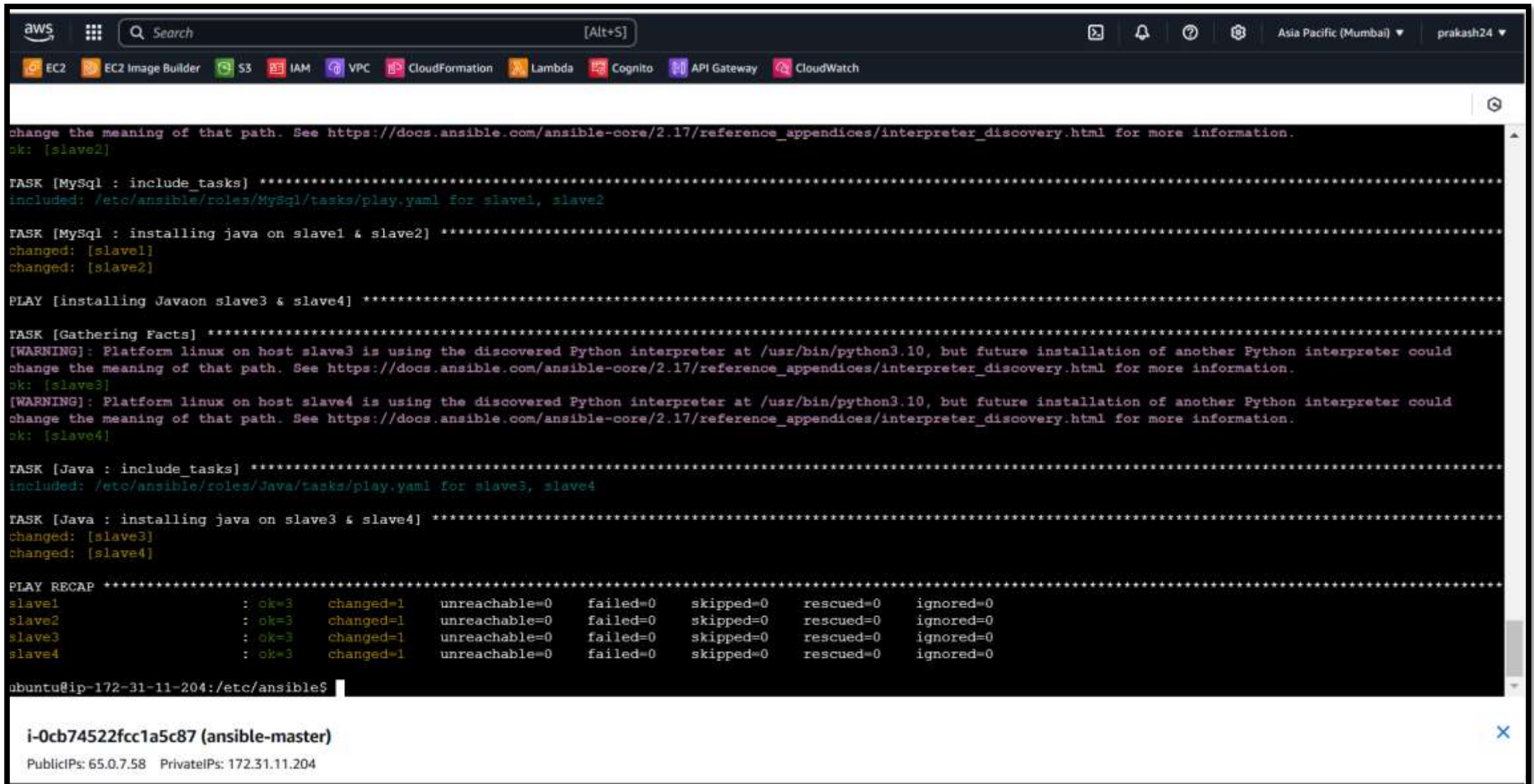
The screenshot shows the AWS Management Console interface. At the top, there's a navigation bar with the AWS logo, a search bar, and a list of services including EC2, EC2 Image Builder, S3, IAM, VPC, CloudFormation, Lambda, Cognito, API Gateway, and CloudWatch. The user is logged in as 'prakash24' in the 'Asia Pacific (Mumbai)' region. Below the navigation bar, a terminal window is open, displaying the GNU nano 6.2 editor. The editor is editing a file named 'play.yaml'. The content of the file is an Ansible playbook with two plays. The first play is named 'installing MySql on slave1 & slave2' and targets the 'prod' hosts. It sets 'become: true' and includes the 'MySql' role. The second play is named 'installing Javaon slave3 & slave4' and targets the 'test' hosts. It also sets 'become: true' and includes the 'Java' role. At the bottom of the terminal window, there's a status bar showing the instance ID 'i-0cb74522fcc1a5c87 (ansible-master)' and its public and private IP addresses: 'PublicIPs: 65.0.7.58 PrivateIPs: 172.31.11.204'.

```
GNU nano 6.2 play.yaml *
---
- name: installing MySql on slave1 & slave2
  hosts: prod
  become: true
  roles:
    - MySql
- name: installing Javaon slave3 & slave4
  hosts: test
  become: true
  roles:
    - Java
```

Help Write Out Where Is Cut Execute Location M-U Undo M-A Set Mark M-l To Bracket M-Q Previous
Exit Read File Replace Paste Justify Go To Line M-E Redo M-6 Copy M-Q Where Was M-W Next

i-0cb74522fcc1a5c87 (ansible-master)
PublicIPs: 65.0.7.58 PrivateIPs: 172.31.11.204

Running playbook file in master node



```
aws [Search] [Alt+S] Asia Pacific (Mumbai) prakash24
EC2 EC2 Image Builder S3 IAM VPC CloudFormation Lambda Cognito API Gateway CloudWatch

change the meaning of that path. See https://docs.ansible.com/ansible-core/2.17/reference_appendices/interpreter_discovery.html for more information.
ok: [slave2]

TASK [MySQL : include_tasks] *****
included: /etc/ansible/roles/MySQL/tasks/play.yaml for slave1, slave2

TASK [MySQL : installing java on slave1 & slave2] *****
changed: [slave1]
changed: [slave2]

PLAY [installing Javaon slave3 & slave4] *****

TASK [Gathering Facts] *****
[WARNING]: Platform linux on host slave3 is using the discovered Python interpreter at /usr/bin/python3.10, but future installation of another Python interpreter could
change the meaning of that path. See https://docs.ansible.com/ansible-core/2.17/reference_appendices/interpreter_discovery.html for more information.
ok: [slave3]
[WARNING]: Platform linux on host slave4 is using the discovered Python interpreter at /usr/bin/python3.10, but future installation of another Python interpreter could
change the meaning of that path. See https://docs.ansible.com/ansible-core/2.17/reference_appendices/interpreter_discovery.html for more information.
ok: [slave4]

TASK [Java : include_tasks] *****
included: /etc/ansible/roles/Java/tasks/play.yaml for slave3, slave4

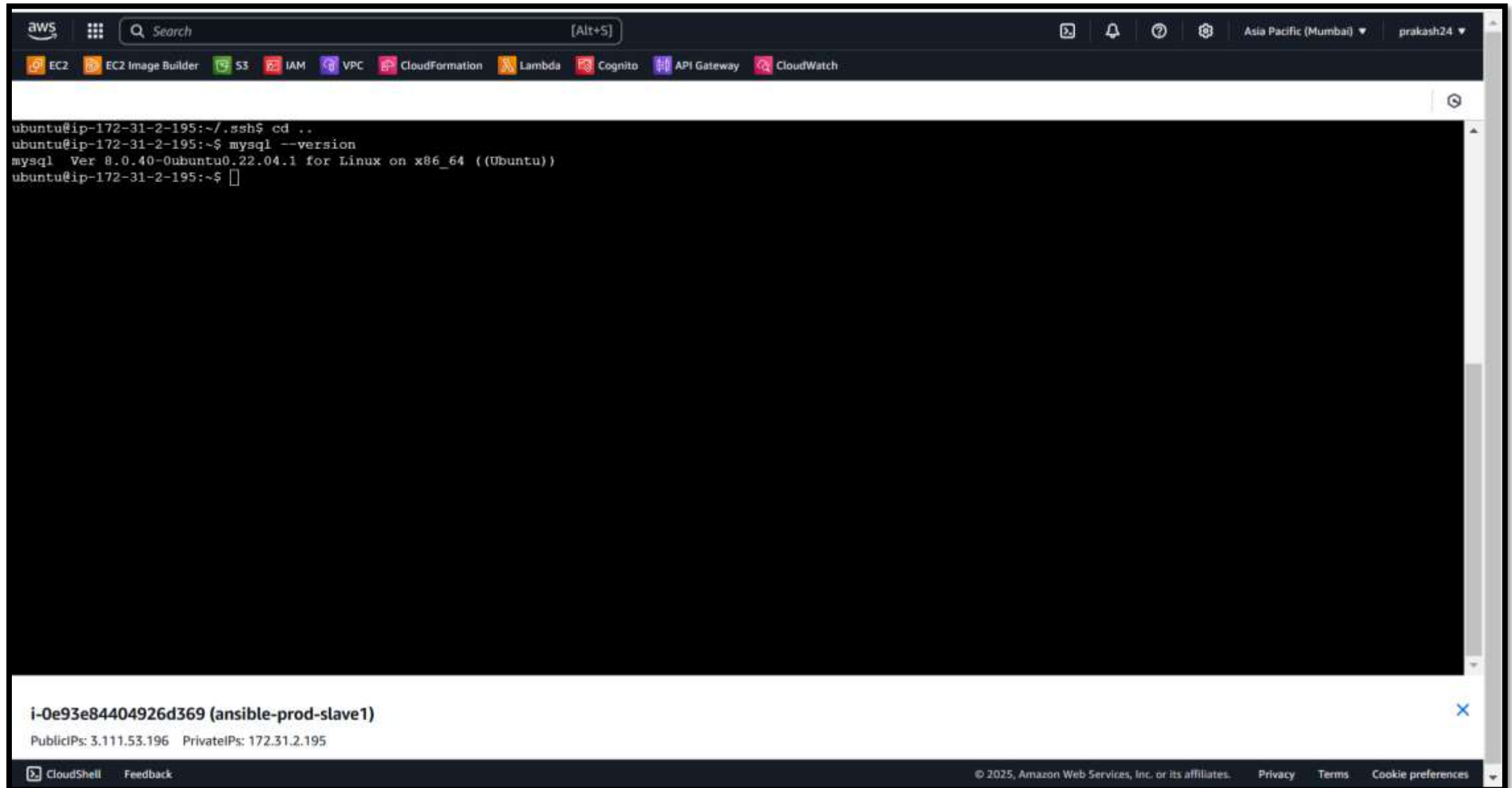
TASK [Java : installing java on slave3 & slave4] *****
changed: [slave3]
changed: [slave4]

PLAY RECAP *****
slave1      : ok=3    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
slave2      : ok=3    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
slave3      : ok=3    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
slave4      : ok=3    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

ubuntu@ip-172-31-11-204:/etc/ansible$
```

i-0cb74522fcc1a5c87 (ansible-master)
PublicIPs: 65.0.7.58 PrivateIPs: 172.31.11.204

Validating MySQL version in slave1



The screenshot displays the AWS CloudShell interface. At the top, there is a navigation bar with the AWS logo, a search bar, and various service icons including EC2, EC2 Image Builder, S3, IAM, VPC, CloudFormation, Lambda, Cognito, API Gateway, and CloudWatch. The region is set to Asia Pacific (Mumbai) and the user is prakash24.

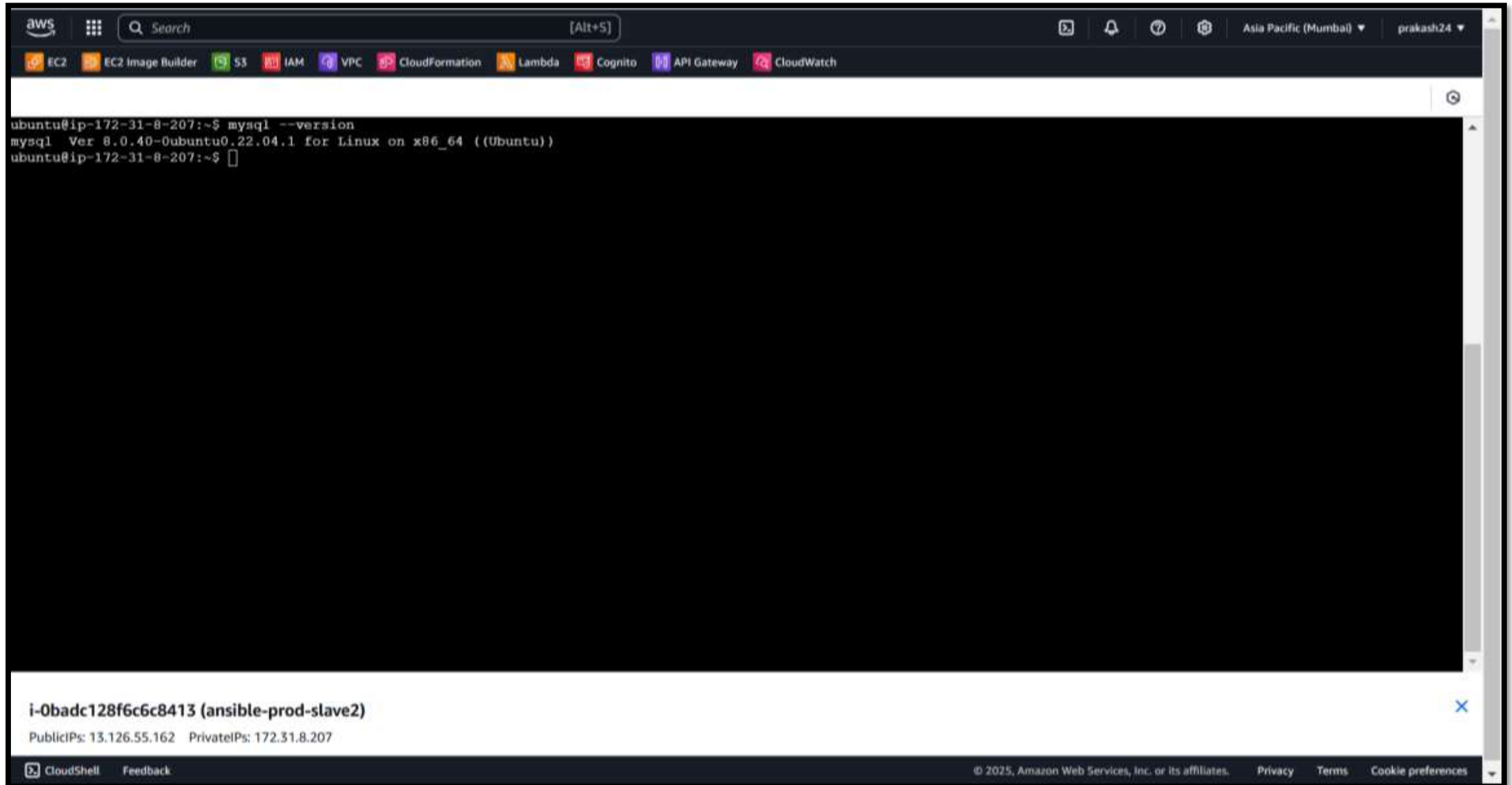
The terminal window shows the following commands and output:

```
ubuntu@ip-172-31-2-195:~/.ssh$ cd ..
ubuntu@ip-172-31-2-195:~$ mysql --version
mysql Ver 8.0.40-0ubuntu0.22.04.1 for Linux on x86_64 ((Ubuntu))
ubuntu@ip-172-31-2-195:~$
```

At the bottom of the terminal window, the instance ID **i-0e93e84404926d369 (ansible-prod-slave1)** is displayed, along with its PublicIPs (3.111.53.196) and PrivateIPs (172.31.2.195).

The footer of the CloudShell interface includes the CloudShell logo, a feedback link, and copyright information: © 2025, Amazon Web Services, Inc. or its affiliates. It also provides links for Privacy, Terms, and Cookie preferences.

Validating MySQL version in slave2



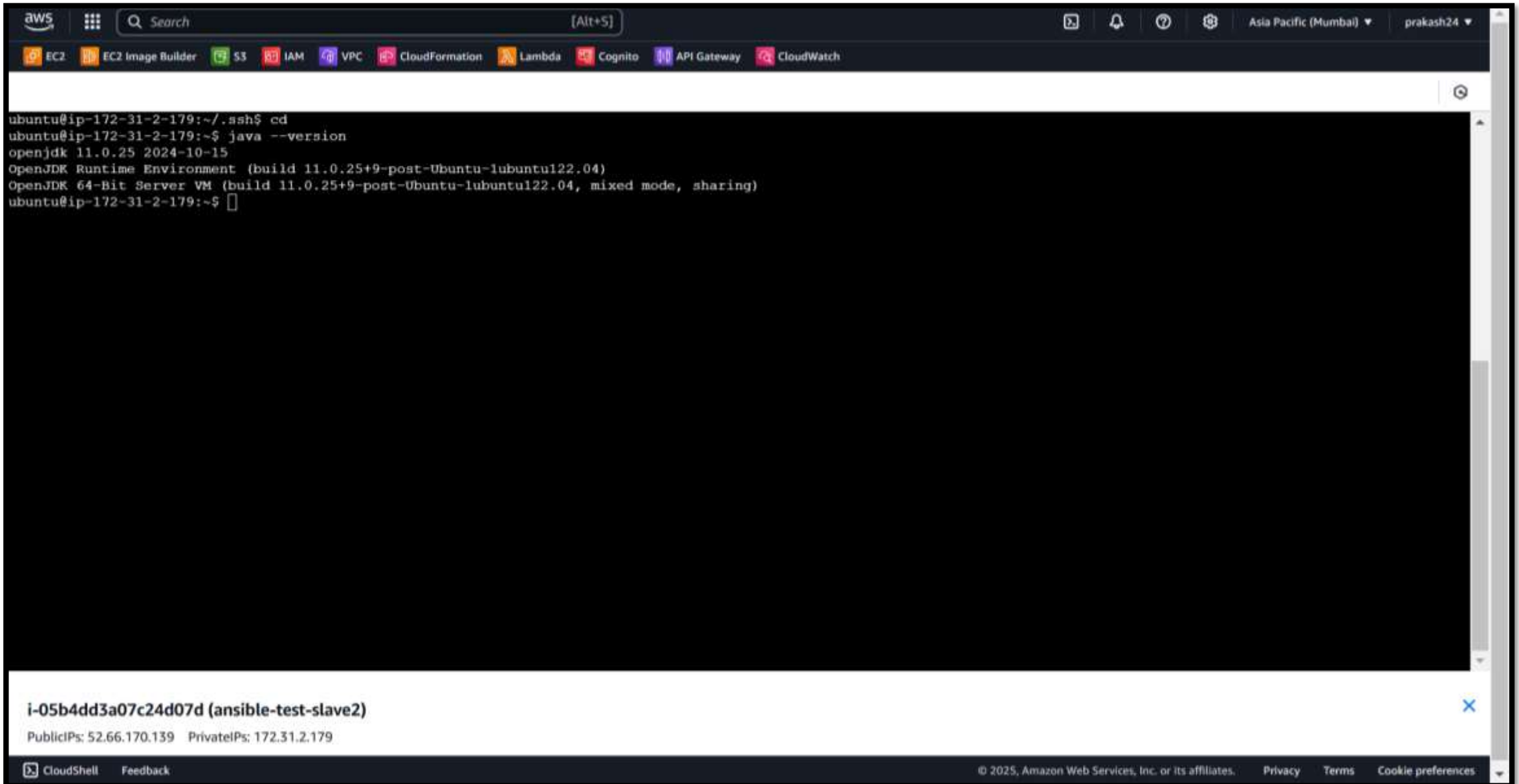
The screenshot displays the AWS CloudShell interface. At the top, there is a navigation bar with the AWS logo, a search bar, and a list of services including EC2, EC2 Image Builder, S3, IAM, VPC, CloudFormation, Lambda, Cognito, API Gateway, and CloudWatch. The region is set to Asia Pacific (Mumbai) and the user is prakash24. The main terminal area shows the command `mysql --version` being executed, resulting in the output: `mysql Ver 8.0.40-0ubuntu0.22.04.1 for Linux on x86_64 ((Ubuntu))`. Below the terminal, a summary box for instance `i-0badc128f6c6c8413 (ansible-prod-slave2)` shows public and private IP addresses. The footer contains the CloudShell logo, a feedback link, and copyright information for Amazon Web Services.

```
ubuntu@ip-172-31-8-207:~$ mysql --version
mysql Ver 8.0.40-0ubuntu0.22.04.1 for Linux on x86_64 ((Ubuntu))
ubuntu@ip-172-31-8-207:~$
```

i-0badc128f6c6c8413 (ansible-prod-slave2)
PublicIPs: 13.126.55.162 PrivateIPs: 172.31.8.207

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Validating Java version in slave3



The screenshot shows the AWS CloudShell interface. At the top, there's a navigation bar with the AWS logo, a search bar, and a list of services: EC2, EC2 Image Builder, S3, IAM, VPC, CloudFormation, Lambda, Cognito, API Gateway, and CloudWatch. The region is set to Asia Pacific (Mumbai) and the user is prakash24. The terminal window shows the following commands and output:

```
ubuntu@ip-172-31-2-179:~/.ssh$ cd
ubuntu@ip-172-31-2-179:~$ java --version
openjdk 11.0.25 2024-10-15
OpenJDK Runtime Environment (build 11.0.25+9-post-Ubuntu-1ubuntu122.04)
OpenJDK 64-Bit Server VM (build 11.0.25+9-post-Ubuntu-1ubuntu122.04, mixed mode, sharing)
ubuntu@ip-172-31-2-179:~$
```

At the bottom of the terminal window, there's a status bar for the instance **i-05b4dd3a07c24d07d (ansible-test-slave2)**, showing Public IPs: 52.66.170.139 and Private IPs: 172.31.2.179. The footer of the CloudShell interface includes the CloudShell logo, a feedback link, and copyright information: © 2025, Amazon Web Services, Inc. or its affiliates. It also has links for Privacy, Terms, and Cookie preferences.

Validating Java version in slave4

The screenshot displays the AWS Management Console interface. At the top, there's a navigation bar with the AWS logo, a search bar, and a list of services including EC2, EC2 Image Builder, S3, IAM, VPC, CloudFormation, Lambda, Cognito, API Gateway, and CloudWatch. The region is set to Asia Pacific (Mumbai) and the user is prakash24.

The main content area shows a terminal window for an EC2 instance. The terminal output is as follows:

```
ibuntu@ip-172-31-6-24:~/.ssh$ cd ..
ibuntu@ip-172-31-6-24:~$ java --version
openjdk 11.0.25 2024-10-15
openJDK Runtime Environment (build 11.0.25+9-post-Ubuntu-1ubuntu122.04)
openJDK 64-Bit Server VM (build 11.0.25+9-post-Ubuntu-1ubuntu122.04, mixed mode, sharing)
ibuntu@ip-172-31-6-24:~$
```

The bottom status bar of the console shows the instance ID **i-062db80751d6ee0e2 (ansible-test-slave1)** and its IP addresses: Public IPs: 15.207.247.137, Private IPs: 172.31.6.24.