Project Documentation: EC2 Amazon Linux 2 Instances Setup for Jenkins, Ansible, and Web Server

1. Introduction

This project involves setting up and configuring three EC2 Amazon Linux 2 instances:
- Jenkins Server**: For continuous integration and deployment.
- Ansible Server**: For automation and configuration management.
- Web Server**: To serve content on port 8080.

These instances are created using the AWS EC2 Free Tier selections and corresponding security group settings to ensure proper security and functionality.

2. Setting Up EC2 Amazon Linux 2 Instances

2.1. Create EC2 Instances

Follow these steps:

1. Sign in to AWS Management Console.

2. Navigate to **EC2 > Launch Instances**.

3. Choose **Amazon Linux 2 AMI** (Free Tier eligible).

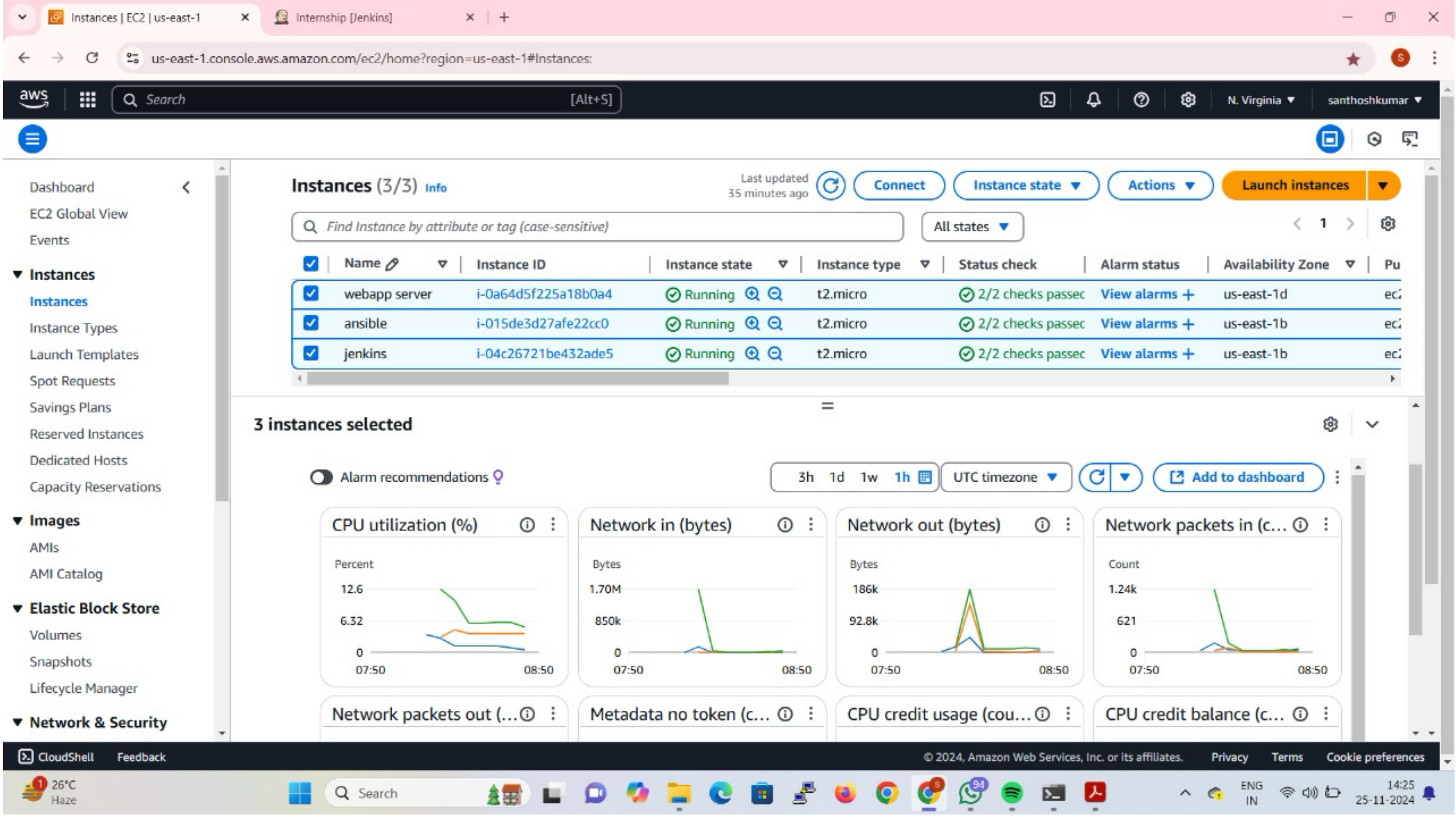
4. Select **t2.micro instance type** (Free Tier eligible).

5. Configure instance details (default settings for basic setups).

6. Add storage (8 GB of General Purpose SSD is sufficient).

7. Configure **Security Groups** (details provided below).

8. Review and launch the instance.



Instances (3/3) Info

☑	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Pu
☑	webapp server	i-0a64d5f225a18b0a4	Running	t2.micro	2/2 checks passec	View alarms +	us-east-1d	ec:
☑	ansible	i-015de3d27afe22cc0	Running	t2.micro	2/2 checks passec	View alarms +	us-east-1b	ec:
☑	jenkins	i-04c26721be432ade5	Running	t2.micro	2/2 checks passec	View alarms +	us-east-1b	ec:

3 instances selected

Repeat this process for separate instances for Jenkins, Ansible, and the Web Server.

2.2. Configure Security Groups

1. Jenkins Security Group:

   - Open Port 8080 (default Jenkins port).

   - Type: Custom TCP | Port Range: 8080 | Source: Anywhere (or restrict IP ranges for security).

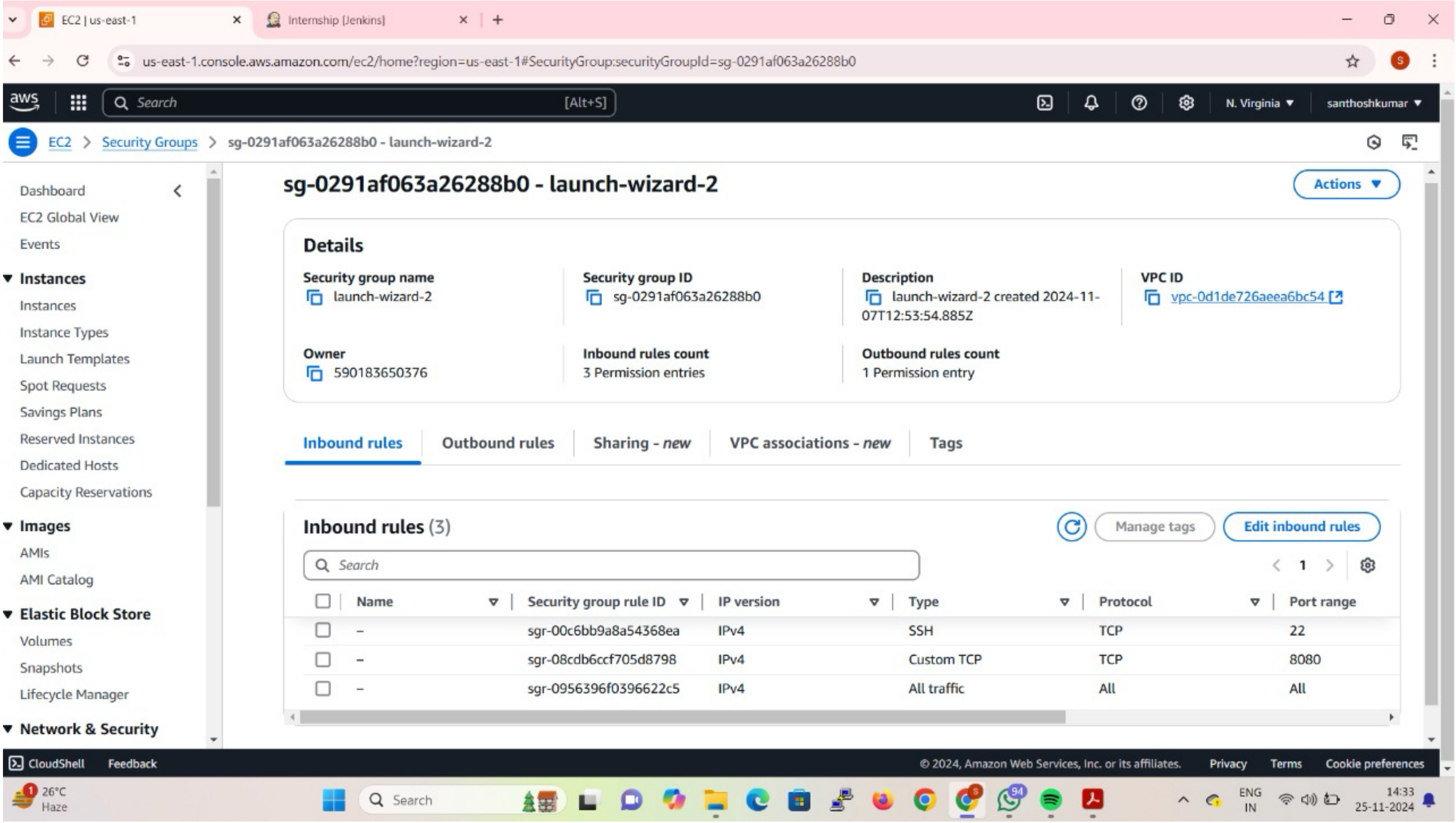
2. Ansible Security Group:

   - Open Port 22 (SSH).

   - Type: SSH | Port Range: 22 | Source: Custom IP range.

3. Web Server Security Group:

- Open Port 8080 (for HTTP traffic).
- Type: Custom TCP | Port Range: 8080 | Source: Anywhere.



sg-0291af063a26288b0 - launch-wizard-2

Details

Security group name launch-wizard-2	Security group ID sg-0291af063a26288b0	Description launch-wizard-2 created 2024-11-07T12:53:54.885Z	VPC ID vpc-0d1de726aeea6bc54
Owner 590183650376	Inbound rules count 3 Permission entries	Outbound rules count 1 Permission entry	

Inbound rules (3)

Name	Security group rule ID	IP version	Type	Protocol	Port range
–	sgr-00c6bb9a8a54368ea	IPv4	SSH	TCP	22
–	sgr-08cdb6ccf705d8798	IPv4	Custom TCP	TCP	8080
–	sgr-0956396f0396622c5	IPv4	All traffic	All	All

3. Setting Up Jenkins Server

3.1. Install Jenkins

1. SSH into the Jenkins EC2 instance:

```bash
ssh -i your-key.pem ec2-user@your-jenkins-public-ip
```

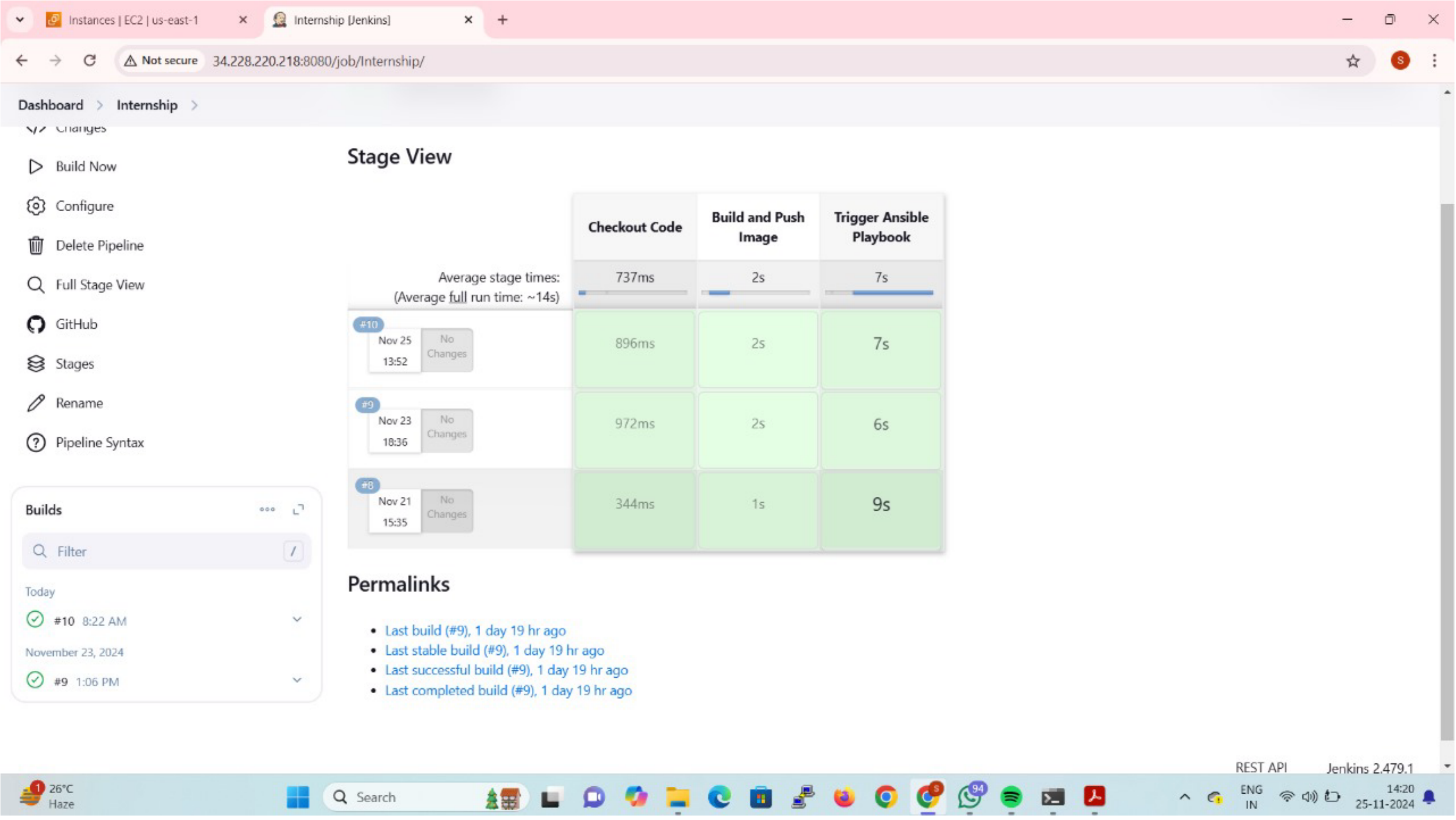
2. Install Java and Jenkins:

```bash
sudo yum update -y
sudo amazon-linux-extras enable corretto8
sudo yum install java-1.8.0-amazon-corretto -y
wget -O /etc/yum.repos.d/jenkins.repo https://pkg.jenkins.io/redhat-stable/jenkins.repo
rpm --import https://pkg.jenkins.io/redhat-stable/jenkins.io.key
sudo yum install jenkins -y
sudo systemctl start jenkins
sudo systemctl enable jenkins
```

3. Access Jenkins at `http://your-jenkins-public-ip:8080`.

3.2. Open Port for Jenkins

Ensure port 8080 is open as mentioned in 2.2.



Stage View

	Checkout Code	Build and Push Image	Trigger Ansible Playbook
Average stage times: (Average full run time: ~14s)	737ms	2s	7s
#10 Nov 25 13:52 No Changes	896ms	2s	7s
#9 Nov 23 18:36 No Changes	972ms	2s	6s
#8 Nov 21 15:35 No Changes	344ms	1s	9s

Permalinks

- Last build (#9), 1 day 19 hr ago
- Last stable build (#9), 1 day 19 hr ago
- Last successful build (#9), 1 day 19 hr ago
- Last completed build (#9), 1 day 19 hr ago

4. Setting Up Ansible Server

4.1. Install Ansible

1. SSH into the Ansible EC2 instance:

```bash
ssh -i your-key.pem ec2-user@your-ansible-public-ip
```

2. Install Ansible:

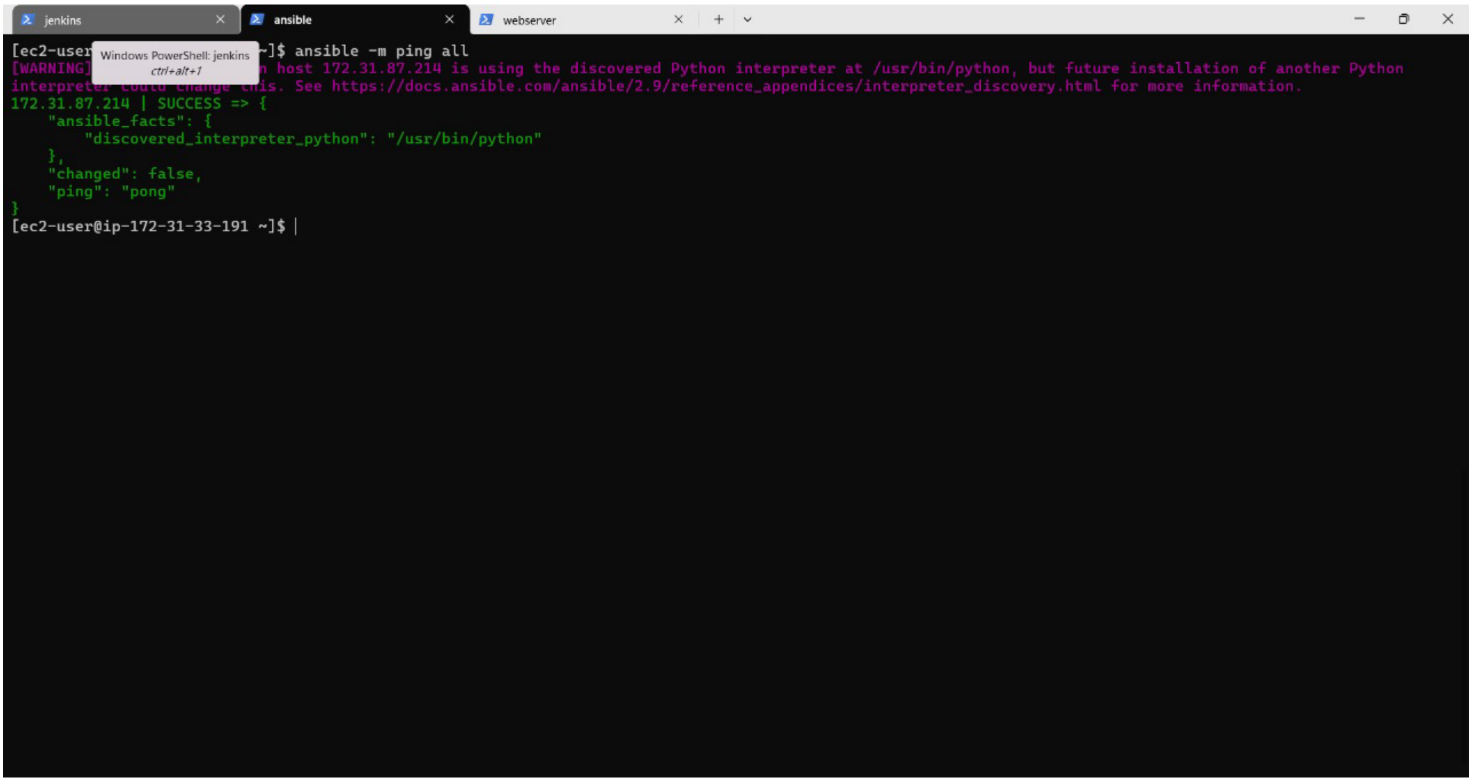
```bash
sudo amazon-linux-extras enable ansible2
sudo yum install ansible -y
```

4.2. Configure Ansible

Edit the `/etc/ansible/hosts` file to add target server IPs:

```ini
[webserver]
your-webserver-ip ansible_user=ec2-user

[jenkins]
your-jenkins-ip ansible_user=ec2-user
```



```
[ec2-user                    ~]$ ansible -m ping all
[WARNING]                    n host 172.31.87.214 is using the discovered Python interpreter at /usr/bin/python, but future installation of another Python
interpreter could change this. See https://docs.ansible.com/ansible/2.9/reference_appendices/interpreter_discovery.html for more information.
172.31.87.214 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python"
    },
    "changed": false,
    "ping": "pong"
}
[ec2-user@ip-172-31-33-191 ~]$ |
```

5. Setting Up Web Server

5.1. Install Docker on Amazon Linux 2

1. SSH into the Web Server EC2 instance:

   ```bash
   ssh -i your-key.pem ec2-user@your-webserver-public-ip
   ```

2. Install Docker:

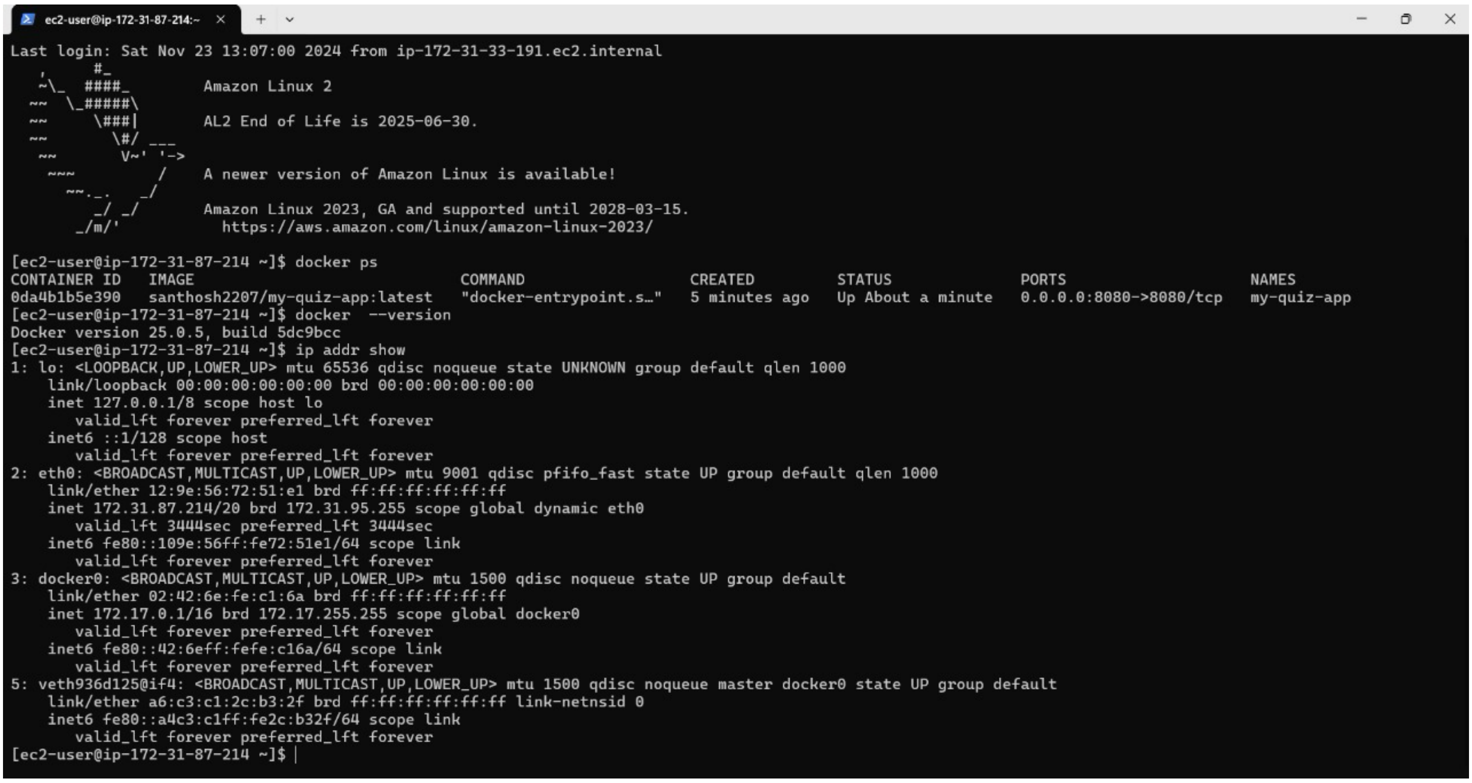
```bash
sudo yum update -y
sudo amazon-linux-extras enable docker
sudo yum install docker -y
sudo systemctl start docker
sudo systemctl enable docker
```

3. Verify installation:

```bash
docker --version
```

5.2. Configure Web Server

Deploy your application via Docker or set up a traditional web server.



```
Last login: Sat Nov 23 13:07:00 2024 from ip-172-31-33-191.ec2.internal
   ,     #_
   ~\_  ####_        Amazon Linux 2
  ~~  \_#####\
  ~~     \###|       AL2 End of Life is 2025-06-30.
  ~~       \#/ ___
   ~~       V~' '->
    ~~~         /    A newer version of Amazon Linux is available!
      ~~._.   _/
         _/ _/       Amazon Linux 2023, GA and supported until 2028-03-15.
       _/m/'           https://aws.amazon.com/linux/amazon-linux-2023/

[ec2-user@ip-172-31-87-214 ~]$ docker ps
CONTAINER ID   IMAGE                           COMMAND                  CREATED         STATUS             PORTS                    NAMES
0da4b1b5e390   santhosh2207/my-quiz-app:latest   "docker-entrypoint.s…"   5 minutes ago   Up About a minute   0.0.0.0:8080->8080/tcp   my-quiz-app
[ec2-user@ip-172-31-87-214 ~]$ docker  --version
Docker version 25.0.5, build 5dc9bcc
[ec2-user@ip-172-31-87-214 ~]$ ip addr show
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 9001 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 12:9e:56:72:51:e1 brd ff:ff:ff:ff:ff:ff
    inet 172.31.87.214/20 brd 172.31.95.255 scope global dynamic eth0
       valid_lft 3444sec preferred_lft 3444sec
    inet6 fe80::109e:56ff:fe72:51e1/64 scope link
       valid_lft forever preferred_lft forever
3: docker0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default
    link/ether 02:42:6e:fe:c1:6a brd ff:ff:ff:ff:ff:ff
    inet 172.17.0.1/16 brd 172.17.255.255 scope global docker0
       valid_lft forever preferred_lft forever
    inet6 fe80::42:6eff:fefe:c16a/64 scope link
       valid_lft forever preferred_lft forever
5: veth936d125@if4: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue master docker0 state UP group default
    link/ether a6:c3:c1:2c:b3:2f brd ff:ff:ff:ff:ff:ff link-netnsid 0
    inet6 fe80::a4c3:c1ff:fe2c:b32f/64 scope link
       valid_lft forever preferred_lft forever
[ec2-user@ip-172-31-87-214 ~]$ |
```

6. Final Steps

- Test all servers to ensure Jenkins, Ansible, and the Web Server are operational.
- Configure Jenkins jobs for CI/CD pipelines.
- Secure servers by limiting security group rules to trusted IPs and using key pairs for SSH.