Installing Jenkins on Ec2 Ubuntu

Introduction

This Documentation provides step by step guide for installing Jenkins on an EC2 instance running Ubuntu allowing you to leverage its capabilities to streamline your development workflow.

Prerequisites for Installing Jenkins on AWS EC2

Before proceeding with the installation of Jenkins on an AWS EC2 instance, ensure you have the following prerequisites in place

- ✓ AWS Account
- ✓ EC2 Instance
- ✓ Java Runtime Environment (JRE)
- ✓ SSH Key Pair
- ✓ Basic of AWS Services

What is Jenkins ?

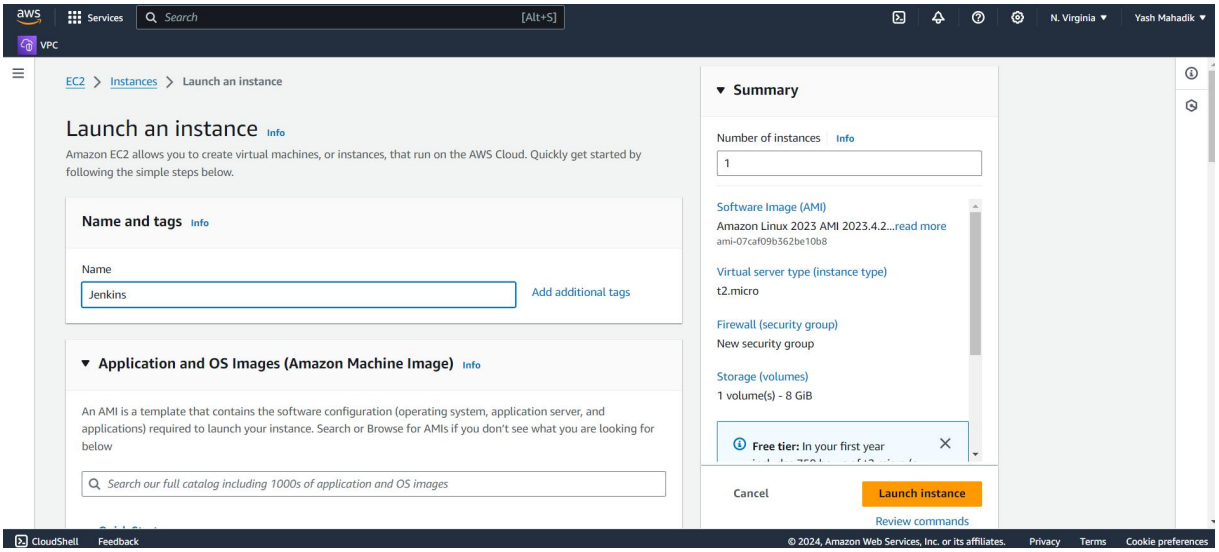
Jenkins is an open-source automation server that helps to automate various stages of software development. It is primarily used for Continuous Integration and Continuous Delivery/Deployment (CI/CD) pipelines, which allow developers to quickly build, test, and deploy their code to production.

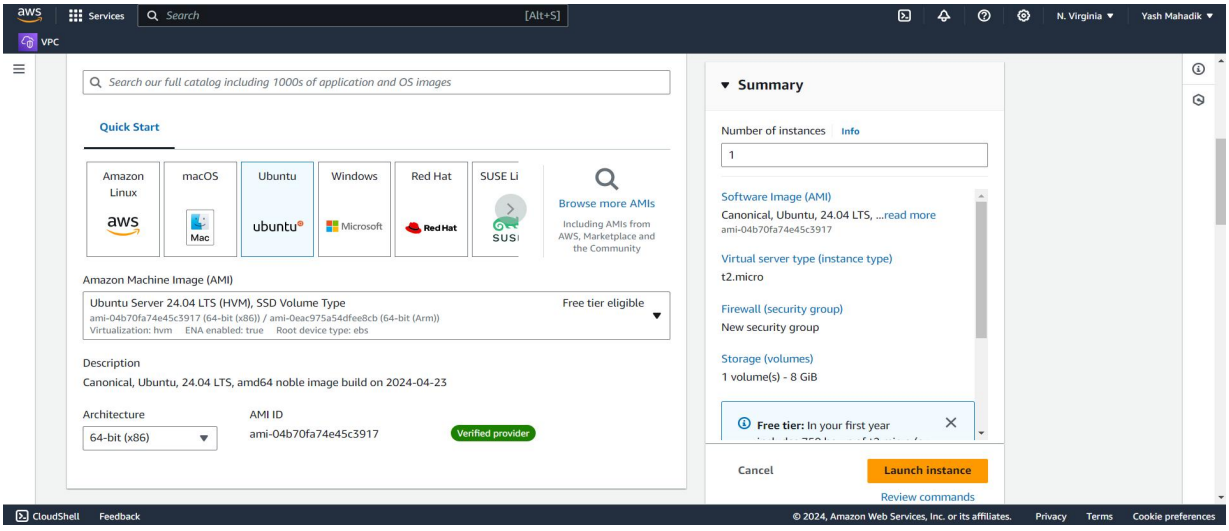
What is AWS Cloud ?

AWS (Amazon Web Services) is a cloud computing platform that provides a range of services for developers and businesses to build and deploy applications on the cloud. AWS has a wide range of services, including computing, storage, database, networking, and machine learning.

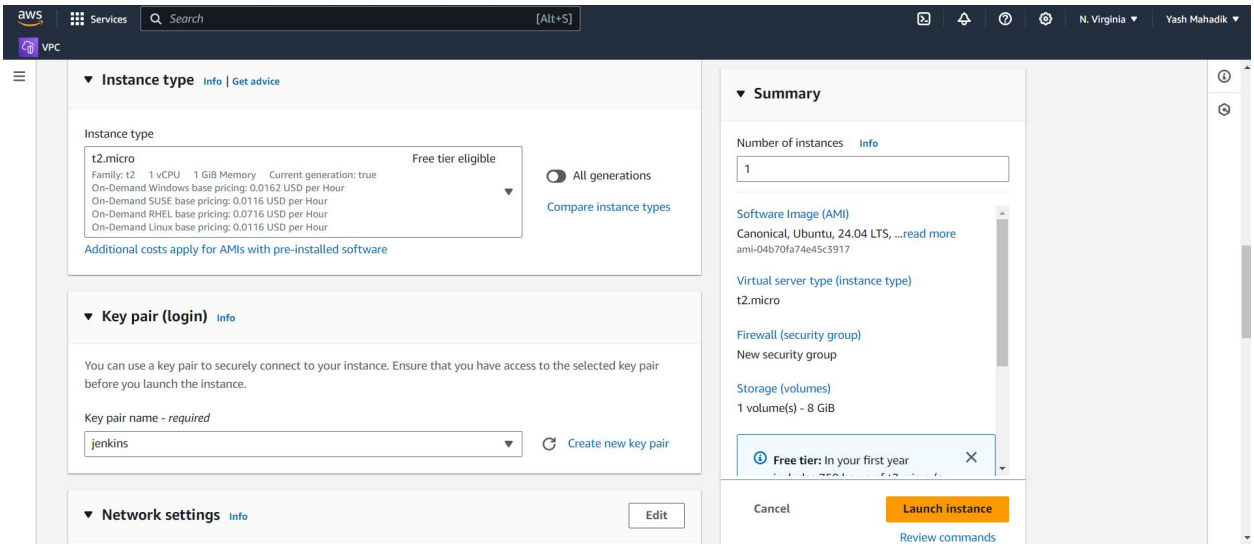
A step-by-step guide to installing Jenkins on an AWS EC2 Ubuntu 22.04 instance:

1) To First Login into your AWS account and to Launch Ec2 instance by giving the name of the instances as Jenkins-demo.

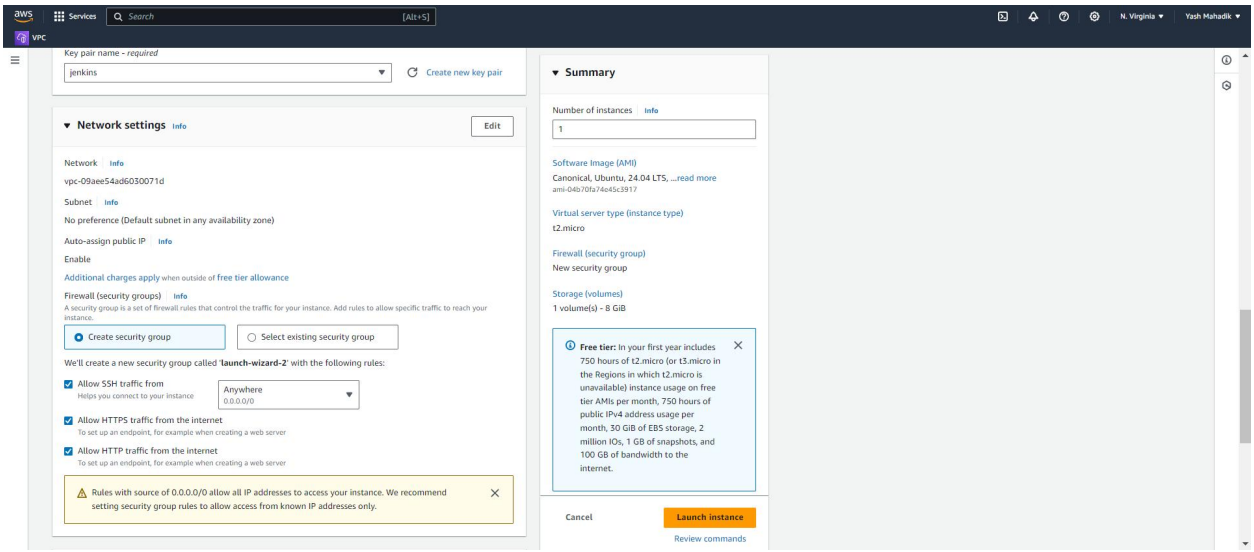




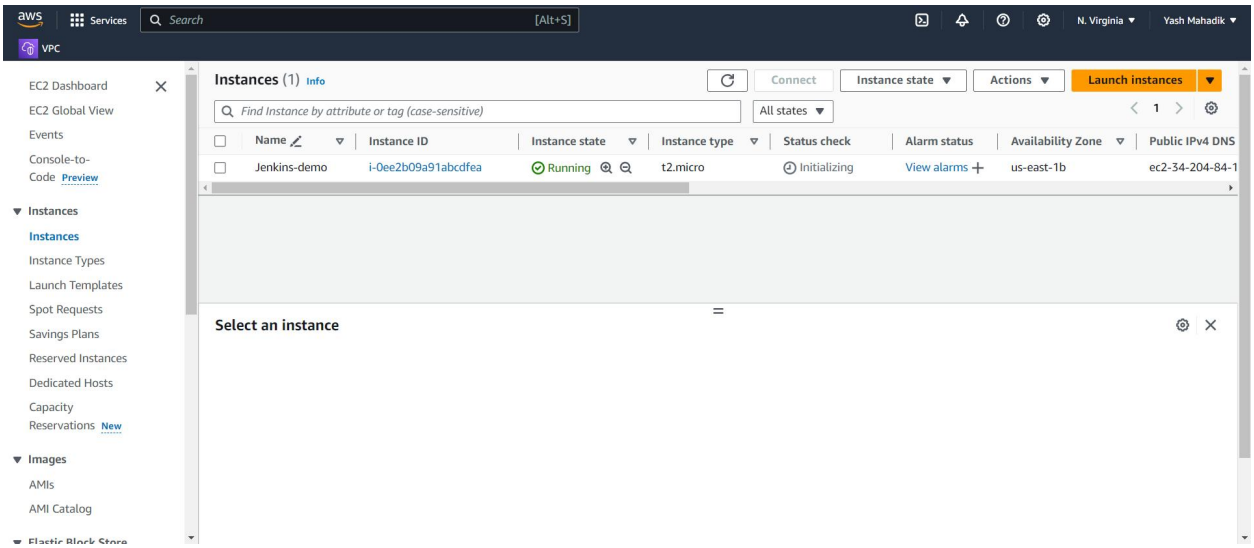
2) Use instance type as **t2.micro** which is **free tier** and create a **key pair**. You can select the appropriate instance type based on the requirements of your project.

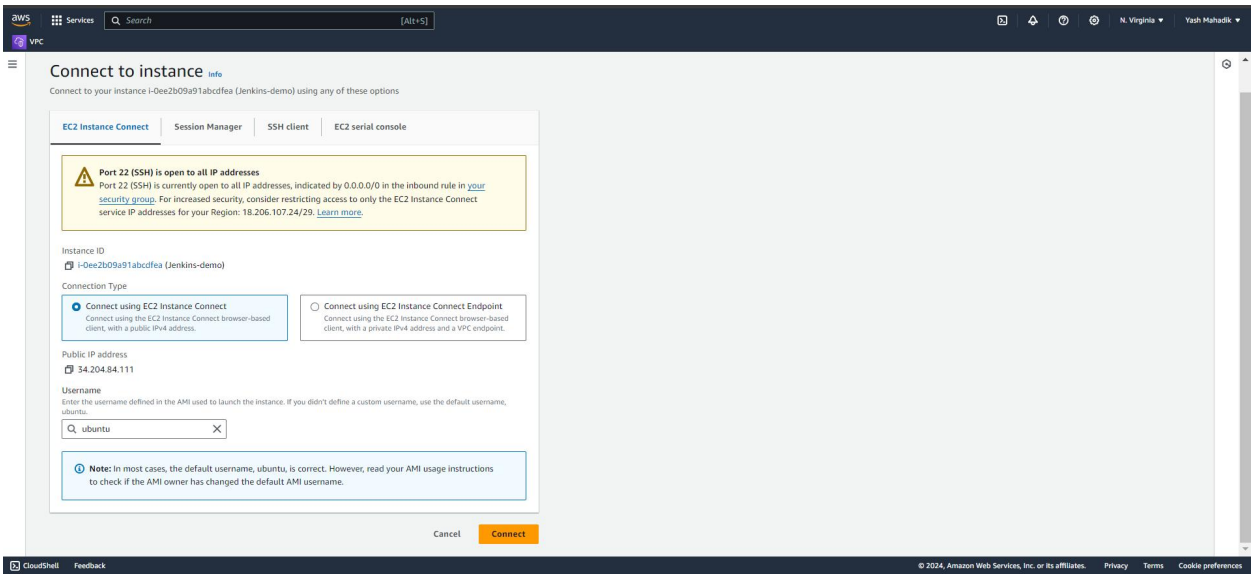


3) In network section just check box ☑ on Allow HTTPS traffic from the internet and Click on Launch instance.



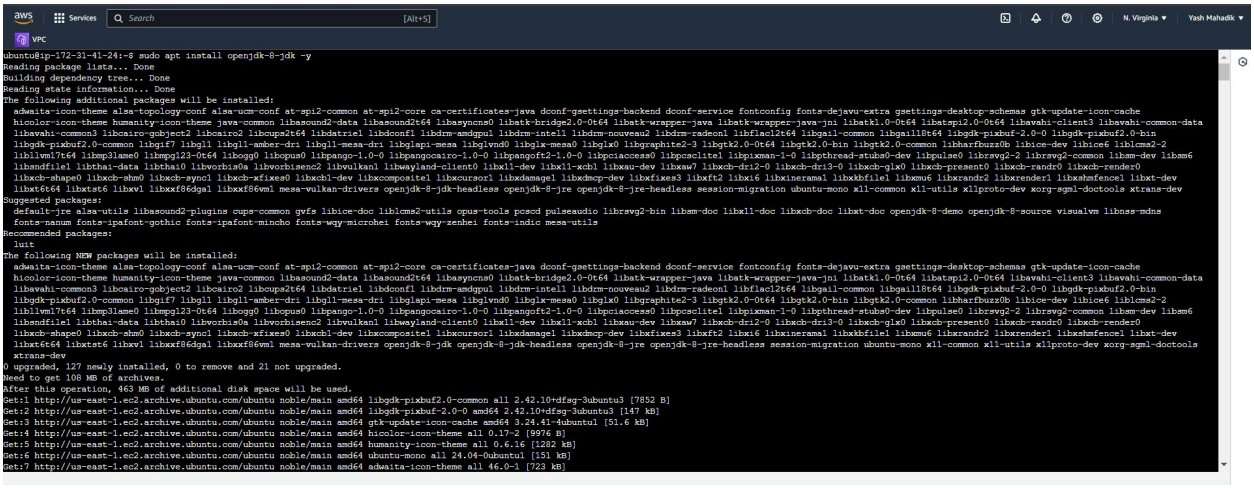
4) Once the instance is created click on Instance and click on connect to open the terminal.



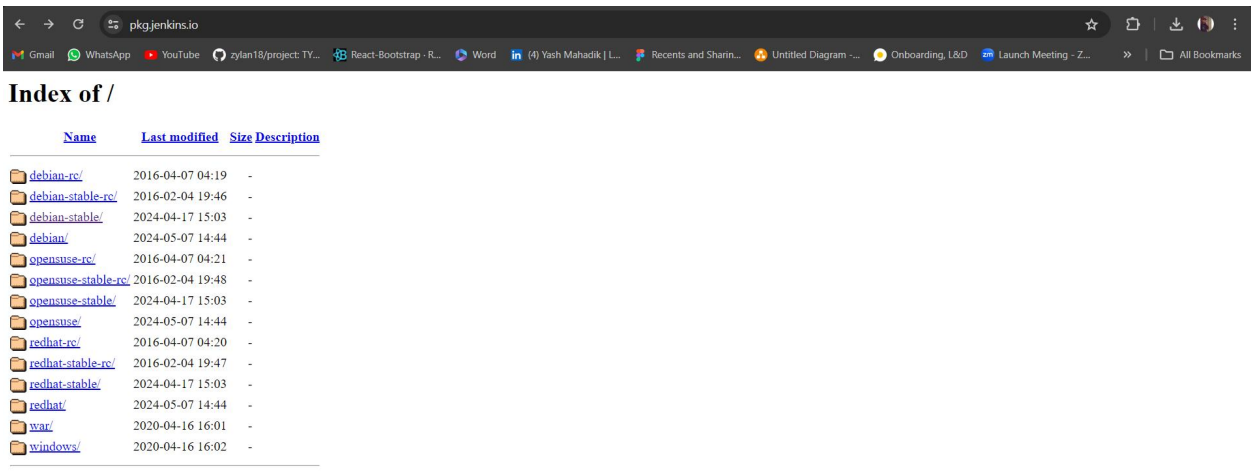


5) Jenkins requires Java to run, so the first step is to install Java on the Ubuntu instance. You can do this by running the following command: (Here we installing jdk version 8)

```
sudo apt-get update
sudo apt-get install openjdk-8-jdk
```



6) Search pkg.jenkins.io in Url and select Debian-stable.

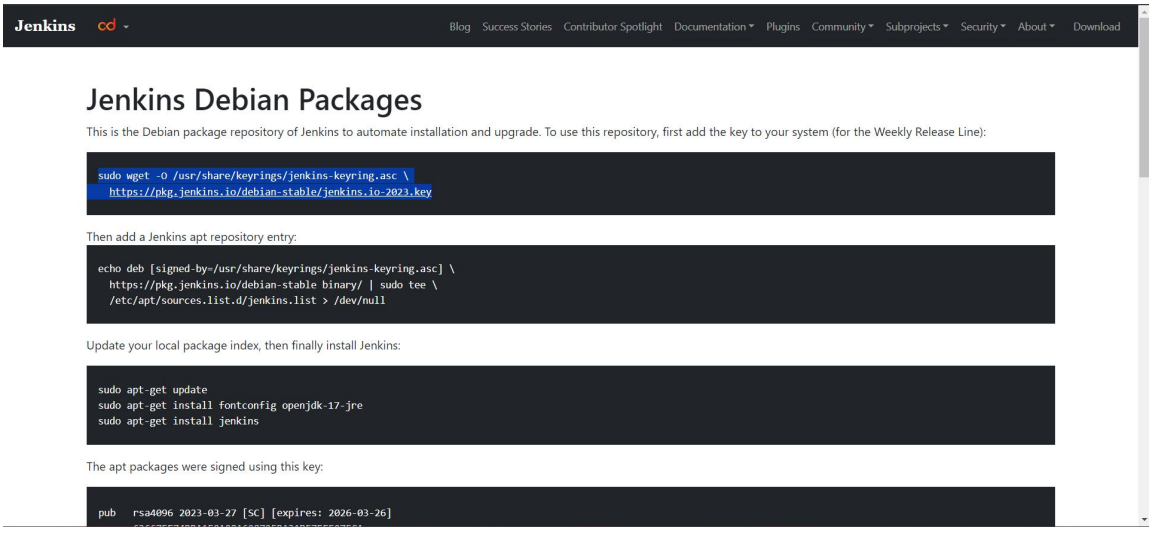


7)Copy each command and paste it on the terminal for installing jenkins.

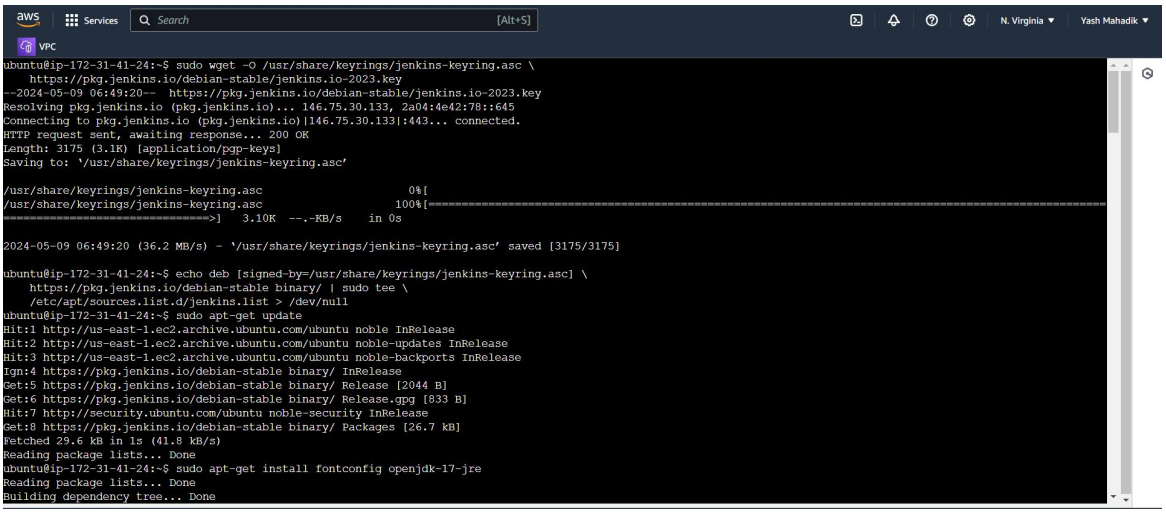
```
sudo wget -O /usr/share/keyrings/jenkins-keyring.asc ¥
  https://pkg.jenkins.io/debian-stable/jenkins.io-2023.key

echo deb [signed-by=/usr/share/keyrings/jenkins-keyring.asc] ¥
  https://pkg.jenkins.io/debian-stable binary/ | sudo tee ¥
  /etc/apt/sources.list.d/jenkins.list > /dev/null
```

Update your local package index, then finally install Jenkins:

```
sudo apt-get update
sudo apt-get install fontconfig openjdk-17-jre
sudo apt-get install jenkins
```



8) Jenkins gets installed on the terminal.

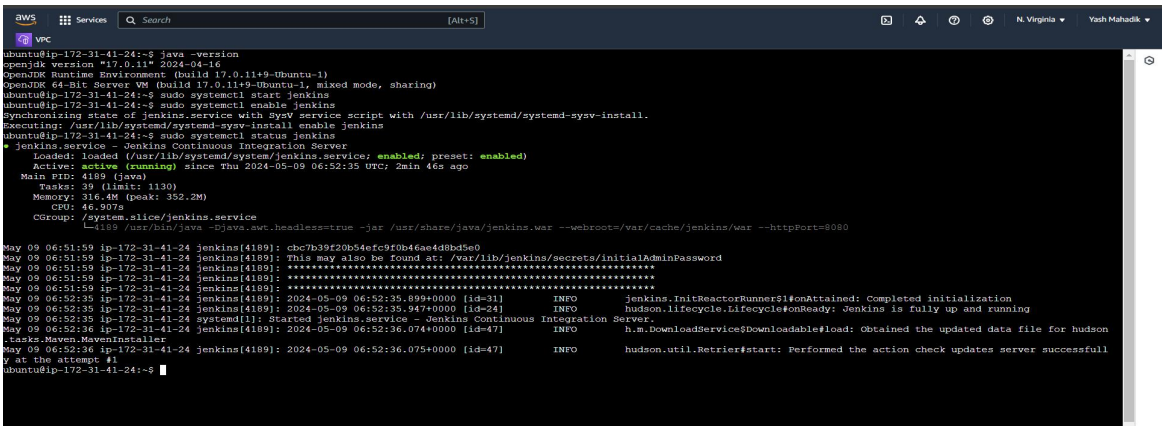


9) Once Jenkins is installed, start & enable the Jenkins service using the following command:

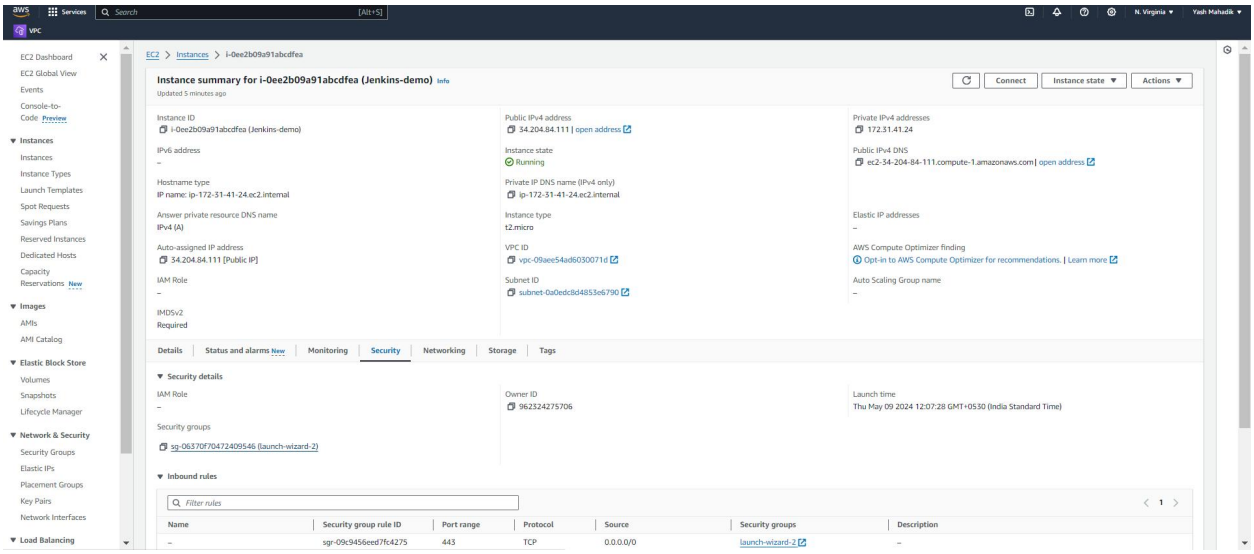
```
sudo systemctl start jenkins
```

```
sudo systemctl status jenkins
```

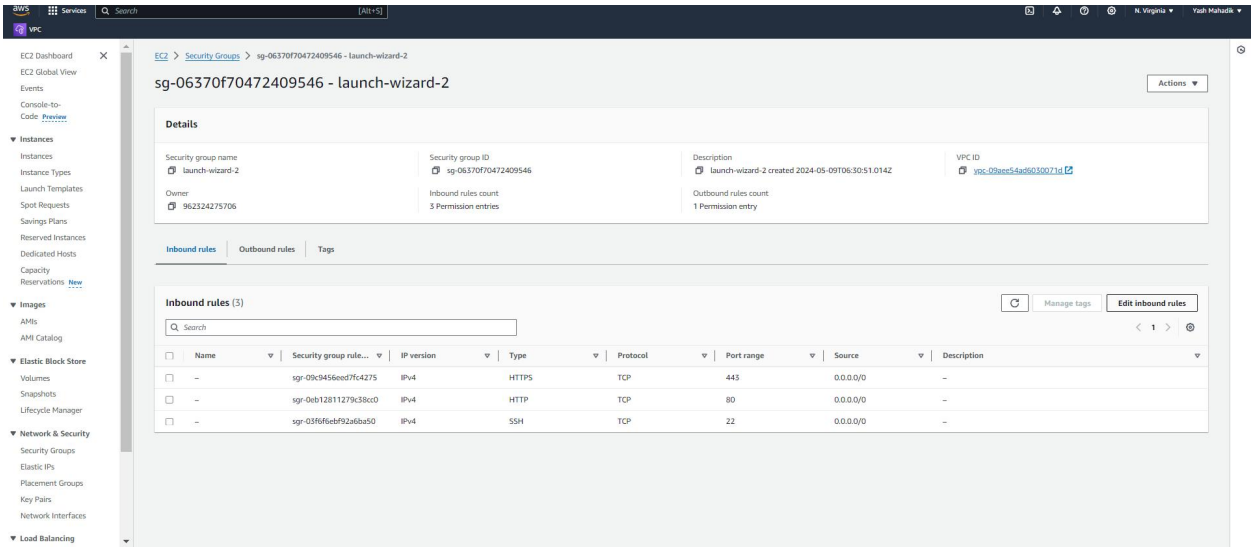
```
sudo systemctl enable jenkins
```



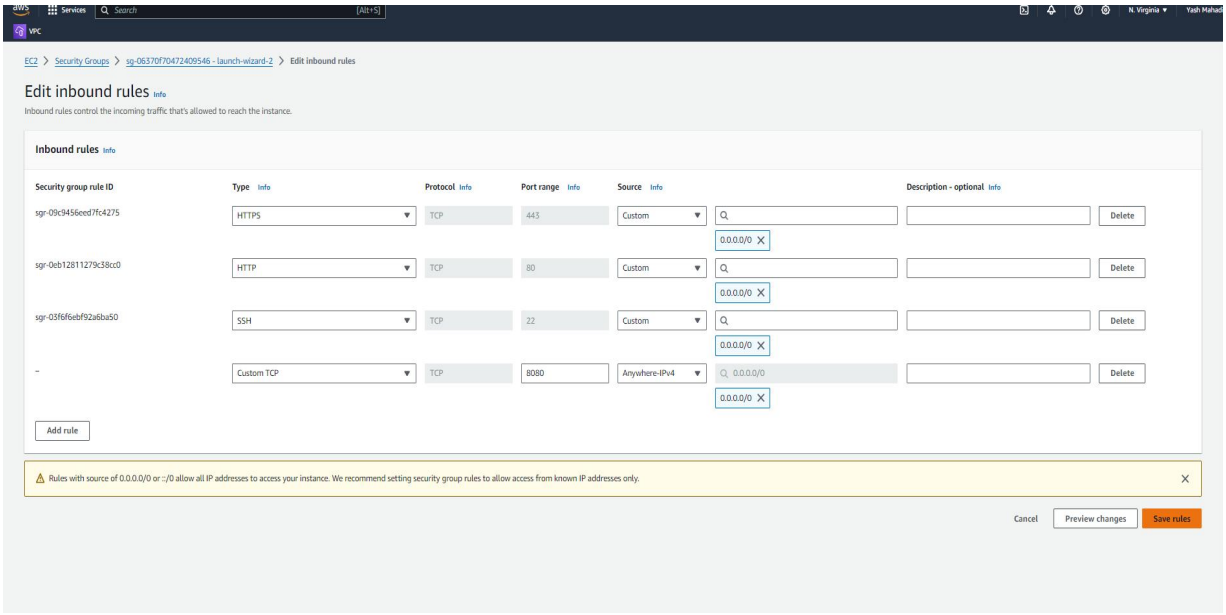
10) As we know Jenkins runs on port 8080, so we need to add a security group.

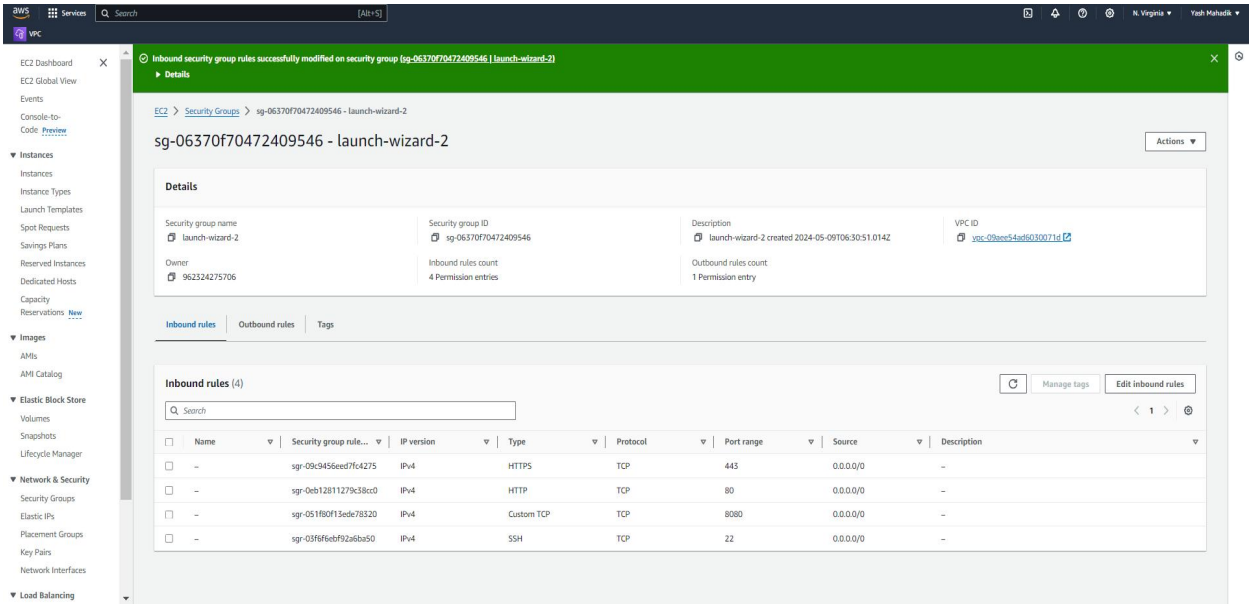


11) In the inbound section select edit inbound rules.

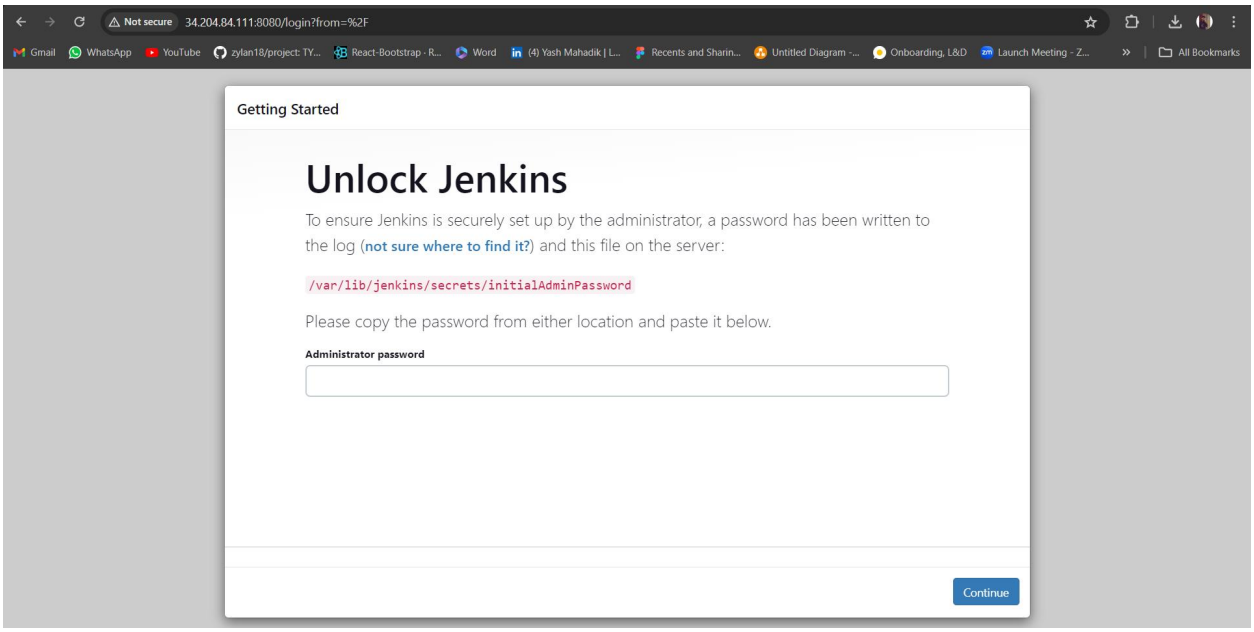


12) Add a new section with port range 8080 and click on save rules.

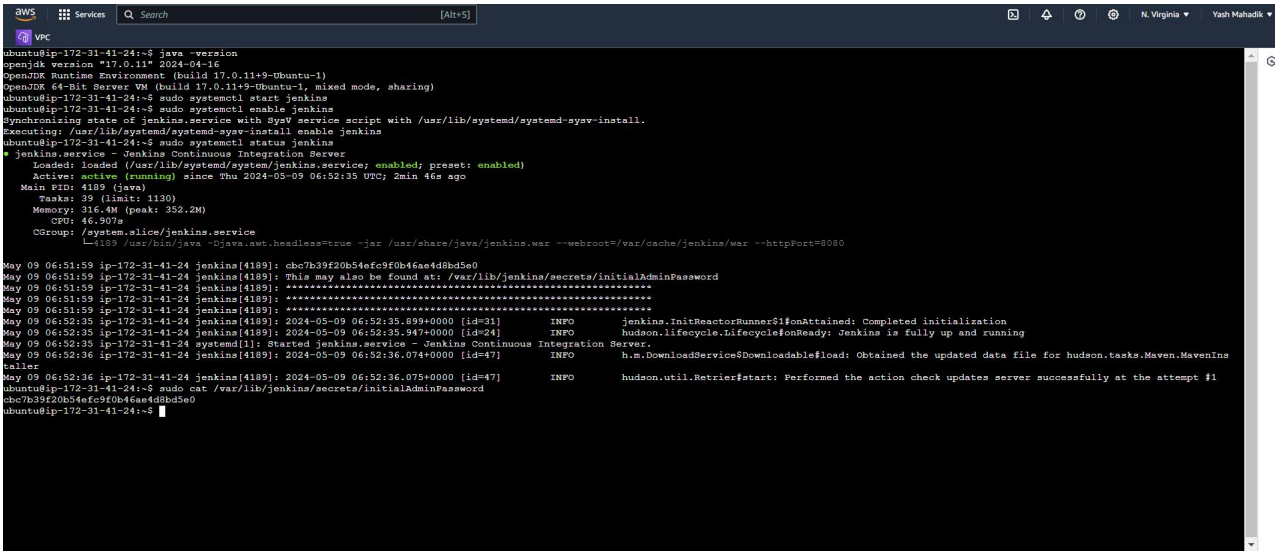




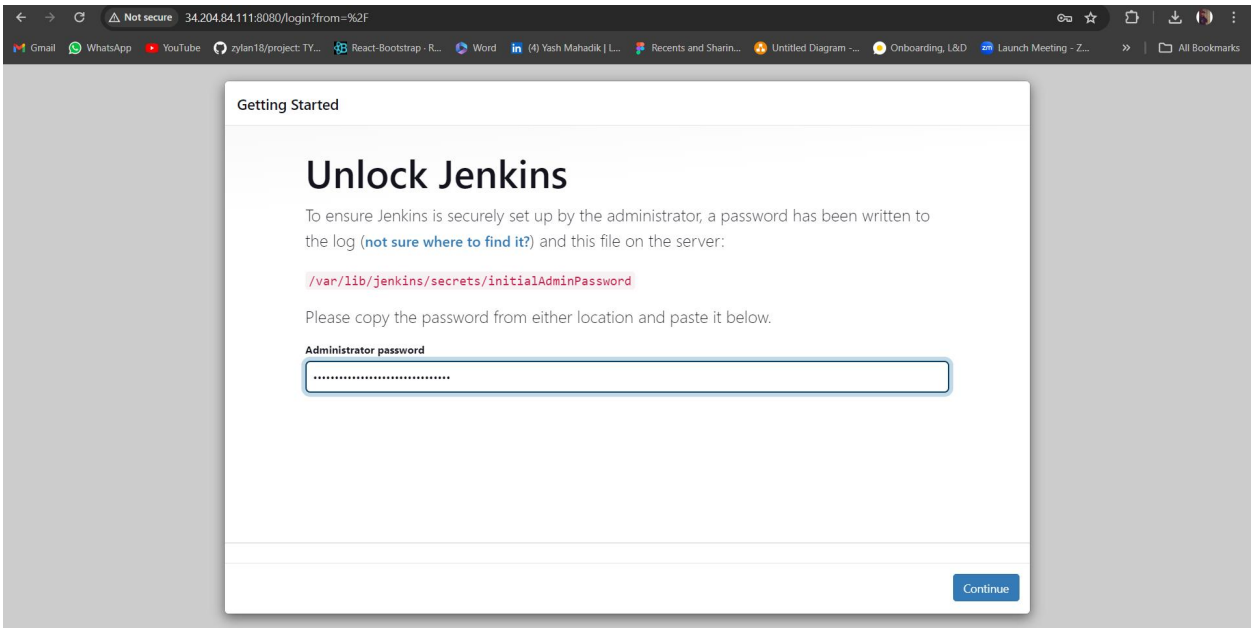
13) We can see port 8080 is running and we can get start with Jenkins. Finally, you can configure Jenkins by accessing the Jenkins web interface using the public IP address of your EC2 instance and the default port 8080 (e.g., http://<EC2_Public_IP>:8080).



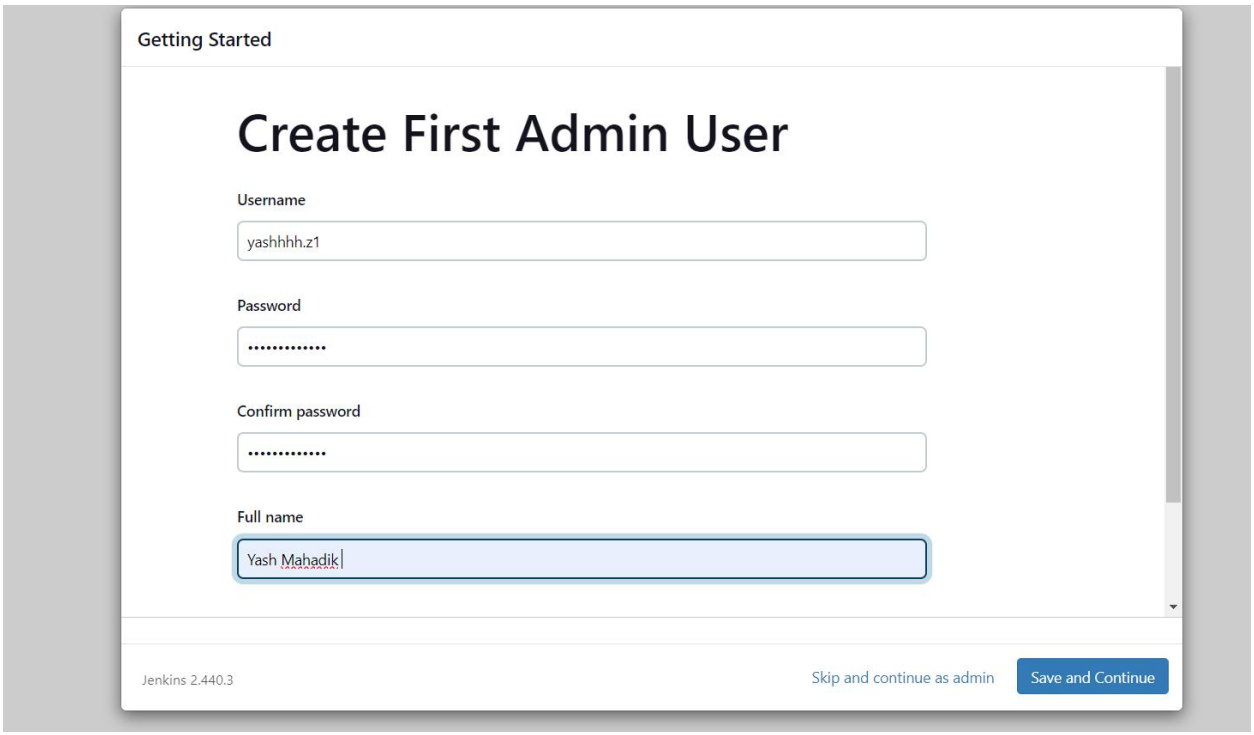
14) Copy the text and run **sudo cat <text you copied>** on terminal, a secret key will be generated and copy that key and paste in into you Jenkins as Administrator password



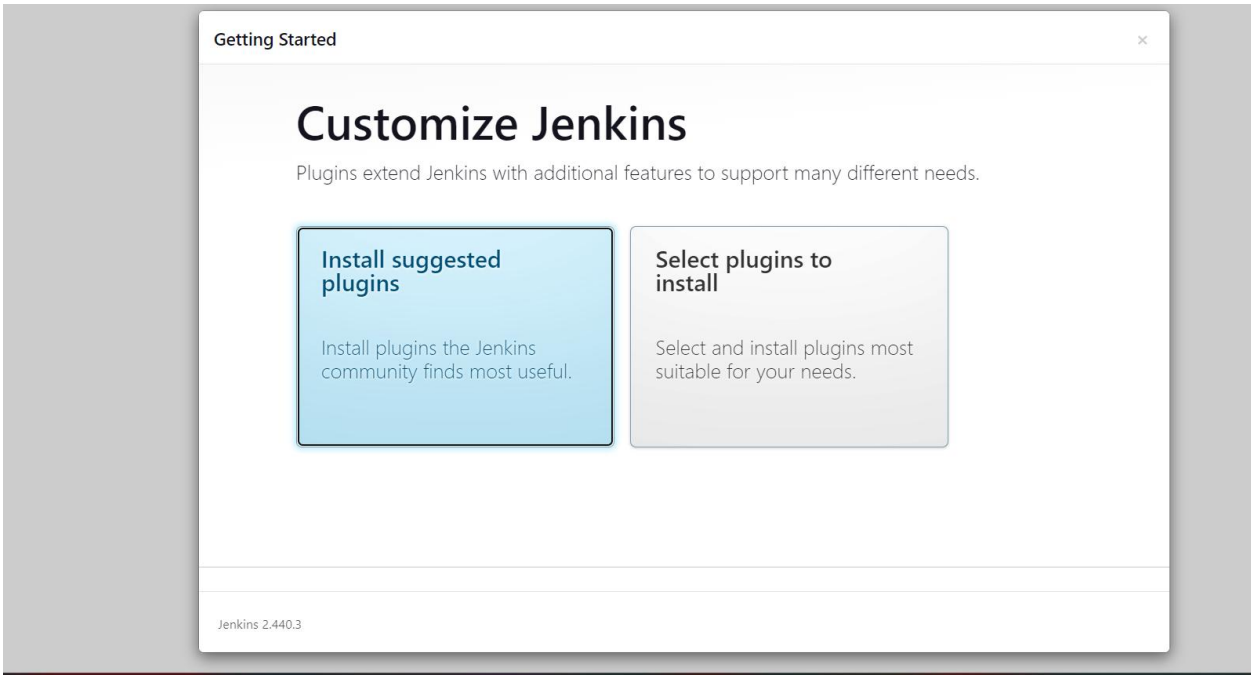
15) Add secret key to Administrator password



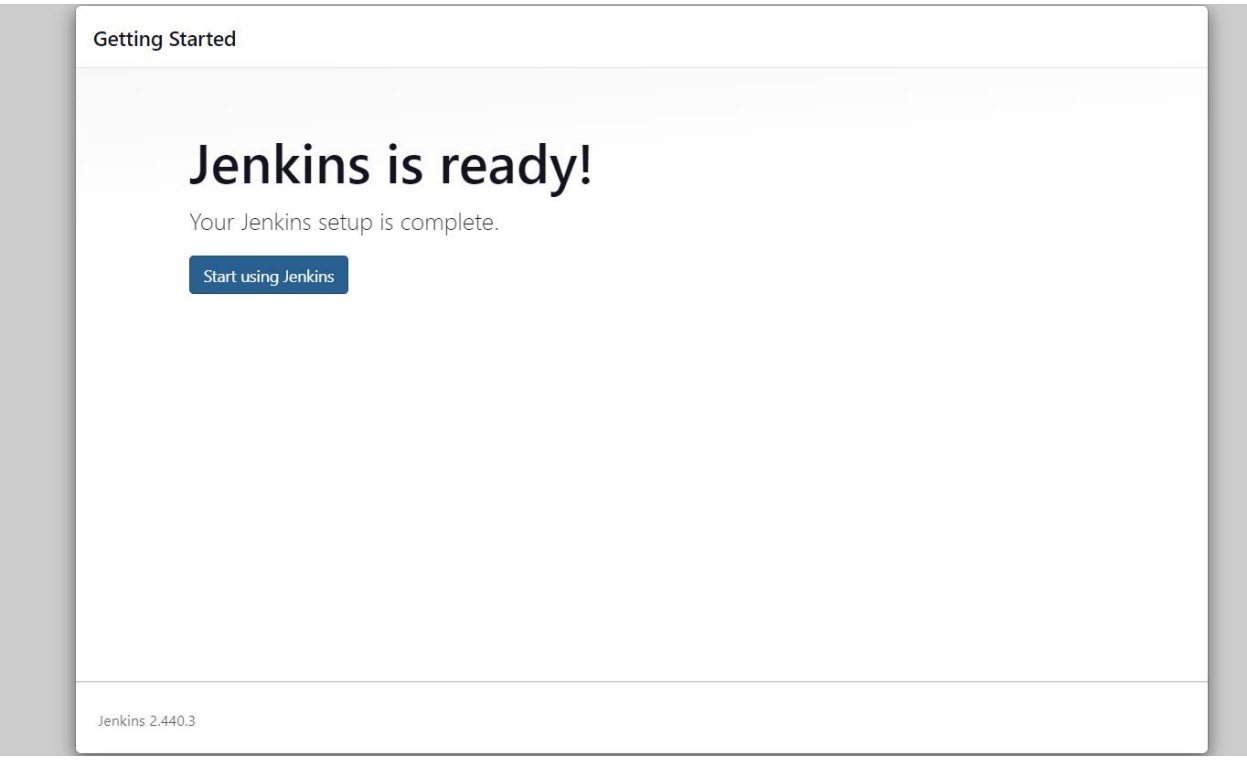
16) Create Admin user and click on save and continue.



17) Install necessary plugins.



18) Jenkins in Ready! Start using Jenkins



Conclusion: Installing Jenkins on an EC2 Ubuntu instance provides a robust foundation for implementing CI/CD pipelines, fostering collaboration among development teams, and accelerating software delivery. By following the documentation and incorporating best practices, users can leverage Jenkins to enhance the quality, reliability, and efficiency of their software development process.