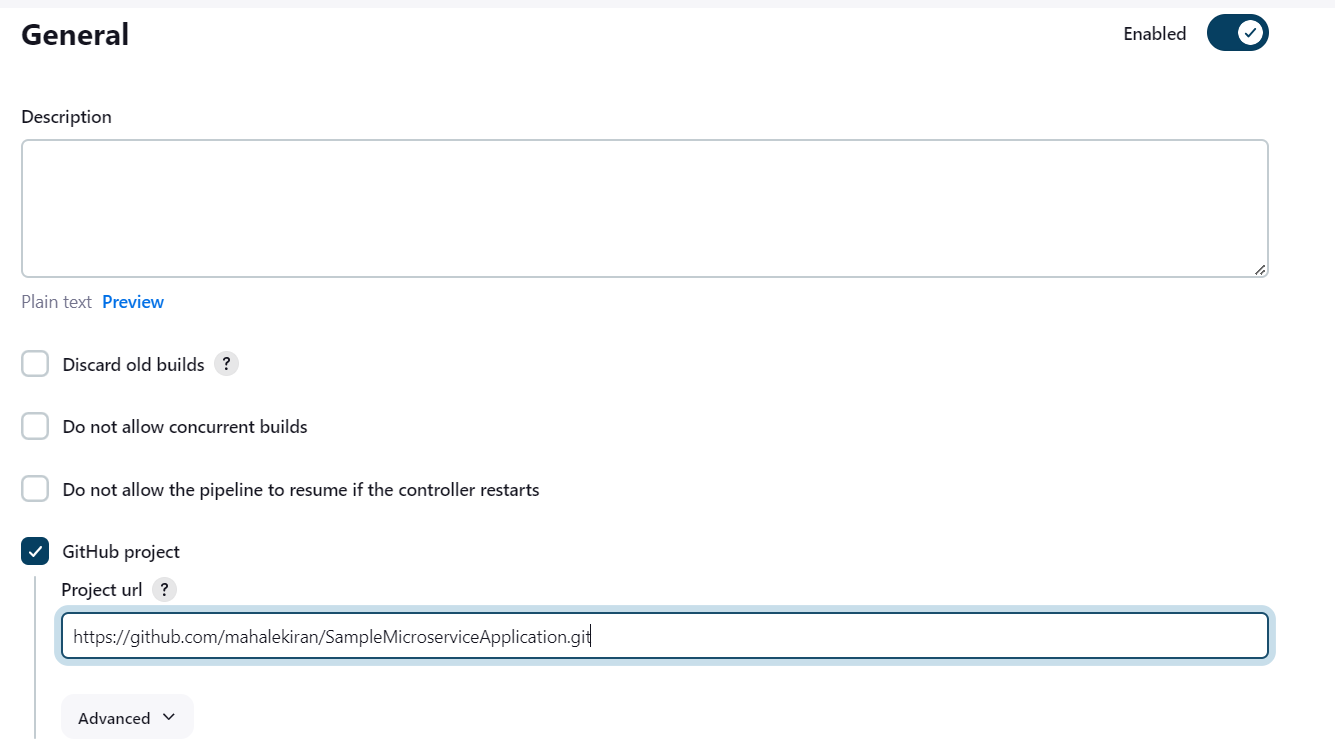
**CI/CD Pipeline for Microservice Application using Jenkins**

**Step 1:** Go to Jenkins instance <http://3.6.41.13:8080/>

**Step 2:** Go to Dashboard -> New Item

Enter the Item Name and click on Pipeline.

**Step 3:** Enter GitHub url:



Here I have forked the repository <https://github.com/UnpredictablePrashant/SampleMicroserviceApplication>

**Step 4:** We need to add pipeline syntax in Pipeline section

In the next 2 steps I have shown how to create SSH Credential ID which we have used in this Pipeline and Email Configuration.

**Script:**

pipeline {

agent any

environment {

GIT\_URL = 'https://github.com/mahalekiran/SampleMicroserviceApplication.git'

GIT\_BRANCH = 'main'

}

tools {

maven 'Maven 3.9.6'

}

stages {

stage('Hello') {

steps {

echo 'Hello World'

}

}

stage ('Initialize') {

steps {

sh '''

echo "PATH = ${PATH}"

echo "M2\_HOME = ${M2\_HOME}"

'''

}

}

stage('git\_repo\_cloning') {

steps {

git branch: env.GIT\_BRANCH , url: env.GIT\_URL

}

}

stage('Build') {

steps {

sh 'mvn clean verify -DskipITs=true'

}

}

stage('Test'){

steps{

sh 'mvn test'

}

}

stage('Artifact'){

steps{

sh 'mvn package'

}

}

stage('Deployment'){

steps{

sshagent(credentials:['Kiran\_SShKey']){

sh '''

scp -o StrictHostKeyChecking=no -r ${WORKSPACE}/\* ubuntu@18.191.226.7:/home/ubuntu/

'''

}

}

}

}

post {

success {

emailext (

subject: "Jenkins Build Successful: ${currentBuild.fullDisplayName}",

body: "The build was successful. It's ready for deployment.",

to: 'krn1988@gmail.com'

)

}

failure {

emailext (

subject: "Jenkins Build Failed: ${currentBuild.fullDisplayName}",

body: "The build failed. Please check the Jenkins console for details.",

to: 'krn1988@gmail.com'

)

}

}

}

**SSH Key:**

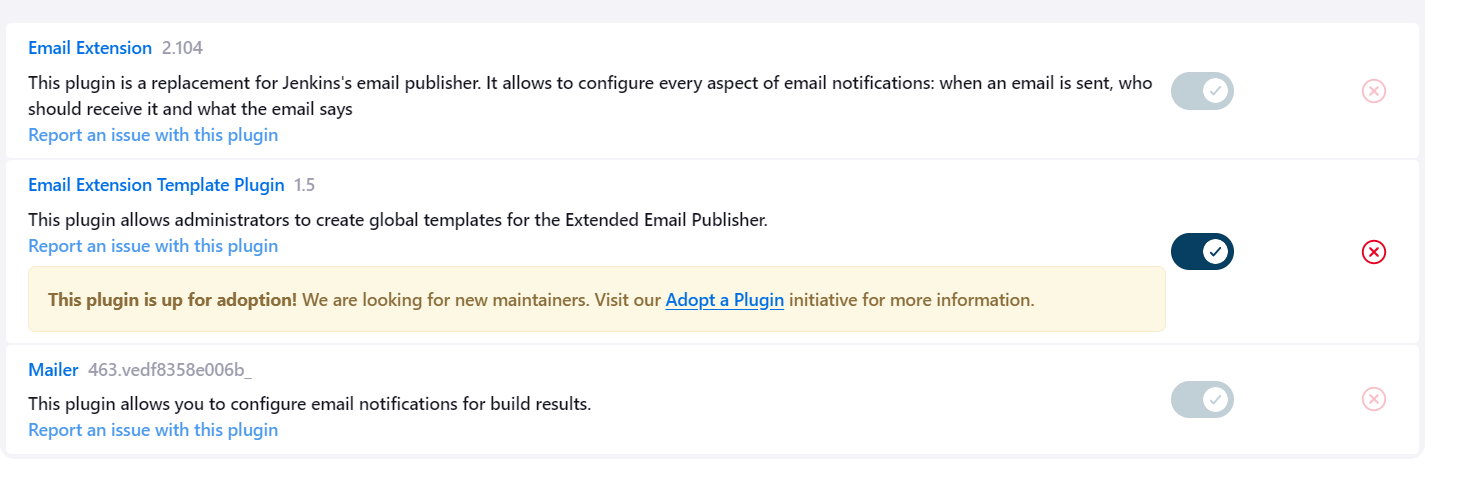
I have created my credential by following below steps:

1. Go to Dashboard -> Manage Jenkins -> Credentials -> System -> Global credentials
2. Add the ID -> Kiran\_SShKey
3. Description -> Kiran\_SShKey
4. Username -> ubuntu
5. Private Key -> in my case KiranM\_All.pem from EC2 instance.
6. Save
7. This key we have used in the above pipeline to connect to EC2 instance.

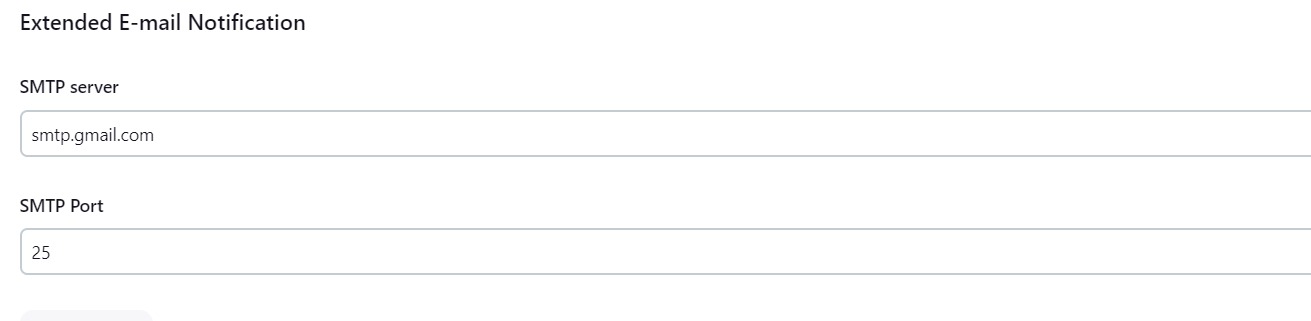
**Email configuration:**

In the above pipeline, we need to send email on success and failure. For which we need to configure the Email Plugin.

1. First, we have to install the Email Extension plugin if not done



1. Go to Extended E-mail Notification from Dashboard -> Manage Jenkins -> System
2. Add the below to configure it to send email to and from gmail SMTP.
3. SMTP server -> smtp.gmail.com
4. SMTP Port -> 25



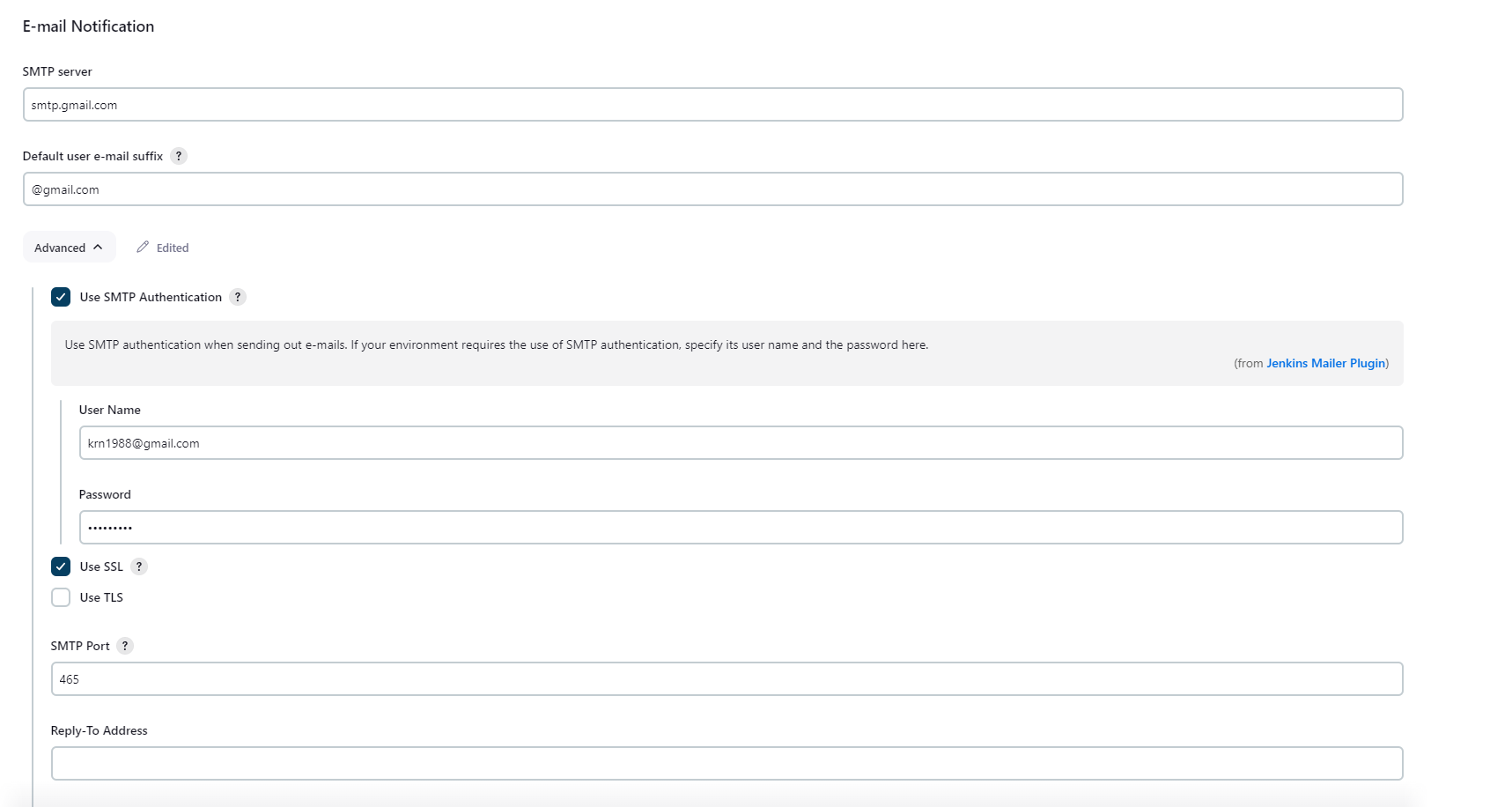
1. Add the other configuration in E-mail Notification
2. SMTP server -> smtp.gmail.com
3. Default user e-mail suffix -> @gmail.com
4. Click on Advanced and enter the details:

Click on Use SMTP Authentication

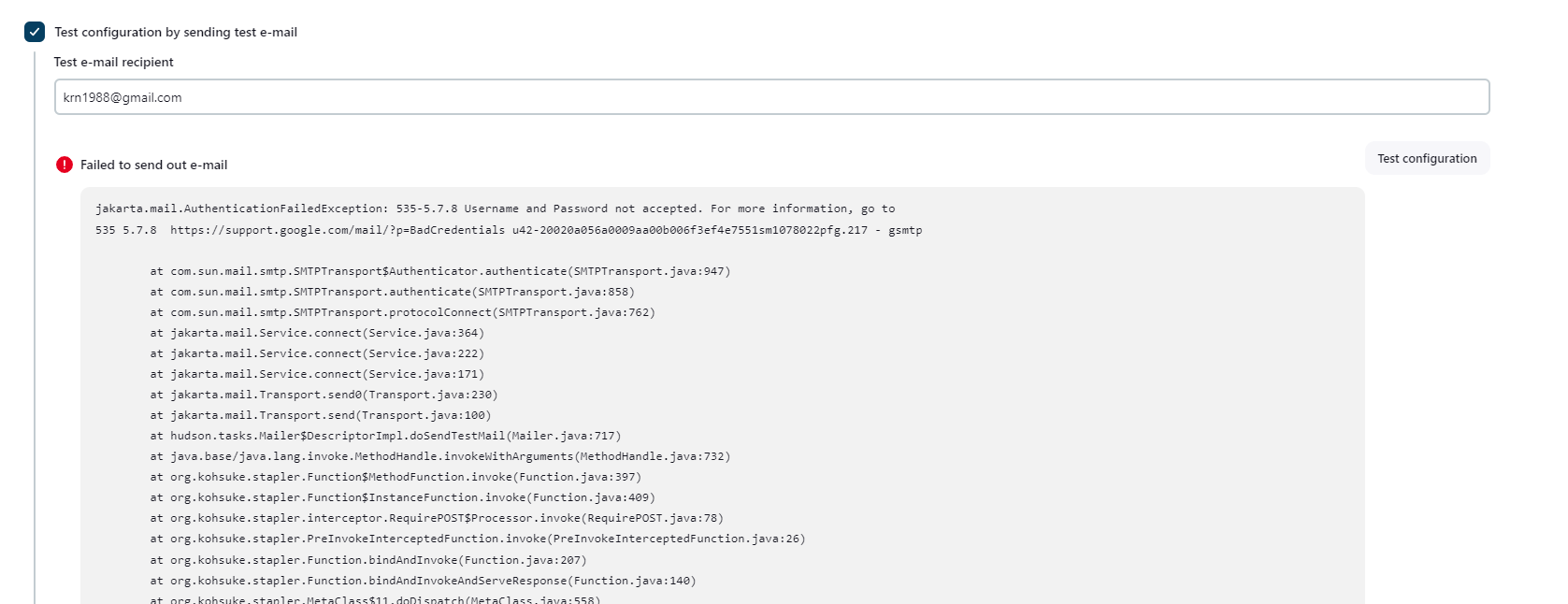
Add UserName and Password

Click on Use SSL

Enter SMTP Port as 465

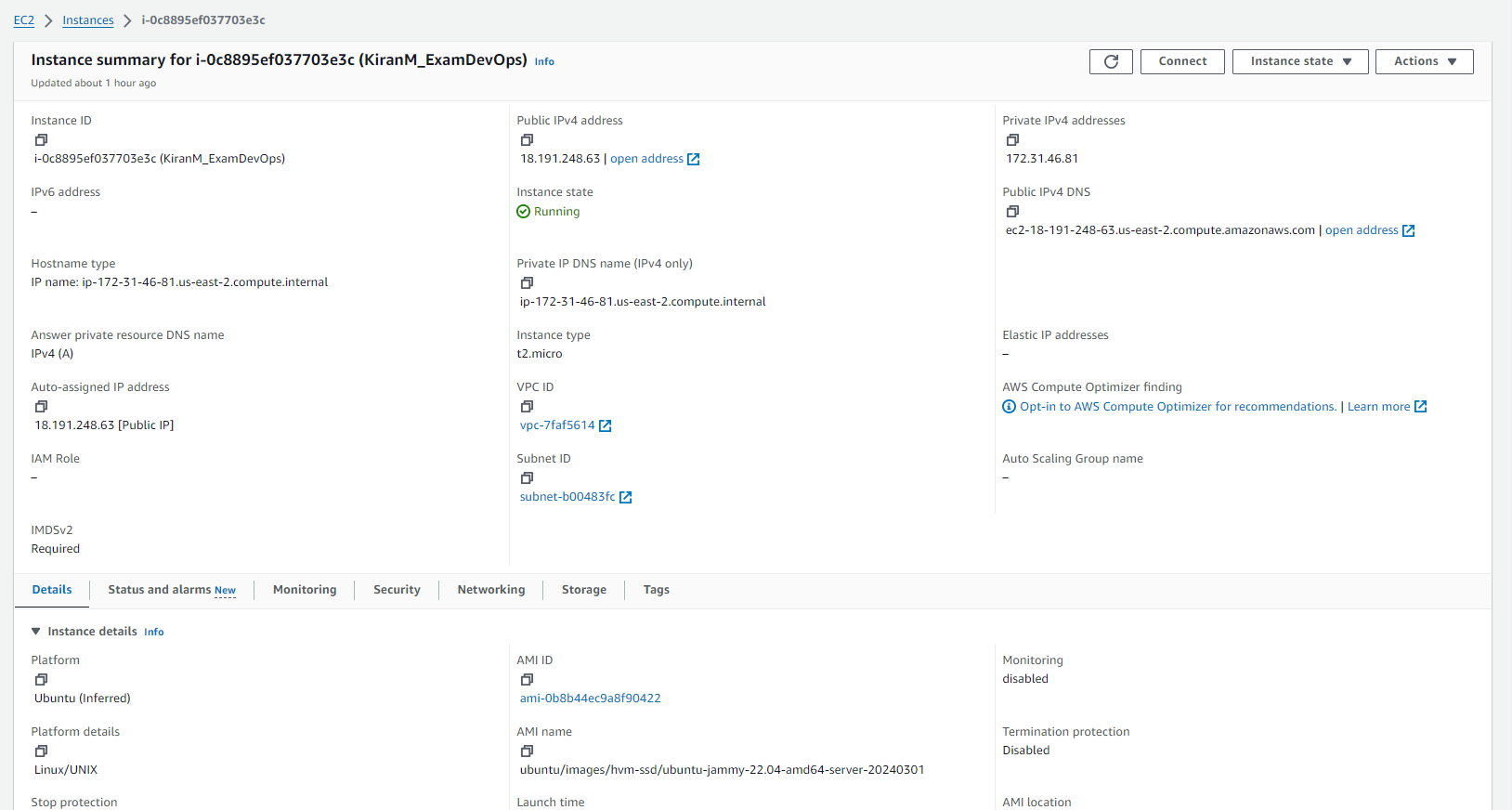


1. I tried to test this but it showed error:

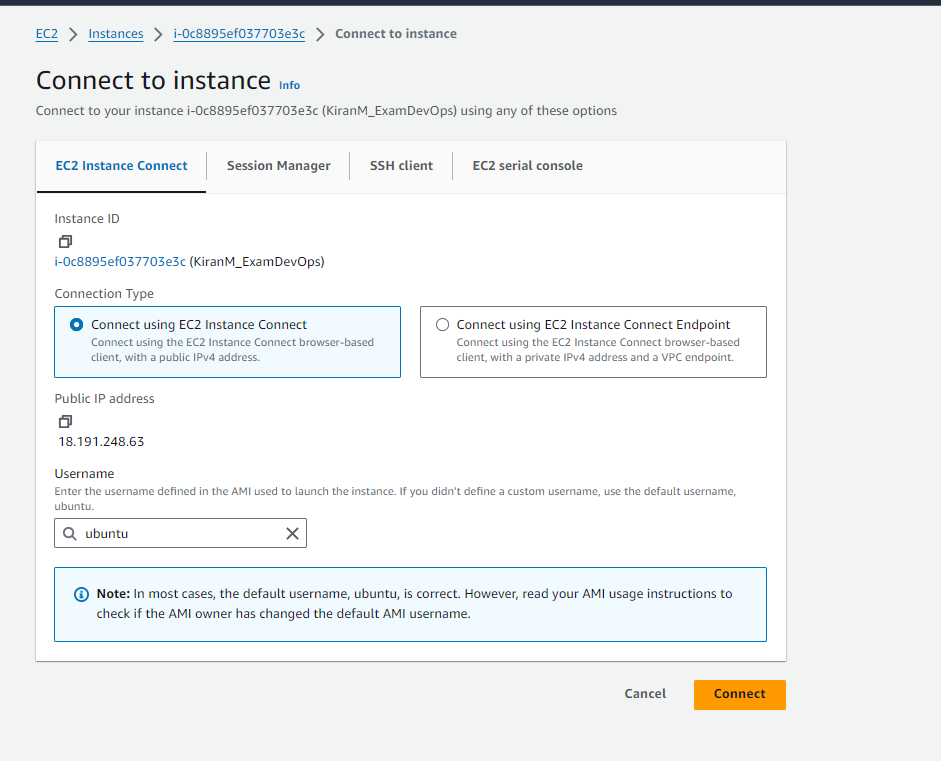


**EC2 Instance:**

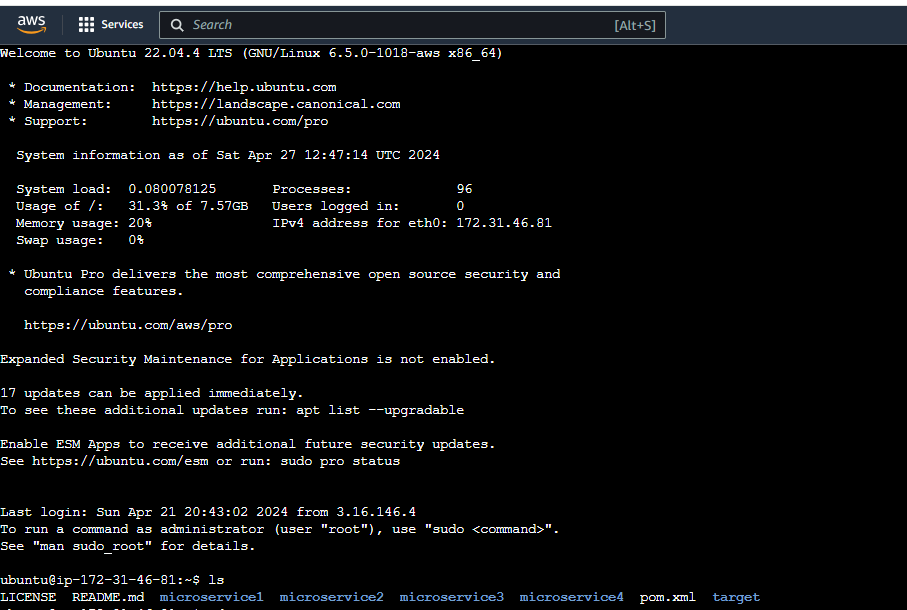
1. Go to your Instance



1. Click on Connect.
2. After going to EC2 Instance Connect, click on connect again



1. Enter ls command to check if the artifact is added here

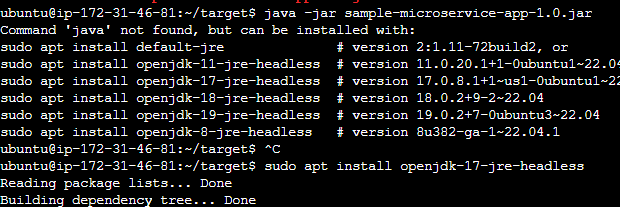


1. Go to target folder where artifact is present



1. Execute the jar command:

java -jar sample-microservice-app-1.0.jar



If you see in the above screenshot, it throwed error of java not found.

So we have to install the java 17 version.

Command: sudo apt install openjdk-17-jre-headless

1. We will re-run the java jar command again



Here the problem is with development code. It cannot find the manifest.mf file.