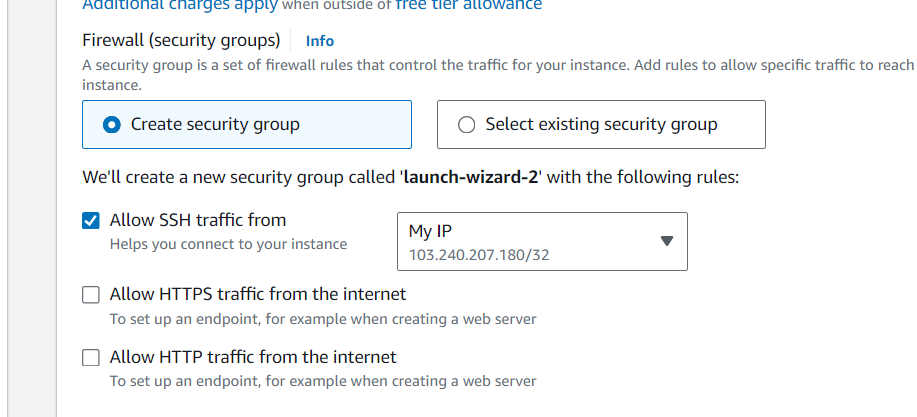
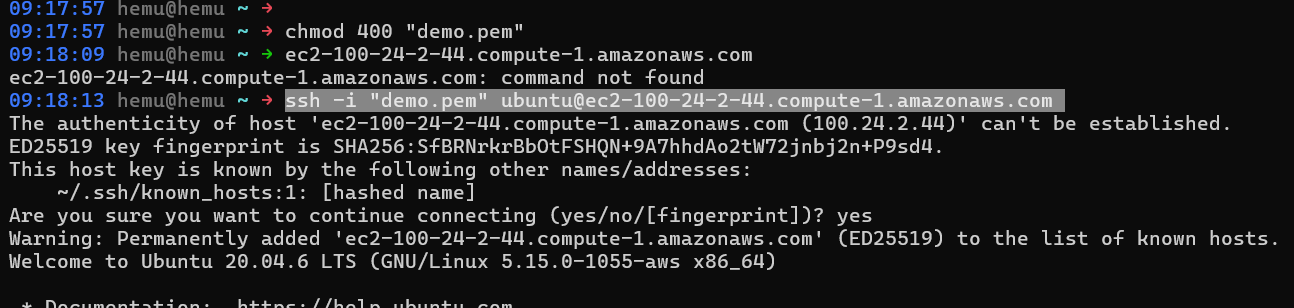
Created an account in AWS (use free tier services & criteria for handson – run the app locally and accessing from browser )

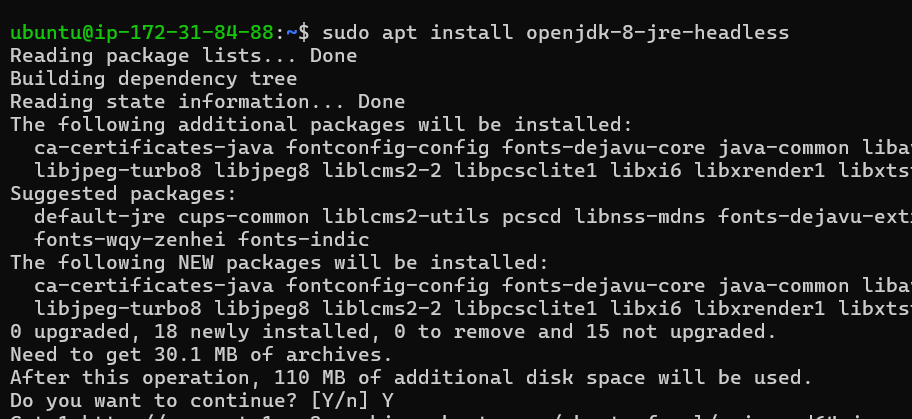
Created an EC2 instance (ubuntu OS free tier)

Configured Firewall rule to open port 22 for your IP address



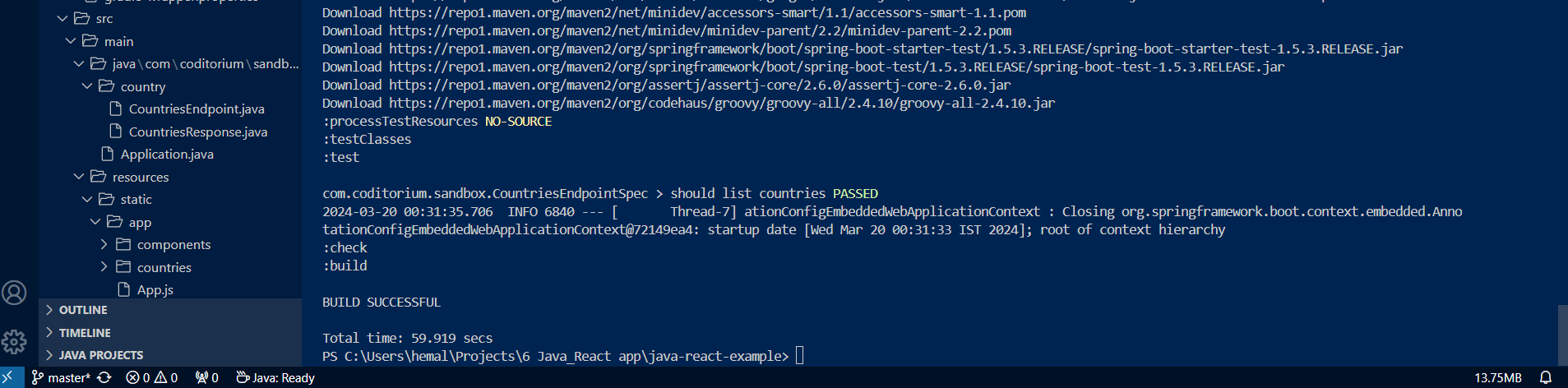
Connected to ec2 instance from SSH client

Installed java 8

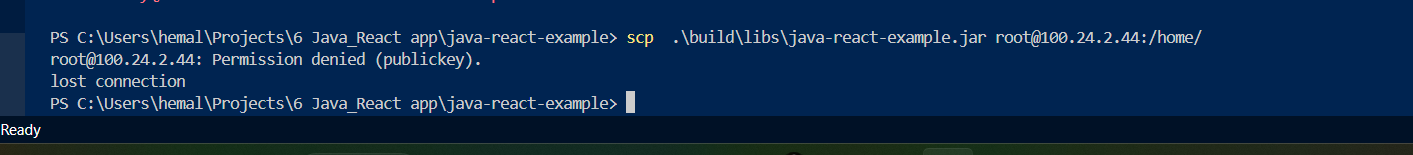


Clone this repo locally : https://github.com/pmendelski/java-react-example.git

Built Jar File with ./gradlew build command

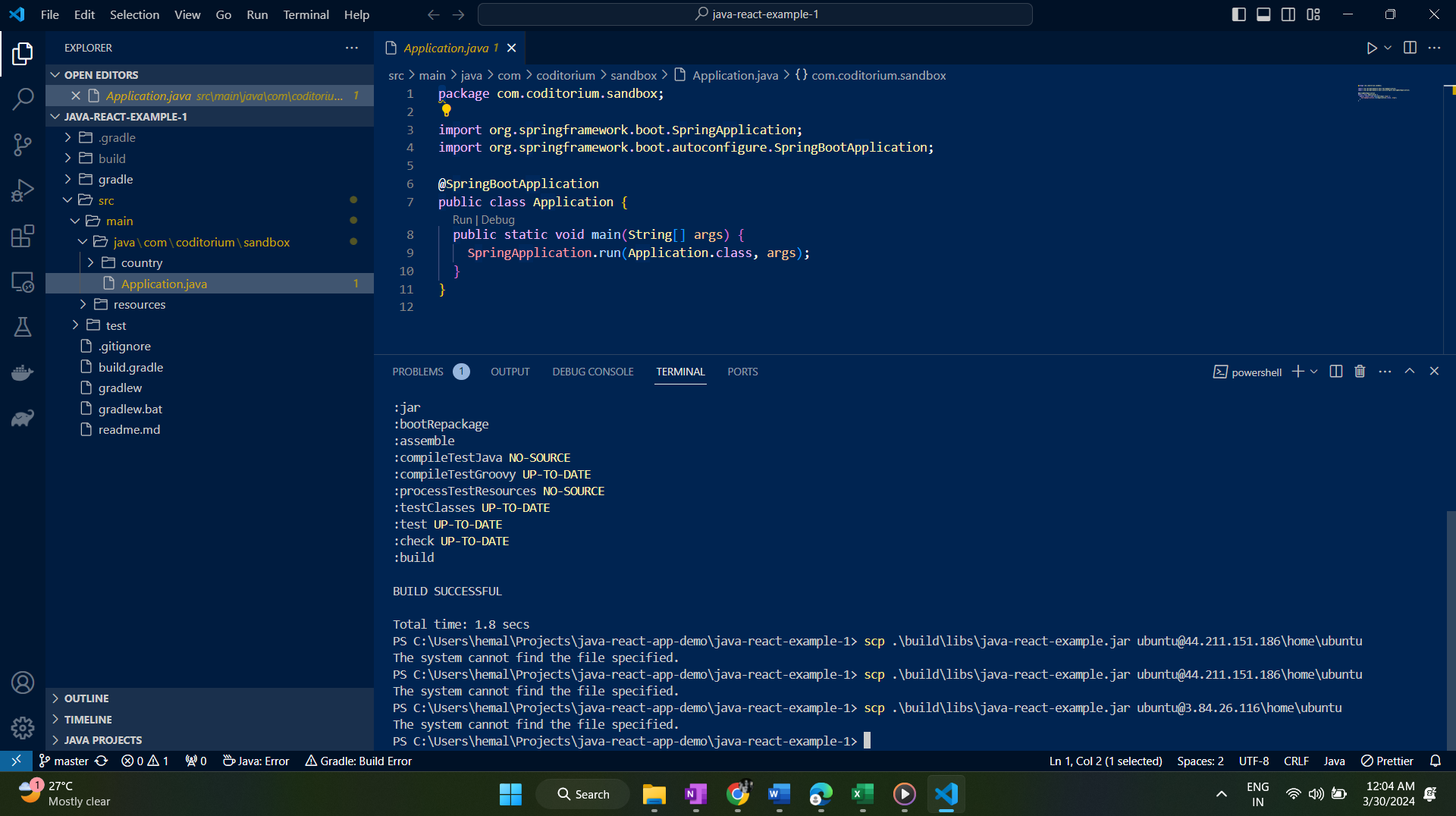


Scp = secure copy 🡪 to run the jar file on the server use <scp file destination>



After building .. vscode is unable to locate .jar file

This error was coming because I missed : after public dns name of ec2 instance



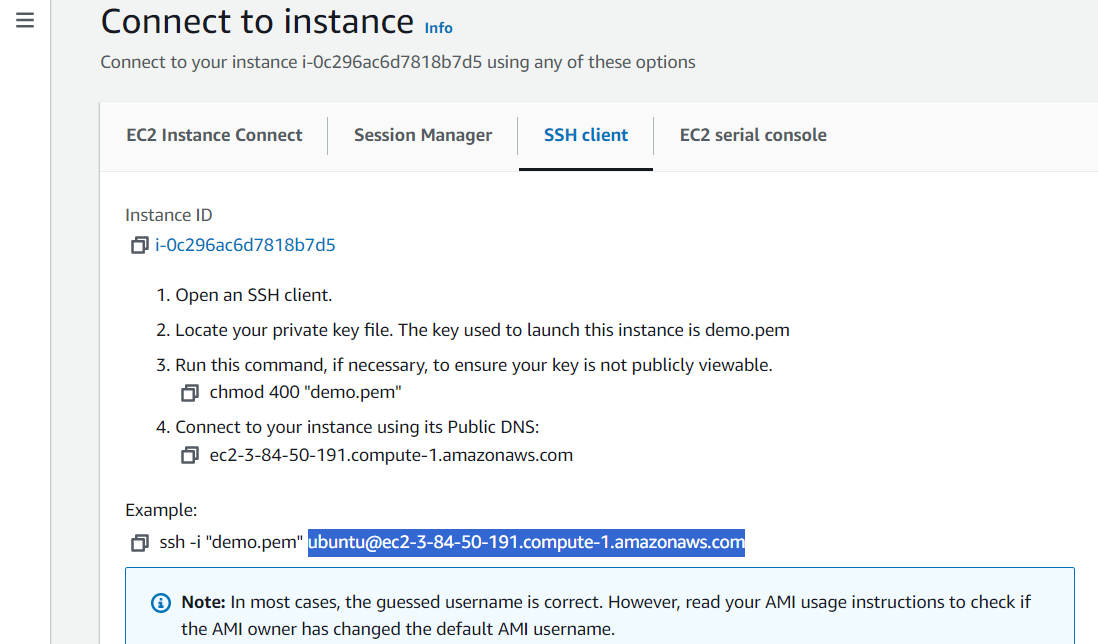
Having figured it out.. now make sure to be in the folder locally where the key pair is kept (downloaded initially at launch of instance)

Make sure to use full dns name and not just the public ip as its giving connection lost error

[ubuntu@ec2-3-84-50-191.compute-1.amazonaws.com](mailto:ubuntu@ec2-3-84-50-191.compute-1.amazonaws.com)

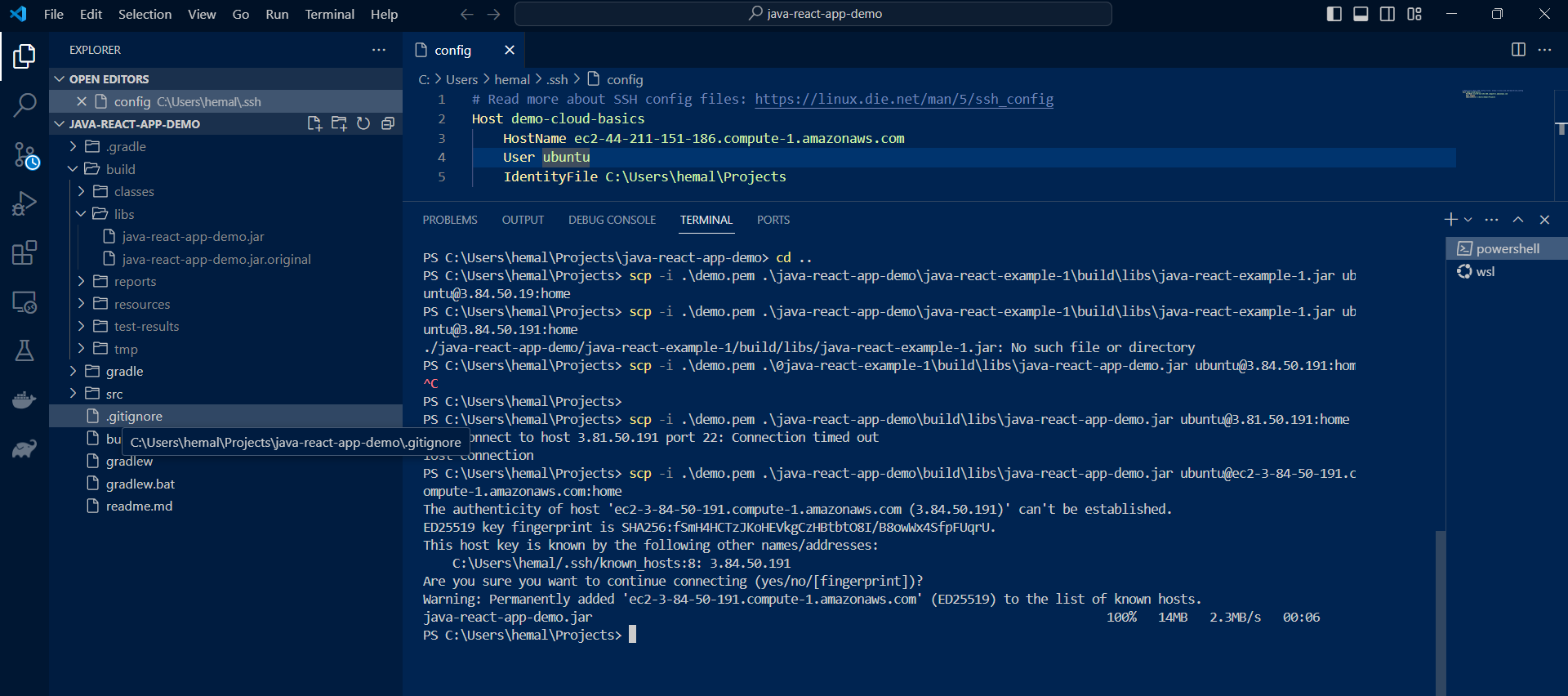
like this

available under connect tab of ec2

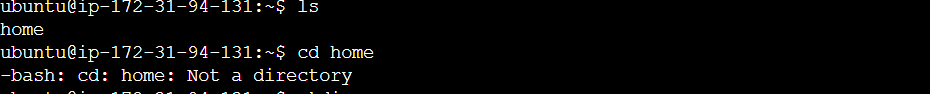


Now run the below command to copy jar file from local to ec2 home directory

scp -i .\demo.pem .\java-react-app-demo\build\libs\java-react-app-demo.jar [ubuntu@ec2-3-84-50-191.compute-1.amazonaws.com:home](mailto:ubuntu@ec2-3-84-50-191.compute-1.amazonaws.com:home)



Wohooo… finally uploaded jar file to ec2 instance.



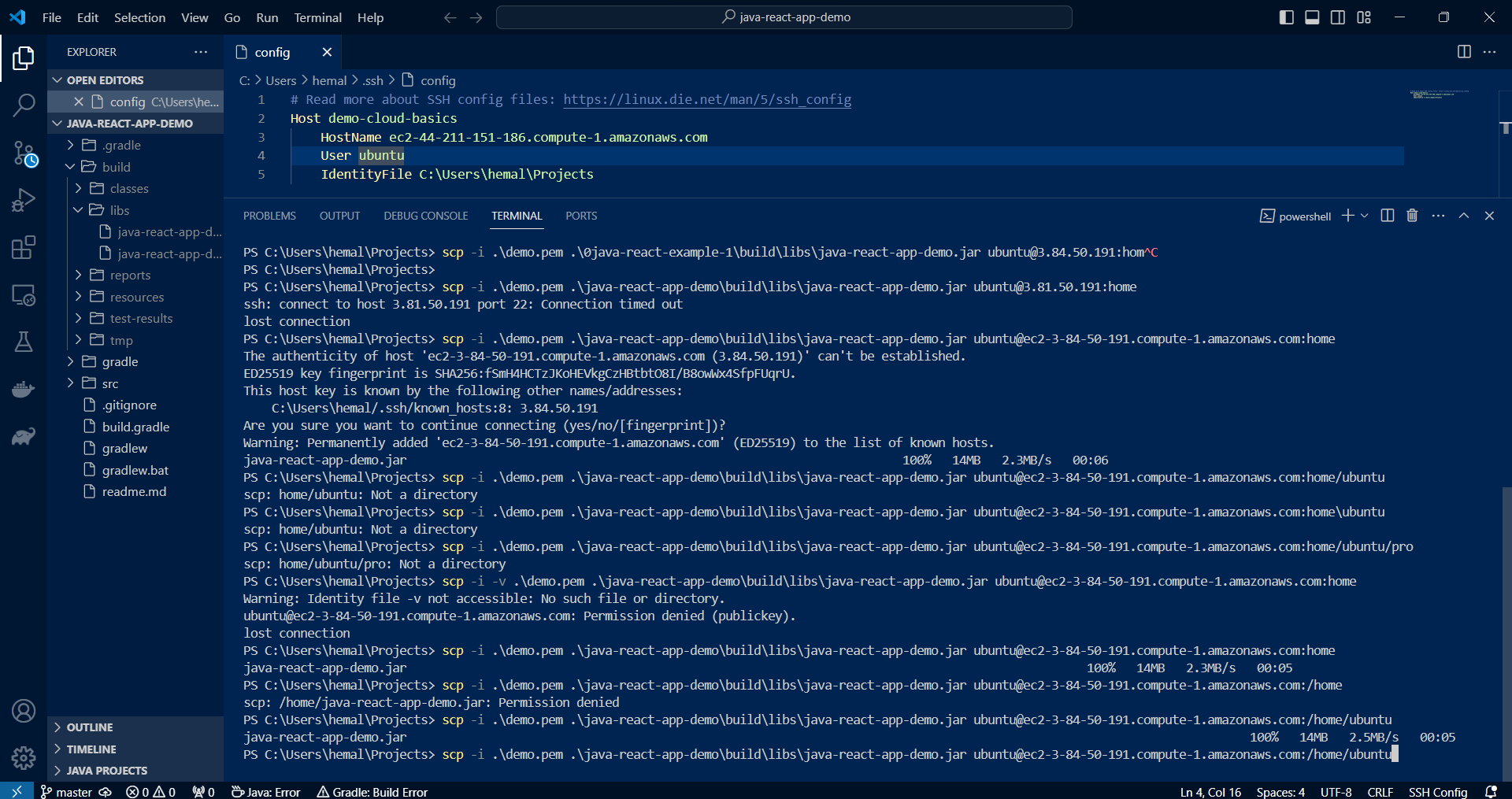
Still not available on server

Make sure to give read and write permission to .pem file from UI of windows

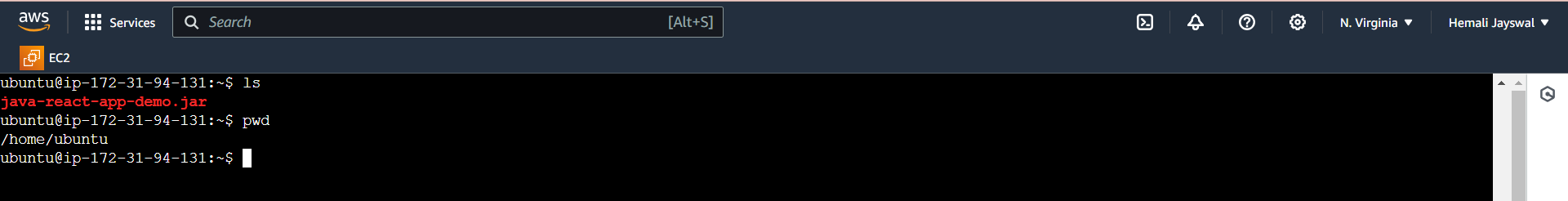
Also make sure to give “/ “ 🡪 root folder location after colon else it will not copy the file

So the command now modifies to:

scp -i .\demo.pem .\java-react-app-demo\build\libs\java-react-app-demo.jar [ubuntu@ec2-3-84-50-191.compute-1.amazonaws.com:/home/ubuntu](mailto:ubuntu@ec2-3-84-50-191.compute-1.amazonaws.com:/home/ubuntu)

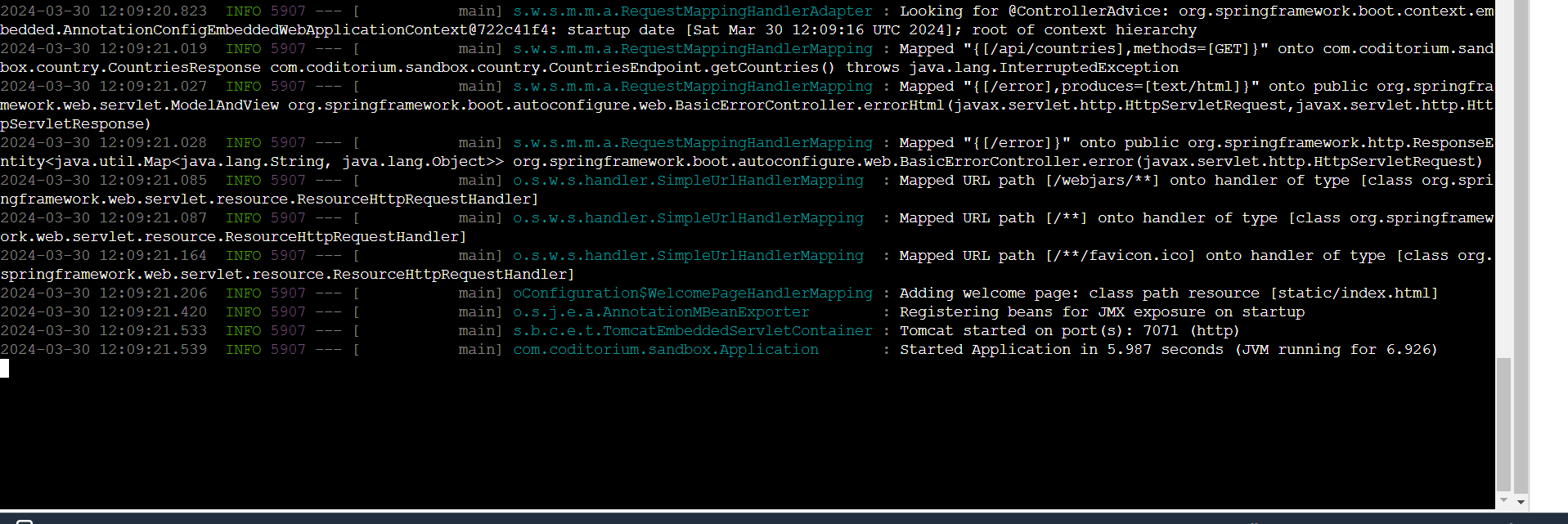
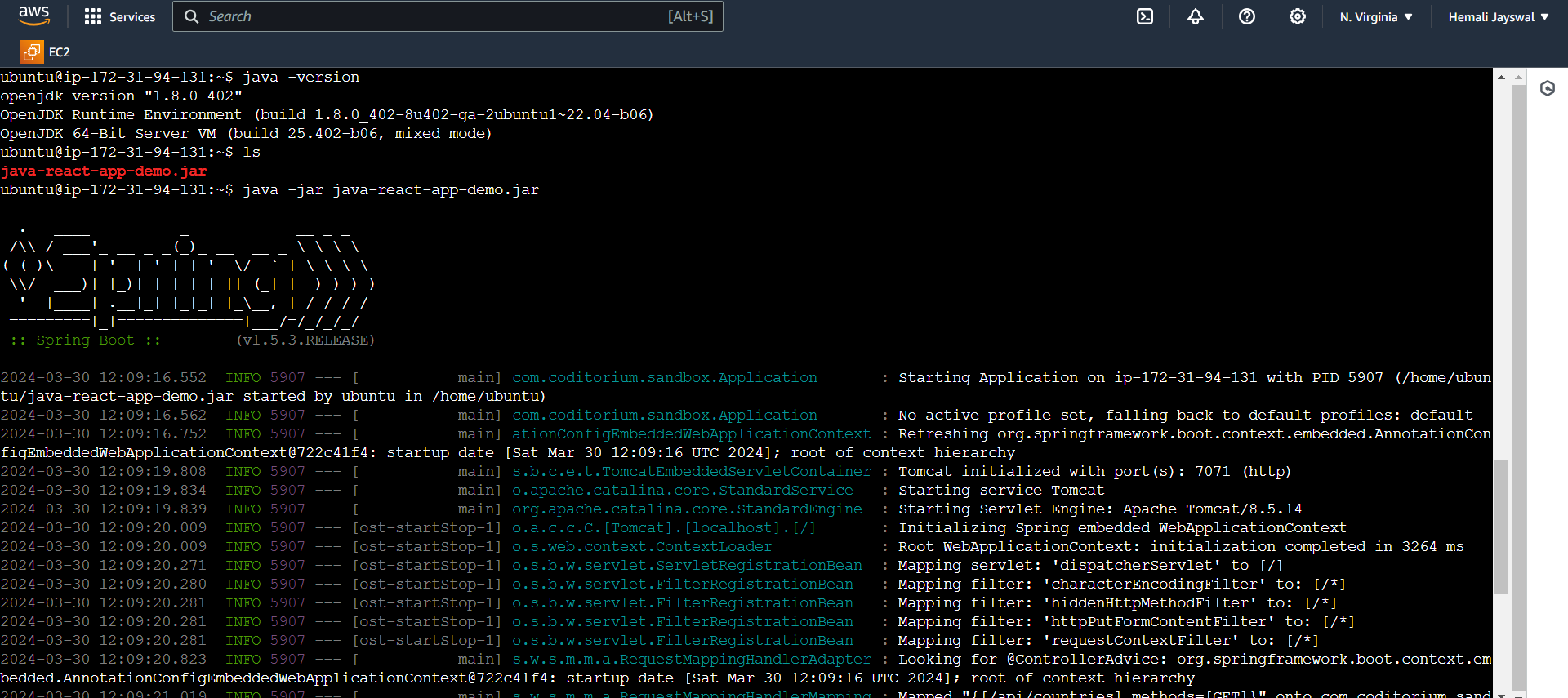


Checking on server:



Run the jar file on server

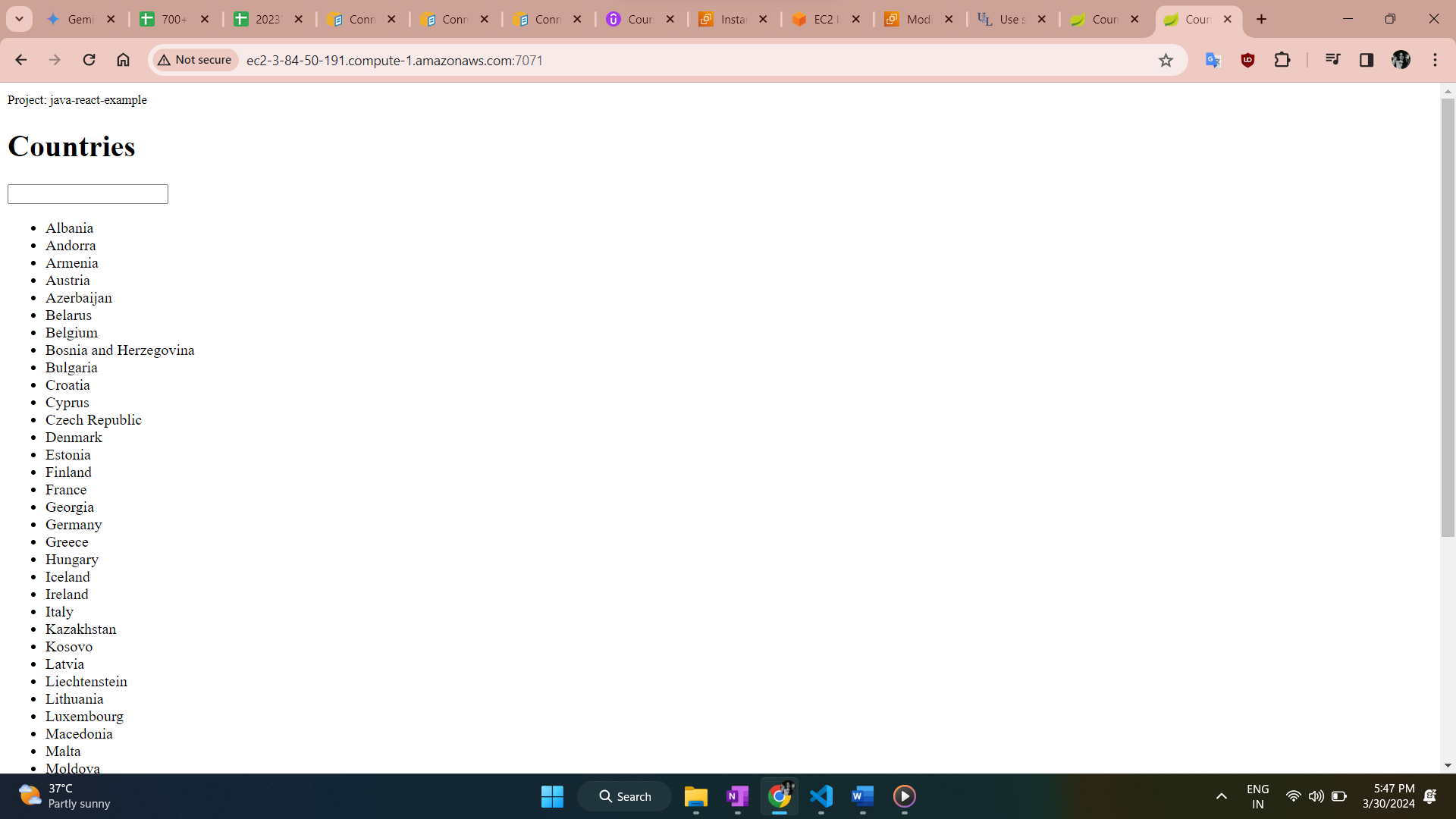
Java -jar java-react-app-demo.jar



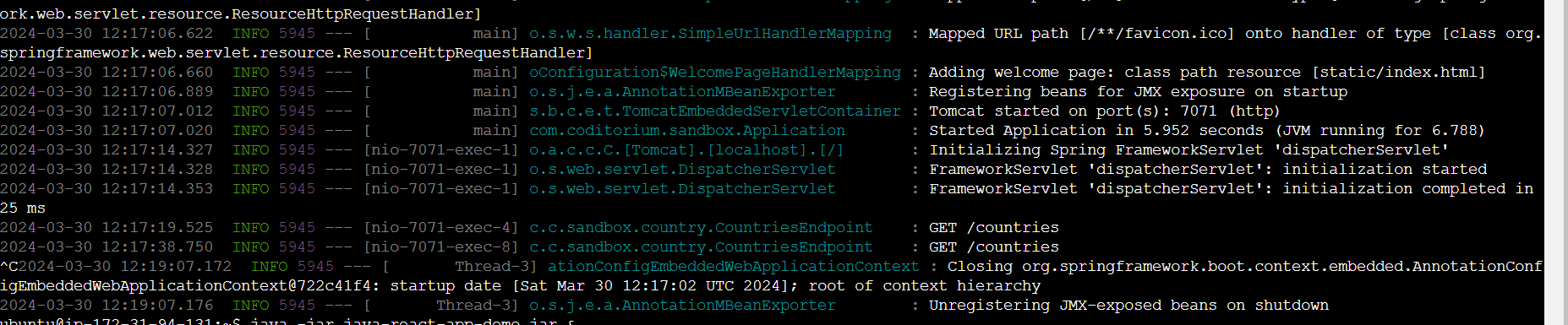
Application is running on port 7071

2024-03-30 12:09:21.533 INFO 5907 --- [ main] s.b.c.e.t.TomcatEmbeddedServletContainer : Tomcat started on port(s): 7071 (http)

So using public IP of the instance on port7071, anyone can access this app



Requests made can be seen on the server itself



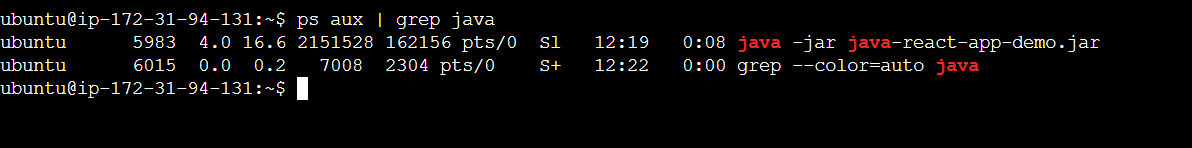
To run the application in detached mode: use & at the end

ubuntu@ip-172-31-94-131:~$ java -jar java-react-app-demo.jar &

To check whether the application is running or not or see the process id of the app

use below cmd:

ubuntu@ip-172-31-94-131:~$ ps aux | grep java

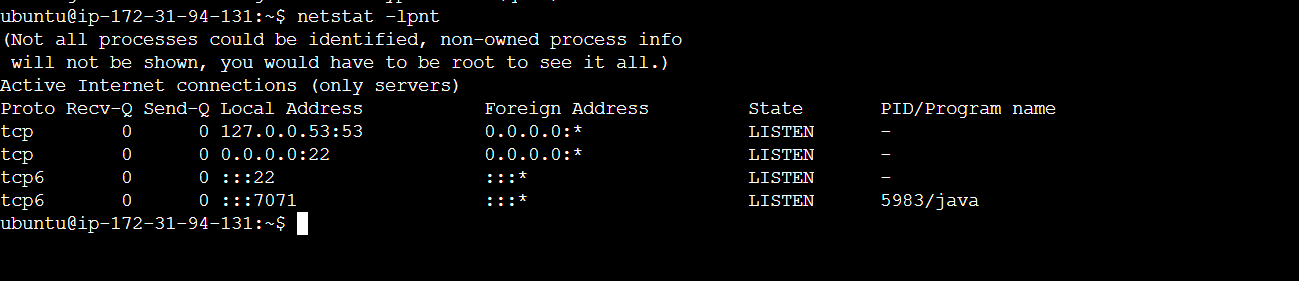


5983 is process id

If we don’t know the port on which app is running we can find using netstat command

ubuntu@ip-172-31-94-131:~$ netstat -lpnt

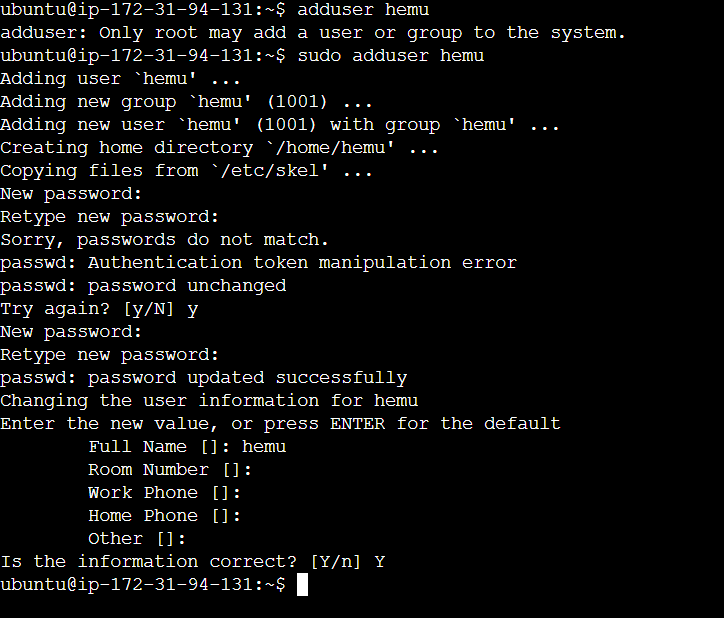
lists servers with active internet connections



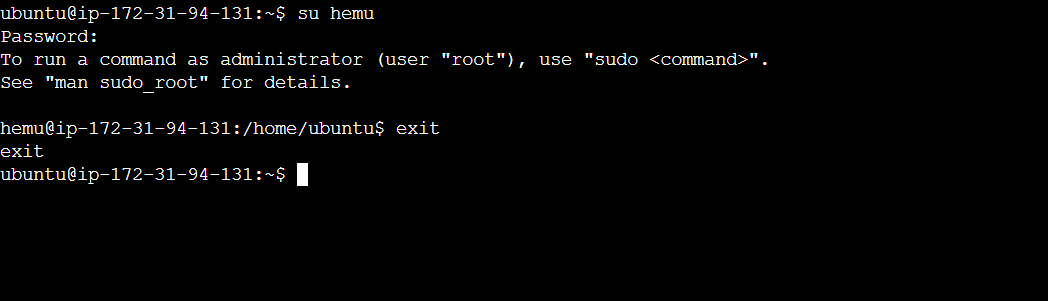
Security best practice is to not use root user for day-to-day activities, create user with admin user

(permission given) and then do the required tasks

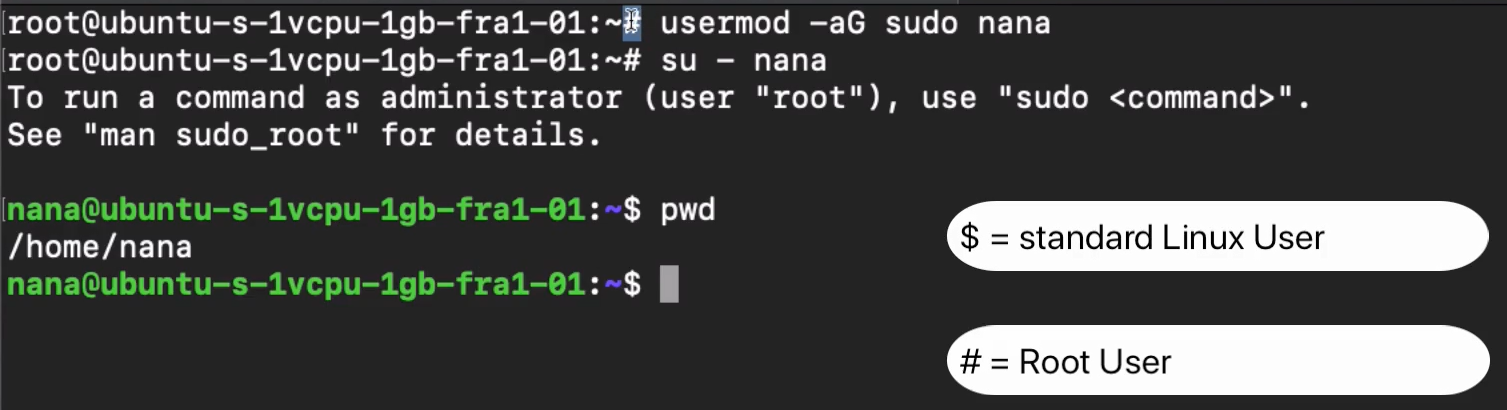
* Run: adduser hemu
* Add user to sudo group to give them root user access:



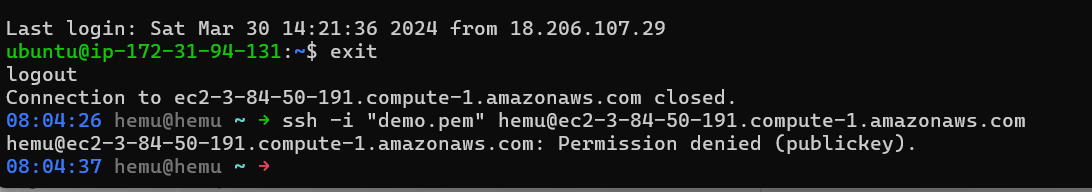
Usermod -aG sudo hemu



Switch user hemu, exit from user



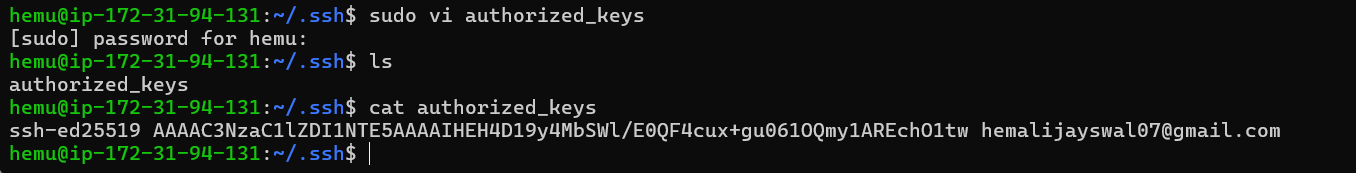
If we try to ssh into the instance using this user, it gives permission denied error as the demo.pem key is added just for root user, for this user we need to add .pem key separately



Copy the .pem public key from local machine, wherever its stored

ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIHEH4D19y4MbSWl/E0QF4cux+gu061OQmy1AREchO1tw [hemalijayswal07@gmail.com](mailto:hemalijayswal07@gmail.com)

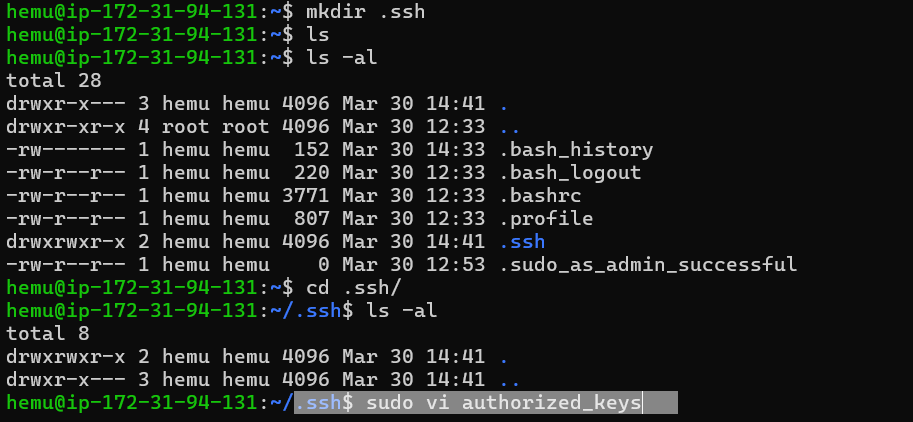
* Again ssh to ec2 instance , su – hemu
* Create .ssh folder in user home directory
* Create authorized\_keys file in .ssh folder



Now we should be able to ssh into the server using this hemu user

🡺 or copy authorized key file content from root user in .ssh folder

Paste the same content in hemu user .ssh folder authorized\_key file content



Now we are able to ssh into the server even using hemu user

