Urvish Jaiswal

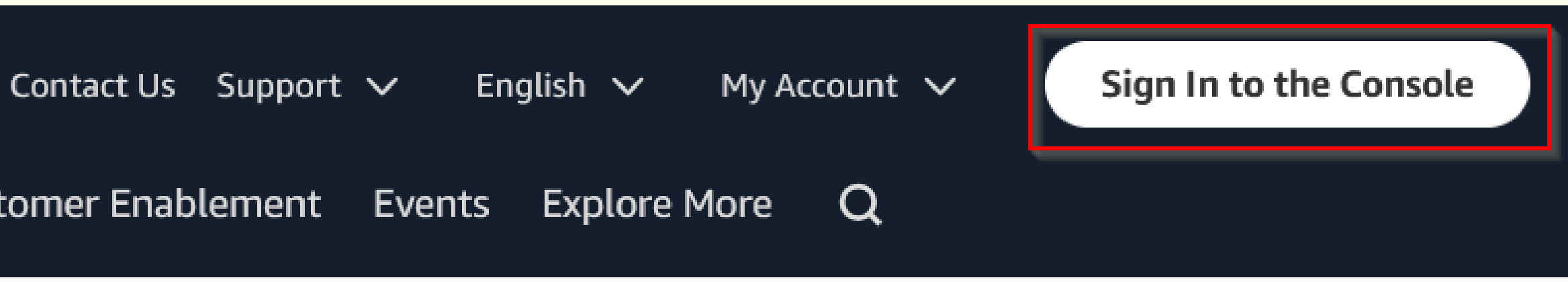
AWS EC2

For Beginners

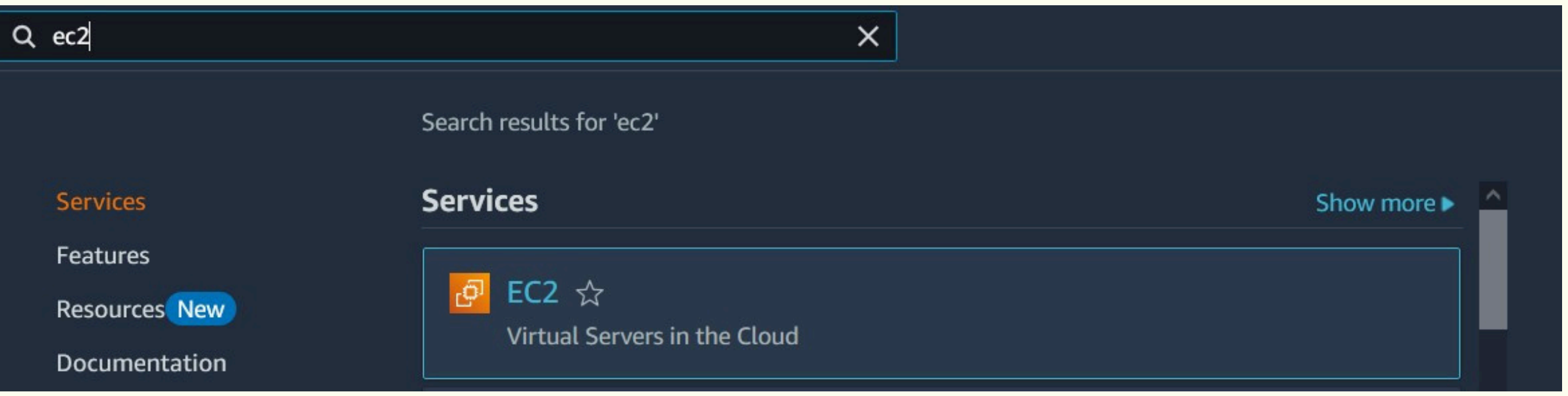
Learn AWS EC2 instance With Project

Swipe for more

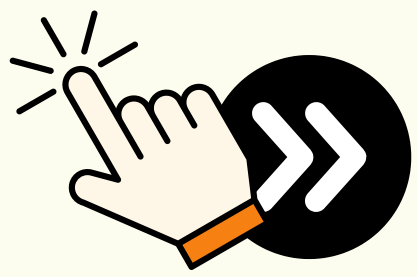
Step 1: Log in to AWS Console



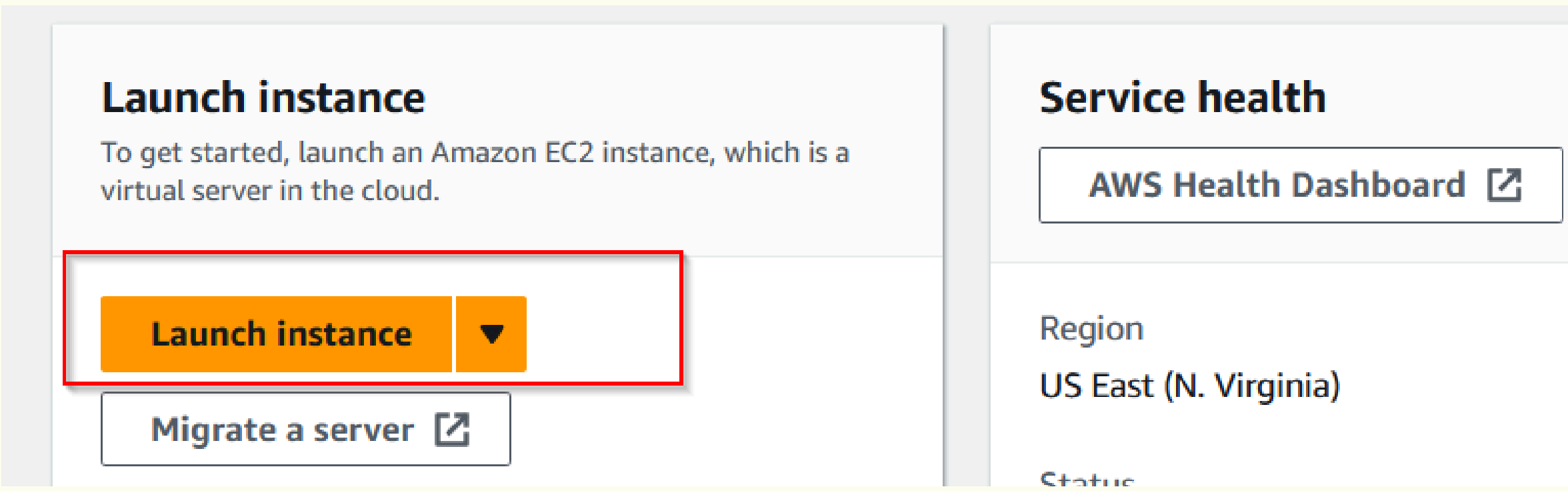
Step 2: Open EC2 Dashboard



Swipe for more



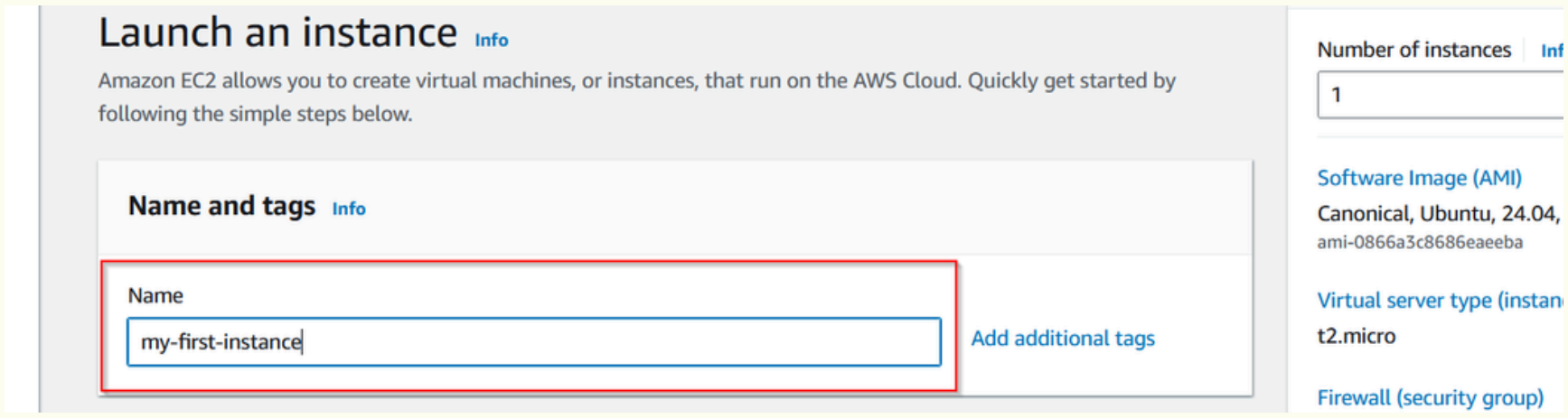
Step 3: Click “Launch Instance”



You will find these option just after logging into or EC2

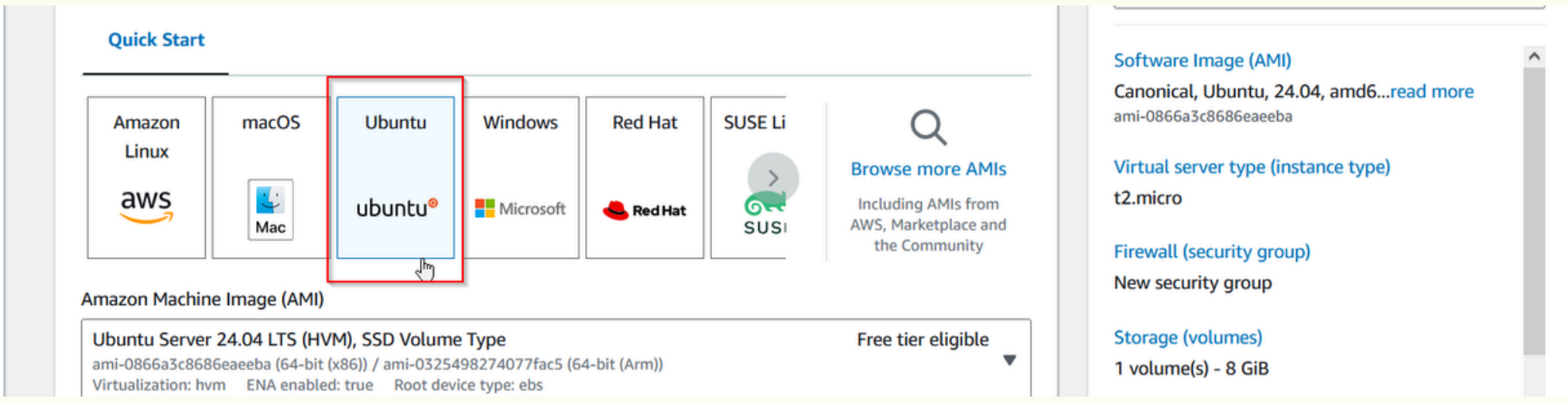
Step 4: EC2 instance Config

4.1: Name the instance

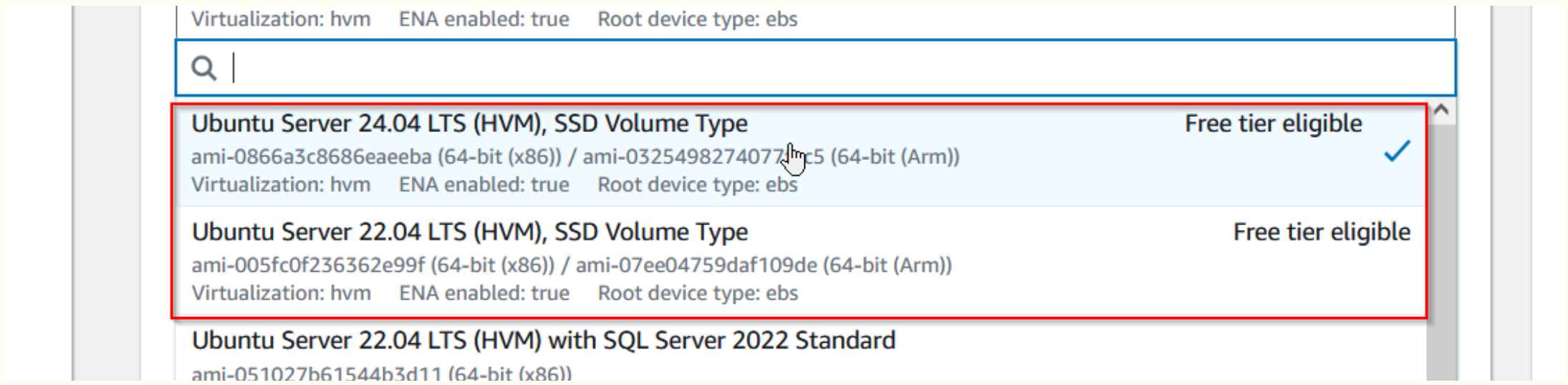


Swipe for more

4.2:Choose an AMI (Amazon Machine Image)

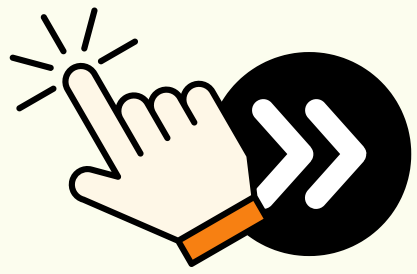


Since we're using the free tier, it's important to select a lightweight OS. I recommend Ubuntu, but feel free to choose one that suits your needs best.

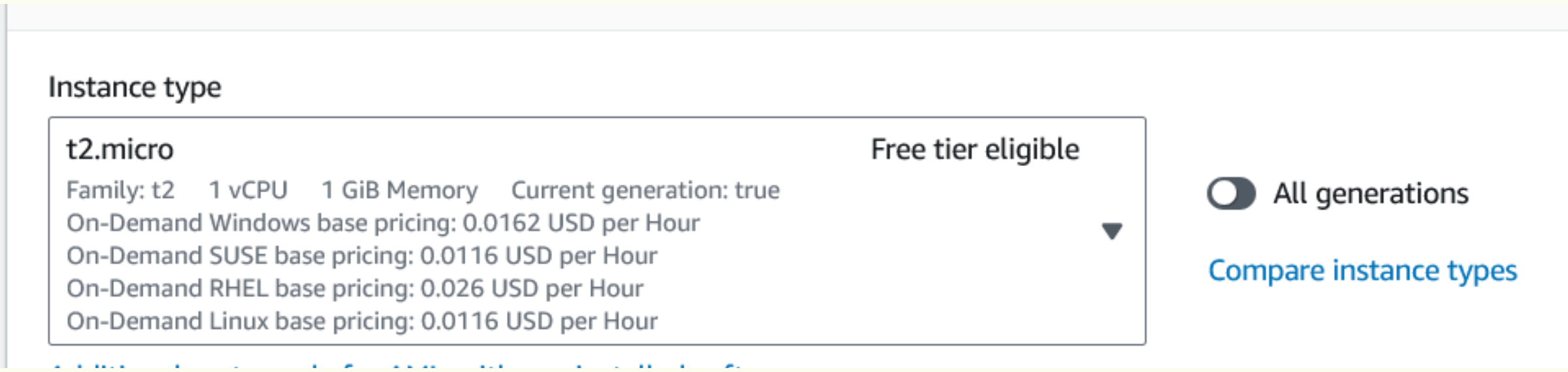


Here we can select any LTS version of Ubuntu which falls under free tier

Swipe for more

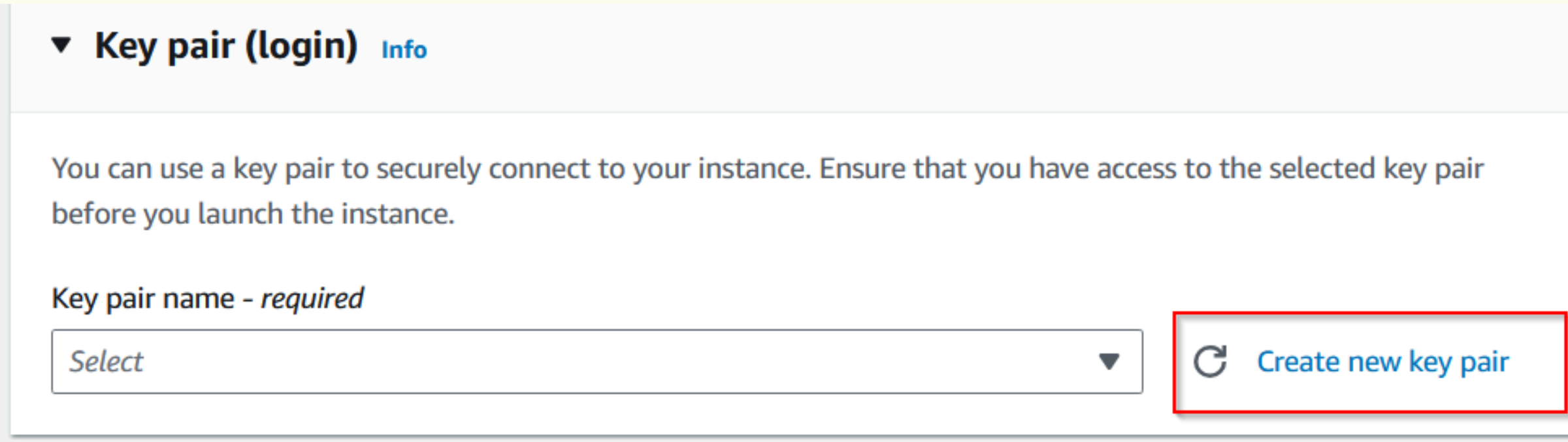


4.3: Select an Instance Type



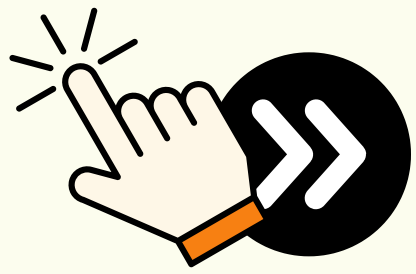
Here we can select any LTS version of Ubuntu which falls under free tier

4.4: Select an Key Pair

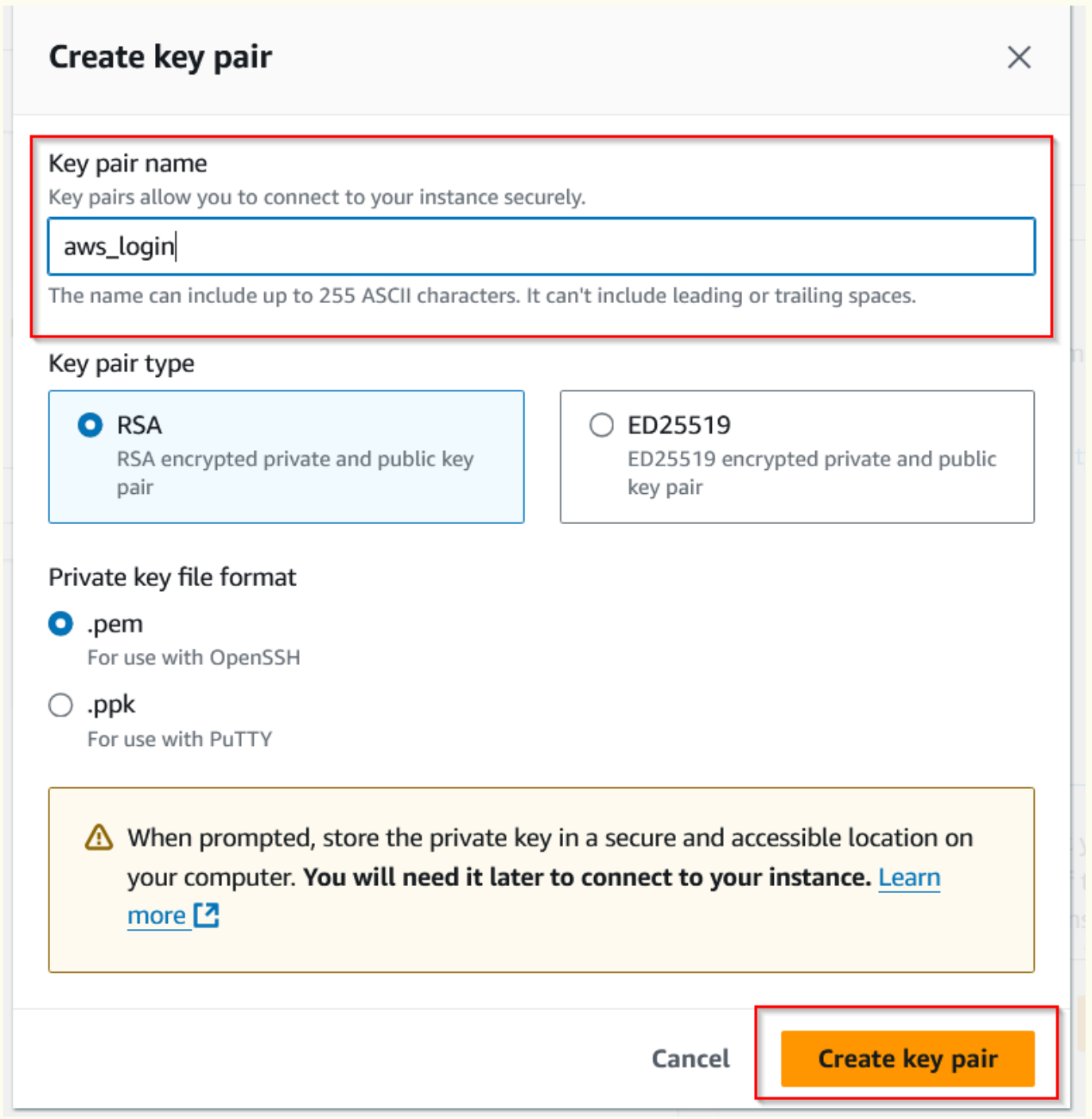


A key pair in AWS EC2 is a security credential (a combination of a public and private key) used to securely connect to your instance. You'll need the private key to access your server via SSH or RDP.

Swipe for more



4.4.1: Name Your Key Pair



Step 1: Name Your Key Pair: Give your key pair a name (e.g., "aws_login").

Step 2: Select Key Format: Choose the format for the private key file—PEM for SSH (Linux) or PPK for PuTTY (Windows).

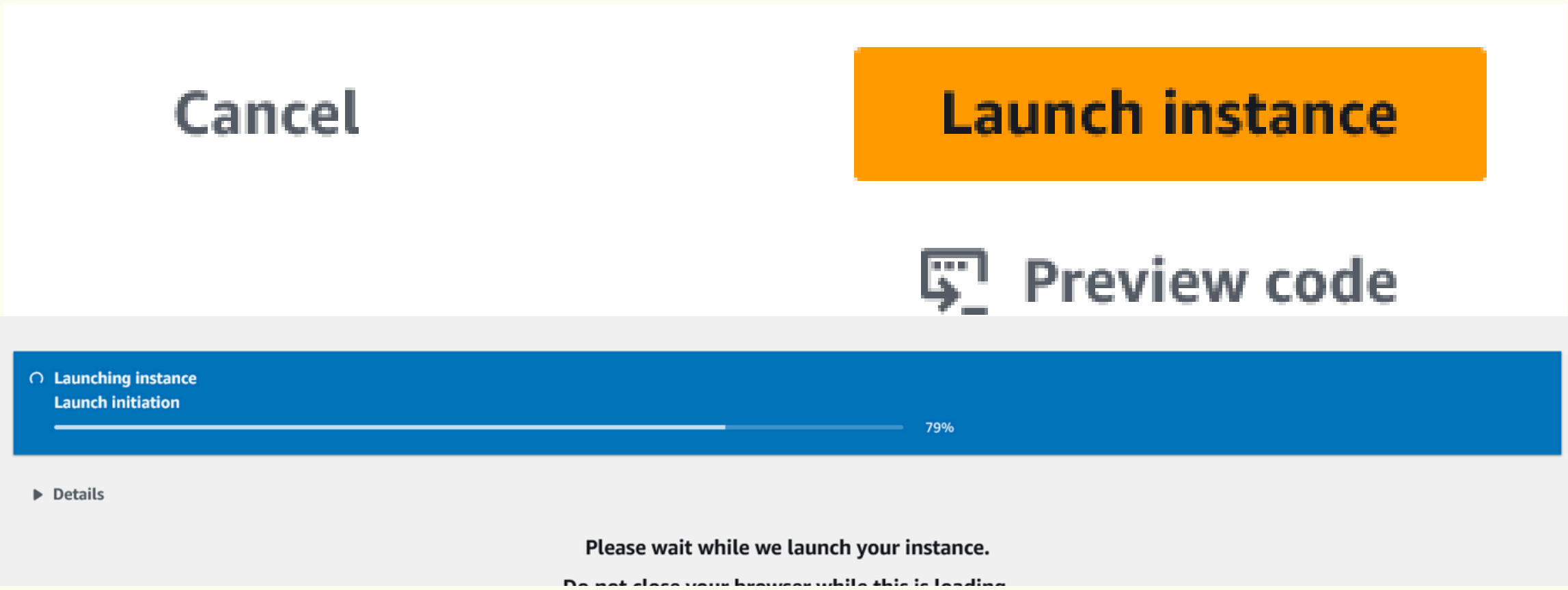
Step 3: Download Key Pair: Click Create key pair, and the private key file will be automatically downloaded. Store it securely.

Saving your private key is crucial because without it, you can't access your EC2 instance, and AWS won't let you download it again. Keep it safe and secure!

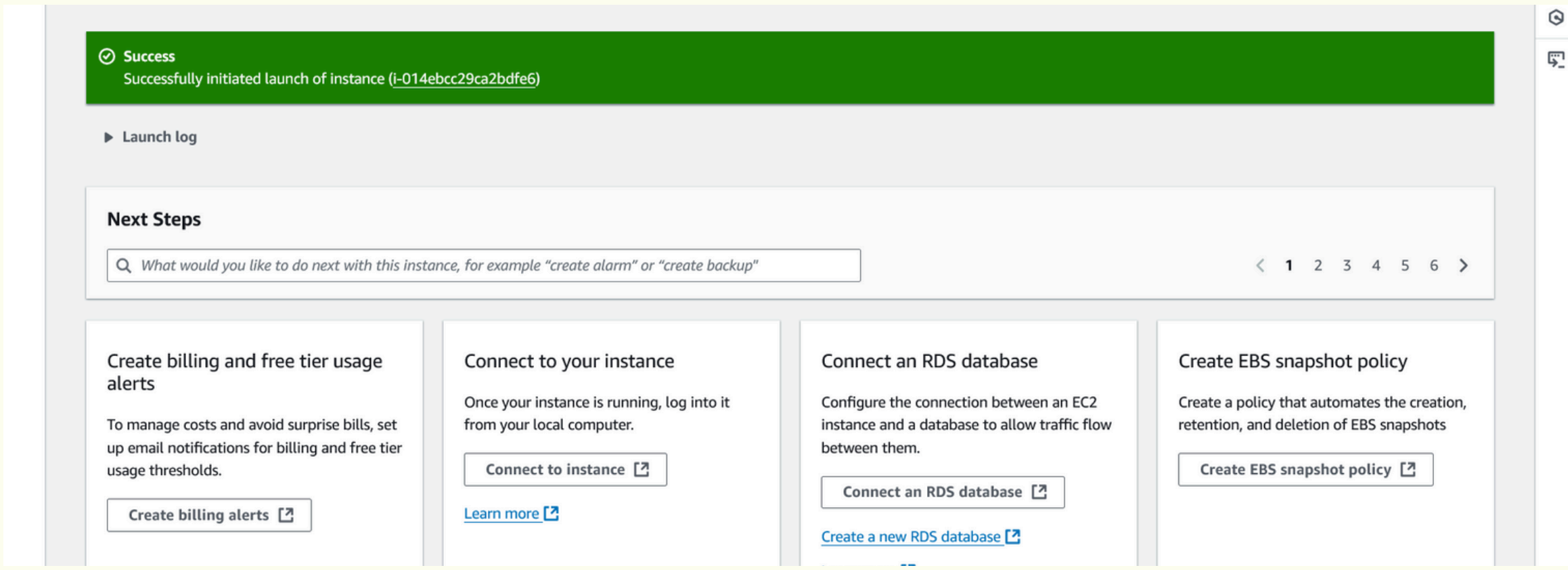
Swipe for more

4.5: Launch Your Instance

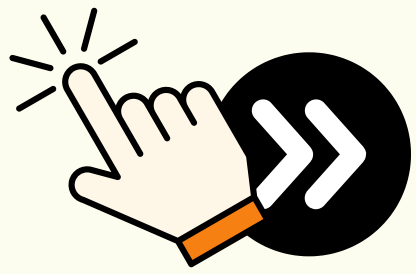
After downloading the key pair, click Launch Instances. Your EC2 instance is now being created!



Congratulations🎉!!!! If you followed me along properly you would see the interface like me. That means your EC2 instance created successfully

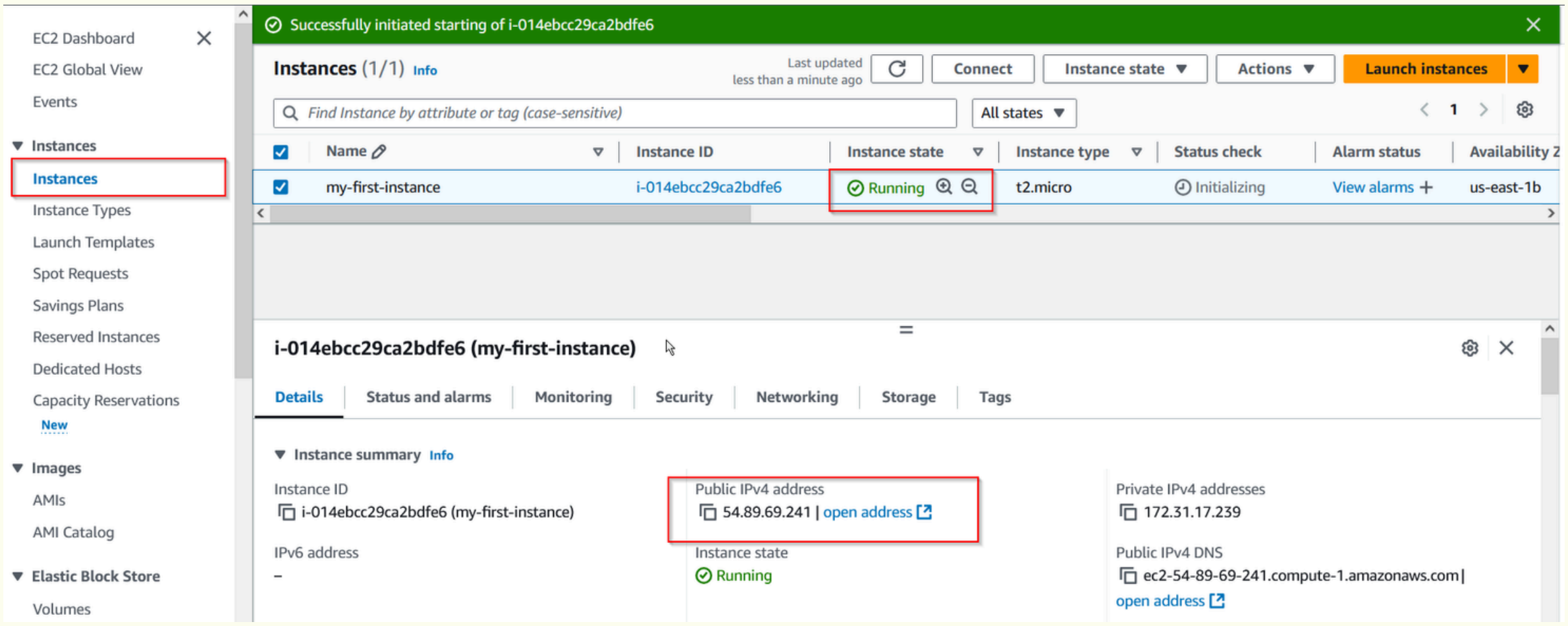


Swipe for more



4.6: Connect with Your Instance and install Jenkins

Step1 : Copy the Public IPv4 address we need it in our next Step



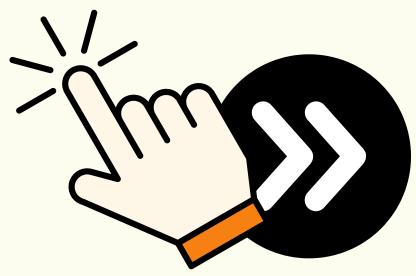
If you're a Windows user, I recommend using Git Bash to log into the instance we created via SSH. Alternatively, you can also use PuTTY.

Step2 : Using CMD go to the folder where you kept your keys. In my case its underneath of Downloads folder

Step3 : Use Command “ssh -i keyname image_name@copied ipv4 address”

```
Urvish Jaiswal@MUSS408 MINGW64 ~/OneDrive/Pictures/project/Downloads
$ ssh -i aws_login.pem ubuntu@18.206.173.203
The authenticity of host '18.206.173.203 (18.206.173.203)' can't be established.
ED25519 key fingerprint is SHA256:5G6ZfgLvNO65VW69qqU1tFK7d9uNjCN5mpbb2WCOBks.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
```

Swipe for more



Note : During ssh login we would going to face these issue of not login

```
rvish Jaiswal@MUSS408 MINGW64 ~/OneDrive/Pictures/project/Downloads
 ssh -i aws_login.pem ubuntu@18.206.173.203
he authenticity of host '18.206.173.203 (18.206.173.203)' can't be established.
D25519 key fingerprint is SHA256:5G6ZfgLvNO65VW69qqU1tFK7d9uNjCN5mpbb2WCOBks.
his key is not known by any other names
re you sure you want to continue connecting (yes/no/[fingerprint])? yes
arning: Permanently added '18.206.173.203' (ED25519) to the list of known hosts.
onnection closed by 18.206.173.203 port 22
```

Reason behind that is simple we need to change the permissions of key file which we have downloaded previously to chmod 600

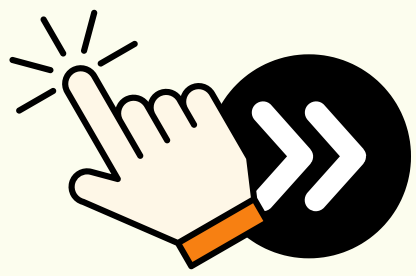
```
Urvish Jaiswal@MUSS408 MINGW64 ~/OneDrive/Pictures/project/Downloads
$ chmod 600 aws_login.pem
```

Step4 : If changed the permission you would easily be able to login as a ubuntu user

Step5 : We are going to install JDK. Its important to have before we install jenkins. In order to install any software we need to update apt in Ubuntu.

```
root@ip-172-31-17-239:~# apt update
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble InRelease
Hit:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease
Hit:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease
Hit:4 http://security.ubuntu.com/ubuntu noble-security InRelease
Reading package lists... Done
```

Swipe for more



Step6 : Now install JDK-11

```
ubuntu@ip-172-31-17-239:/$ sudo apt install openjdk-11-jdk
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
```

Confirm by checking the version

```
ubuntu@ip-172-31-17-239:/$ java --version
openjdk 11.0.24 2024-07-16
OpenJDK Runtime Environment (build 11.0.24+8-post-Ubuntu-1ubuntu324.04.1)
OpenJDK 64-Bit Server VM (build 11.0.24+8-post-Ubuntu-1ubuntu324.04.1, mixed mode, sharing)
ubuntu@ip-172-31-17-239:/$
```

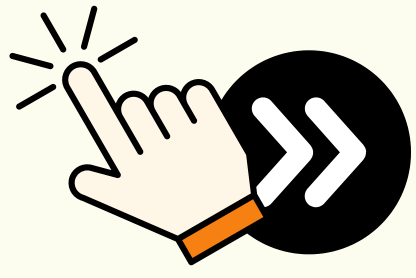
Step7 : Now install Jenkins follow the command

```
ubuntu@ip-172-31-17-239:/$ sudo wget -O /usr/share/keyrings/jenkins-keyring.asc \
  https://pkg.jenkins.io/debian/jenkins.io-2023.key
echo "deb [signed-by=/usr/share/keyrings/jenkins-keyring.asc]" \
  https://pkg.jenkins.io/debian binary/ | sudo tee \
  /etc/apt/sources.list.d/jenkins.list > /dev/null
sudo apt-get update
sudo apt-get install jenkins
--2024-10-21 02:59:06--  https://pkg.jenkins.io/debian/jenkins.io-2023.key
Resolving pkg.jenkins.io (pkg.jenkins.io)... 146.75.34.133, 2a04:4e42:79::645
Connecting to pkg.jenkins.io (pkg.jenkins.io)|146.75.34.133|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 3175 (3.1K) [application/pgp-keys]
Saving to: '/usr/share/keyrings/jenkins-keyring.asc'

/usr/share/keyrings/jenkins-keyring.asc             100%[=============================================================================================
```

Step8 : Enable the service and check the status of Jenkins

```
-172-31-17-239:~$ sudo systemctl enable jenkins
zing state of jenkins.service with SysV service script with /usr/lib/systemd/systemd-sysv-install.
 /usr/lib/systemd/systemd-sysv-install enable jenkins
-172-31-17-239:~$ sudo systemctl status jenkins
 service - Jenkins Continuous Integration Server
ed: loaded (/usr/lib/systemd/system/jenkins.service; enabled; preset: enabled)
ve: active (running) since Mon 2024-10-21 03:29:43 UTC; 1min 1s ago
ID: 6058 (java)
ks: 43 (limit: 1130)
ry: 314.1M (peak: 348.7M)
PU: 14.759s
up: /system.slice/jenkins.service
    └─6058 /usr/bin/java -Djava.awt.headless=true -jar /usr/share/java/jenkins.war --webroot=/var/cache/jenkins/war --httpPort=8080

 29:37 ip-172-31-17-239 jenkins[6058]: 7341934f722245b39ac978a00b8b677f
 29:37 ip-172-31-17-239 jenkins[6058]: This may also be found at: /var/lib/jenkins/secrets/initialAdminPassword
 29:37 ip-172-31-17-239 jenkins[6058]: ************************************************************
 29:37 ip-172-31-17-239 jenkins[6058]: ************************************************************
 29:37 ip-172-31-17-239 jenkins[6058]: ************************************************************
 29:42 ip-172-31-17-239 jenkins[6058]: 2024-10-21 03:29:42.992+0000 [id=31]        INFO        jenkins.InitReactorRunner$1#onAttained: Completed initialization
 29:43 ip-172-31-17-239 jenkins[6058]: 2024-10-21 03:29:43.034+0000 [id=23]        INFO        hudson.lifecycle.Lifecycle#onReady: Jenkins is fully up and running
 29:43 ip-172-31-17-239 systemd[1]: Started jenkins.service - Jenkins Continuous Integration Server.
 29:43 ip-172-31-17-239 jenkins[6058]: 2024-10-21 03:29:43.090+0000 [id=46]        INFO        h.m.DownloadService$Downloadable#load: Obtained the updated data file for hudson.tasks.Maven
 29:43 ip-172-31-17-239 jenkins[6058]: 2024-10-21 03:29:43.092+0000 [id=46]        INFO        hudson.util.Retrier#start: Performed the action check updates server successfully at the att
/20 (END)
```

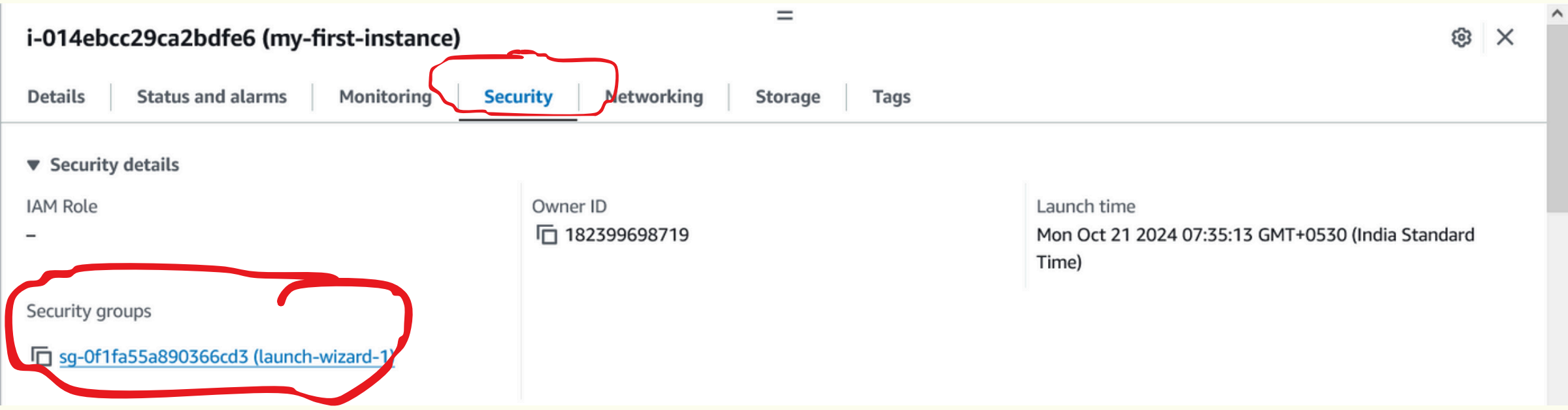
Swipe for more



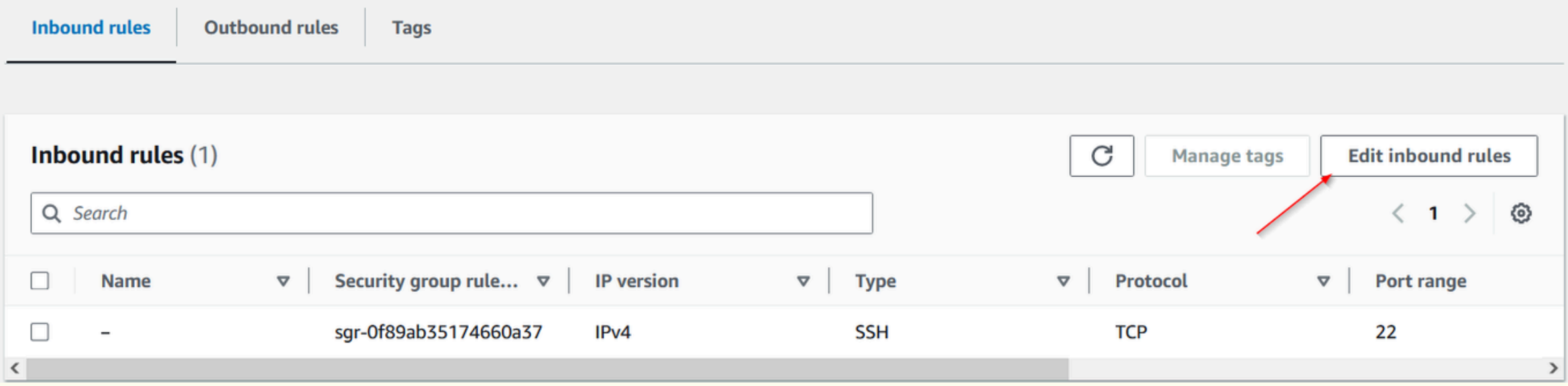
5 : Change the Security Group settings

Step1 : Go to your current running instance from Dashboard>Instance

Step2 : Select a security tab and click on security groups

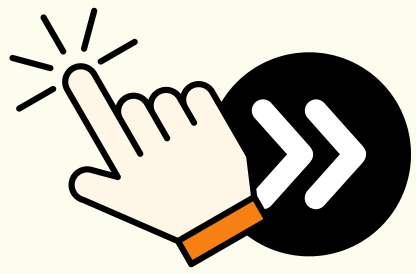


Step3 : Click on edit inbound rules

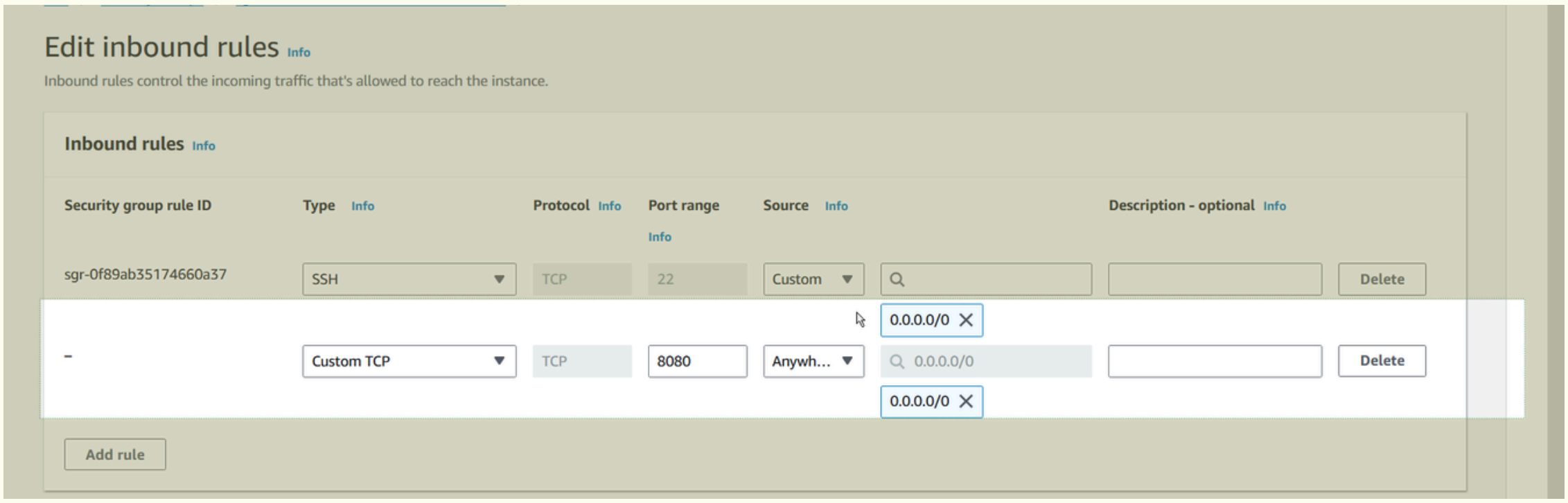


We click on "Edit Inbound Rules" to allow specific types of traffic, like HTTP or SSH, to access our server. This ensures the server can receive the right kind of connections.

Swipe for more

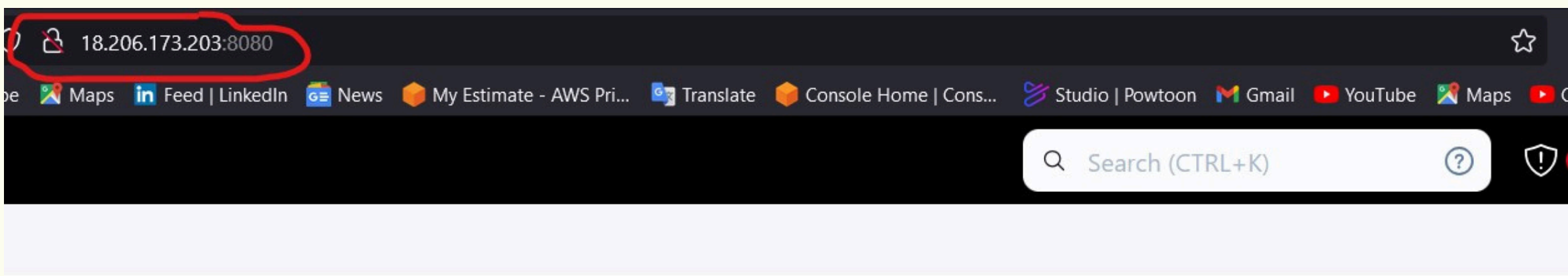


Step4 : Add the following inbound security rule



Step5 : Save the changes

6 : Run the public IP on browser with Port 8080



Swipe for more

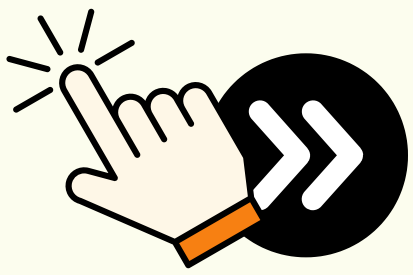
7 : Jenkins Config

Step1 : Go the cmd again and past the upper path

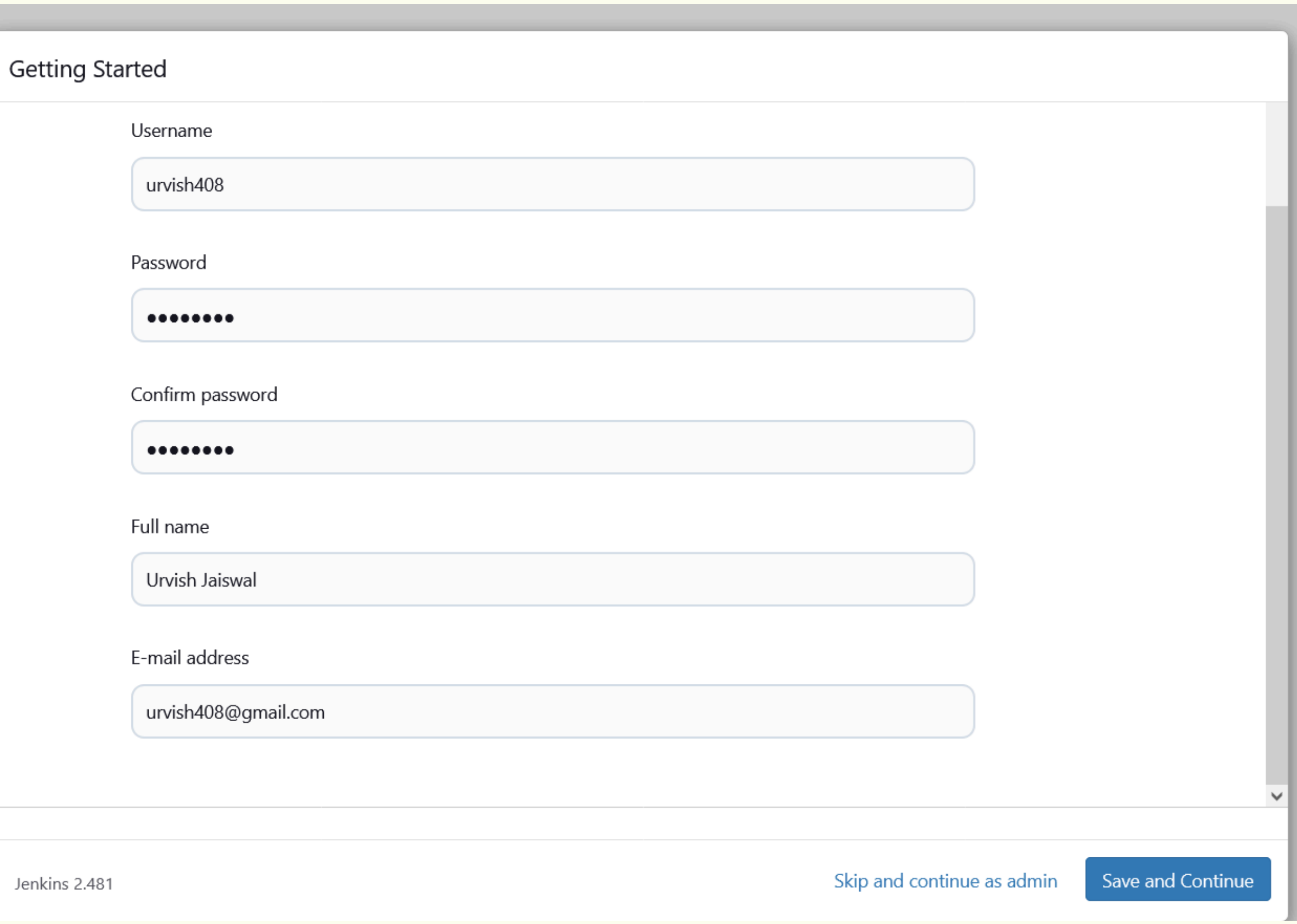
```
ubuntu@ip-172-31-17-239:~$ sudo cat /var/lib/jenkins/secrets/initialAdminPassword
7341934f722245b39ac978a00b8b677f
ubuntu@ip-172-31-17-239:~$
```

Step2 : Copy the pass and paste on to the Jenkins

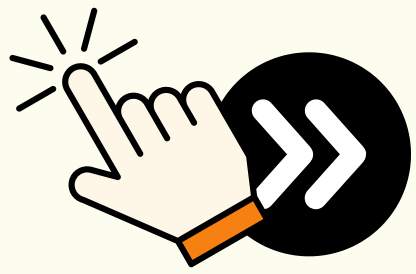
Swipe for more



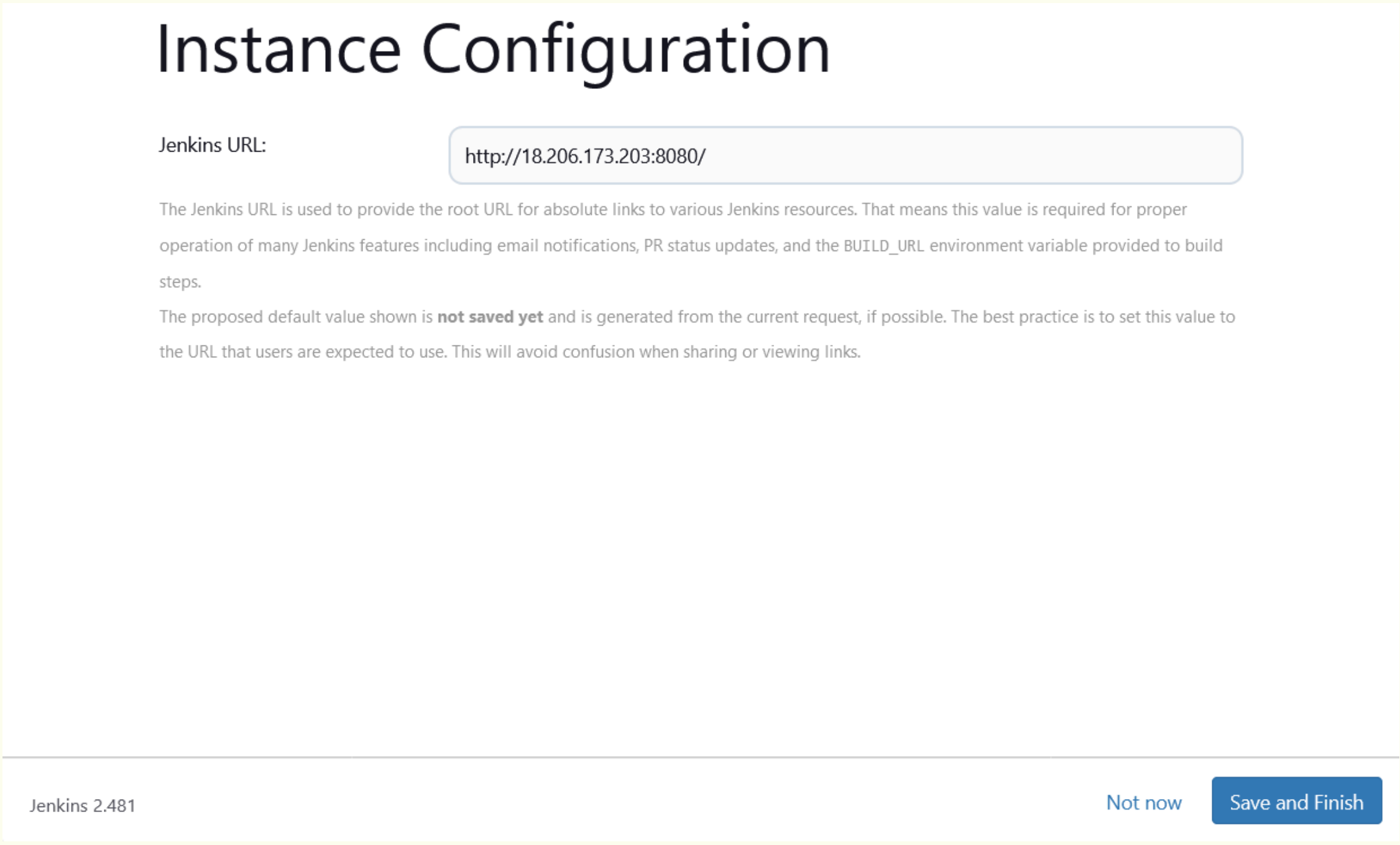
Step3 : Fill up the formand click next



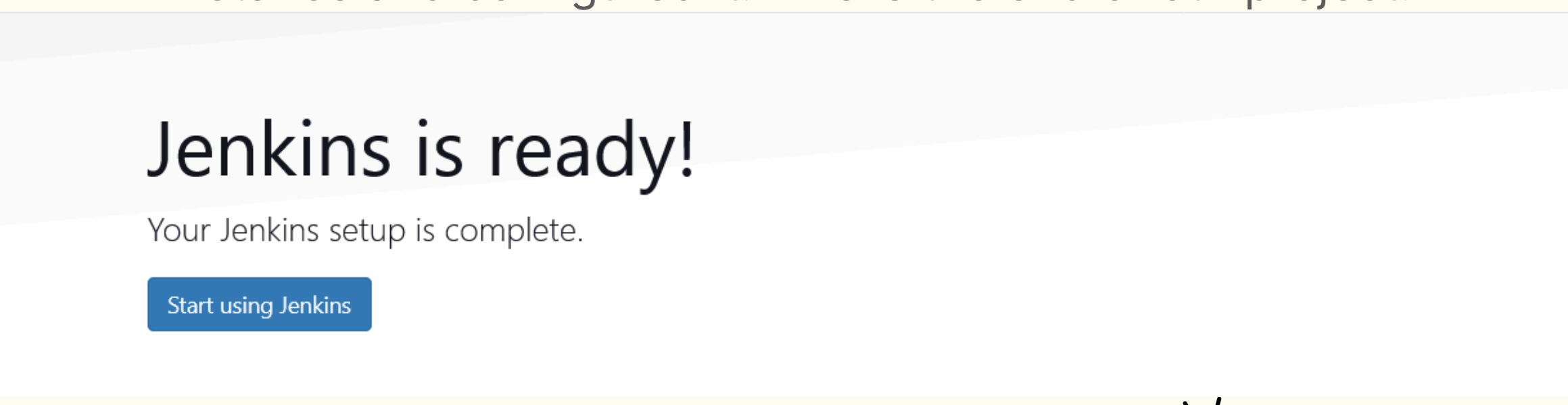
Swipe for more



Step4 : Confirm the instance configration and save it



Step5: Congratulations🎉!! you have successfully installed Jenkins on EC2 instance and configured it. This is the end of our project!



Swipe for more

You did phenomenal job and many congratulations🥳 on the completion of your project🎉.

May be for someone it was their first ever project 🥹 .

I feel proud and you should feel it even took a minor step of opening their AWS console.. yes console after these document. Feel proud of even a single action you are taking towards your goal no matter how small it may seem. Every steps is a vote to your new identity!

Because “The Journey of thousand miles begins with a single step”

Special thanks to my AWS mentor, Abhishek Veeramalla, whose video helped me in curating this document.

Urvish Jaiswal

Help Others