AWS CloudSpace Academy

Class promotion: AWS Cloud & DevOps Engineer 2025

Student: Ebsiy Anslem Ndimongang

Course: Cloud Computing hands on

Teacher: Narcisse Tchuemoe

SETUP YOUR BASE INFRASTRUCTURE AS FOLLOWS:

Create a VPC named “awesome_vpc” with the primary CIDR “10.0.0.0/16”

Create an Internet Gateway “awesome_igw” and attach it to the VPC.

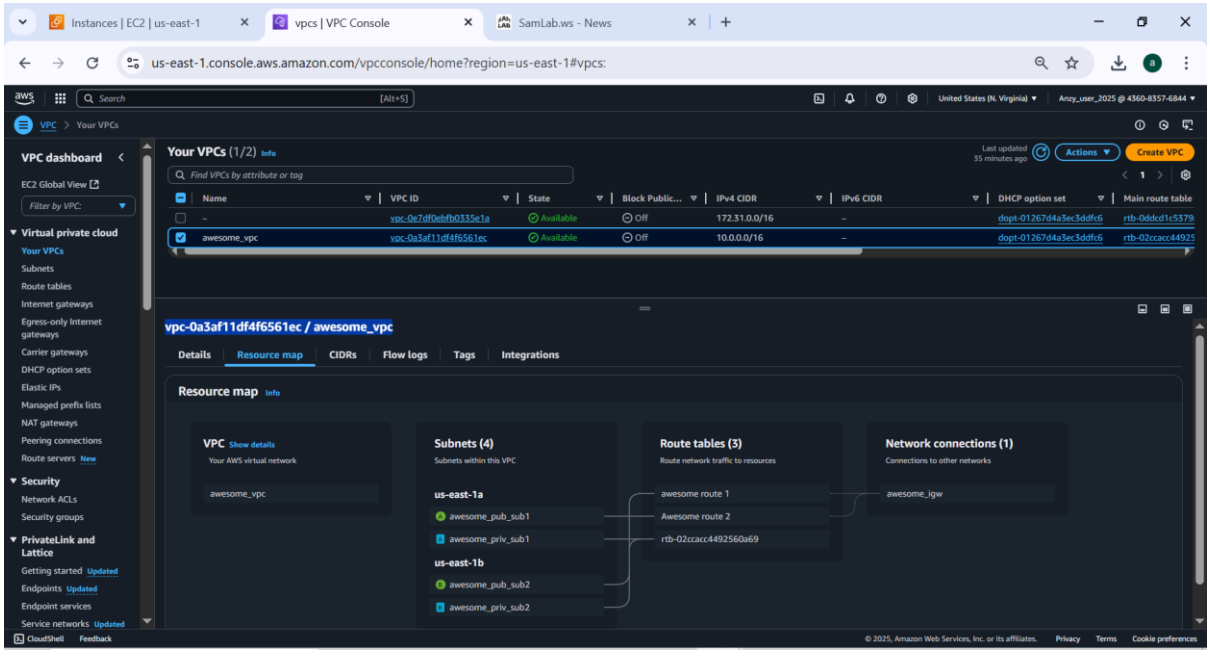
Create a Public subnet “awesome_pub_sub1” – “10.0.1.0/24” in US-EAST-1A

Create a Public subnet “awesome_pub_sub2” – “10.0.2.0/24” in US-EAST-1B

Create a Private subnet “awesome_priv_sub1” – “10.0.3.0/24” in US-EAST-1A

Create a Private subnet “awesome_priv_sub2” – “10.0.4.0/24” in US-EAST-1B

Attach a screenshot of the setup



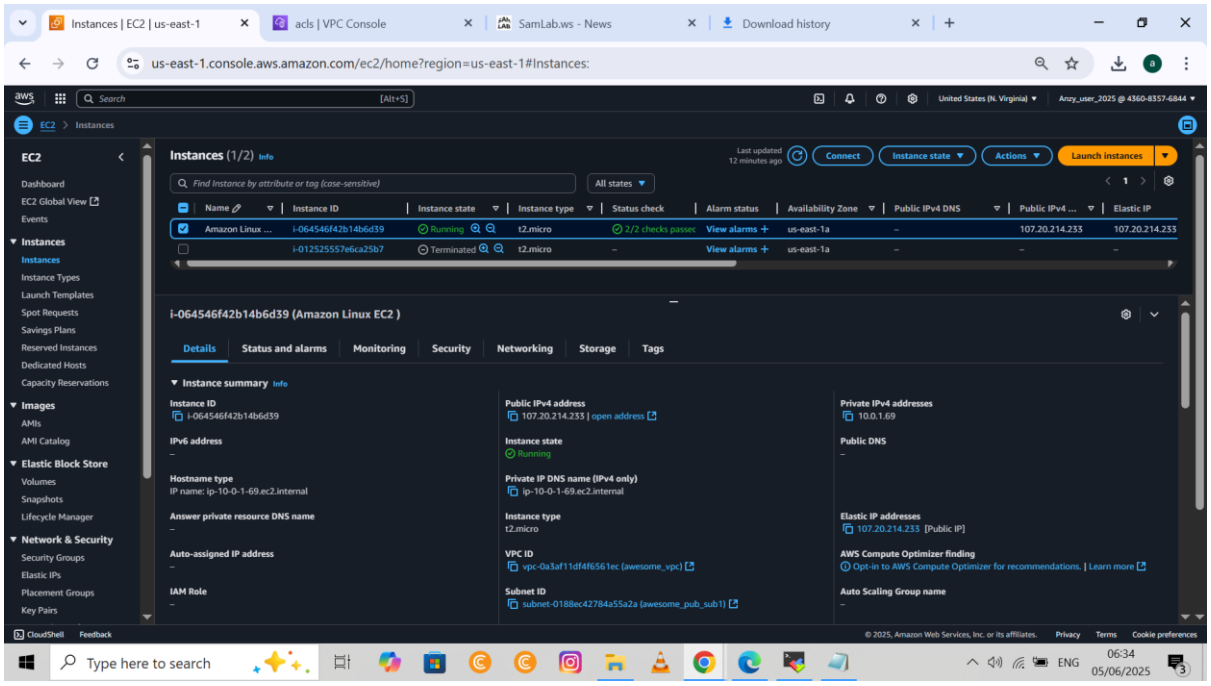
Homework 1

a) Launch an Amazon linux instance in awesome_pub_sub1 subnet as follows:

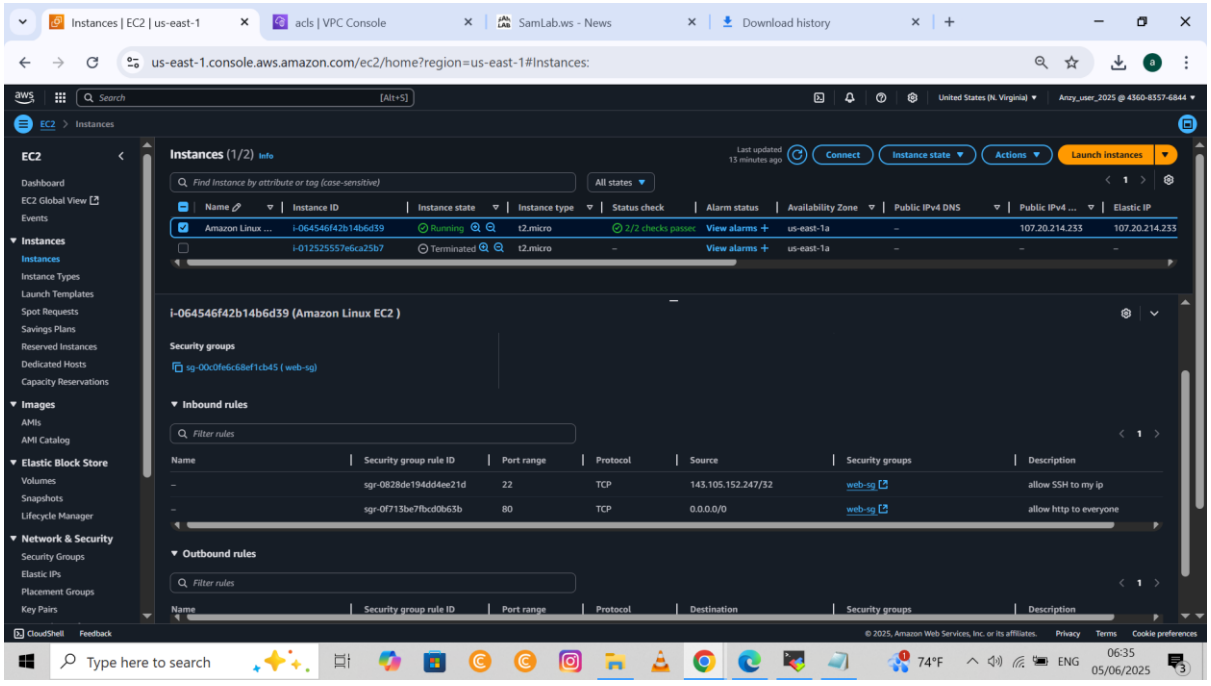
- AMI: Amazon linux 2
- Specs: t2.micro
- Storage: 8GB

- Attach a Security Group call web-sg allowing SSH traffic to your IP address and HTTP traffic to the outside world.
- Tag the instance our-first-ec2
- KeyPair: awesome-key-east1

Attach a screenshot of your EC2 Dashboard with the running instance



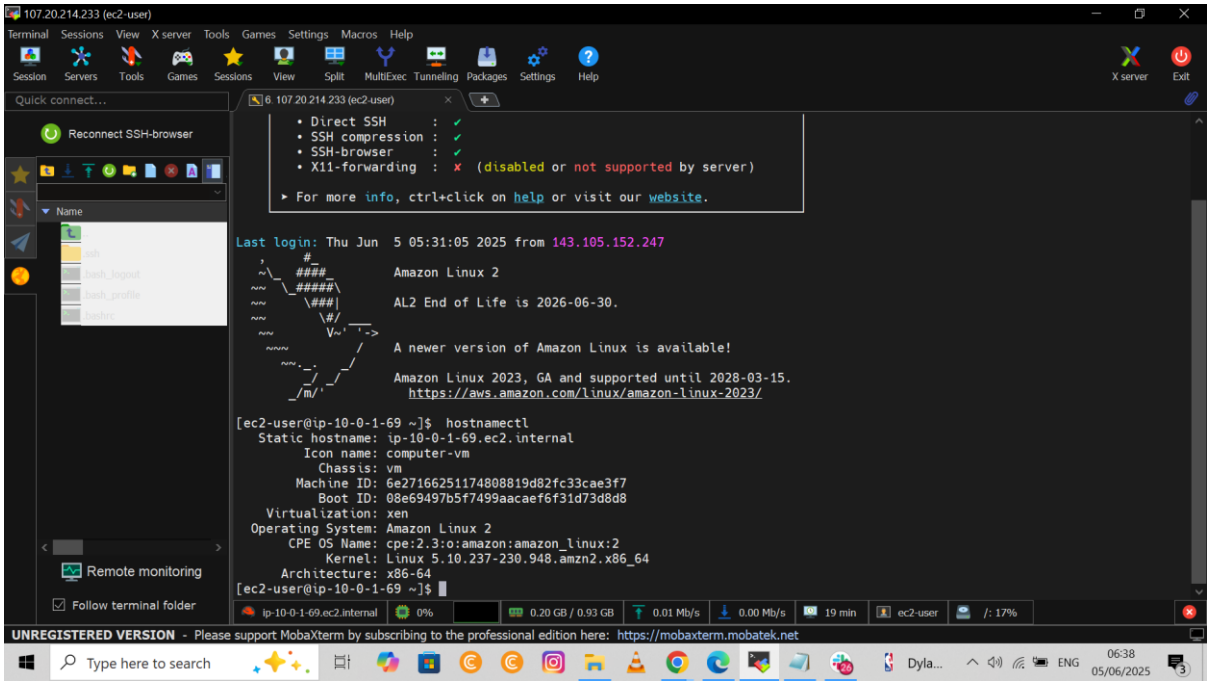
Attach a screenshot of your security group displaying all the rules in the same



Homework 2:

a) Log into the EC2 (our-first-ec2) created in homework1 using SSH,run the command hostnamectl on your ec2 serve

Attach a screenshot of the output of the hostnamectl command

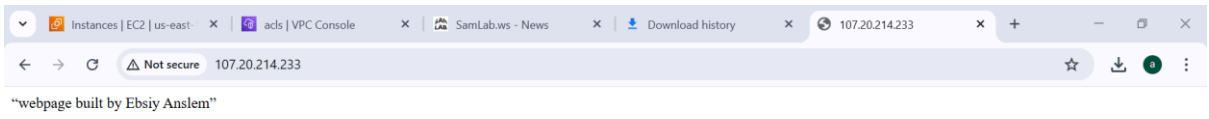


HOMEWORK 3:

a)Launch an EC2 Linux instance and connect to it via SSH.

Manually deploy Apache HTTP server on a Linux EC2 instance using the script below.

Display the message “webpage built by Ebsiy Anslem” on a webpage.



HOMEWORK 4:

a) – Work with a cluster member

Create an AMI based on our previous EC2 Apache HTTP Server.

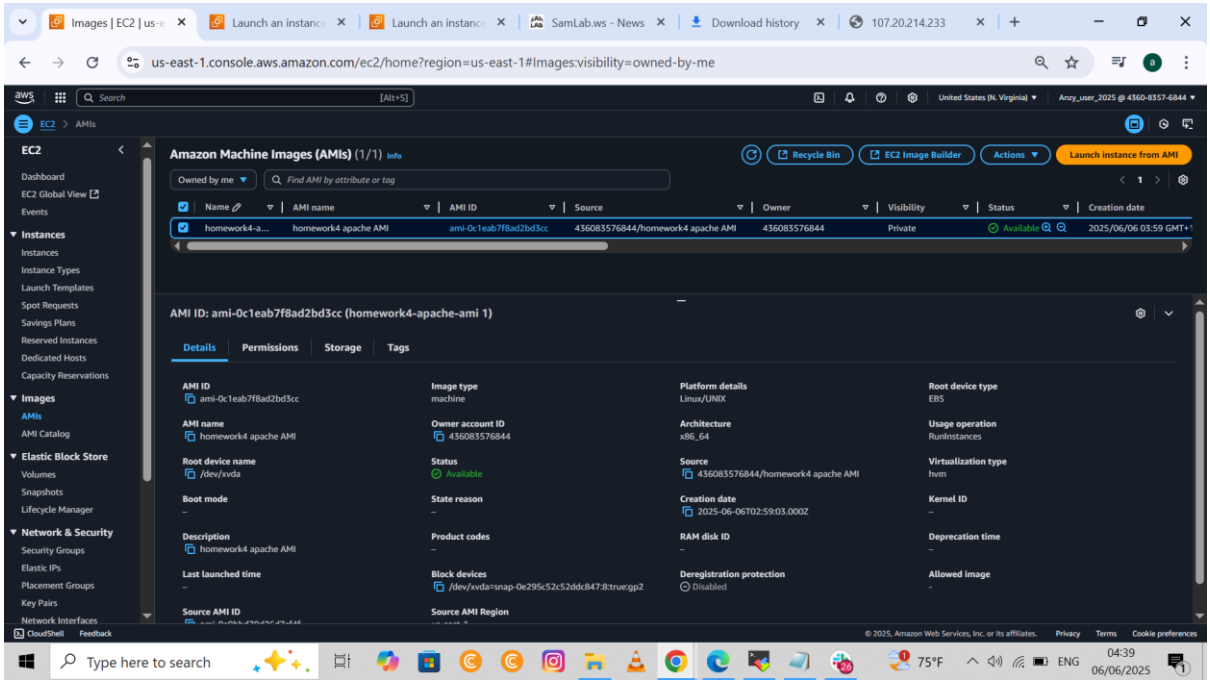
Tag AMI with value “homework4-apache-ami”.

Share the AMI with one member of your cluster.

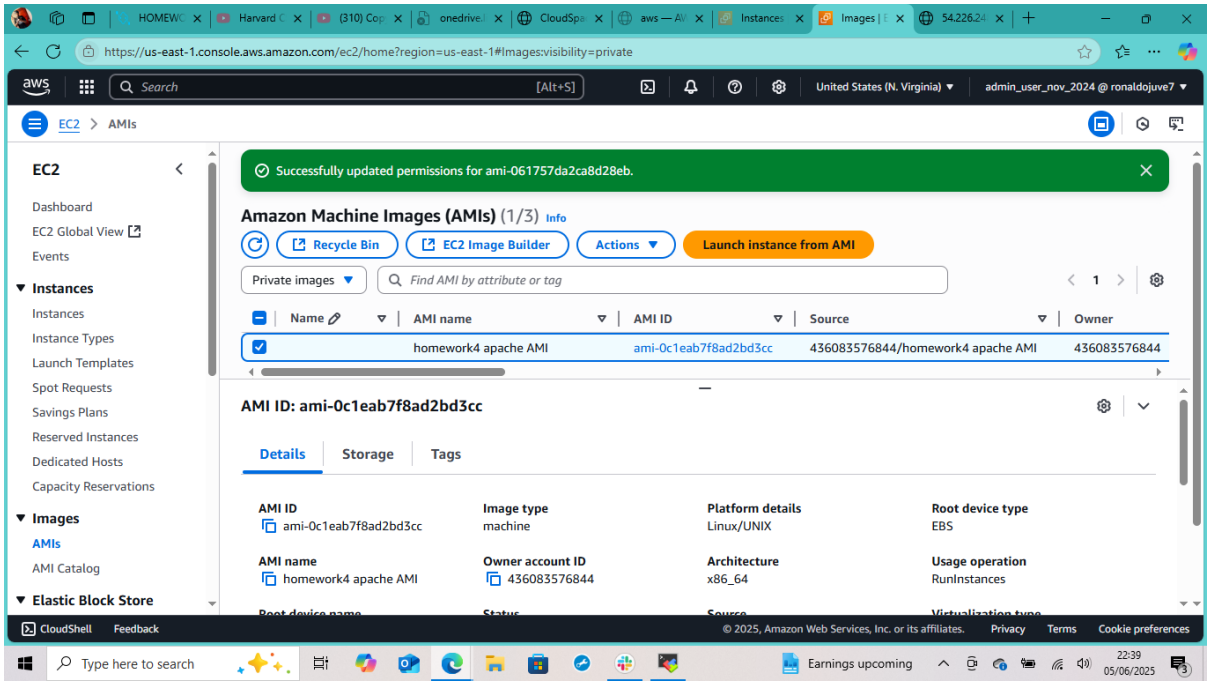
Launch an EC2 instance using the AMI created in the previous step.

SUBMISSION: Post a screenshot of:

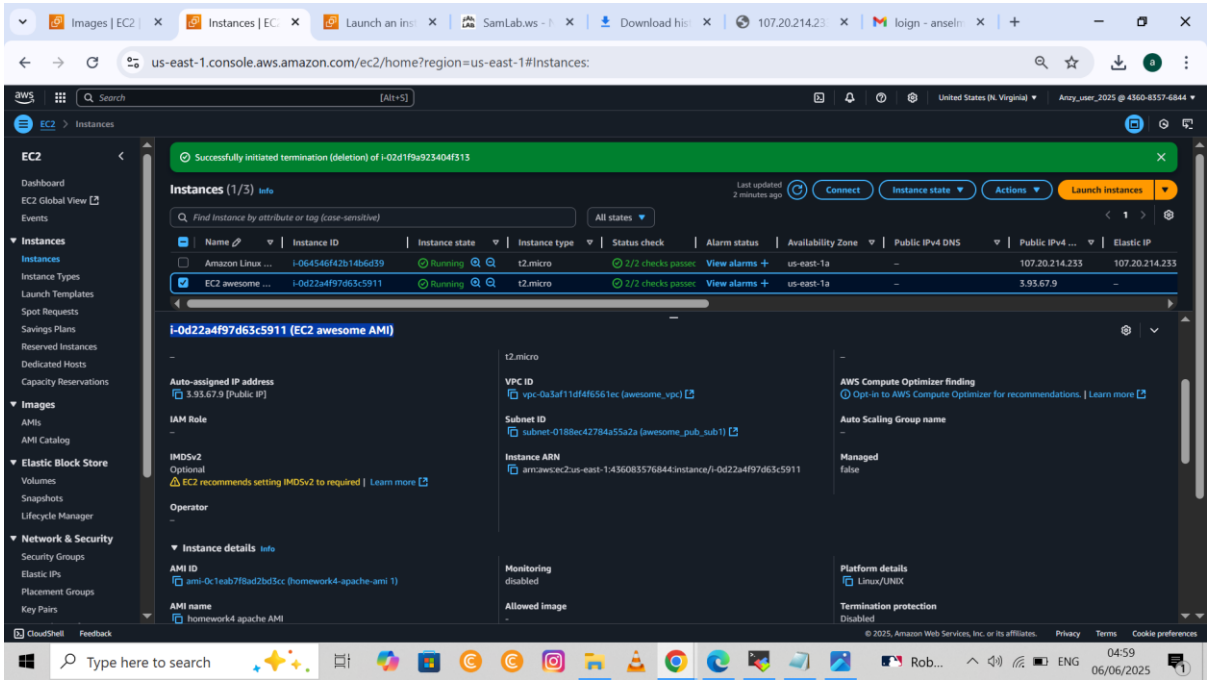
AMI in my account



AMI in the destination account



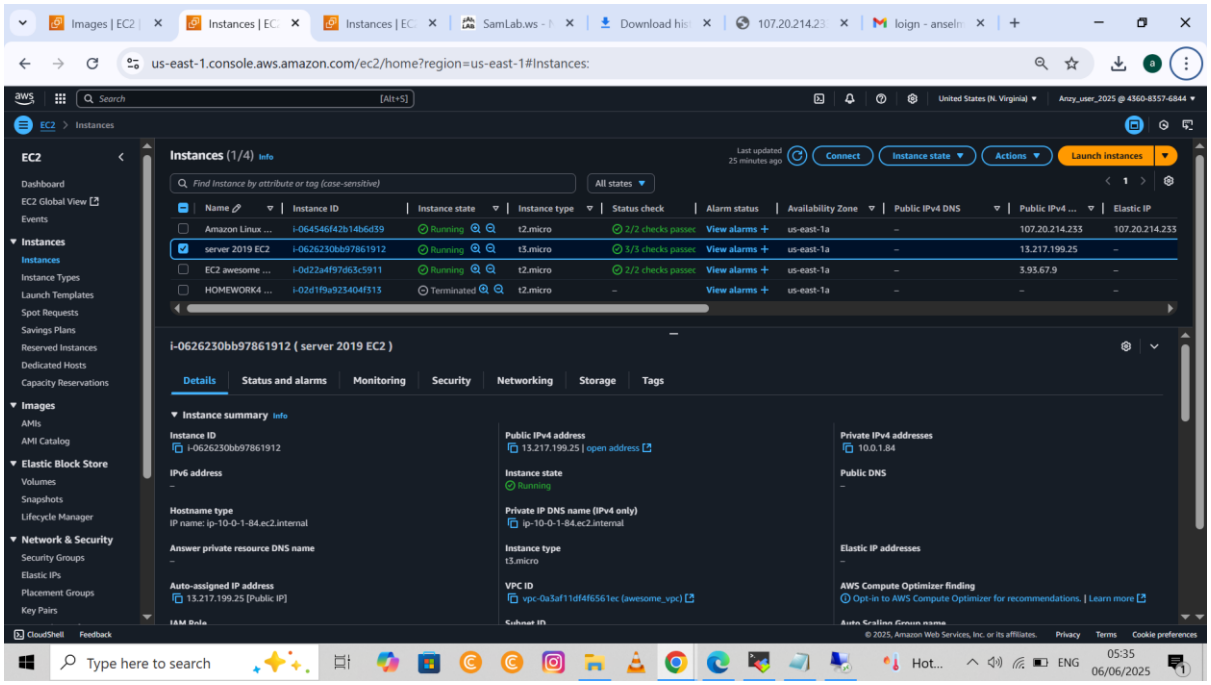
Screenshot of the **peer launch a new EC2 instance using the shared AMI**



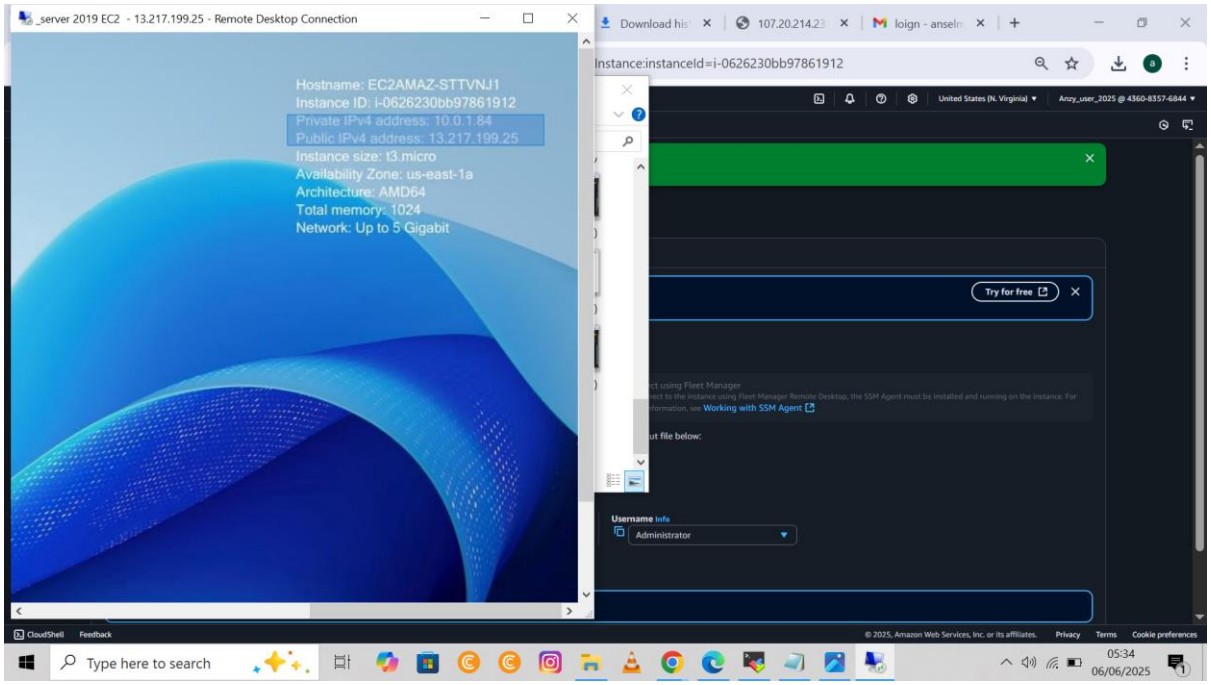
HOMEWORK 5:

– Launch a windows server 2019 EC2 instance and connect to it using a RDP client of your choice.

SUBMISSION: Post a screenshot of the Windows Server landing page after successful authentication. Attach your screenshot to a Word or pdf document.

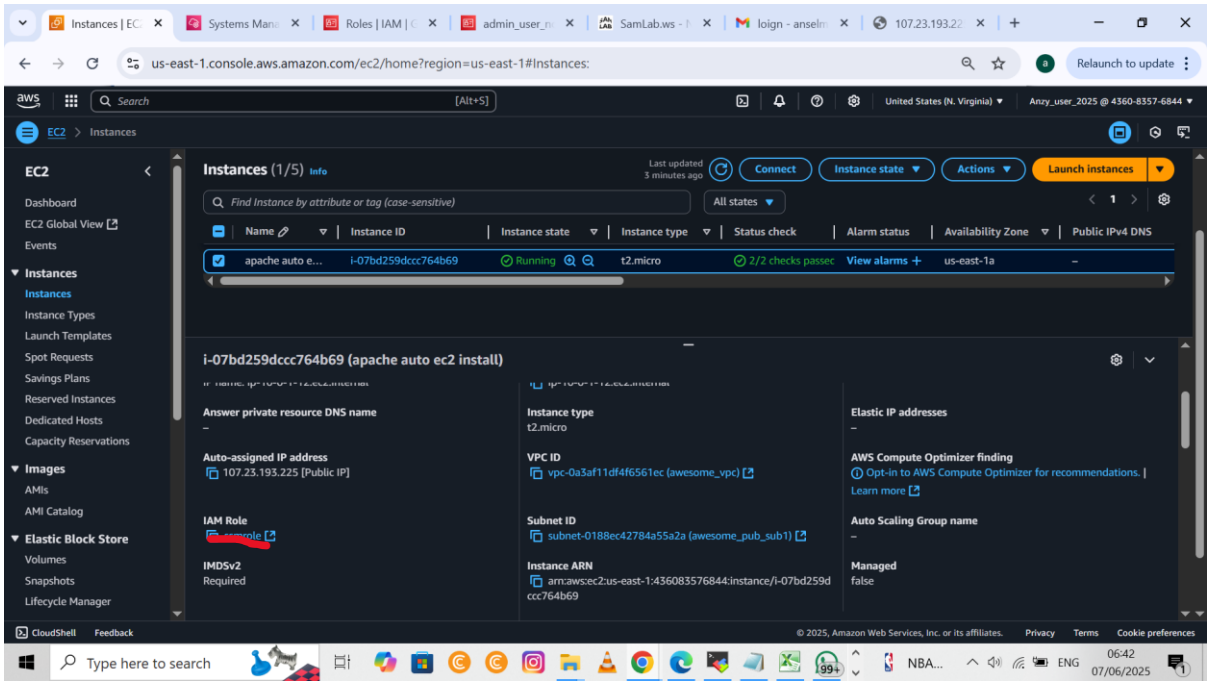


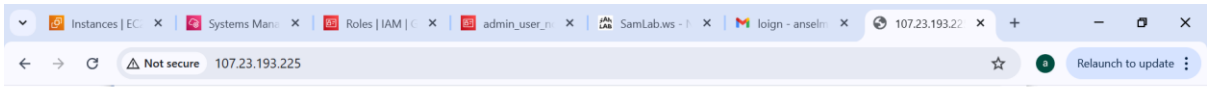
Sreenshot of the Windows Server landing page



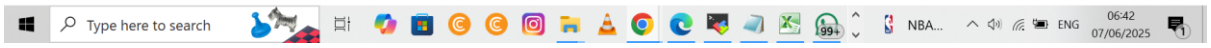
HOMEWORK 6: USER DATA

a)Automate your HTTP server creation using EC2 user-data.
Make sure to attach an SSM role at launch that will allow you to login into the instance within your browser.



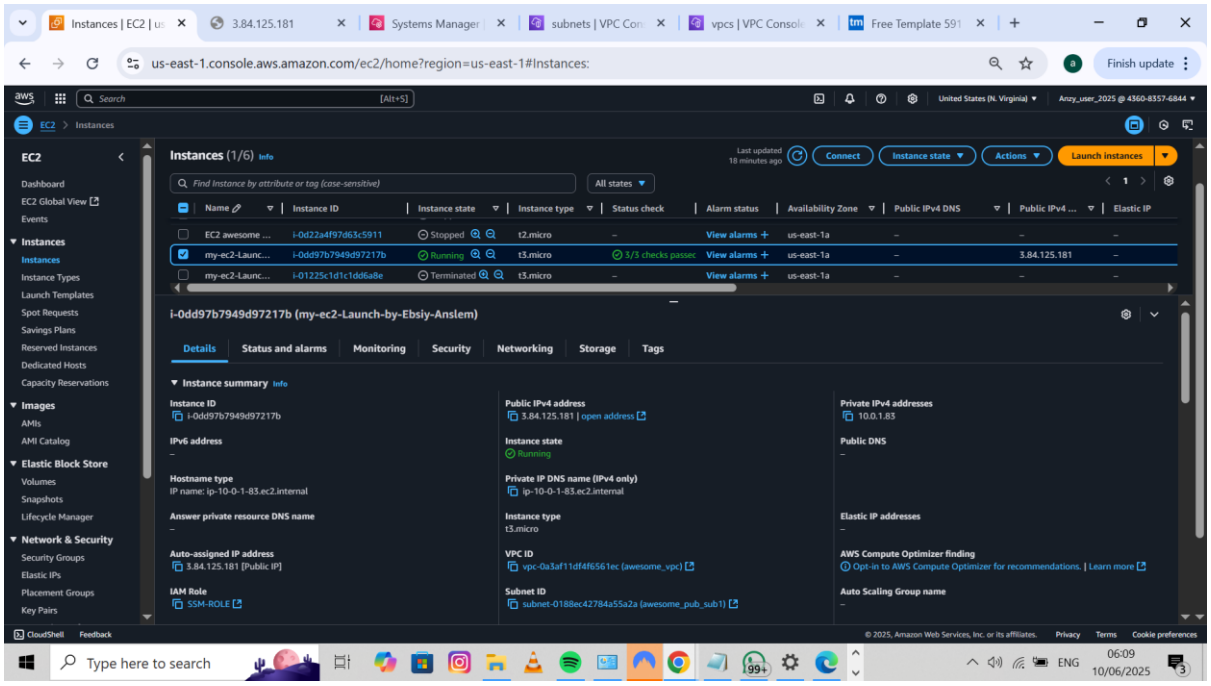


"Automated Apache web server deployment using EC2 UserData – Ebsiy Anselm"

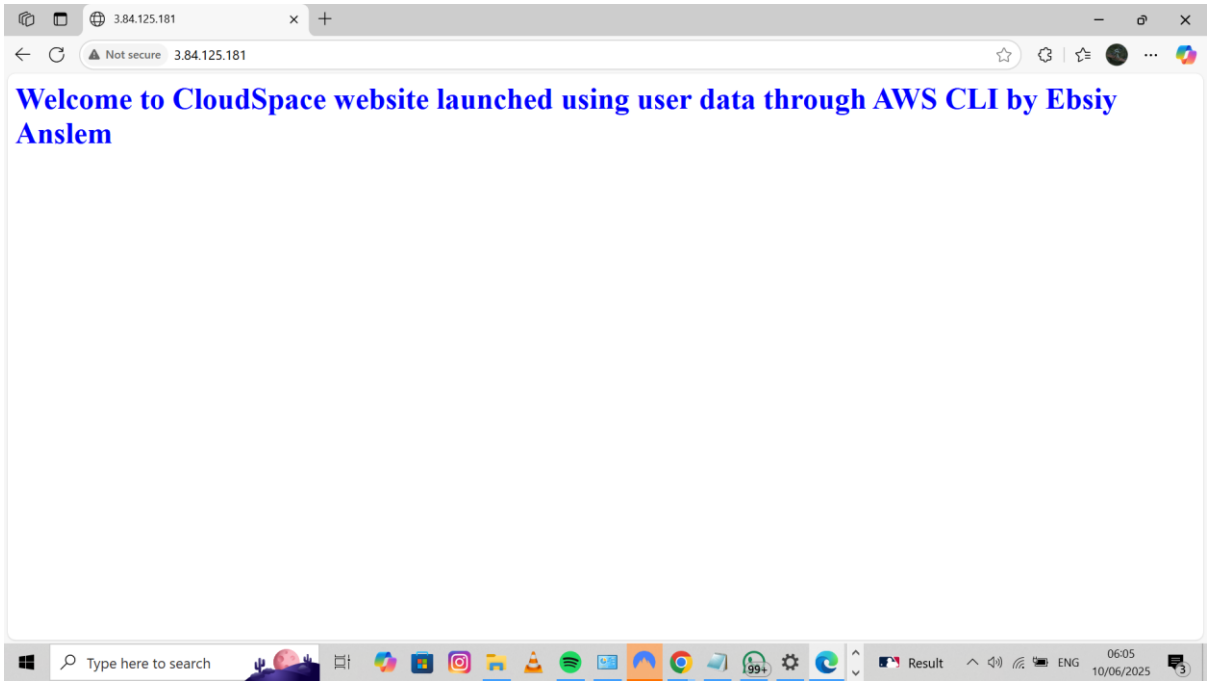


HOMEWORK 7: AWS CLI

a)Launch a linux EC2 instance using AWS CLI and Tag it with "my-ec2-created-through-CLI". Be carefull with **N**ame tag

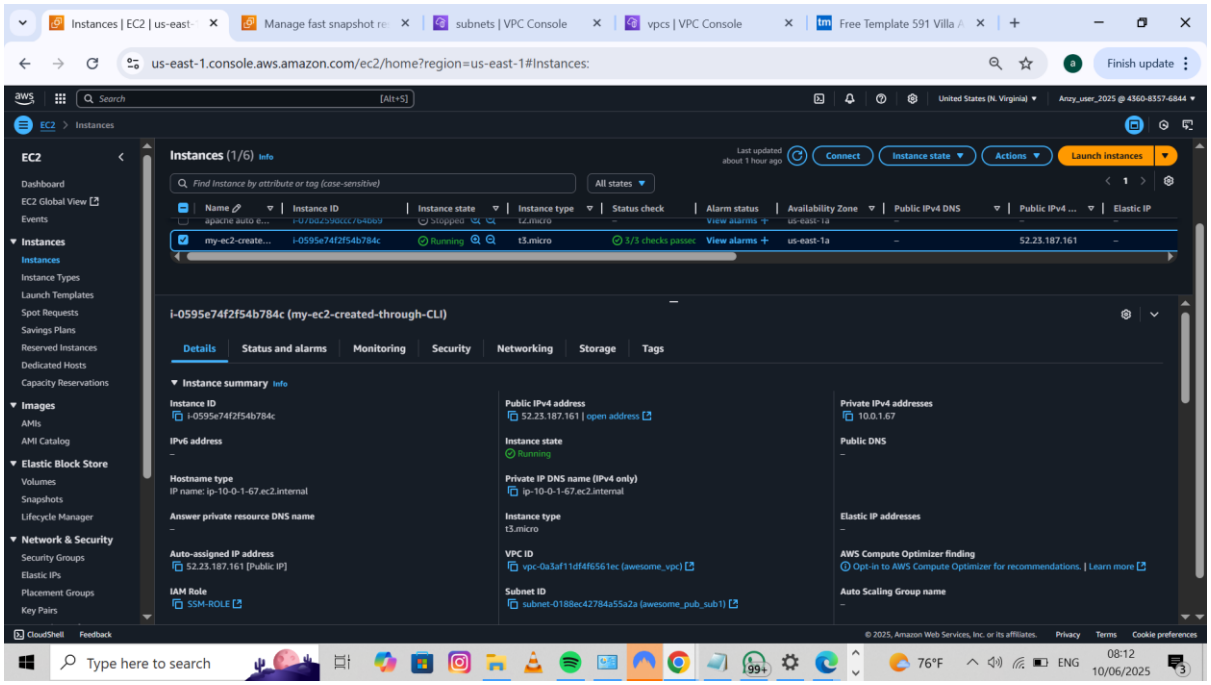


b) Using AWS CLI, Automate your HTTP server creation using EC2 user-data.
User-data Script:

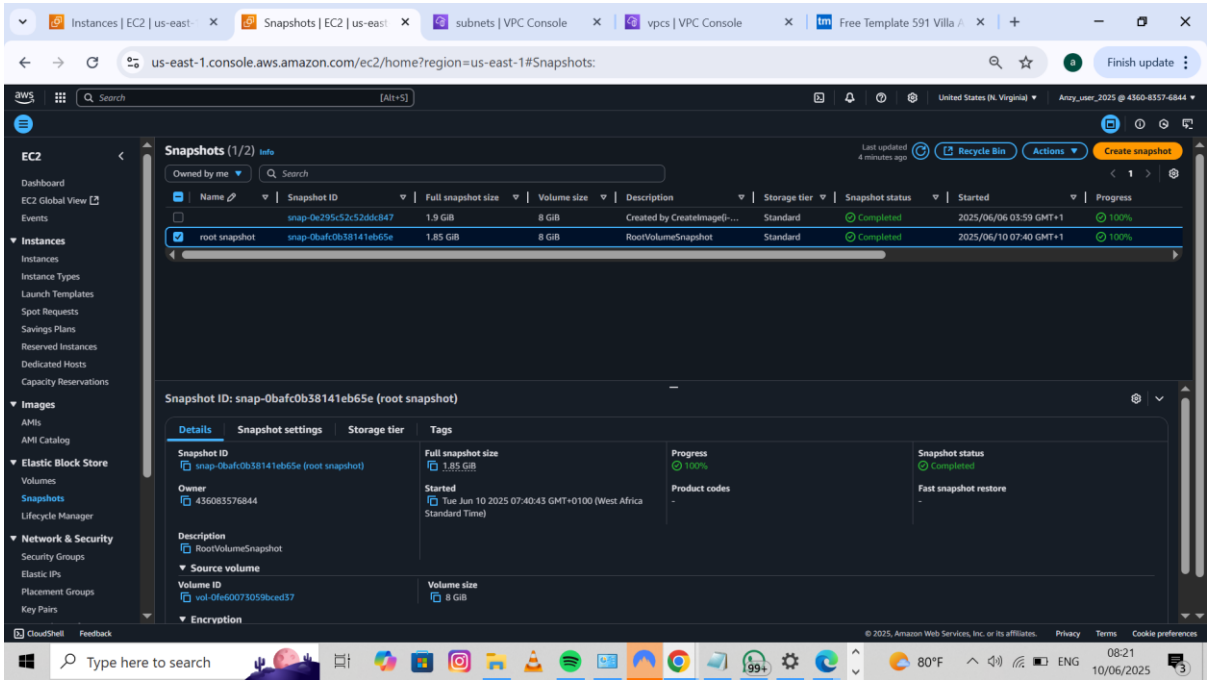


HOMEWORK 8: EBS & SNAPSHOT

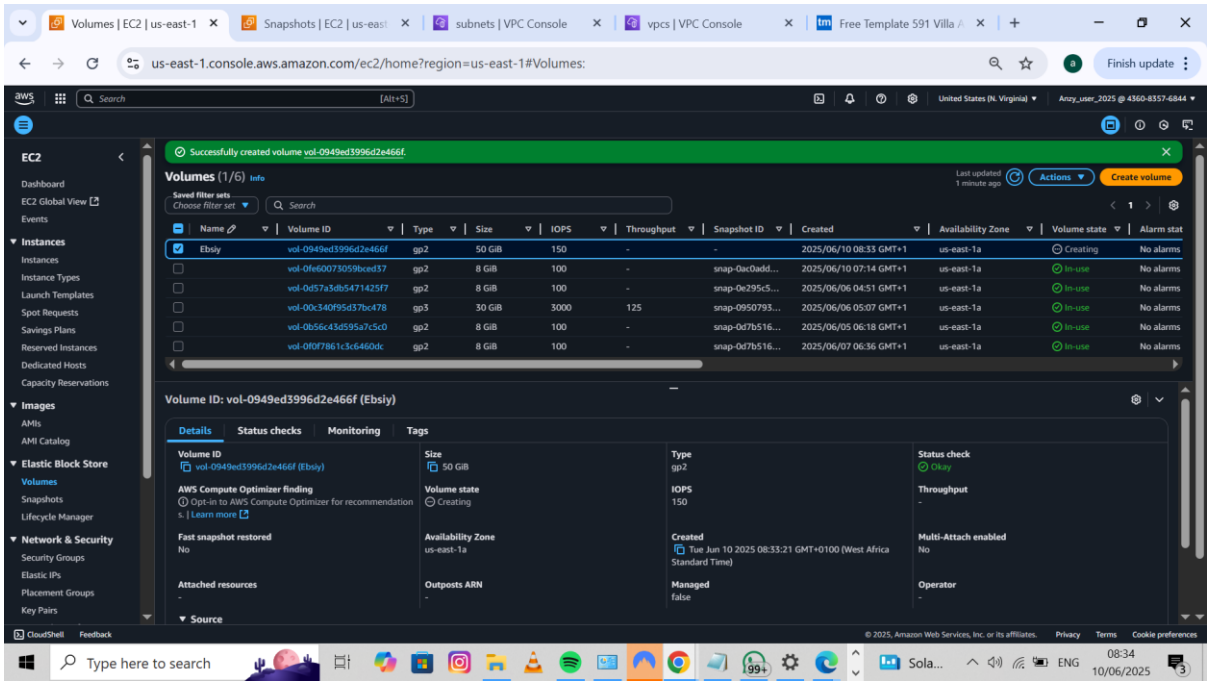
a)Launch a linux EC2 instance using AWS CLI and Tag it with “my-ec2-created-through-CLI”. Be carefull with **N**ame tag



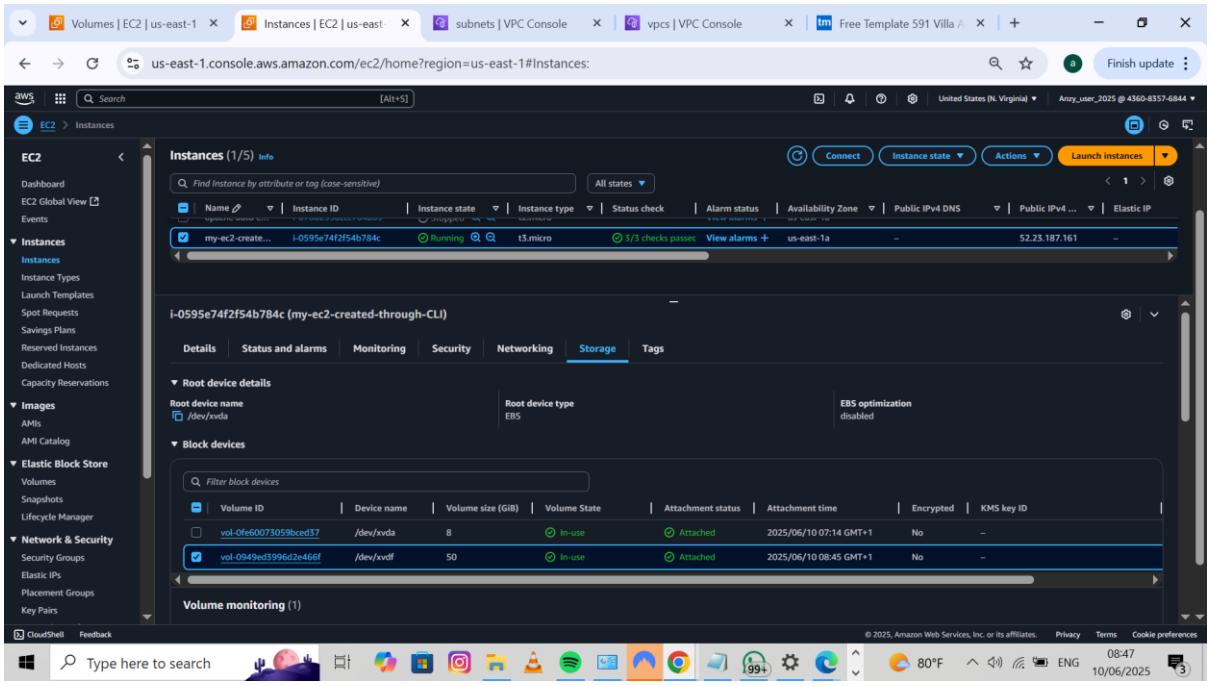
b)Take a snapshot of the Root Volume



c)Create a new Volume (with **Name** tag "**your_first_name**", size 50 GB) in the same Availability Zone (AZ) as the EC2 instance. Take screenshot



d)Attach the new volume to the EC2 created in step 1. Take a screenshot showing both Volumes in the console.



e) login to the EC2 instance and run the command “**lsblk**”. Take a screenshot showing both Volumes size

Exemble of **lsblk** output

[ec2-user@ip-10-0-7-44 ~]$ **lsblk**

NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT

xvda 202:0 0 8G 0 disk

└─xvda1 202:1 0 **8G** 0 part /

xvdf 202:80 0 **50G** 0 disk