**Tasks Assigned:**

**Assignment-1 : Terraform\_AWS VPC Setup**

**Assignment-2 : Ansible- AWS EC2-HTTPS Setup**

**Github Source Code Location:**

<https://github.com/mohanamuri/hantt-terraform-aws-vpc>

<https://github.com/mohanamuri/hantt-ansible-ec2-https/>

<https://github.com/mohanamuri/hantt-ansible-ec2-https-windows>

**Terraform\_AWS VPC Setup - Assignment Results**

**Targeted Setup/ Result:** Create a separate VPC setup with other relevant components, try to install apache2 server in EC2 instance which is part of the public subnet and access the apache server.

**Setup VPC and Other Components:**

**Steps- Followed:**

* Create VPC (Provide CIDR, i.e IP address range)
* Create Internet Gateway && Attach Internet Gateway to VPC
* Create Subnet/s - 1- Private and 1- Public
* Create NAT Gateway for private Subnet (just created but not used)
* Create Route Tables - 1- public and 1-private
* Create Subnet Association
* Create EC2 Instance in Public Subnet with Apache installation script

**Instructions/Suggestions:**

* Configure AWS Credentials (Accesskey, Secretkey) and Public key (to Associate with EC2 keypair) in terraform.tfvar file.
* Also, check other configure other settings line region, subnets etc in the same terraform.tfvar file.
* Used ssh-keygen to generate key-pair for EC2 instances.

ssh-keygen -t rsa -b 2048

* Few Commands Used:

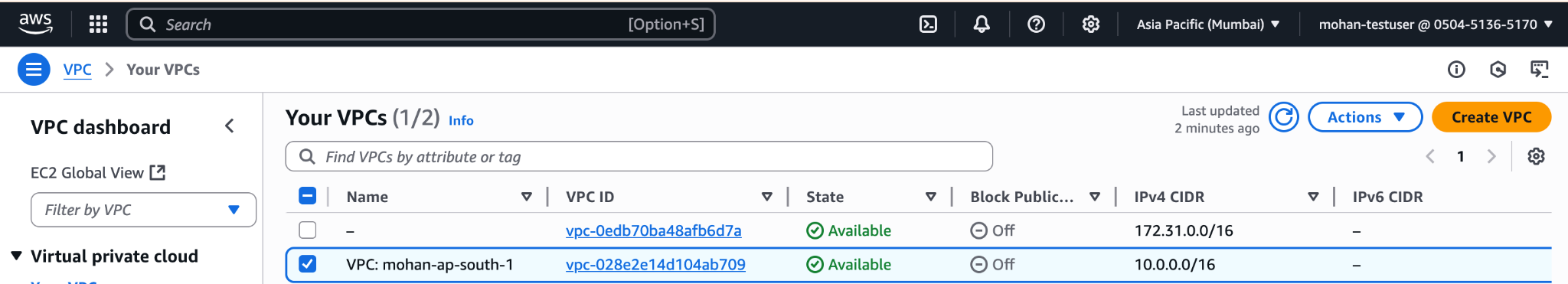
Terraform init

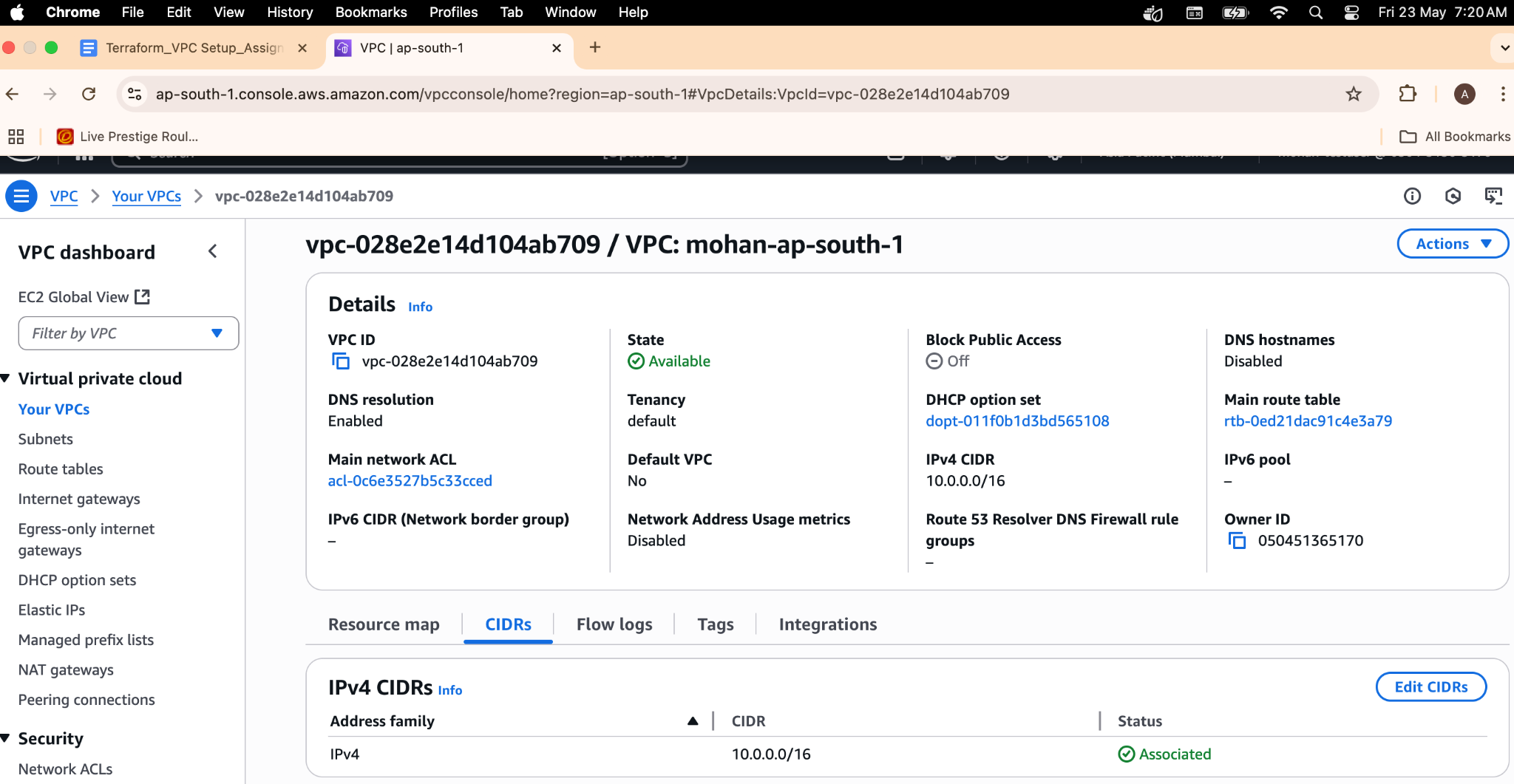
Terraform plan

Terraform apply

**Assignment-1 Results - ScreenShots:**

**VPC**

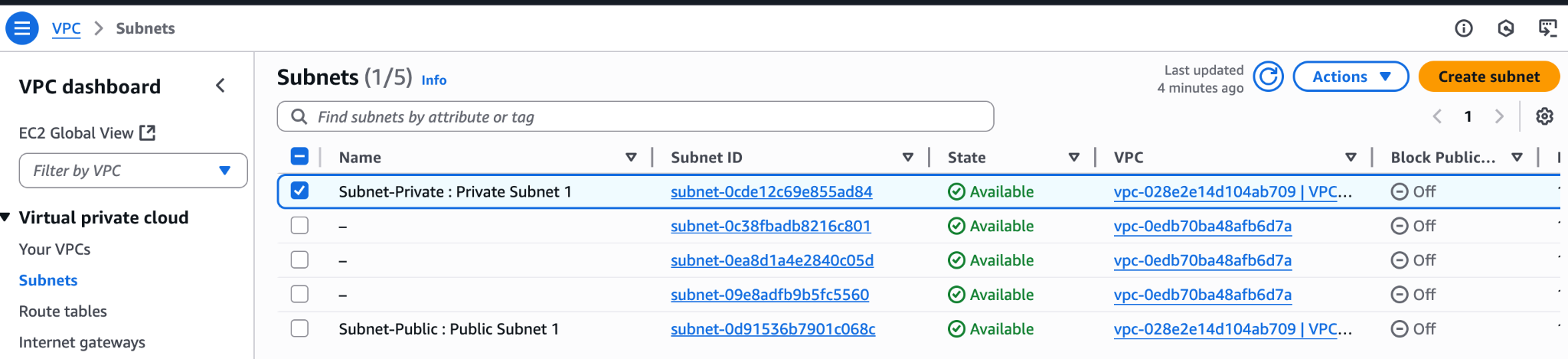




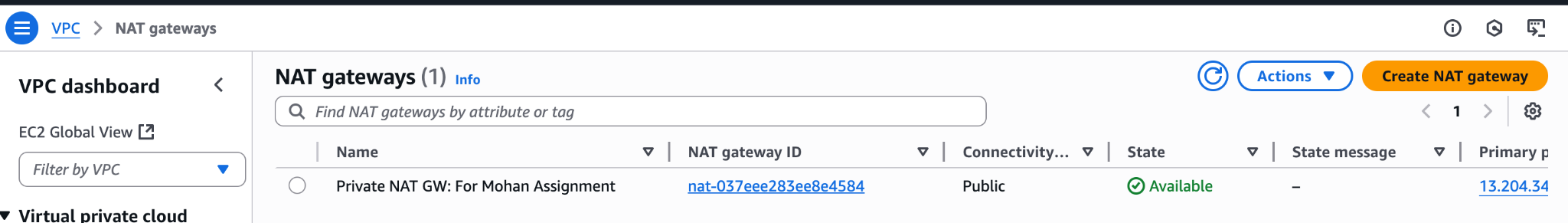
**Internet Gateways:**



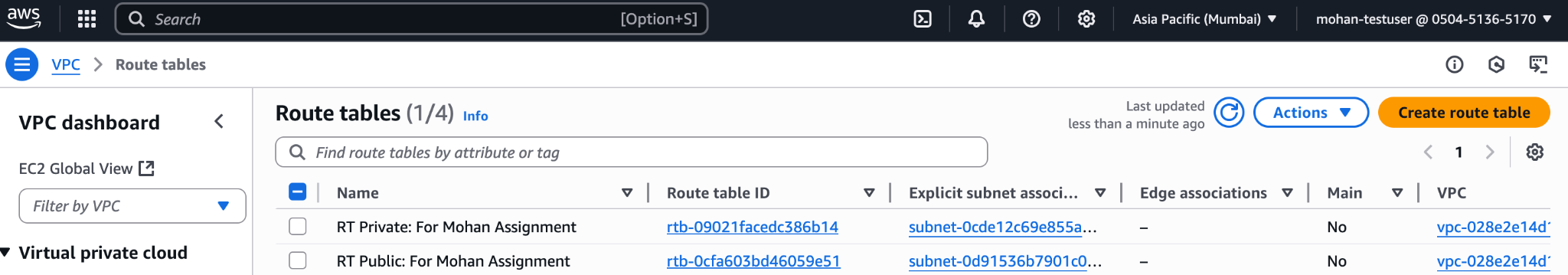
Subnets:



**NAT Gateways:**

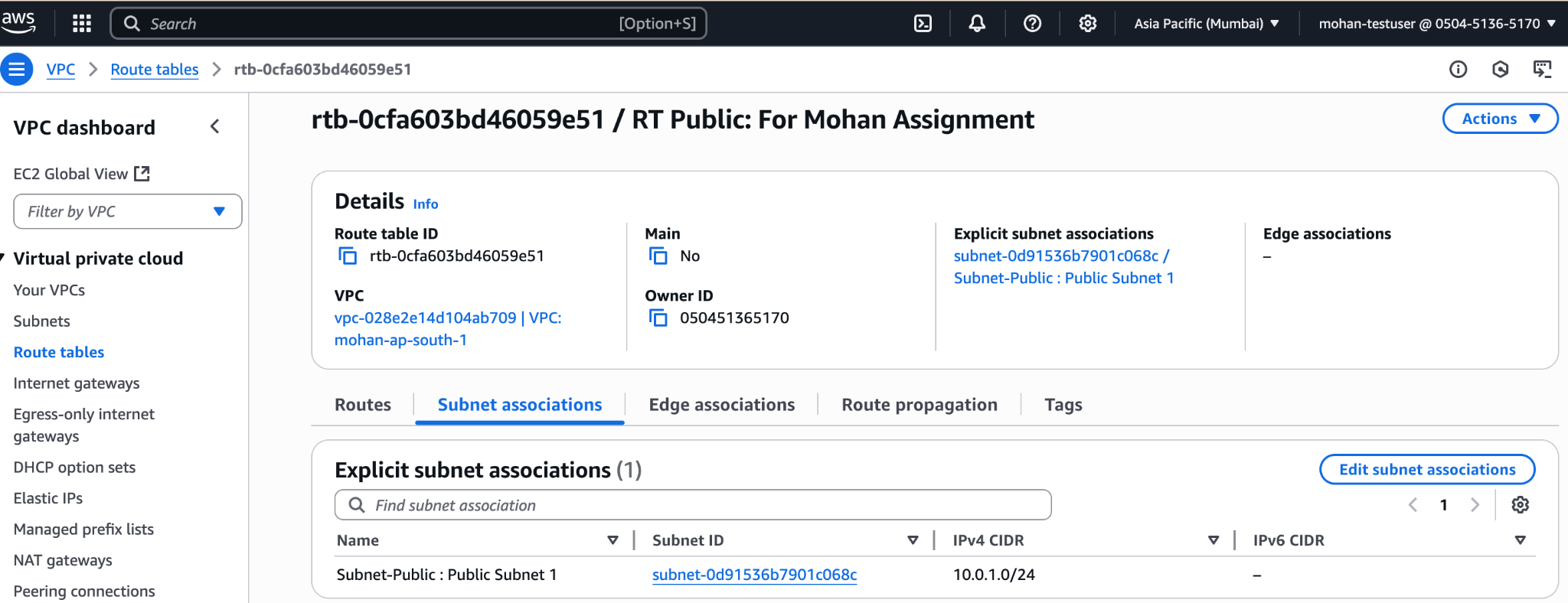


**Route Tables:**

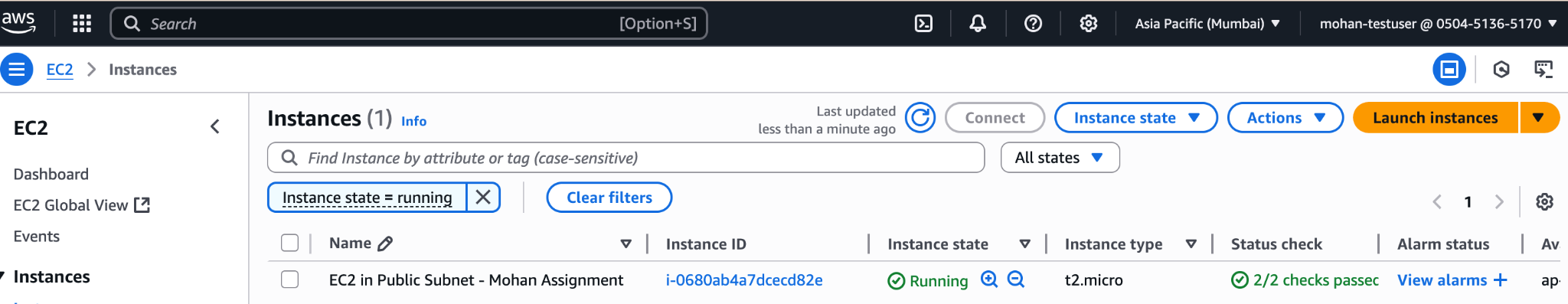


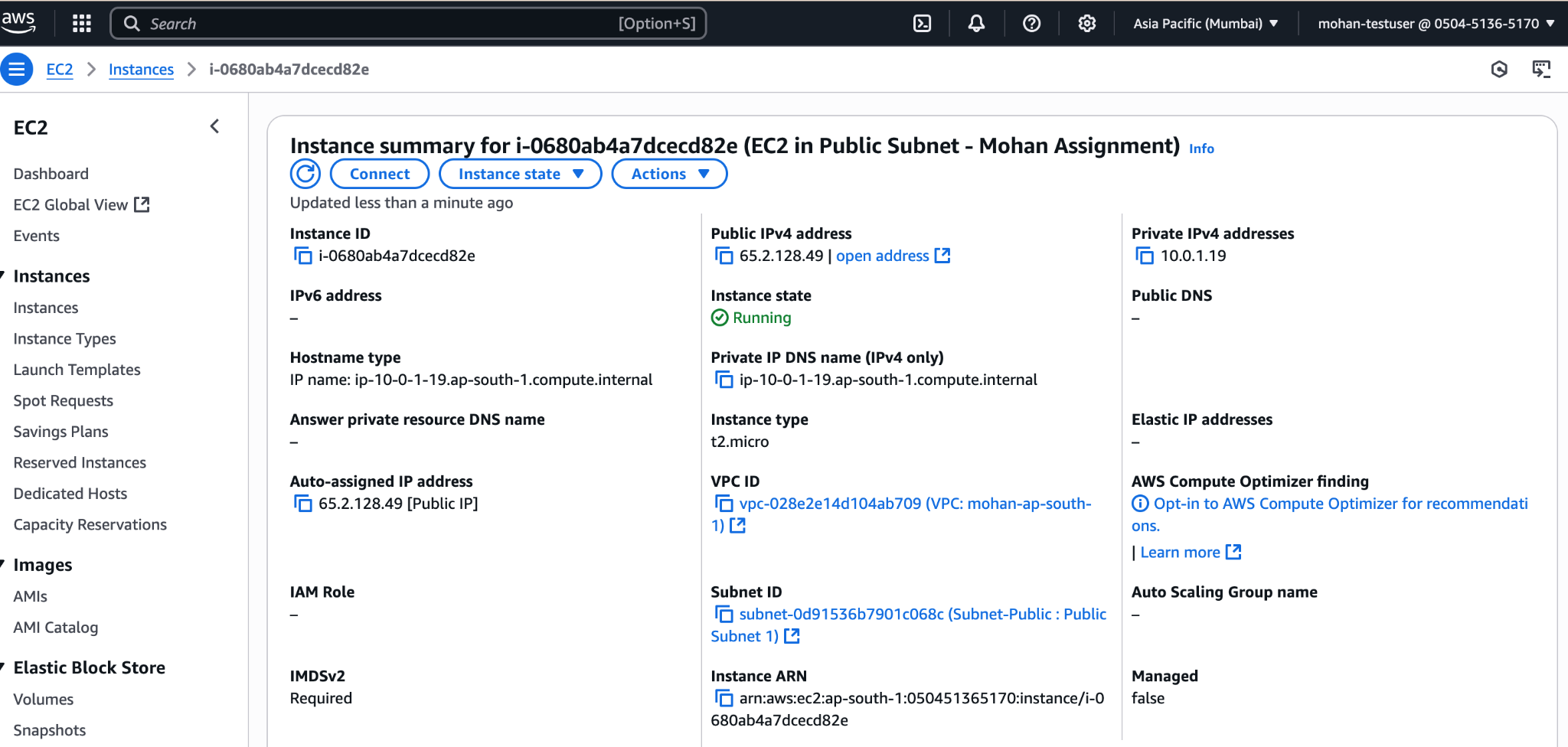
**Subnet-Association:**



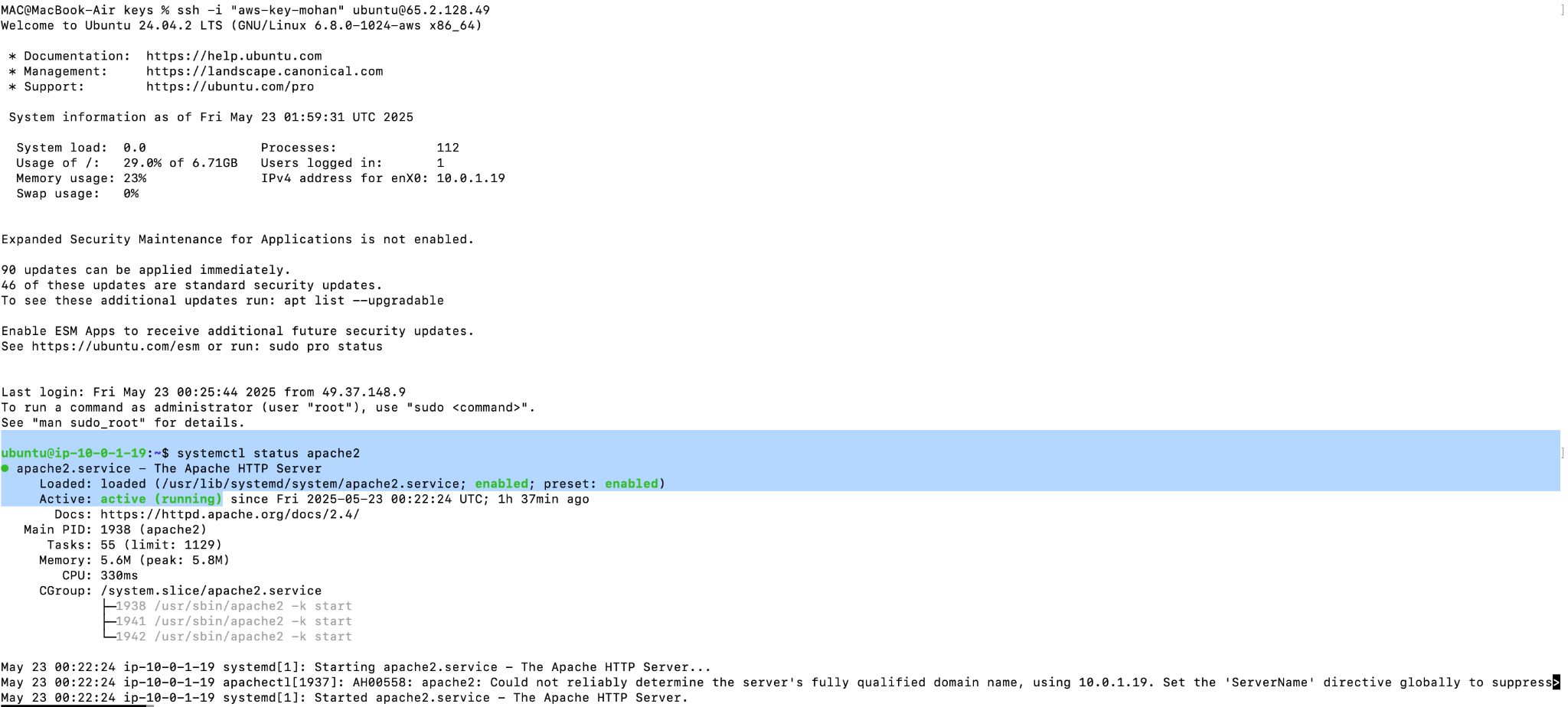


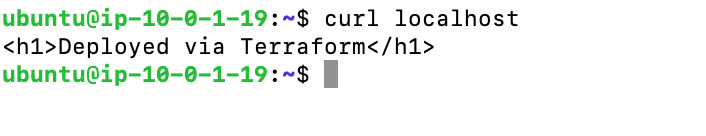
**EC2 - Instance (Public Subnet):**





**SSH EC2 Instance and Verify Apache Server**





**Observation:**

* Successfully connected to the Apache Server from public Subnet EC2. All the resources are provisioned through Terraform Configurations

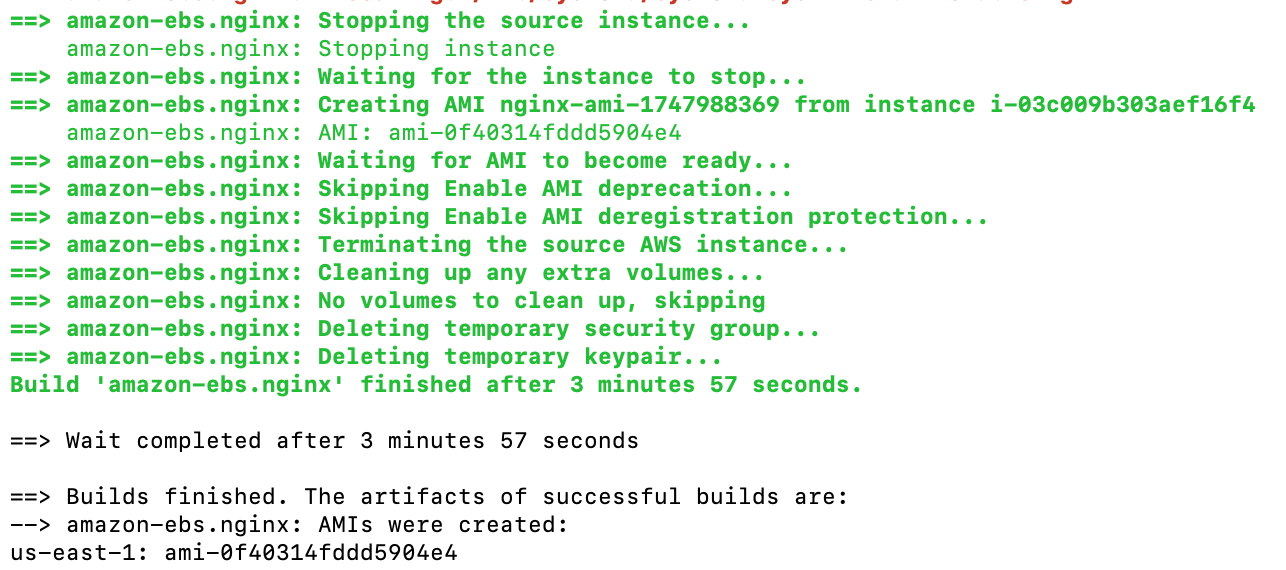
**Task-2: Ansible - AWS EC2 Assignment Results**

**Pre-requiteses**

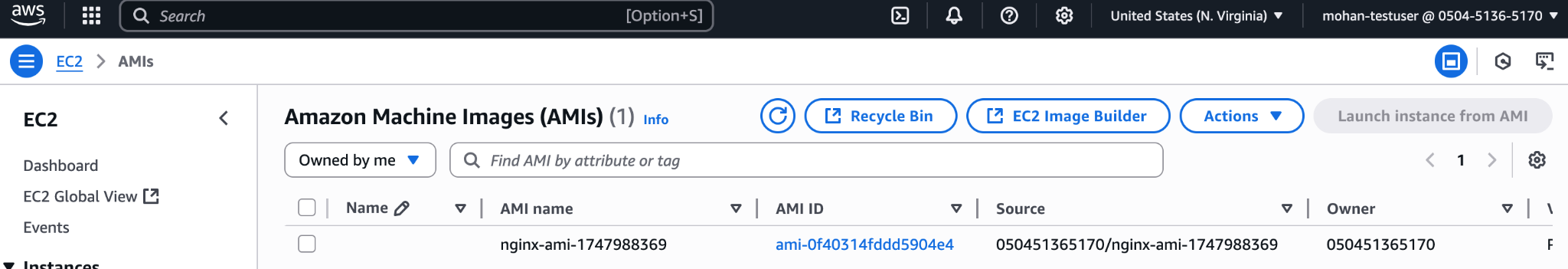
* Installed Ansible, AWS CLI, packer and SSL.
* Generated Self-Signed Certificates.
* Few Packer Commands Used while generating AMI-ID
* p**acker init .**
* **packer validate nginx-https.pkr.hcl**
* **packer build nginx-https.pkr.hcl**

**Packer Build Result**

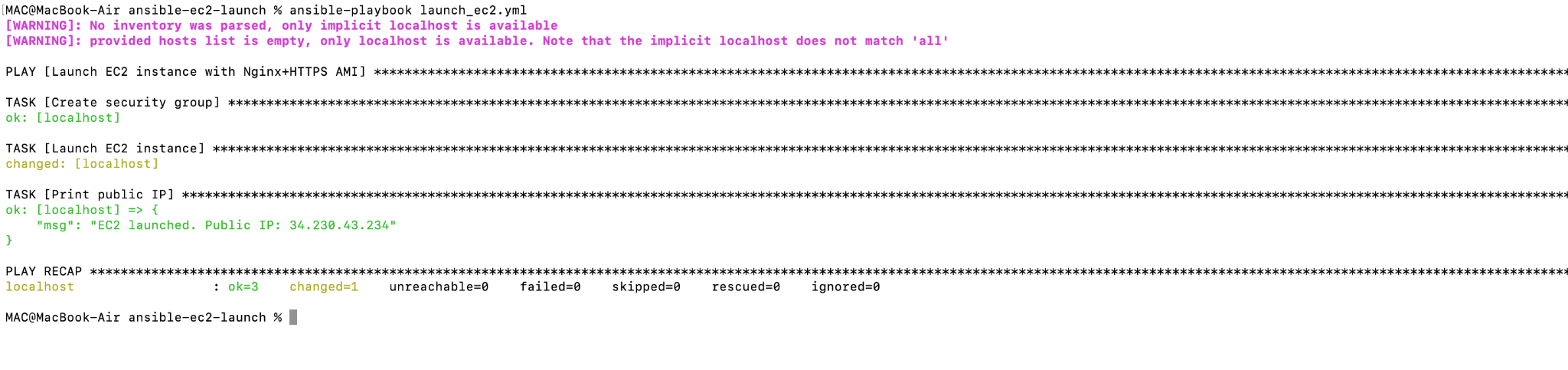
**Observation:** Packer build - Success - AMI Created

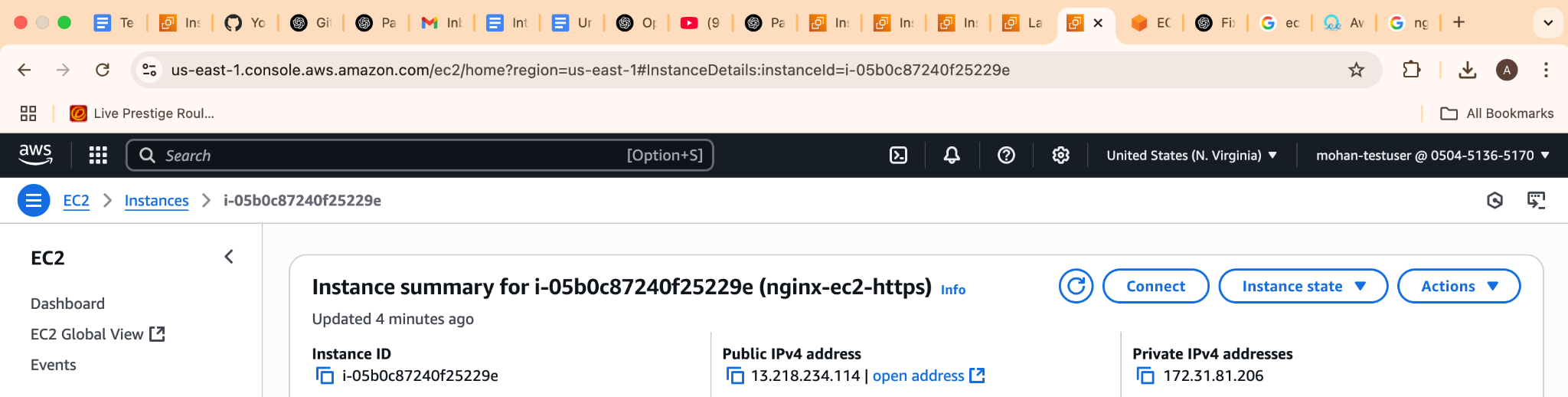


**Screenshot:**

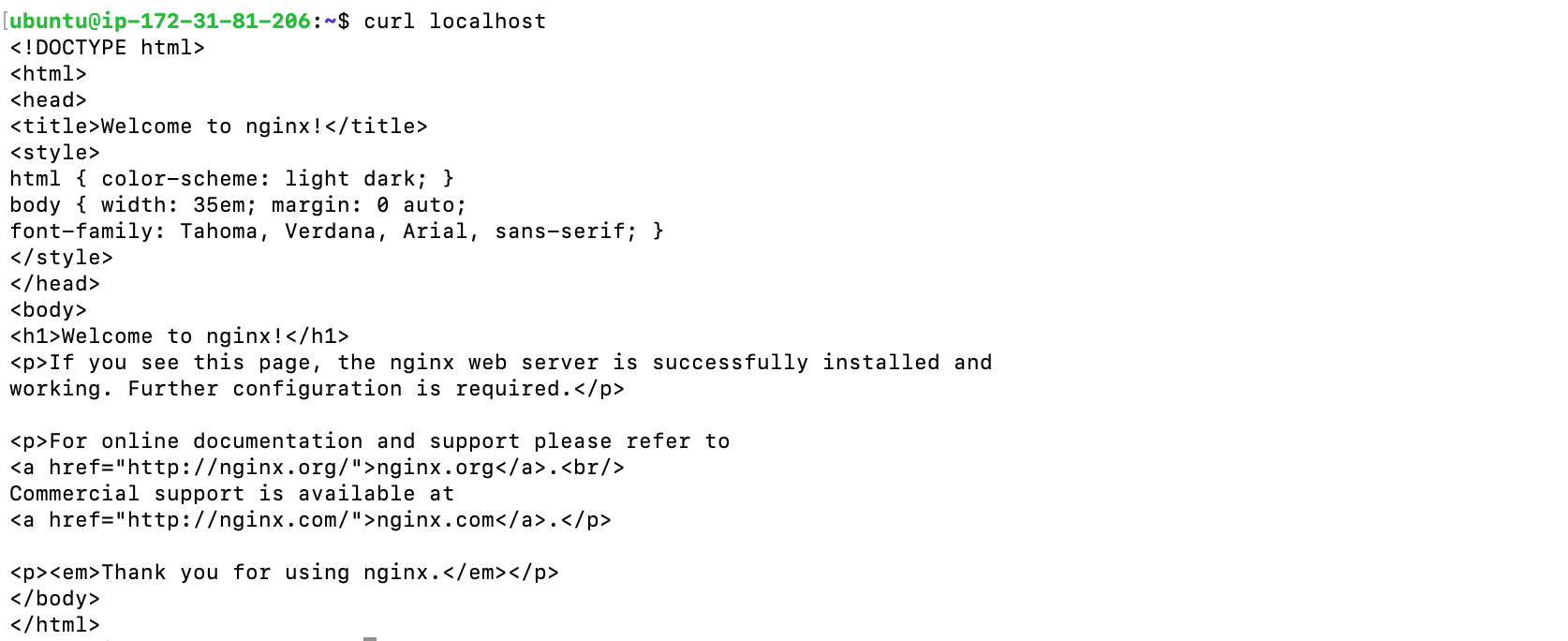


**EC2- Launch -Successful**





**Accessing Nginix Server:**



**Observation:**

**Successfully able to Access Nginx Server**

**Windows-Setup and Results:**

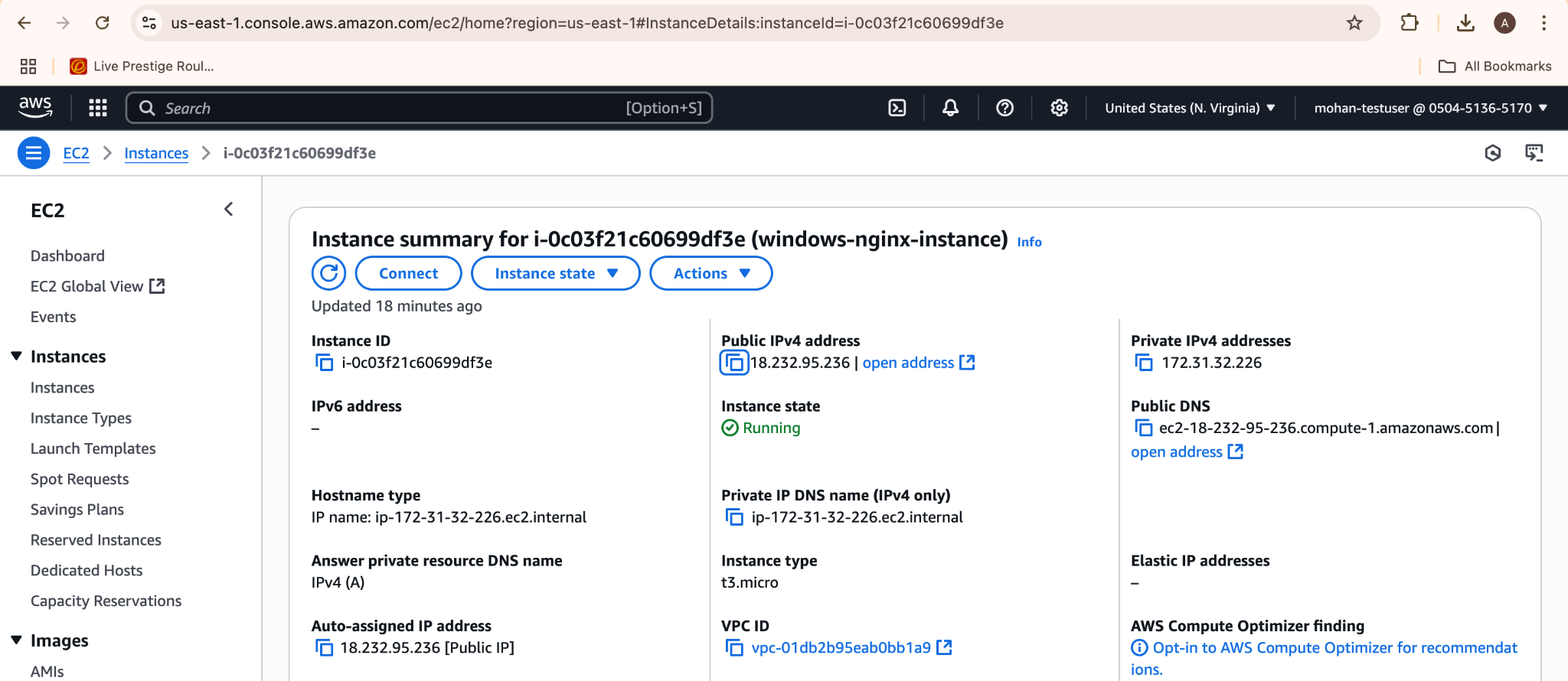
**Command Used to get the Self signed Certificates**

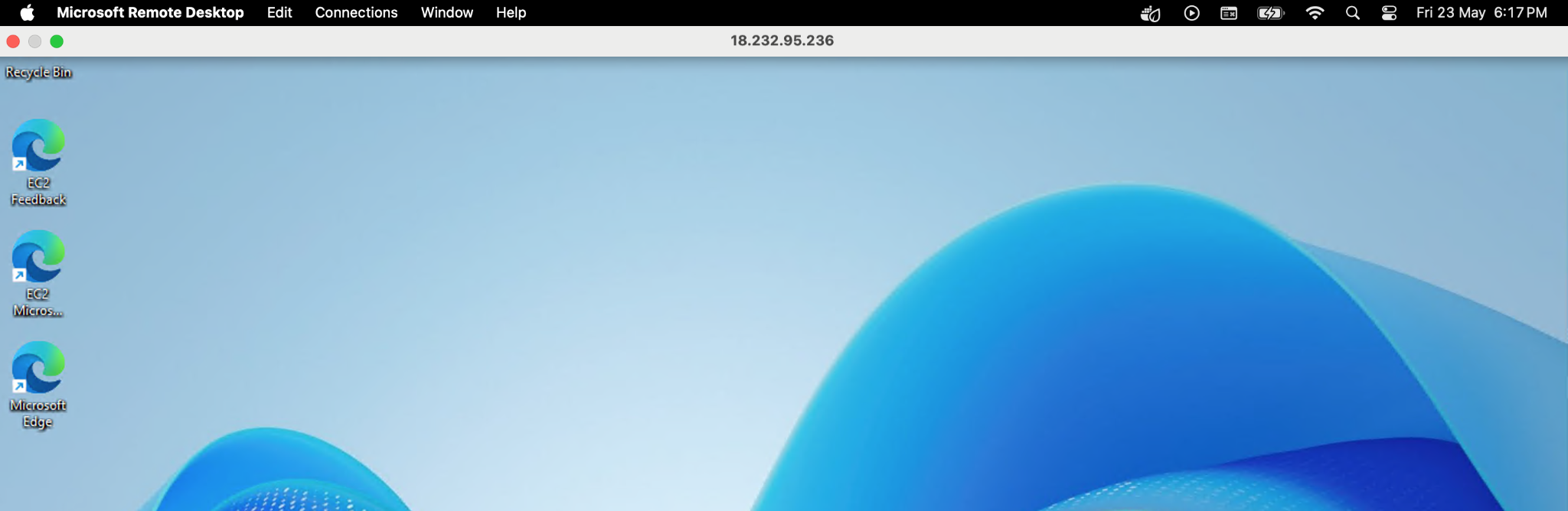
**openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout nginx.key -out nginx.crt**

**Credentials:** Just FYI

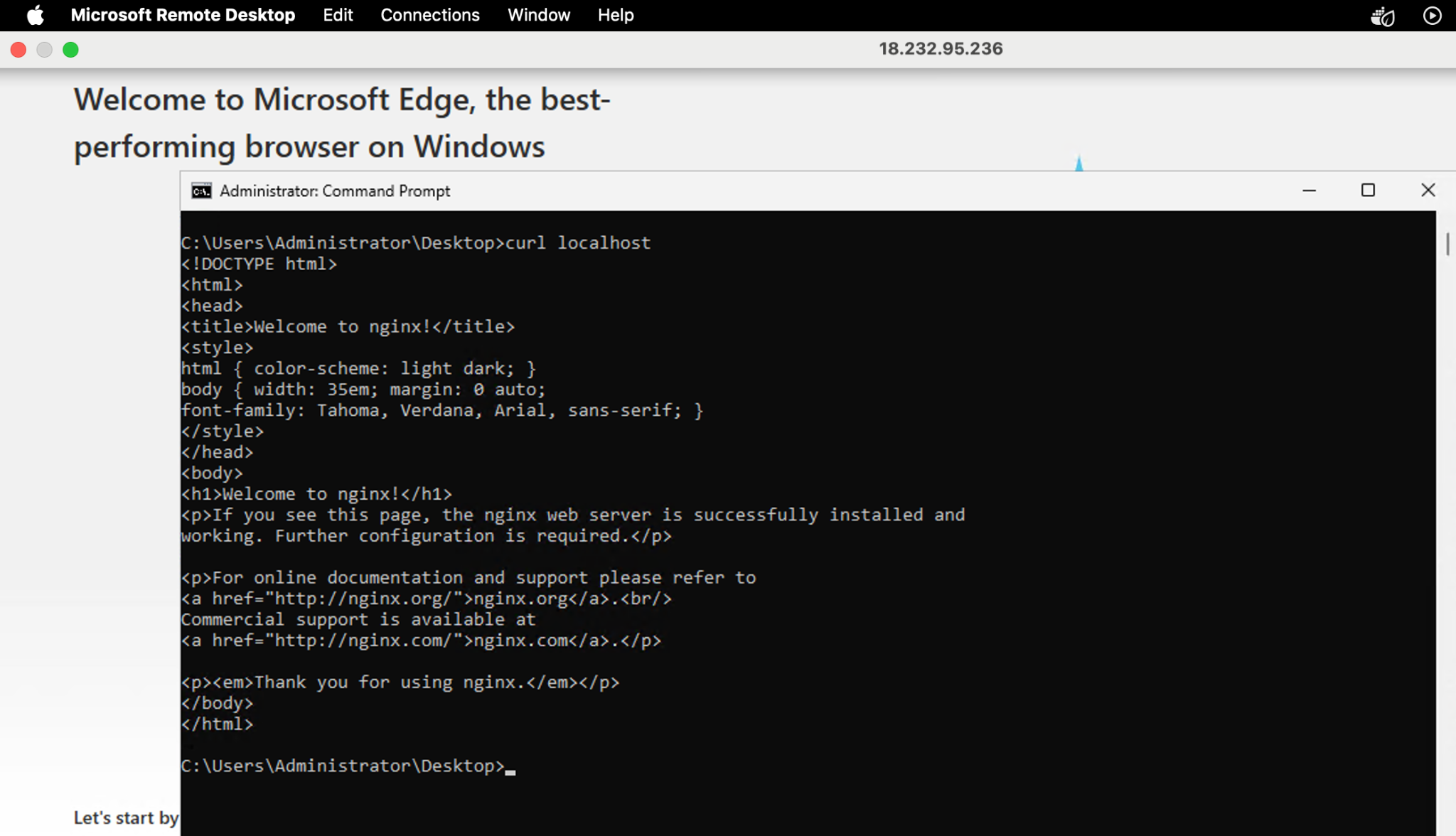
* Administrator is the username
* Generate the password by uploading private key and decrypt the password

**Results- ScreenShots**





**Accessing Nginx From Windows:**



**Observation:**

**Successfully able to Access Nginx Server**