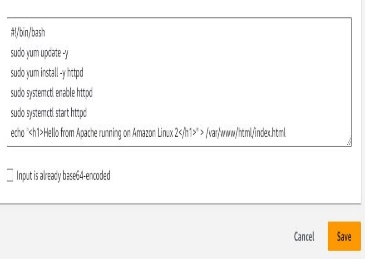
**Task on Ec2**

**1)Launch one ec2 using Amazon Linux 2 image and add script in user data to install Apache.**

* + Login to AWS Management console
  + Navigate to Ec2 Dashboard and click on Lunch instance
  + Provide details
    - Name and tags: Give a name at your creating required Instance
    - Application and OS image (AMI): Select Amazon Linux 2 AMI
    - Instance type: Choose t2.micro (it is selected type)
    - Key pair: create a key pair access to your instance.
    - Network Settings
      * + Security group name : Add security group will be added to all network interfaces (My\_Security\_Group).
        + Description : Help you identify the security group (Allowing specific port number).
        + Ensure to **allow HTTP traffic (port 80)** in the security group rules.
  + Select Advanced Details add User data Script to Install Apache(httpd).



Script:

#!/bin/bash

sudo yum update -y

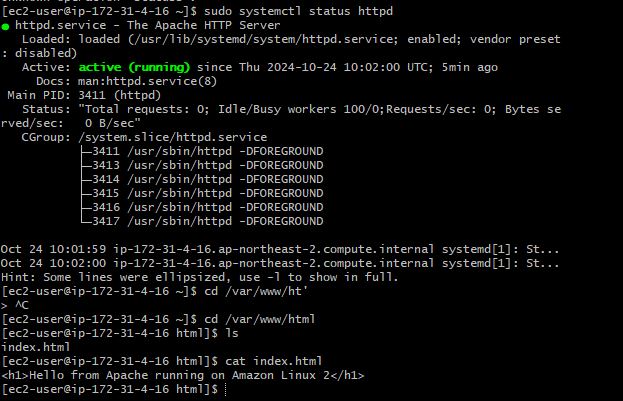
sudo yum install -y httpd

sudo systemctl enable httpd

sudo systemctl start httpd

echo “<html><body><h1>Hello from Apache install in Amazon

Linux 2</h1></body></html>” > /var/www/html/index.html



* + In CLI
    - Run 🡪 sudo systemctl status httpd
    - Change the Directory 🡪 cd /var/www/html
    - View the content on file 🡪 cat index.html

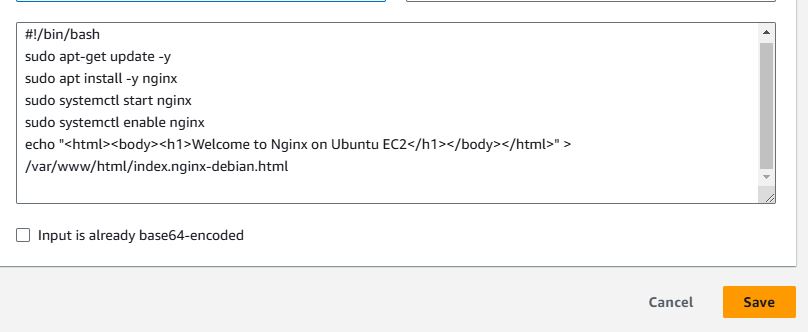


* + Browse
    - Browse the IP address with help of port number

<ip:80>

**2) Launch one ec2 using Ubuntu image and add script in user data to install Nginx.**

* + Login to AWS Management console
  + Navigate to Ec2 Dashboard and click on Lunch instance
  + Provide details
    - Name and tags: Give a name at your creating required Instance(i.e., ‘Nginx-Ubuntu-EC2’)
    - Application and OS image (AMI): Select Ubuntu Server 22.04 LTS
    - Instance type: Choose t2.micro (it is selected type)
    - Key pair: create a key pair access to your instance.
    - Network Settings
      * + Security group name : Add security group will be added to all network interfaces (My\_Security\_Group).
        + Description : Help you identify the security group (Allowing specific port number).
        + Ensure to **allow HTTP traffic (port 80)** in the security group rules.
  + Select Advanced Details add User data Script to Install Nginx.



Script:

#!/bin/bash

sudo apt-get update -y

sudo apt install -y nginx

sudo systemctl start nginx

sudo systemctl enable nginx

echo “<h1>Welcome to Nginx on Ubuntu Ec2</h1>” >

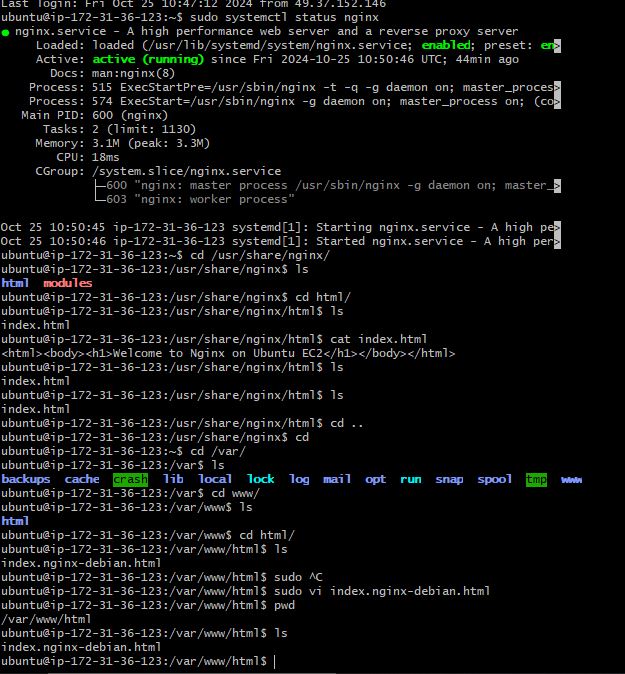
/var/www/html/index.nginx-debian.html

or

echo “<h1>Welcome to Nginx on Ubuntu Ec2</h1>” >

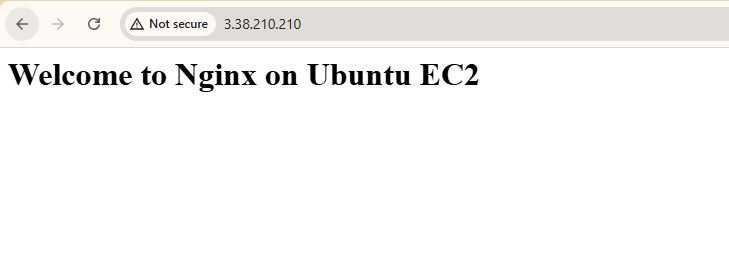
/usr/share/nginx/html/index.html

**There are two different default location nginx**

/usr/share/nginx/html/

/var/www/html/

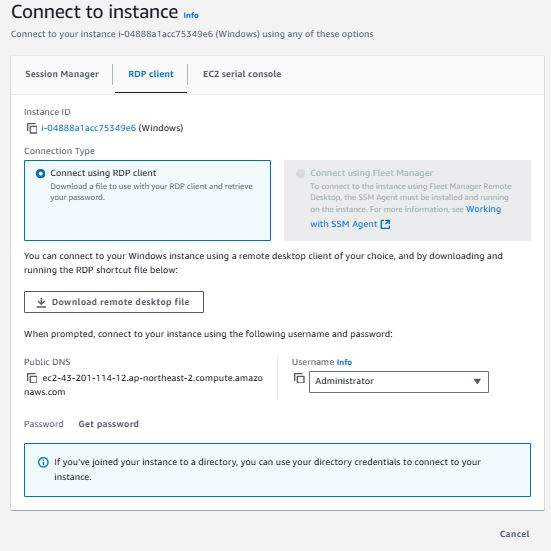
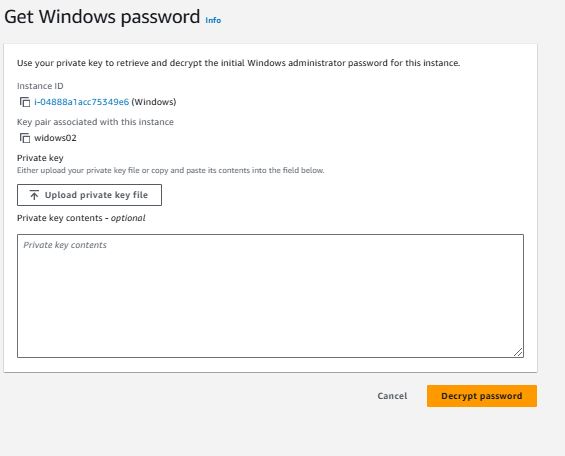
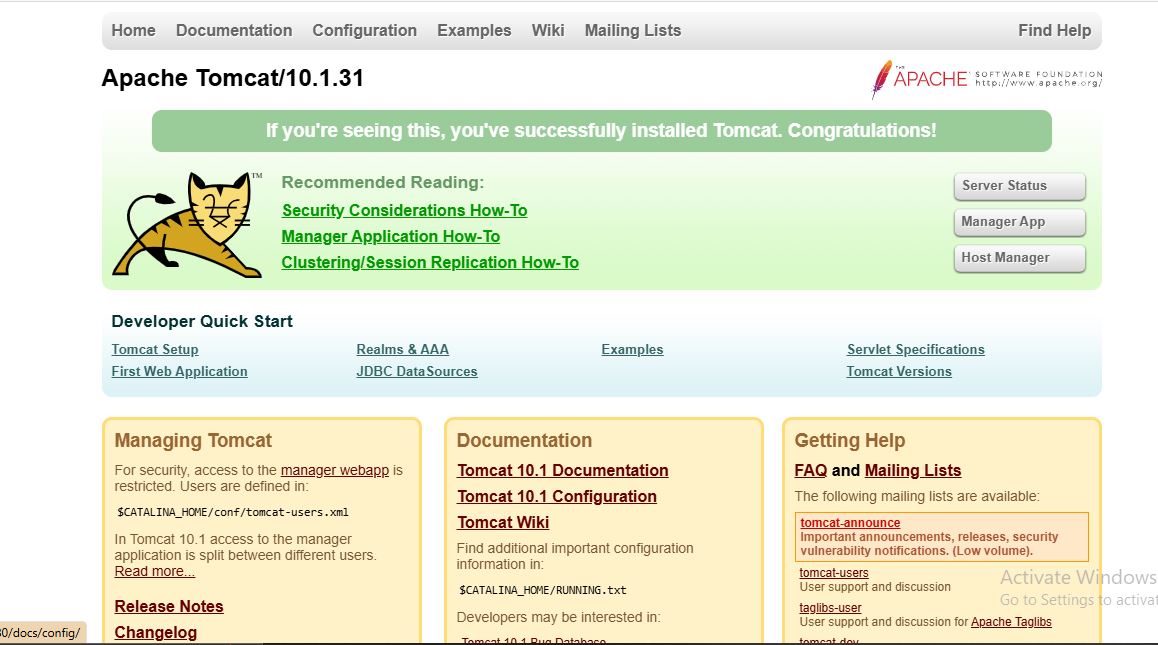
* + In CLI
    - Run 🡪 sudo systemctl status nginx
    - Change the Directory 🡪 cd /var/www/html
    - View the content on file 🡪 cat index. nginx-debian .html

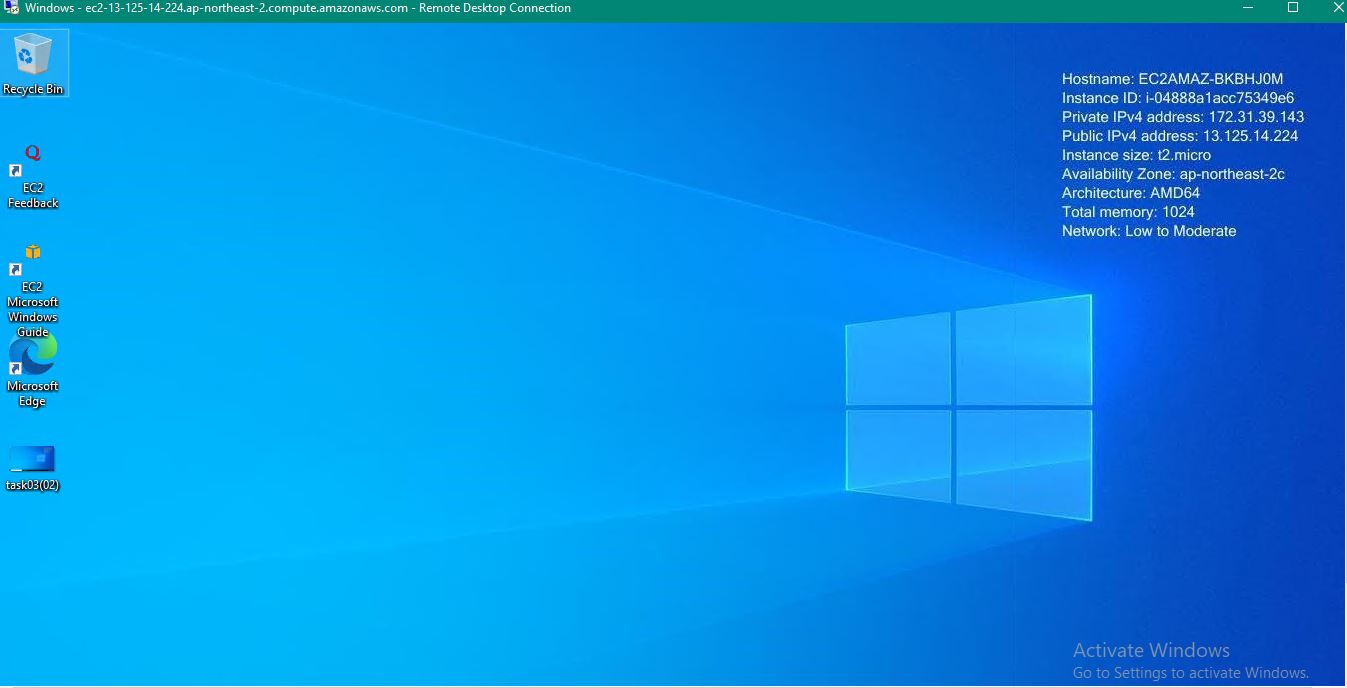


* + Browse
    - Browse the IP address with help of port number

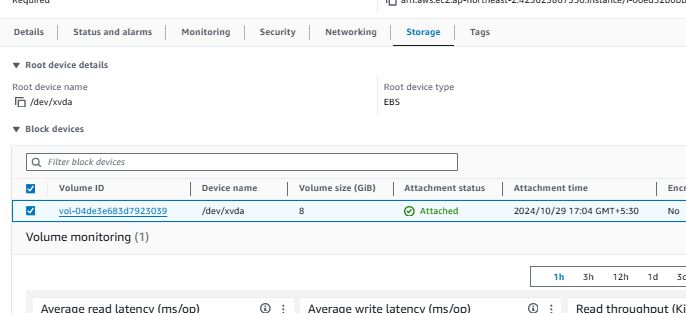
<ip:80>

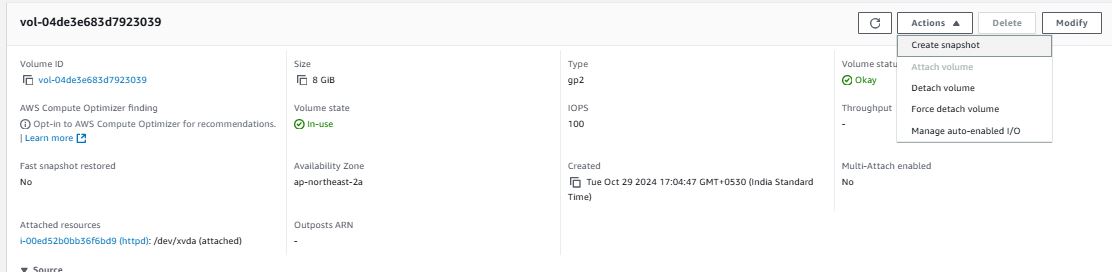
**3) Launch one windows server and install tomcat in windows.**

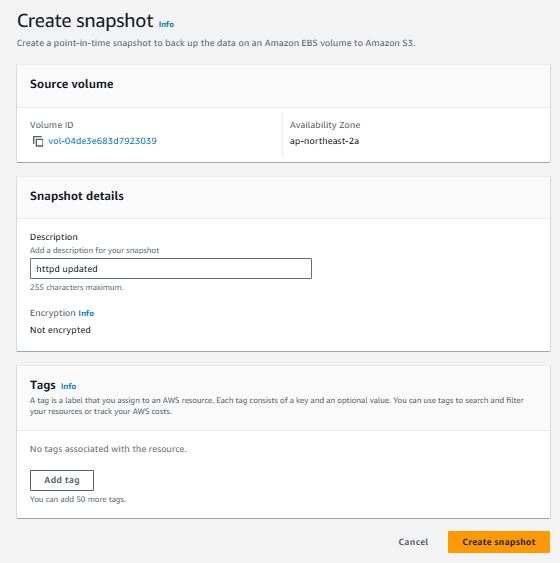
* + Login to AWS Management console
  + Navigate to Ec2 Dashboard and click on Lunch instance
  + Provide details
    - Name and tags: Give a name at your creating required Instance(i.e., Window)
    - Application and OS image (AMI): Select Microsoft windows Server 2022 or 2019
    - Instance type: Choose t2.micro (it is selected type).
    - Key pair: create a key pair access to your instance.

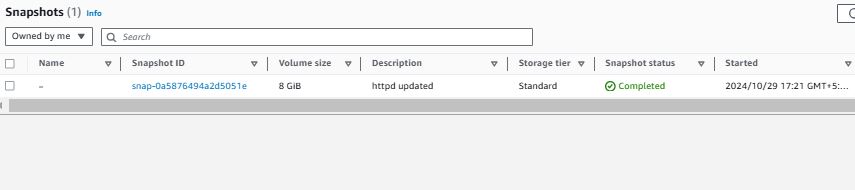


4**) Take snapshot of the instane created in Task 1**

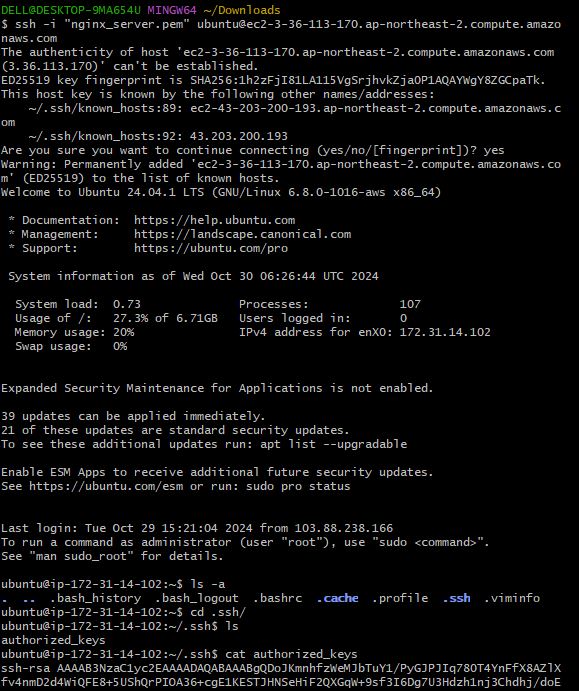
****

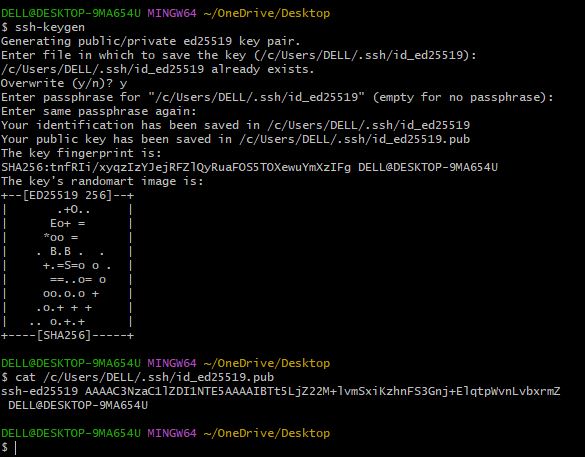


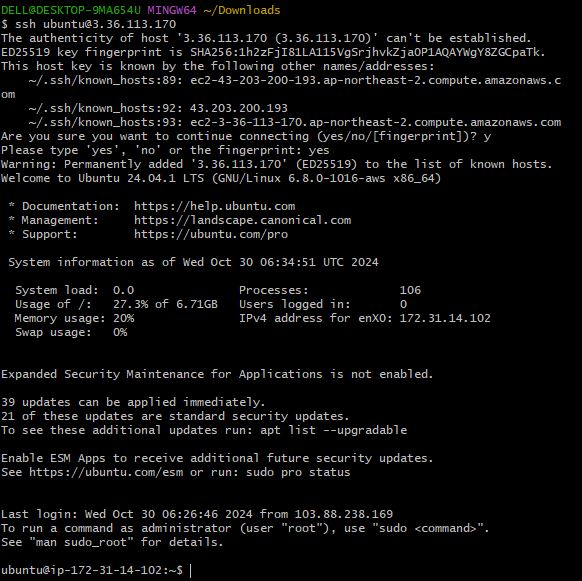


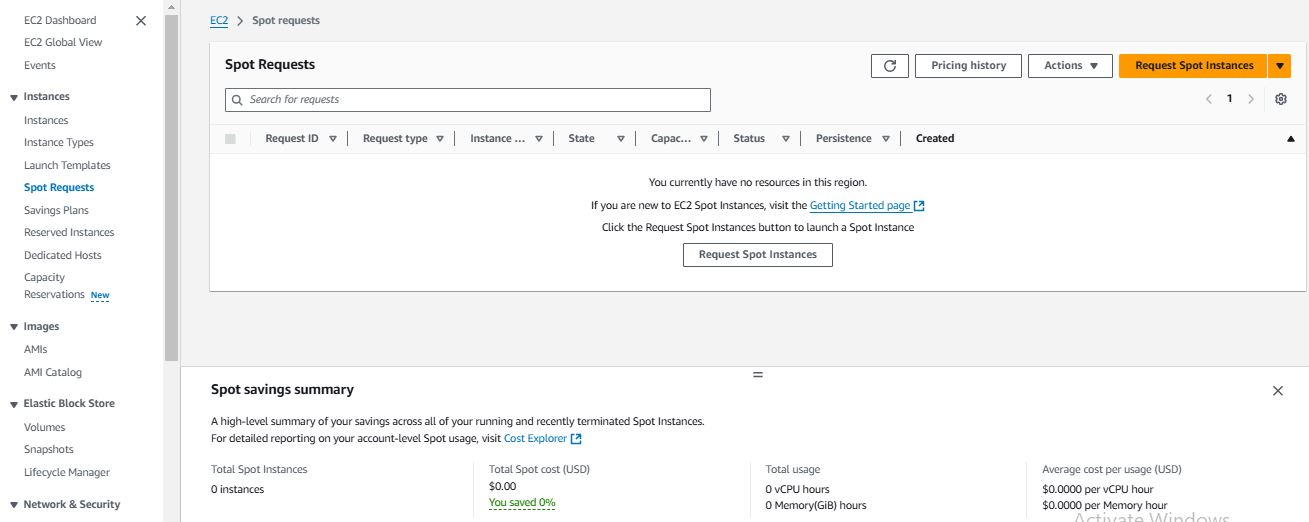
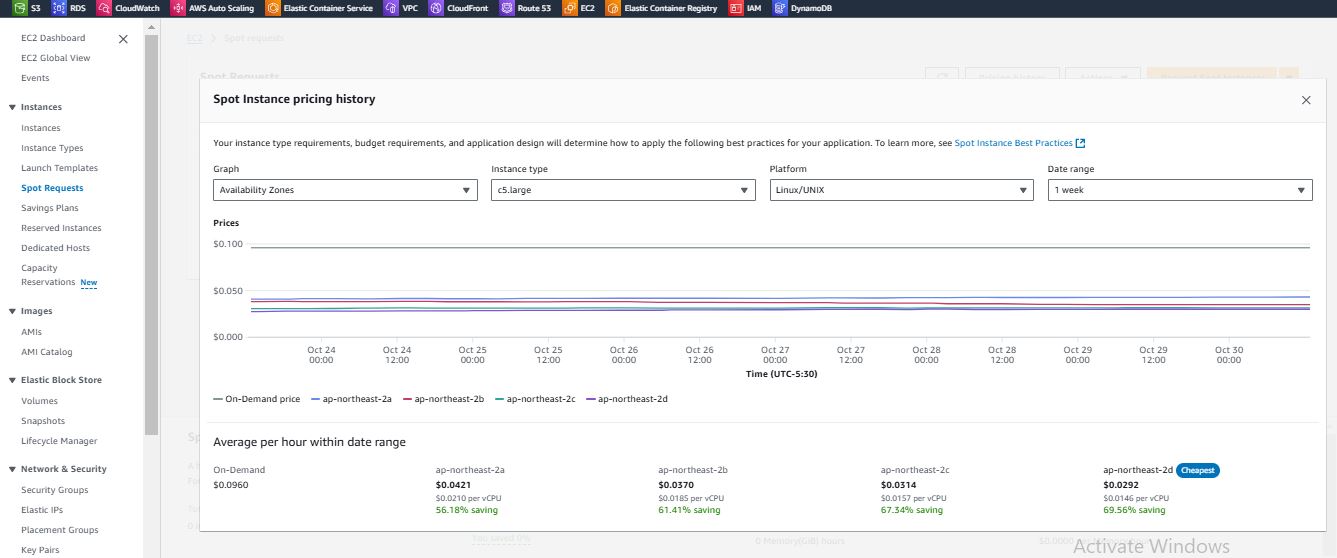


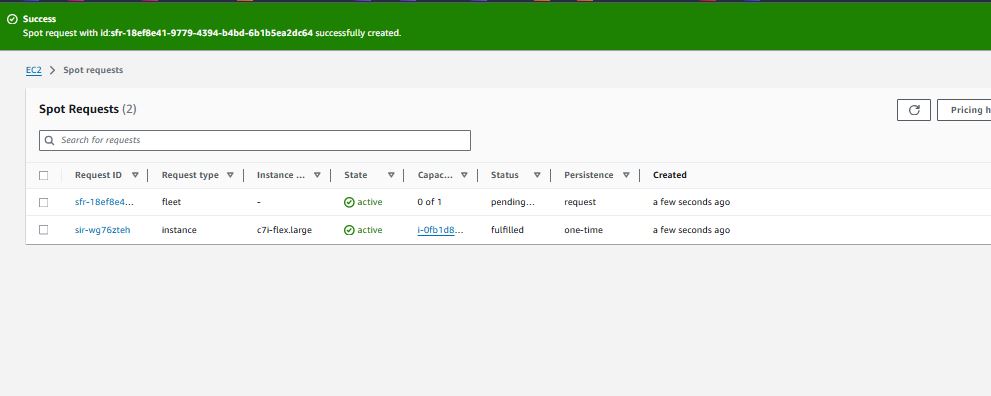
**5) Assign password less authentication for ec2 created on Task 2.**

****

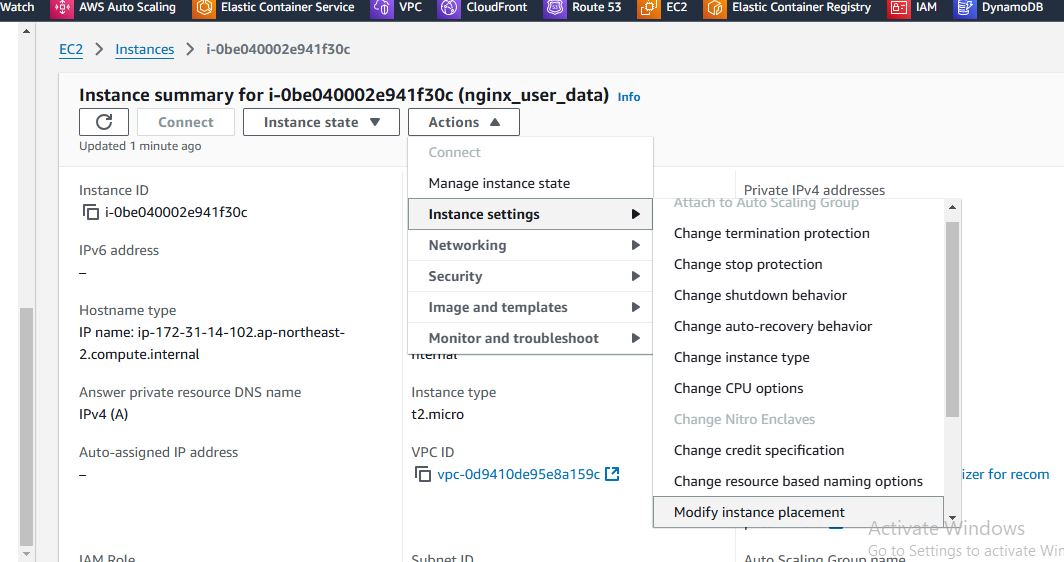
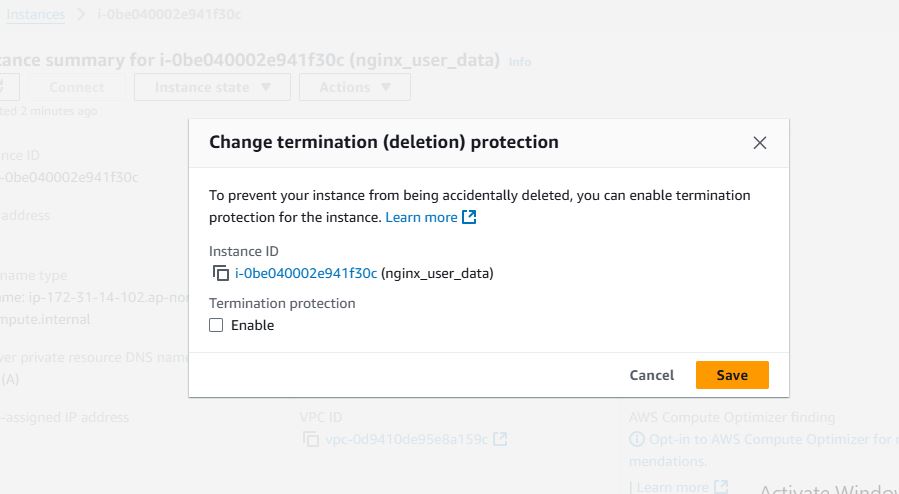
****

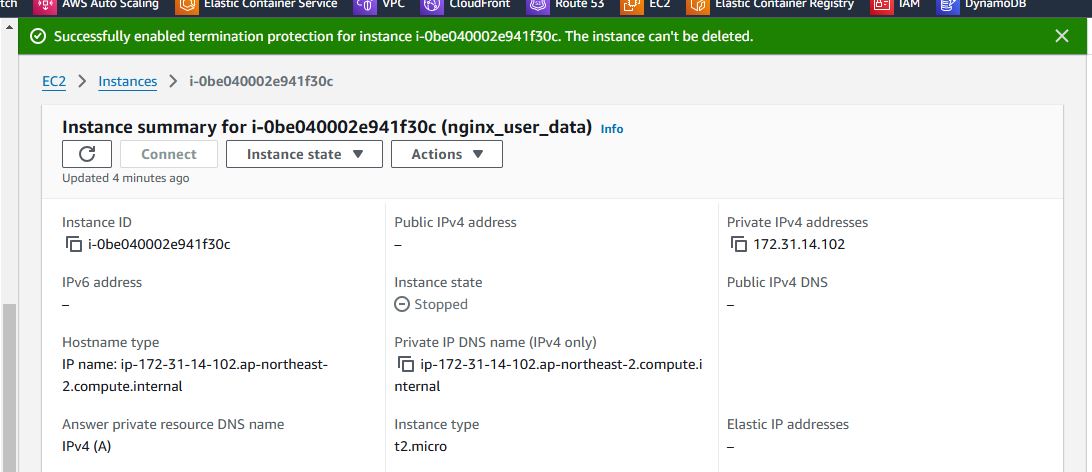


6) Launch any ec2 using spot purchasing option.



7) Enable Termination policy on ec2 created in Task 2.





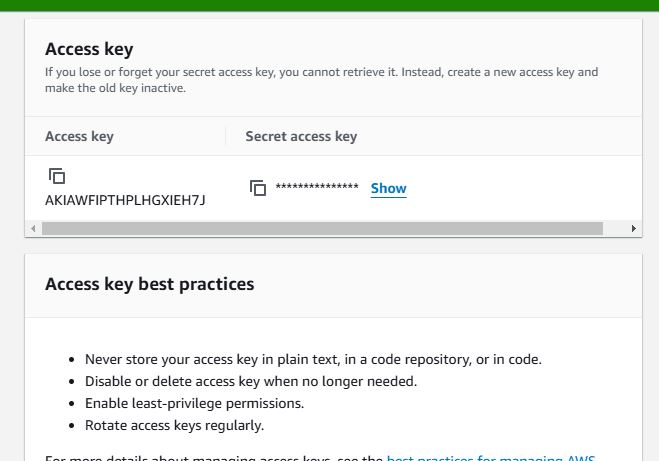
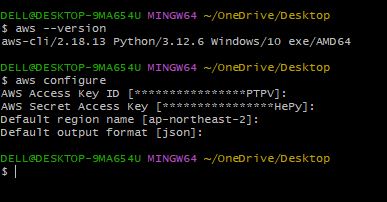
8) Launch one ec2 using Aws CLI.

Download aws cli local machine 🡪 Go to <https://docs.aws.amazon.com/cli/latest/userguide/getting-started-install.html>

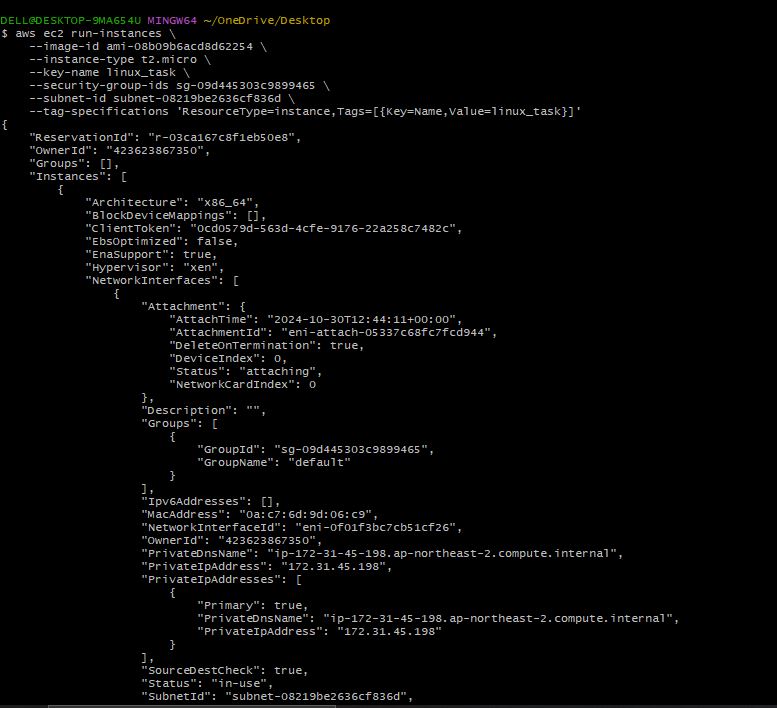
Alternate methode

C:\> msiexec.exe /i <https://awscli.amazonaws.com/AWSCLIV2.msi>

* aws –version



* In CLI

$ aws ec2 run-instances

--image-id ami-08b09b6acd8d62254

--instance-type t2.micro

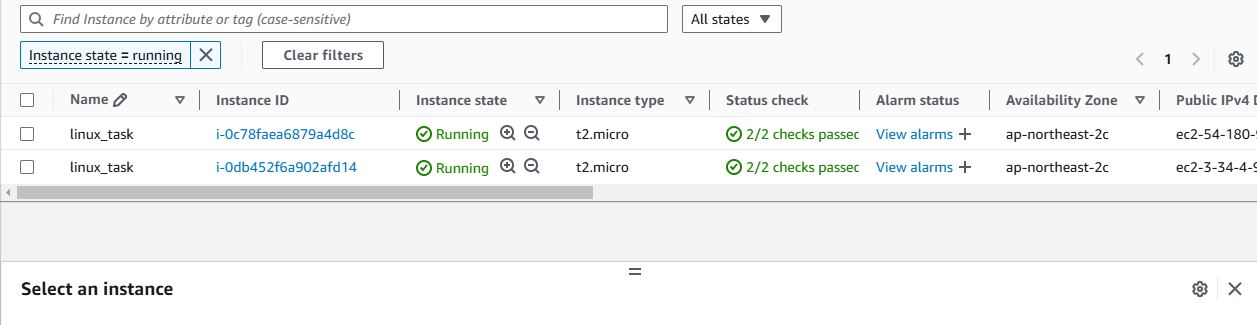
--key-name linux\_task

--security-group-ids sg-0d00d87955d7f2bb3

--subnet-id subnet-08219be2636cf836d

--tag-specifications 'ResourceType=instance,Tags

=[{Key=Name,Value=linux\_task}]'

---- Out Put ----