Aws Assignment

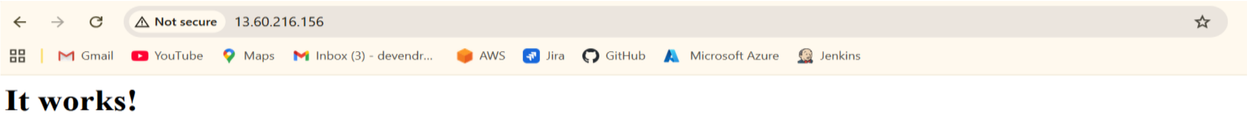
1. Launch one EC2 using Amazon Linux 2 image and add a script in user data to install Apache

- Launch a ec2 instance with a pem key and your public ip.
- Then check the security inbound rules ssh should be 22 and http in 80
- Then go to the bottom instance and additional info open it
- And use a bash script for it
- Then go the git bash connect to the server and take the public ip and paste it with last gave that pot number of :80
- Here the results are

Bash script download apache process:

```
    #!/bin/bash

    # Update system

    yum update -y

    # Install Apache (httpd)

    yum install -y httpd

    # Enable Apache to start on boot

    systemctl enable httpd

    # Start Apache service

    systemctl start httpd

    # Create a test index.html

echo "<h1>Welcome to Apache on Amazon Linux 2!</h1>" > /var/www/html/index.html
```

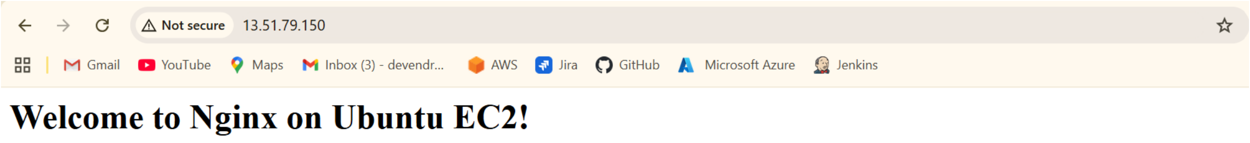
Note: Using # it means gave a comment it will gave you a message who ever check it that one he will understand easily.



It works!

2). Launch one EC2 using Ubuntu image and add a script in user data to install nginx

- Lauch a ec2 instance with a pem key and your public ip.
- Then check the security inbound rules ssh should be 22 and httpd in 80
- Then go to the bottom instance and additional info open it
- And use a bash script for it nginx
- Then go the git bash connect to the server and take the publc ip and paste it with last gave that port number of :80
- Here the results are



Welcome to Nginx on Ubuntu EC2!

- **Bash script for nginx:**

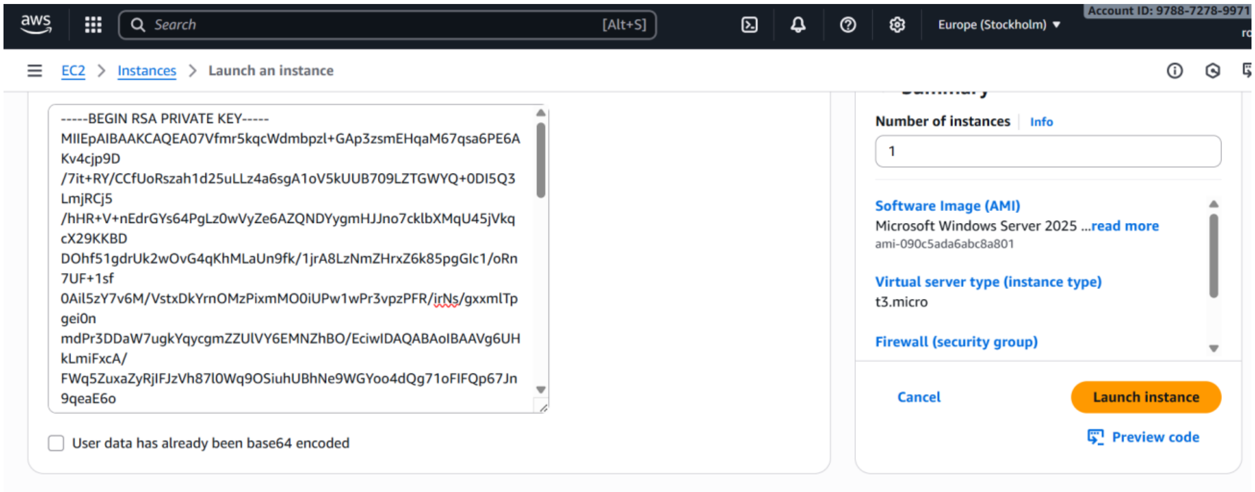
```
#1/bin/bash
#update system packages
Apt-get update –y
#install nginx
Apt-get install –y nginx
#Enable nginx to start on boot
Systemctl enable nginx
```

#start nginx service
Systemctl start nginx
#ceation a custom index page

Echo “<h1> welcome to nginx on Ubuntu EC2!</h1>”> /var/ww/html/index.nginx-debian.html

Note: # uses for only gave a comments just want to know the purpose what we are doing in bash scripting.

3) Launch one Windows server and install Tomcat on Windows



```
-----BEGIN RSA PRIVATE KEY-----
MIIEpAIBAAKCAQEA07Vfmr5kqcWdmbpzl+GAp3zsmEHqaM67qsa6PE6A
Kv4cjp9D
/7it+RY/CCfUoRszah1d25uLLz4a6sgA1oV5kUUB709LZTGWYQ+0DI5Q3
LmjRCj5
/hHR+V+nEdrGYs64PgLz0wVyZe6AZQNDYygmHJJno7cklbXMqU45jVkq
cX29KKBD
DOhf51gdrUk2wOvG4qKhMLaUn9fk/1jrA8LzNmZHrxZ6k85pgGlc1/oRn
7UF+1sf
0Ail5zY7v6M/VstxDkYrnOMzPixmMO0iUPw1wPr3vpzPFR/irNs/gxxmlTp
gei0n
mdPr3DDaW7ugkYqycgmZZUlVY6EMNZhBO/EciwIDAQABAoIBAAVg6UH
kLmiFxcA/
FWq5ZuxaZyRjIFJzVh87l0Wq9OSiuhUBhNe9WGYoo4dQg71oFIFQp67Jn
9qeaE6o
```

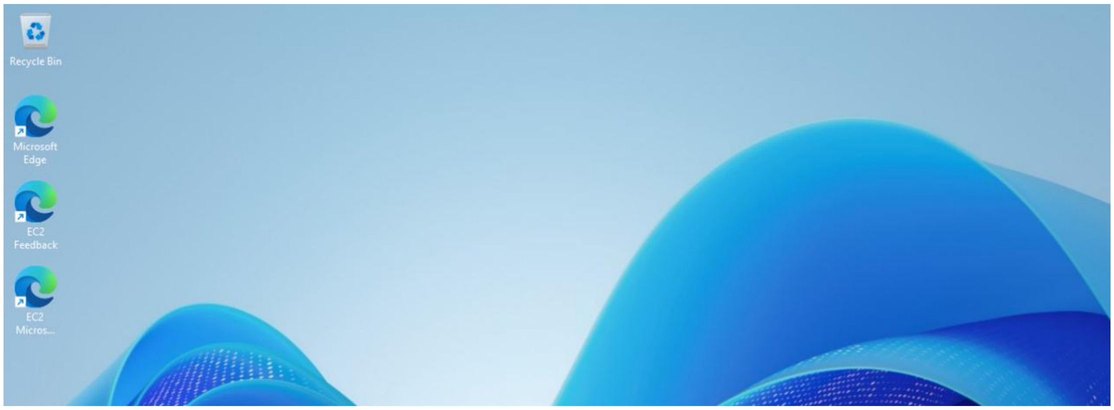
☐ User data has already been base64 encoded

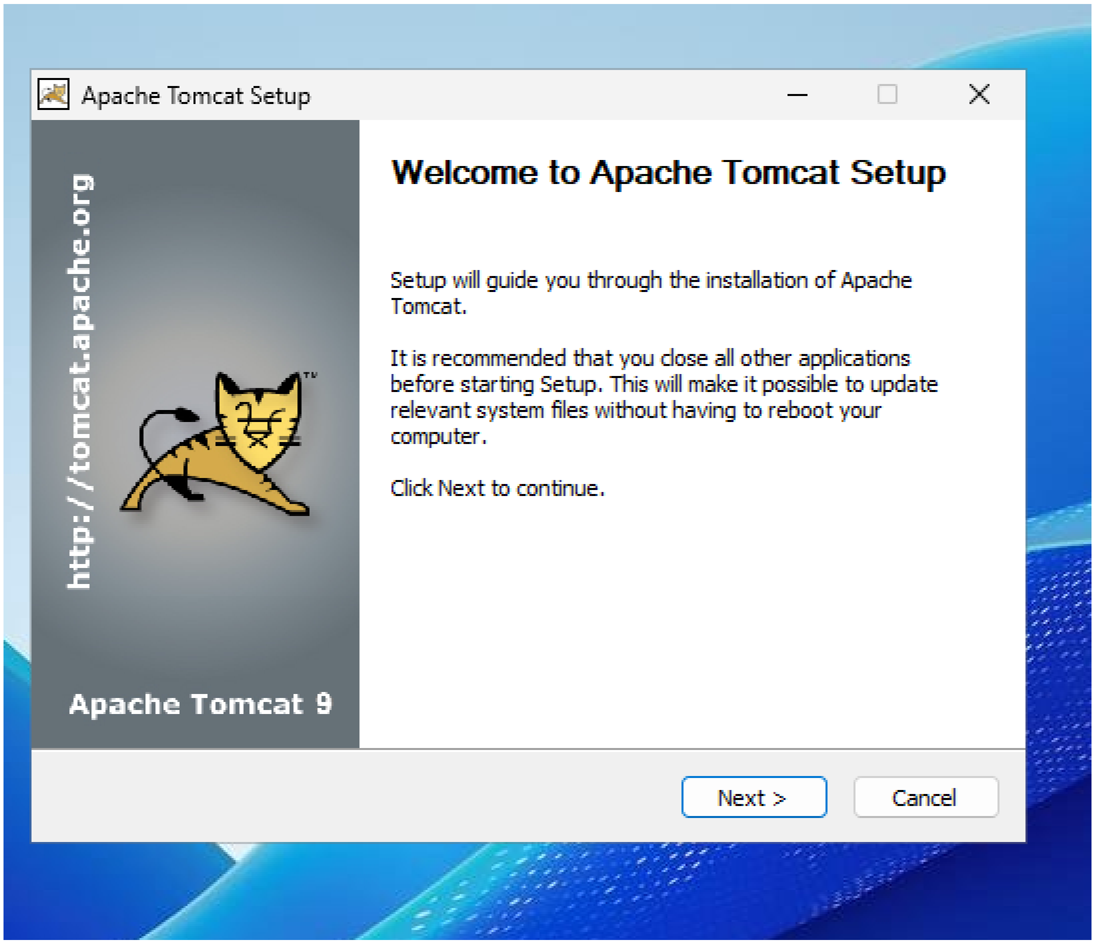
Number of instances: 1

Software Image (AMI)
Microsoft Windows Server 2025 ...read more
ami-090c5ada6abc8a801

Virtual server type (instance type)
t3.micro

Firewall (security group)





Welcome to Apache Tomcat Setup

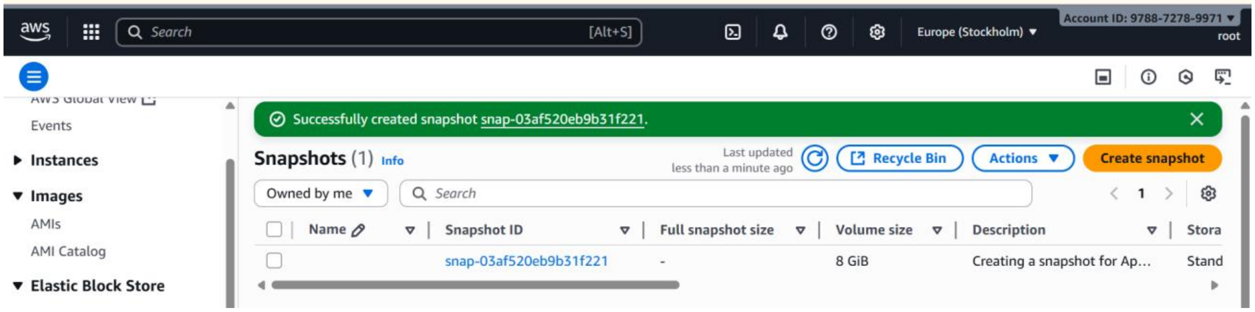
Setup will guide you through the installation of Apache Tomcat.

It is recommended that you close all other applications before starting Setup. This will make it possible to update relevant system files without having to reboot your computer.

Click Next to continue.

4) Take a snapshot of the instance created in Task 1

- Go to EC2 dashboard → click on instances
- Select the instance you created in nginx web check
- Scroll down to storage → find the volume ID (attached root volume)
- Click the Volume Id → you'll be redirected to the EBS volumes page.
- Create snapshot and select the instance which one you want here the results are..



Successfully created snapshot snap-03af520eb9b31f221.

Snapshots (1) Info

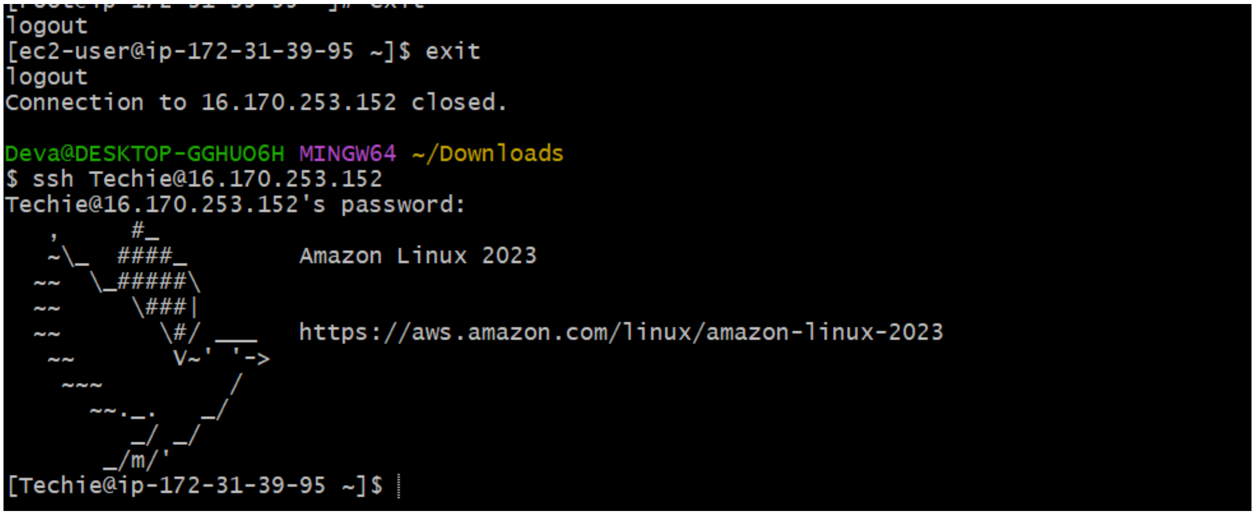
Last updated less than a minute ago

□	Name	Snapshot ID	Full snapshot size	Volume size	Description	Stora
□		snap-03af520eb9b31f221	-	8 GiB	Creating a snapshot for Ap...	Stand

5) Assign password less authentication for the EC2 created in Task 2.

- To create password less authentication:
- Crate a ssh-key in our local machine – ssh –keygen
- Then copy your public key using --- at/c/users/DELL/ .ssh/id_rsa.pub

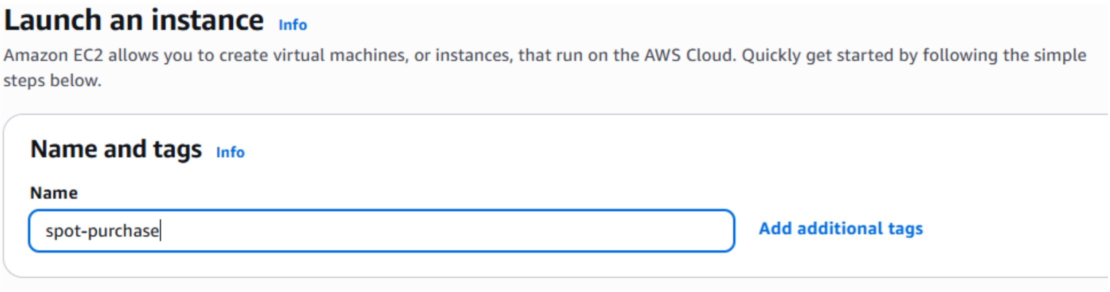
- Create an user --- useradd techie
- Password for user --- passwd techie
- Ssh-keygen --- crate a ssh key in ec2 machine
- Vi /root/.ssh/id_rsa.pub (paste your local machine key here by keeping)
- Present key as same ..
- Vi/etc/ssh/sshd_config --- enable password authentication as yes
- Systemctl restart sshd --- restart your machine :
- Ssh techie@public-ip
- Allow fingerprint authentication :yes



```
[root@ip-172-31-39-95 ~]# exit
logout
[ec2-user@ip-172-31-39-95 ~]$ exit
logout
Connection to 16.170.253.152 closed.

Deva@DESKTOP-GGHUO6H MINGW64 ~/Downloads
$ ssh Techie@16.170.253.152
Techie@16.170.253.152's password:
   ,     #_
   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\
  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
[Techie@ip-172-31-39-95 ~]$
```

6) Launch any EC2 using the spot purchasing option.

- Click on launch instance
- By using spot instance launch one instance

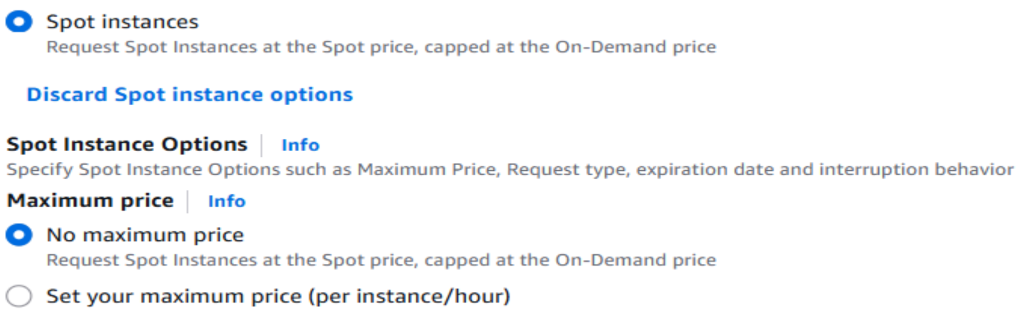


Launch an instance Info

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

Name and tags Info

Name: spot-purchase



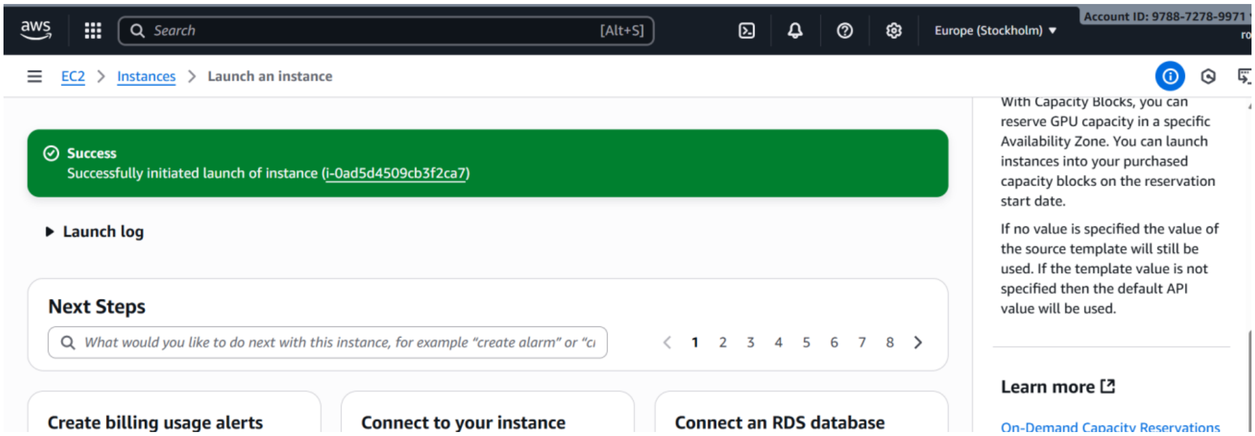
● Spot instances
Request Spot Instances at the Spot price, capped at the On-Demand price

Spot Instance Options | Info
Specify Spot Instance Options such as Maximum Price, Request type, expiration date and interruption behavior

Maximum price | Info

● No maximum price
Request Spot Instances at the Spot price, capped at the On-Demand price

○ Set your maximum price (per instance/hour)



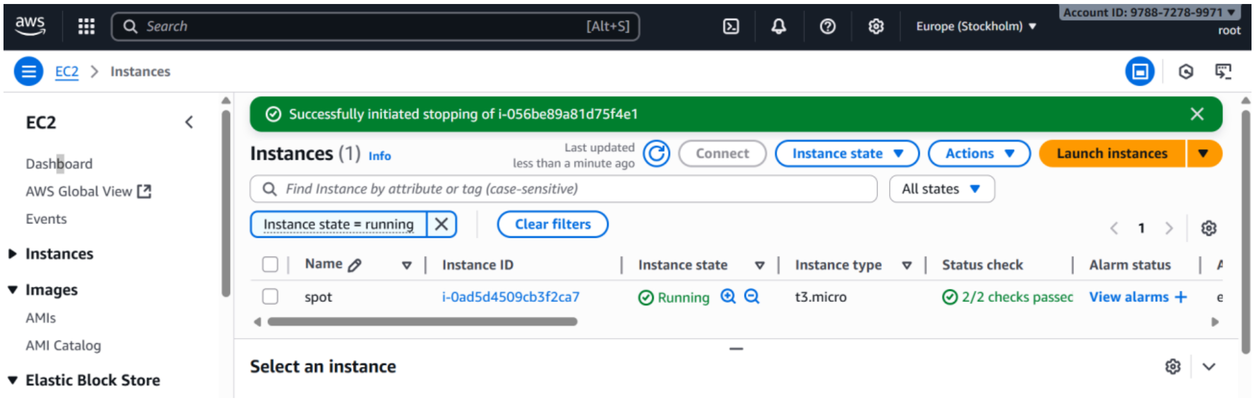
aws
Search [Alt+S]
Europe (Stockholm)
Account ID: 9788-7278-9971
EC2 > Instances > Launch an instance

Success
Successfully initiated launch of instance (i-0ad5d4509cb3f2ca7)

Launch log
Next Steps
What would you like to do next with this instance, for example “create alarm” or “ci
1 2 3 4 5 6 7 8
Create billing usage alerts
Connect to your instance
Connect an RDS database

With Capacity Blocks, you can reserve GPU capacity in a specific Availability Zone. You can launch instances into your purchased capacity blocks on the reservation start date.

If no value is specified the value of the source template will still be used. If the template value is not specified then the default API value will be used.



Successfully initiated stopping of i-056be89a81d75f4e1

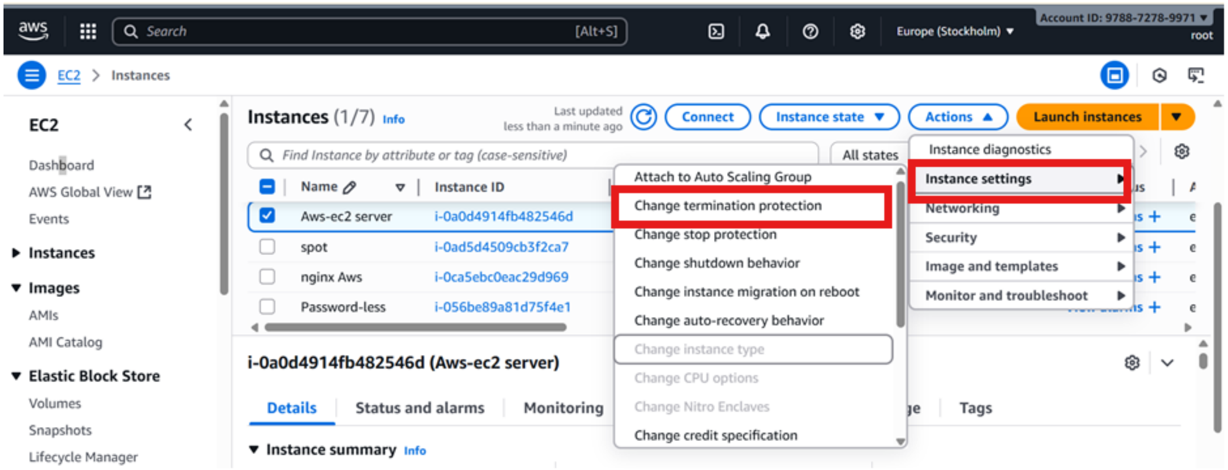
Instance state = running

Name	Instance ID	Instance state	Instance type	Status check	Alarm status
spot	i-0ad5d4509cb3f2ca7	Running	t3.micro	2/2 checks passec	View alarms +

7) Enable termination policy on the EC2 created in Task 2.

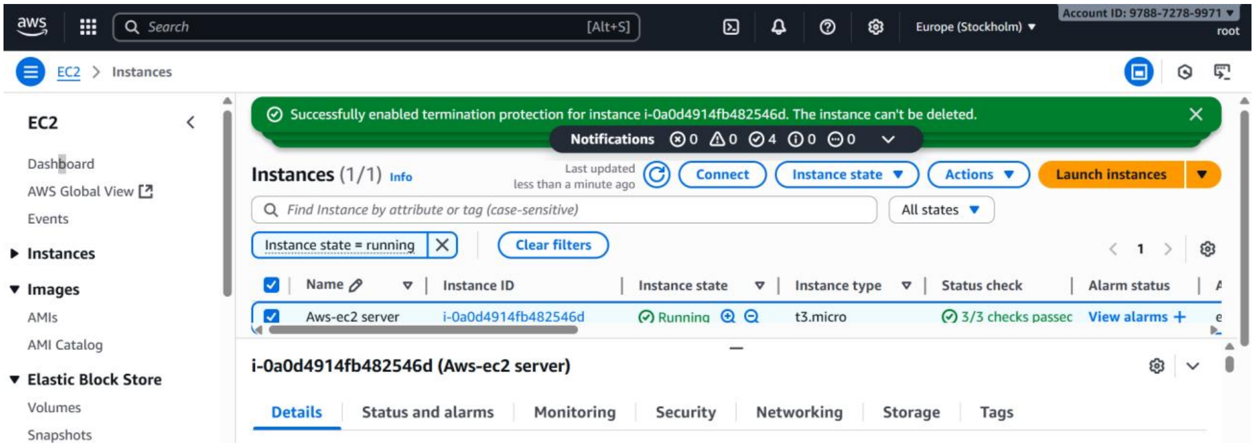
- Go to the EC2 Dashboard in AWS console

- Select the EC2 instance you created in task 2
- In the instance settings menu, click change termination protection
- Select Enablr and save .



	Name	Instance ID
☑	Aws-ec2 server	i-0a0d4914fb482546d
□	spot	i-0ad5d4509cb3f2ca7
□	nginx Aws	i-0ca5ebc0eac29d969
□	Password-less	i-056be89a81d75f4e1

i-0a0d4914fb482546d (Aws-ec2 server)



Successfully enabled termination protection for instance i-0a0d4914fb482546d. The instance can't be deleted.

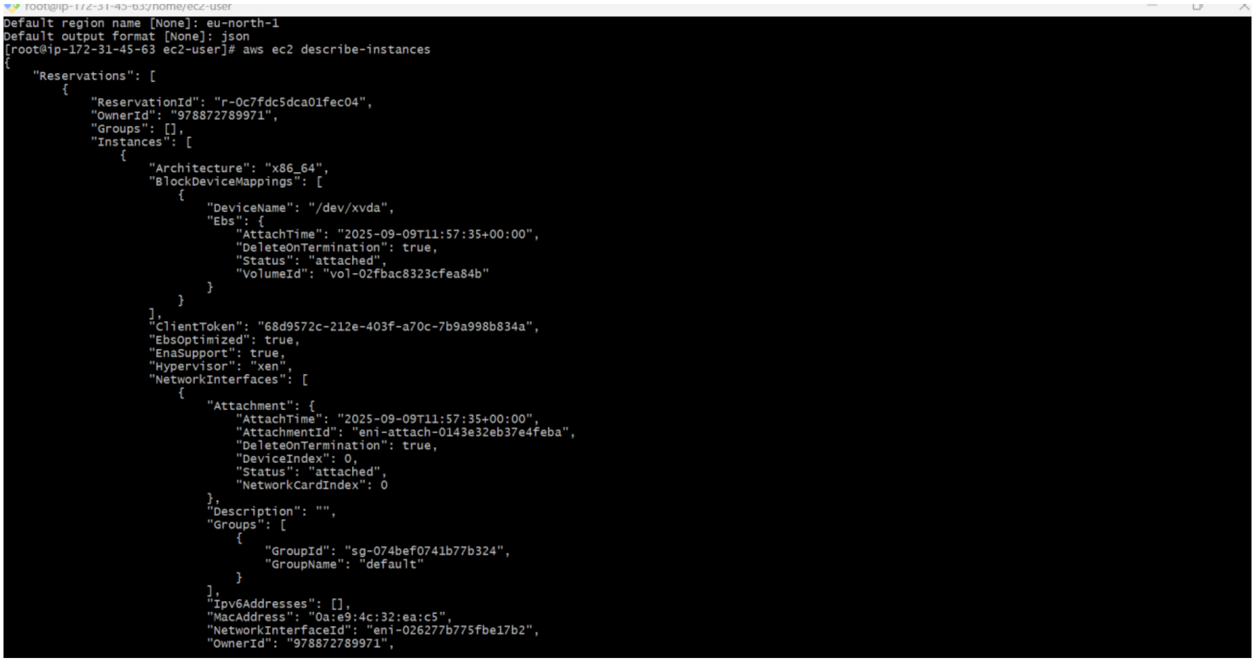
	Name	Instance ID	Instance state	Instance type	Status check	Alarm status
☑	Aws-ec2 server	i-0a0d4914fb482546d	Running	t3.micro	3/3 checks passec	View alarms +

i-0a0d4914fb482546d (Aws-ec2 server)

8) Launch one EC2 using AWS CLI.

- Launch instance name with AWS
- And to connect to the server then
- Wget download cli https://awscli.amazonaws.com/AWSCLIV2.msi
- Then gave a command of aws configure it will show you a options like
- Access key
- Secret access key
- Region and format
- For access key and secrete key open our profile then security credentials and it will generate the keys

- Then aws ec2 describe-instances
- Here the results



```
Default region name [None]: eu-north-1
Default output format [None]: json
[root@ip-172-31-45-63 ec2-user]# aws ec2 describe-instances
{
    "Reservations": [
        {
            "ReservationId": "r-0c7fdc5dca01fec04",
            "OwnerId": "978872789971",
            "Groups": [],
            "Instances": [
                {
                    "Architecture": "x86_64",
                    "BlockDeviceMappings": [
                        {
                            "DeviceName": "/dev/xvda",
                            "Ebs": {
                                "AttachTime": "2025-09-09T11:57:35+00:00",
                                "DeleteOnTermination": true,
                                "Status": "attached",
                                "VolumeId": "vol-02fbac8323cfea84b"
                            }
                        }
                    ],
                    "ClientToken": "68d9572c-212e-403f-a70c-7b9a998b834a",
                    "EbsOptimized": true,
                    "EnaSupport": true,
                    "Hypervisor": "xen",
                    "NetworkInterfaces": [
                        {
                            "Attachment": {
                                "AttachTime": "2025-09-09T11:57:35+00:00",
                                "AttachmentId": "eni-attach-0143e32eb37e4feba",
                                "DeleteOnTermination": true,
                                "DeviceIndex": 0,
                                "Status": "attached",
                                "NetworkCardIndex": 0
                            },
                            "Description": "",
                            "Groups": [
                                {
                                    "GroupId": "sg-074bef0741b77b324",
                                    "GroupName": "default"
                                }
                            ],
                            "Ipv6Addresses": [],
                            "MacAddress": "0a:e9:4c:32:ea:c5",
                            "NetworkInterfaceId": "eni-026277b775fbe17b2",
                            "OwnerId": "978872789971",
```