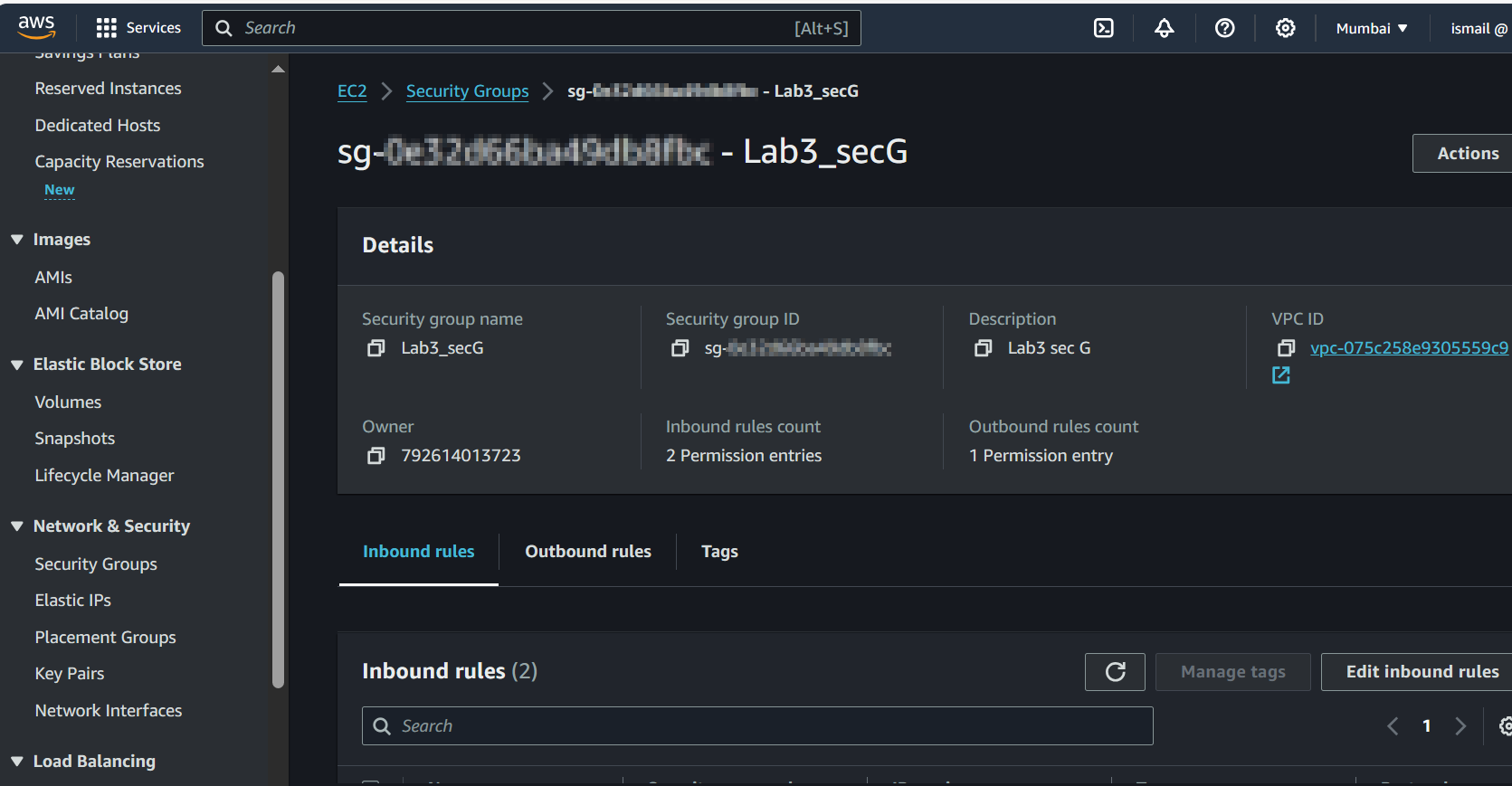
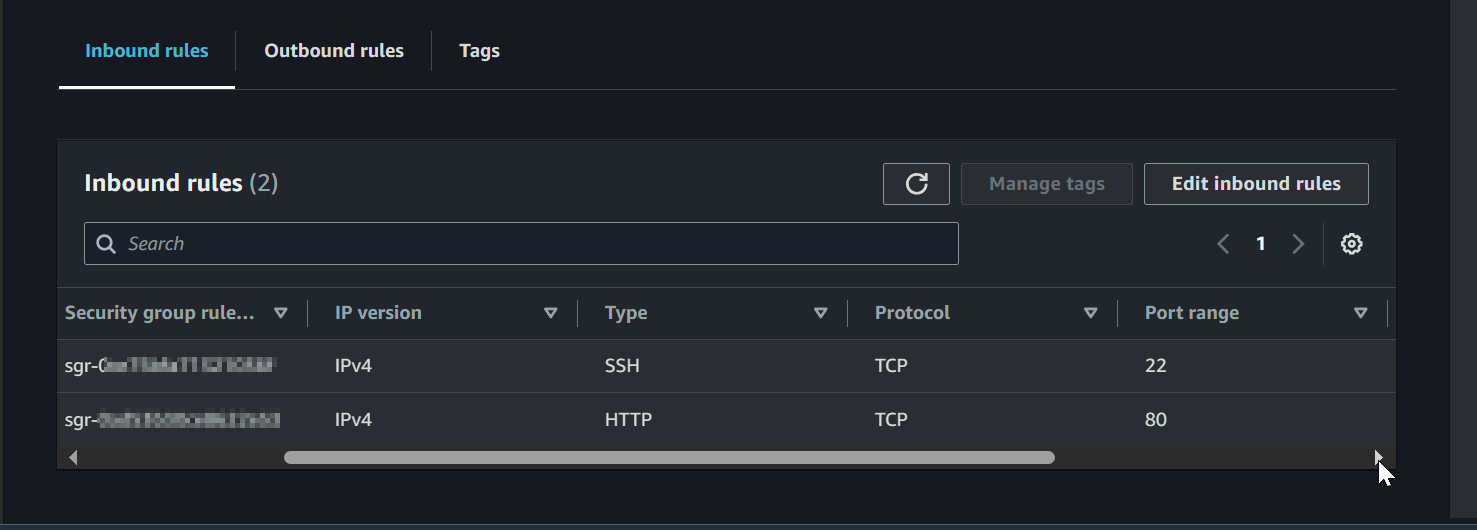
**AWS Hand On (CONSOLE)**

1. Create Security Group:

- Create one security group for the web server.

- Configure inbound rules for the web server security group to allow HTTP traffic (port 80) and SSH traffic (port 22) from any source.





==================================================================================

2. Launch EC2 Instance:

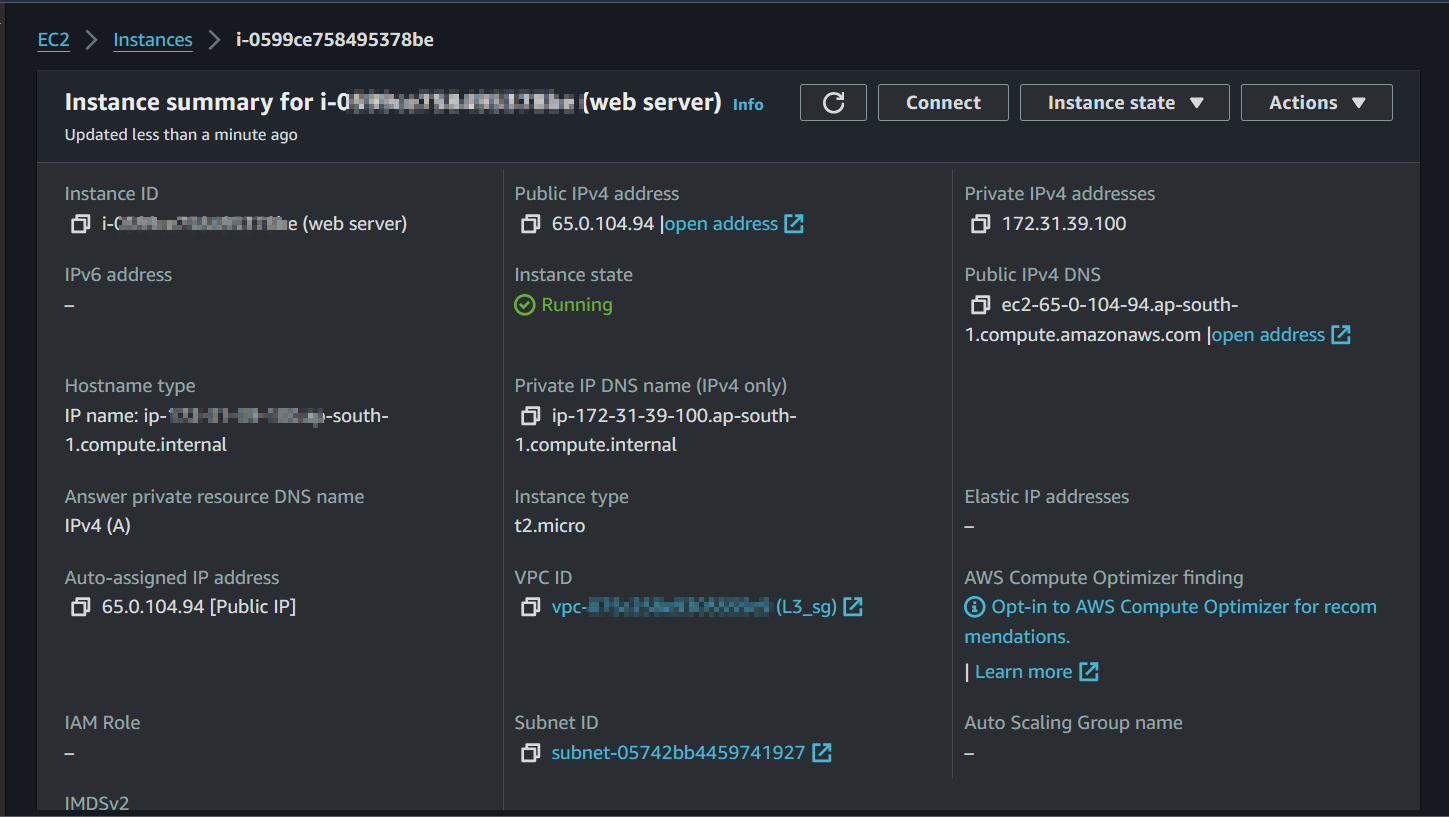
- Launch an EC2 instance for the web server using Amazon Linux 2 AMI.

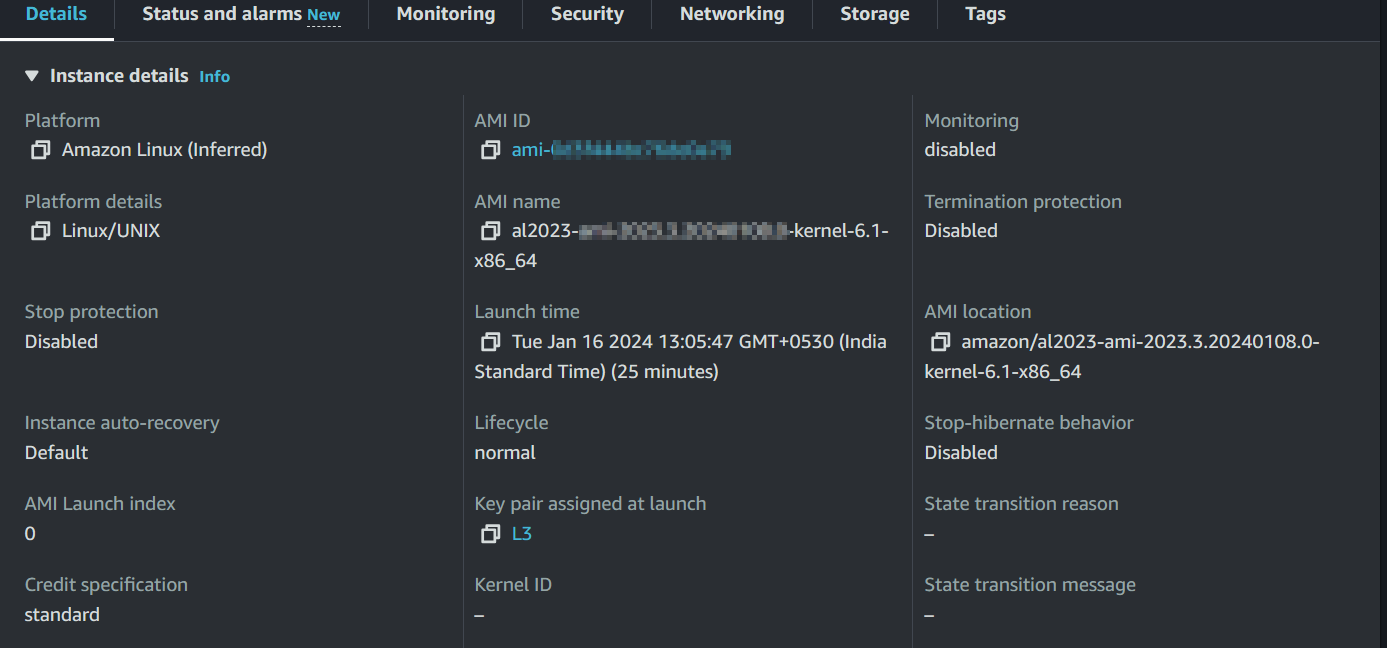
- Associate the web server security group created earlier with this instance.

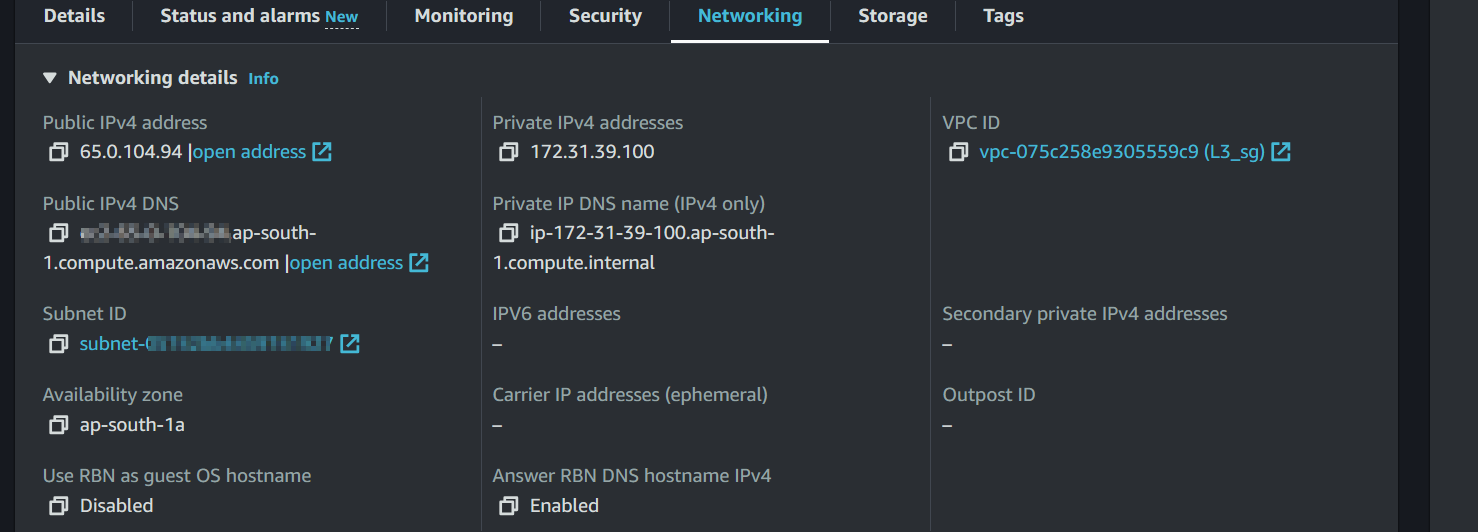
- Use an appropriate instance type for a web server.

- Ensure the instance has a public IP address.

Sol:-







3. SSH Access:

- Generate an SSH key pair for secure access to the instances.

- Configure the web server instance to accept SSH connections using the generated key pair.

- Attempt to SSH into the web server instance to verify successful access.

Sol:-

cd /mnt/c/Users/Dell/Downloads

chmod 400 L3.pem

root@DESKTOP-SQGRBLG:/mnt/c/Users/dell/Downloads# chmod 400 L3.pem

root@DESKTOP-SQGRBLG:/mnt/c/Users/dell/Downloads# ssh -i L3.pem ec2-user@65.0.104.94

(aws\_key)

The authenticity of host '65.0.104.94 (65.0.104.94)' can't be established.

ED25519 key fingerprint is SHA256:qa8ZTMK1WPjBu4MidV73nkUBEYzpVNXEMcjfvb5cTiQ.

This key is not known by any other names

Are you sure you want to continue connecting (yes/no/[fingerprint])? yes

Warning: Permanently added '65.0.104.94' (ED25519) to the list of known hosts.

, #\_

~\\_ ####\_ Amazon Linux 2023

~~ \\_#####\

~~ \###|

~~ \#/ \_\_\_ https://aws.amazon.com/linux/amazon-linux-2023

~~ V~' '->

~~~ /

~~.\_. \_/

\_/ \_/

\_/m/'

[ec2-user@ip-172-31-39-100 ~]$

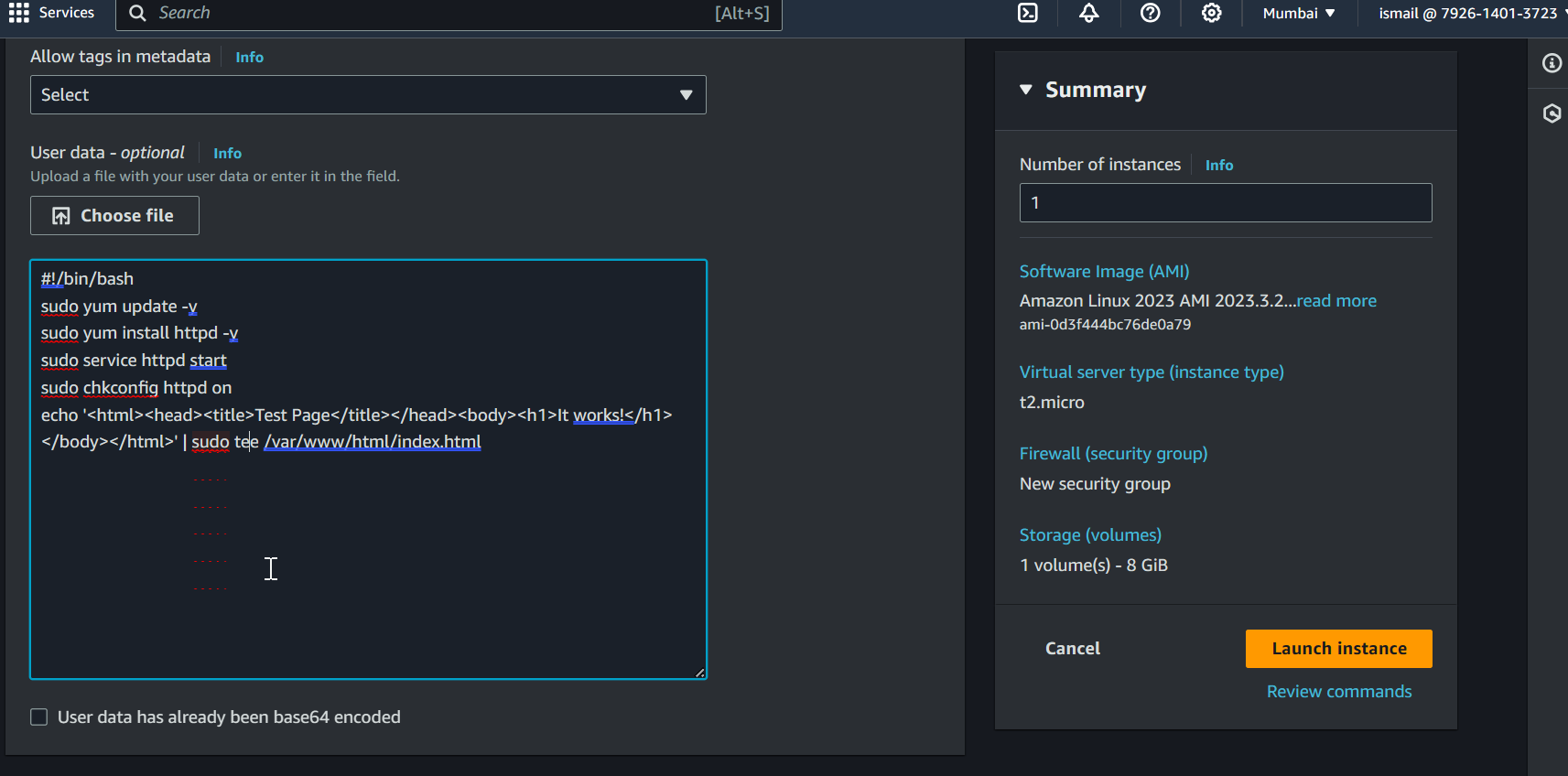
==================================================================================

4. Web Application Setup:

- Install a web server (e.g., Apache or Nginx) on the web server instance.

- Create a simple HTML page to confirm the web server is working.

- Test accessing the web server's public IP address in a web browser.



root@DESKTOP-SQGRBLG:/mnt/c/Users/Dell/Downloads# ssh -i L3.pem ec2-user@13.126.82.113

The authenticity of host '13.126.82.113 (13.126.82.113)' can't be established.

ED25519 key fingerprint is SHA256:VVE1hZ/5trgsFGmP0hDLYeB22kGwDdDRsSGdNB5XJYk.

This key is not known by any other names

Are you sure you want to continue connecting (yes/no/[fingerprint])? yes

Warning: Permanently added '13.126.82.113' (ED25519) to the list of known hosts.

, #\_

~\\_ ####\_ Amazon Linux 2023

~~ \\_#####\

~~ \###|

~~ \#/ \_\_\_ https://aws.amazon.com/linux/amazon-linux-2023

~~ V~' '->

~~~ /

~~.\_. \_/

\_/ \_/

\_/m/'

[ec2-user@ip-172-31-44-123 ~]$ sudo su

[root@ip-172-31-44-123 ec2-user]# cd /var/www/html/

[root@ip-172-31-44-123 html]# ll

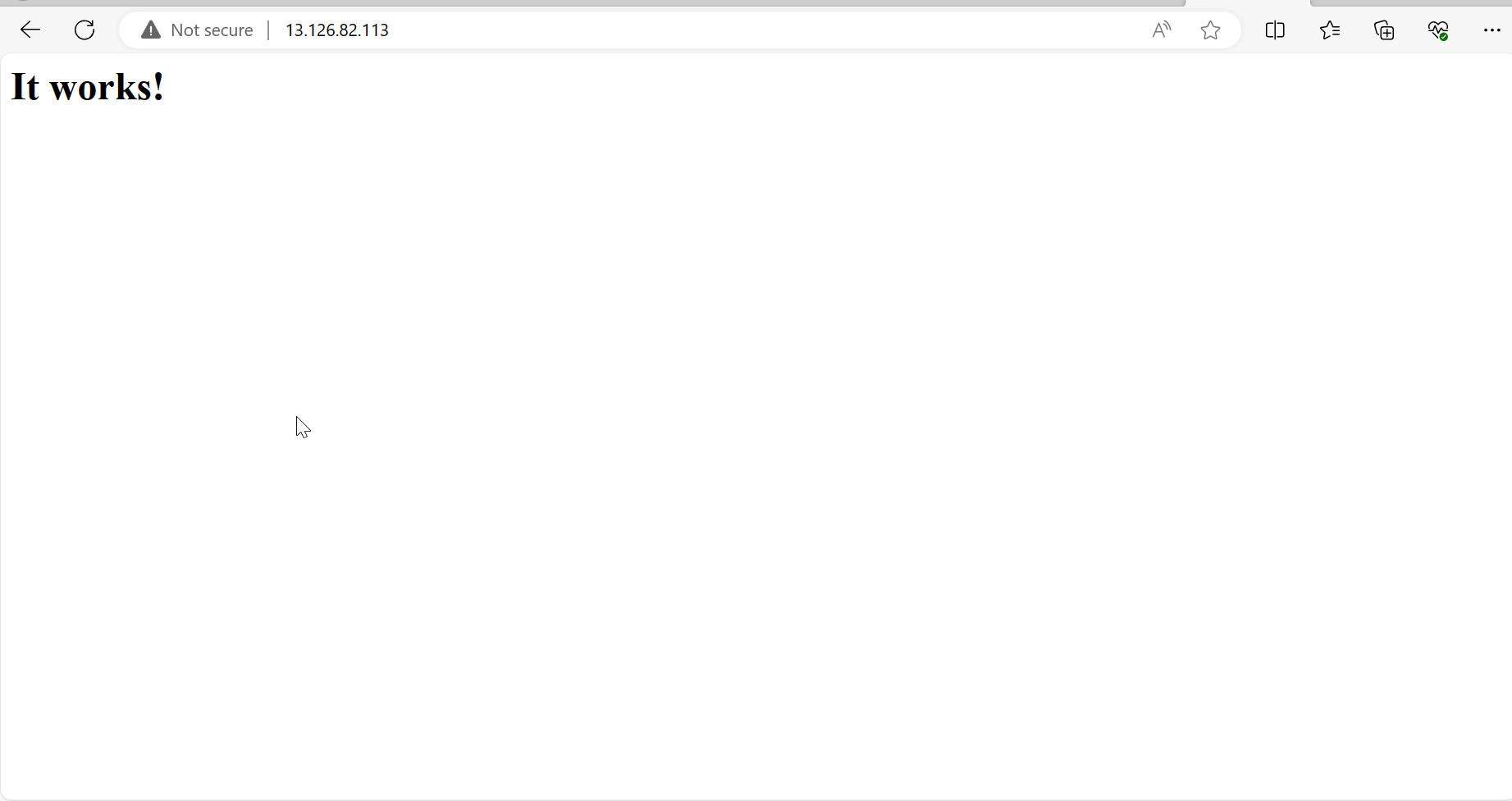
total 4

-rw-r--r--. 1 root root 82 Jan 16 11:31 index.html

[root@ip-172-31-44-123 html]# cat index.html

<html><head><title>Test Page</title></head><body><h1>It works!</h1></body></html>

[root@ip-172-31-44-123 html]#



==================================================================================

**==================================================================================**

(ON CLI)

1. Create Security Group for Web Server Using AWS CLI:

- Use the AWS CLI to create a security group for the web server.

- Configure inbound rules to allow HTTP traffic (port 80) and SSH traffic (port 22) from any source.

Sol:-

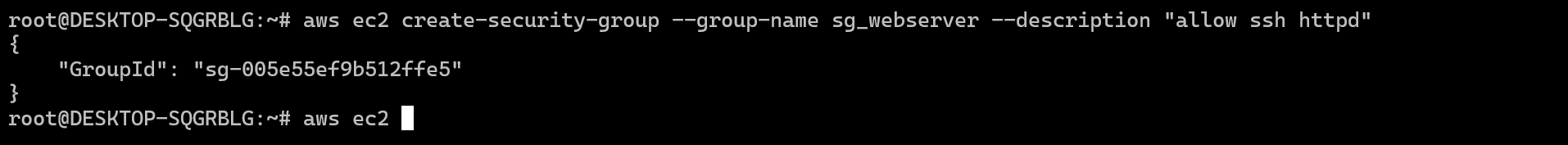
aws ec2 create-security-group --group-name sg\_webserver --description "allow ssh httpd"

{

"GroupId": "sg-005e55ef9b512ffe5"

}

root@DESKTOP-SQGRBLG:~#



root@DESKTOP-SQGRBLG:~# aws ec2 authorize-security-group-ingress --group-id "sg-0965cc6b8d9917da2" --protocol tcp --port 80 --cidr 0.0.0.0/0

{

"Return": true,

"SecurityGroupRules": [

{

"SecurityGroupRuleId": "sgr-08fb76940a54583d3",

"GroupId": "sg-0965cc6b8d9917da2",

"GroupOwnerId": "792614013723",

"IsEgress": false,

"IpProtocol": "tcp",

"FromPort": 80,

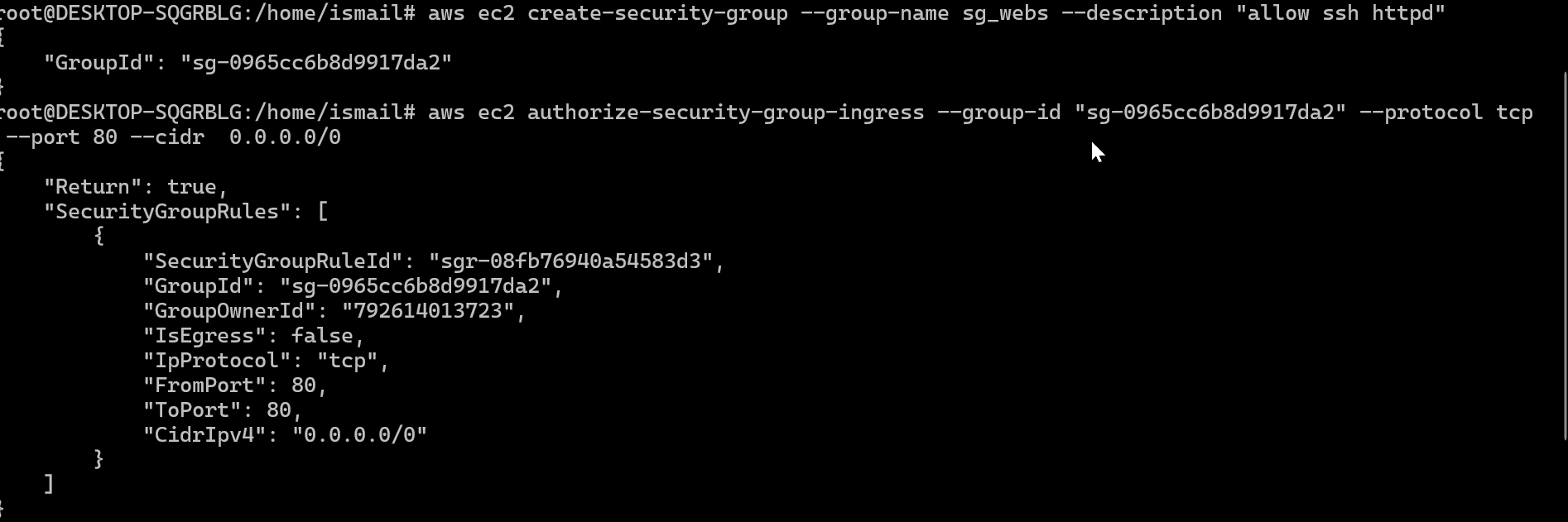
"ToPort": 80,

"CidrIpv4": "0.0.0.0/0"

}

]

}



root@DESKTOP-SQGRBLG:/home/ismail# aws ec2 authorize-security-group-ingress --group-id "sg-0965cc6b8d9917da2" --protocol tcp --port 22 --cidr 0.0.0.0/0

{

"Return": true,

"SecurityGroupRules": [

{

"SecurityGroupRuleId": "sgr-0fb42bf322cb47917",

"GroupId": "sg-0965cc6b8d9917da2",

"GroupOwnerId": "792614013723",

"IsEgress": false,

"IpProtocol": "tcp",

"FromPort": 22,

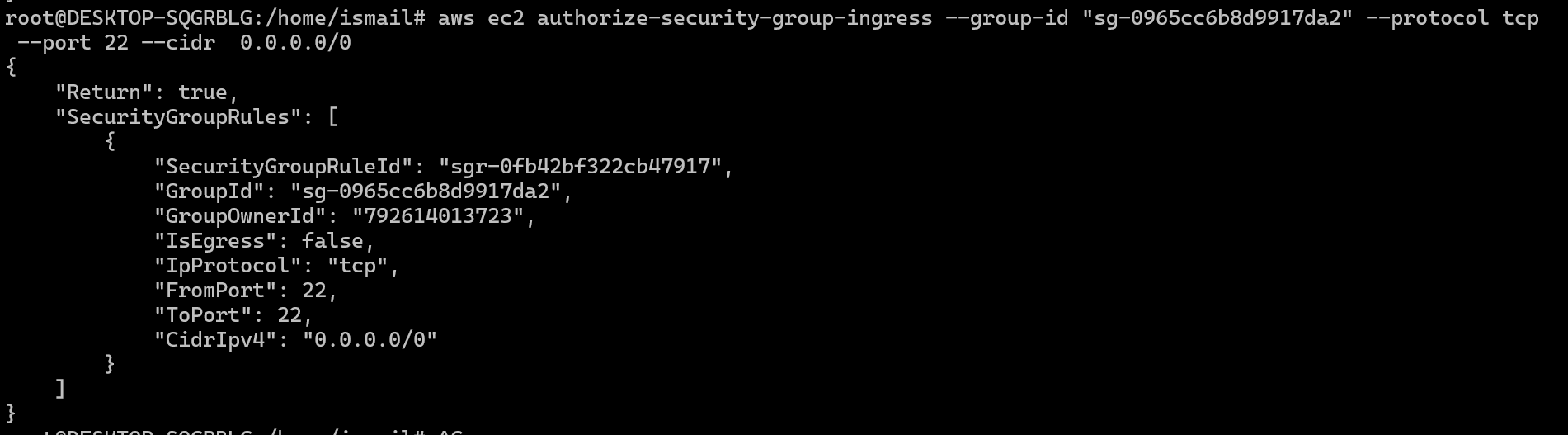
"ToPort": 22,

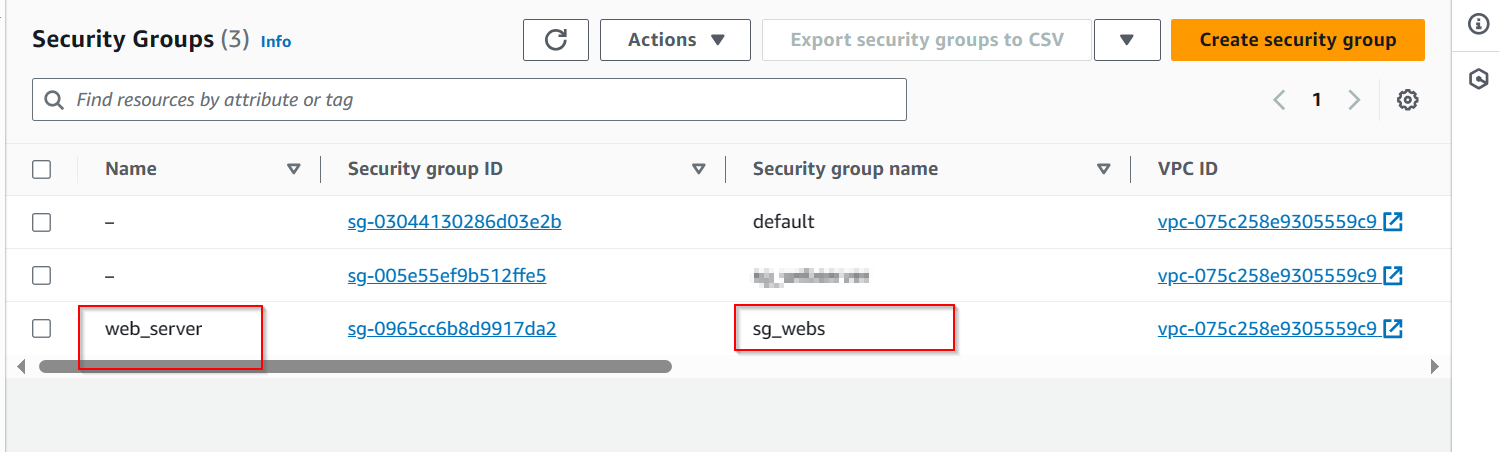
"CidrIpv4": "0.0.0.0/0"

}

]

}root@DESKTOP-SQGRBLG:~#



 ==================================================================================

2. Launch EC2 Instance for Web Server Using AWS CLI:

- Use the AWS CLI to launch an EC2 instance for the web server using Amazon Linux 2 AMI.

- Associate the security group created earlier with this instance.

- Use an appropriate instance type for a web server.

- Ensure the instance has a public IP address.

Sol:-

aws ec2 run-instances --image-id ami-xxxxxxxx --count 1 --instance-type t2.micro --key-name MyKeyPair --security-group-ids sg-903004f8 --subnet-id subnet-6e7f829e

root@DESKTOP-SQGRBLG:~# aws ec2 run-instances --image-id ami-0d3f444bc76de0a79 --count 1 --instance-type t2.micro --key-name Linux\_key --security-group-ids sg-0965cc6b8d9917da2 --associate-public-ip-address --tag-specifications 'ResourceType=instance

,Tags=[{Key=Name,Value=Web\_Instance}]'

{

"Groups": [],

"Instances": [

{

"AmiLaunchIndex": 0,

"ImageId": "ami-0d3f444bc76de0a79",

"InstanceId": "i-0f60b44d704face60",

"InstanceType": "t2.micro",

"KeyName": "Linux\_key",

"LaunchTime": "2024-01-19T08:03:14.000Z",

"Monitoring": {

"State": "disabled"

},

"Placement": {

"AvailabilityZone": "ap-south-1a",

"GroupName": "",

"Tenancy": "default"

},

"PrivateDnsName": "ip-172-31-34-139.ap-south-1.compute.internal",

"PrivateIpAddress": "172.31.34.139",

"ProductCodes": [],

"PublicDnsName": "",

"State": {

"Code": 0,

"Name": "pending"

},

"StateTransitionReason": "",

"SubnetId": "subnet-05742bb4459741927",

"VpcId": "vpc-075c258e9305559c9",

"Architecture": "x86\_64",

"BlockDeviceMappings": [],

"ClientToken": "b84bed53-7ad6-4070-9adf-95b62661abdc",

"EbsOptimized": false,

"EnaSupport": true,

"Hypervisor": "xen",

"NetworkInterfaces": [

{

"Attachment": {

"AttachTime": "2024-01-19T08:03:14.000Z",

"AttachmentId": "eni-attach-00fb8235798cb5776",

"DeleteOnTermination": true,

"DeviceIndex": 0,

"Status": "attaching",

"NetworkCardIndex": 0

},

"Description": "",

"Groups": [

{

"GroupName": "sg\_webs",

"GroupId": "sg-0965cc6b8d9917da2"

}

],

"Ipv6Addresses": [],

"MacAddress": "02:c0:a9:7a:a1:0d",

"NetworkInterfaceId": "eni-0b39df6bd085fbdff",

"OwnerId": "792614013723",

"PrivateDnsName": "ip-172-31-34-139.ap-south-1.compute.internal",

"PrivateIpAddress": "172.31.34.139",

"PrivateIpAddresses": [

{

"Primary": true,

"PrivateDnsName": "ip-172-31-34-139.ap-south-1.compute.internal",

"PrivateIpAddress": "172.31.34.139"

}

],

"SourceDestCheck": true,

"Status": "in-use",

"SubnetId": "subnet-05742bb4459741927",

"VpcId": "vpc-075c258e9305559c9",

"InterfaceType": "interface"

}

],

"RootDeviceName": "/dev/xvda",

"RootDeviceType": "ebs",

"SecurityGroups": [

{

"GroupName": "sg\_webs",

"GroupId": "sg-0965cc6b8d9917da2"

}

],

"SourceDestCheck": true,

"StateReason": {

"Code": "pending",

"Message": "pending"

},

"Tags": [

{

"Key": "Name",

"Value": "Web\_Instance"

}

],

"VirtualizationType": "hvm",

"CpuOptions": {

"CoreCount": 1,

"ThreadsPerCore": 1

},

"CapacityReservationSpecification": {

"CapacityReservationPreference": "open"

},

"MetadataOptions": {

"State": "pending",

"HttpTokens": "required",

"HttpPutResponseHopLimit": 2,

"HttpEndpoint": "enabled",

"HttpProtocolIpv6": "disabled",

"InstanceMetadataTags": "disabled"

},

"EnclaveOptions": {

"Enabled": false

},

"BootMode": "uefi-preferred",

"PrivateDnsNameOptions": {

"HostnameType": "ip-name",

"EnableResourceNameDnsARecord": false,

"EnableResourceNameDnsAAAARecord": false

}

}

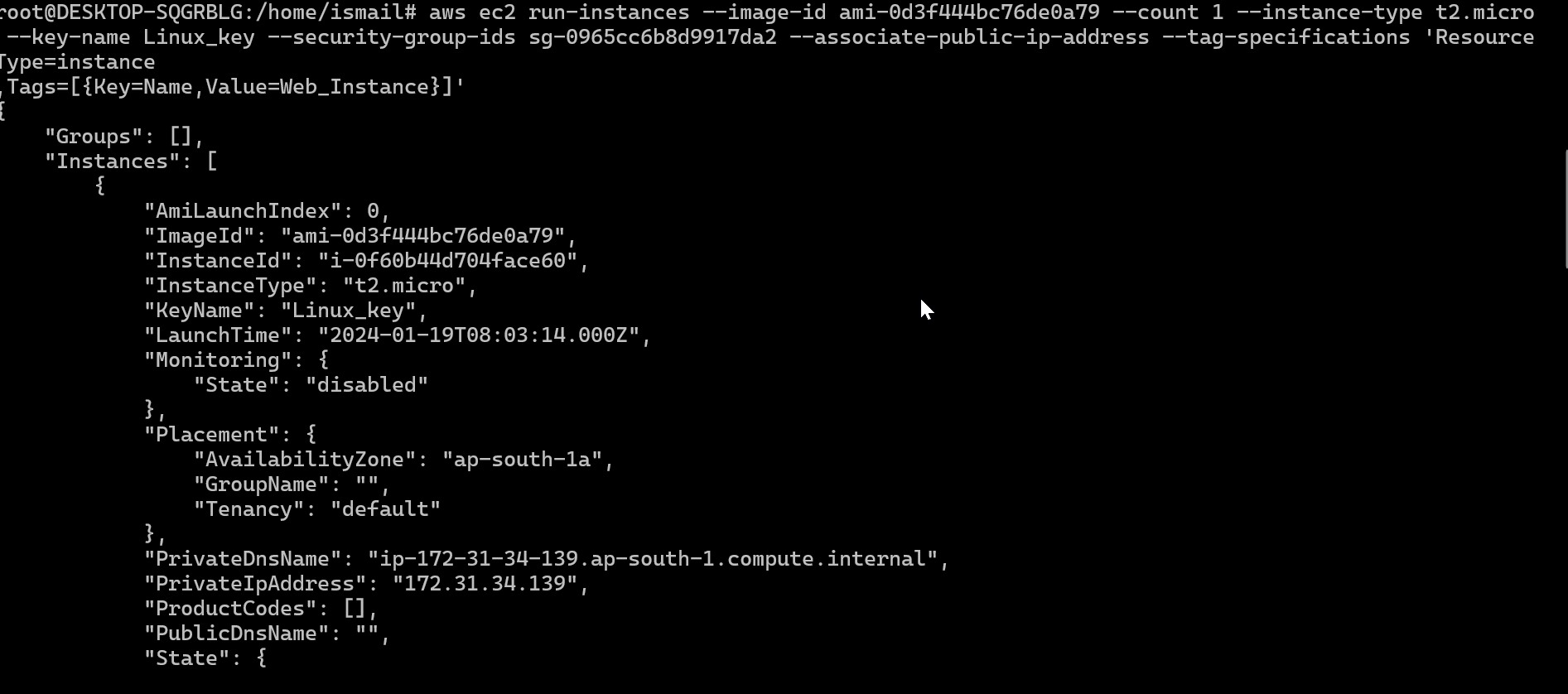
],

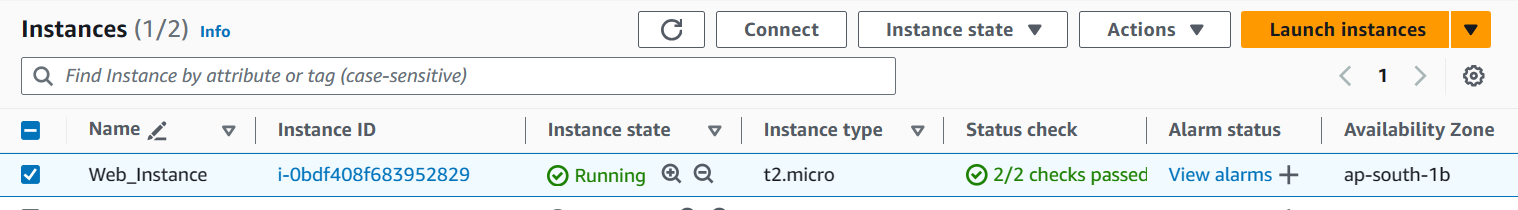
"OwnerId": "792614013723",

"ReservationId": "r-0294615e6c29ecebb"

}

root@DESKTOP-SQGRBLG:/home/ismail#





==================================================================================

3. SSH Access Using AWS CLI:

- Use the AWS CLI to generate an SSH key pair for secure access to the web server instance.

- Configure the web server instance to accept SSH connections using the generated key pair.

- Use the AWS CLI to attempt to SSH into the web server instance to verify successful access.

:/home/ismail# aws ec2 describe-instances --instance-ids i-02d1549ad54ea14c2 --query 'Reservations[0].Instances[0].PublicIpAddress' --output text

43.204.149.135

root@DESKTOP-SQGRBLG:/home/ismail# ssh -i w\_key.pem ec2-user@43.204.149.135

The authenticity of host '43.204.149.135 (43.204.149.135)' can't be established.

ED25519 key fingerprint is SHA256:PwhuUZkVwOgVgr8UUayea3MDKN0rnQlb8Tw91Ihw1Z0.

This key is not known by any other names

Are you sure you want to continue connecting (yes/no/[fingerprint])? yes

Warning: Permanently added '43.204.149.135' (ED25519) to the list of known hosts.

A newer release of "Amazon Linux" is available.

Version 2023.3.20240117:

Run "/usr/bin/dnf check-release-update" for full release and version update info

, #\_

~\\_ ####\_ Amazon Linux 2023

~~ \\_#####\

~~ \###|

~~ \#/ \_\_\_ https://aws.amazon.com/linux/amazon-linux-2023

~~ V~' '->

~~~ /

~~.\_. \_/

\_/ \_/

\_/m/'

[ec2-user@ip-172-31-9-151 ~]$

==================================================================================

4. Web Application Setup Using AWS CLI:

- Use the AWS CLI to install a web server (e.g., Apache or Nginx) on the web server instance.

- Create a simple HTML page using the AWS CLI to confirm the web server is working.

- Use the AWS CLI to test accessing the web server's public IP address in a web browser.

root@DESKTOP-SQGRBLG:/home/ismail# vim user\_data.sh

root@DESKTOP-SQGRBLG:/home/ismail# cat user\_data.sh

**#!/bin/bash**

**sudo yum update -y**

**sudo yum install httpd -y**

**sudo service httpd start**

**sudo chkconfig httpd on**

**echo '<html><head><title>Welcome Page</title></head><body><h1>Welcome to CloudEthix!</h1></body></html>' | sudo tee /var/www/html/index.html**

root@DESKTOP-SQGRBLG:/home/ismail#

root@DESKTOP-SQGRBLG:/home/ismail# aws ec2 run-instances --image-id ami-0d3f444bc76de0a79 --key-name w\_key --instance-type t

2.micro --security-group-ids sg-094483fe78e78813c --associate-public-ip-address --tag-specifications 'ResourceType=instance,Tags=[{Key=Name,Value=Ec2\_Instance}]' --user-data file://user\_data.sh

{

"Groups": [],

"Instances": [

{

"AmiLaunchIndex": 0,

"ImageId": "ami-0d3f444bc76de0a79",

"InstanceId": "i-0c298d969649c8228",

"InstanceType": "t2.micro",

"KeyName": "w\_key",

"LaunchTime": "2024-01-19T10:44:55.000Z",

"Monitoring": {

"State": "disabled"

},

"Placement": {

"AvailabilityZone": "ap-south-1b",

"GroupName": "",

"Tenancy": "default"

},

"PrivateDnsName": "ip-172-31-12-213.ap-south-1.compute.internal",

"PrivateIpAddress": "172.31.12.213",

"ProductCodes": [],

"PublicDnsName": "",

"State": {

"Code": 0,

"Name": "pending"

},

"StateTransitionReason": "",

"SubnetId": "subnet-08d0556b25ce1ccb8",

"VpcId": "vpc-075c258e9305559c9",

"Architecture": "x86\_64",

"BlockDeviceMappings": [],

"ClientToken": "6acb96f2-e1a3-46de-8cb9-8398eddd8829",

"EbsOptimized": false,

"EnaSupport": true,

"Hypervisor": "xen",

"NetworkInterfaces": [

{

"Attachment": {

"AttachTime": "2024-01-19T10:44:55.000Z",

"AttachmentId": "eni-attach-019bbc20c7e5a2f31",

"DeleteOnTermination": true,

"DeviceIndex": 0,

"Status": "attaching",

"NetworkCardIndex": 0

},

"Description": "",

"Groups": [

{

"GroupName": "sg\_web\_server",

"GroupId": "sg-094483fe78e78813c"

}

],

"Ipv6Addresses": [],

"MacAddress": "0a:52:fd:93:47:85",

"NetworkInterfaceId": "eni-073fca07b66ea1357",

"OwnerId": "792614013723",

"PrivateDnsName": "ip-172-31-12-213.ap-south-1.compute.internal",

"PrivateIpAddress": "172.31.12.213",

"PrivateIpAddresses": [

{

"Primary": true,

"PrivateDnsName": "ip-172-31-12-213.ap-south-1.compute.internal",

"PrivateIpAddress": "172.31.12.213"

}

],

"SourceDestCheck": true,

"Status": "in-use",

"SubnetId": "subnet-08d0556b25ce1ccb8",

"VpcId": "vpc-075c258e9305559c9",

"InterfaceType": "interface"

}

],

"RootDeviceName": "/dev/xvda",

"RootDeviceType": "ebs",

"SecurityGroups": [

{

"GroupName": "sg\_web\_server",

"GroupId": "sg-094483fe78e78813c"

}

],

"SourceDestCheck": true,

"StateReason": {

"Code": "pending",

"Message": "pending"

},

"Tags": [

{

"Key": "Name",

"Value": "Ec2\_Instance"

}

],

"VirtualizationType": "hvm",

"CpuOptions": {

"CoreCount": 1,

"ThreadsPerCore": 1

},

"CapacityReservationSpecification": {

"CapacityReservationPreference": "open"

},

"MetadataOptions": {

"State": "pending",

"HttpTokens": "required",

"HttpPutResponseHopLimit": 2,

"HttpEndpoint": "enabled",

"HttpProtocolIpv6": "disabled",

"InstanceMetadataTags": "disabled"

},

"EnclaveOptions": {

"Enabled": false

},

"BootMode": "uefi-preferred",

"PrivateDnsNameOptions": {

"HostnameType": "ip-name",

"EnableResourceNameDnsARecord": false,

"EnableResourceNameDnsAAAARecord": false

}

}

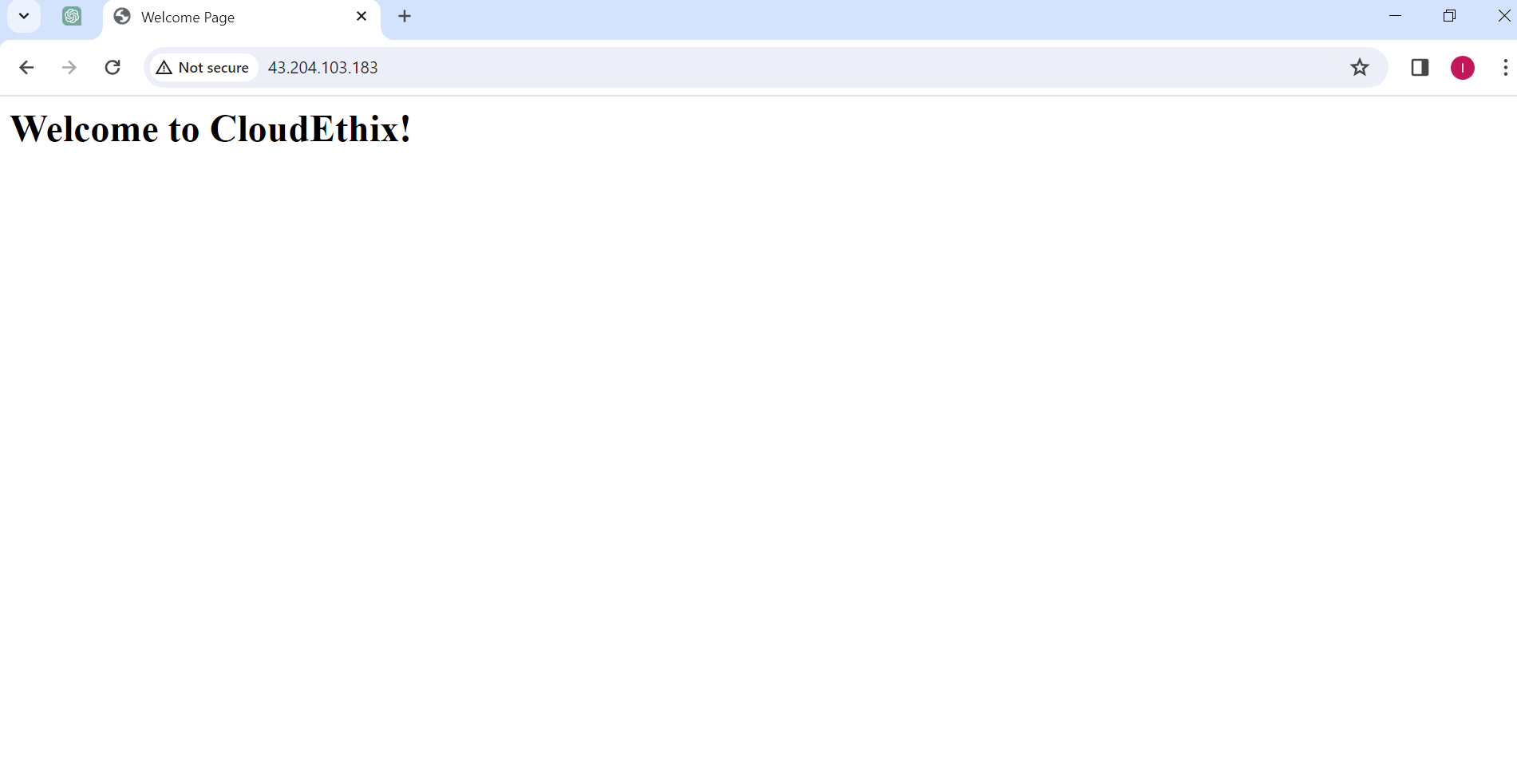
],

"OwnerId": "792614013723",

"ReservationId": "r-048e730f38e0ac232"

}

root@DESKTOP-SQGRBLG:/home/ismail#



==================================================================================

5. Documentation:

- Provide clear documentation in a text file outlining the AWS CLI commands used for each task along with their outputs.

- Include any relevant information such as IP addresses, instance IDs, etc.

==================================================================================