

ASSIGNMENT – 10

Problem definition: Deploy a project from GitHub to EC2 by creating a new security group and user Data.

Solution:

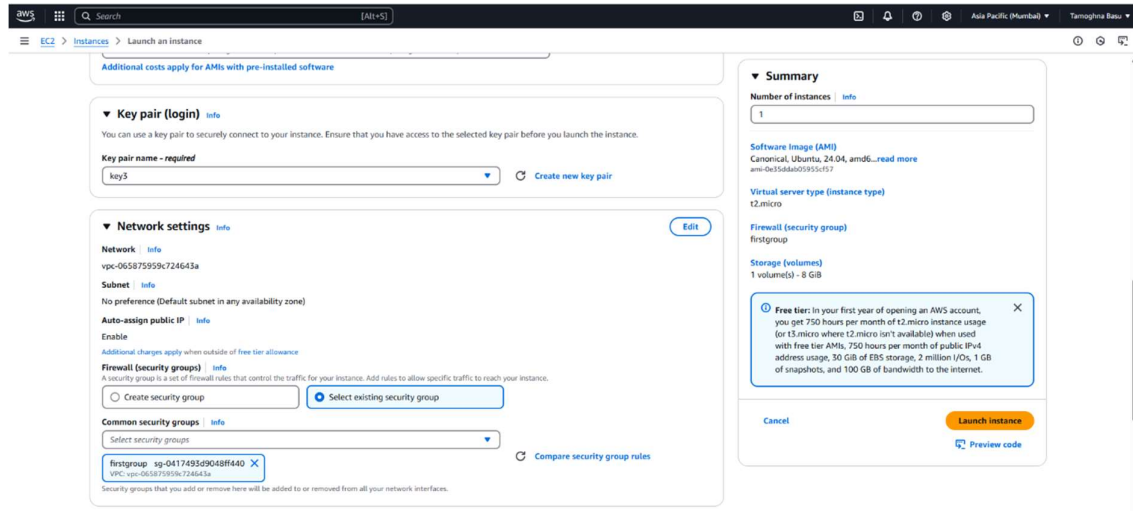
Step1: From aws console we select EC2 ,from EC2 network & security select security groups then select create security group.

Step2: :Next enter the security group name ,description and in inbound rule add rule type custom TCP ,port range 4000, source anywhere and 0.0.0.0/0 .

Type	Protocol	Port range	Source	Description - optional
Custom TCP	TCP	4000	Anywhere...	0.0.0.0/0
SSH	TCP	22	Anywhere...	0.0.0.0/0
HTTP	TCP	80	Anywhere...	0.0.0.0/0
HTTPS	TCP	443	Anywhere...	0.0.0.0/0

Step3: After creating the security group we will launch an instance on EC2 using the same security group.

Step4: To launch new instance we give instance name, select ubuntu, select key pair and in network setting's firewall part check select existing security group. From the dropdown select newly created security group.



Step 5: In advanced settings option's user data part add some command. Then click launch instance

```
#!/bin/bash
```

```
apt-get update
```

```
apt-get install -y nginx
```

```
systemctl start nginx
```

```
systemctl enable nginx
```

```
apt-get install -y git
```

```
curl -SL https://deb.nodesource.com/setup\_16.x | sudo -E
```

```
bash -
```

```
apt-get install -y nodejs
```

```
git clone http://github.com/sudip7407/Repo1.git
```

```
cd Repo1
```

```
npm install
```

```
node index.js
```

V2 only (token required)

⚠ For V2 requests, you must include a session token in all Instance metadata requests. Applications or agents that use V1 for instance metadata access will break.

Metadata response hop limit [Info](#)
2

Allow tags in metadata [Info](#)
Select

User data - optional [Info](#)
Upload a file with your user data or enter it in the field.
[Choose file](#)

```
#!/bin/bash
apt-get update
apt-get install -y nginx
systemctl start nginx
systemctl enable nginx
apt-get install -y git
curl -SL https://deb.nodesource.com/setup_16.x | sudo -E bash -
apt-get install -y nodejs
git clone https://github.com/sudip7407/Repo1.git
cd Repo1
npm install
node index.js
```

☐ User data has already been base64 encoded

Summary

Number of instances [Info](#)
1

Software Image (AMI)
Canonical, Ubuntu, 24.04, amd64...[read more](#)
ami-0e355ab05955c457

Virtual server type (Instance type)
t2.micro

Firewall (security group)
firstgroup

Storage (volumes)
1 volume(s) - 8 GiB

Free tier: In your first year of opening an AWS account, you get 750 hours per month of t2.micro instance usage (or t3.micro where t2.micro isn't available) when used with free tier AMIs, 750 hours per month of public IPv4 address usage, 30 GiB of EBS storage, 2 million I/Os, 1 GB of snapshots, and 100 GB of bandwidth to the internet.

[Cancel](#) [Launch instance](#) [Preview code](#)

Step6: After that from instance copy public ip address, which is 13.200.215.24

Instances (1/1) [Info](#)

Last updated less than a minute ago [Connect](#) [Instance state](#) [Actions](#) [Launch instances](#)

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...	Elastic IP
myawsserver	i-06f33d2670250ea60	Running	t2.micro	Initializing	View alarms +	ap-south-1b	ec2-3-109-155-34.ap-s...	3.109.155.34	-

i-06f33d2670250ea60 (myawsserver)

[Details](#) [Status and alarms](#) [Monitoring](#) [Security](#) [Networking](#) [Storage](#) [Tags](#)

Instance summary [Info](#)

<p>Instance ID i-06f33d2670250ea60</p> <p>IPv6 address -</p> <p>Hostname type IP name: ip-172-31-9-234.ap-south-1.compute.internal</p> <p>Answer private resource DNS name IPv4 (A)</p>	<p>Public IPv4 address 3.109.155.34 open address</p> <p>Instance state Running</p> <p>Private IP DNS name (IPv4 only) ip-172-31-9-234.ap-south-1.compute.internal</p> <p>Instance type t2.micro</p>	<p>Private IPv4 addresses 172.31.9.234</p> <p>Public IPv4 DNS ec2-3-109-155-34.ap-south-1.compute.amazonaws.com open address</p> <p>Elastic IP addresses -</p>
---	---	--

Step7: At last we paste the ip address with port number on browser.



Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

Thank you for using nginx.

Hello World