

# Assignment 10

## Deploy a project from github to EC2 by creating new security group and user data.

Objective: To deploy a Node.js project from GitHub to an EC2 instance using a custom security group and user data script during launch.

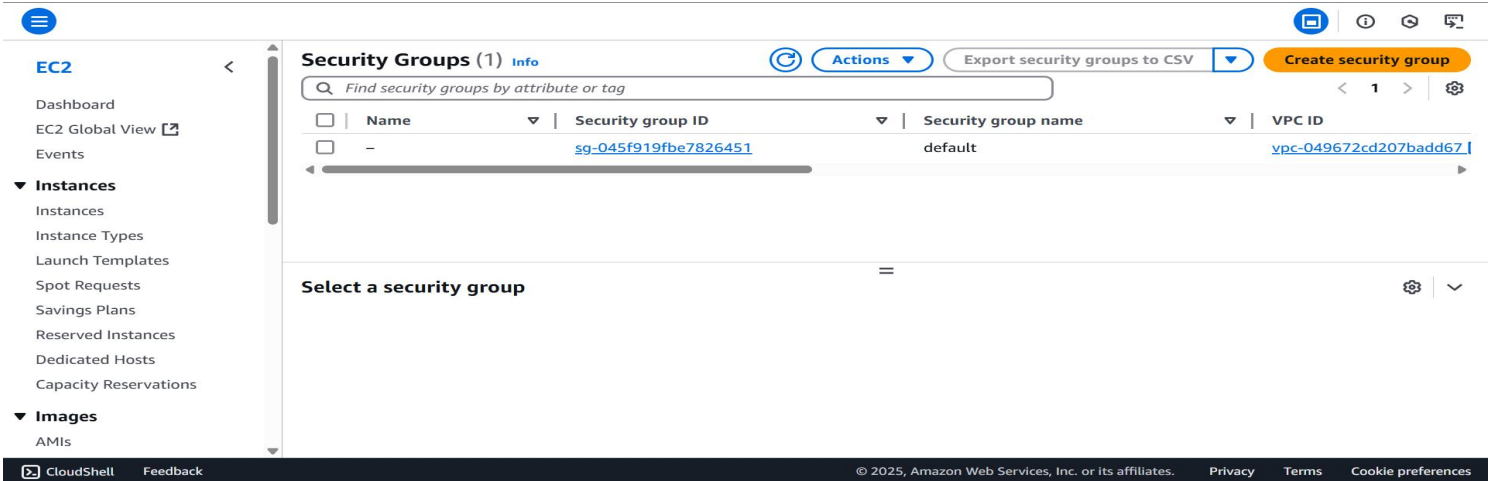
### Part 1: Create a New Security Group

#### ◆ Step 1: Open EC2 Dashboard

- Log in to the AWS Management Console.
- Navigate to **EC2 Dashboard** (under the “Services” menu).

#### ◆ Step 2: Delete Existing (Non-default) Security Groups

- Go to **Network & Security > Security Groups** in the EC2 menu.
- Select any **non-default** security groups.
- Click **Actions > Delete Security Groups** (You cannot delete the default one).



#### Step 3: Create a New Security Group

- Click on “Create security group”.
- Fill in the following:
  - **Security group name:** SnehaSecurityGroup
  - **Description:** A brief description (e.g., Security group for Node.js app)
  - **VPC:** Leave it as the default.

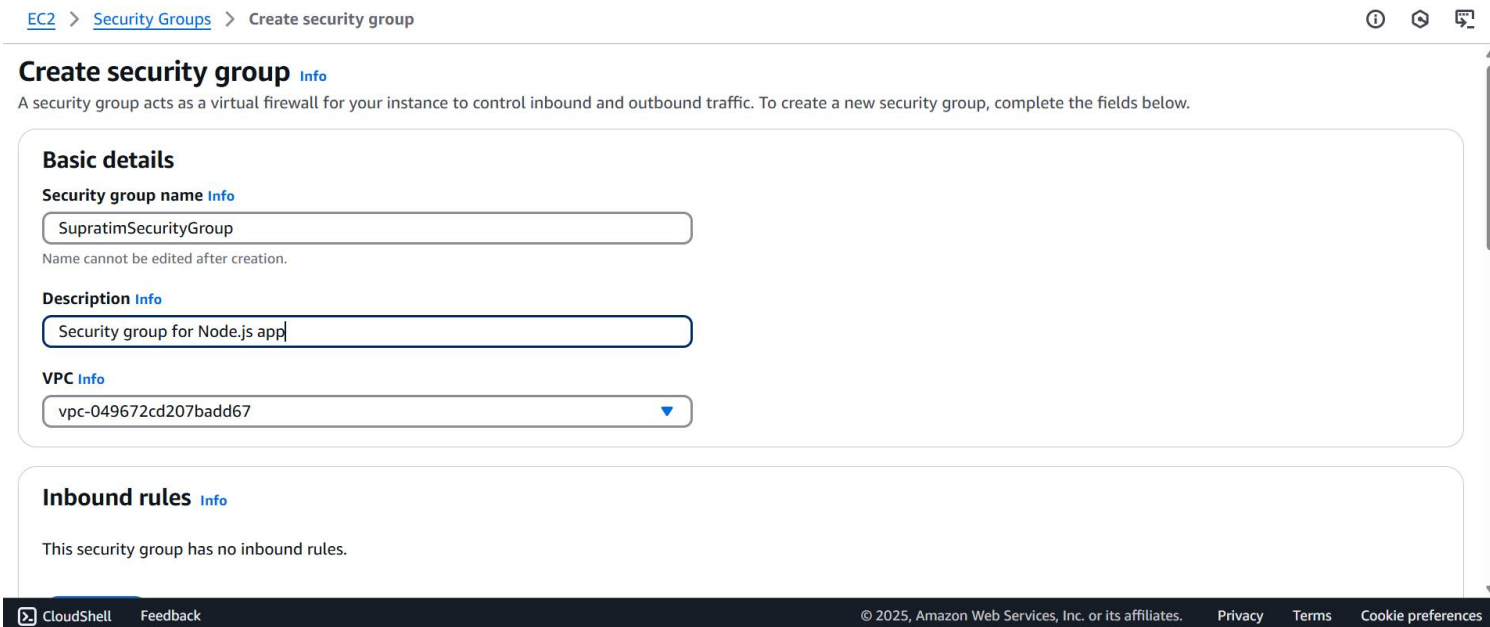
#### Step 4: Add Inbound Rules

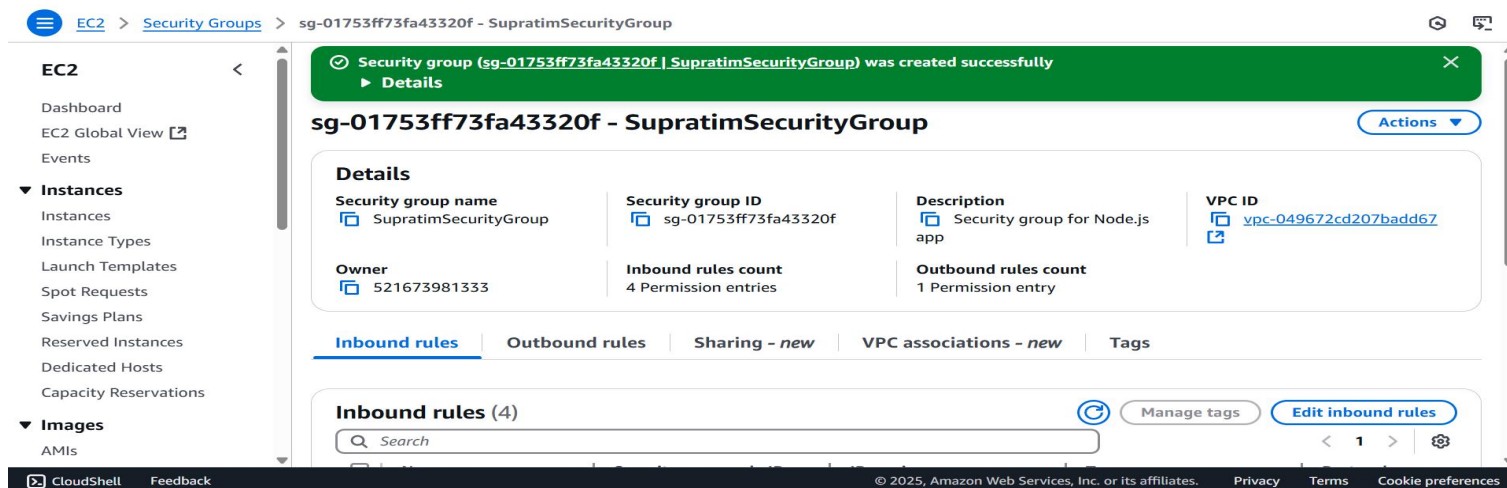
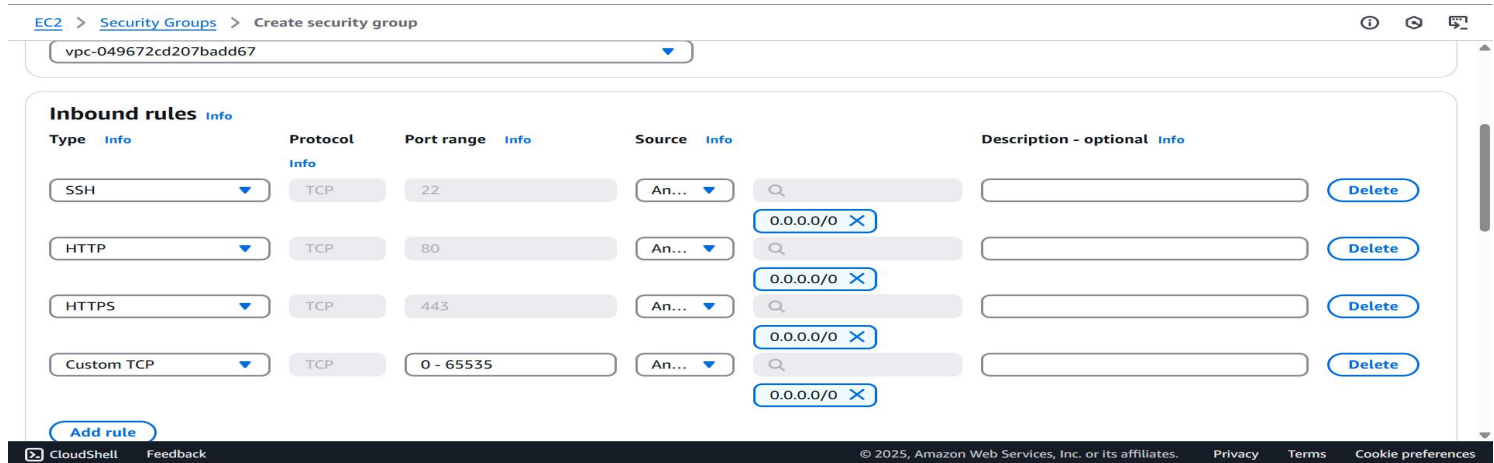
Click “Add Rule” and input the following:

⚠ **Note:** Be cautious using 0.0.0.0/0 as it allows access from anywhere. For production, restrict this.

#### ◆ Step 5: Create Security Group

- Click “Create security group” to save it.





## Part 2: Launch EC2 Instance and Deploy App

### ◆ Step 1: Launch New Instance

- Go to **EC2 Dashboard > Instances > Launch Instance**.

### ◆ Step 2: Instance Configuration

- **Name:** myinstance21
- **Application and OS Image (AMI):** Choose **Ubuntu (Free tier eligible)**.
- **Instance type:** t2.micro

### ◆ Step 3: Key Pair

- Under **Key pair (login)**, choose your existing key pair (snehaa1234) or:
  - Click **Create new key pair**
  - Download the .pem file for SSH access.

### ◆ Step 4: Network Settings

- Click **Edit** in the Network settings section.
- Choose **"Select existing security group"**
- Select SnehaSecurityGroup created earlier.

### ◆ Step 5: Configure User Data (Auto-deploy app)

Scroll to **Advanced Details > User data**, and paste the following script:

Replace the GitHub repo path with your actual repository

e.g., <https://github.com/itsmesneha/SNEHAREPO>

```
#!/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_18.x | sudo -E bash -
```

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

```
git clone https://github.com/snsupratim/college-aws.git
```

```
cd college-aws
```

```
npm install
```

```
node index.js
```

EC2 > Instances > Launch an instance

VPC - required | Info

vpc-049672cd207badd67  
172.31.0.0/16 (default)

Subnet | Info

No preference

Auto-assign public IP | Info

Enable

Additional charges apply when outside of free tier allowance

Firewall (security groups) | Info

A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

☐ Create security group

☒ Select existing security group

Common security groups | Info

Select security groups

SupratimSecurityGroup sg-01753ff73fa43320f

Security groups that you add or remove here will be added to or removed from all your network interfaces.

Compare security group rules

Summary

Number of instances | Info

1

Software Image (AMI)

Canonical, Ubuntu, 24.04, amd6...read more  
ami-01938df366ac2d954

Virtual server type (instance type)

t2.micro

Firewall (security group)

SupratimSecurityGroup

Storage (volumes)

Cancel

Launch instance

Preview code

CloudShell Feedback

© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

EC2 > Instances > Launch an instance

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\_18.x | sudo -E bash -  
apt-get install -y nodejs  
git clone https://github.com/snsupratim/college-aws.git  
cd college-aws  
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, amd6...read more  
ami-01938df366ac2d954

Virtual server type (instance type)

t2.micro

Firewall (security group)

SupratimSecurityGroup

Storage (volumes)

Cancel

Launch instance

Preview code

CloudShell Feedback

© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

- ◆ **Step 6: Launch Instance**
- Click **Launch instance** and wait until it is in the **running state**.

## 🌐 Part 3: Test the Deployment

- ◆ **Step 1: Open Instance Summary**
  - Go to **Instances**, click on your newly created instance name.
  - ◆ **Step 2: Get Public IPv4**
  - Copy the **Public IPv4 address** from the summary panel.
  - ◆ **Step 3: Access App via Browser**
  - Paste the address into your browser (e.g., `http://<your-ip-address>`)
  - If your app runs on a port (e.g., 3000), try `http://<your-ip>:3000`
- You should see your deployed application running!

EC2 > Instances > i-059df6bf0bbc521ee

EC2

Dashboard  
EC2 Global View  
Events

Instances

Instances  
Instance Types  
Launch Templates  
Spot Requests  
Savings Plans  
Reserved Instances  
Dedicated Hosts  
Capacity Reservations

Images

AMIs

Instance summary for i-059df6bf0bbc521ee (myserver) | Info

Connect

Instance state

Actions

Updated less than a minute ago

Instance ID

i-059df6bf0bbc521ee

IPv6 address

-

Hostname type

IP name: ip-172-31-31-138.ap-southeast-1.compute.internal

Answer private resource DNS name

IPv4 (A)

Public IPv4 address

54.169.133.230 | open address

Instance state

Running

Private IP DNS name (IPv4 only)

ip-172-31-31-138.ap-southeast-1.compute.internal

Instance type

t2.micro

Private IPv4 addresses

172.31.31.138

Public IPv4 DNS

ec2-54-169-133-230.ap-southeast-1.compute.amazonaws.com | open address

Elastic IP addresses

-

CloudShell Feedback

© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

## 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](https://nginx.org). Commercial support is available at [nginx.com](https://nginx.com).

Thank you for using nginx.