

Assignment : 10

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

Step 1: First we go to EC2 Instance then we navigate to "Security Groups" under Network & Security.

The screenshot shows the AWS Management Console with the 'Instances' page selected. A table lists two instances. The second instance, 'New_inst' with ID 'i-03f502a9b089070a2', is in a 'Running' state. Below the table, the details for this instance are displayed, including its public IPv4 address (13.203.210.17) and its running status.

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IP
New_inst	i-071db21ac3cc7d31b	Terminated	t2.micro	-	View alarms +	ap-south-1b	-
New_inst	i-03f502a9b089070a2	Running	t2.micro	2/2 checks pass	View alarms +	ap-south-1b	ec2-13-2

i-03f502a9b089070a2 (New_inst)

Instance summary

Instance ID i-03f502a9b089070a2	Public IPv4 address 13.203.210.17 open address	Private IPv4 addresses 172.31.5.191
IPv6 address -	Instance state Running	Public IPv4 DNS ec2-13-203-210-17.ap-south-1.compute.amazonaws.com open address

Step 2: Then we create security group

The screenshot shows the AWS Management Console with the 'Security Groups' page selected. A table lists two security groups. The first one, 'launch-wizard-2' with ID 'sg-0f2c72025cf1ba5b6', is selected. Below the table, the details for this security group are displayed, including its inbound and outbound rules.

Name	Security group ID	Security group name	VPC ID	Description
-	sg-0f2c72025cf1ba5b6	launch-wizard-2	vpc-0e2181ce3777a313c	launch-wizard-2
-	sg-034eb43712646092d	default	vpc-0e2181ce3777a313c	default VPC security group

sg-0f2c72025cf1ba5b6 - launch-wizard-2

Details

Inbound rules: SSH (TCP, port 22, source 0.0.0.0/0), HTTP (TCP, port 80, source 0.0.0.0/0).

Step 3: In Create Security group we name it as we like then give some description.

The screenshot shows the 'Create security group' page in the AWS Management Console. The 'Basic details' section is filled out with 'Secure' as the security group name and 'new security rule' as the description. The VPC ID is set to 'vpc-0e2181ce3777a313c'. The 'Inbound rules' section shows two rules: SSH (TCP, port 22, source 0.0.0.0/0) and HTTP (TCP, port 80, source 0.0.0.0/0).

Create security group

A security group acts as a virtual firewall for your instance to control inbound and outbound traffic. To create a new security group, complete the fields below.

Basic details

Security group name: Secure

Description: new security rule

VPC: vpc-0e2181ce3777a313c

Inbound rules

Type	Protocol	Port range	Source	Description - optional
SSH	TCP	22	Anywhere - IPv4	0.0.0.0/0
HTTP	TCP	80	Anywhere - IPv4	0.0.0.0/0

Step 4: Now in Inbound rules we add rule SSH,HTTP,HTTPS,Custom TCP, where for port range we keep it default for SSH,HTTP,HTTPS and for Custom TCP we set it to 4000. We set source to "Anywhere-IPV4" 0.0.0.0/0.

CloudShellFeedback

aws

Search

[Alt+S]

🔍🔔🔄⚙️

Asia Pacific (Mumbai)

EC2

Security Groups

sg-0823bda1649d0b69b - Secure

EC2

Dashboard

EC2 Global View

Events

Instances

Instance Types

Launch Templates

Spot Requests

Savings Plans

Reserved Instances

Dedicated Hosts

Capacity Reservations

Images

AMIs

AMI Catalog

Elastic Block Store

Volumes

Snapshots

Security group (sg-0823bda1649d0b69b | Secure) was created successfully

Details

sg-0823bda1649d0b69b - Secure

Actions

Details

Security group name

Secure

Security group ID

sg-0823bda1649d0b69b

Description

new security rule

VPC ID

vpc-0e2181ce3777a313c

Owner

586794457897

Inbound rules count

4 Permission entries

Outbound rules count

1 Permission entry

Inbound rules

Outbound rules

Sharing - new

VPC associations - new

Tags

Inbound rules (4)

Manage tags

Edit inbound rules

Search

<input type="checkbox"/>	Name	Security group rule ID	IP version	Type	Protocol	Port range
<input type="checkbox"/>	-	sgr-0141ae12f2aa44869	IPv4	SSH	TCP	22

Step 6: Now we create a EC2 instance

Launch an instance

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

Name: New1

Application and OS Images (Amazon Machine Image)

Search our full catalog including 1000s of application and OS images

Recents: Quick Start

Amazon Linux, macOS, Ubuntu, Windows, Red Hat, SUSE Linux, Debian

Summary

Number of instances: 1

Software Image (AMI): Amazon Linux 2023 AMI 2023.6.2...read more

Virtual server type (instance type): t2.micro

Firewall (security group): New security group

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)

Launch instance

Step 7: In Firewall(security group) we select our existing rule that we made earlier.

Launch an instance

Auto-assign public IP

Enable

Firewall (security groups)

Create security group | Select existing security group

Common security groups

Select security groups

default VPC: vpc-0e2181ce3777a313c sg-034eb43712646092d

Secure VPC: vpc-0e2181ce3777a313c sg-0823bda1649d0b69b

launch-wizard-2 VPC: vpc-0e2181ce3777a313c sg-0f2c72025cf1ba5b6

Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage

Add new volume

Summary

Number of instances: 1

Software Image (AMI): Canonical, Ubuntu, 24.04, amd64...read more

Virtual server type (instance type): t2.micro

Firewall (security group): -

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)

Launch instance

Step 8: Now we scroll down to the advance details and then in user data we write the commands

```
#!/bin/bash
apt-get update
apt-get upgrade
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"
And launch the instance.
```

Metadata response hop limit: 2

Allow tags in metadata: Select

User data - optional: Upload a file with your user data or enter it in the field.

```
#!/bin/bash
apt-get update
apt-get upgrade
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
```

Summary:

- Number of instances: 1
- Software Image (AMI): Canonical, Ubuntu, 24.04, amd64...read more
- Virtual server type (instance type): t2.micro
- Firewall (security group): Secure
- 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)

Launch instance

Step 9: Copy the instance IP then paste it in Bitvise SSH Client and import the client key then log in

Instance summary for i-006053a221b8a9fa0

Instance ID: i-006053a221b8a9fa0

IPv6 address: -

Hostname type: IP name: ip-172-31-9-231.ap-south-1.compute.internal

Answer private resource DNS name: IPv4 (A)

Auto-assigned IP address: 52.66.243.175 [Public IP]

IAM Role: -

IMDSv2: Required

Instance state: Running

Private IP DNS name (IPv4 only): ip-172-31-9-231.ap-south-1.compute.internal

Instance type: t2.micro

VPC ID: vpc-0e2181ce3777a313c

Subnet ID: subnet-0e06b1c09b402835f

Instance ARN: arn:aws:ec2:ap-south-1:586794457897:instance/i-006053a221b8a9fa0

Private IPv4 addresses: 172.31.9.231

Public IPv4 DNS: ec2-52-66-243-175.ap-south-1.compute.amazonaws.com

Elastic IP addresses: -

AWS Compute Optimizer finding: Opt-in to AWS Compute Optimizer for recommendation

Auto Scaling Group name: -

Managed: false

Bitvise SSH Client 9.42

Default profile

Login Options Terminal RDP SFTP Services C2S S2C SSH Notes About*

Server

Host: 52.66.243.175

Port: 22

Enable obfuscation: ☐

Obfuscation keyword:

Kerberos

SPN:

GSS/Kerberos key exchange: ☐

Request delegation: ☐

gssapi-keyex authentication: ☒

Authentication

Username: ubuntu

Initial method: publickey

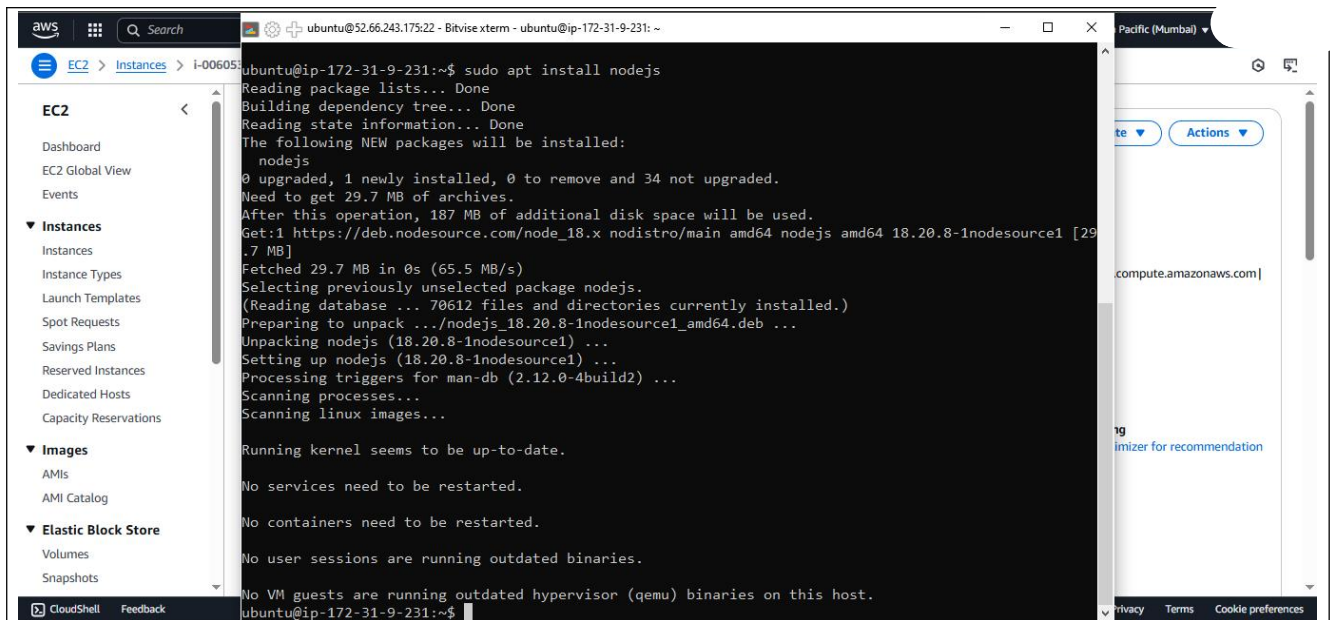
Client key: Global 1

Passphrase:

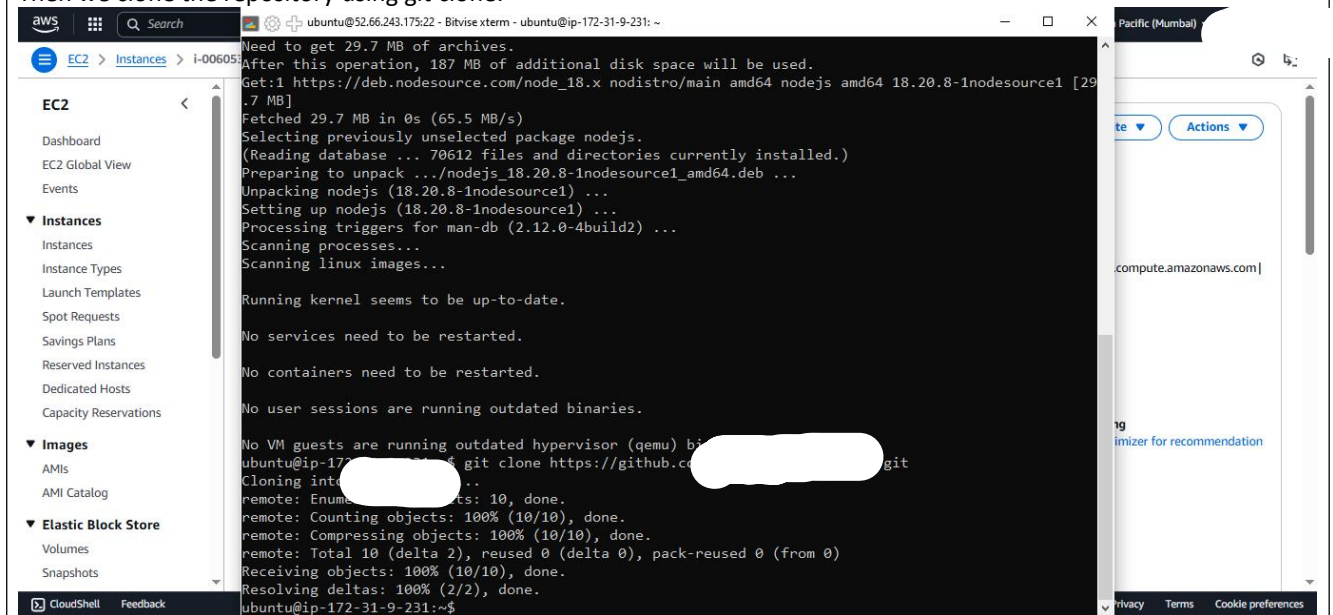
Elevation: Default

Log in

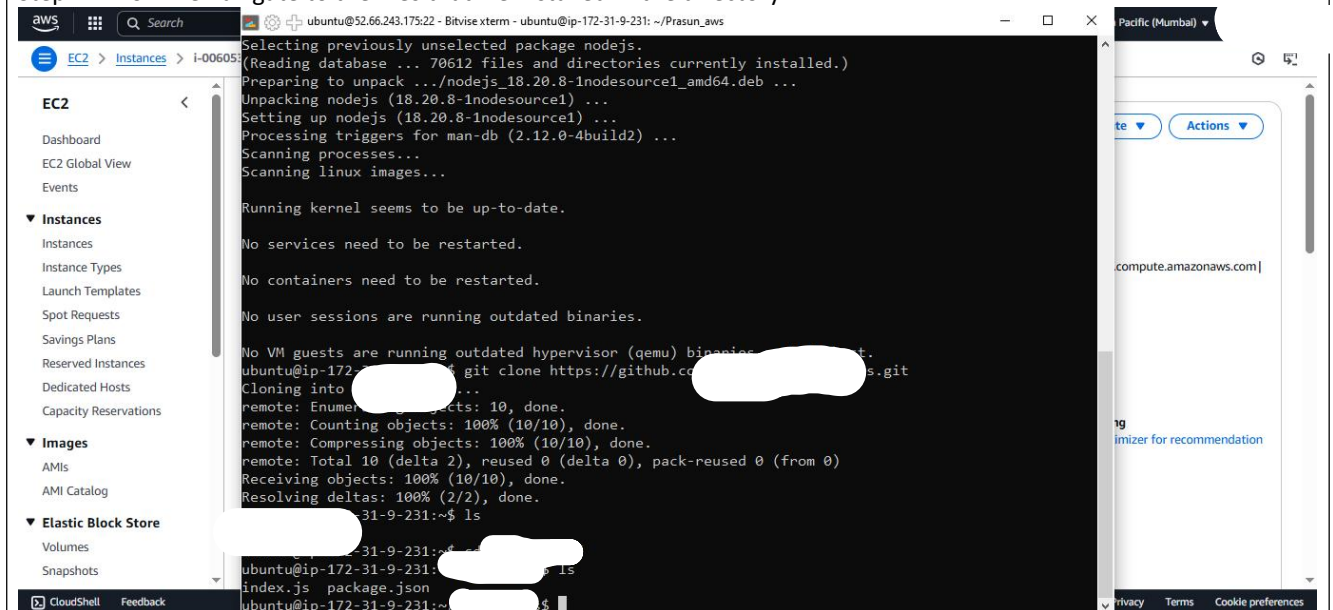
Step 10: Now that we have predefined commands we only need to run only few commands:
First command will be "sudo apt install nodejs"



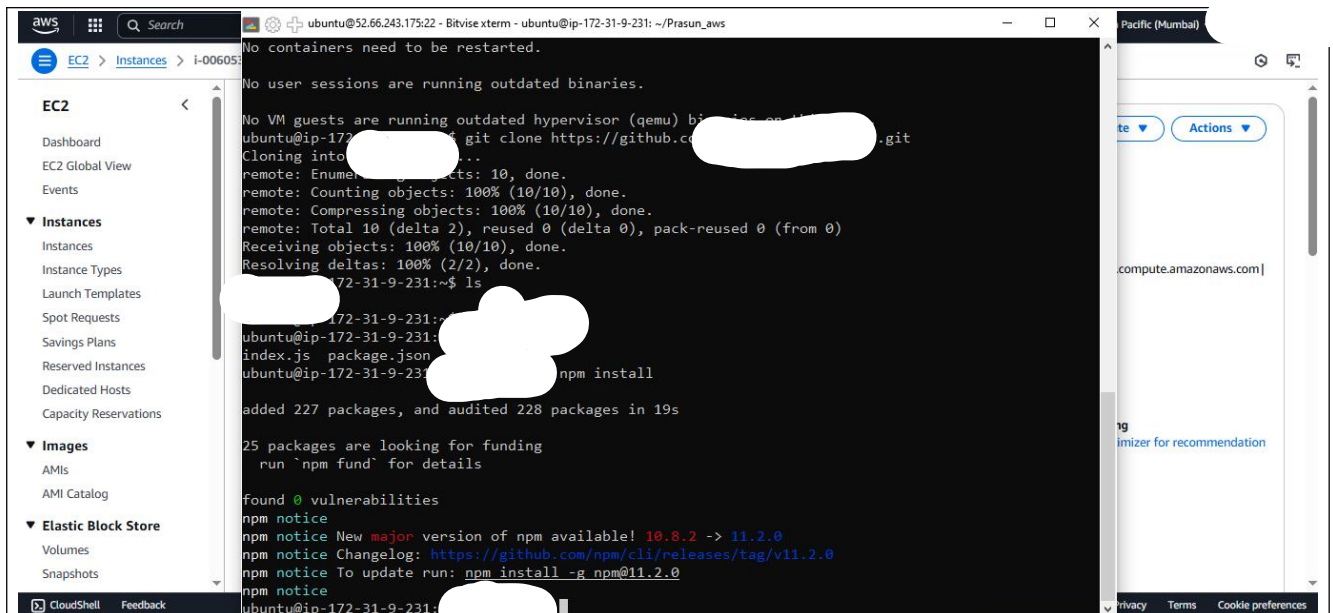
Then we clone the repository using git clone:



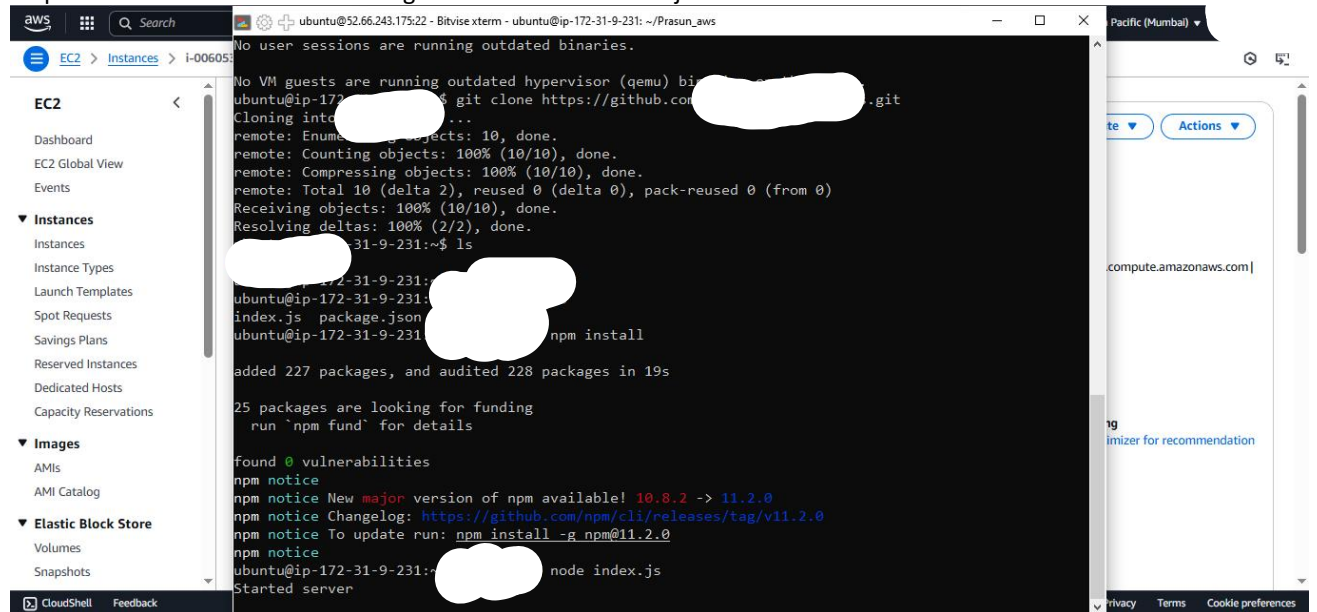
Step 11: Now we navigate to the files that we installed in the directory



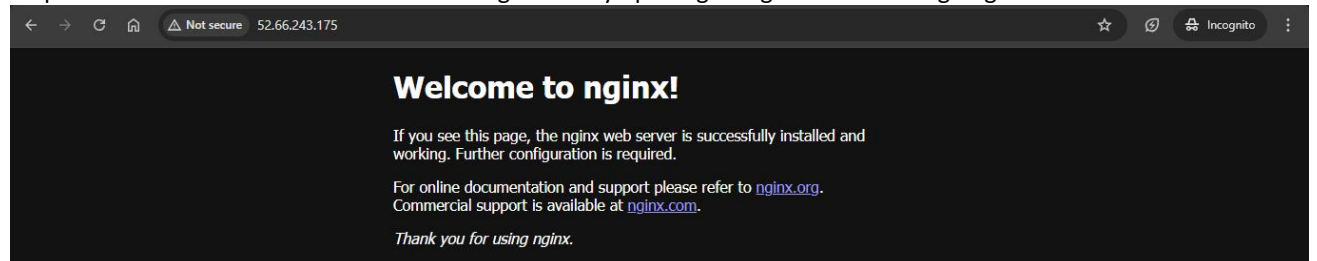
Step 12: Now we run the command "npm install"



Step 13: Now we run the server using command "node index.js"



Step 14: Now we check if the server is running or not by opening incognito mode and going to the IP.



Step 15: Now we go to the IP 52.66.255.244:4000 to check our hosted project.



We have successfully hosted our project by using new security group.