

Lesson-End Project

Customizing EC2 Instance for Web Instance

Project agenda: To create a VPC and launch a web instance for hosting your application

Description: As a cloud engineer at a tech firm, you need to set up a custom network infrastructure. This involves creating a custom VPC, attaching a subnet, setting up an internet gateway and route table, and launching an EC2 instance. This setup will ensure a secure and scalable environment for your application.

Tools required: AWS Management Console

Prerequisites: AWS account

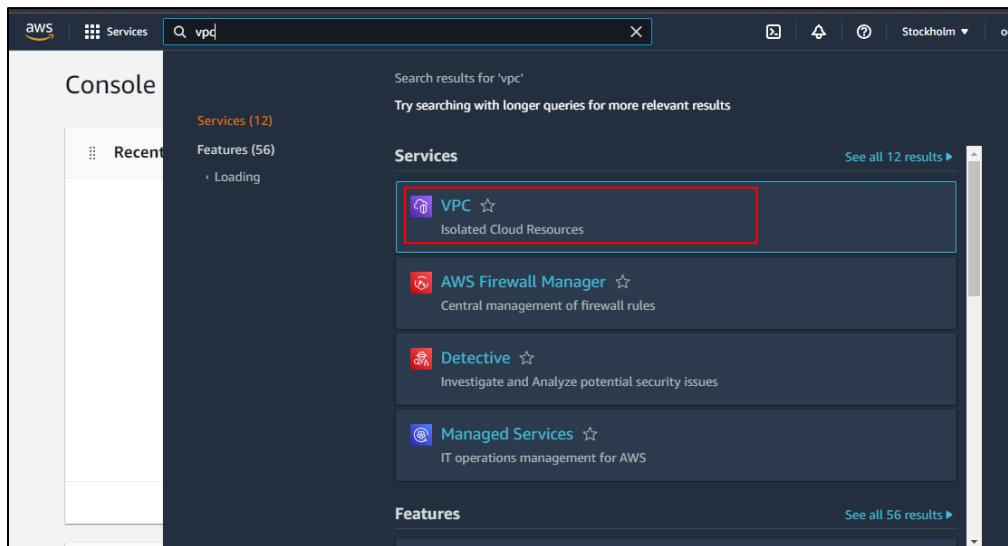
Expected deliverables: EC2 web instance

Steps to be followed:

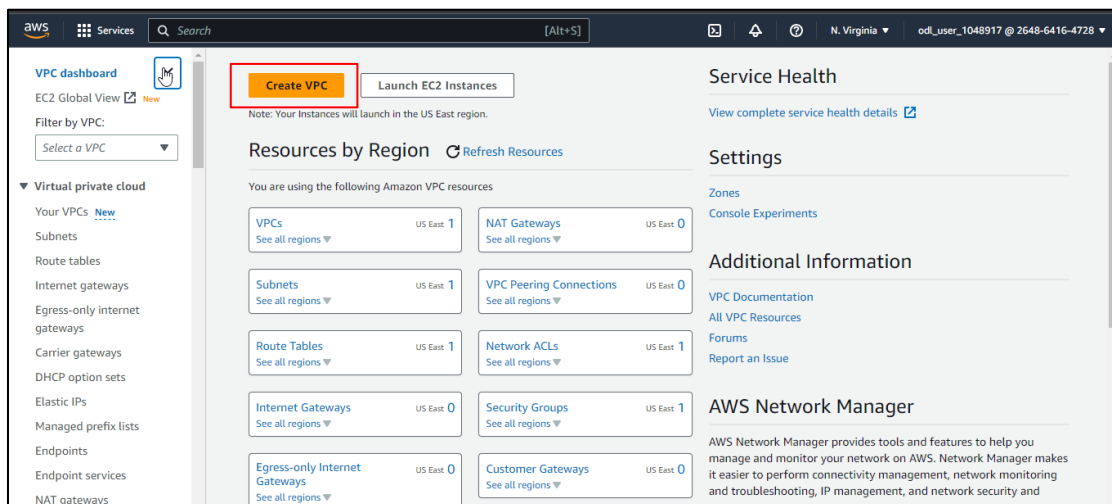
1. Create a custom VPC and enable DNS hostname
2. Create an internet gateway and attach it to the VPC
3. Create a subnet and a route table
4. Configure a route table
5. Launch the EC2 instance

Step 1: Create a custom VPC and enable DNS hostname

1.1 Open the AWS Management Console and search for VPC



1.2 Click on Create VPC



1.3 Enter **my-custom-VPC** as the VPC Name and **10.0.0.0/16** as the IPv4 CIDR

A VPC is an isolated portion of the AWS Cloud populated by AWS objects, such as Amazon EC2 instances.

VPC settings

Resources to create [Info](#)
Create only the VPC resource or the VPC and other networking resources.

☒ VPC only ☐ VPC and more

Name tag - *optional*
Creates a tag with a key of 'Name' and a value that you specify.

IPv4 CIDR block [Info](#)

☒ IPv4 CIDR manual input
☐ IPAM-allocated IPv4 CIDR block

IPv4 CIDR

IPv6 CIDR block [Info](#)

☒ No IPv6 CIDR block
☐ IPAM-allocated IPv6 CIDR block
☐ Amazon-provided IPv6 CIDR block
☐ IPv6 CIDR owned by me

1.4 Click on **Create VPC**

IPv6 CIDR block [Info](#)

☒ No IPv6 CIDR block
☐ IPAM-allocated IPv6 CIDR block
☐ Amazon-provided IPv6 CIDR block
☐ IPv6 CIDR owned by me

Tenancy [Info](#)

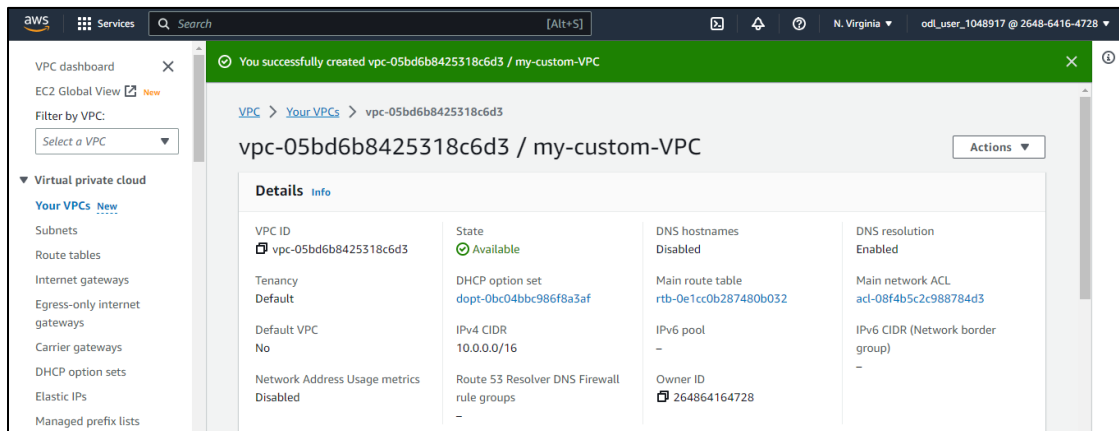
Default

Tags

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

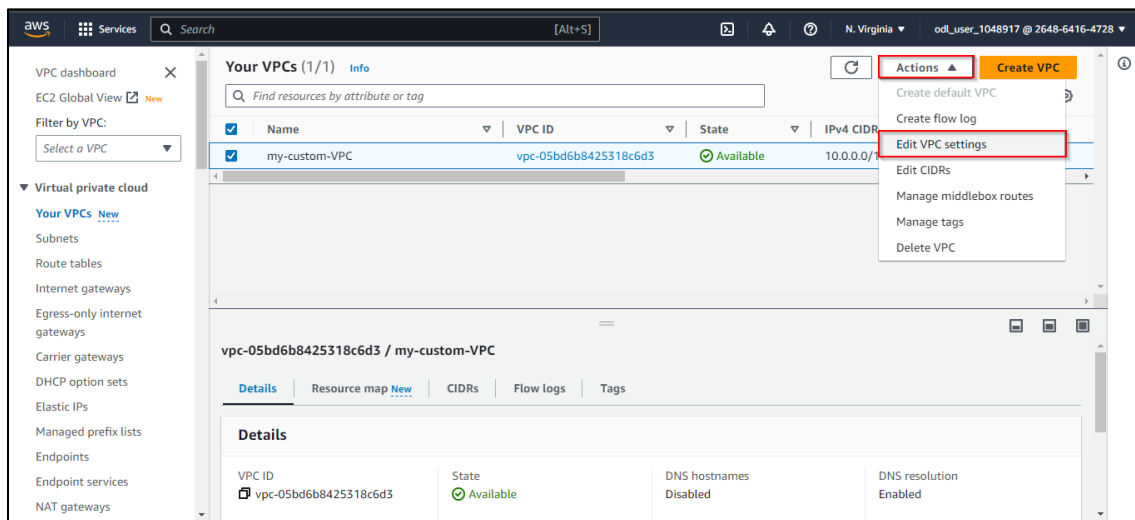
Key	Value - optional	
<input type="text" value="Name"/>	<input type="text" value="my-custom-VPC"/>	<input type="button" value="Remove tag"/>

You can add 49 more tags



The VPC has been created successfully.

1.5 Select the created VPC and click on **Edit VPC settings** under the **Actions** tab



1.6 Select **Enable DNS hostnames** under **DNS settings**, and then click on **Save**

DHCP option set [Info](#)

dopt-0bc04bbc986f8a3af

DNS settings

☒ Enable DNS resolution [Info](#)

☒ Enable DNS hostnames [Info](#)

Network Address Usage metrics settings

☐ Enable Network Address Usage metrics [Info](#)

Cancel **Save**

aws Services Search [Alt+S] N. Virginia odI_user_1048917 @ 2648-6416-4728

VPC dashboard

EC2 Global View [New](#)

Filter by VPC: Select a VPC

Virtual private cloud

Your VPCs [New](#)

Subnets

Route tables

You have successfully modified the settings for vpc-05bd6b8425318c6d3 / my-custom-VPC.

Your VPCs (1/1) [Info](#)

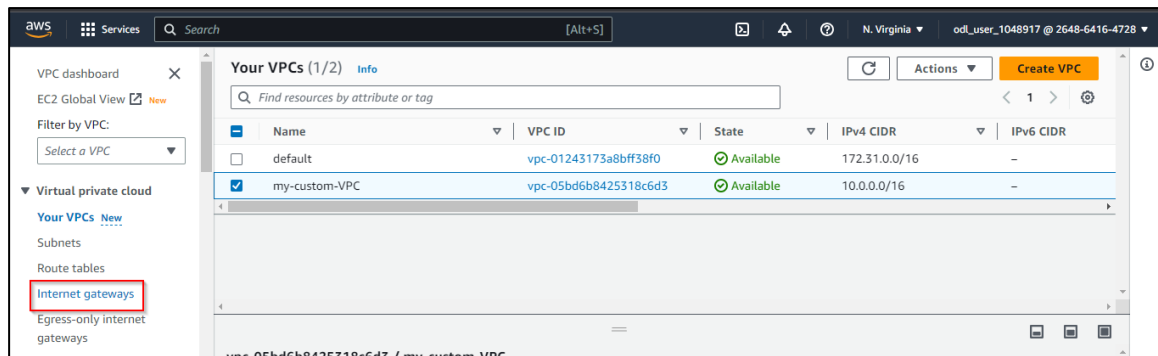
Find resources by attribute or tag

<input checked="" type="checkbox"/>	Name	VPC ID	State	IPv4 CIDR	IPv6 CIDR
<input checked="" type="checkbox"/>	my-custom-VPC	vpc-05bd6b8425318c6d3	Available	10.0.0.0/16	-

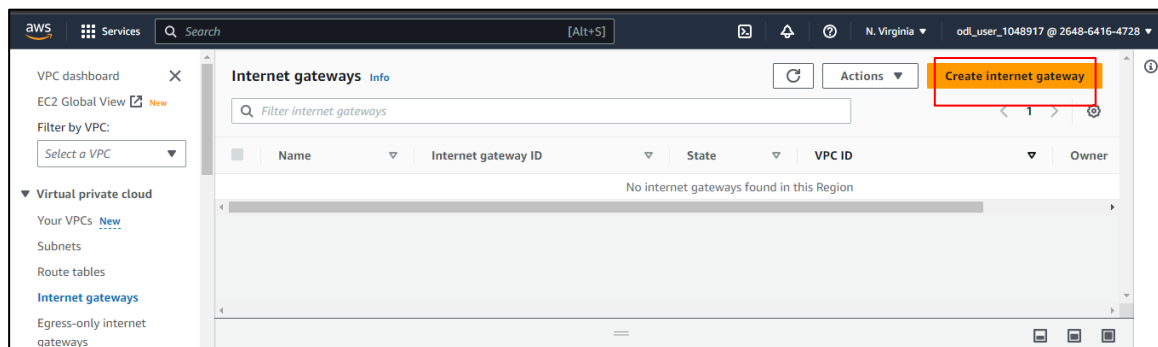
DNS has been successfully enabled.

Step 2: Create an internet gateway and attach it to the VPC

2.1 Navigate to the Internet gateways page



2.2 Click on Create internet gateway



2.3 Enter the name **my-custom-gateway** and click on **Create internet gateway**

An internet gateway is a virtual router that connects a VPC to the internet. To create a new internet gateway specify the name for the gateway below.

Internet gateway settings

Name tag
Creates a tag with a key of 'Name' and a value that you specify.

Tags - optional

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Key: X

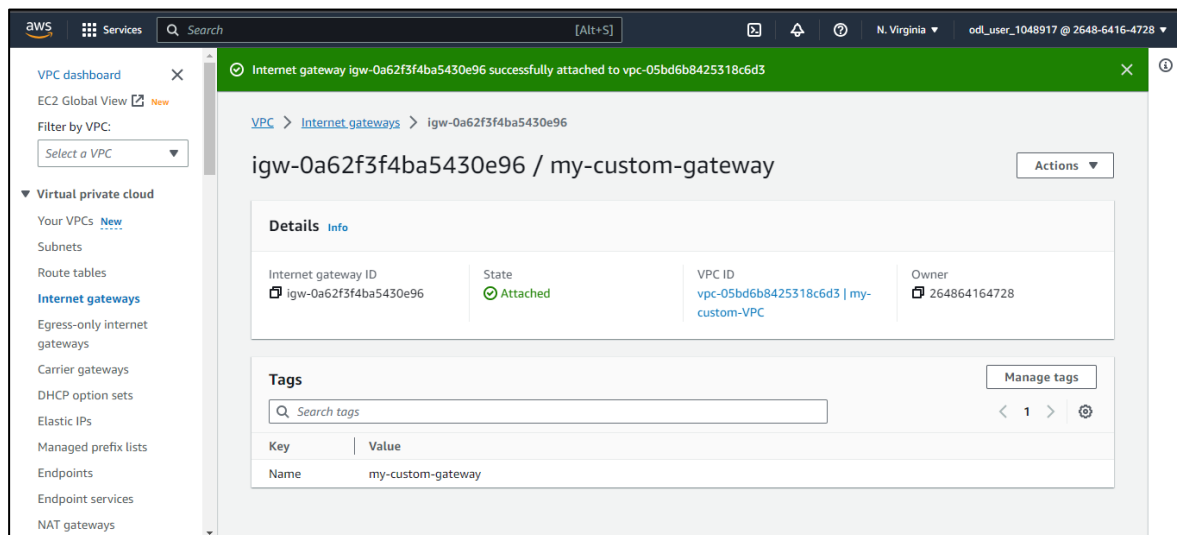
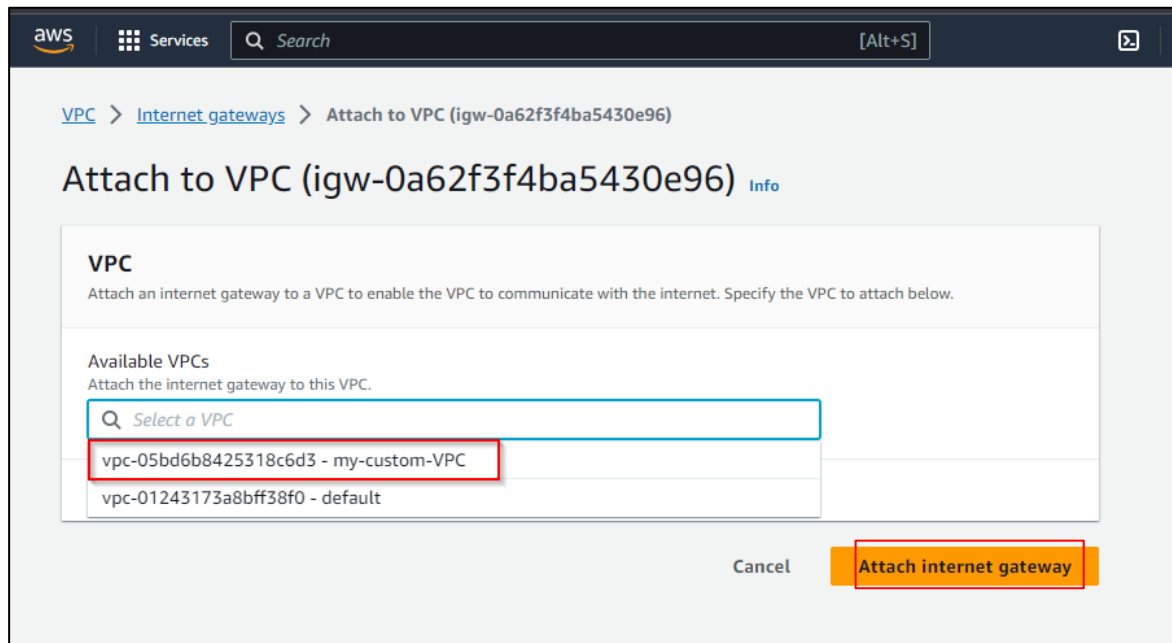
Value - optional: X

You can add 49 more tags.

2.4 Click on **Attach to VPC** under the **Actions** tab

The screenshot shows the AWS Management Console interface. On the left, the 'Virtual private cloud' sidebar is visible. The main content area is titled 'Internet gateways (1/1) Info'. A table lists the gateway 'my-custom-gateway' with ID 'igw-0a62f3f4ba5430e96' and state 'Detached'. The 'Actions' dropdown menu is open, showing options: 'View details', 'Attach to VPC' (highlighted with a red box), 'Detach from VPC', 'Manage tags', and 'Delete internet gateway'. Below the table, the 'Details' tab for the selected gateway is shown, displaying fields for Internet gateway ID, State, VPC ID, and Owner.

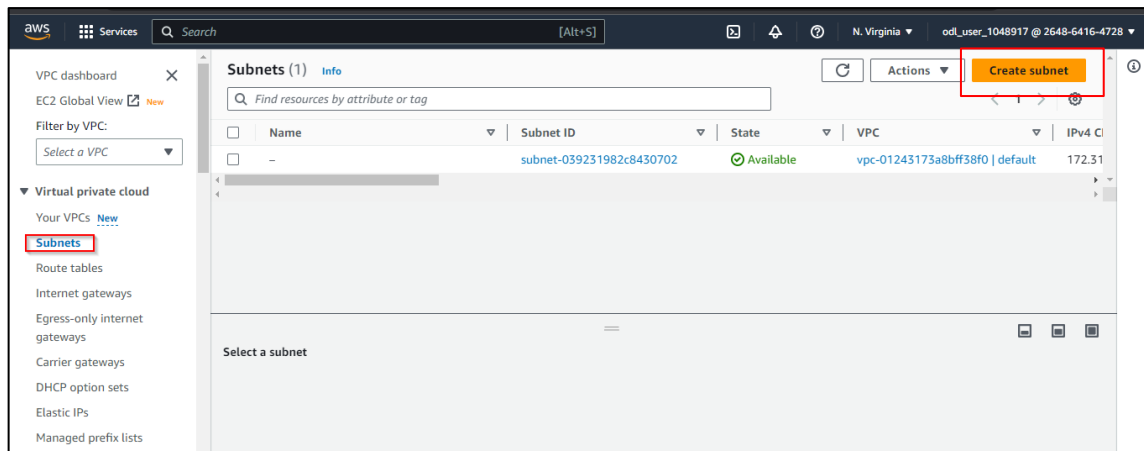
2.5 Select **my-custom-VPC** and click on **Attach internet gateway**



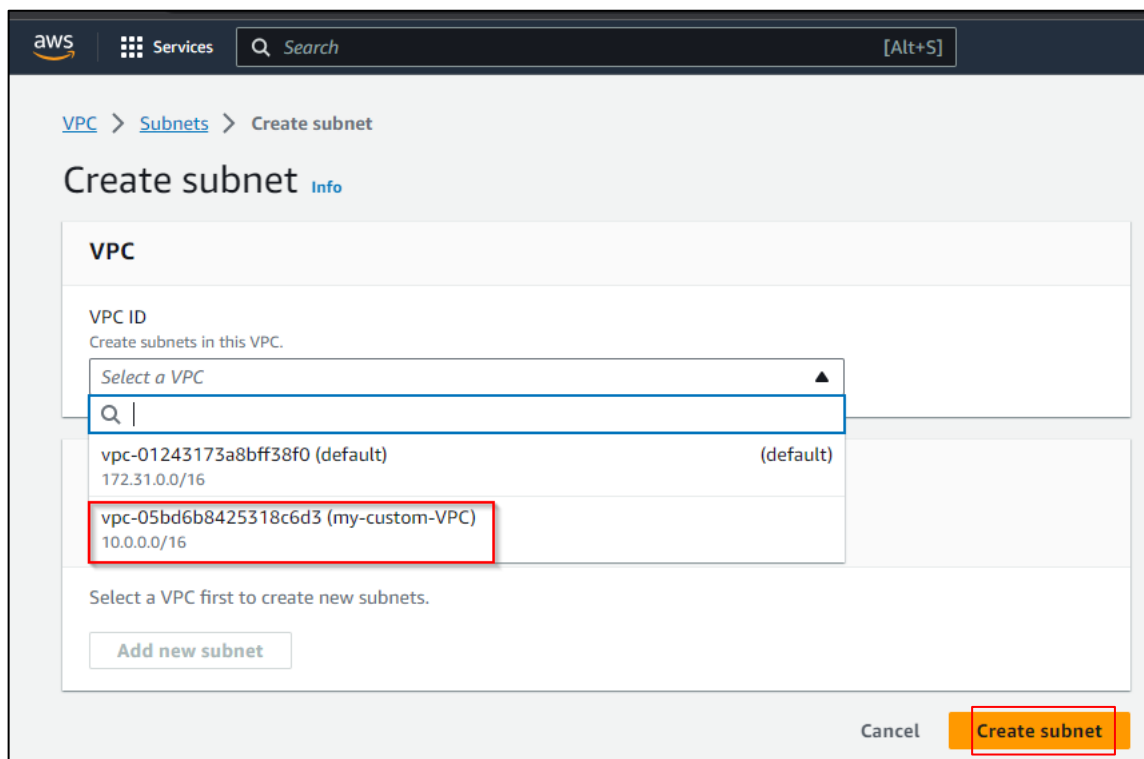
The internet gateway has been attached successfully.

Step 3: Create a subnet and a route table

3.1 Navigate to Subnets and click on Create Subnet



3.2 Choose my-custom-VPC and click on Create subnet



3.3 Enter **my-subnet-custom** as the **Subnet name**, choose the **Availability Zone**, set **10.0.0.0/16** as the **IPv4 CIDR block**, and click on **Create subnet**

Specify the CIDR blocks and Availability Zone for the subnet.

Subnet 1 of 1

Subnet name
Create a tag with a key of 'Name' and a value that you specify.

The name can be up to 256 characters long.

Availability Zone [Info](#)
Choose the zone in which your subnet will reside, or let Amazon choose one for you.

IPv4 CIDR block [Info](#)

► **Tags - optional**

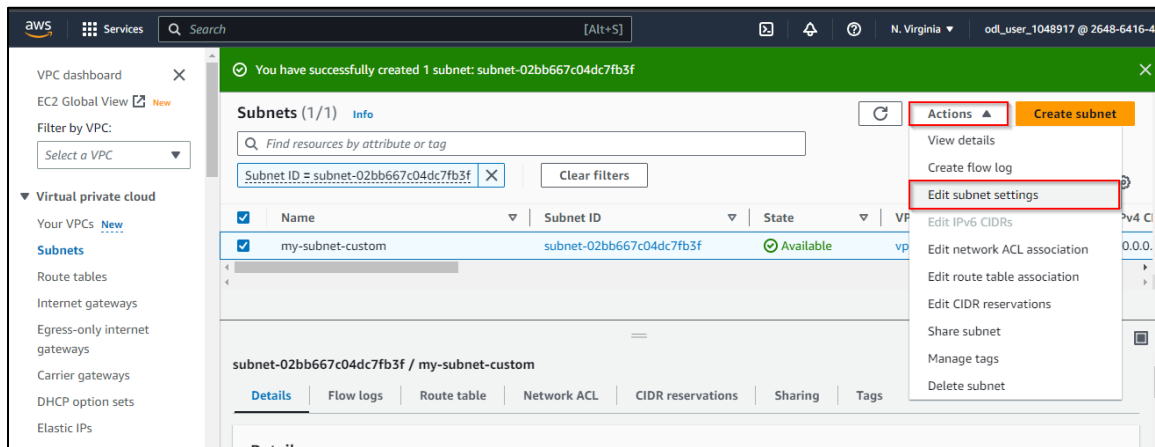
Subnets (1) [Info](#)

<input type="checkbox"/>	Name	Subnet ID	State	VPC	IPv4 CIDR block
<input type="checkbox"/>	my-subnet-custom	subnet-02bb667c04dc7fb3f	Available	vpc-05bd6b8425318c6d3 my-...	10.0.0.0/16

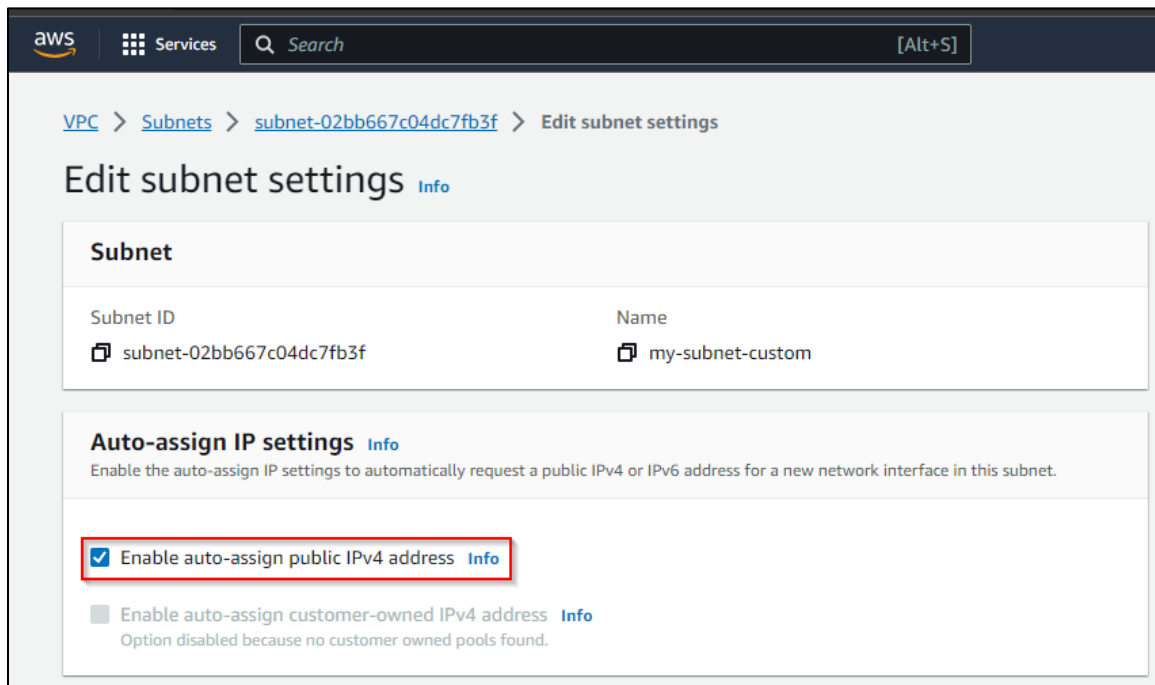
Select a subnet

The subnet has been created successfully.

3.4 Select the subnet, click on **Actions**, and choose **Edit subnet settings**



3.5 Select the **Enable auto-assign public IPv4 addresses** option and click **Save**



Resource-based name (RBN) settings [Info](#)

Specify the hostname type for EC2 instances in this subnet and optional RBN DNS query settings.

☐ Enable resource name DNS A record on launch [Info](#)

☐ Enable resource name DNS AAAA record on launch [Info](#)

Hostname type [Info](#)

☐ Resource name
 ☒ IP name

DNS64 settings

Enable DNS64 to allow IPv6-only services in Amazon VPC to communicate with IPv4-only services and networks.

☐ Enable DNS64 [Info](#)

Cancel
Save

3.6 Navigate to the **Route tables** page and click on **Create route table**

The screenshot shows the AWS Management Console interface. On the left sidebar, under the 'Virtual private cloud' section, the 'Route tables' link is highlighted with a red rectangular box. In the main content area, the 'Route tables (2)' page is displayed. In the top right corner of this page, the 'Create route table' button is highlighted with a red rectangular box. The page also shows a table of existing route tables with columns for Name, Route table ID, Explicit subnet associations, Edge associations, and Main status.

Name	Route table ID	Explicit subnet associati...	Edge associations	Main
-	rtb-0e1cc0b287480b032	-	-	Yes
default	rtb-0ed10aa0bc5283436	-	-	Yes

3.7 Enter **my-custom-route** as the **Name**, select **my-custom-VPC** in the VPC field, and click one **Create the route table**

connection.

Route table settings

Name - optional
Create a tag with a key of 'Name' and a value that you specify.

my-custom-route

VPC
The VPC to use for this route table.

vpc-05bd6b8425318c6d3 (my-custom-VPC)

Tags

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Key **Value - optional**

Q Name X Q my-custom-route X Remove

Add new tag

You can add 49 more tags.

Cancel **Create route table**

3.8 Select the route table, click on **Actions**, and choose **Edit routes**

VPC dashboard X

EC2 Global View New

Filter by VPC: Select a VPC

Virtual private cloud

Your VPCs New

Subnets

Route tables

Internet gateways

Egress-only internet gateways

Carrier gateways

DHCP option sets

Elastic IPs

Managed prefix lists

Endpoints

Endpoint services

NAT gateways

Route tables (1/3) Info

Find resources by attribute or tag

	Name	Route table ID	Explicit subnet associati..
<input type="checkbox"/>	-	rtb-0e1cc0b287480b032	-
<input type="checkbox"/>	default	rtb-0ed10aa0bc5283436	-
<input checked="" type="checkbox"/>	my-custom-route	rtb-02efbf395cfd1834e	-

rtb-02efbf395cfd1834e / my-custom-route

Details Routes Subnet associations Edge associations Route propagation Tags

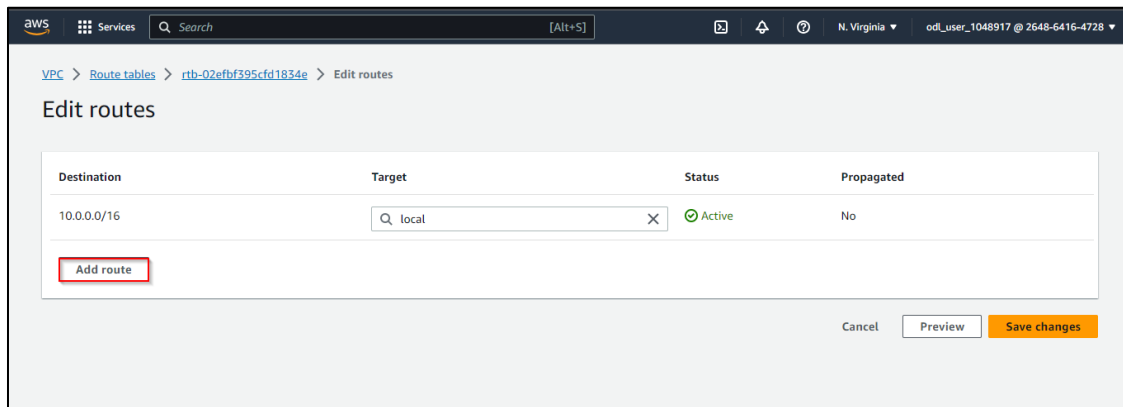
You can now check network connectivity with Reachability Analyzer Run Reachability Analyzer X

Details

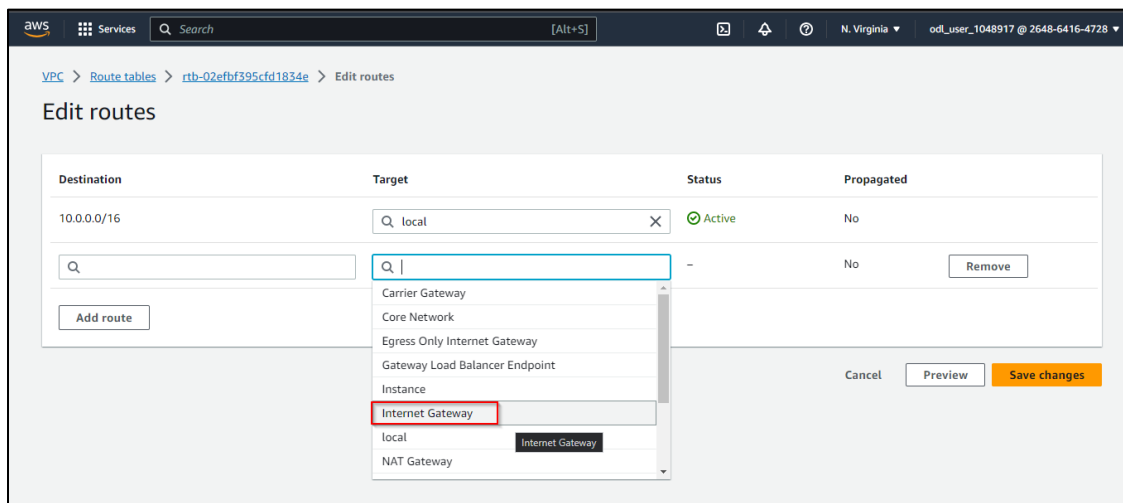
Actions

- View details
- Set main route table
- Edit subnet associations
- Edit edge associations
- Edit route propagation
- Edit routes**
- Manage tags
- Delete route table
- Troubleshoot
- Trace network reachability

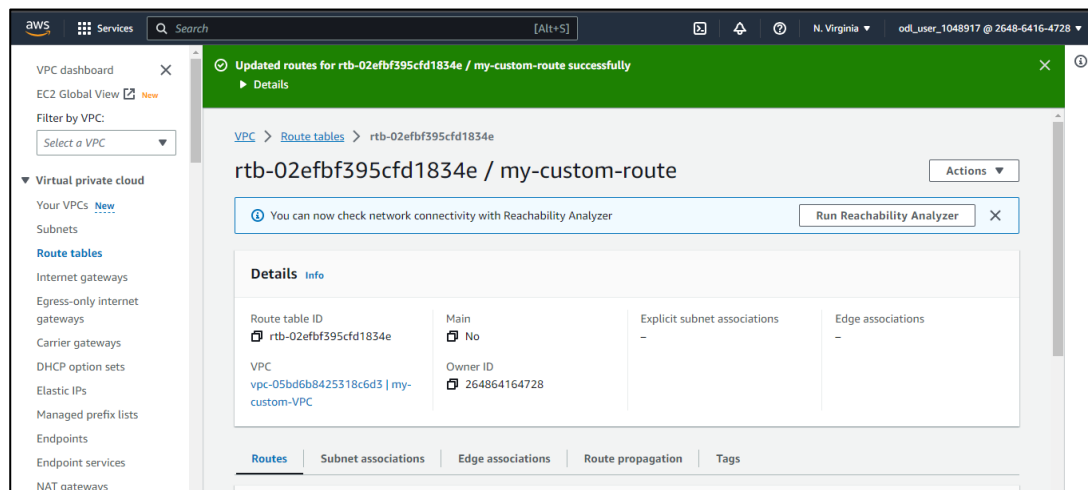
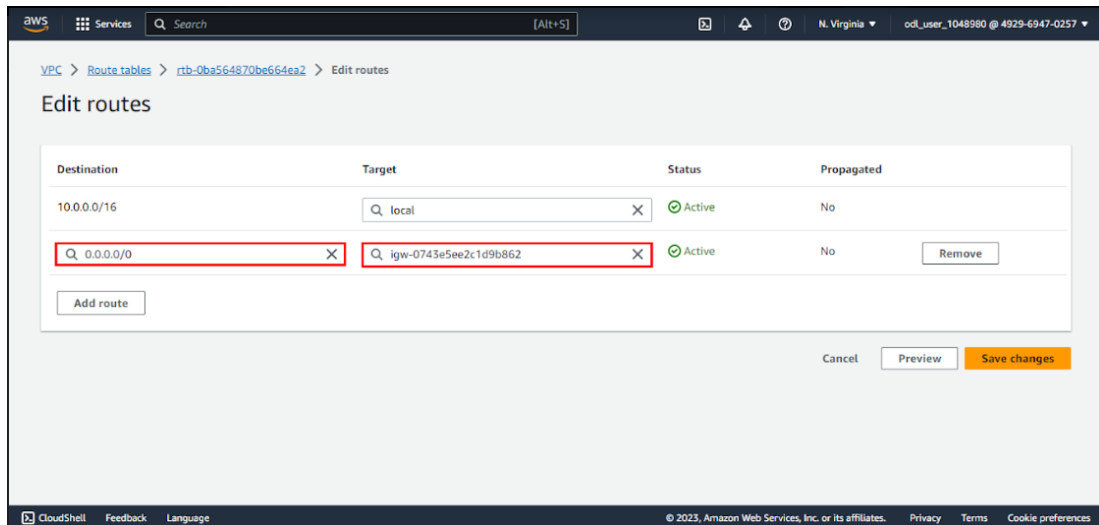
3.9 Click on **Add route**



3.10 Select **Internet Gateway** in the **Target** field



3.11 Click on **Save changes**



The edit route table has been successfully created.

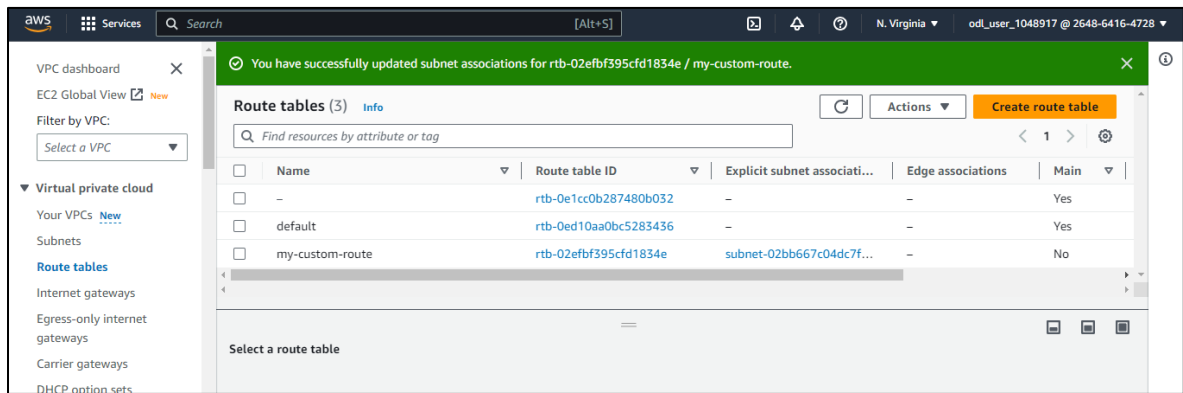
Step 4: Configure a route table

4.1 In the **Route tables** dashboard, click on **Edit subnet ociations** under **Actions**

The screenshot shows the AWS Management Console interface for a specific route table. The left sidebar contains the 'Virtual private cloud' menu with options like 'Your VPCs', 'Subnets', 'Route tables', 'Internet gateways', etc. The main content area displays the details for route table 'rtb-09f0f8defe77417da / my-custom-route'. The 'Details' tab is active, showing information such as Route table ID, VPC, Main status, and Owner ID. The 'Actions' dropdown menu is open, and the option 'Edit subnet associations' is highlighted with a red rectangular box.

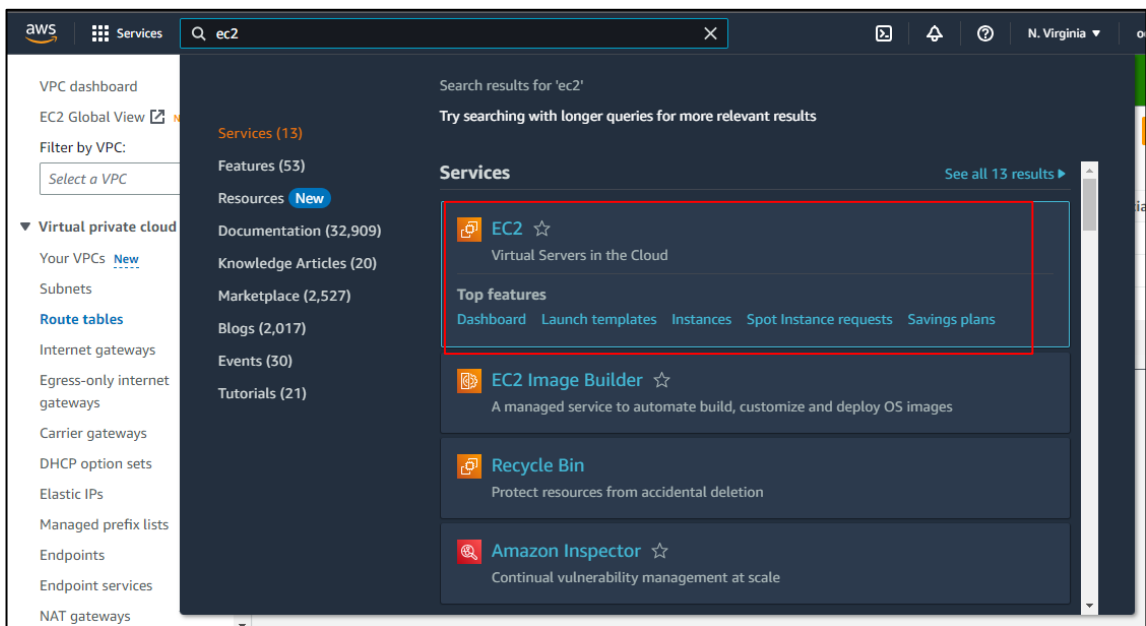
4.2 Select **my-subnet-custom** and click on **Save associations**

The screenshot shows the 'Edit subnet associations' page in the AWS Management Console. The breadcrumb trail indicates the path: 'VPC > Route tables > rtb-02efbf395cfd1834e > Edit subnet associations'. The page title is 'Edit subnet associations' with a subtitle 'Change which subnets are associated with this route table.' Below this, there is a section 'Available subnets (1/1)' with a search bar and a table. The table has columns for Name, Subnet ID, IPv4 CIDR, IPv6 CIDR, and Route table ID. The row for 'my-subnet-custom' is selected, and its checkbox is checked, highlighted with a red box. Below the table, the 'Selected subnets' section shows 'subnet-02bb667c04dc7fb3f / my-subnet-custom' with a close button. At the bottom right, the 'Save associations' button is highlighted with a red rectangular box.

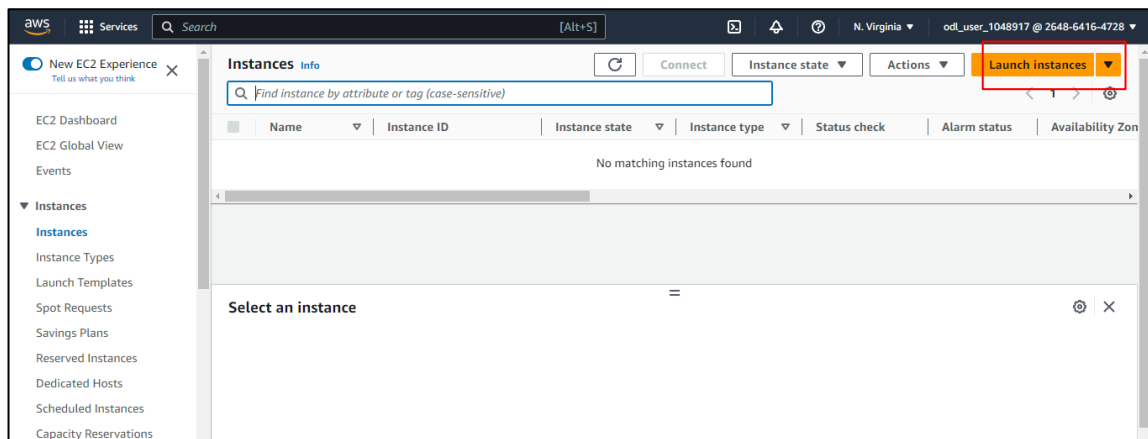


Step 5: Launch the EC2 instance

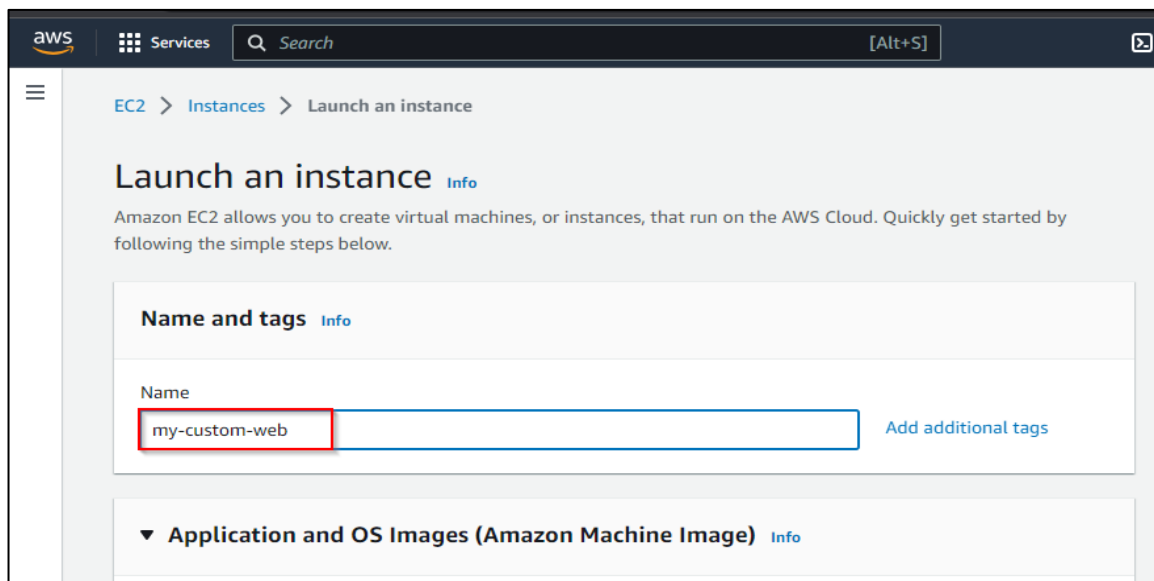
5.1 Navigate back to the EC2 console



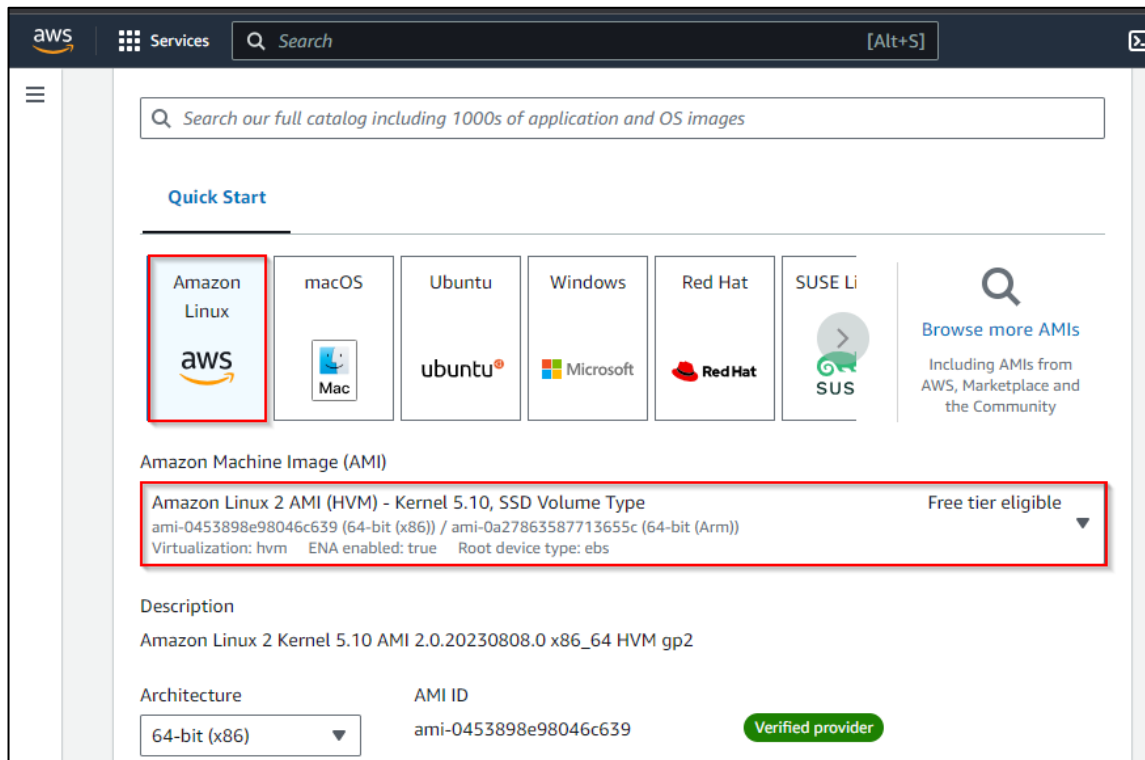
5.2 Click on **Launch instances**



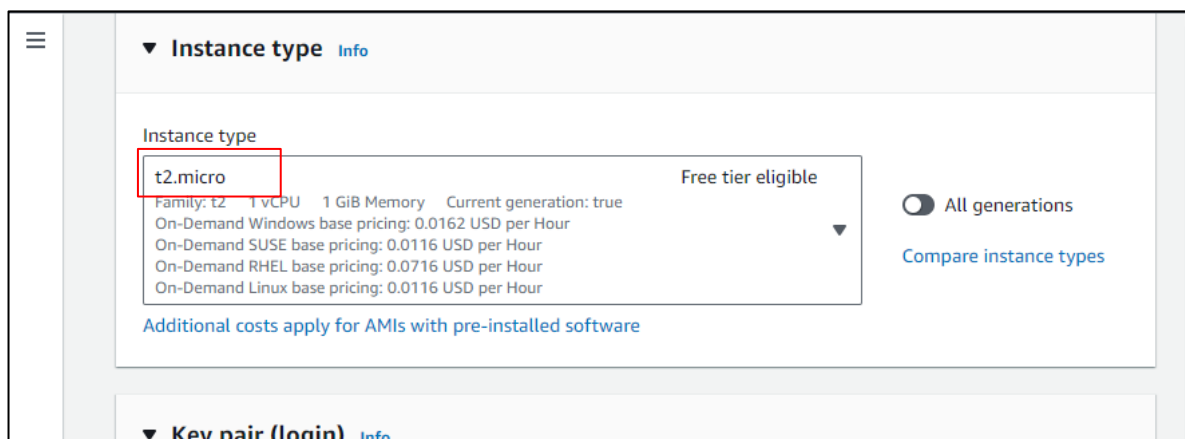
5.3 Enter the name **my-custom-web** in the field



5.4 Click on the **Amazon Linux** option and select the **SSD Volume Type** as **AMI**



5.5 Select the instance type as **t2.micro**



5.6 Enter the **Key pair name** and click on the **Create key pair** button

Create key pair

Key pair name

Key pairs allow you to connect to your instance securely.

key

The name can include upto 255 ASCII characters. It can't include leading or trailing spaces.

Key pair type


☒ RSA
RSA encrypted private and public key pair

☐ ED25519
ED25519 encrypted private and public key pair

Private key file format

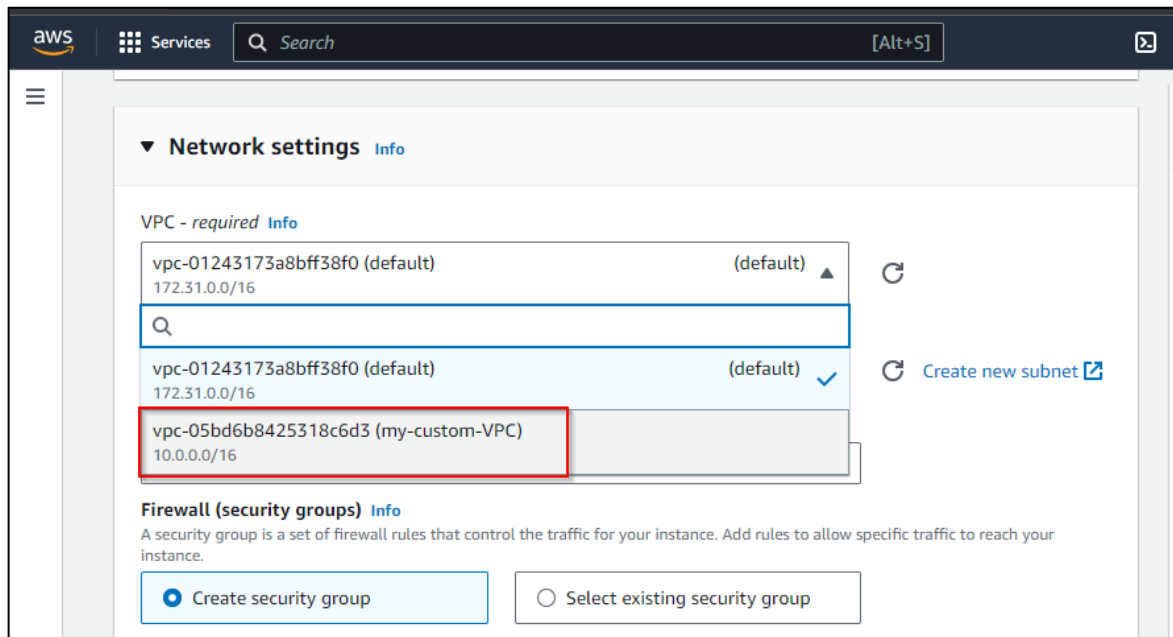
☒ .pem
For use with OpenSSH

☐ .ppk
For use with PuTTY

 When prompted, store the private key in a secure and accessible location on your computer. You will need it later to connect to your instance. [Learn](#)

Cancel Create key pair

5.7 Configure **Network settings** by selecting your **VPC** and an availability zone



Network settings [Info](#)

VPC - required [Info](#)

vpc-01243173a8bff38f0 (default) 172.31.0.0/16 (default) ▲ ↻

Q

vpc-01243173a8bff38f0 (default) 172.31.0.0/16 (default) ✓ ↻ [Create new subnet](#)

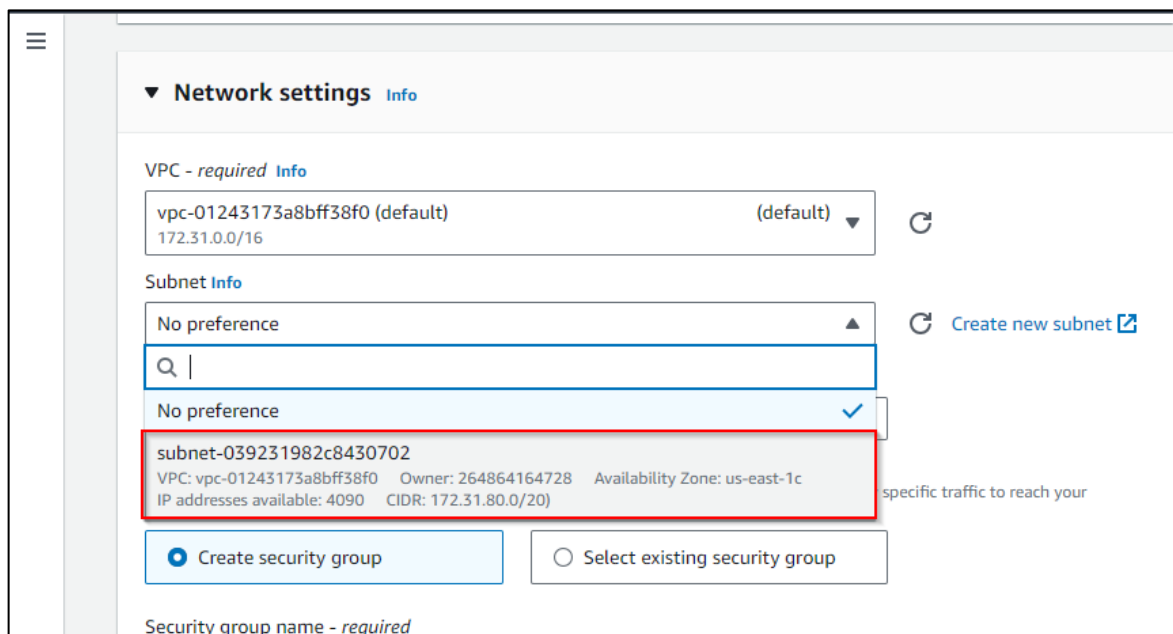
vpc-05bd6b8425318c6d3 (my-custom-VPC) 10.0.0.0/16

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

5.8 Select the **availability zone** in the **Subnet**



Network settings [Info](#)

VPC - required [Info](#)

vpc-01243173a8bff38f0 (default) 172.31.0.0/16 (default) ▼ ↻

Subnet [Info](#)

No preference ▲ ↻ [Create new subnet](#)

Q |

No preference ✓

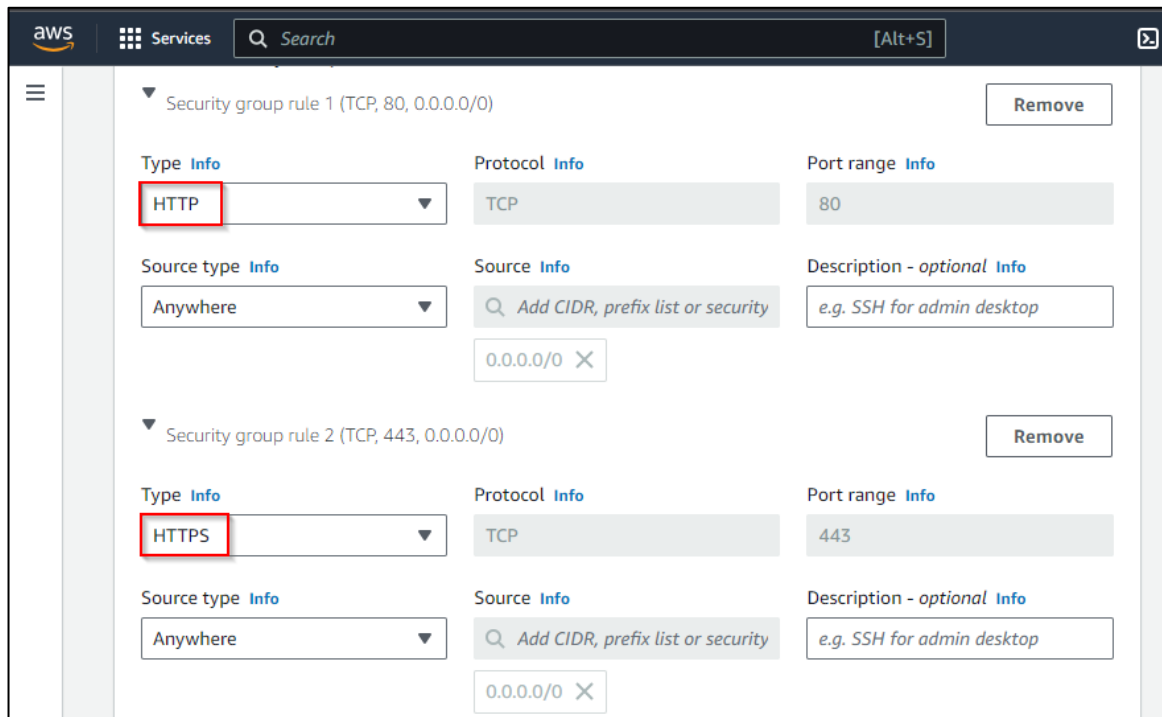
subnet-039231982c8430702

VPC: vpc-01243173a8bff38f0 Owner: 264864164728 Availability Zone: us-east-1c
IP addresses available: 4090 CIDR: 172.31.80.0/20

☒ Create security group ☐ Select existing security group

Security group name - required

5.9 Add inbound rules for HTTP and HTTPS



The screenshot displays the AWS Management Console interface for configuring security group rules. It shows two rules, 'Security group rule 1' and 'Security group rule 2', both configured for TCP traffic. Rule 1 is for port 80 (HTTP) and Rule 2 is for port 443 (HTTPS). Both rules allow traffic from 'Anywhere' (0.0.0.0/0). The 'Type' dropdown menu for each rule is highlighted with a red box, showing 'HTTP' for rule 1 and 'HTTPS' for rule 2.

Rule Name	Type	Protocol	Port Range	Source Type	Source	Description
Security group rule 1 (TCP, 80, 0.0.0.0/0)	HTTP	TCP	80	Anywhere	0.0.0.0/0	e.g. SSH for admin desktop
Security group rule 2 (TCP, 443, 0.0.0.0/0)	HTTPS	TCP	443	Anywhere	0.0.0.0/0	e.g. SSH for admin desktop

5.10 Click on **Advanced details**, enter the code in **User data**, and click on **Launch instance**

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

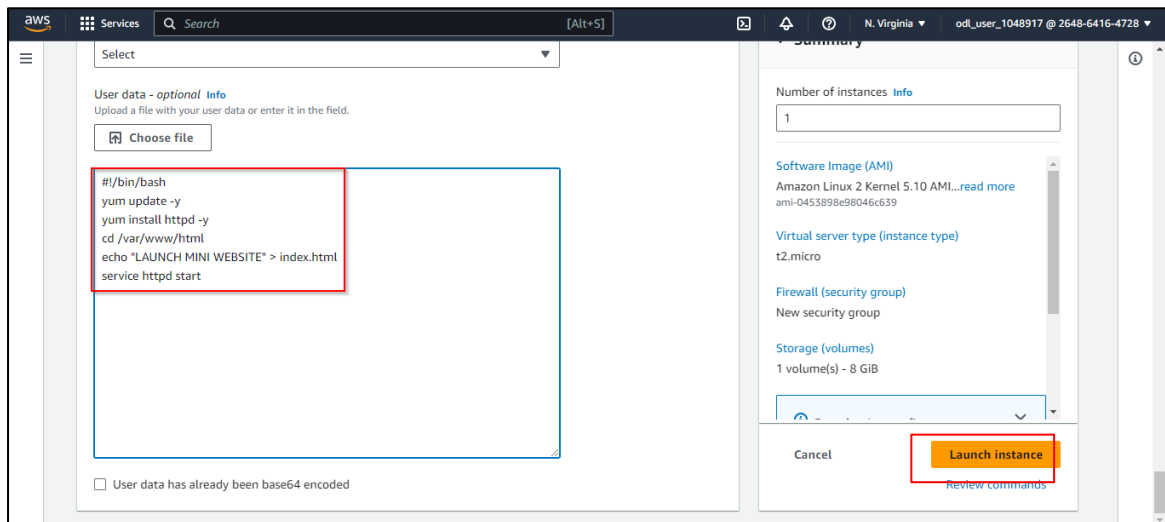
```
yum update -y
```

```
yum install httpd -y
```

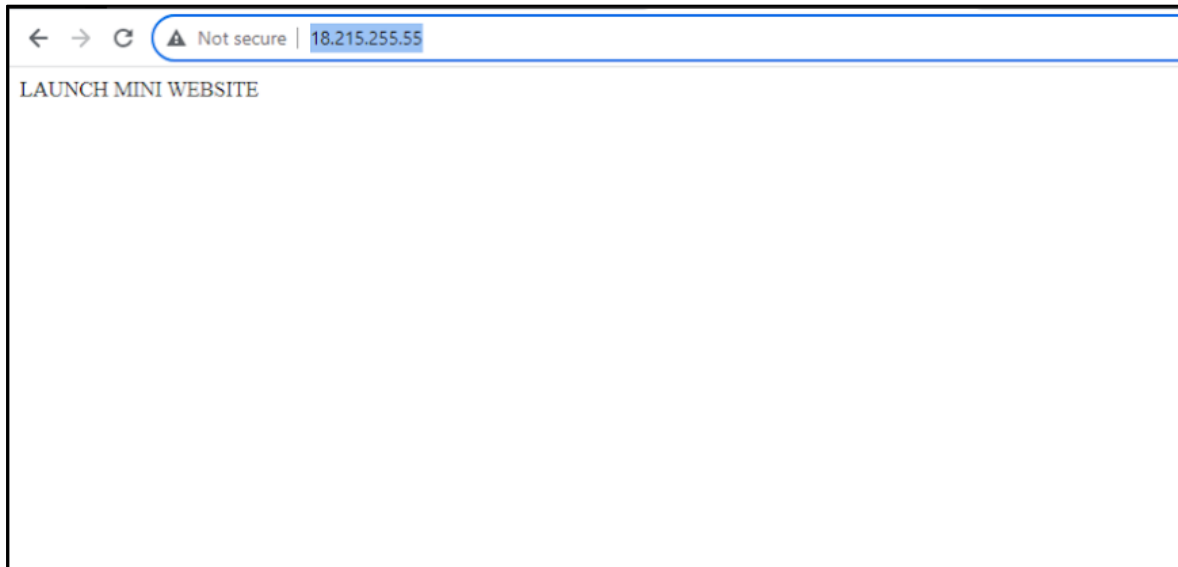
```
cd /var/www/html
```

```
echo "LAUNCH MINI WEBSITE" > index.html
```

```
service httpd start
```



5.11 After the instance is running, copy the IPv4 address and paste it into a browser to view the mini website



By following these steps, you have successfully customized an EC2 instance for web deployment within an Amazon Web Services (AWS) environment.