

## VPC Networking Lab

### How do you document this lab?

1. Capture the full screenshot of the AWS console; this should contain your name on the screenshot.

Ex:

The screenshot shows the AWS VPC dashboard. At the top right, the region is set to "N. Virginia" with the account ID "vxs220071" next to it. A red circle highlights this area. The left sidebar lists various VPC components like Your VPCs, Subnets, Route tables, etc. The main content area displays "Resources by Region" for the US East region, showing counts for VPCs (2), Subnets (10), Route Tables (4), Internet Gateways (2), Egress-only Internet Gateways (0), DHCP option sets (1), NAT Gateways (0), VPC Peering Connections (0), Network ACLs (2), Security Groups (4), Customer Gateways (0), and Virtual Private Gateways (0). The "Service Health" and "Settings" sections are also visible on the right.

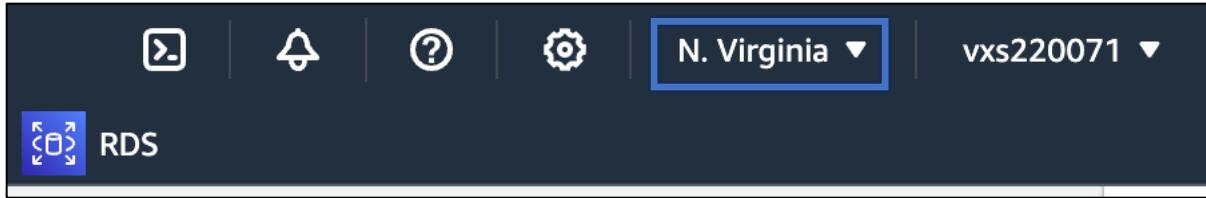
2. Make sure you do your labs in the North Virginia region; this is not mandatory but is advice. This will help troubleshoot a lot of things, including your billing.

**Deliverables:** Screenshots at points g, n, s, y, and cc must be submitted. A total of five screenshots should be submitted.

#### Step 1: Create VPC

- Login to AWS Console and set the region to N.Virginia. This option is available at the top right corner beside your account name.

## Build AWS Network



- b. Now, search for VPC service and click on **Create VPC**. Choose/Fill the options as given below and Create the VPC. Note: Understand each column in detail. Understand that VPC spans the entire region.

VPC > Your VPCs > Create VPC

### Create VPC Info

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.  
my-first-vpc

IPv4 CIDR block Info  
 IPv4 CIDR manual input  IPAM-allocated IPv4 CIDR block

IPv4 CIDR  
10.0.0.0/16  
CIDR block size must be between /16 and /28.

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

## Build AWS Network

The screenshot shows the 'Tags' configuration dialog for a new VPC. It includes a description of what tags are, a key-value pair ('Name: my-first-vpc'), and buttons for 'Add tag', 'Remove tag', 'Cancel', and 'Create 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*

You can add 49 more tags

- c. This is how it looks when you first create your VPC. Observe the resource map as we move forward while creating each resource in VPC.

You have to understand that for every VPC created, a **route table** is created by default.

The screenshot shows the 'Resource map' tab for a VPC named 'my-first-vpc'. It displays four categories: VPC, Subnets (0), Route tables (1), and Network connections (0). The Route tables section shows one entry: 'rtb-0f8104f12d11d5bf7'.

vpc-04c8225ab0993706c / my-first-vpc

Details | **Resource map** | CIDRs | Flow logs | Tags | Integrations

**Resource map** Info

VPC Show details  
Your AWS virtual network  
my-first-vpc

Subnets (0)  
Subnets within this VPC

Route tables (1)  
Route network traffic to resources  
rtb-0f8104f12d11d5bf7

Network connections (0)  
Connections to other networks

### Step 2: Create an Internet Gateway and Attach it to VPC.

- d. Now, Let's create an Internet Gateway.

What is an Internet Gateway? It's simply a virtual device to connect to the Internet. So, let's say we have a server created in a VPC. How do you connect to the internet from that server? That's where Internet Gateway comes into the picture. It's pretty similar to the Spectrum Wi-Fi, right? (Analogy)

- e. Go to the VPC page and look out for Internet Gateway. Create One as shown below.

## Build AWS Network

VPC > Internet gateways > Create internet gateway

### Create internet gateway Info

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	Value - optional
<input type="text" value="Name"/> <input type="button" value="X"/>	<input type="text" value="my-first-igw"/> <input type="button" value="X"/> <input type="button" value="Remove"/>
<input type="button" value="Add new tag"/>	

You can add 49 more tags.

- f. Now that we have created an IGW= Internet Gateway, we must attach it to the VPC, just like how we link our Router to an Internet Port at home.

Click on Attach to VPC; on the next page, select the VPC created and attach it.

Internet gateways (1/2) <small>Info</small>				Actions ▲	<input type="button" value="Create internet gateway"/>
<input type="text" value="Search"/>				<input type="button" value="View details"/>	<input type="button" value="Owner"/>
Name	Internet gateway ID	State		<input type="button" value="Attach to VPC"/>	<input type="button" value="Detach from VPC"/>
-	igw-09ac8b7373929f424	Attached			<input type="button" value="Manage tags"/> <input type="button" value="Delete internet gateway"/>
my-first-igw	igw-0eda1b8b040963a7b	Detached			<input type="button" value="Manage tags"/> <input type="button" value="Delete internet gateway"/>

## Build AWS Network

VPC > Internet gateways > Attach to VPC (igw-0eda1b8b040963a7b)

### Attach to VPC (igw-0eda1b8b040963a7b) Info

**VPC**  
Attach an internet gateway to a VPC to enable the VPC to communicate with the internet. Specify the VPC to attach below.

**Available VPCs**  
Attach the internet gateway to this VPC.

Select a VPC

vpc-04c8225ab0993706c - my-first-vpc

▶ AWS Command Line Interface command

**Cancel** **Attach internet gateway**

VPC > Route tables > rtb-09d5391cf19617ae4 > Edit routes

### Edit routes

Destination	Target	Status	Propagated
10.0.0.0/16	local	Active	No
Q 0.0.0.0/0	Carrier Gateway Core Network Egress Only Internet Gateway Gateway Load Balancer Endpoint Instance Internet Gateway local NAT Gateway Network Interface Outpost Local Gateway Peering Connection Transit Gateway Virtual Private Gateway	-	No

Add route

**Remove**

**Cancel** **Preview** **Save changes**

VPC > Route tables > rtb-09d5391cf19617ae4 > Edit routes

### Edit routes

Destination	Target	Status	Propagated
10.0.0.0/16	local	Active	No
Q 0.0.0.0/0	Internet Gateway igw- Use: "igw-" igw-04bffa84be93c03b6 (my-first-igw)	-	No

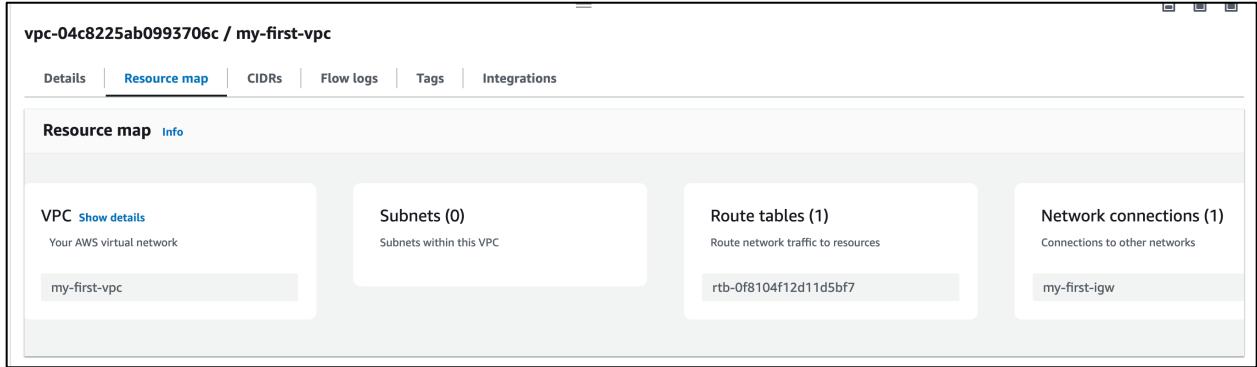
Add route

**Remove**

**Cancel** **Preview** **Save changes**

g. This is what the network diagram looks like after this setup.

## Build AWS Network



### Step 3: Create Private and Public Subnets.

- h. Let's divide our VPC network into public and private subnets.

Example: Think of an office building with different departments like HR, Finance, and IT, each occupying its own section of the building. Similarly, subnets in AWS VPC segregate resources based on their function or purpose, ensuring that different parts of your cloud infrastructure remain isolated and secure.

- i. We are creating two public and two private subnets.

- j. Public Subnet 1

[VPC](#) > [Subnets](#) > Create subnet

## Create subnet Info

### VPC

#### VPC ID

Create subnets in this VPC.

vpc-04c8225ab0993706c (my-first-vpc) ▾

### Associated VPC CIDRs

#### IPv4 CIDRs

10.0.0.0/16

### Subnet settings

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.

my-public-subnet-1

The name can be up to 256 characters long.

## Build AWS Network

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 VPC CIDR block** [Info](#)  
Choose the VPC's IPv4 CIDR block for the subnet. The subnet's IPv4 CIDR must lie within this block.  
▼

**IPv4 subnet CIDR block**  
 256 IPs  
< > ^ v

**▼ Tags - optional**

Key	Value - optional	Remove
<input type="text" value="Name"/> X	<input type="text" value="my-public-subnet-1"/> X	Remove

Add new tag

You can add 49 more tags.

Remove

Add new subnet

- k. Public Subnet 2: Your task is to create a public subnet with the name **my-public-subnet-2**, the CIDR block **10.0.2.0/24**, and the region is **us-east-1b**.
- l. Private Subnet 1:

## Build AWS Network

**Subnet 3 of 4**

**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 VPC CIDR block** [Info](#)  
Choose the VPC's IPv4 CIDR block for the subnet. The subnet's IPv4 CIDR must lie within this block.

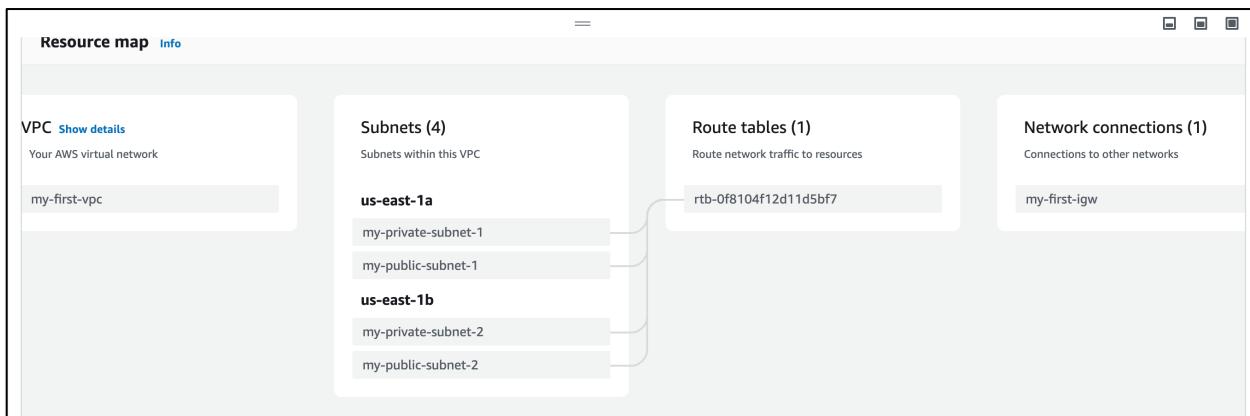
**IPv4 subnet CIDR block**  
 256 IPs  
< > ^ v

**▼ Tags - optional**

Key	Value - optional	Remove
<input type="text" value="Name"/> X	<input type="text" value="my-private-subnet-1"/> X	Remove

Add new tag  
You can add 49 more tags.  
Remove

- m. Private Subnet 2: Your task is to create a private subnet named **my-private-subnet-2**, the CIDR block **10.0.4.0/24**, and the region is **us-east-1b**.
- n. After creating the subnets, this is what my network map looks like.



### Step 4: Create Private Route Tables.

- o. After creating the subnets, we should define the networks by making the Route Tables.

What is a Route Table, and what is it used for?

Example: In a city, there are different roads leading to various neighborhoods and landmarks. Likewise, route tables in AWS VPC act as digital roadmaps, steering data traffic to other parts of your cloud infrastructure and ensuring efficient communication between services and resources.

Remember that a default route table was created when we created the VPC. We will use it for public communication purposes.

- p. Let's create a Private Route Table.

This private route table allows communication for resources in a private subnet.

The screenshot shows the 'Create route table' wizard in the AWS VPC console. The steps are:

- VPC > Route tables > Create route table
- Create route table Info

**Route table settings**

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

**VPC**  
The VPC to use for this route table.

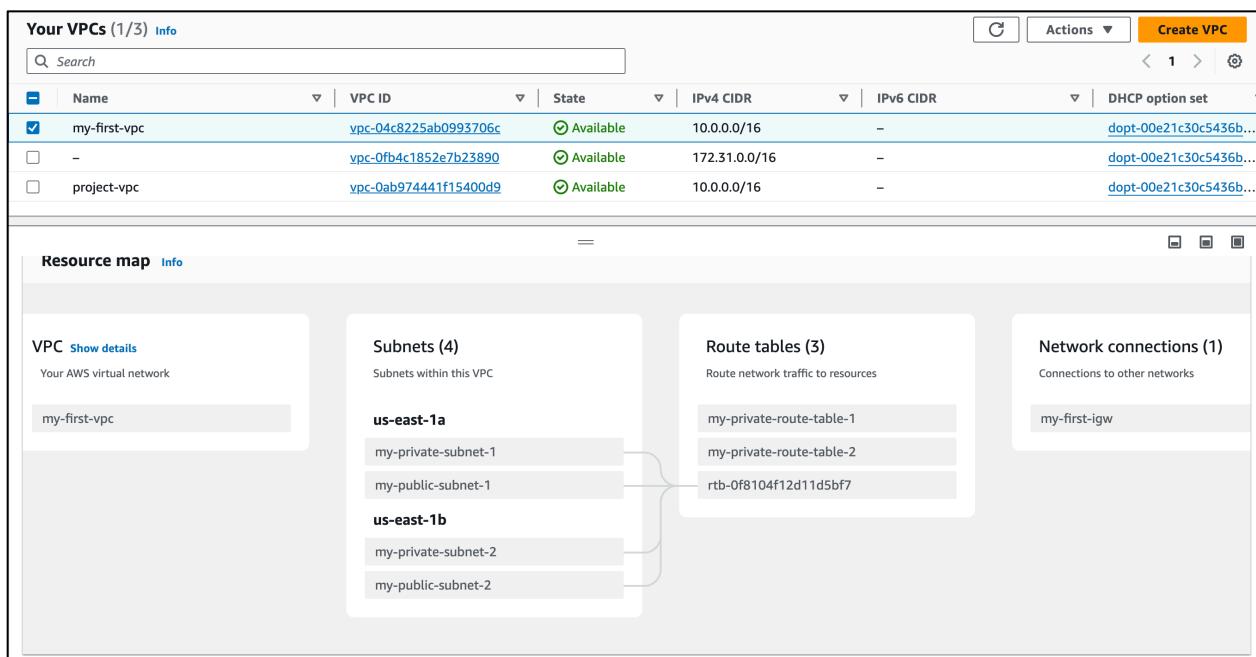
**Tags**  
A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Key	Value - optional
<input type="text" value="Name"/> <input type="button" value="X"/>	<input type="text" value="my-private-route-table-1"/> <input type="button" value="X"/> <input type="button" value="Remove"/>

You can add 49 more tags.

## Build AWS Network

- q. Your task is to create another Route Table named **my-private-route-table-2**.
- r. Now, the updated resource map looks like it, as shown below. See the connection between subnets and route tables. The communication or transfer of data still occurs through the default route table. Now, we make the necessary changes to build the perfect network.
- s. Let's review the map after creating the route tables.



### Step 5: Associate route tables to the subnets appropriately.

- t. Since we have the default route table we want to use for public communication; we make changes in the private route tables alone.
- u. Let's edit the subnet association and add private subnet – 1 to this route table.

## Build AWS Network

The screenshot shows the AWS VPC Route Tables page. The route table ID is rtb-01f44331f86480a9f, and it is associated with my-first-vpc. It has no explicit subnet associations or edge associations.

Name	Owner ID	Explicit subnet associations	Edge associations
No	vpc-04c8225ab0993706c   my-first-vpc	-	-

Subnet associations tab is selected. There are no explicit subnet associations listed.

The screenshot shows the 'Edit subnet associations' dialog for route table rtb-01f44331f86480a9f. It lists available subnets and selected subnets.

Name	Subnet ID	IPv4 CIDR	IPv6 CIDR	Route table ID
my-private-subnet-1	subnet-0f4deb81b18727418	10.0.3.0/24	-	Main (rtb-0f8104f12d11d5bf7)
my-public-subnet-1	subnet-0221cc7de703d33a8	10.0.10.0/24	-	Main (rtb-0f8104f12d11d5bf7)
my-private-subnet-2	subnet-0ab9d23345d2b6283	10.0.4.0/24	-	Main (rtb-0f8104f12d11d5bf7)
my-public-subnet-2	subnet-02c985fa8331ee838	10.0.2.0/24	-	Main (rtb-0f8104f12d11d5bf7)

Selected subnets: subnet-0f4deb81b18727418 / my-private-subnet-1

Buttons: Cancel, Save associations

- v. Your task is to associate the private route table 2 with **my-private-subnet-2**.
- w. Let's review the Resource map now. This is what my map looks like right now. See, my public subnets are still associated with the default route table. So, we don't need to associate anything. But do you think about the communication between public resources and the internet? We have to add a route connecting to the internet gateway.

## Build AWS Network

**Your VPCs (1/2) Info**

Name	VPC ID	State	IPv4 CIDR	IPv6 CIDR	DHCP option set
my-first-vpc	vpc-04c8225ab0993706c	Available	10.0.0.0/16	-	dopt-00e21c30c5436b..
-	vpc-0fb4c1852e7b23890	Available	172.31.0.0/16	-	dopt-00e21c30c5436b..

**Resource map** Info

**VPC** Show details Your AWS virtual network my-first-vpc

**Subnets (4)** Subnets within this VPC

- us-east-1a**
  - my-private-subnet-1
  - my-public-subnet-1**
- us-east-1b**
  - my-private-subnet-2
  - my-public-subnet-2**

**Route tables (3)** Route network traffic to resources

- my-private-route-table-1
- my-private-route-table-2
- rtb-0f8104f12d11d5bf7**

**Network connections (1)** Connections to other networks my-first-igw

- x. Let's edit the route option by adding the internet gateway to the default route table. Only after adding this route do our public servers placed in the public subnets communicate to the internet.

**Route tables (1/4) Info**

Name	Route table ID	Explicit subnet associ...	Edge associations	Main	VPC
my-private-route-table-1	rtb-01f44331f86480a9f	subnet-0f4deb81b18727...	-	No	vpc-04c8225ab0993706c   my-first-vpc
my-private-route-table-2	rtb-0966be51bc1bdbf50	subnet-0ab9d23345d2b6...	-	No	vpc-04c8225ab0993706c   my-first-vpc
<b>-</b>	<b>rtb-0f8104f12d11d5bf7</b>	-	-	Yes	vpc-04c8225ab0993706c   my-first-vpc
-	rtb-02ae7471ccbf80341	-	-	Yes	vpc-0fb4c1852e7b23890

**rtb-0f8104f12d11d5bf7**

Details | **Routes** | Subnet associations | Edge associations | Route propagation | Tags

**Routes (1)**

Destination	Target	Status	Propagated
10.0.0.0/16	local	Active	No

## Build AWS Network

VPC > Route tables > rtb-0f8104f12d11d5bf7 > Edit routes

Edit routes

Destination	Target	Status	Propagated
10.0.0.0/16	local	Active	No
<input type="text" value="0.0.0.0"/>		-	No

Add route

Cancel Preview Save changes

Destination Target Status Propagated

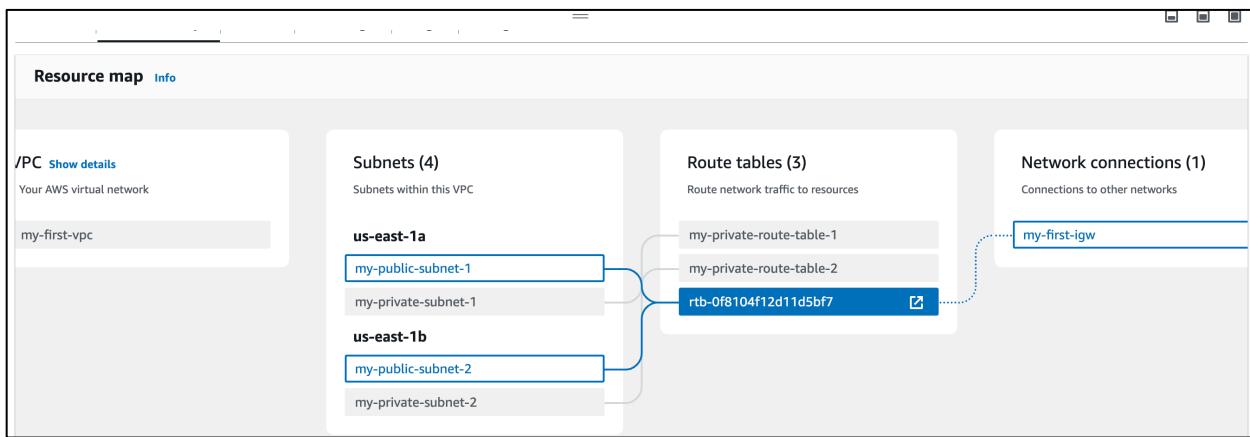
10.0.0.0/16 local Active No

Internet Gateway - No Remove

Add route

Cancel Preview Save changes

y. Let's review our Resource Map again.



z. Do you think about how the servers in the private subnet communicate to the internet when needed? NAT Gateways is the answer. We will place this Nat Gateway in the public subnet.

## Build AWS Network

Let's create one.

VPC > NAT gateways > Create NAT gateway

### Create NAT gateway Info

A highly available, managed Network Address Translation (NAT) service that instances in private subnets can use to connect to services in other VPCs, on-premises networks, or the internet.

#### NAT gateway settings

##### Name - *optional*

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

The name can be up to 256 characters long.

##### Subnet

Select a subnet in which to create the NAT gateway.



##### Connectivity type

Select a connectivity type for the NAT gateway.

- Public
- Private

##### Elastic IP allocation ID Info

Assign an Elastic IP address to the NAT gateway.

#### ► Additional settings Info

## Build AWS Network

**Connectivity type**  
Select a connectivity type for the NAT gateway.

Public  
 Private

**Elastic IP allocation ID** [Info](#)  
Assign an Elastic IP address to the NAT gateway.

eipalloc-05f8292d6aecf0f0c [▼](#) [Allocate Elastic IP](#)

**► Additional settings** [Info](#)

**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	Remove
<input type="text"/> Name <a href="#">X</a>	<input type="text"/> my-nat-gateway <a href="#">X</a>	<a href="#">Remove</a>

[Add new tag](#)

You can add 49 more tags.

[Cancel](#) [Create NAT gateway](#)

aa. Remember, we have added a route from the Main(public) Route Table to the internet gateway. Similarly, we will add a route from the private route table to the Nat gateway.

VPC > [Route tables](#) > rtb-01f44331f86480a9f

### rtb-01f44331f86480a9f / my-private-route-table-1 [Actions ▾](#)

**Details** [Info](#)

Route table ID <a href="#">rtb-01f44331f86480a9f</a>	Main <input type="checkbox"/> No	Explicit subnet associations <a href="#">subnet-0f4deb81b18727418 / my-private-subnet-1</a>	Edge associations -
VPC <a href="#">vpc-04c8225ab0993706c   my-first-vpc</a>	Owner ID <a href="#">506960532205</a>		

[Routes](#) [Subnet associations](#) [Edge associations](#) [Route propagation](#) [Tags](#)

**Routes (1)** [Edit routes](#)

Destination	Target	Status	Propagated
10.0.0.0/16	local	<a href="#">Active</a>	No

## Build AWS Network

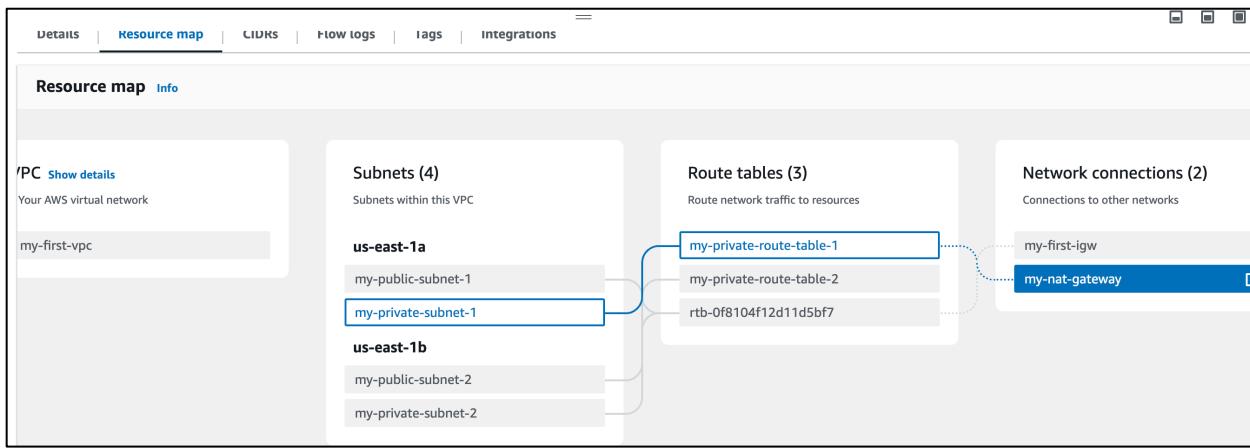
VPC > Route tables > rtb-01f44331f86480a9f > Edit routes

### Edit routes

Destination	Target	Status	Propagated	
10.0.0.0/16	local	Active	No	
<input type="text" value="0.0.0.0/0"/>	NAT Gateway	-	No	<button style="border: 1px solid #ccc; padding: 2px 10px;">Remove</button>
	<input type="text" value="nat-09cfb6b49bd6382a6"/>			

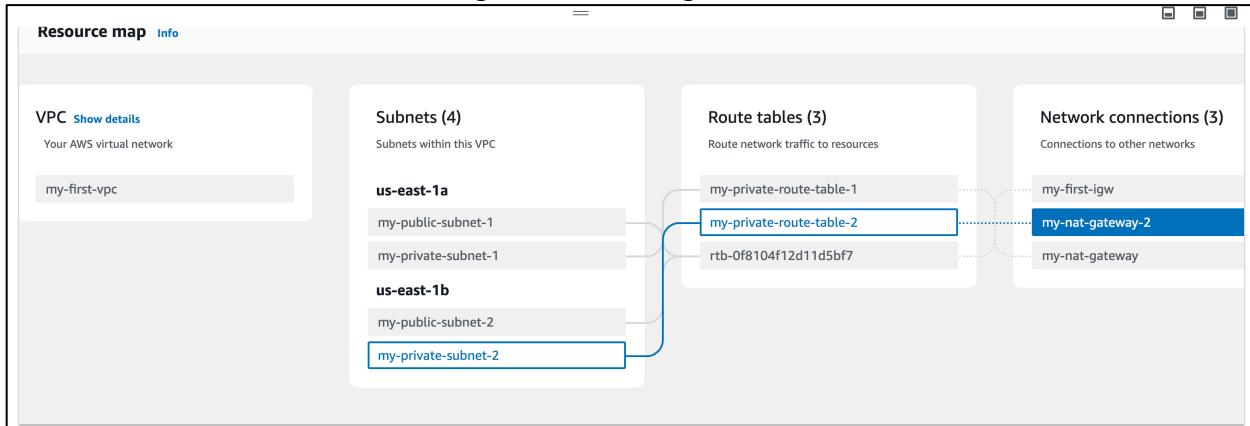
[Add route](#)

[Cancel](#) [Preview](#) [Save changes](#)



bb. Now, your task is to create another Nat gateway named my-nat-gateway-2 and perform the steps mentioned above. The Nat gateway two should be linked to the private route table 2 and private subnet 2.

cc. Here is the final Network Diagram of our assignment.



**Deliverables:** Screenshots at points g, n, s, y, and cc must be submitted. A total of five screenshots should be submitted.