AWS VPC (Virtual Private Cloud)

It is related to Network and Security part

VPC provides isolated network for resources in AWS cloud
Is AWS a public or private cloud -> it is a public (anyone can access the AWS website, create acount, use resources)
Can anyone login and work with resources in AWS? Yes, then it is a public cloud

In that case, protecting my resources is very crucial. If I login into my account, can I access other accounts' resources, answer is no

With my login, I shouldn't access your resources and vice-versa. that's called as isolated resources

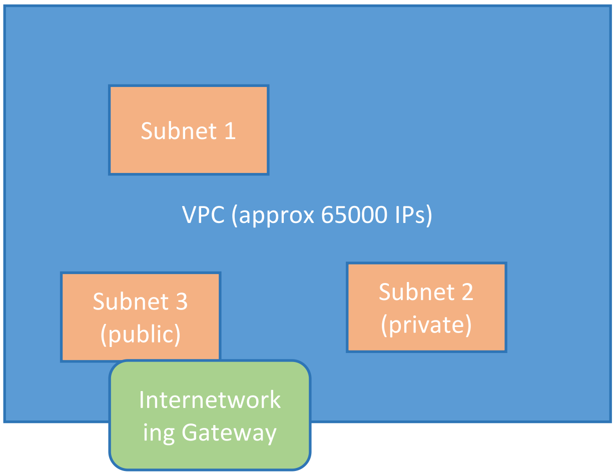
With the help of VPC, we can protect our resources in AWS cloud

VPC provides flexible and secured network to maintain and manage our resources in AWS cloud

Can everyone access every resource available in your home? The answer is No. Mostly outsiders can access only up to front door

VPC is like our home, before creating the resources, first part is to create a VPC

Even though, we haven't created our own VPC, still while creating other resources, we used the default VPC being created. Maximum we can create 65000 IPs, that many IPs we can allow within one VPC. Maximum approx 65000 IPs will be there in one VPC



We can have any number of Subnets inside VPCs, we have something called as Public Subnet and Private Subnet

VPC terminology:
1. VPC
2. Subnet (Private or Public)
3. CIDR Block (IP ranges) -> VPC sizing
4. Route tables
5. Internet gateway
6. NAT gateway
7. VPC peering
8. Security groups --> at Resource level --> Rules (both Inbound [who can access] and Outbound [what we can access]) to allow who can access
9. NACL --> Subnet level --> but at Subnet level, we can have rules for allowing also and denying also

What's one Subnet? Like a small network within VPC. We can allocate like 256 IPs in the subnet.
In Subnet SN2, we can allocate 124 IPs

Private Subnet is for the internal communication
Public subnet can be accessed through internetworking. Internetworking means outsiders can also access. Say if I attached an Internetworking Gateway (INW) to a public subnet, anyone can access. Any of the Subnet, if I attach Internet Gateway (INW), it becomes Public otherwise it is Private
Public subnet --> Internet Gateway, Private subnet --> NAT Gateway

IPs
There are several types of IPs (Internet Protocol) address used in a computer network. All devices like mobile phones, laptops, desktops etc all have IPs. If devices have to get connected over internet, IP address is a must
IPv4: 32 bits --> 4 sets of numbers
IPv4 ---> 32 bits numeric addresses written in four sets of numbers separated by periods: 192.168.1.1
It is most widely used IP version and supports approximately 4.3 billion unique addresses
IPv6: 128 bits, IPv6 is trying to solve the problem of IP shortage, which might occur in the future due to IPv4 --> 8 sets of four hexadecimal digits will be there
IPv6 uses hexadecimal notation with colons eg: 2001:0db8:85a3:0000:0000:8a2e:0370:7334
Supports 340 undecillion unique addresses

Public IP
Private IP
Static IP address
Dynamic IP

VPC Sizing: process of allocating IPs to VPC subnets. In general, allocating IPs is called as VPC sizing, within VPC how many IPs are required and within each subnet how many IPs are required. When we create a new VPC, that time we decide how many IPs we want --> Decision of allocating IPs to VPCs is only called as VPC sizing

IP Ranges we will be doing with CIDR --> Class less Inter domain Range

VPCs are regional-specific
Can VPCs interact? Yes that's called as VPC peering

Any device, if it has to access resources of internet, is IP compusory? Yes

On a regular basis, number of new devices are rapidly increasing and are using internet, and any device if it has to use internet, then IP is mandatory and a must and there might be possibility of running out of IPs, which has to be unique. To overcome this issue, IPv6 was introduced

When we connect Internet Gateway to a Private Subnet, it becomes Public Subnet. Also Route Table (RT) will be there between INW and Public SN1 to make it public. Rules of routing, with the help of route tables, INW, outsiders can access resources within one Subnet

VPC sizing: Sizing will be calculated in 2 power or power of 2

IPv4
10.0.0.1/16 ==> means 2 power (32-16) ==> 2 power 16. 32 is the maximum . Maximum one VPC can have is 65,536
10.0.0.1/32 ==> 2 power (32-32) ==> 2 power 0 = 1 IPs address, not recommended to create VPC or Subnet with just one IP address
Therefore, AWS supports from 28
10.0.0.1/28 ==> 2 power (32-28) ==> 2 power 4 = 16. this is the minimum number of IPs AWS supports.
AWS supports from 28 and less

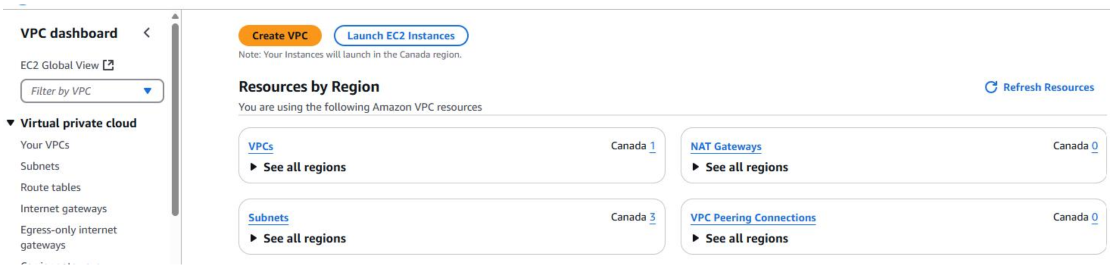
AWS supports minimum 16 to maximum 65,536 IPs
Similarly, AWS does not support ==> 10.0.0.1/15 less than 16

Subnet ranges can be given from /16 to /28

Recommended to use /24
10.0.0.1/24 ==> 2 power (32-24) = 2 power 8 = 256 IPs that's generally we use. In real-life scenario also 256 IP is more than enough for Subnets

VPC allows us to create and manage our own networks or isolated networks within AWS cloud
In VPC, users can define their own IP address range, subnets, route table and network gateways
It provides control over network configuration such as setting up access-control policies, firewall rules (security group rules) and network traffic routing
MobaXterm is an outsider, not part of AWS, if you are able to connect with MobaXTerm it is a public subnet, otherwise private subnet

Practical task:
1. Create VPC
    a) CIDR block: 10.0.0.0/16
    b) Select No IPv6 CIDR block
    c) Rest select default options
    d) Click create VPC
    e) Note: One Route table will be created for VPC by default if necessary re-name
2. Create two subnets ->
    a) Create subnet-1
        i. Name: subnet-public
        ii. CIDR block: 10.0.0.0/24 (it will take 256 IPs)
    b) Create subnet-2
        i. Name: subnet-private
        ii. CIDR block: 10.0.1.0/24 (I will slightly change IP to avoid collision)
    c) Route tables: The moment we create VPC default route table will be created go to route table section and rename it to private
    d) Create a NEW Route table and name it to Public
    e) After Route tables are created, Edit and attach Subnet association
    f) Public RT attach to Public subnet
    g) Private RT Attach to Private subnet
    h) Create new Internet Gateway
    i) Click on Public-RT --> routes --> Add route ---> Internet Gateway and select the IGW we had created earlier and select 0.0.0.0/0 in the other input field
3. Create 2 EC2s
    a) One EC2 in public subnet and another in private subnet
    b) Tested by connect to EC2 of public subnet, it worked and we were NOT able to connect to Private EC2 / private subnet because no Internet Gateway was attached



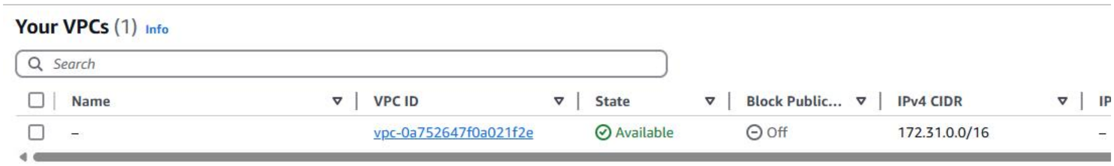
Note: Your Instances will launch in the Canada region.

Resources by Region

You are using the following Amazon VPC resources

VPCs	Canada 1
NAT Gateways	Canada 0
Subnets	Canada 3
VPC Peering Connections	Canada 0

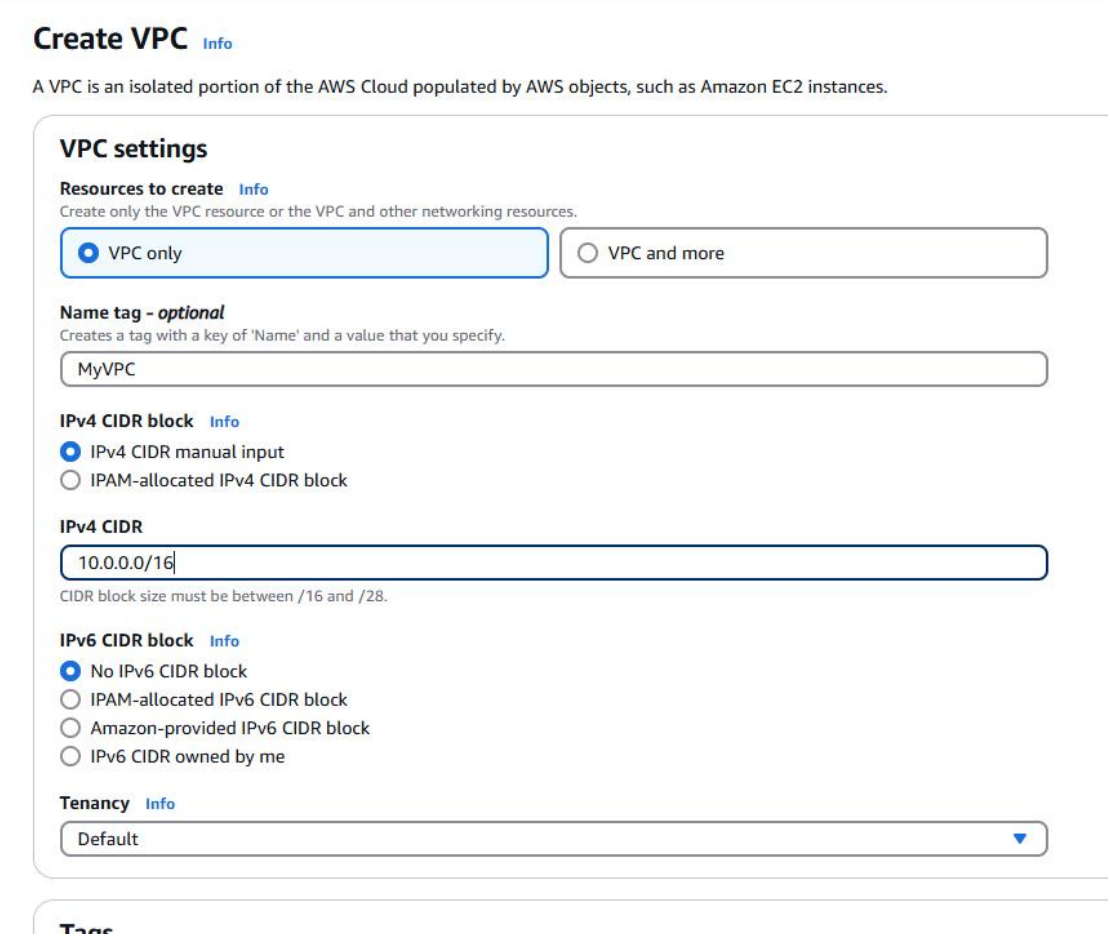
One default VPC is already there



Your VPCs (1) Info

	Name	VPC ID	State	Block Public...	IPv4 CIDR	IP
□	–	vpc-0a752647f0a021f2e	Available	Off	172.31.0.0/16	–

Use the following settings: 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.

MyVPC

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

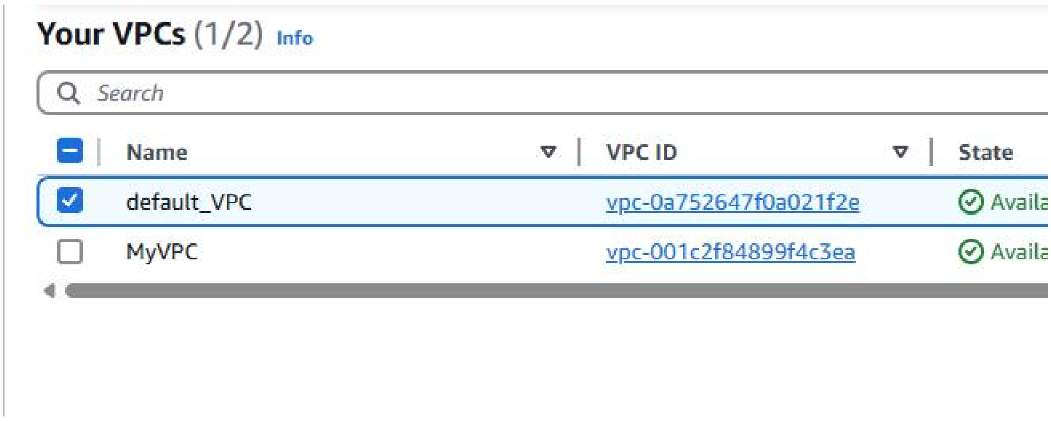
○ IPv6 CIDR owned by me

Tenancy Info

Default

Tags

10.0.0.0/16 means almost 65000 IPs are allowed in this VPC

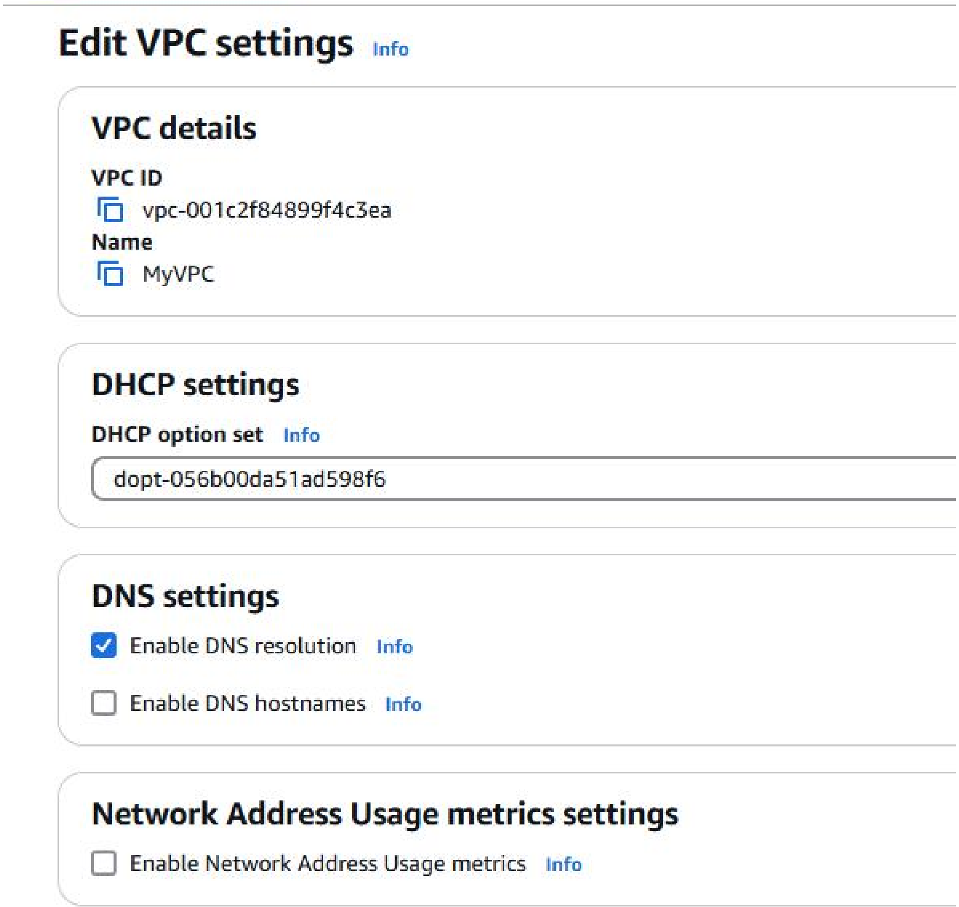


Your VPCs (1/2) Info

Search

☐	Name	VPC ID	State
☑	default_VPC	vpc-0a752647f0a021f2e	✓ Availa
☐	MyVPC	vpc-001c2f84899f4c3ea	✓ Availa

Edit VPC settings



Edit VPC settings Info

VPC details

VPC ID

vpc-001c2f84899f4c3ea

Name

MyVPC

DHCP settings

DHCP option set Info

dopt-056b00da51ad598f6

DNS settings

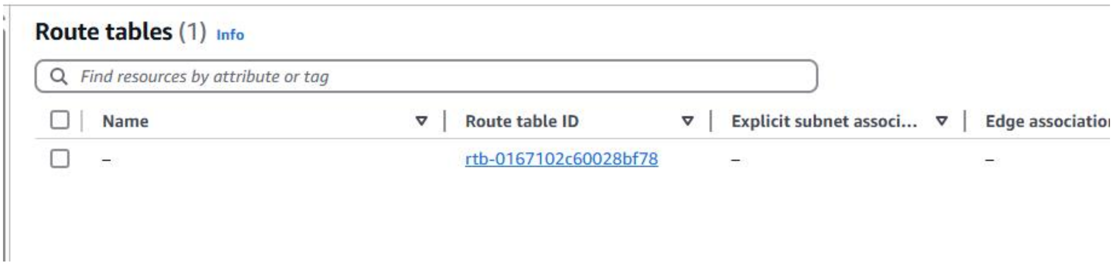
☑ Enable DNS resolution Info

☐ Enable DNS hostnames Info

Network Address Usage metrics settings

☐ Enable Network Address Usage metrics Info

Click Route tables

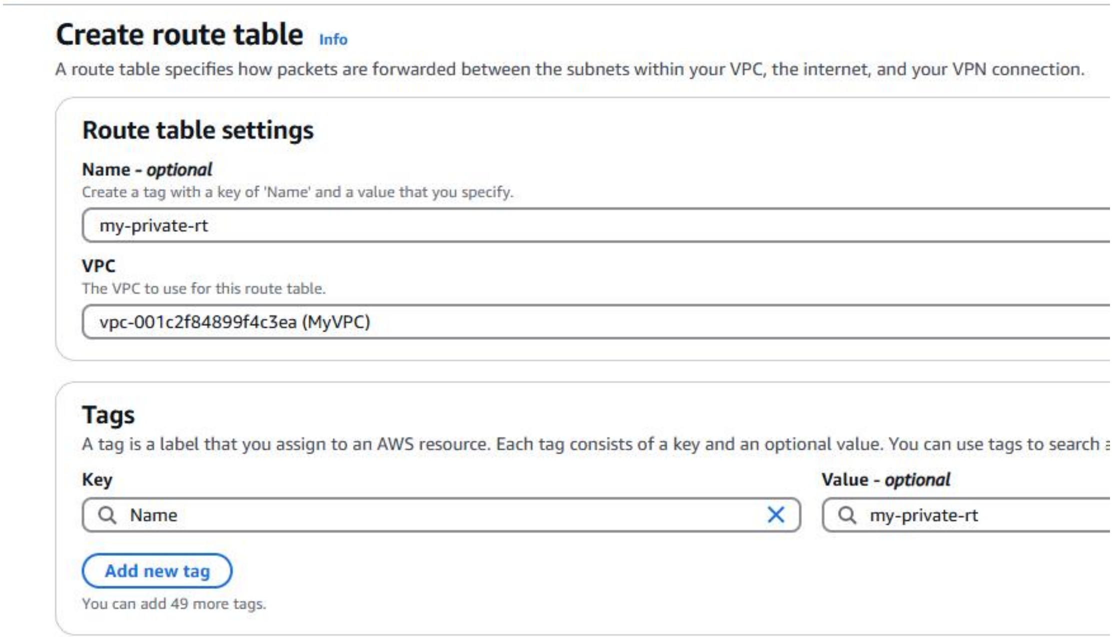


Route tables (1) Info

Find resources by attribute or tag

□	Name	Route table ID	Explicit subnet associ…	Edge associatio
□	–	rtb-0167102c60028bf78	–	–

Create Route table



Create route table

A route table specifies how packets are forwarded between the subnets within your VPC, the internet, and your VPN connection.

Route table settings

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

VPC
The VPC to use for this route table.
vpc-001c2f84899f4c3ea (MyVPC)

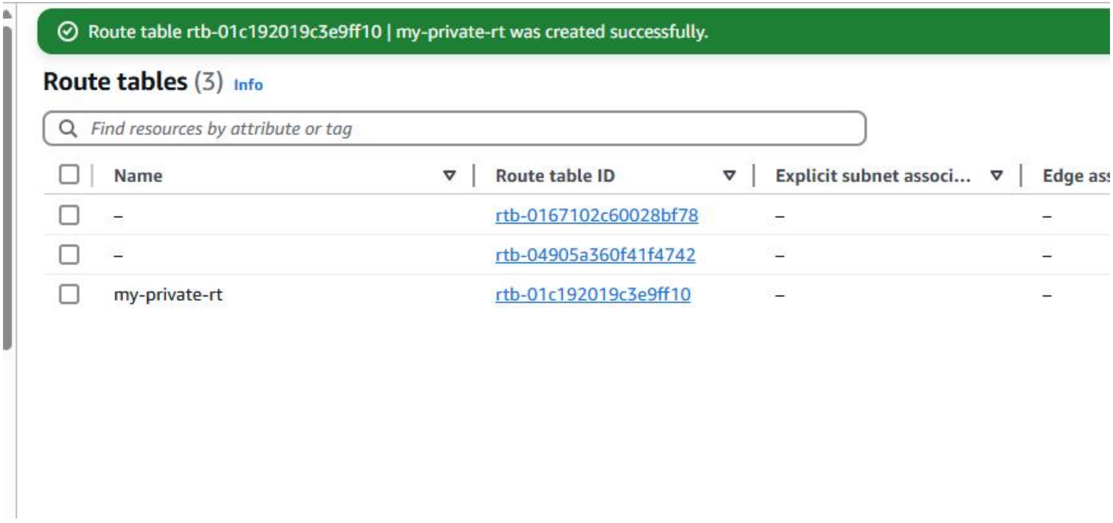
Tags

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 a

Key	Value - *optional*
Name	my-private-rt

You can add 49 more tags.

Click route table
Note: One Route table is created along with VPC by default



Route table rtb-01c192019c3e9ff10 | my-private-rt was created successfully.

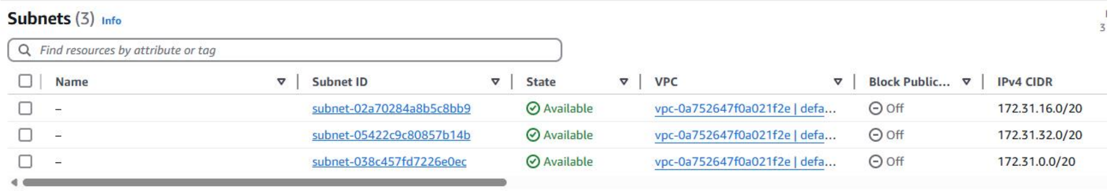
Route tables (3) Info

Find resources by attribute or tag

□	Name	Route table ID	Explicit subnet associ…	Edge ass
□	–	rtb-0167102c60028bf78	–	–
□	–	rtb-04905a360f41f4742	–	–
□	my-private-rt	rtb-01c192019c3e9ff10	–	–

Go to Subnets

Default subnets are there

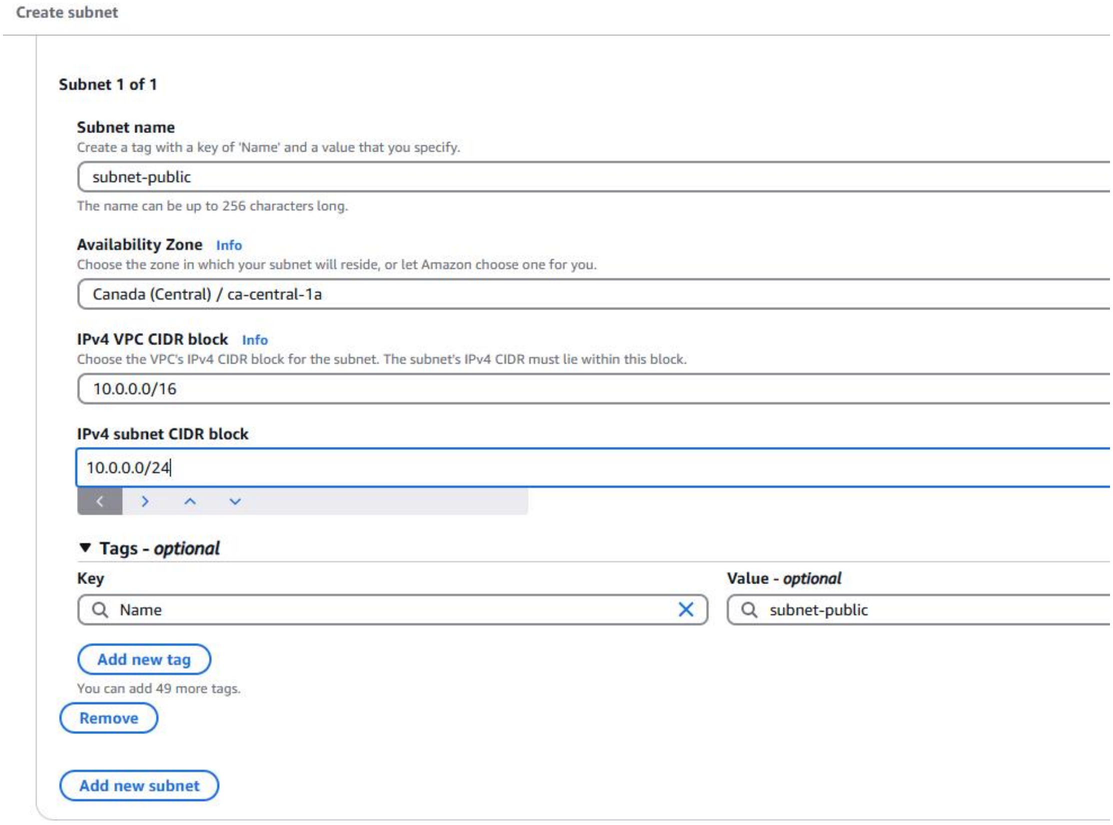


Subnets (3) Info

Find resources by attribute or tag

	Name	Subnet ID	State	VPC	Block Public...	IPv4 CIDR
□	–	subnet-02a70284a8b5c8bb9	✓ Available	vpc-0a752647f0a021f2e | defa...	⊖ Off	172.31.16.0/20
□	–	subnet-05422c9c80857b14b	✓ Available	vpc-0a752647f0a021f2e | defa...	⊖ Off	172.31.32.0/20
□	–	subnet-038c457fd7226e0ec	✓ Available	vpc-0a752647f0a021f2e | defa...	⊖ Off	172.31.0.0/20

Create a new Subnet



Create subnet

Subnet 1 of 1

Subnet name
Create a tag with a key of 'Name' and a value that you specify.
subnet-public
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.
Canada (Central) / ca-central-1a

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.
10.0.0.0/16

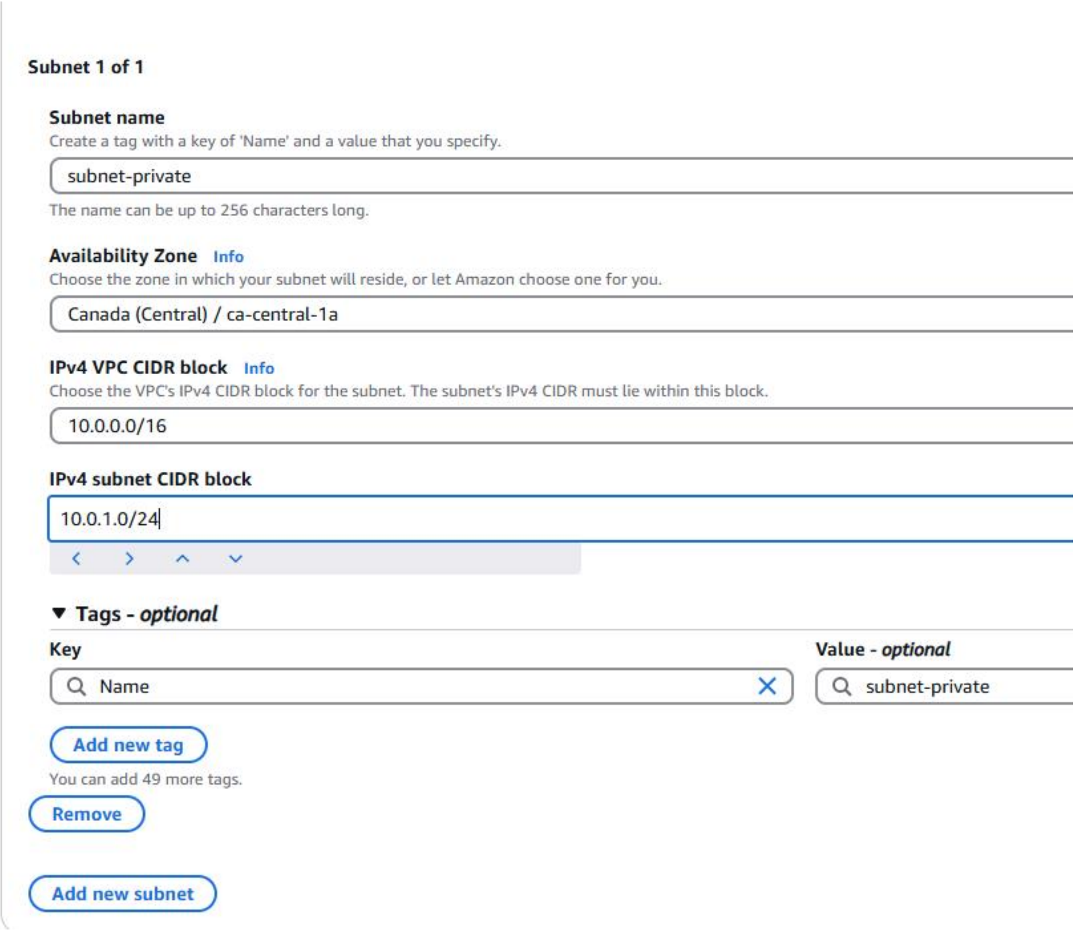
IPv4 subnet CIDR block
10.0.0.0/24

▼ Tags - *optional*

Key	Value - *optional*
Name	subnet-public

You can add 49 more tags.

10.0.0.0/24 ---> IPv4 subnet CIDR block



Subnet 1 of 1

Subnet name
Create a tag with a key of 'Name' and a value that you specify.
subnet-private
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.
Canada (Central) / ca-central-1a

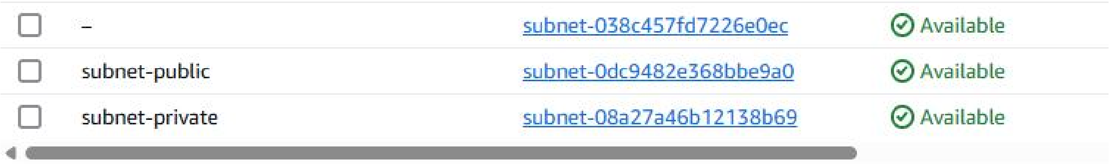
IPv4 VPC CIDR block Info
Choose the VPC's IPv4 CIDR block for the subnet. The subnet's IPv4 CIDR must lie within this block.
10.0.0.0/16

IPv4 subnet CIDR block
10.0.1.0/24

▼ Tags - *optional*

Key	Value - *optional*
Name	subnet-private

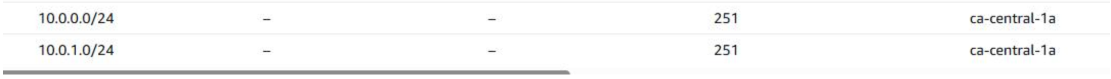
You can add 49 more tags.



□	–	subnet-038c457fd7226e0ec	✓ Available
□	subnet-public	subnet-0dc9482e368bbe9a0	✓ Available
□	subnet-private	subnet-08a27a46b12138b69	✓ Available

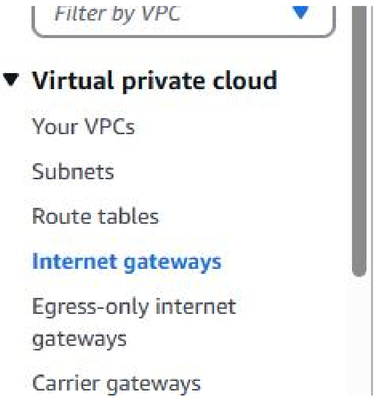
If you attach Internet gateway, it is public, if you don’t attach then it is private

We have 251 IP addresses in our subnets:

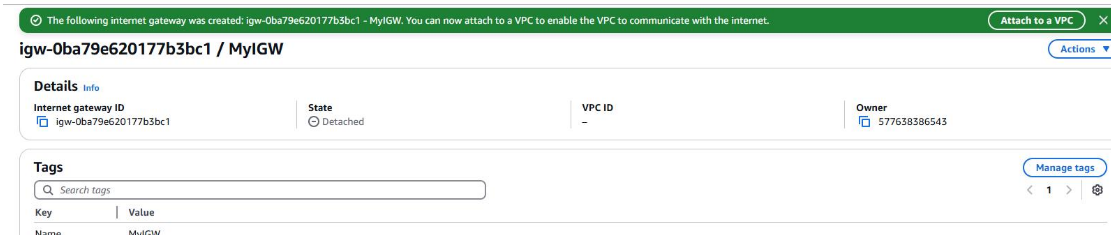


10.0.0.0/24	–	–	251	ca-central-1a
10.0.1.0/24	–	–	251	ca-central-1a

Go to Internet Gateways



Create Internet Gateway then Attach to VPC



The following internet gateway was created: igw-0ba79e620177b3bc1 - MyIGW. You can now attach to a VPC to enable the VPC to communicate with the internet.

igw-0ba79e620177b3bc1 / MyIGW

Details

Internet gateway ID: igw-0ba79e620177b3bc1

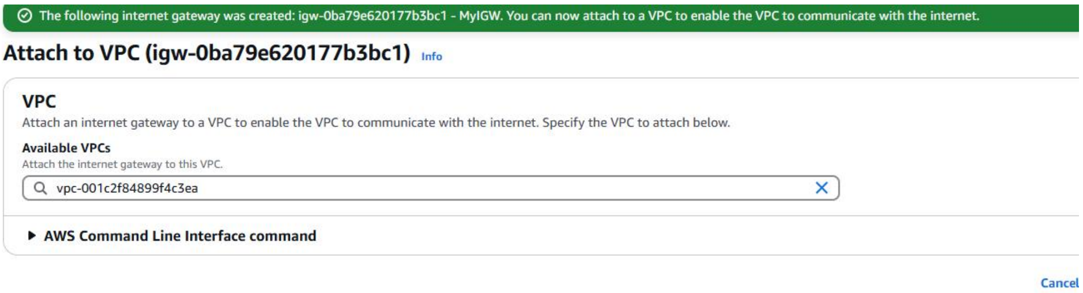
State: Detached

VPC ID: –

Owner: 577638386543

Tags

Key	Value
Name	MyIGW



The following internet gateway was created: igw-0ba79e620177b3bc1 - MyIGW. You can now attach to a VPC to enable the VPC to communicate with the internet.

Attach to VPC (igw-0ba79e620177b3bc1)

VPC

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

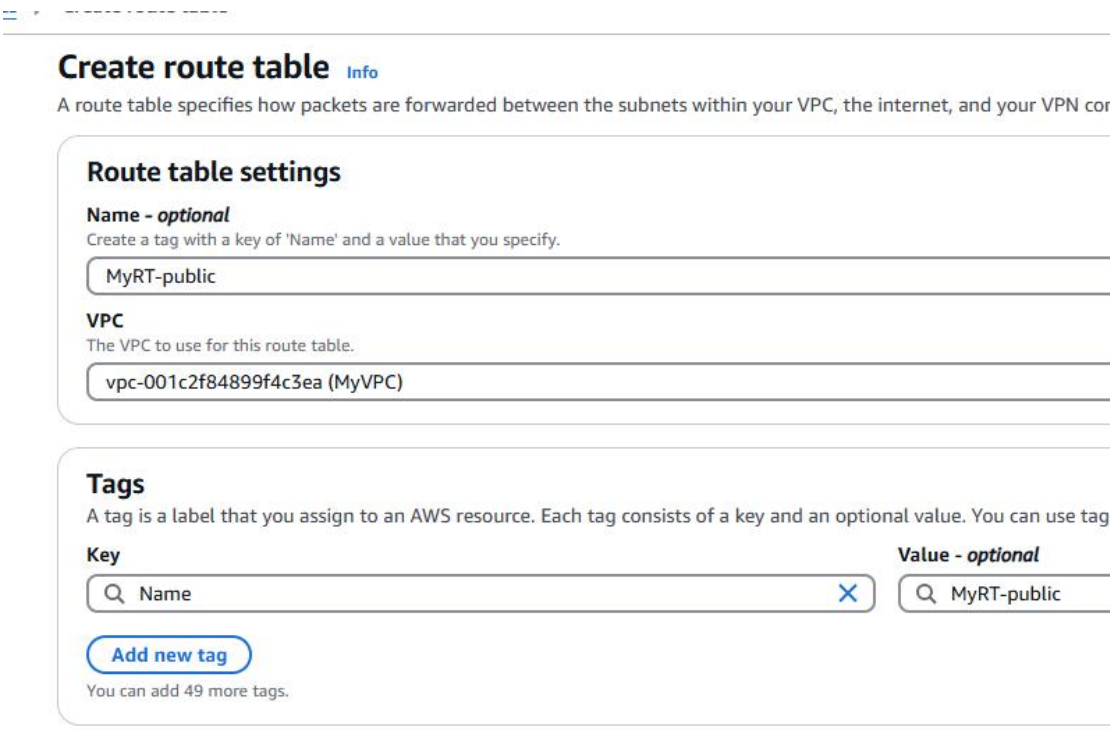
Available VPCs

Attach the internet gateway to this VPC.

vpc-001c2f84899f4c3ea

Select MyVPC
Click Attach internet gateway

Create a public route table



Create route table Info

A route table specifies how packets are forwarded between the subnets within your VPC, the internet, and your VPN cor

Route table settings

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

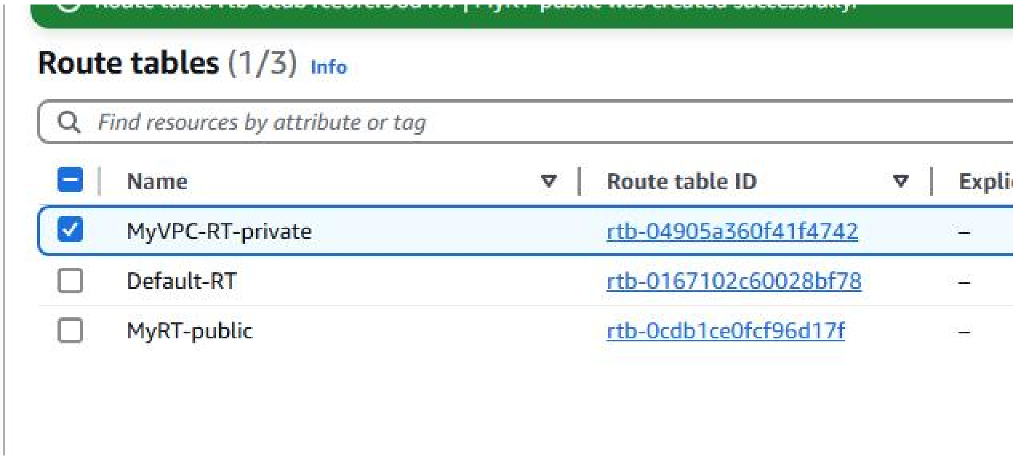
VPC
The VPC to use for this route table.
vpc-001c2f84899f4c3ea (MyVPC)

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 tag

Key	Value - *optional*
Name	MyRT-public

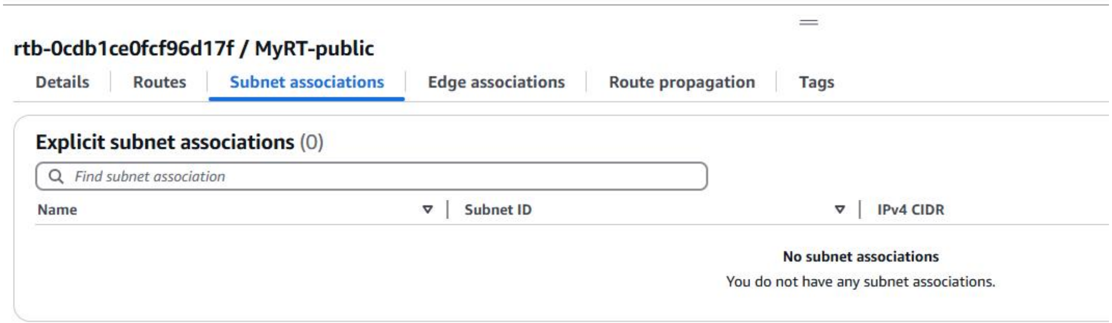
You can add 49 more tags.



Route tables (1/3) Info

☐	Name	Route table ID	Expli
☑	MyVPC-RT-private	rtb-04905a360f41f4742	–
☐	Default-RT	rtb-0167102c60028bf78	–
☐	MyRT-public	rtb-0cdb1ce0fcf96d17f	–

No subnet associations for Route table public



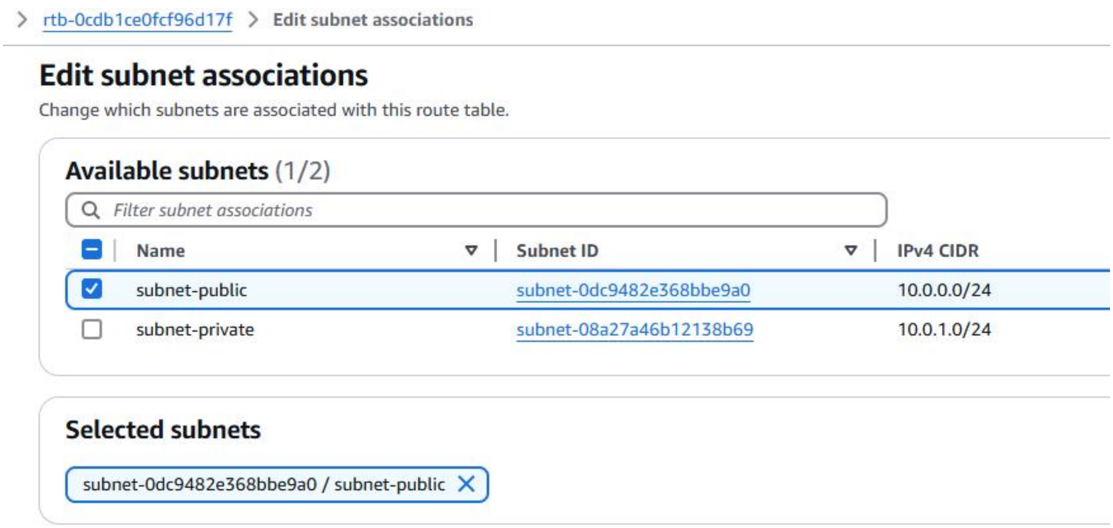
rtb-0cdb1ce0fcf96d17f / MyRT-public

Explicit subnet associations (0)

Name	Subnet ID	IPv4 CIDR

No subnet associations
You do not have any subnet associations.

Click Public subnet for public VPC



> rtb-0cdb1ce0fcf96d17f > Edit subnet associations

Edit subnet associations

Change which subnets are associated with this route table.

Available subnets (1/2)

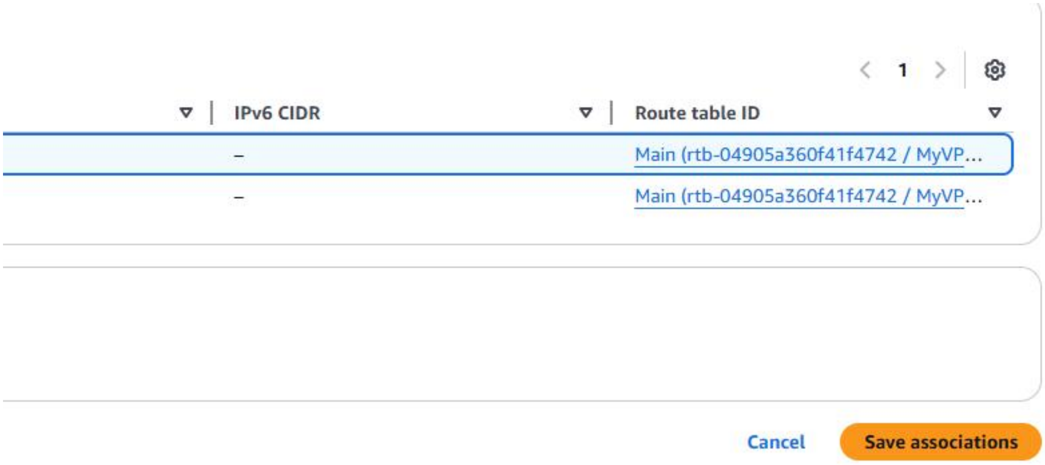
Filter subnet associations

☐	Name	Subnet ID	IPv4 CIDR
☑	subnet-public	subnet-0dc9482e368bbe9a0	10.0.0.0/24
☐	subnet-private	subnet-08a27a46b12138b69	10.0.1.0/24

Selected subnets

subnet-0dc9482e368bbe9a0 / subnet-public ✕

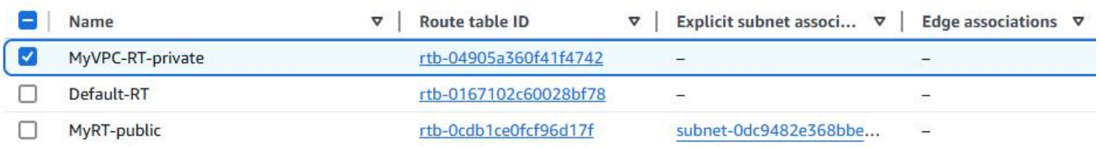
Click Save associations



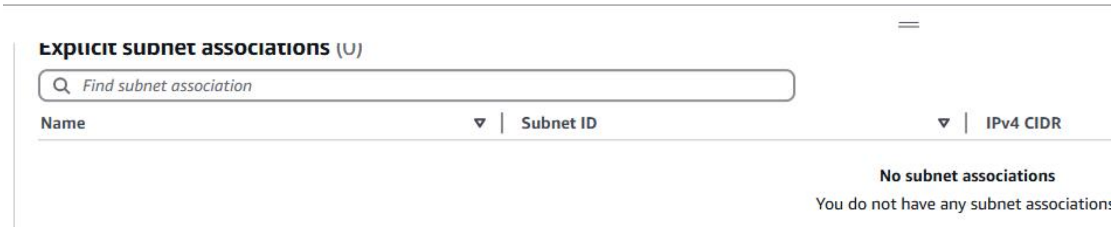
< 1 >

IPv6 CIDR	Route table ID
–	Main (rtb-04905a360f41f4742 / MyVP…
–	Main (rtb-04905a360f41f4742 / MyVP…

Same thing click on Private subnet --> no associations at the moment



⊟	Name	Route table ID	Explicit subnet associ…	Edge associations
☑	MyVPC-RT-private	rtb-04905a360f41f4742	–	–
□	Default-RT	rtb-0167102c60028bf78	–	–
□	MyRT-public	rtb-0cdb1ce0fcf96d17f	subnet-0dc9482e368bbe…	–

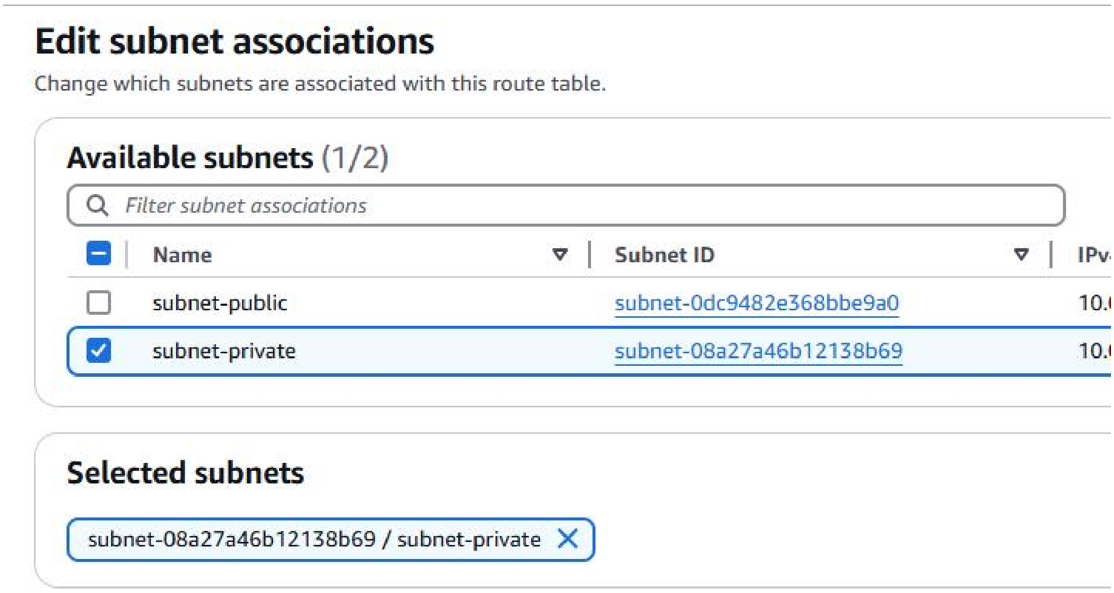


Explicit subnet associations (0)

Name	Subnet ID	IPv4 CIDR

No subnet associations

You do not have any subnet associations



Edit subnet associations

Change which subnets are associated with this route table.

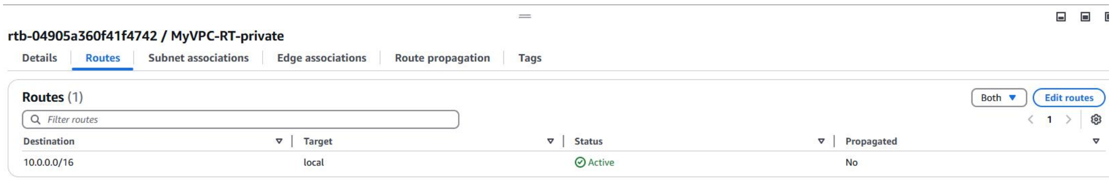
Available subnets (1/2)

⊟	Name	Subnet ID	IPv
□	subnet-public	subnet-0dc9482e368bbe9a0	10.(
☑	subnet-private	subnet-08a27a46b12138b69	10.(

Selected subnets

subnet-08a27a46b12138b69 / subnet-private ✕

Click on edit routes --> it is done only for the PUBLIC subnet

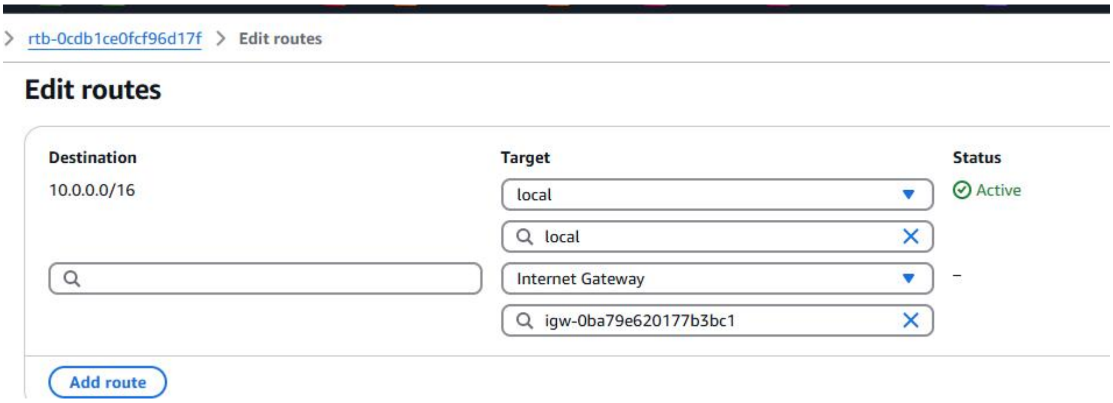


rtb-04905a360f41f4742 / MyVPC-RT-private

Routes (1)

Destination	Target	Status	Propagated
10.0.0.0/16	local	✓ Active	No

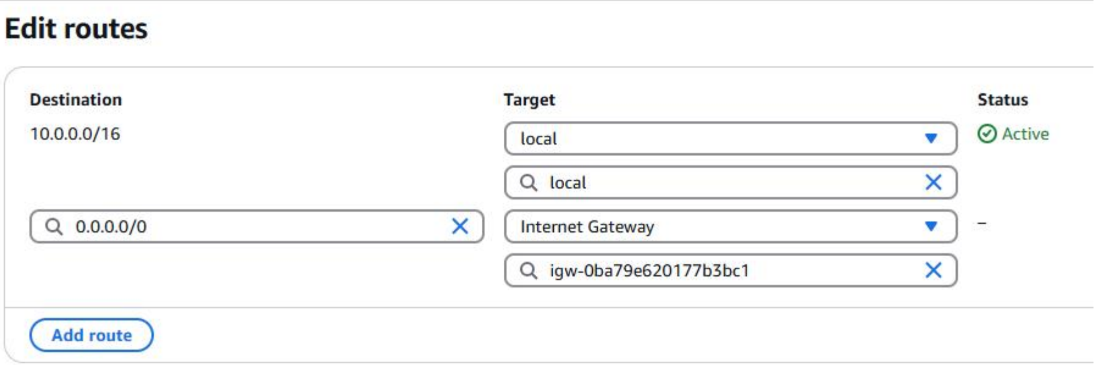
Add route ---> select Internet Gateway



> rtb-0cdb1ce0fcf96d17f > Edit routes

Edit routes

Destination	Target	Status
10.0.0.0/16	local local	Active
	Internet Gateway igw-0ba79e620177b3bc1	–

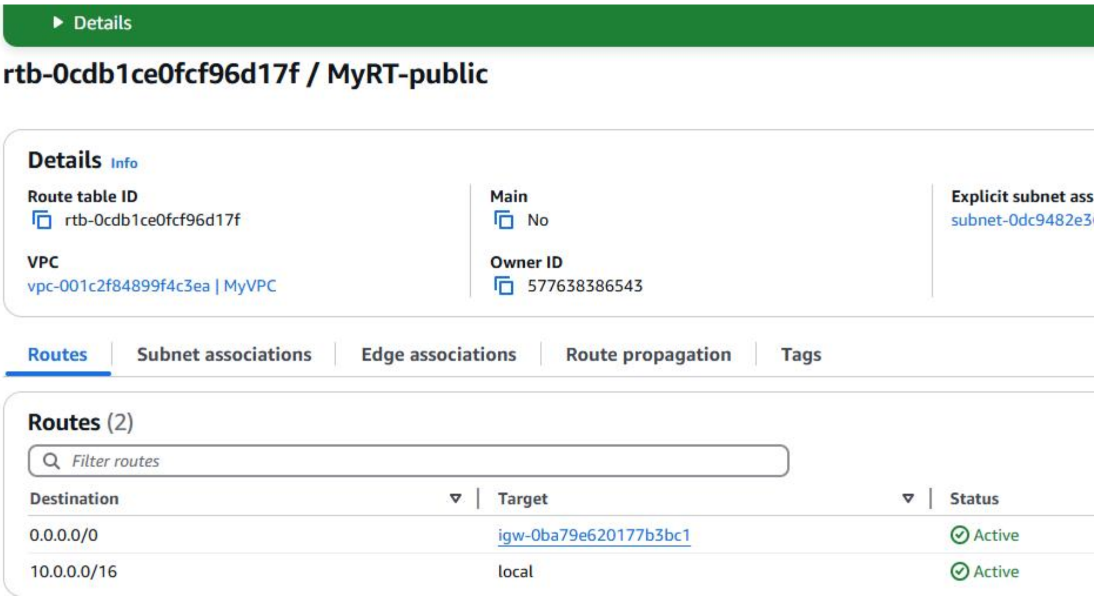


Edit routes

Destination	Target	Status
10.0.0.0/16	local local	Active
0.0.0.0/0	Internet Gateway igw-0ba79e620177b3bc1	–

Click Save changes

Internet Gateway associated only with the Public route table



rtb-0cdb1ce0fcf96d17f / MyRT-public

Details Info

Route table ID: rtb-0cdb1ce0fcf96d17f

VPC: vpc-001c2f84899f4c3ea | MyVPC

Main: No

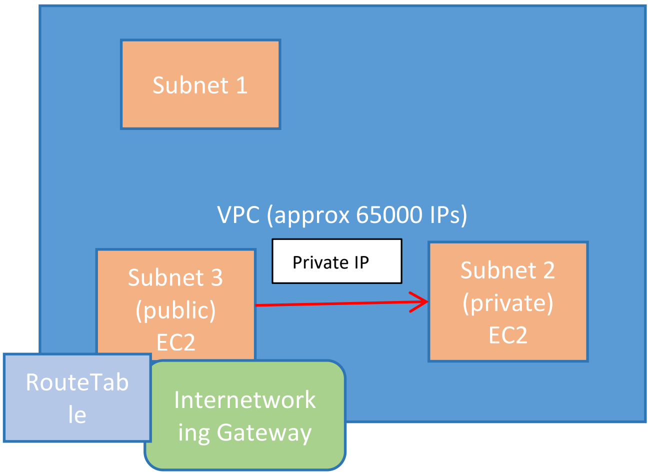
Owner ID: 577638386543

Explicit subnet ass: subnet-0dc9482e3

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

Routes (2)

Destination	Target	Status
0.0.0.0/0	igw-0ba79e620177b3bc1	Active
10.0.0.0/16	local	Active

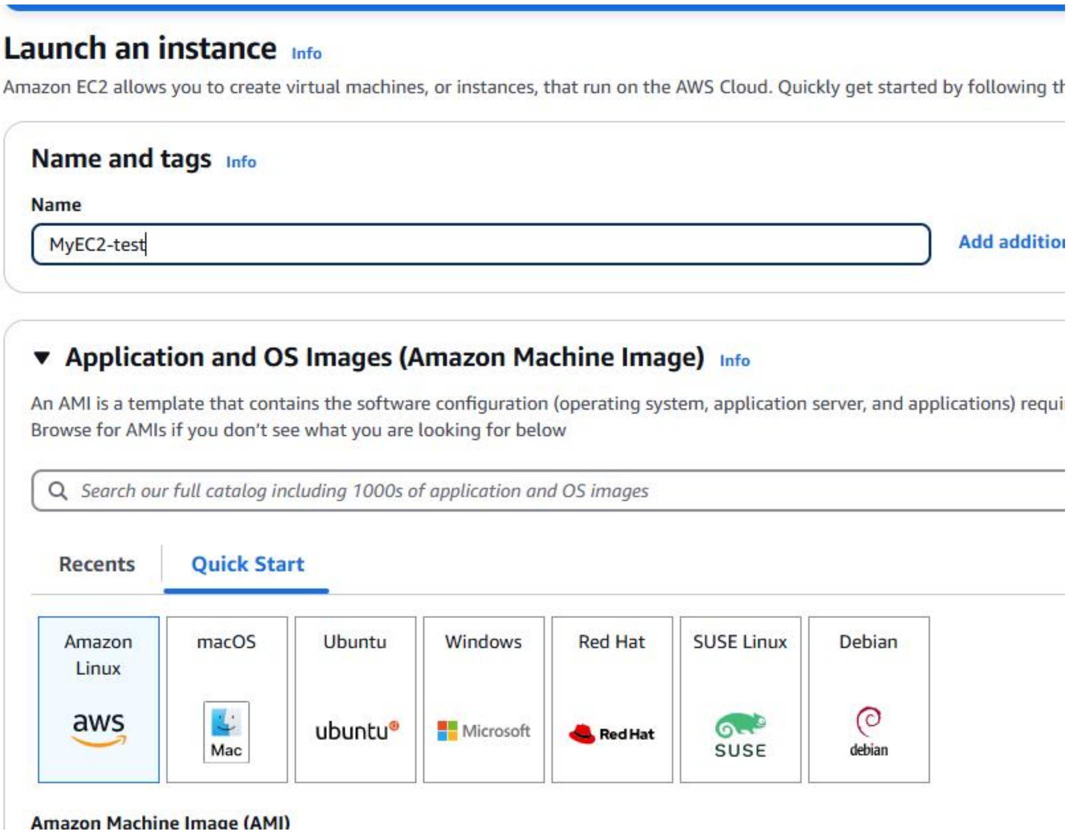


Note:
We have created a Public subnet, then Route tables --> then we have attached Internet gateway to RT

Route tables: The moment we create VPC default route table will be created go to route table section and rename it to private
Then Create a NEW Route table and name it to Public
After Route tables are created, Edit and attach Subnet association
Public RT attach to Public subnet
Private RT Attach to Private subnet
Create new Internet Gateway
Click on Public-RT --> routes --> Add route ---> Internet Gateway and select the IGW we had created earlier

Summary
1. Create VPC
2. Create 2 subnets
3. Create IGW
4. Create Route tables
5. Associate RT to respective Subnets
6. Attach IGW to Public RT
7. One EC2 in public subnet and another in private subnet
8. Tested by connect to EC2 of public subnet, it worked and we were NOT able to connect to Private EC2 / private subnet because no Internet Gateway was attached

Now go to EC2



Launch an instance Info

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following th

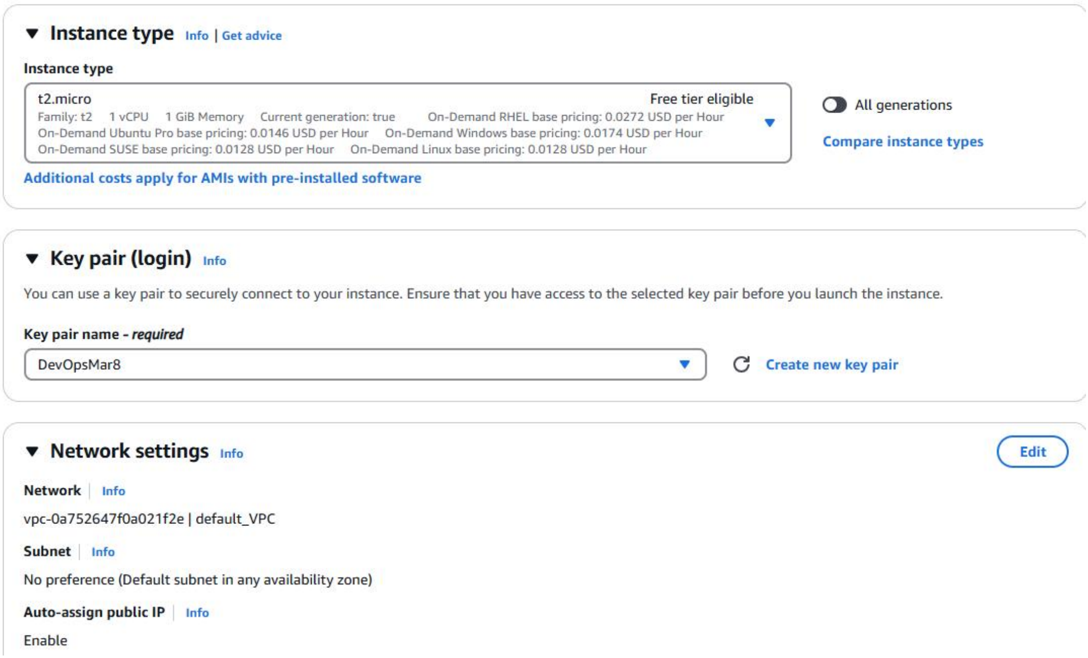
Name and tags Info

Name: MyEC2-test

▼ Application and OS Images (Amazon Machine Image) Info

An AMI is a template that contains the software configuration (operating system, application server, and applications) requi
Browse for AMIs if you don't see what you are looking for below

Amazon Machine Image (AMI)



▼ Instance type Info | Get advice

Instance type

t2.micro Free tier eligible
Family: t2 1 vCPU 1 GiB Memory Current generation: true On-Demand RHEL base pricing: 0.0272 USD per Hour
On-Demand Ubuntu Pro base pricing: 0.0146 USD per Hour On-Demand Windows base pricing: 0.0174 USD per Hour
On-Demand SUSE base pricing: 0.0128 USD per Hour On-Demand Linux base pricing: 0.0128 USD per Hour

Additional costs apply for AMIs with pre-installed software

All generations

Compare instance types

▼ Key pair (login) Info

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name - *required*

DevOpsMar8

Create new key pair

▼ Network settings Info

Edit

Network Info

vpc-0a752647f0a021f2e | default_VPC

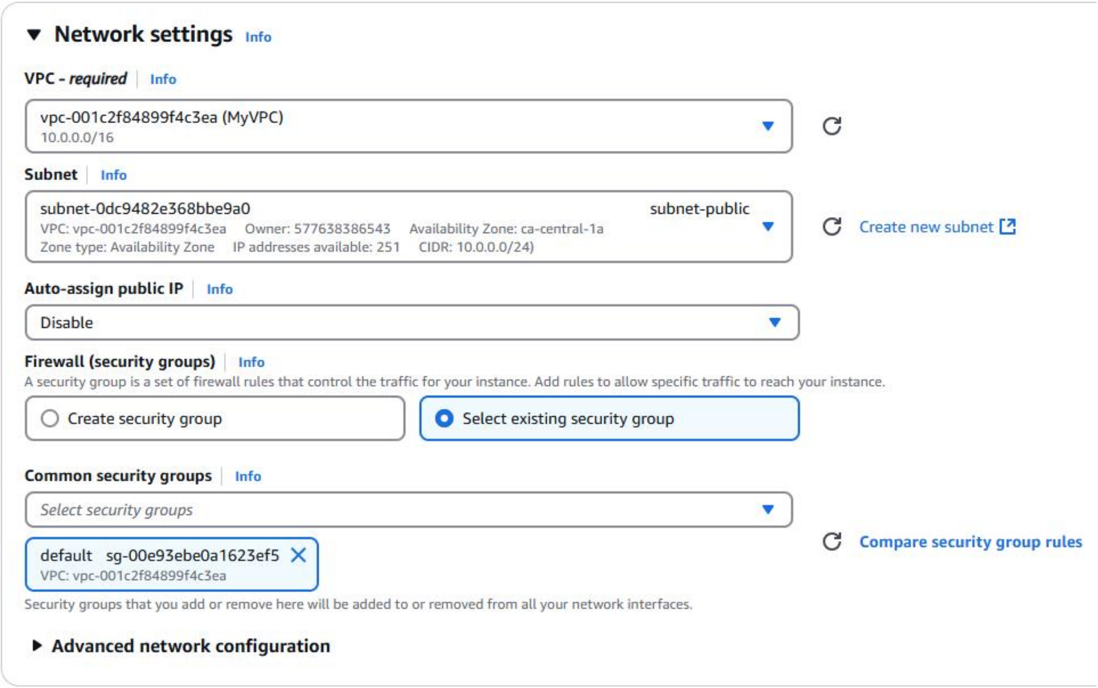
Subnet Info

No preference (Default subnet in any availability zone)

Auto-assign public IP Info

Enable

Change VPC to MyVPC, Select subnet-public, Security group default



▼ Network settings Info

VPC - *required* Info

vpc-001c2f84899f4c3ea (MyVPC)
10.0.0.0/16

Subnet Info

subnet-0dc9482e368bbe9a0 subnet-public
VPC: vpc-001c2f84899f4c3ea Owner: 577638386543 Availability Zone: ca-central-1a
Zone type: Availability Zone IP addresses available: 251 CIDR: 10.0.0.0/24)

Create new subnet

Auto-assign public IP Info

Disable

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

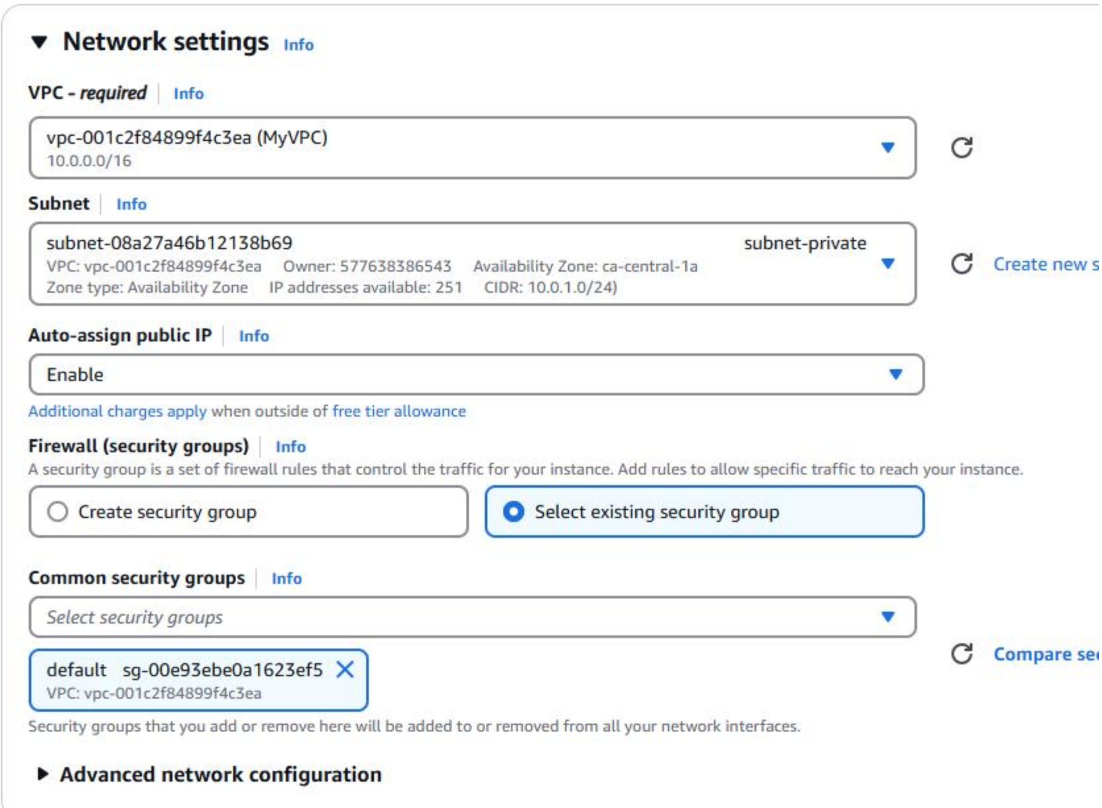
default sg-00e93ebe0a1623ef5
VPC: vpc-001c2f84899f4c3ea

Compare security group rules

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

▶ Advanced network configuration

Next create a Private-VM
Use the following settings
Auto-assign public IP is ‘Enable’



▼ **Network settings** Info

VPC - *required* Info

vpc-001c2f84899f4c3ea (MyVPC)
10.0.0.0/16

Subnet Info

subnet-08a27a46b12138b69 — subnet-private
VPC: vpc-001c2f84899f4c3ea Owner: 577638386543 Availability Zone: ca-central-1a
Zone type: Availability Zone IP addresses available: 251 CIDR: 10.0.1.0/24)

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

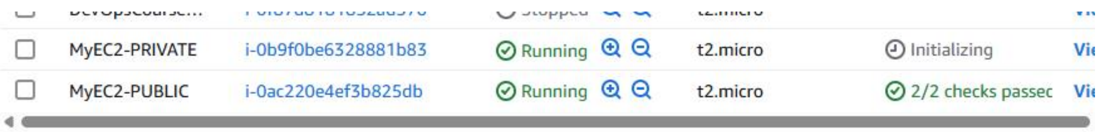
Common security groups Info

Select security groups

default sg-00e93ebe0a1623ef5
VPC: vpc-001c2f84899f4c3ea

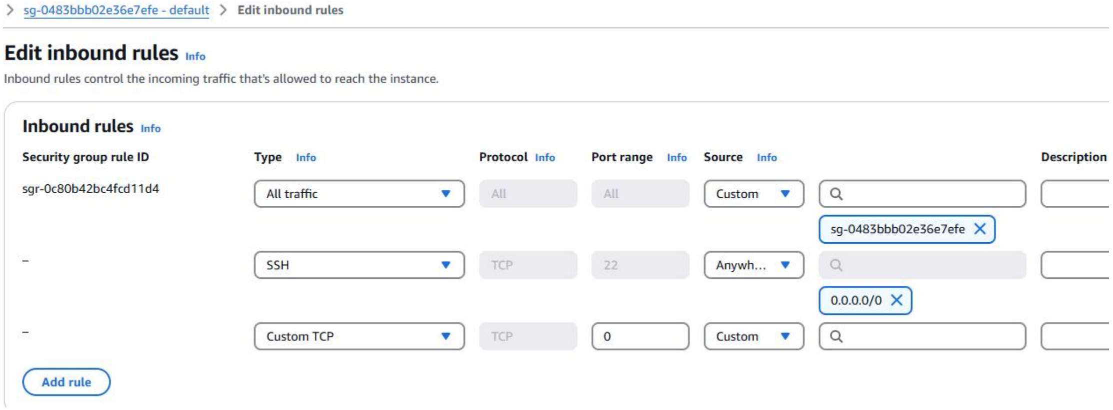
Security groups that you add or remove here will be added to or removed from all your network interfaces.

▶ **Advanced network configuration**



□	MyEC2-PRIVATE	i-0b9f0be6328881b83	Running	t2.micro	Initializing	Vi
□	MyEC2-PUBLIC	i-0ac220e4ef3b825db	Running	t2.micro	2/2 checks passec	Vi

Edit Security Group --> Inbound rules of VPC default SG



> sg-0483bbb02e36e7efe - default > Edit inbound rules

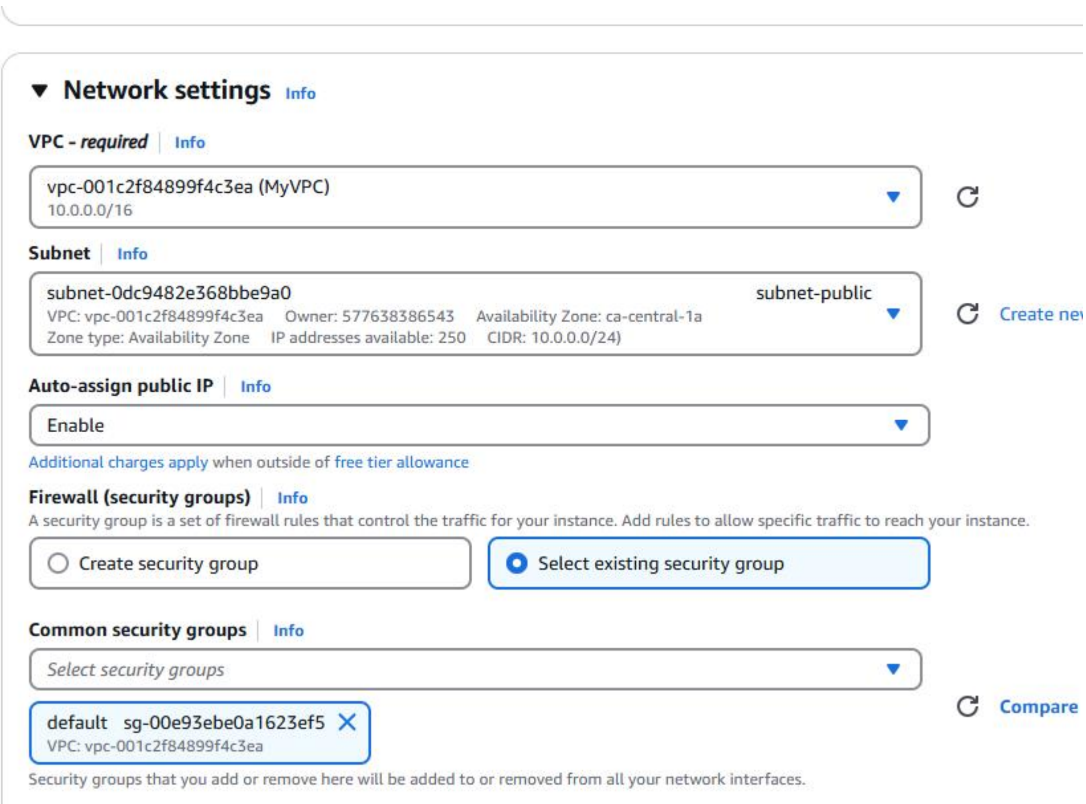
Edit inbound rules Info

Inbound rules control the incoming traffic that's allowed to reach the instance.

Inbound rules Info

Security group rule ID	Type	Protocol	Port range	Source		Description
sgr-0c80b42bc4fcd11d4	All traffic	All	All	Custom	sg-0483bbb02e36e7efe	
–	SSH	TCP	22	Anywh...	0.0.0.0/0	
–	Custom TCP	TCP	0	Custom		

For public EC2 also make sure Auto-assign public IP is enabled



▼ **Network settings** Info

VPC - *required* Info

vpc-001c2f84899f4c3ea (MyVPC)
10.0.0.0/16

Subnet Info

subnet-0dc9482e368bbe9a0 subnet-public
VPC: vpc-001c2f84899f4c3ea Owner: 577638386543 Availability Zone: ca-central-1a
Zone type: Availability Zone IP addresses available: 250 CIDR: 10.0.0.0/24)

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

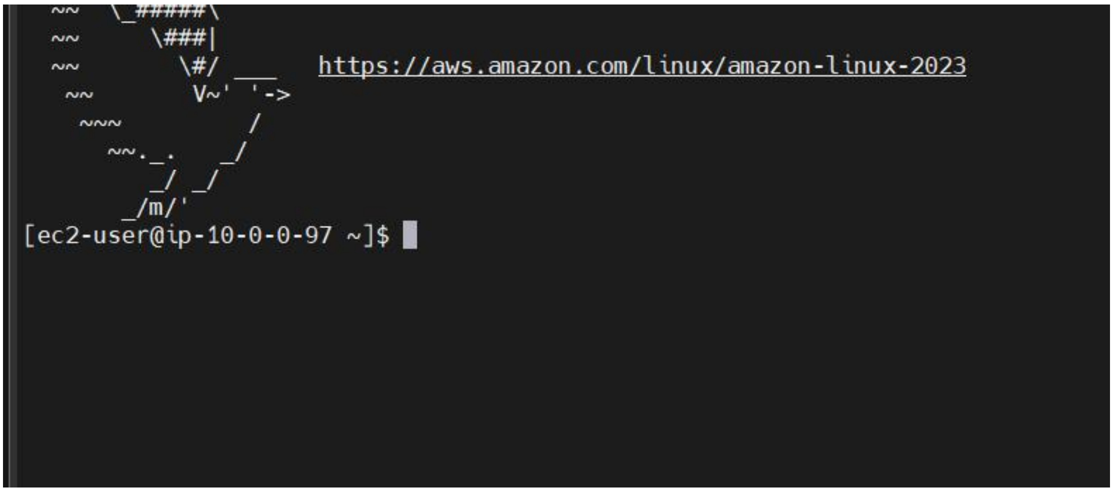
Select security groups

default sg-00e93ebe0a1623ef5 ✕
VPC: vpc-001c2f84899f4c3ea

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

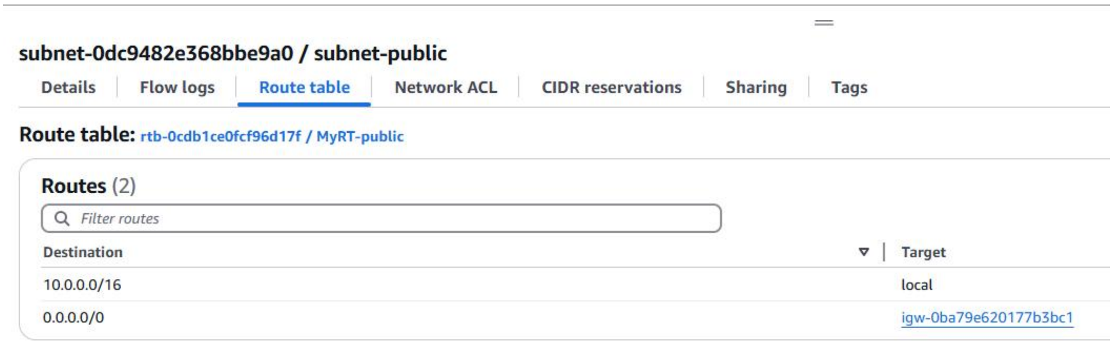
Copy Public IP then connect using MobaXTerm

It should connect



```
  ~~   \_#####\
  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
[ec2-user@ip-10-0-0-97 ~]$ 
```

Keep in mind for subnet-public, we have attached igw



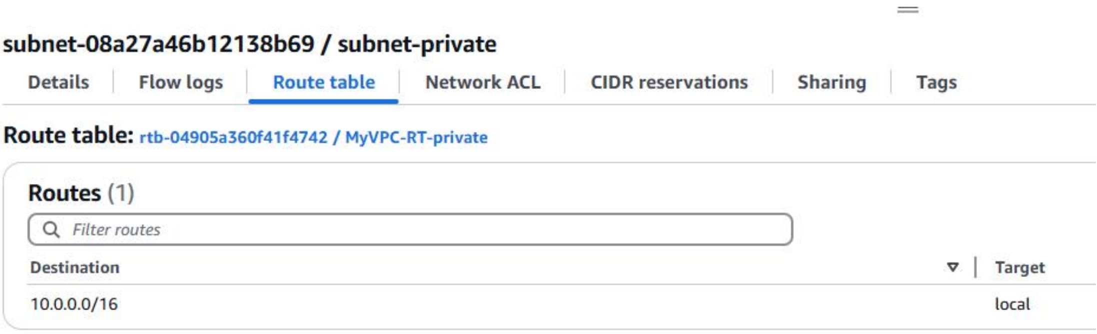
subnet-0dc9482e368bbe9a0 / subnet-public

Route table: rtb-0cdb1ce0fcf96d17f / MyRT-public

Routes (2)

Destination	Target
10.0.0.0/16	local
0.0.0.0/0	igw-0ba79e620177b3bc1

For subnet-private, we have local



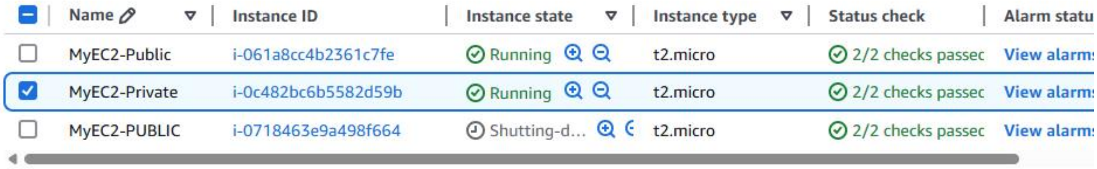
subnet-08a27a46b12138b69 / subnet-private

Route table: rtb-04905a360f41f4742 / MyVPC-RT-private

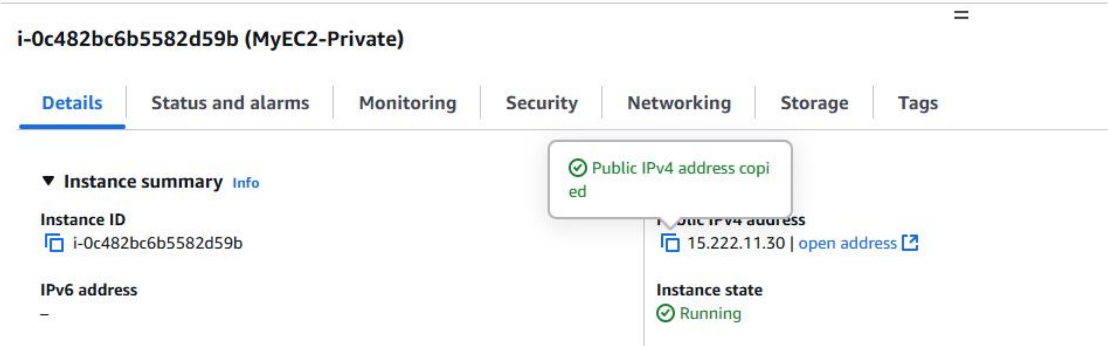
Routes (1)

Destination	Target
10.0.0.0/16	local

Go back to EC2, connect to private-EC2



☐	Name	Instance ID	Instance state	Instance type	Status check	Alarm statu
☐	MyEC2-Public	i-061a8cc4b2361c7fe	Running	t2.micro	2/2 checks passec	View alarm:
☑	MyEC2-Private	i-0c482bc6b5582d59b	Running	t2.micro	2/2 checks passec	View alarm:
☐	MyEC2-PUBLIC	i-0718463e9a498f664	Shutting-d...	t2.micro	2/2 checks passec	View alarm:



i-0c482bc6b5582d59b (MyEC2-Private)

▼ Instance summary Info

Public IPv4 address copied

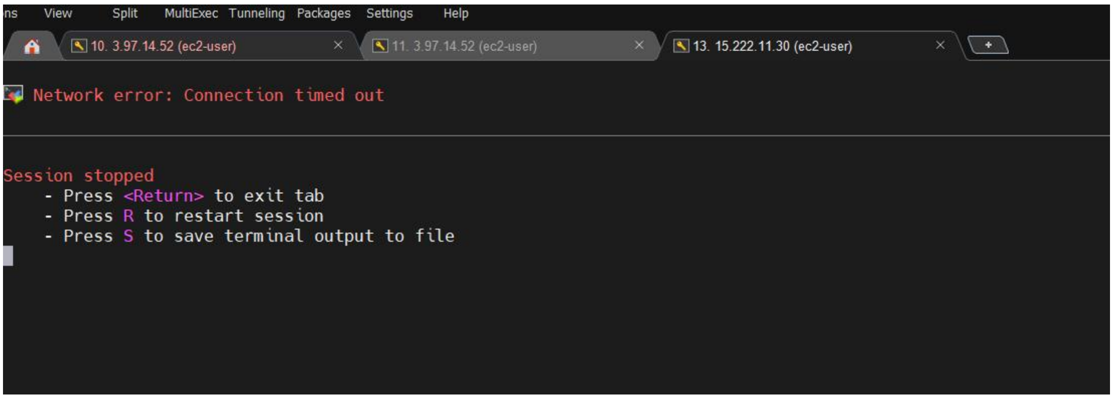
Instance ID
i-0c482bc6b5582d59b

Public IPv4 address
15.222.11.30 | open address

IPv6 address
–

Instance state
Running

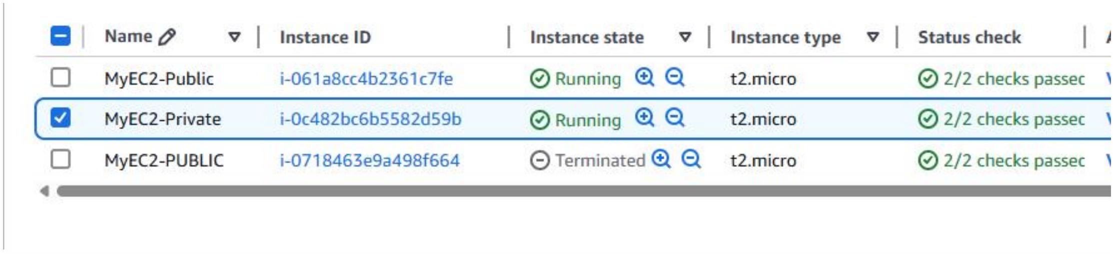
MyEC2-private
We are not able to establish the connection for private VM


ns View Split MultiExec Tunneling Packages Settings Help
10. 3.97.14.52 (ec2-user)
11. 3.97.14.52 (ec2-user)
13. 15.222.11.30 (ec2-user)

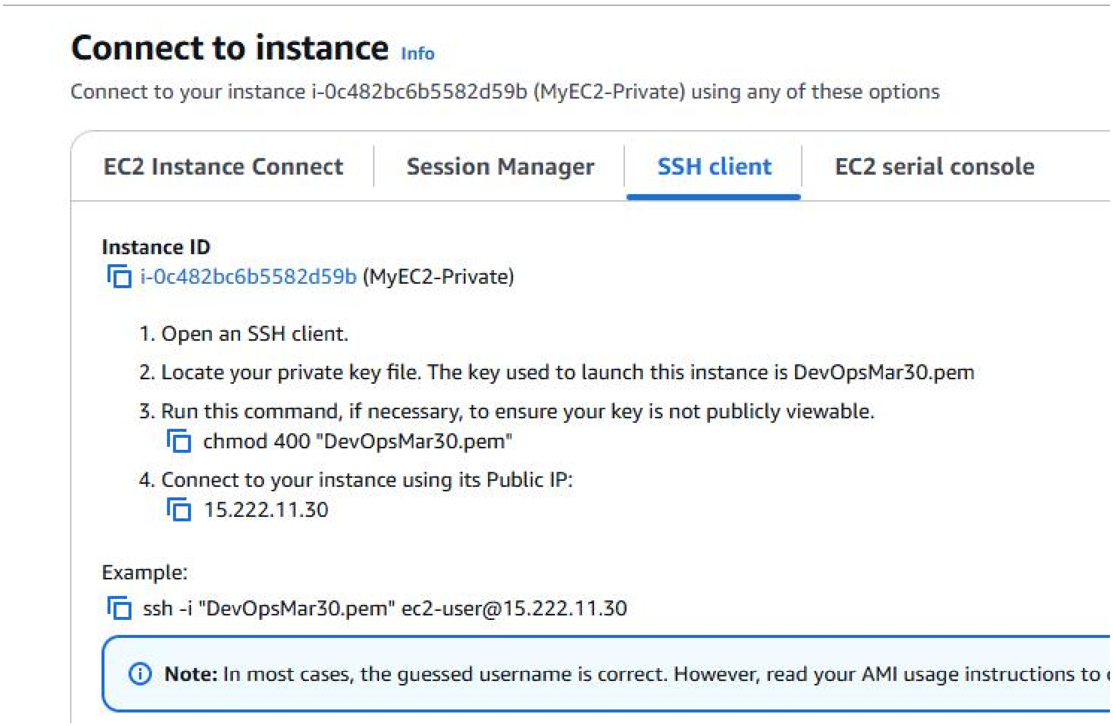
```
Network error: Connection timed out

Session stopped
    - Press <Return> to exit tab
    - Press R to restart session
    - Press S to save terminal output to file
```

Connect with Private EC2 of Private Subnet from Public EC2 of Public Subnet using SSH connection:

Select Private EC2, click Connect



☐	Name	Instance ID	Instance state	Instance type	Status check
☐	MyEC2-Public	i-061a8cc4b2361c7fe	Running	t2.micro	2/2 checks passec
☑	MyEC2-Private	i-0c482bc6b5582d59b	Running	t2.micro	2/2 checks passec
☐	MyEC2-PUBLIC	i-0718463e9a498f664	Terminated	t2.micro	2/2 checks passec



Connect to instance Info

Connect to your instance i-0c482bc6b5582d59b (MyEC2-Private) using any of these options

Instance ID

i-0c482bc6b5582d59b (MyEC2-Private)

1. Open an SSH client.
2. Locate your private key file. The key used to launch this instance is DevOpsMar30.pem
3. Run this command, if necessary, to ensure your key is not publicly viewable.
   chmod 400 "DevOpsMar30.pem"
4. Connect to your instance using its Public IP:
   15.222.11.30

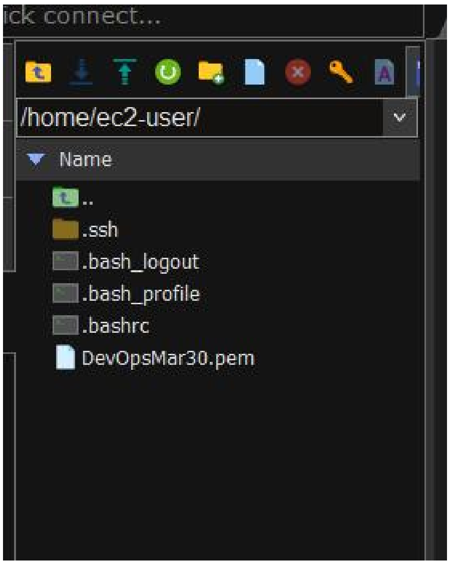
Example:

ssh -i "DevOpsMar30.pem" ec2-user@15.222.11.30

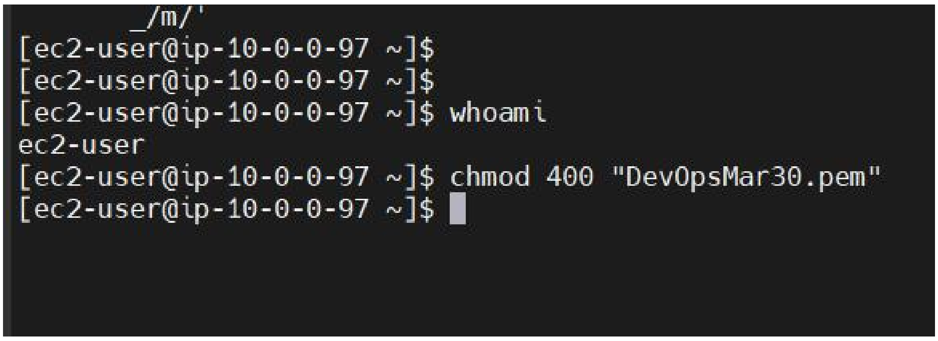
Note: In most cases, the guessed username is correct. However, read your AMI usage instructions to

Copy this and paste in EC2
chmod 400 "DevOpsMar30.pem"

Upload key first

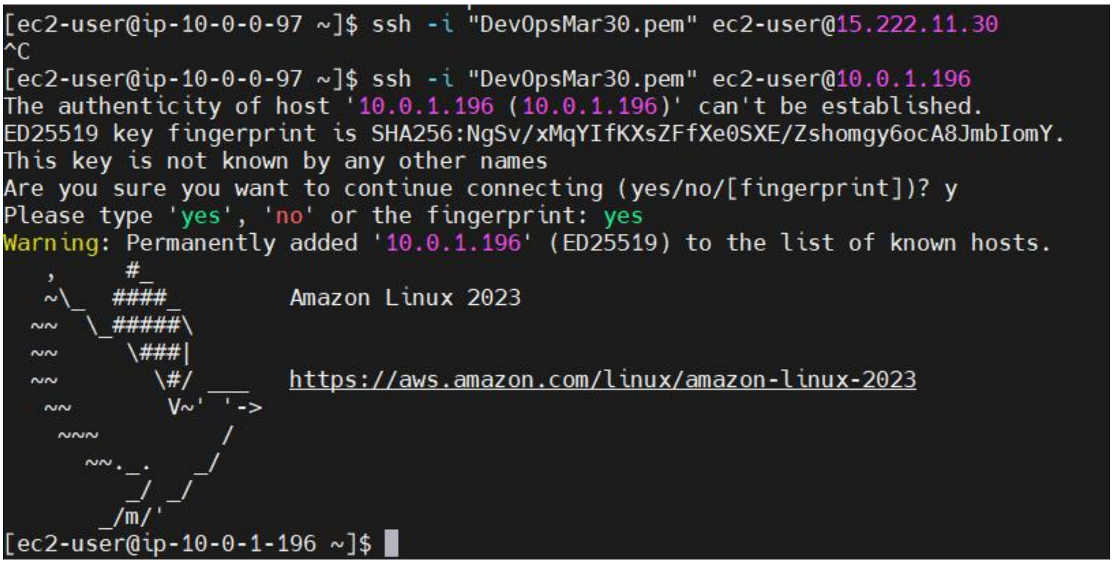


```
ick connect...
/home/ec2-user/
Name
..
.ssh
.bash_logout
.bash_profile
.bashrc
DevOpsMar30.pem
```



```
       _/m/'
[ec2-user@ip-10-0-0-97 ~]$
[ec2-user@ip-10-0-0-97 ~]$
[ec2-user@ip-10-0-0-97 ~]$ whoami
ec2-user
[ec2-user@ip-10-0-0-97 ~]$ chmod 400 "DevOpsMar30.pem"
[ec2-user@ip-10-0-0-97 ~]$ 
```

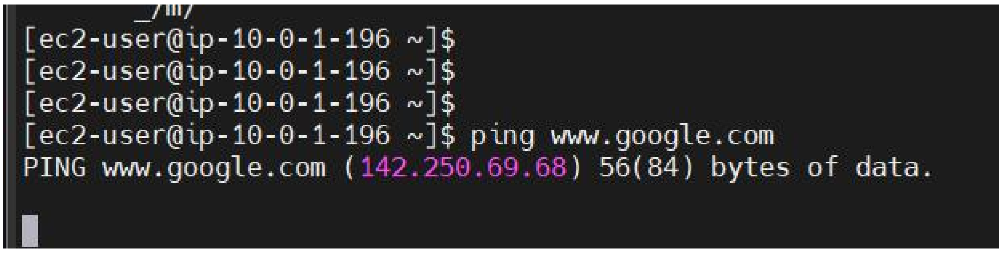
To Connect to Private EC2 from Public EC2
Use PRIVATE IP of Private EC2 Not Public IP then connect



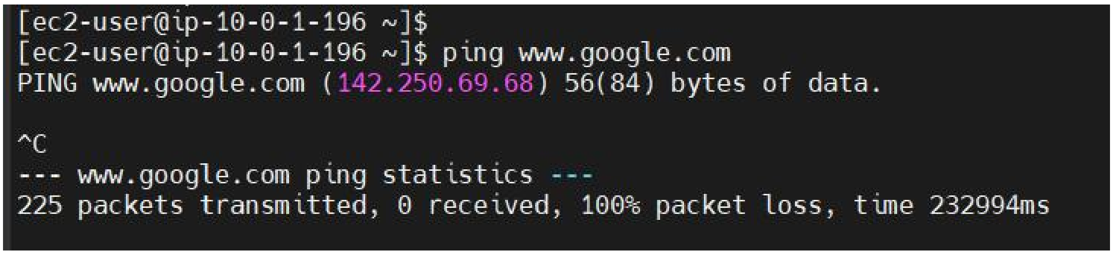
```
[ec2-user@ip-10-0-0-97 ~]$ ssh -i "DevOpsMar30.pem" ec2-user@15.222.11.30
^C
[ec2-user@ip-10-0-0-97 ~]$ ssh -i "DevOpsMar30.pem" ec2-user@10.0.1.196
The authenticity of host '10.0.1.196 (10.0.1.196)' can't be established.
ED25519 key fingerprint is SHA256:NgSv/xMqYIfKXsZFfXe0SXE/Zshomgy6ocA8JmbIomY.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? y
Please type 'yes', 'no' or the fingerprint: yes
Warning: Permanently added '10.0.1.196' (ED25519) to the list of known hosts.
   ,     #_
   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\
  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
[ec2-user@ip-10-0-1-196 ~]$ 
```

When it comes to public EC2, there is no issues with both Incoming and Outgoing. With Private EC2, for Outgoing service to be enabled, we need NatGateway is a must. Without NatGateway it is not possible. It is paid

Now I am on Private EC2 on Private Subnet and outgoing services wont work without NatGateway



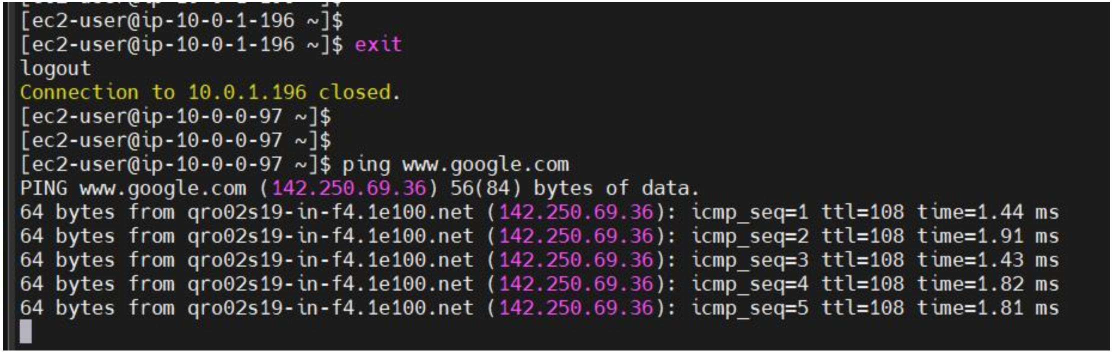
```
    _/m/
[ec2-user@ip-10-0-1-196 ~]$
[ec2-user@ip-10-0-1-196 ~]$
[ec2-user@ip-10-0-1-196 ~]$
[ec2-user@ip-10-0-1-196 ~]$ ping www.google.com
PING www.google.com (142.250.69.68) 56(84) bytes of data.

```



```
[ec2-user@ip-10-0-1-196 ~]$
[ec2-user@ip-10-0-1-196 ~]$ ping www.google.com
PING www.google.com (142.250.69.68) 56(84) bytes of data.

^C
--- www.google.com ping statistics ---
225 packets transmitted, 0 received, 100% packet loss, time 232994ms
```

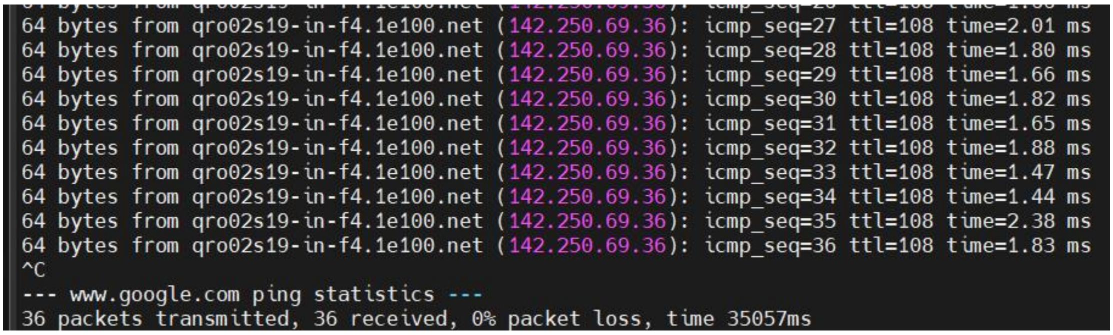
Now go back to Public EC2

I exit out of Private EC2 then do the same thing



```
[ec2-user@ip-10-0-1-196 ~]$
[ec2-user@ip-10-0-1-196 ~]$ exit
logout
Connection to 10.0.1.196 closed.
[ec2-user@ip-10-0-0-97 ~]$
[ec2-user@ip-10-0-0-97 ~]$
[ec2-user@ip-10-0-0-97 ~]$ ping www.google.com
PING www.google.com (142.250.69.36) 56(84) bytes of data.
64 bytes from qro02s19-in-f4.1e100.net (142.250.69.36): icmp_seq=1 ttl=108 time=1.44 ms
64 bytes from qro02s19-in-f4.1e100.net (142.250.69.36): icmp_seq=2 ttl=108 time=1.91 ms
64 bytes from qro02s19-in-f4.1e100.net (142.250.69.36): icmp_seq=3 ttl=108 time=1.43 ms
64 bytes from qro02s19-in-f4.1e100.net (142.250.69.36): icmp_seq=4 ttl=108 time=1.82 ms
64 bytes from qro02s19-in-f4.1e100.net (142.250.69.36): icmp_seq=5 ttl=108 time=1.81 ms
```

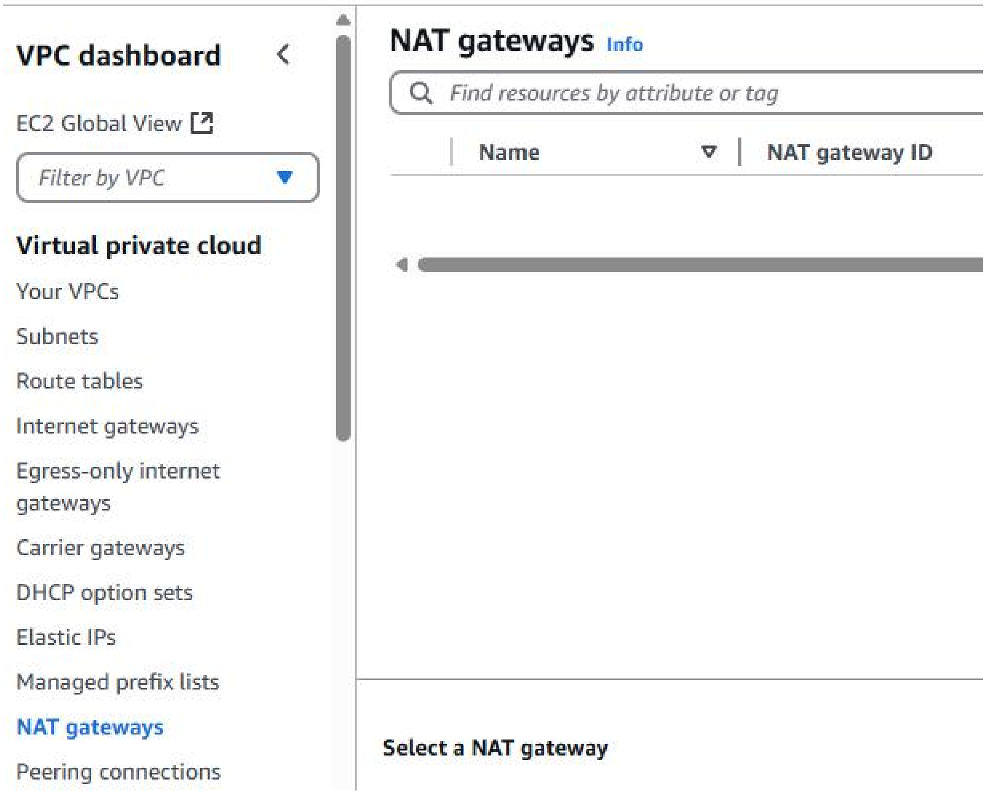
It works! From Public EC2 on Public Subnet



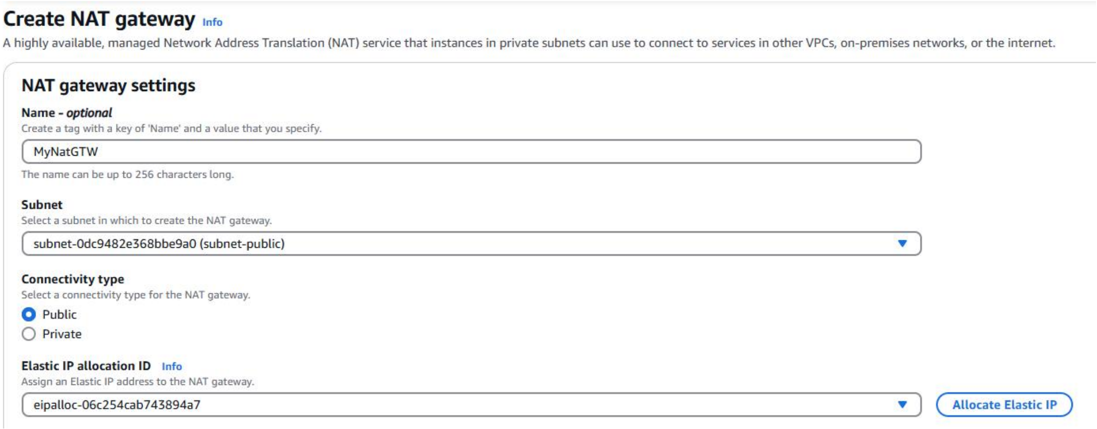
```
64 bytes from qro02s19-in-f4.1e100.net (142.250.69.36): icmp_seq=27 ttl=108 time=2.01 ms
64 bytes from qro02s19-in-f4.1e100.net (142.250.69.36): icmp_seq=28 ttl=108 time=1.80 ms
64 bytes from qro02s19-in-f4.1e100.net (142.250.69.36): icmp_seq=29 ttl=108 time=1.66 ms
64 bytes from qro02s19-in-f4.1e100.net (142.250.69.36): icmp_seq=30 ttl=108 time=1.82 ms
64 bytes from qro02s19-in-f4.1e100.net (142.250.69.36): icmp_seq=31 ttl=108 time=1.65 ms
64 bytes from qro02s19-in-f4.1e100.net (142.250.69.36): icmp_seq=32 ttl=108 time=1.88 ms
64 bytes from qro02s19-in-f4.1e100.net (142.250.69.36): icmp_seq=33 ttl=108 time=1.47 ms
64 bytes from qro02s19-in-f4.1e100.net (142.250.69.36): icmp_seq=34 ttl=108 time=1.44 ms
64 bytes from qro02s19-in-f4.1e100.net (142.250.69.36): icmp_seq=35 ttl=108 time=2.38 ms
64 bytes from qro02s19-in-f4.1e100.net (142.250.69.36): icmp_seq=36 ttl=108 time=1.83 ms
^C
--- www.google.com ping statistics ---
36 packets transmitted, 36 received, 0% packet loss, time 35057ms
```

See Packet loss is 0% from Public Subnet

If you go back to VPC, you will see NAT gateways



To use Nat Gateway in Private Subnet you have to create in a PUBLIC Subnet, internet should be accessible. Select subnet-public
Click Allocate Elastic IP
Elastic IPs and Nat Gateway are paid services. Please delete after practicing



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.

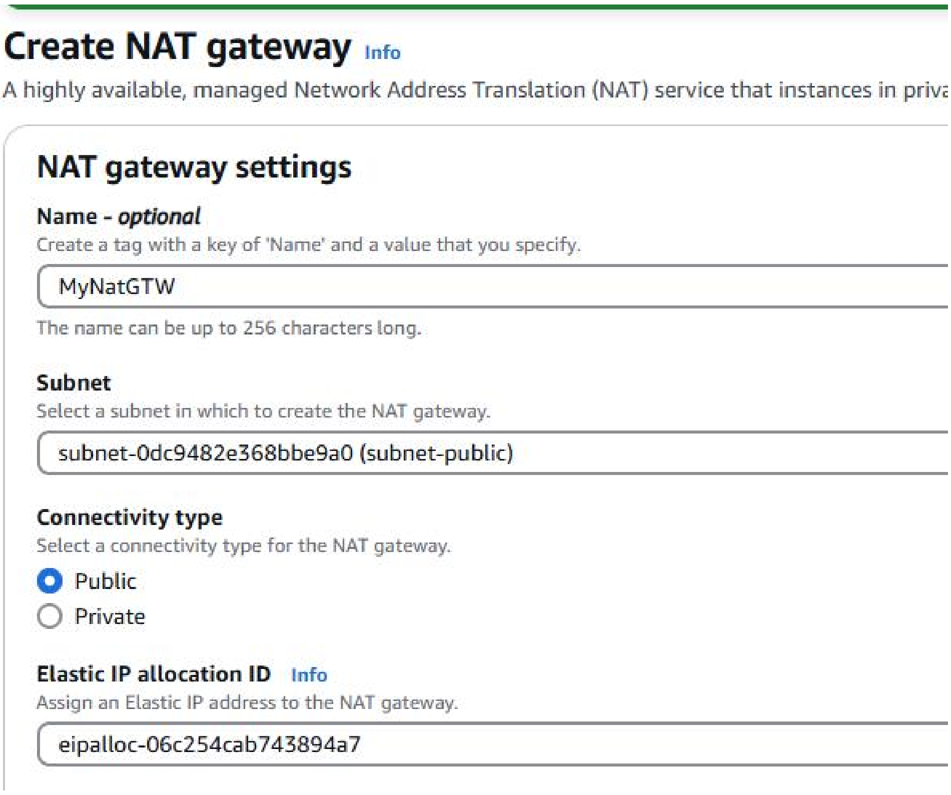
NAT gateway settings

Name - *optional*
Create a tag with a key of 'Name' and a value that you specify.
MyNatGTW
The name can be up to 256 characters long.

Subnet
Select a subnet in which to create the NAT gateway.
subnet-0dc9482e368bbe9a0 (subnet-public)

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-06c254cab743894a7



Create NAT gateway Info

A highly available, managed Network Address Translation (NAT) service that instances in priva

NAT gateway settings

Name - *optional*

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

MyNatGTW

The name can be up to 256 characters long.

Subnet

Select a subnet in which to create the NAT gateway.

subnet-0dc9482e368bbe9a0 (subnet-public)

Connectivity type

Select a connectivity type for the NAT gateway.

● Public

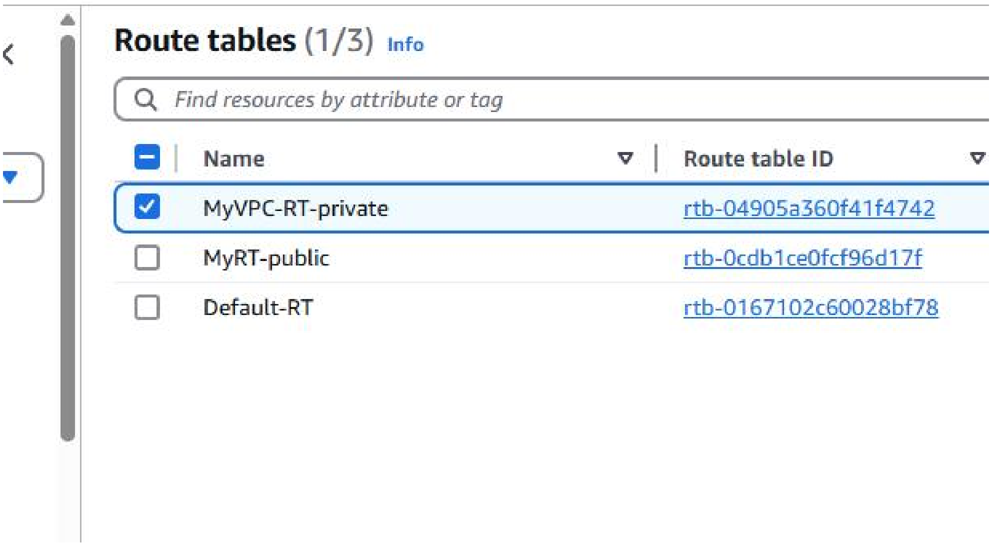
○ Private

Elastic IP allocation ID Info

Assign an Elastic IP address to the NAT gateway.

eipalloc-06c254cab743894a7

Click Create Nat Gateway
Go back to Route tables

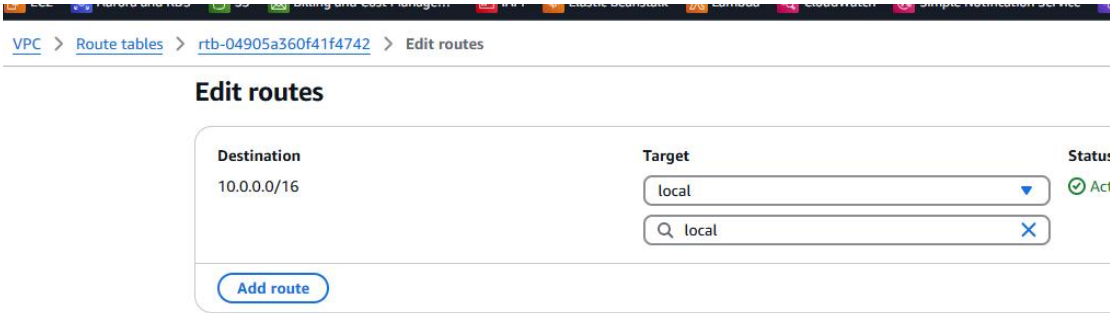


Route tables (1/3) Info

Find resources by attribute or tag

☐	Name	Route table ID
☑	MyVPC-RT-private	rtb-04905a360f41f4742
☐	MyRT-public	rtb-0cdb1ce0fcf96d17f
☐	Default-RT	rtb-0167102c60028bf78

For Private RT, click Edit Routes

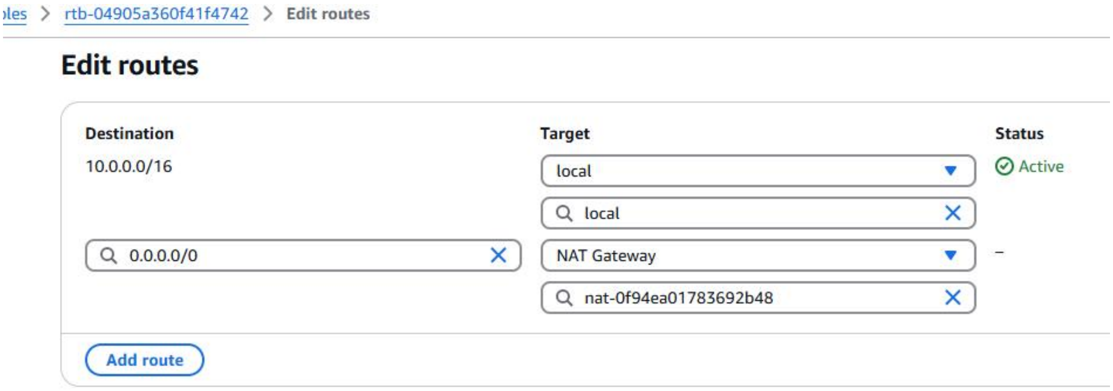


VPC > Route tables > rtb-04905a360f41f4742 > Edit routes

Edit routes

Destination	Target	Statu
10.0.0.0/16	local ▼ local ✕	✓ Act

Add route



les > rtb-04905a360f41f4742 > Edit routes

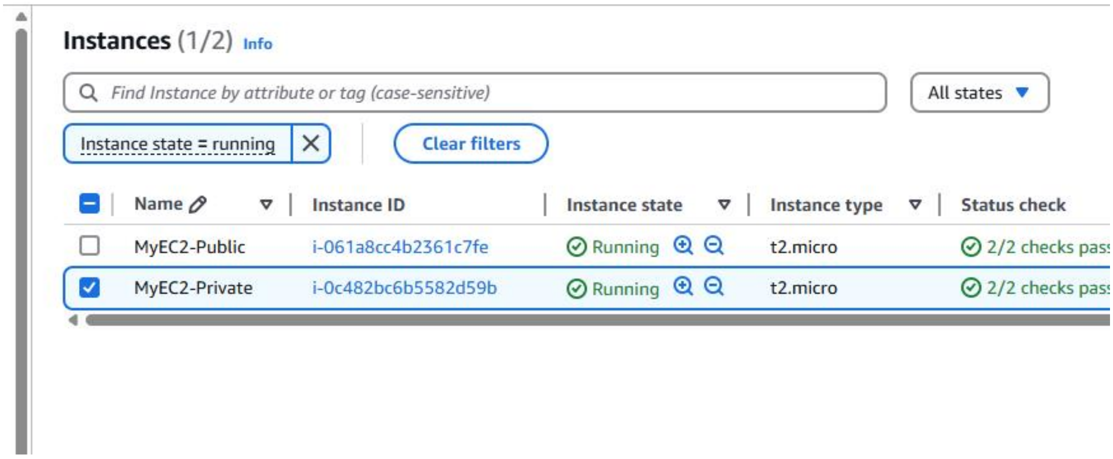
Edit routes

Destination	Target	Status
10.0.0.0/16	local ▼ local ✕	✓ Active
0.0.0.0/0 ✕	NAT Gateway ▼ nat-0f94ea01783692b48 ✕	–

Select 0.0.0.0/0 then NAT Gateway ---> Select the one we have created

Click Save changes

Go back to EC2



Instances (1/2) Info

Find Instance by attribute or tag (case-sensitive)

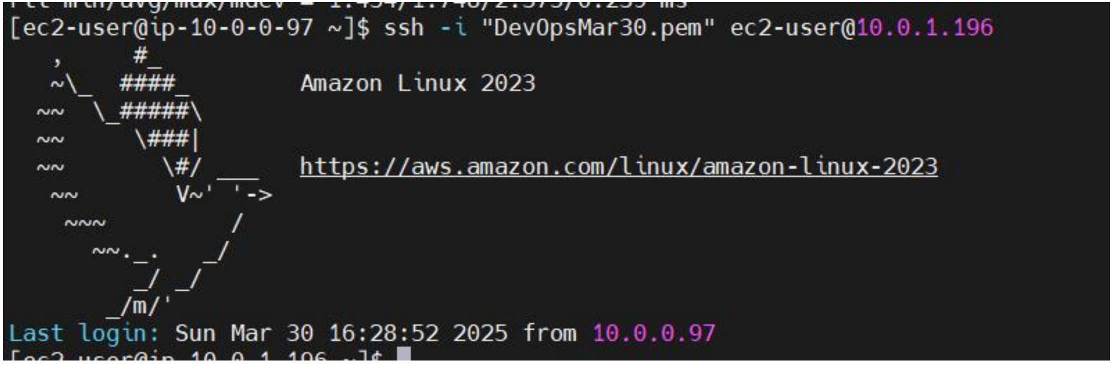
All states ▼

Instance state = running ✕

	Name	Instance ID	Instance state	Instance type	Status check
□	MyEC2-Public	i-061a8cc4b2361c7fe	✓ Running	t2.micro	✓ 2/2 checks pass
☑	MyEC2-Private	i-0c482bc6b5582d59b	✓ Running	t2.micro	✓ 2/2 checks pass

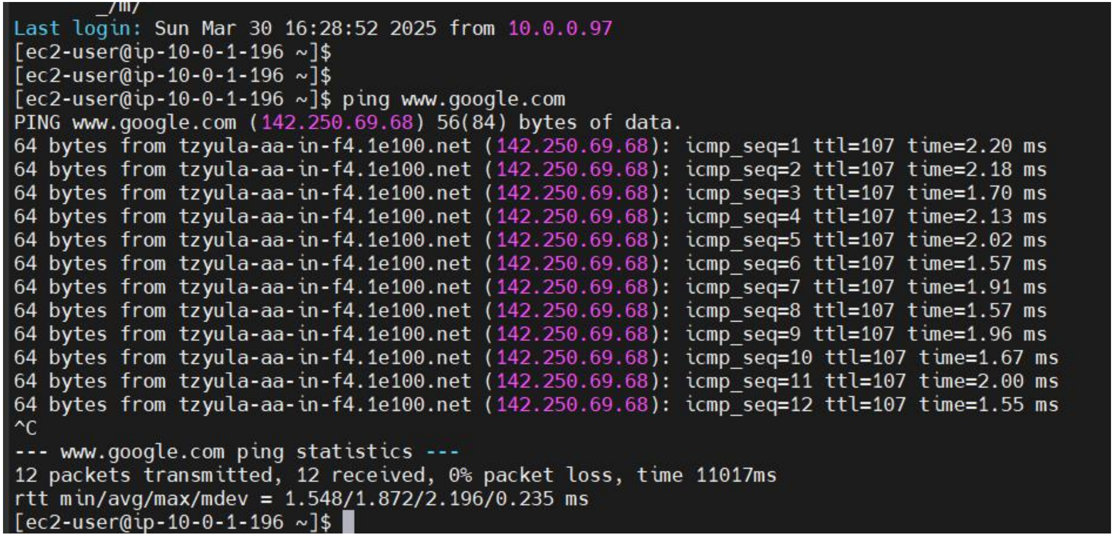
MyEC2-Private --> Connect

Connect to Private EC2 from Public EC2 again



```
[ec2-user@ip-10-0-0-97 ~]$ ssh -i "DevOpsMar30.pem" ec2-user@10.0.1.196
   ,     #_
   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\
  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
Last login: Sun Mar 30 16:28:52 2025 from 10.0.0.97
[ec2-user@ip-10-0-1-196 ~]$
```

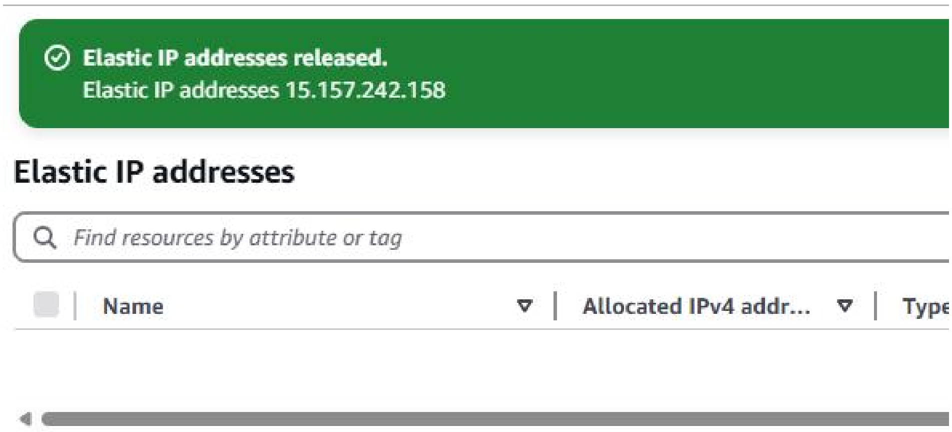
Now ping from Private EC2 and 0% packet loss



```
Last login: Sun Mar 30 16:28:52 2025 from 10.0.0.97
[ec2-user@ip-10-0-1-196 ~]$
[ec2-user@ip-10-0-1-196 ~]$
[ec2-user@ip-10-0-1-196 ~]$ ping www.google.com
PING www.google.com (142.250.69.68) 56(84) bytes of data.
64 bytes from tzyula-aa-in-f4.1e100.net (142.250.69.68): icmp_seq=1 ttl=107 time=2.20 ms
64 bytes from tzyula-aa-in-f4.1e100.net (142.250.69.68): icmp_seq=2 ttl=107 time=2.18 ms
64 bytes from tzyula-aa-in-f4.1e100.net (142.250.69.68): icmp_seq=3 ttl=107 time=1.70 ms
64 bytes from tzyula-aa-in-f4.1e100.net (142.250.69.68): icmp_seq=4 ttl=107 time=2.13 ms
64 bytes from tzyula-aa-in-f4.1e100.net (142.250.69.68): icmp_seq=5 ttl=107 time=2.02 ms
64 bytes from tzyula-aa-in-f4.1e100.net (142.250.69.68): icmp_seq=6 ttl=107 time=1.57 ms
64 bytes from tzyula-aa-in-f4.1e100.net (142.250.69.68): icmp_seq=7 ttl=107 time=1.91 ms
64 bytes from tzyula-aa-in-f4.1e100.net (142.250.69.68): icmp_seq=8 ttl=107 time=1.57 ms
64 bytes from tzyula-aa-in-f4.1e100.net (142.250.69.68): icmp_seq=9 ttl=107 time=1.96 ms
64 bytes from tzyula-aa-in-f4.1e100.net (142.250.69.68): icmp_seq=10 ttl=107 time=1.67 ms
64 bytes from tzyula-aa-in-f4.1e100.net (142.250.69.68): icmp_seq=11 ttl=107 time=2.00 ms
64 bytes from tzyula-aa-in-f4.1e100.net (142.250.69.68): icmp_seq=12 ttl=107 time=1.55 ms
^C
--- www.google.com ping statistics ---
12 packets transmitted, 12 received, 0% packet loss, time 11017ms
rtt min/avg/max/mdev = 1.548/1.872/2.196/0.235 ms
[ec2-user@ip-10-0-1-196 ~]$
```

Please delete NAT Gateways immediately

Delete Elastic IP also --> Release Elastic IP

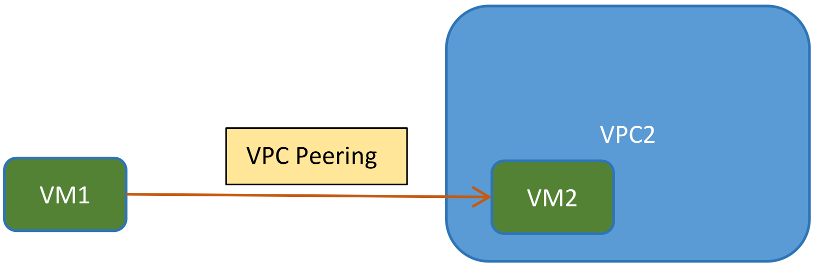


Elastic IP addresses released.
Elastic IP addresses 15.157.242.158

Elastic IP addresses

Right now we established connection between different Subnets within the same VPC

Say you have two different VPCs, owned by two different Users
Can we connect EC2s between them and perform operations?



That’s where VPC peering comes into Picture