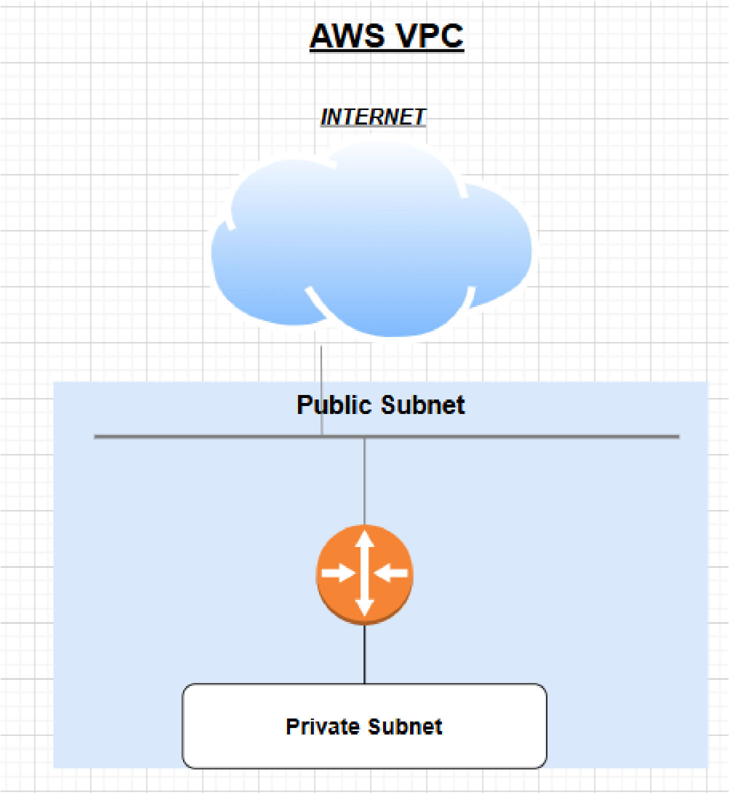
AWS Task-2

Task Description:

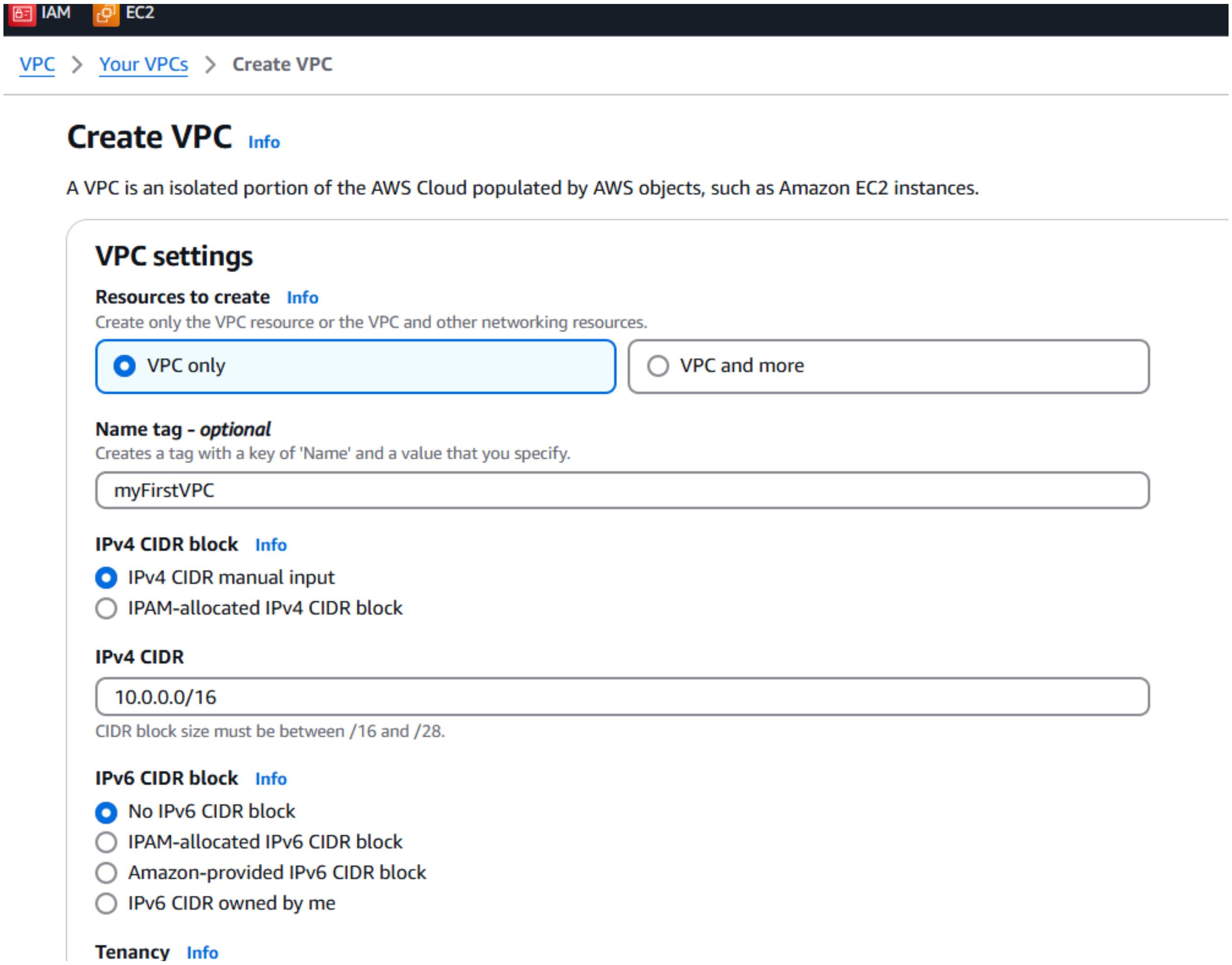
Set up a VPC with an Internet gateway, create a public subnet with 256 IP addresses, a private subnet with 256 IP addresses, make a route table connecting the Internet gateway and the subnets, and launch a Linux EC2 instance by using the above VPC and public subnet.

Techstacks needs to be used :

- AWS VPC
- AWS EC2



Step1: 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.

myFirstVPC

IPv4 CIDR block Info

● IPv4 CIDR manual input
○ IPAM-allocated IPv4 CIDR block

IPv4 CIDR

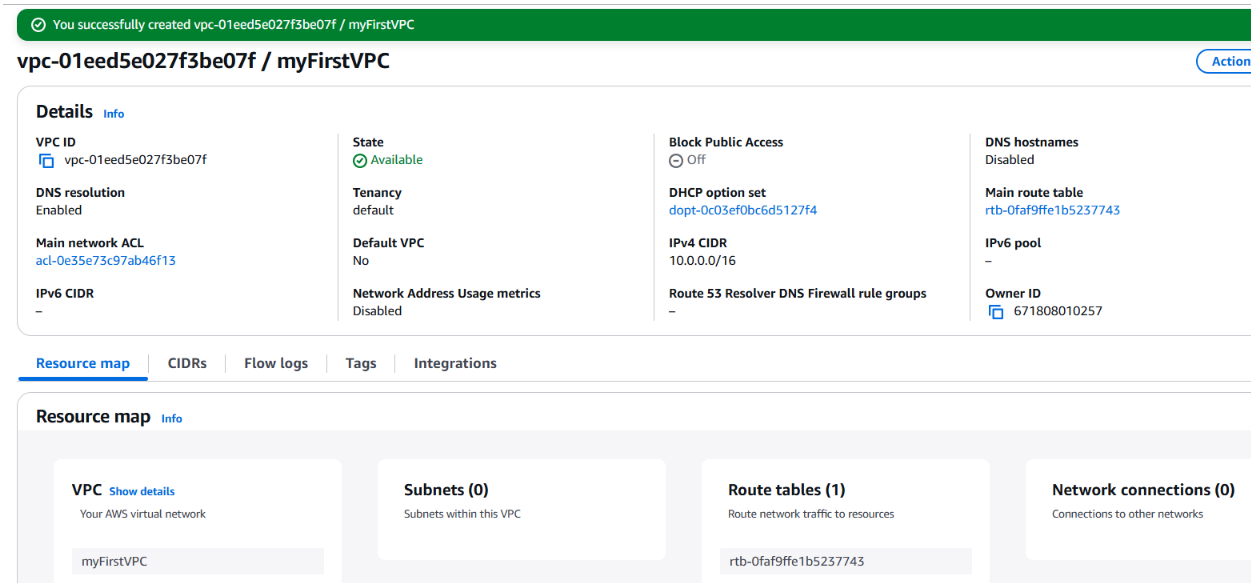
10.0.0.0/16

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


You successfully created vpc-01eed5e027f3be07f / myFirstVPC

vpc-01eed5e027f3be07f / myFirstVPC

Details Info

VPC ID vpc-01eed5e027f3be07f	**State** Available	**Block Public Access** Off	**DNS hostnames** Disabled
DNS resolution Enabled	**Tenancy** default	**DHCP option set** dopt-0c03ef0bc6d5127f4	**Main route table** rtb-0faf9ffe1b5237743
Main network ACL acl-0e35e73c97ab46f13	**Default VPC** No	**IPv4 CIDR** 10.0.0.0/16	**IPv6 pool** –
IPv6 CIDR –	**Network Address Usage metrics** Disabled	**Route 53 Resolver DNS Firewall rule groups** –	**Owner ID** 671808010257

Resource map Info

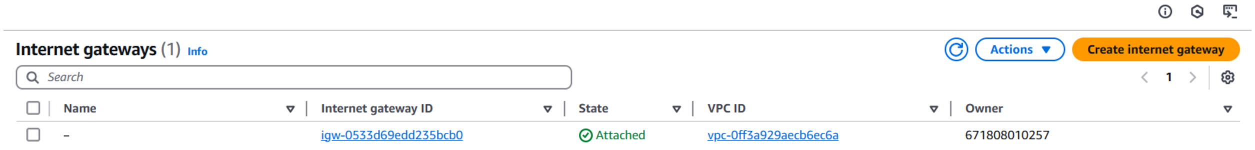
VPC Show details
Your AWS virtual network
myFirstVPC

Subnets (0)
Subnets within this VPC

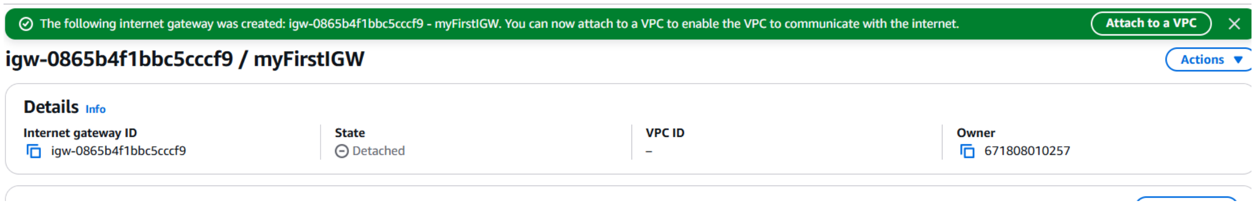
Route tables (1)
Route network traffic to resources
rtb-0faf9ffe1b5237743

Network connections (0)
Connections to other networks

Step2 now create Internet gateway.


Internet gateways (1) Info

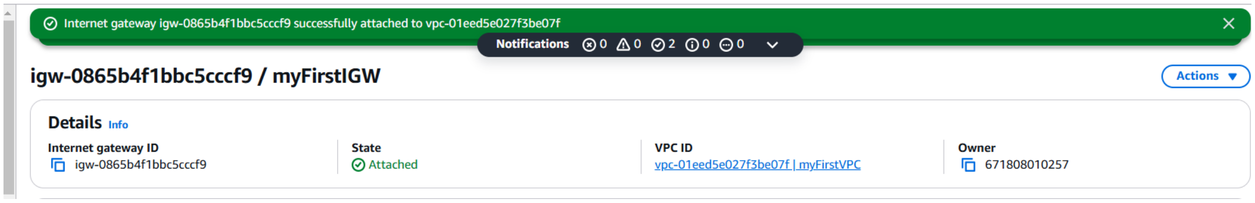
Name	Internet gateway ID	State	VPC ID	Owner
–	igw-0533d69edd235bcb0	Attached	vpc-0ff3a929aecb6ec6a	671808010257


The following internet gateway was created: igw-0865b4f1bbc5cccf9 - myFirstIGW. You can now attach to a VPC to enable the VPC to communicate with the internet.

igw-0865b4f1bbc5cccf9 / myFirstIGW

Details Info

Internet gateway ID igw-0865b4f1bbc5cccf9	**State** Detached	**VPC ID** –	**Owner** 671808010257


Internet gateway igw-0865b4f1bbc5cccf9 successfully attached to vpc-01eed5e027f3be07f

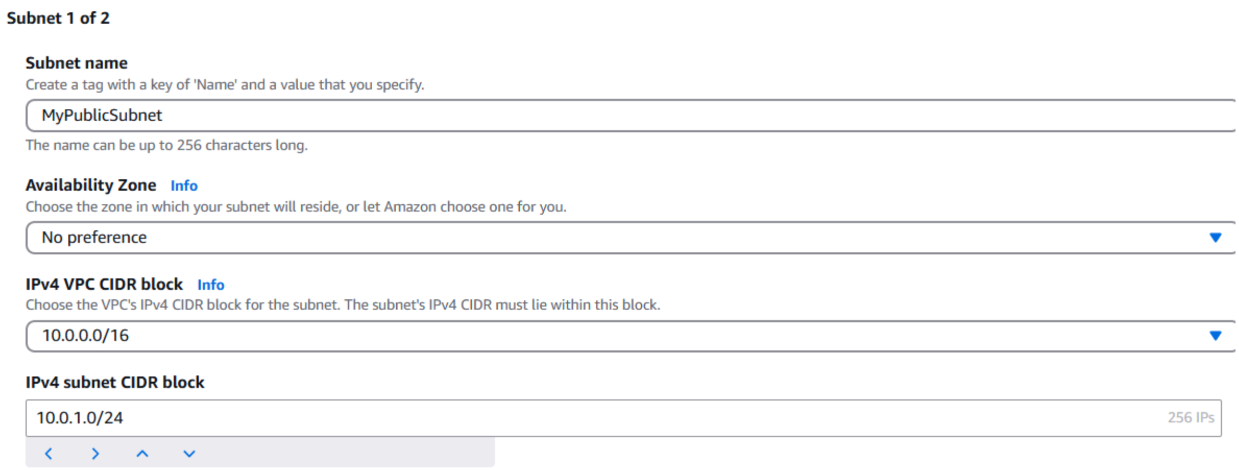
igw-0865b4f1bbc5cccf9 / myFirstIGW

Details Info

Internet gateway ID igw-0865b4f1bbc5cccf9	**State** Attached	**VPC ID** vpc-01eed5e027f3be07f | myFirstVPC	**Owner** 671808010257

Step3 Now create subnets

Public subnet



Subnet 1 of 2

Subnet name

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

MyPublicSubnet

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.

No preference ▼

IPv4 VPC CIDR block Info

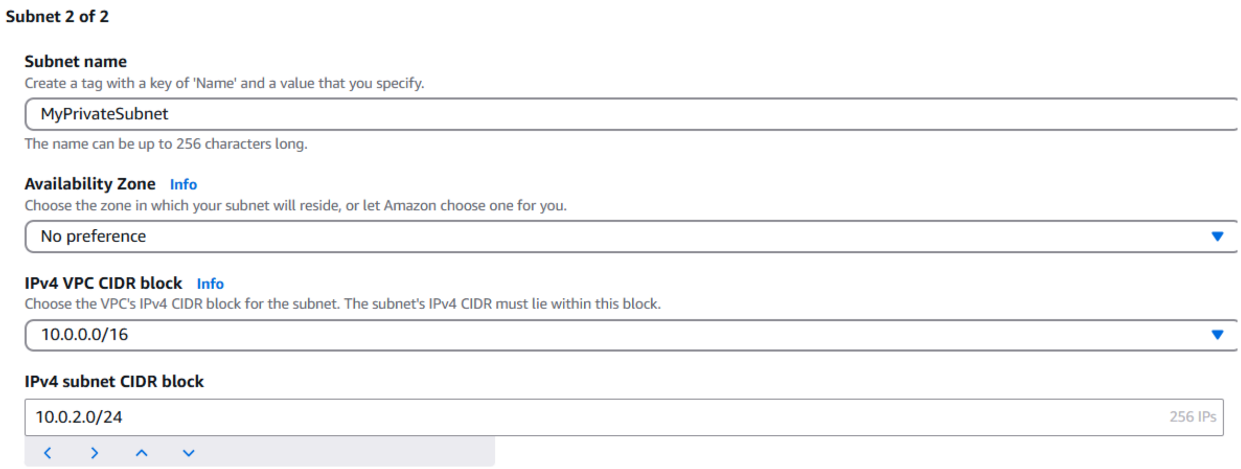
Choose the VPC's IPv4 CIDR block for the subnet. The subnet's IPv4 CIDR must lie within this block.

10.0.0.0/16 ▼

IPv4 subnet CIDR block

10.0.1.0/24 256 IPs

Private Subnet



Subnet 2 of 2

Subnet name

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

MyPrivateSubnet

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.

No preference ▼

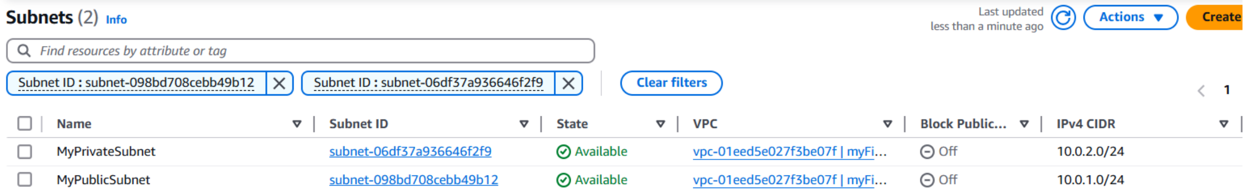
IPv4 VPC CIDR block Info

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

10.0.2.0/24 256 IPs

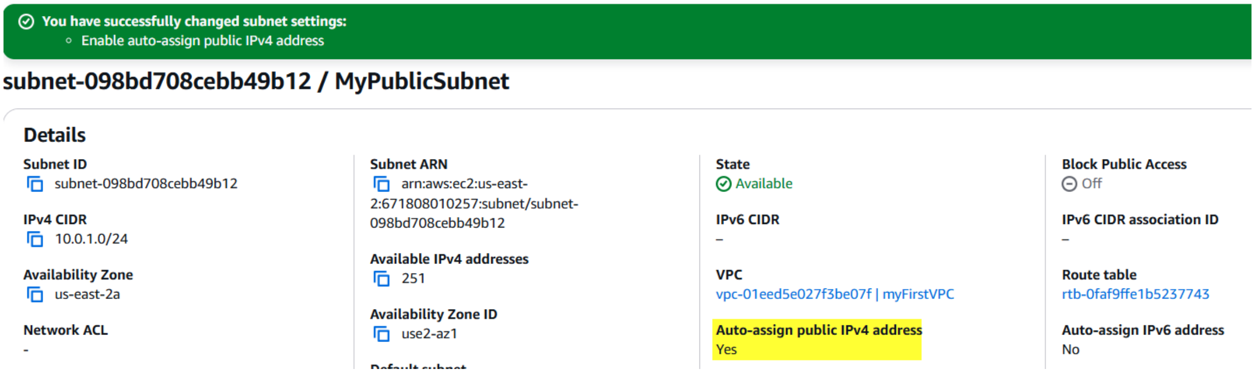


Subnets (2) Info

Subnet ID : subnet-098bd708cebb49b12 ✕ Subnet ID : subnet-06df37a936646f2f9 ✕

□	Name	Subnet ID	State	VPC	Block Public...	IPv4 CIDR
□	MyPrivateSubnet	subnet-06df37a936646f2f9	✓ Available	vpc-01eed5e027f3be07f | myFi...	⊝ Off	10.0.2.0/24
□	MyPublicSubnet	subnet-098bd708cebb49b12	✓ Available	vpc-01eed5e027f3be07f | myFi...	⊝ Off	10.0.1.0/24

Enable auto assign IPV4 subnet in public subnet



You have successfully changed subnet settings:
- Enable auto-assign public IPv4 address

subnet-098bd708cebb49b12 / MyPublicSubnet

Details

Subnet ID: subnet-098bd708cebb49b12

IPv4 CIDR: 10.0.1.0/24

Availability Zone: us-east-2a

Network ACL: -

Subnet ARN: arn:aws:ec2:us-east-2:671808010257:subnet/subnet-098bd708cebb49b12

Available IPv4 addresses: 251

Availability Zone ID: use2-az1

Default subnet

State: Available

IPv6 CIDR: –

VPC: vpc-01eed5e027f3be07f | myFirstVPC

Auto-assign public IPv4 address: Yes

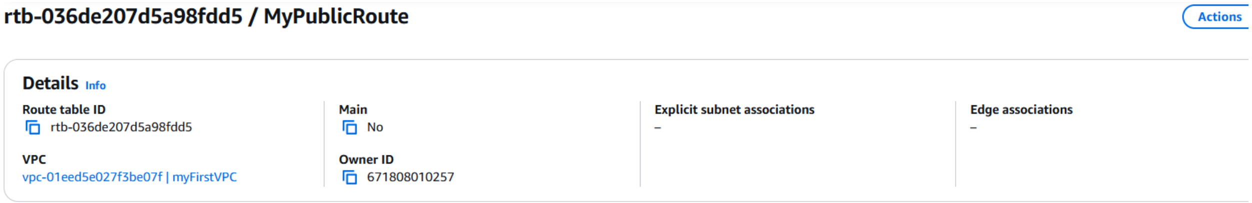
Block Public Access: Off

IPv6 CIDR association ID: –

Route table: rtb-0faf9ffe1b5237743

Auto-assign IPv6 address: No

Step 4: Create Route table for public subnet



rtb-036de207d5a98fdd5 / MyPublicRoute

Details Info

Route table ID: rtb-036de207d5a98fdd5

VPC: vpc-01eed5e027f3be07f | myFirstVPC

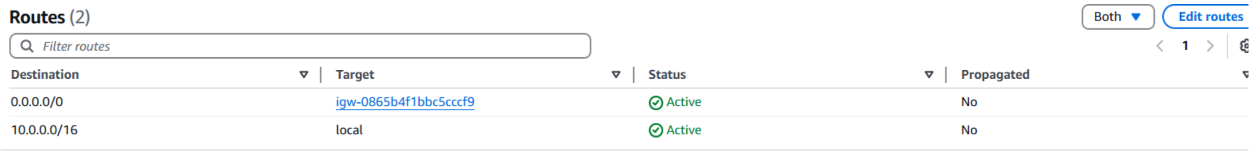
Main: No

Owner ID: 671808010257

Explicit subnet associations: –

Edge associations: –

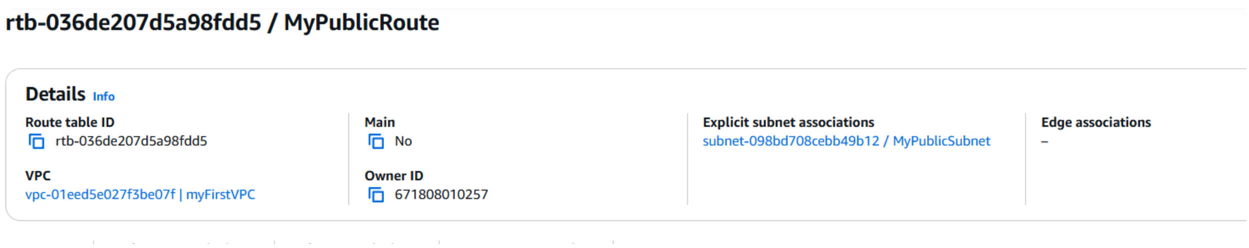
Now update routes and attach to Internet GW to route table



Routes (2)

Destination	Target	Status	Propagated
0.0.0.0/0	igw-0865b4f1bbc5cccf9	Active	No
10.0.0.0/16	local	Active	No

Now add subnet association in Routing table to PublicSubnet and save associations under Routing Table



rtb-036de207d5a98fdd5 / MyPublicRoute

Details Info

Route table ID: rtb-036de207d5a98fdd5

Main: No

Explicit subnet associations: subnet-098bd708cebb49b12 / MyPublicSubnet

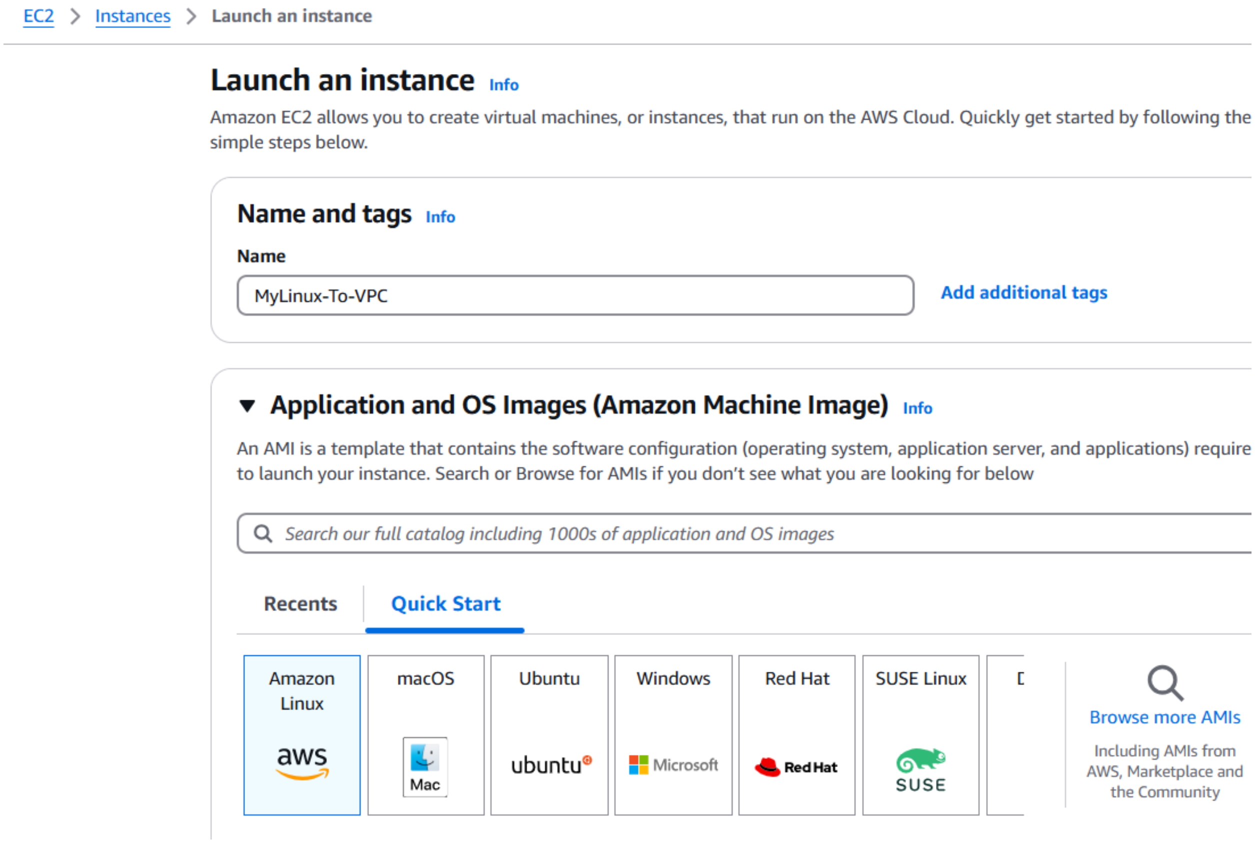
Edge associations: –

VPC: vpc-01eed5e027f3be07f | myFirstVPC

Owner ID: 671808010257

And for private subnet we will not use private subnet to internet therefore we will not associate IGW to private subnet and leave as default

Step 5: Now we launch Linux EC2 instance and select VPC inside in it.



EC2 > Instances > Launch an instance

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 the simple steps below.

Name and tags Info

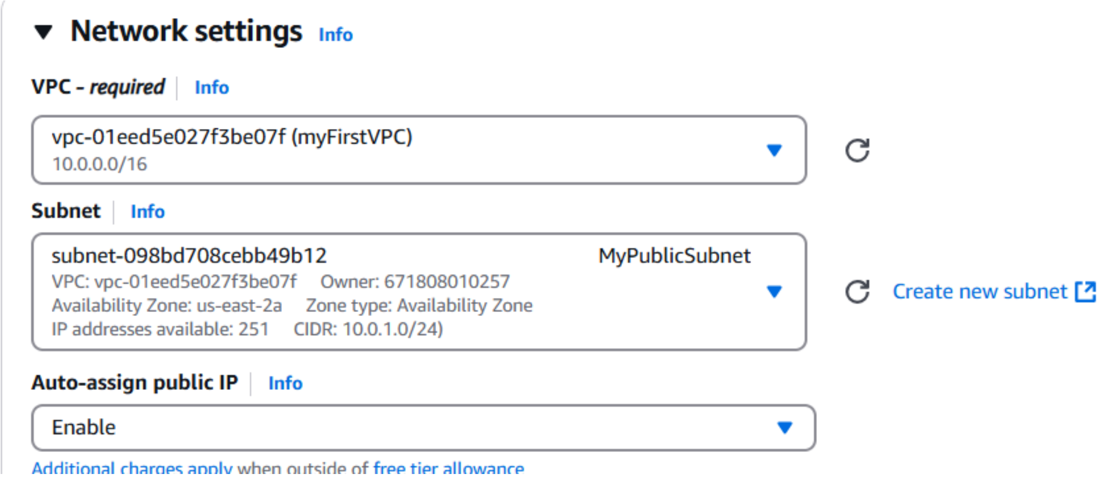
Name: MyLinux-To-VPC

Add additional tags

▼ Application and OS Images (Amazon Machine Image) Info

An AMI is a template that contains the software configuration (operating system, application server, and applications) require to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below

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



▼ **Network settings** Info

VPC - *required* | Info

vpc-01eed5e027f3be07f (myFirstVPC)
10.0.0.0/16

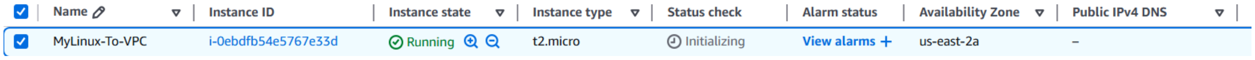
Subnet | Info

subnet-098bd708cebb49b12 MyPublicSubnet
VPC: vpc-01eed5e027f3be07f Owner: 671808010257
Availability Zone: us-east-2a Zone type: Availability Zone
IP addresses available: 251 CIDR: 10.0.1.0/24)

Auto-assign public IP | Info

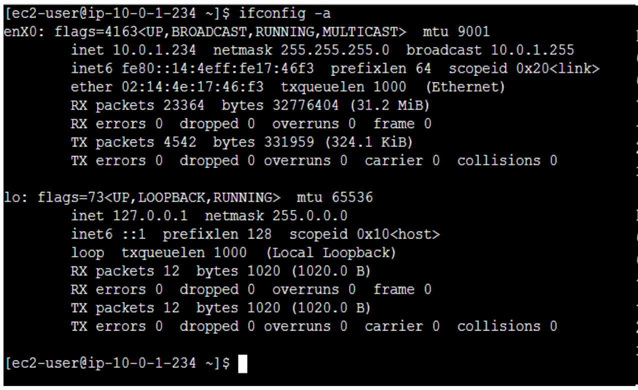
Enable

Additional charges apply when outside of free tier allowance



☑	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS
☑	MyLinux-To-VPC	i-0ebdfb54e5767e33d	Running	t2.micro	Initializing	View alarms +	us-east-2a	–

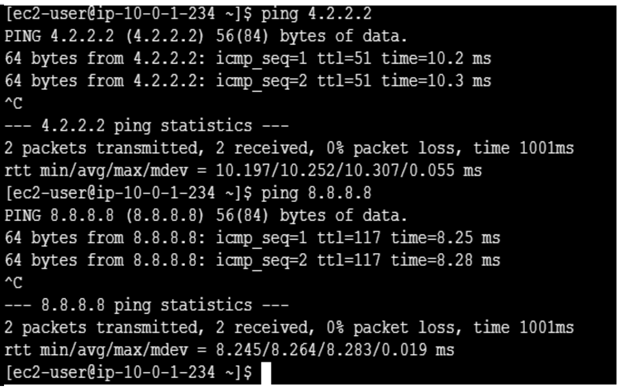
Connect to EC2 Instance and found the public subnet IP 10.0.1.234 to EC2 instance



```
[ec2-user@ip-10-0-1-234 ~]$ ifconfig -a
enX0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 9001
        inet 10.0.1.234  netmask 255.255.255.0  broadcast 10.0.1.255
        inet6 fe80::14:4eff:fe17:46f3  prefixlen 64  scopeid 0x20<link>
        ether 02:14:4e:17:46:f3  txqueuelen 1000  (Ethernet)
        RX packets 23364  bytes 32776404 (31.2 MiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 4542  bytes 331959 (324.1 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 12  bytes 1020 (1020.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 12  bytes 1020 (1020.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

[ec2-user@ip-10-0-1-234 ~]$
```



```
[ec2-user@ip-10-0-1-234 ~]$ ping 4.2.2.2
PING 4.2.2.2 (4.2.2.2) 56(84) bytes of data.
64 bytes from 4.2.2.2: icmp_seq=1 ttl=51 time=10.2 ms
64 bytes from 4.2.2.2: icmp_seq=2 ttl=51 time=10.3 ms
^C
--- 4.2.2.2 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 10.197/10.252/10.307/0.055 ms
[ec2-user@ip-10-0-1-234 ~]$ ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=117 time=8.25 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=117 time=8.28 ms
^C
--- 8.8.8.8 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 8.245/8.264/8.283/0.019 ms
[ec2-user@ip-10-0-1-234 ~]$
```