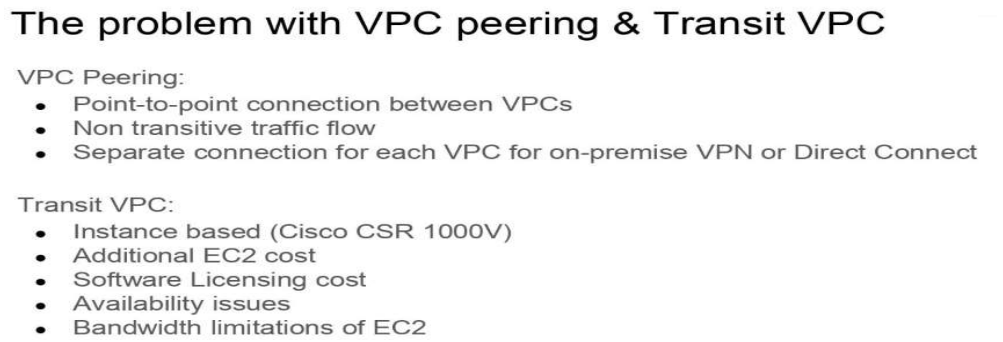
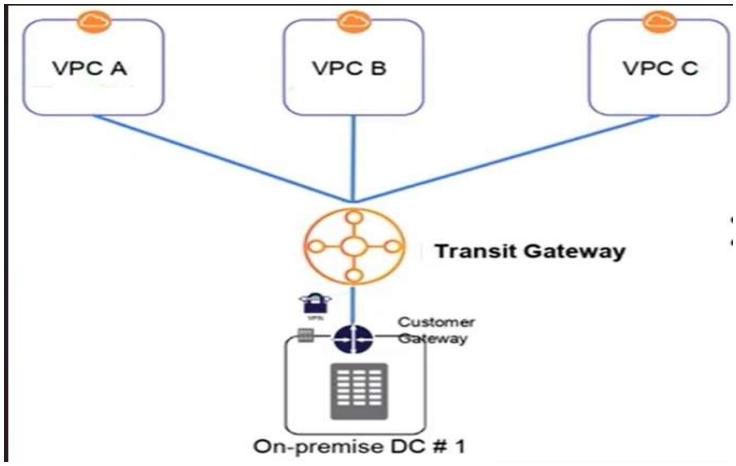
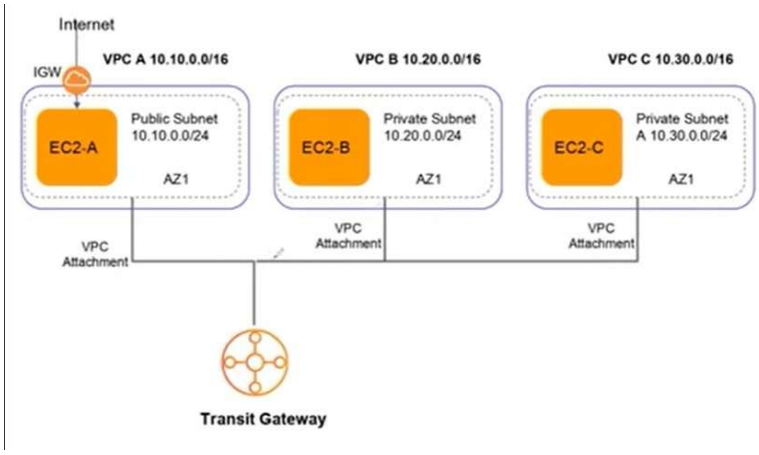
Transit Gateway

A *transit gateway* is a network transit hub that you can use to interconnect your virtual private clouds (VPCs) and on-premises networks. As your cloud infrastructure expands globally, inter-Region peering connects transit gateways together using the AWS Global Infrastructure. All network traffic between AWS data centers is automatically encrypted at the physical layer.

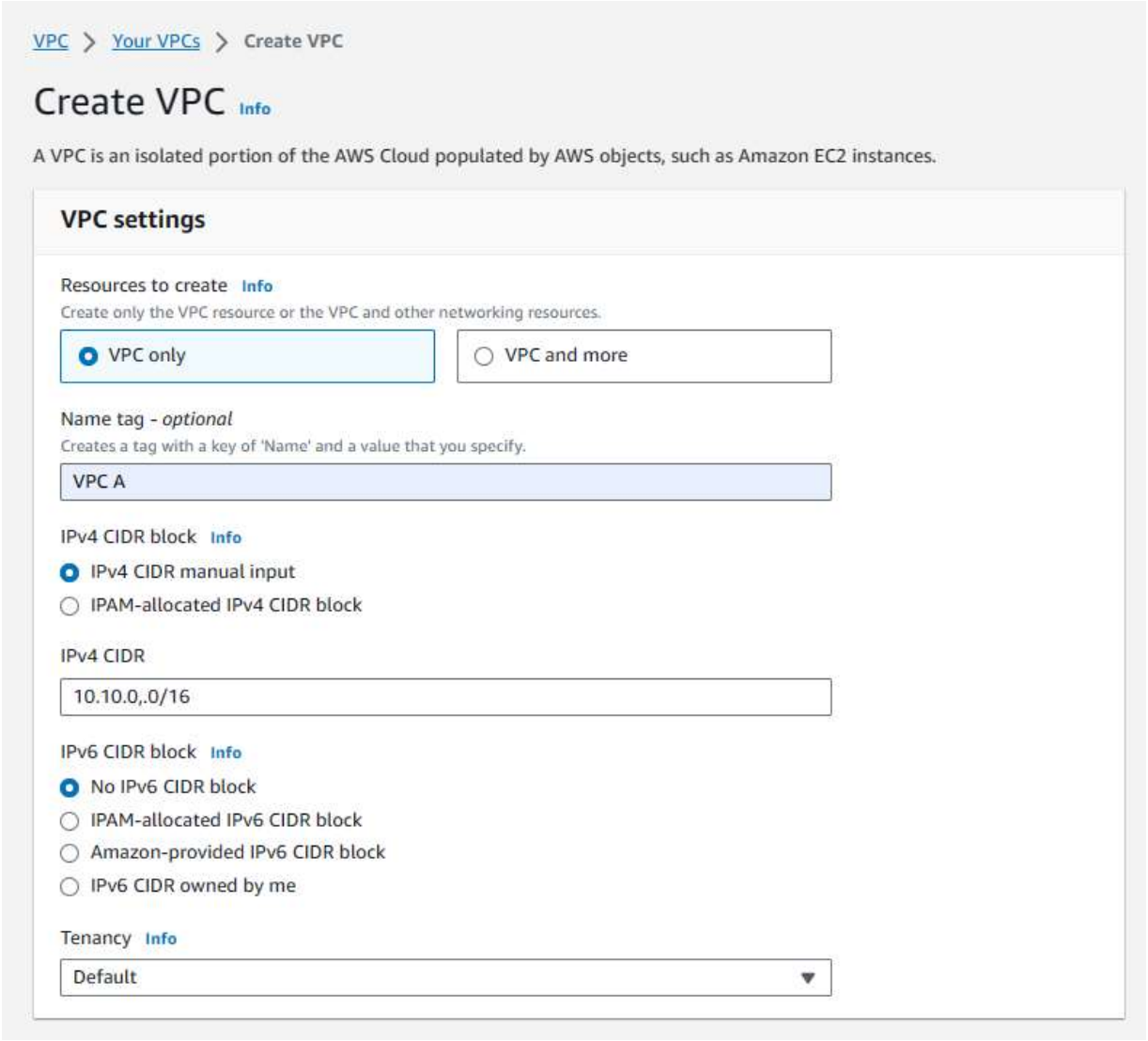


Here we're doing hands-on based on the below diagrams:



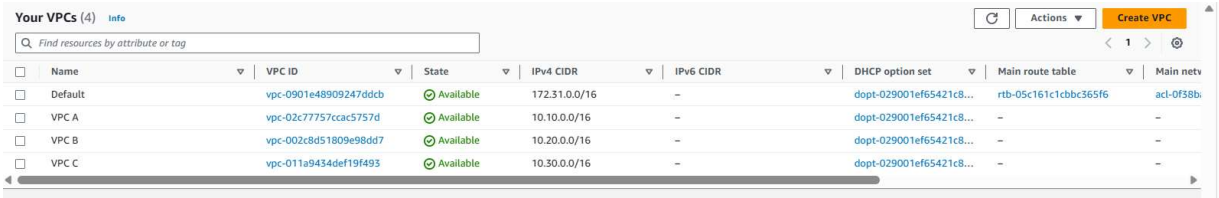


Select VPC from AWS console and Create VPC



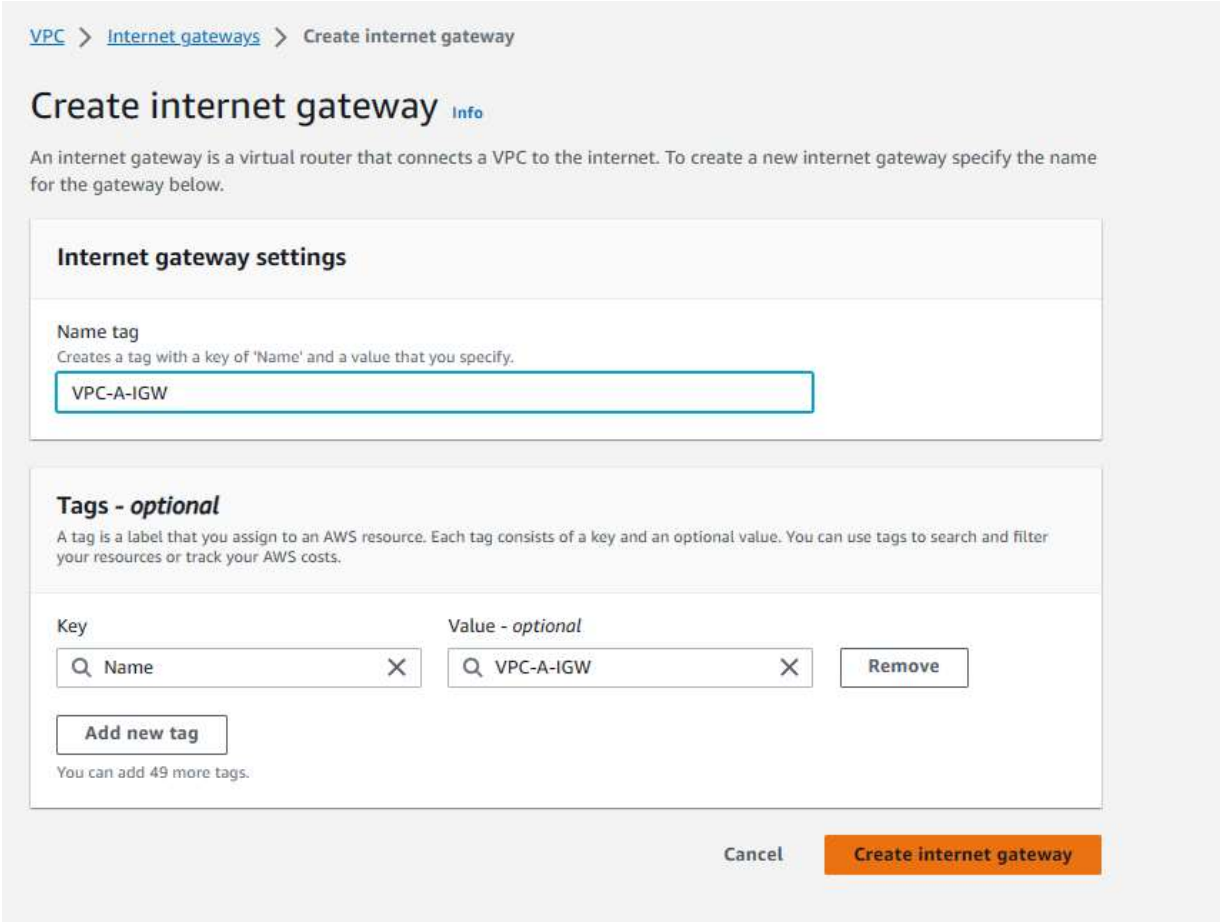
Then click on Create VPC.

In a similar way , create other 2 CPCs naming VPC B and VPC C with 10.20.0.0/16 . 10.30.0.0/16 as IP addresses respectively.

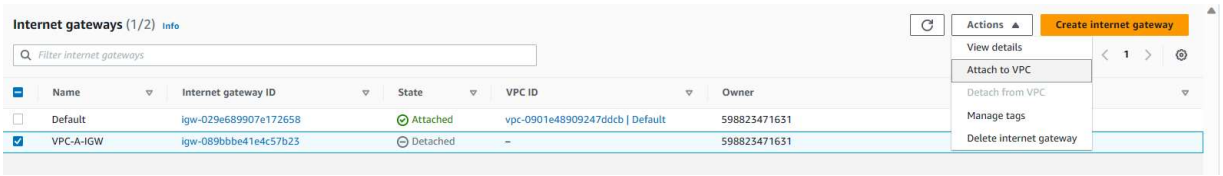


From the diagram , we can see VPC A is public and VPC B & C are private. So we need to configure Internet GateWay for VPC A.

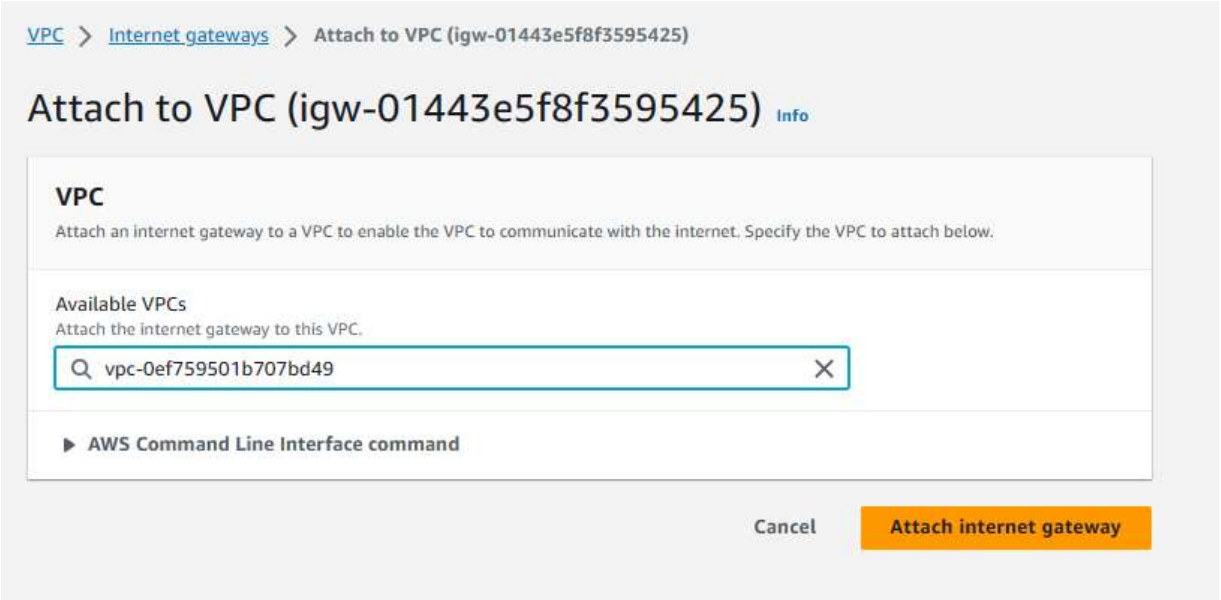
Click on Internet Gateway from the LHS panel and click on create Internet Gateway.



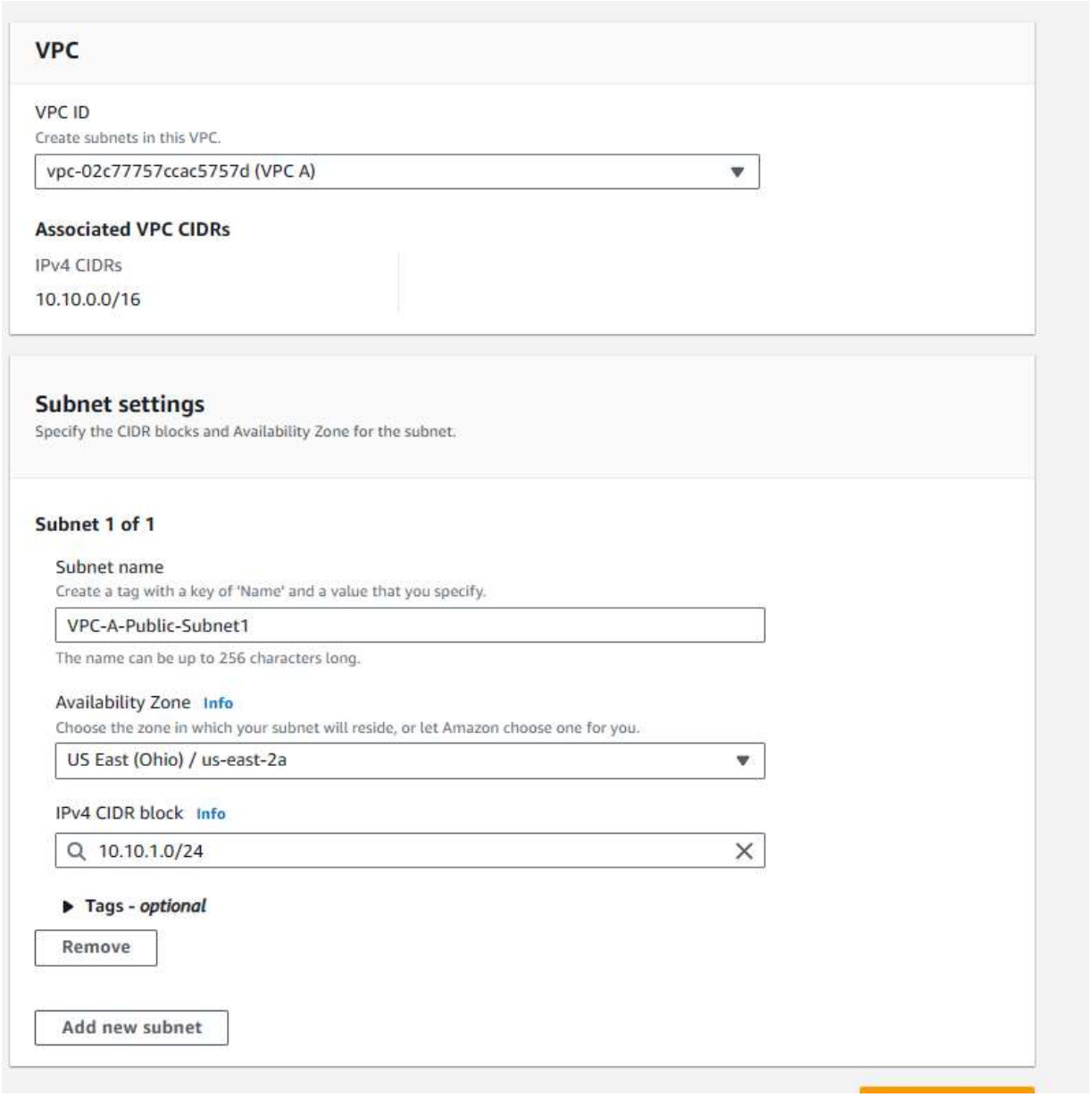
Now select the newly created IGW and select Attach to VPC from Actions.



Select VPC A from drop down and click on Attach internet gateway



Next we need to create Subnets, for that click on Subnets from the LHS panel and click on create subnet for VPC A as per below:



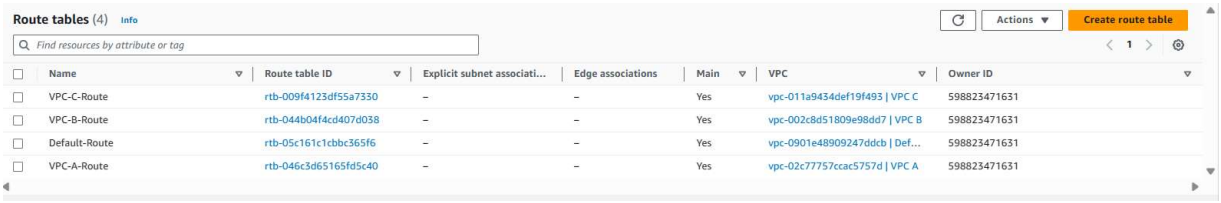
Likewise , we have to create 3 subnets for each VPC. I have created with the info below:

VPC-A-Public-Subnet1
10.10.1.0/24

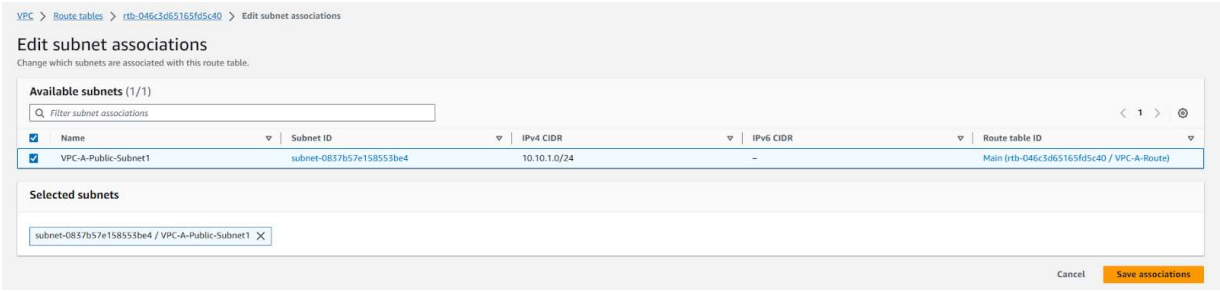
VPC-B-Private-Subnet1
10.20.1.0/24

VPC-C-Private-Subnet1
10.30.1.0/24

Now we need to add the route tables , Click on Route Tables from the LHS panel and click on create route table.

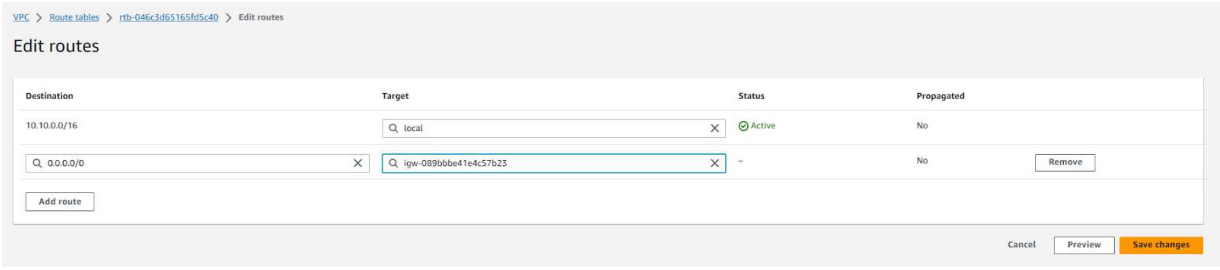


Next we have to associate the subnet with routing table. For that select VPC-A-Route -> Click on Subnet associations -> Edit subnet associations , then select VPC-A-Public-Subnet1 ->Save associations.

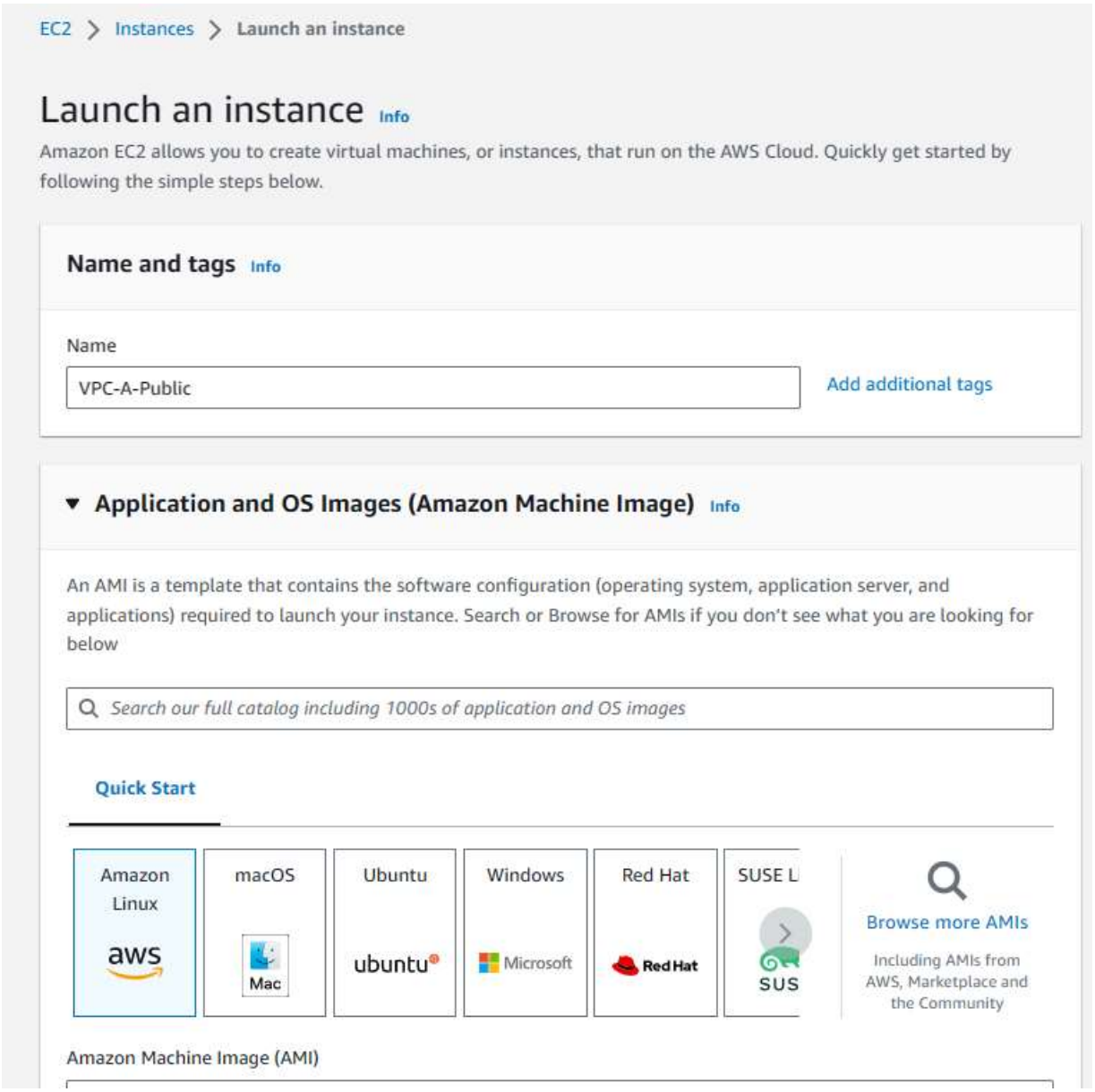


Do the same for VPC-B-Route and VPC-C-Route.

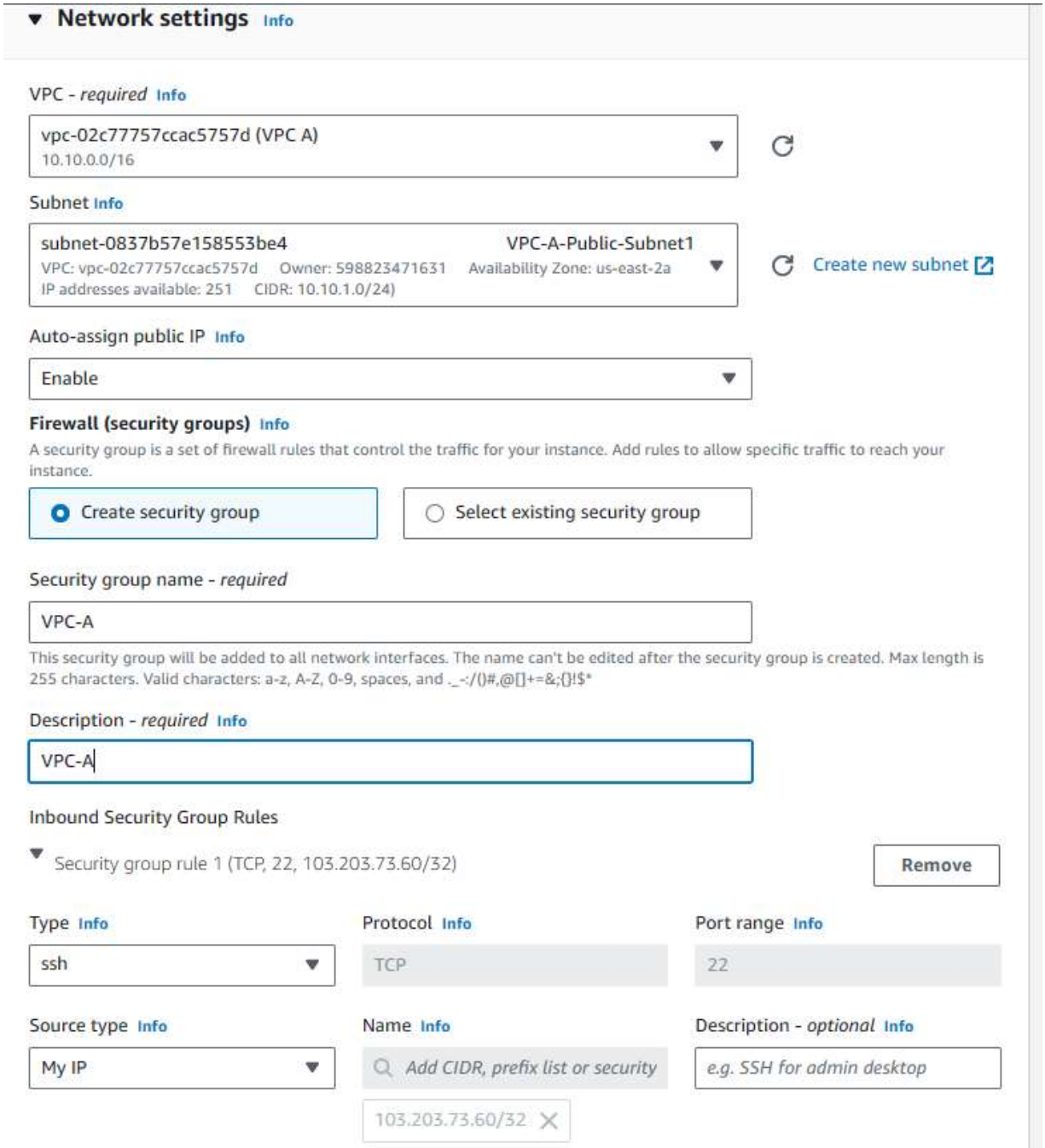
Select the VPC-A-Route and go to Routes->Edit routes and add as per below , then click on save changes.



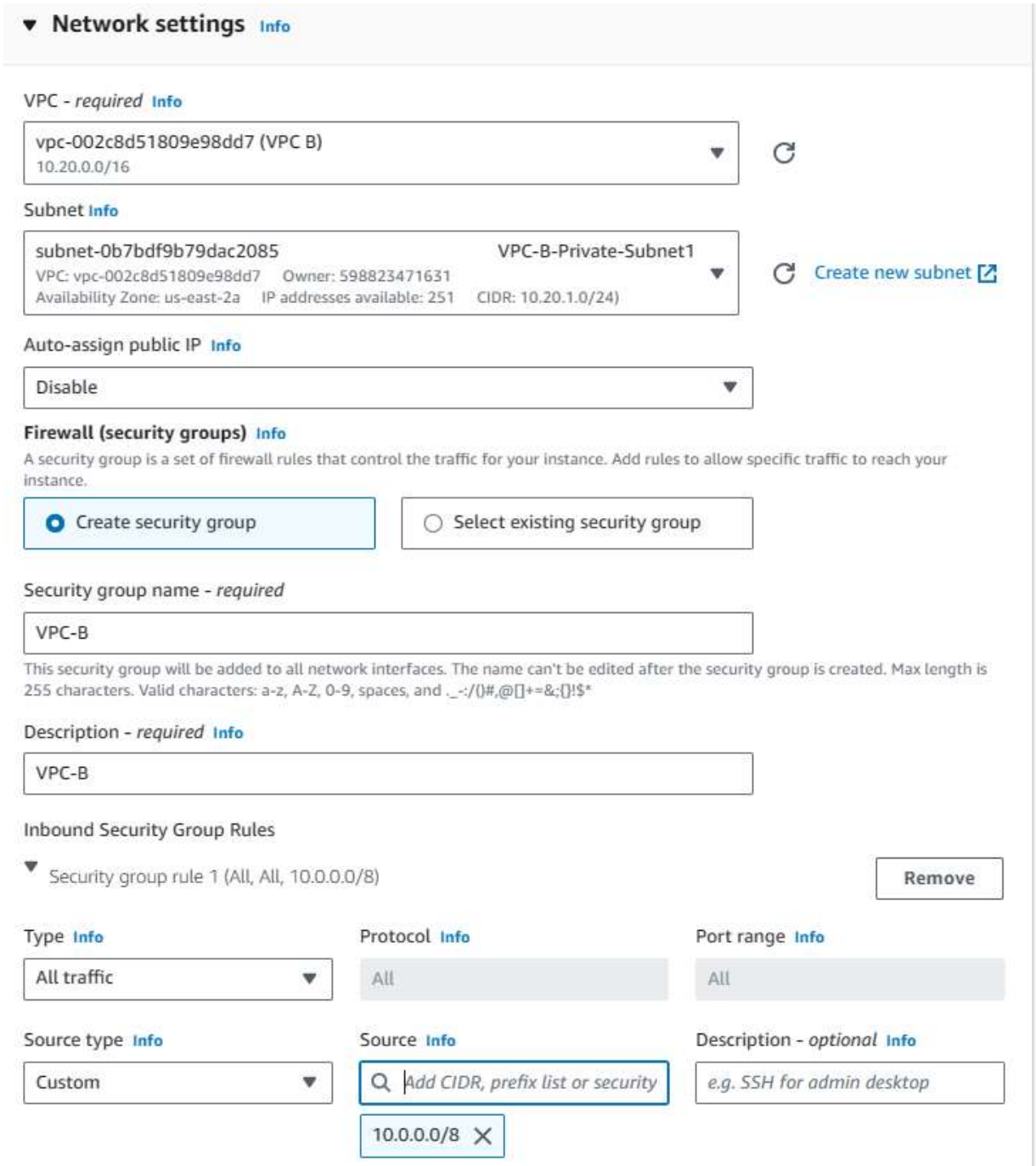
We’re all set in the VPC part , Now select EC2 from AWS console and create 2 instances as per above diagram.



Configure Network Settings as below:

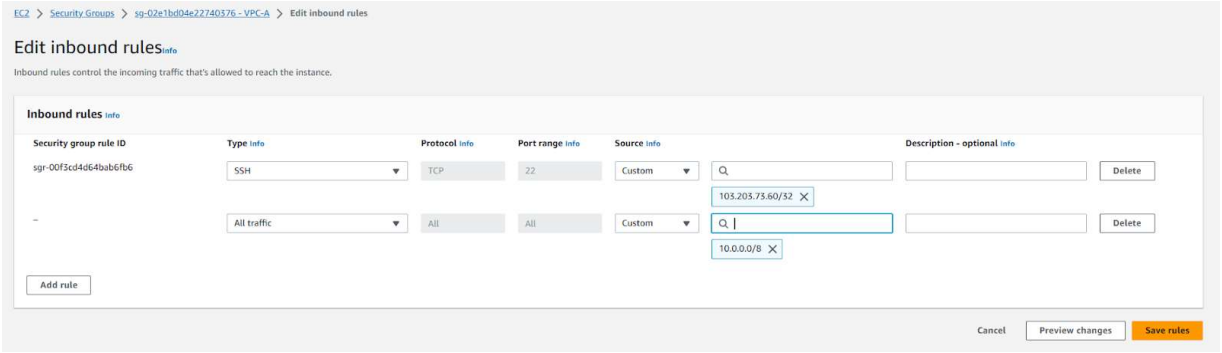


And launch the instance.
Now create VPC-B-Private as below:



Rest all settings are default or same as VPC-A-Public. Then create one more instance same as VPC-B-Private.

Also create one more inbound rule in VPC-A-Public as shown below:



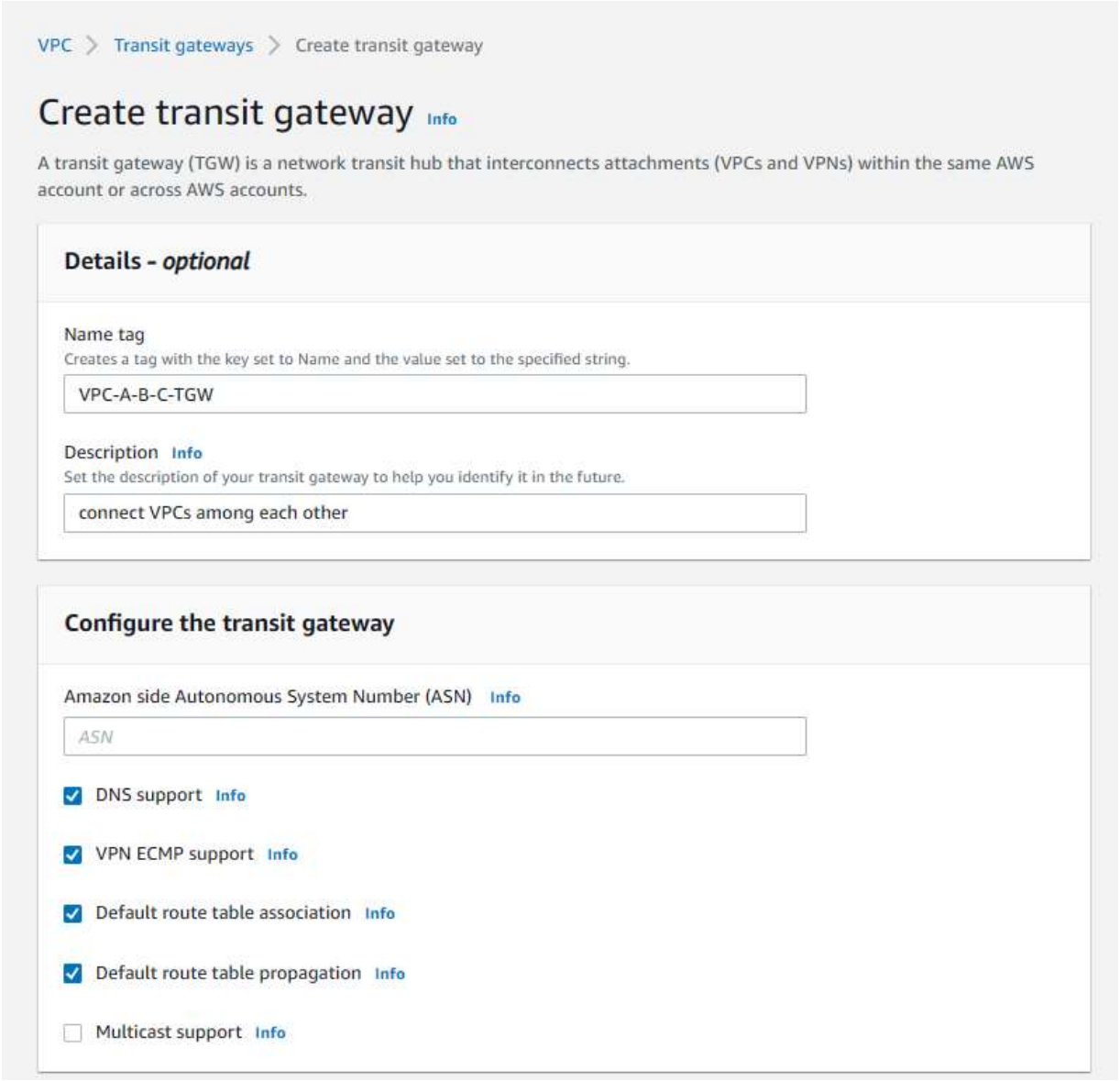
Once the instances are up and running , take the ssh connection and login as root. Then from VPC-A-Public , check if the private IPs of VPC-B-Private and VPC-C-Private are reachable. It should not be reachable as below:

```
[root@ip-10-10-1-23 ec2-user]# ping 10.20.1.233
PING 10.20.1.233 (10.20.1.233) 56(84) bytes of data.
^C
--- 10.20.1.233 ping statistics ---
5 packets transmitted, 0 received, 100% packet loss, time 4130ms

[root@ip-10-10-1-23 ec2-user]# ping 10.30.1.49
PING 10.30.1.49 (10.30.1.49) 56(84) bytes of data.
^C
--- 10.30.1.49 ping statistics ---
6 packets transmitted, 0 received, 100% packet loss, time 5192ms

[root@ip-10-10-1-23 ec2-user]#
```

Now we need to connect VPCs among each other.

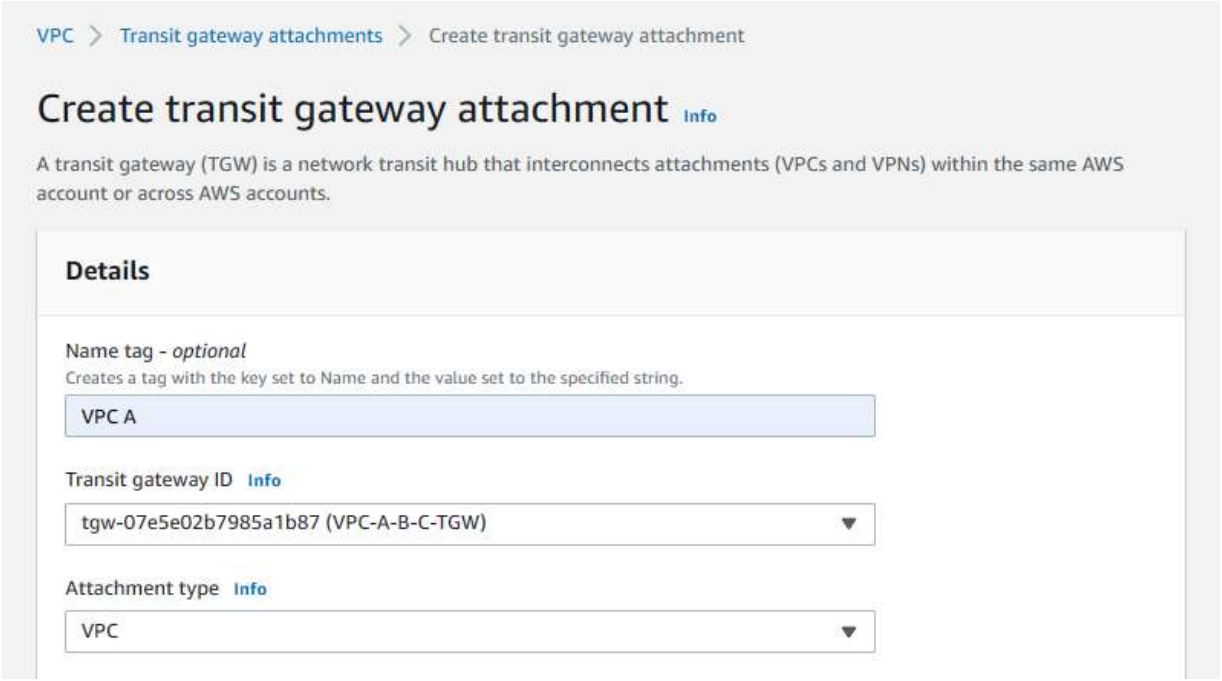
Go to VPC and click on Transit Gateway.

Create Transit Gateway as shown below

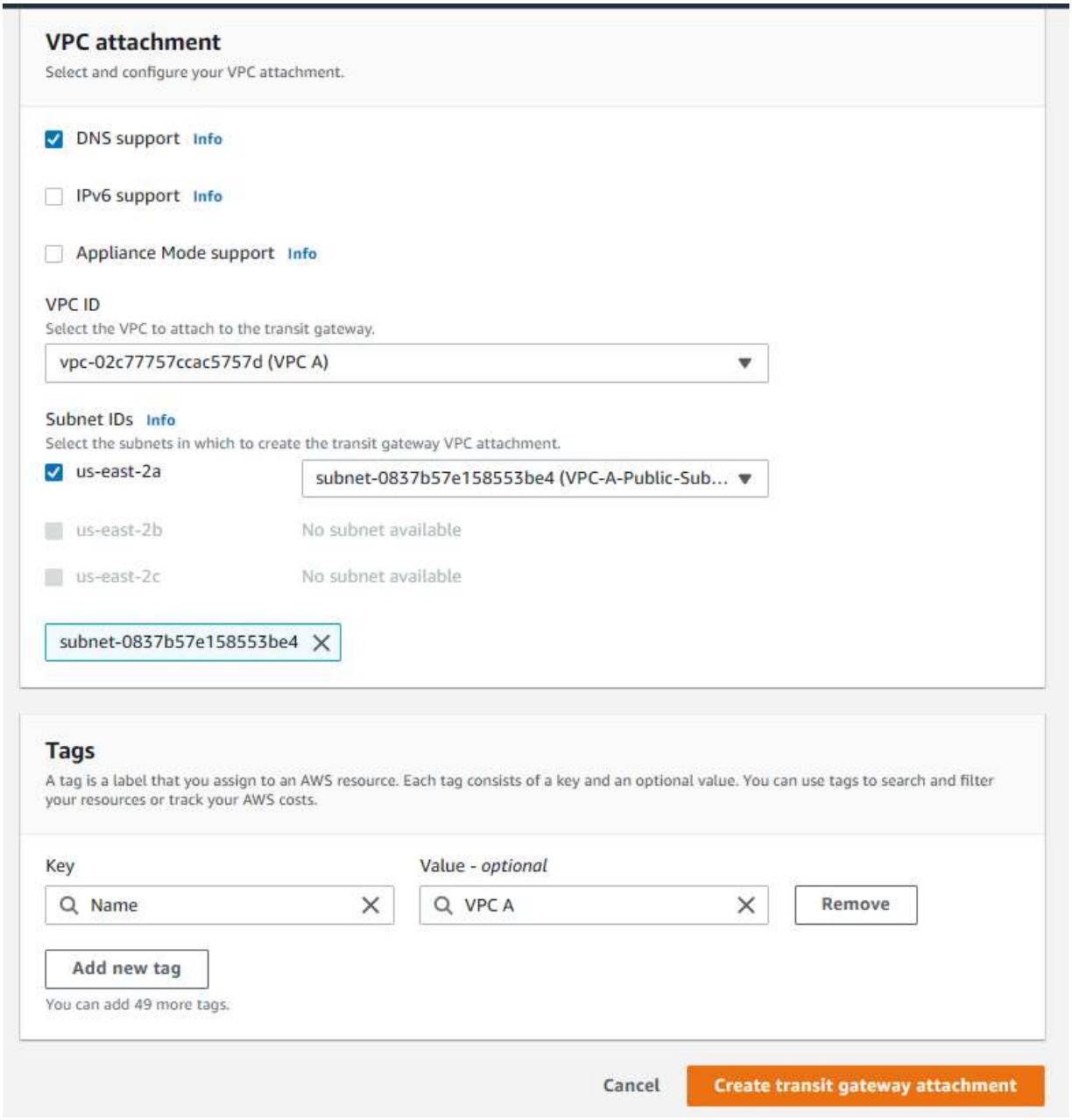


Select Transit gateway attachment from LHS panel and create Transit gateway attachment for every VPCs.

Give a name and select the transit gateway as shown below:

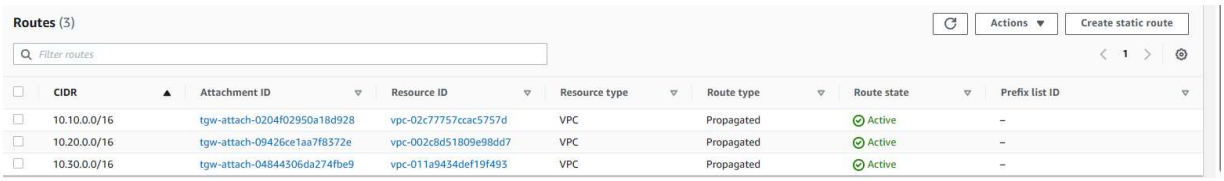


Then configure the attachment as below:

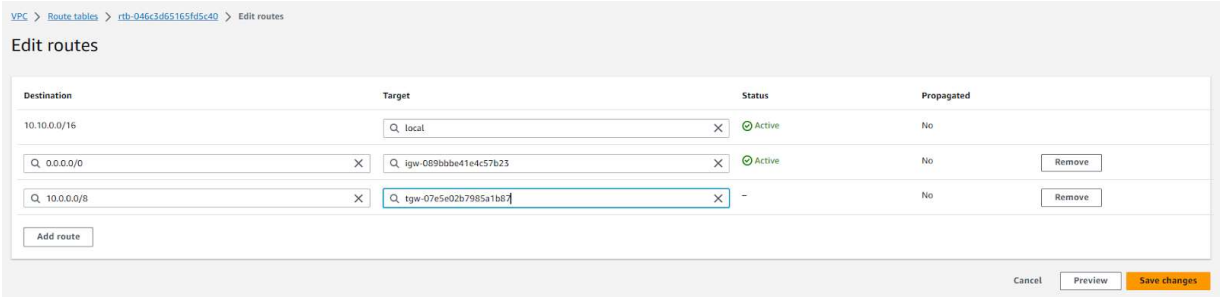


Sameway create another 2 attachments for VPC B and VPC C.

Select Transit gateway route tables from the LHS panel and go to the routes , we can see the VPC CIDR have been listed there.



Now go to Route Tables, select VPC A and add route as shown below:



Update the same for VPC B and VPC C.

Now check the connectivity from VPC-A-Public , check if the private IPs of VPC-B-Private and VPC-C-Private are reachable.

```
[root@ip-10-10-1-23 ec2-user]#
[root@ip-10-10-1-23 ec2-user]# ping 10.20.1.233
PING 10.20.1.233 (10.20.1.233) 56(84) bytes of data.
64 bytes from 10.20.1.233: icmp_seq=1 ttl=126 time=1.12 ms
64 bytes from 10.20.1.233: icmp_seq=2 ttl=126 time=0.546 ms
64 bytes from 10.20.1.233: icmp_seq=3 ttl=126 time=0.638 ms
^C
--- 10.20.1.233 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2079ms
rtt min/avg/max/mdev = 0.546/0.766/1.115/0.249 ms
[root@ip-10-10-1-23 ec2-user]#
[root@ip-10-10-1-23 ec2-user]#
[root@ip-10-10-1-23 ec2-user]#
[root@ip-10-10-1-23 ec2-user]# ping 10.30.1.49
PING 10.30.1.49 (10.30.1.49) 56(84) bytes of data.
64 bytes from 10.30.1.49: icmp_seq=1 ttl=126 time=0.836 ms
64 bytes from 10.30.1.49: icmp_seq=2 ttl=126 time=0.746 ms
64 bytes from 10.30.1.49: icmp_seq=3 ttl=126 time=0.768 ms
64 bytes from 10.30.1.49: icmp_seq=4 ttl=126 time=0.658 ms
^C
--- 10.30.1.49 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3097ms
rtt min/avg/max/mdev = 0.658/0.752/0.836/0.063 ms
[root@ip-10-10-1-23 ec2-user]#
```

We can see it is reachable. This is how the Transit gateway works..!!!!