Create 2 VPC's (VPC1 should be public and VPC 2 should be private) communication between 2 VPC's ping should be worked, AWS peering.

Task:

Create a 2 VPC's (VPC1 should be public and VPC 2 should be private) communication between 2 VPC's ping should be worked, AWS peering:

- **VPC (Virtual Private Cloud):**

VPC allows you to **create a private network within the AWS cloud. You can define your own network topology, set up subnets, and configure network security and routing.**

- **Route Tables:**

Route tables are an essential component of the Virtual Private Cloud (VPC) networking infrastructure. They are **used to control the traffic flow within a VPC, determine how traffic is routed between subnets, and define how traffic is directed to the internet or other AWS resources.**

- **Internet Gateway:**

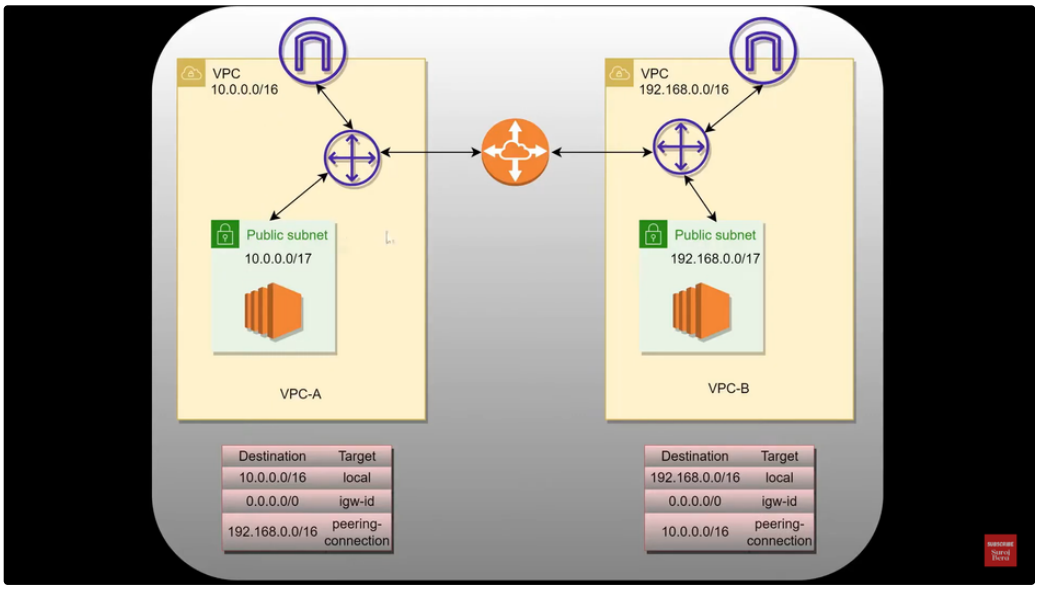
Internet Gateway is a crucial component for enabling communication between resources within your Virtual Private Cloud (VPC) and the public internet.

- **VPC Peering Connection:**

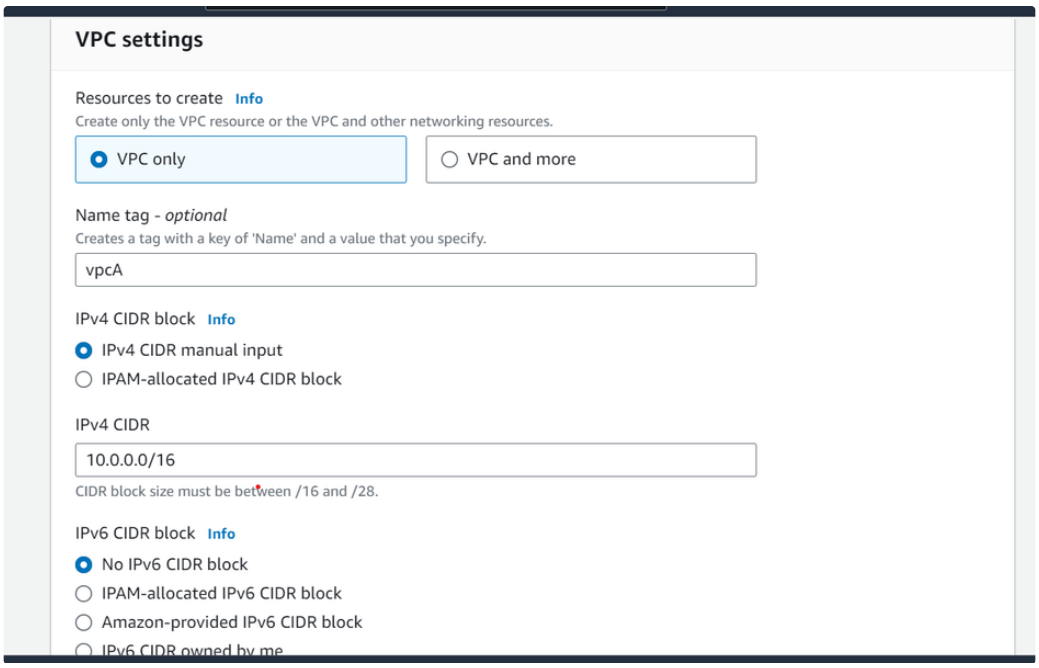
VPC Peering Connection is a network connection between two Virtual Private Clouds (VPCs) that allows them to communicate with each other as if they were on the same network. **VPC Peering is a valuable feature for connecting VPCs within the same AWS account or across different accounts, enabling secure and private communication between resources in separate VPCs.**

- **Security Groups:**

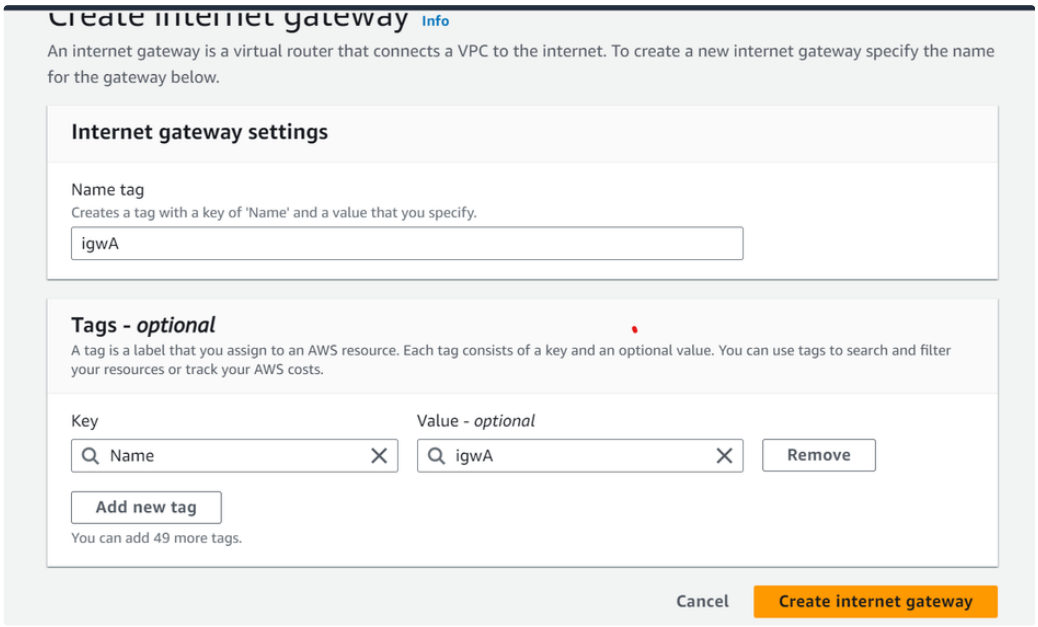
Security Group is a fundamental component for **controlling inbound and outbound network traffic to and from Amazon Elastic Compute Cloud (EC2) instances and other AWS resources. Security Groups act as virtual firewalls for your resources, allowing you to define rules that specify which traffic is allowed or denied.**

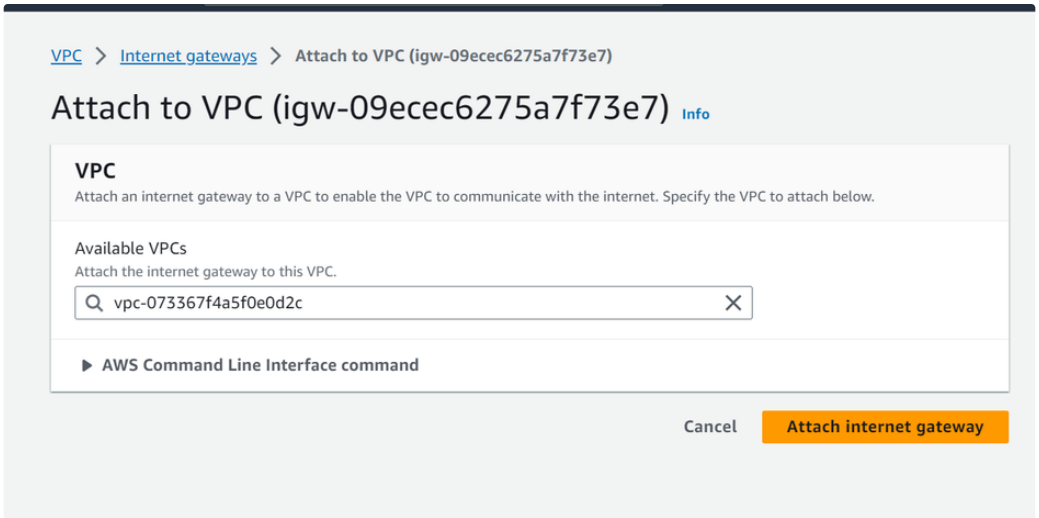


- Create 2 VPC's named as vpcA and vpcB, With IPV4 CIDR Manual Input. IPV4 for vpcA will be 10.0.0.0/16, IPV4 for vpcB will be 192.168.0.0/16.

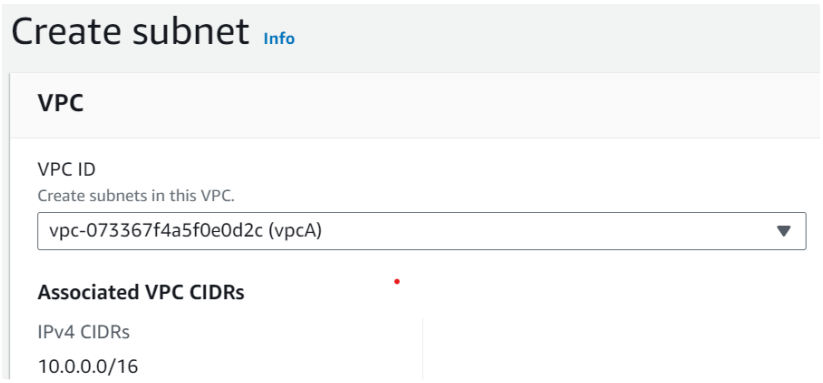


- Now create two internet gateways with names of igwA and igwB. Attach internet gateways with the VPC's created.

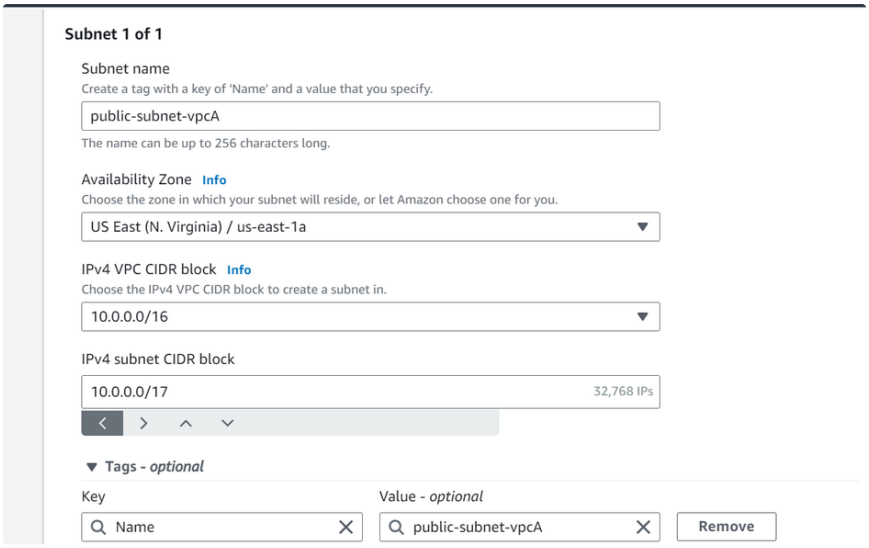




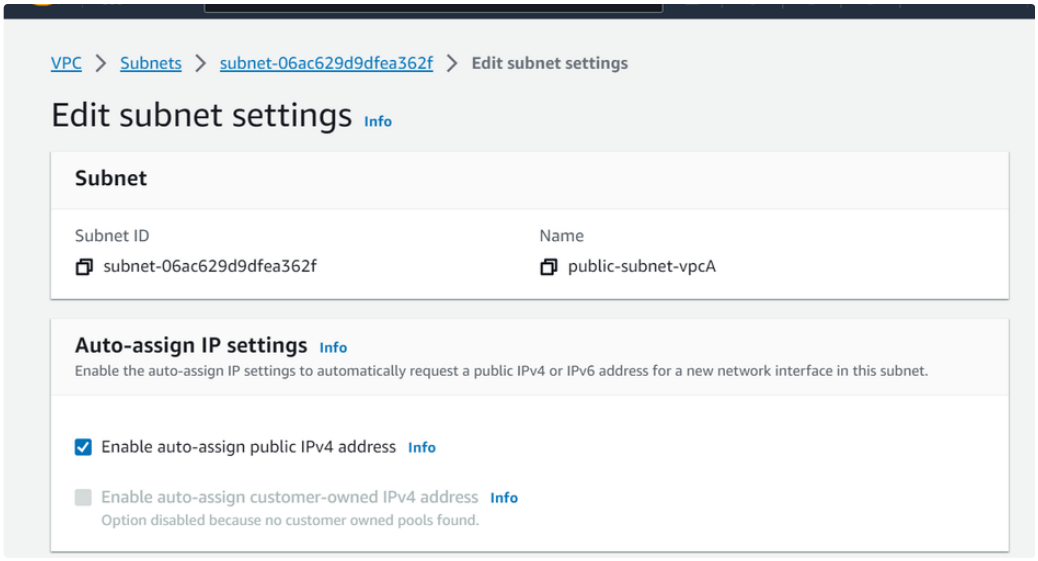
- Now create two subnets in two different VPC's, those will be public subnets. VPC ID should be given individually to both as vpcA and vpcB.



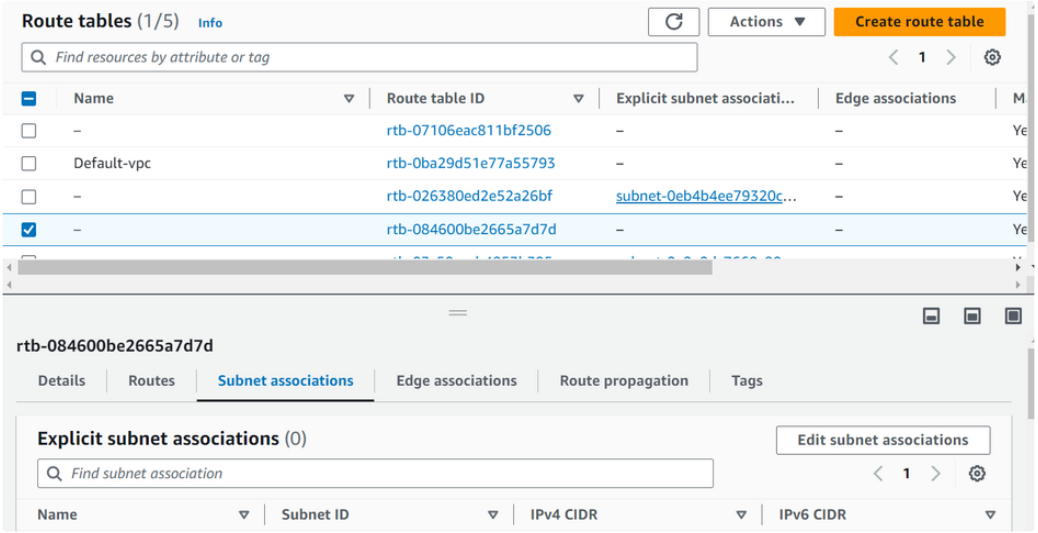
- Subnet names of vpcA is public-subnet-vpcA, name of B will be public-subnet-vpcB. Select the availability zone, Mention IPv4 subnet CIDR block as 10.0.0.0/17 and 192.168.0.0/17 and create subnets.



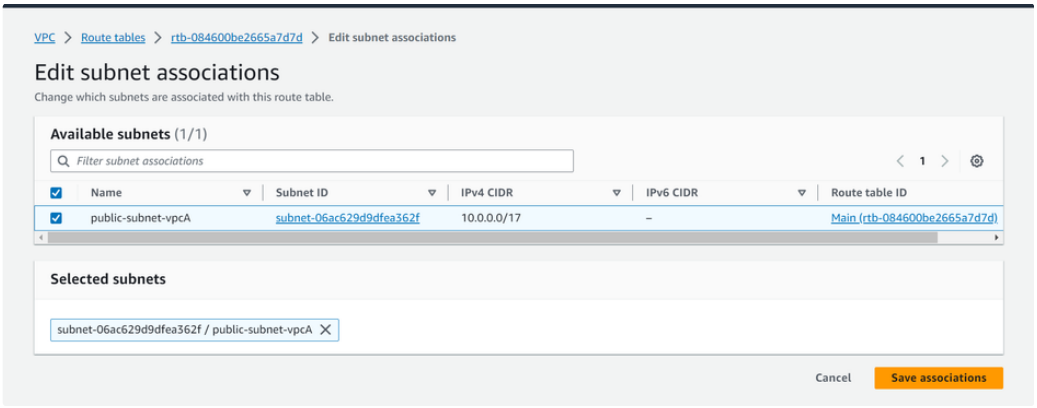
- After creating the subnets, select the subnet and edit subnet settings, Enable the auto-assign IP settings to automatically request a public IPv4 or IPv6 address for a new network interface in this subnet.



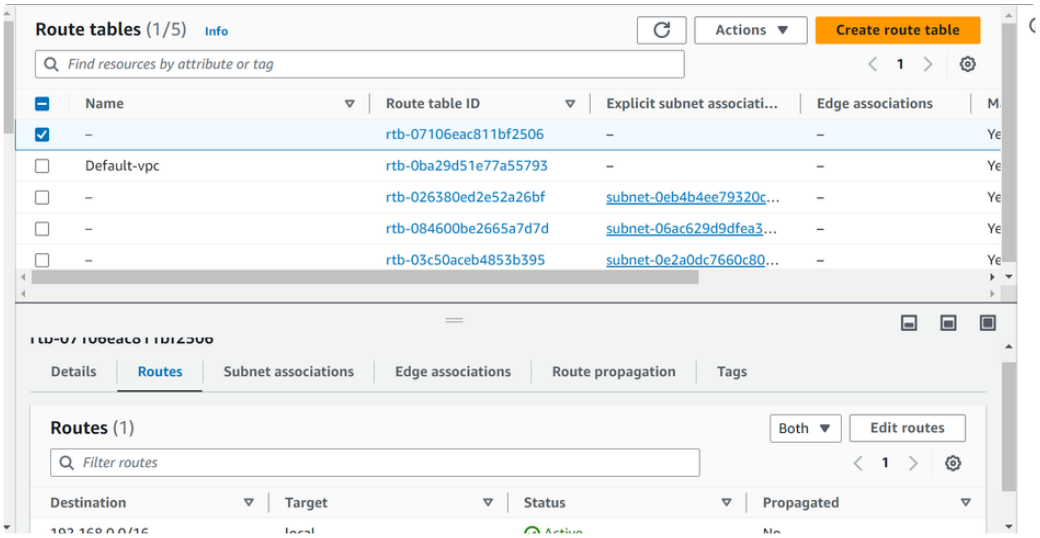
- Now go to route table, here we can find the route tables of vpcA, vpcB. Select the route table and go to subnet associations and navigate to edit subnet associations.



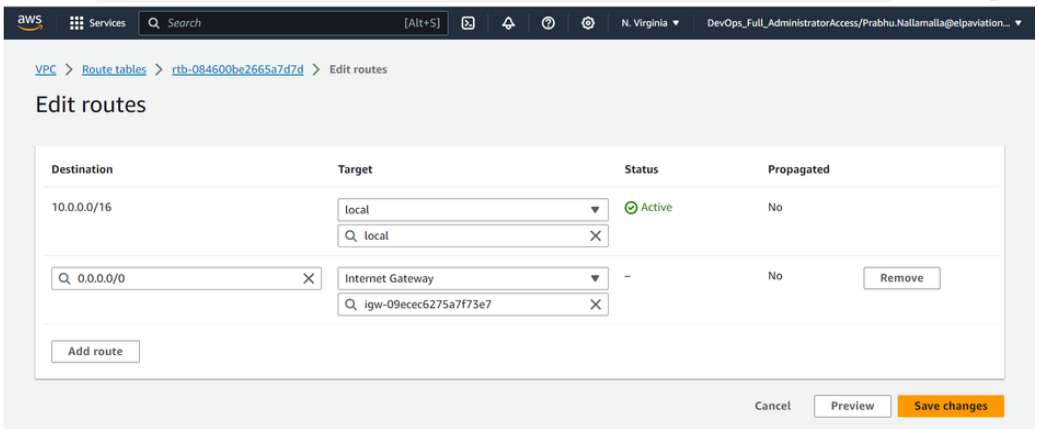
- And attach the subnet with the route table by selecting it and save the associations.



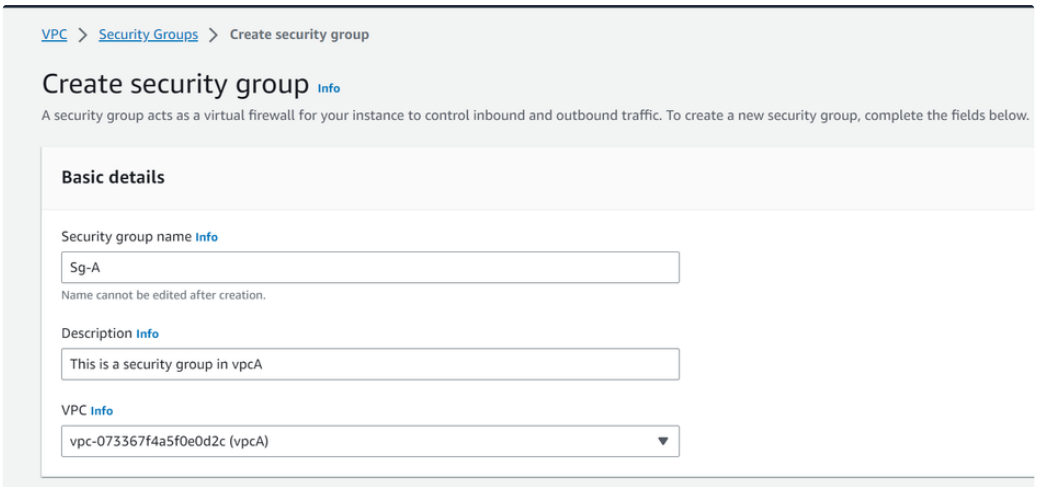
- Now go to the routes, and edit the routes. Defaultly the destination is 10.0.0.0/16 and target is local, which means it will works for local network.



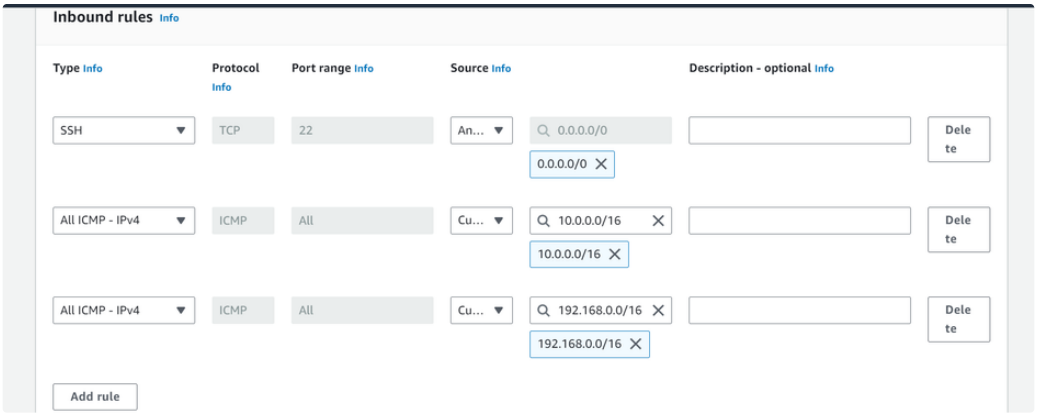
- Now we need to add route, by giving destination as 0.0.0.0/0 and target will be internet gateway for igwA for vpcA and igwB for vpcB



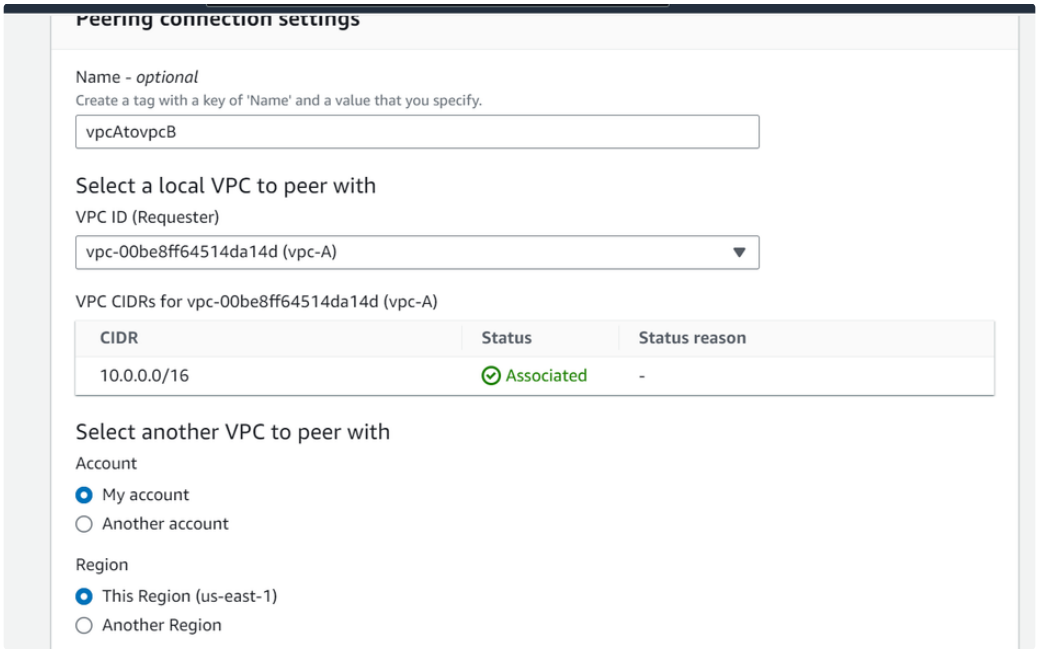
- Create two security groups with names Sg-A, description will be This is a security group in vpcA. And using vpcA.
- Create two security groups with names Sg-B, description will be This is a security group in vpcB. And using vpcB.

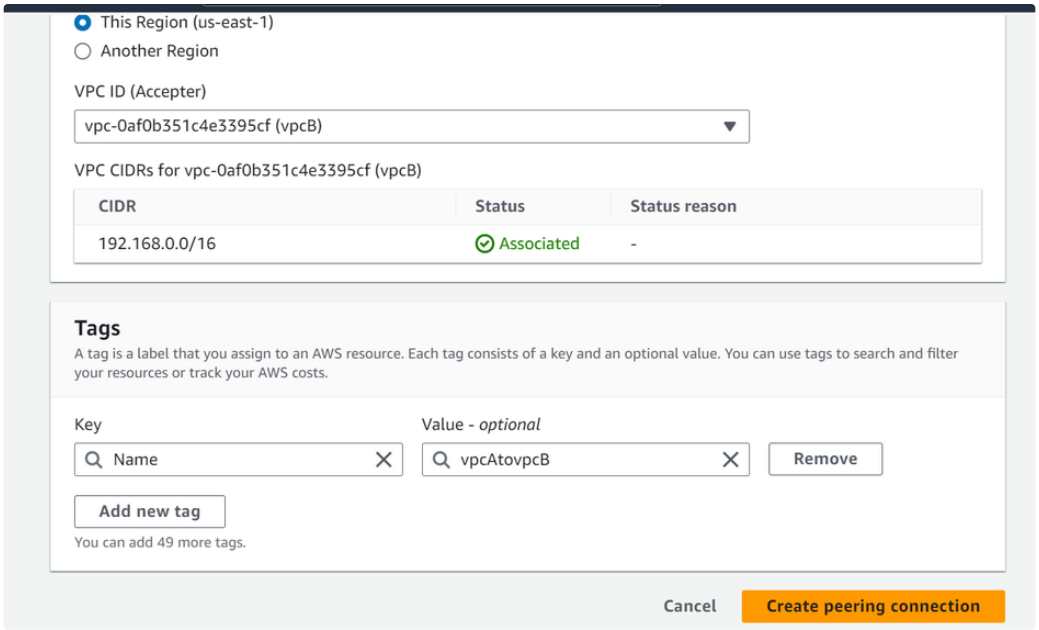


- Now create the inbound rules of SSH, All ICMP-IPv4, using 10.0.0.0/16 and 192.168.0.0/16 in both vpcA and vpcB. Outbound rules will be default.

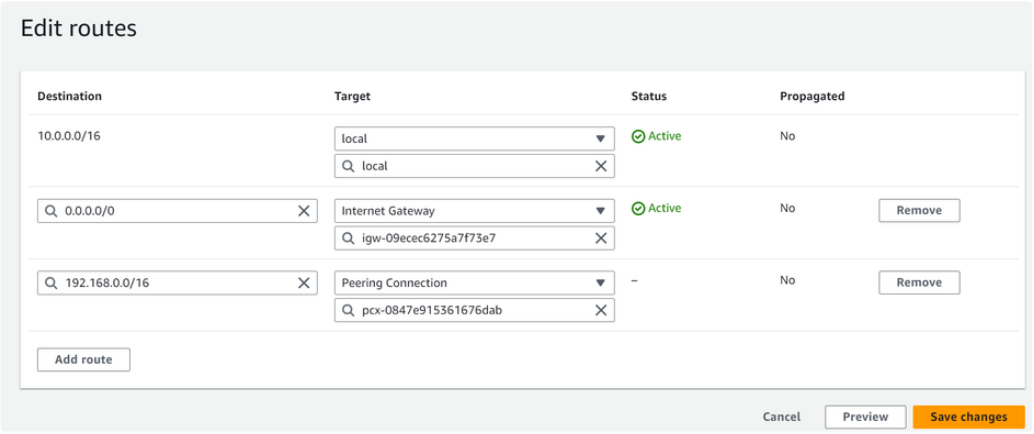


- Now create two EC2 instances, proceed without a key pair. And edit network settings, select the respective VPC, we can find the respected subnet also.
- Auto assign public IP is enabled. Select existing security group, and create EC2 instances and launch the instances.
- Check the internet connection by using ping google.com.
- Now try to ping with private IP of same instance. And it will work.
- Now try to ping with opposite instance private IP's, here it won't work as expected.
- Create a peering connection, give a name as vpcAtovpcB, and select vpcA for requester and vpcB as accepter.

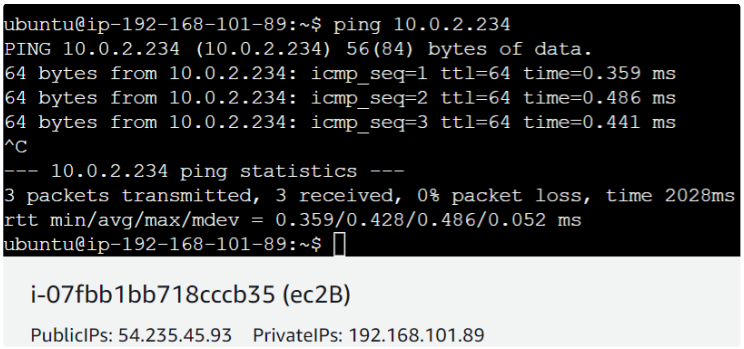




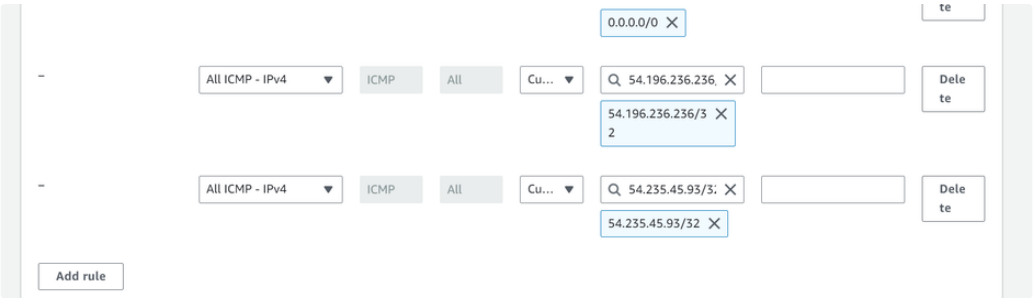
- After creating peering connection, select the connection and accept it. Now click on edit the route tables go to route tables, select and edit the route add 192.168.0.0/16 with target of peering connection of vpcAtovpcB. repeat the same step with 10.0.0.0/16 in route table B.



- Go to ec2 instances and try to ping the IP address of different networks, now it will work.



- Now try to ping using the public IP's, it will not work. Go to security group and edit the inbound rule. By ICMP, give target as public IP of ec2-A and ec2-B.
- Go to security group and edit the inbound rule. By ICMP, give target as public IP of ec2-B and ec2-A.



- Now check the ping connection by using the both public IP's on both the instances it will work.