**VPC PEERING**

**VPC Peering** is a networking connection between two Virtual Private Clouds (VPCs) that allows traffic to be routed privately using private IP addresses.

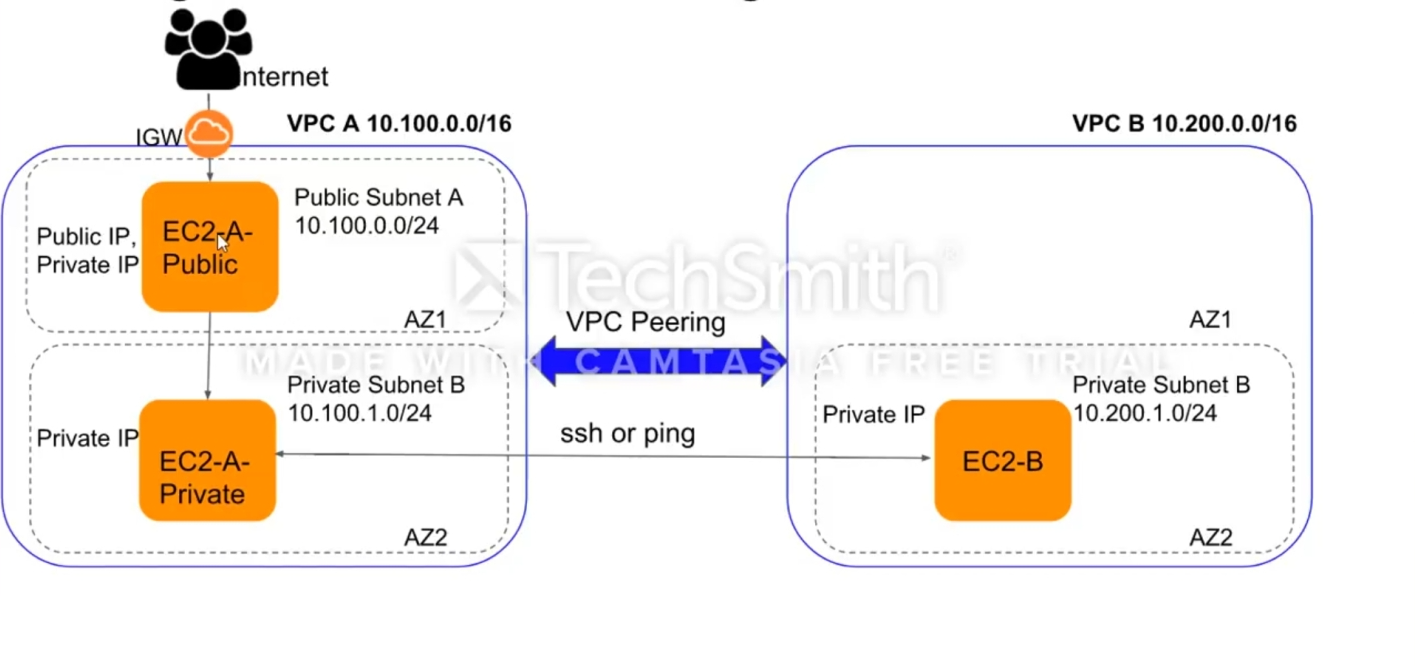
It enables resources in one VPC to communicate with resources in another VPC without using internet or an external gateway.

Peering is only possible if the CIDR blocks of the two VPCs do not overlap.

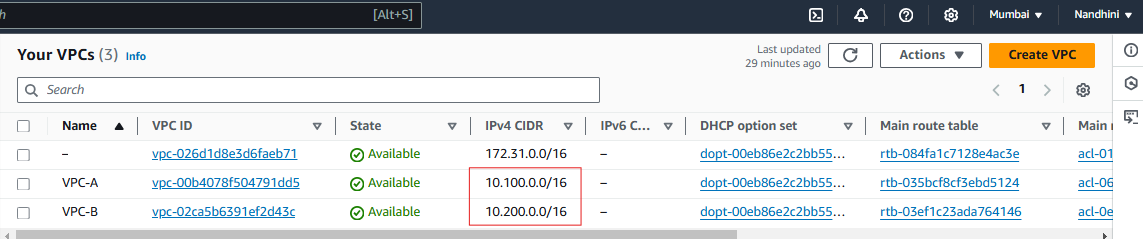
**Key Concepts of VPC Peering**

1. **Direct Communication**: Peered VPCs can communicate with each other directly using their private IP addresses.
2. **No Transitive Peering:** AWS VPC Peering is non-transitive. This means if VPC A is peered with VPC B, and VPC B is peered with VPC C, VPC A cannot communicate with VPC C through VPC B.
3. **Cross-Region and Cross-Account Peering:** AWS supports both cross-region and cross-account peering, allowing you to peer VPCs across different AWS regions and accounts.

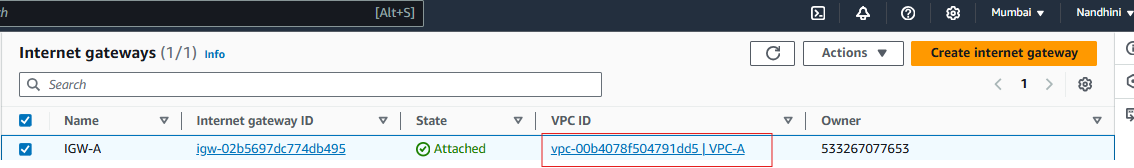
*VPC peering connection between instances of two private subnets from different VPC’s has been done based on the below architecture,*



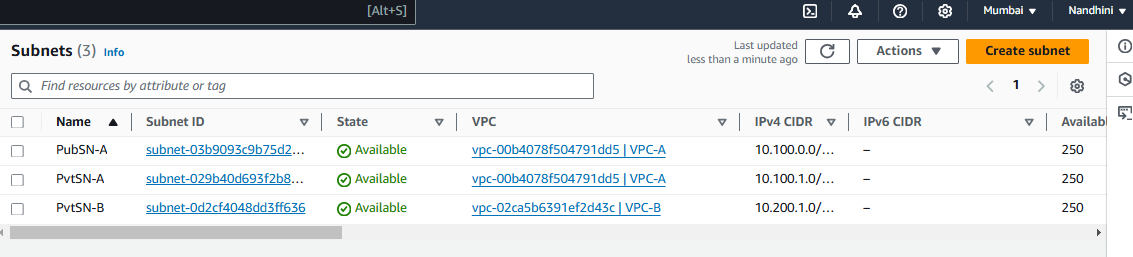
**Step 1: Create two VPC’s in same region with different CIDR blocks.**

****

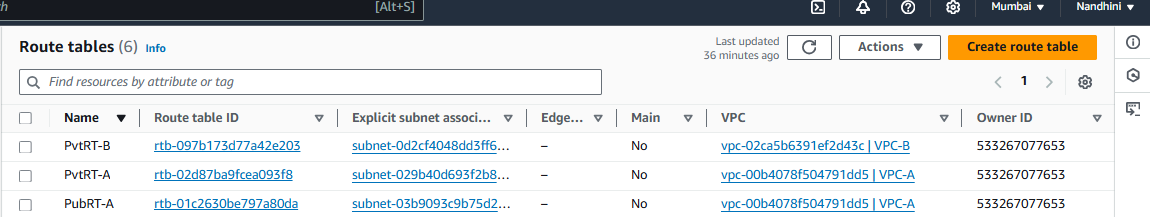
Attached IGW to VPC-A,



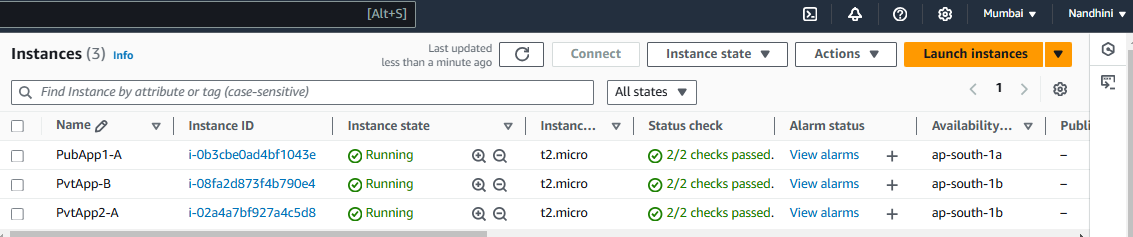
**Step 2: Created subnets fro both VPC A & B with required CIDR blocks,**

****

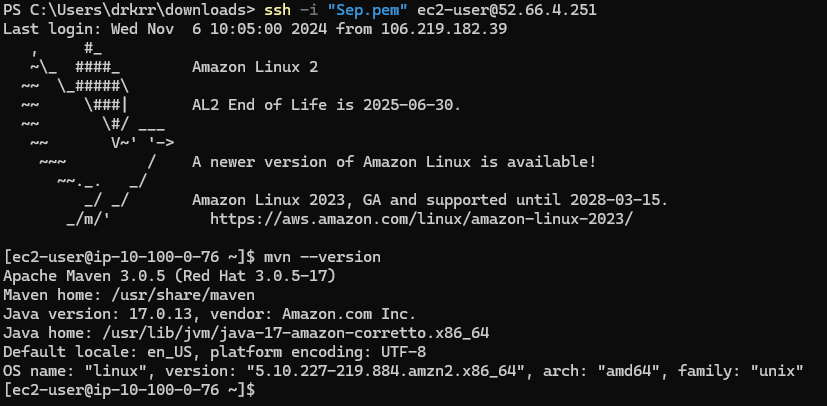
**Step 3: Created Route tables and done subnet association with concern subnets.**

****

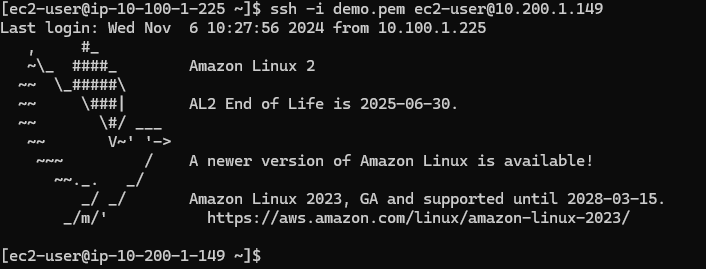
**Step 4: Created Instances in both VPC’s according to the architecture,**

****

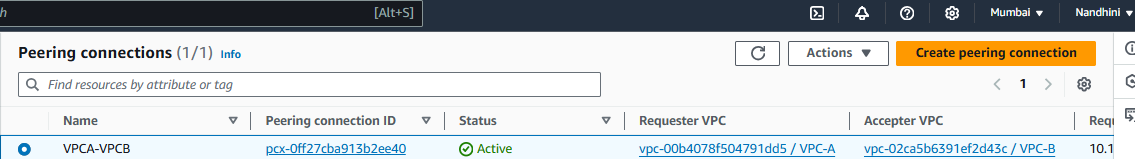
**Step 5: Logged into Public subnet instance from VPC-A, and checked for an internet connection.**

****

**Step 6: Jumped from Public subnet instances to Private subnet instance in same VPC by using .pem key.**

****

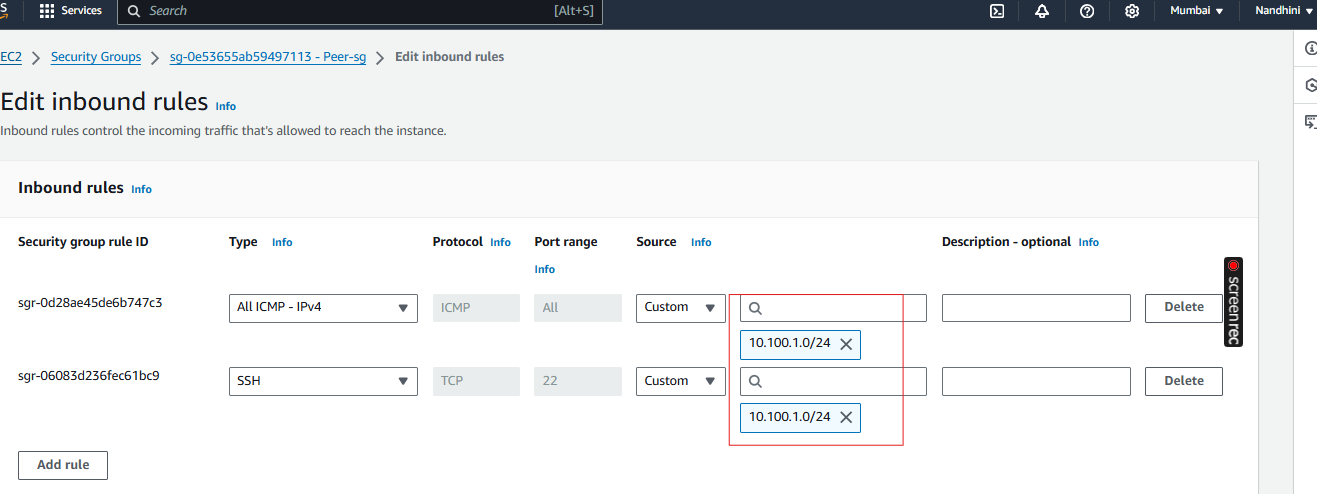
**Step 7: Tried jumping from Private subnet instance of VPC A to Private subnet instance of VPC B using .pem key, but it is not possible because both instances are from different VPC’s, to overcome this issue we use VPC peering.**

****

**Step 8: Edit Security group inbound rule of both instance.**

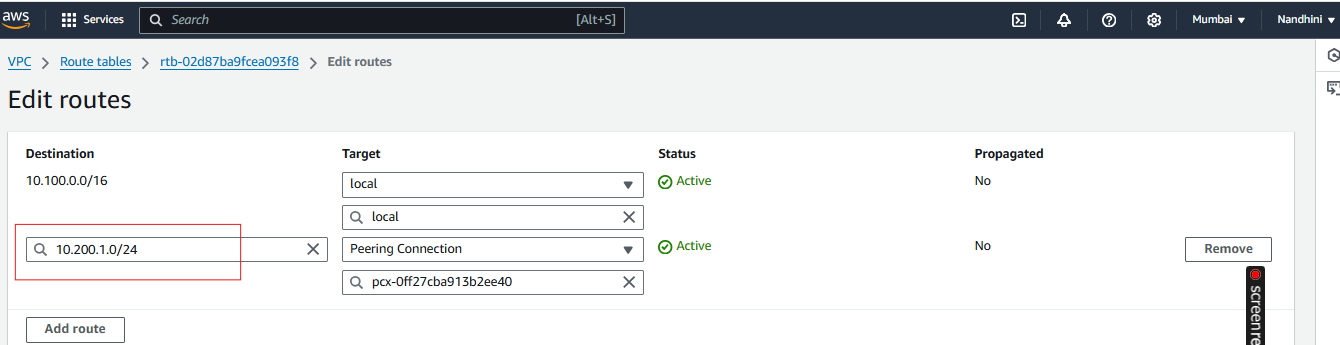
For Ssh and ping edit inbound rules with private subnet CIDR blocks.

**Private Subnet Instance – VPC B:**

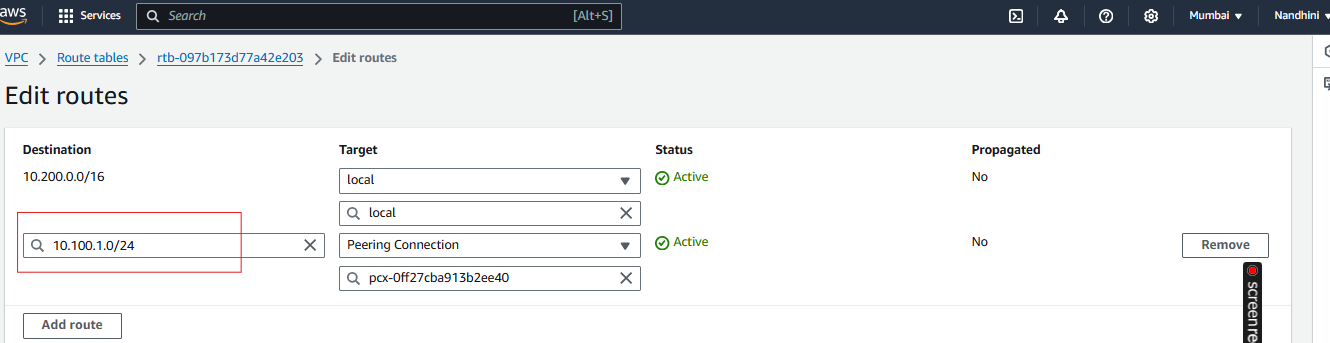


**Step 9: Edit Route tables of both VPC private subnet instances with a peering connection.**

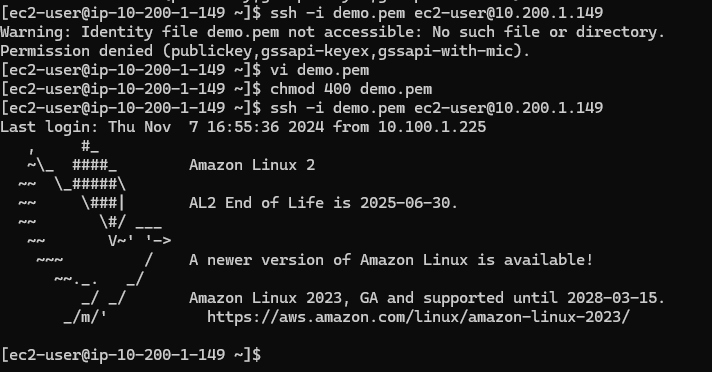
**Private route table –VPC A**

****

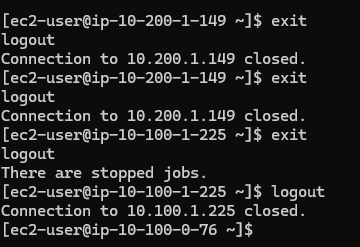
**Private route table – VPC B**

****

After VPC peering done, we can able to access VPC B private subnet instance using the pem key of VPC A.

****

**Step 10: Logged out from all the instances.**

****