**Challenge -02 - TGW**

**Use Case: Multi-Region VPC Architecture with Transit Gateway and VPC Endpoints**

**Scenario:**

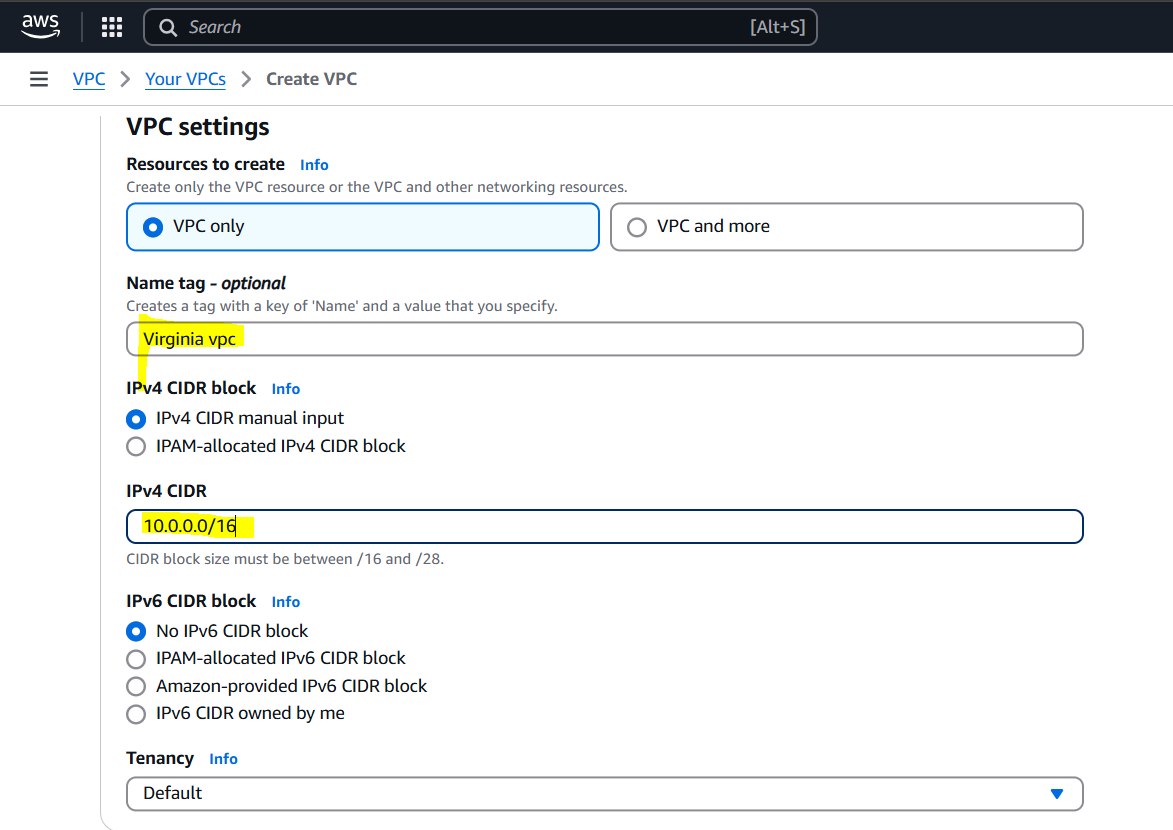
A large organization is migrating on-premises workloads to AWS. It requires secure communication between VPCs across 4 different regions and private access to AWS services without using the public internet.

**Objectives:**

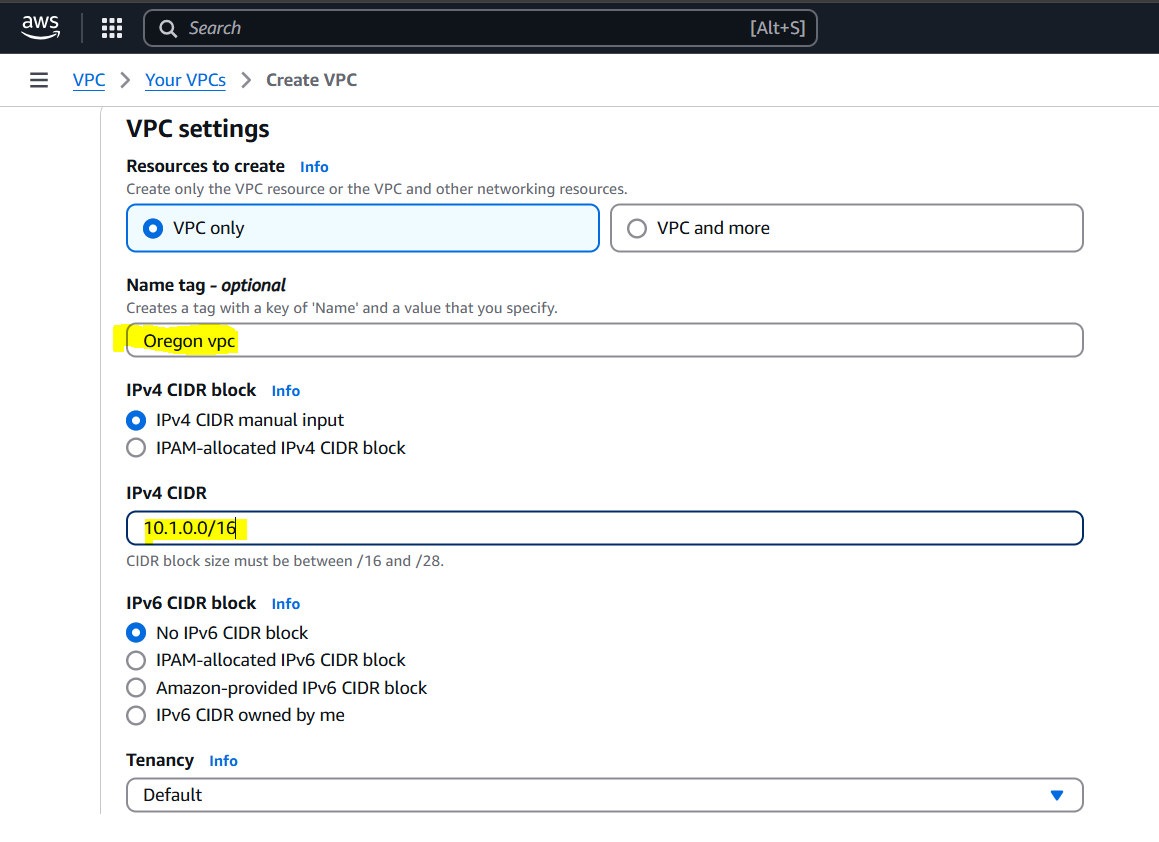
* Design and deploy a scalable network architecture using AWS Transit Gateway to simplify connectivity across VPCs in multiple regions.
* Configure VPC endpoints to securely access AWS services without Internet Gateways or NAT Gateways.

**Step 1: Create 4 VPCs in Different Regions**

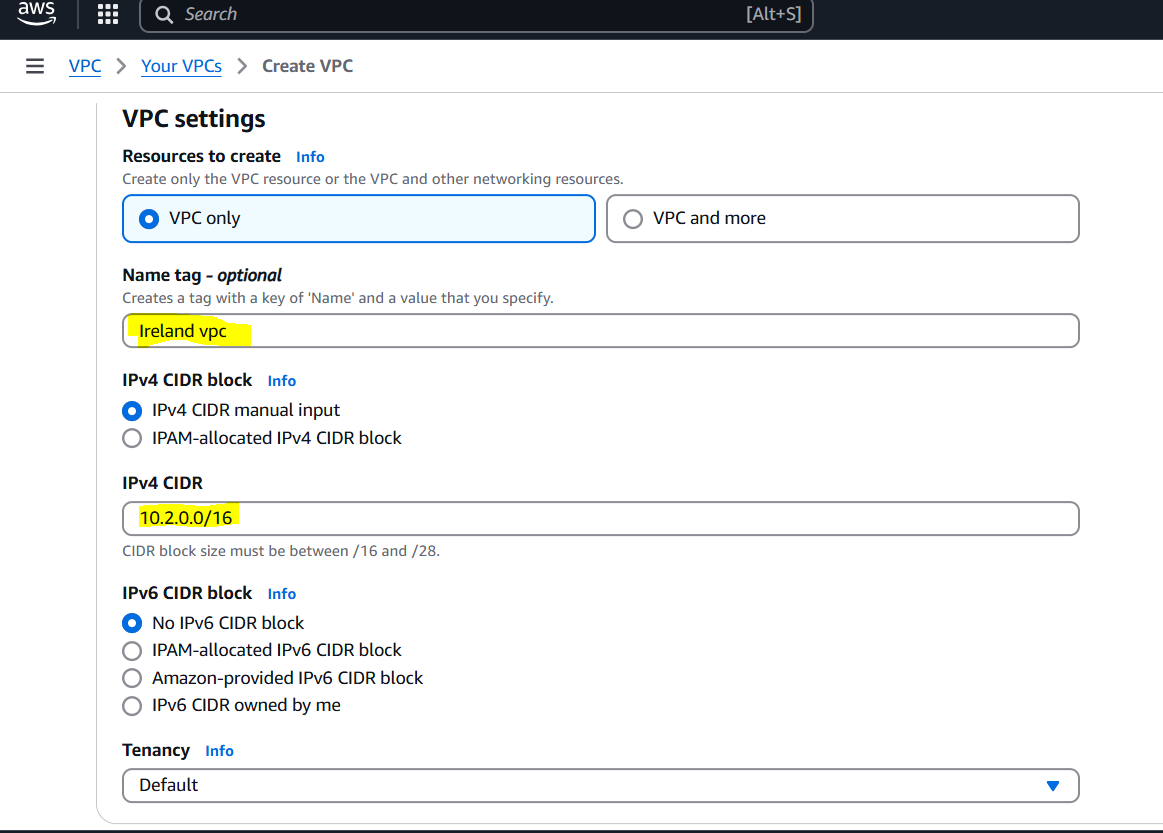
1. Go to the AWS VPC Dashboard and click "Create VPC".
2. Create the following VPCs:
   * VPC-US-East in N. Virginia region with CIDR block 10.0.0.0/16



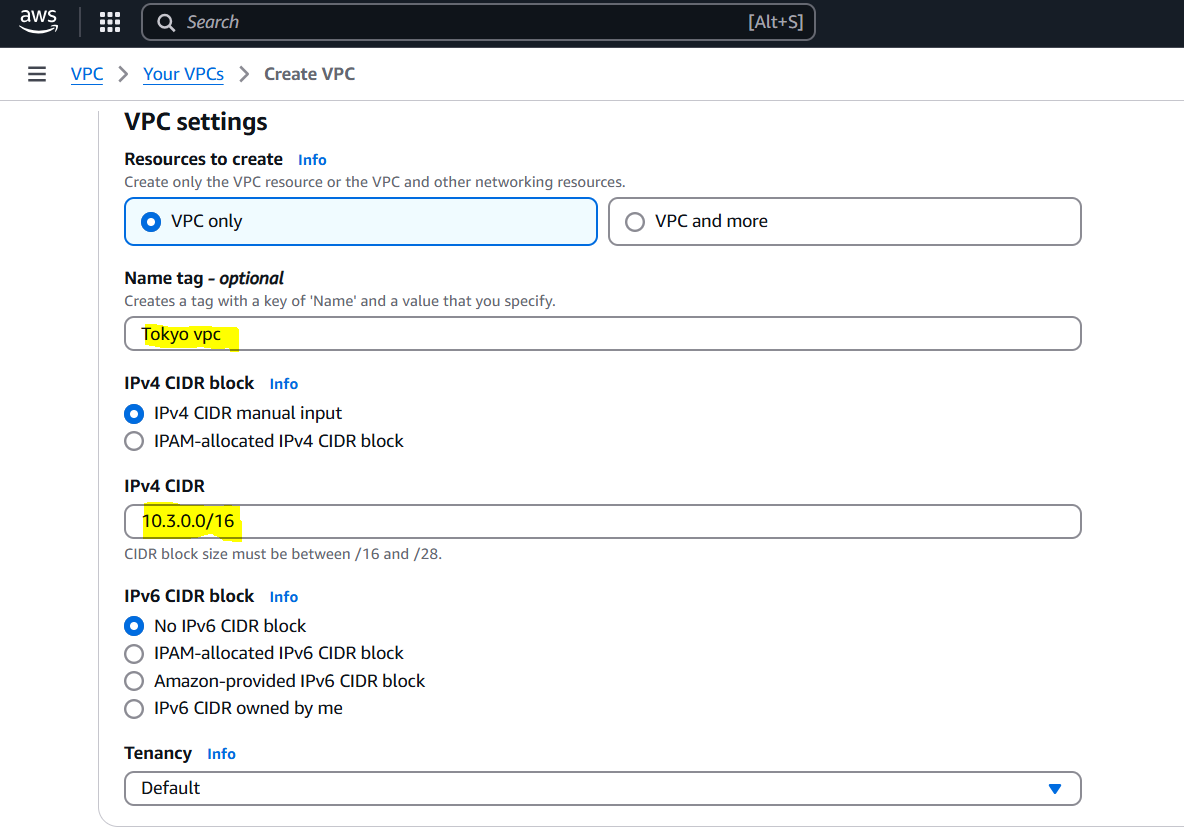
* + VPC-US-West in Oregon region with CIDR block 10.1.0.0/16



* + VPC-EU-Ireland in EU (Ireland) region with CIDR block 10.2.0.0/16



* + VPC-AP-Tokyo in AP (Tokyo) region with CIDR block 10.3.0.0/16



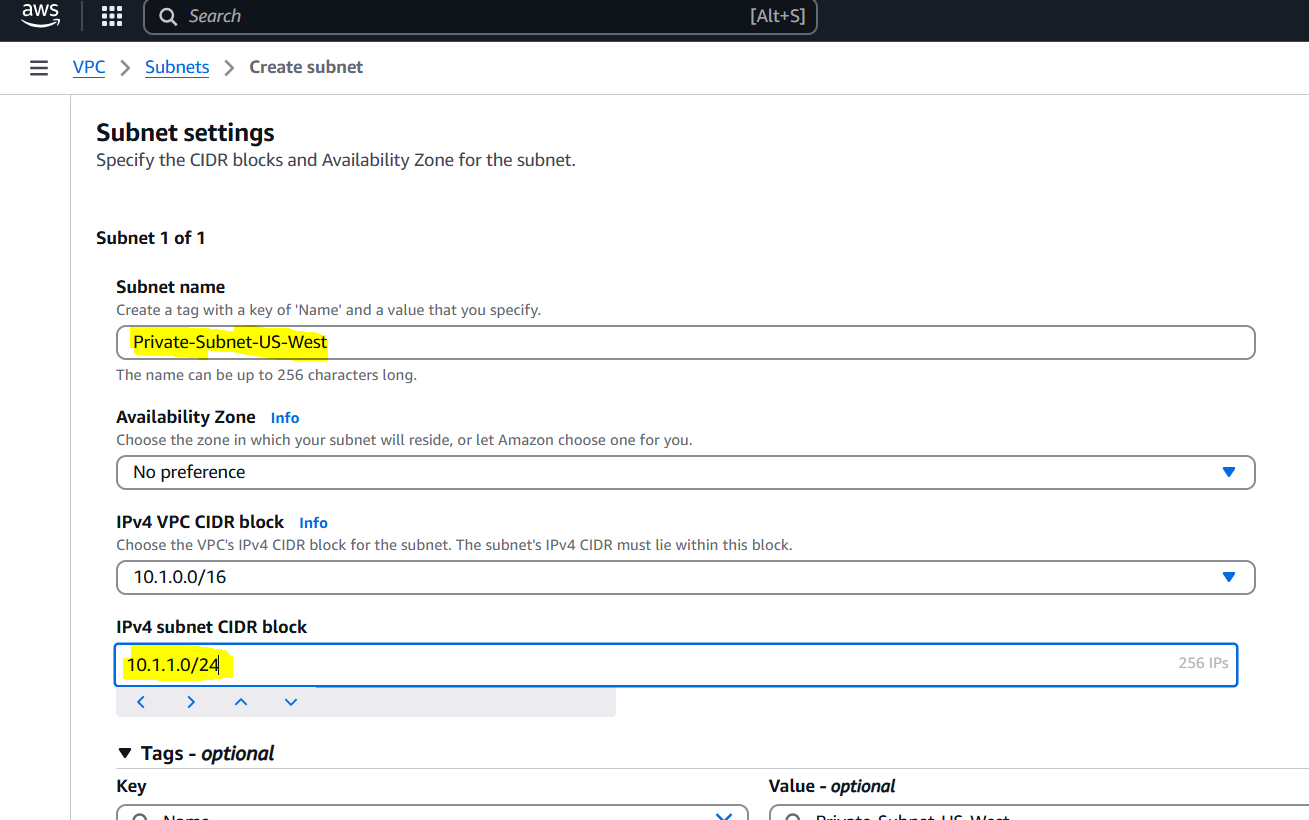
1. Make sure not to use default VPCs.

**Step 2: Create Subnets**

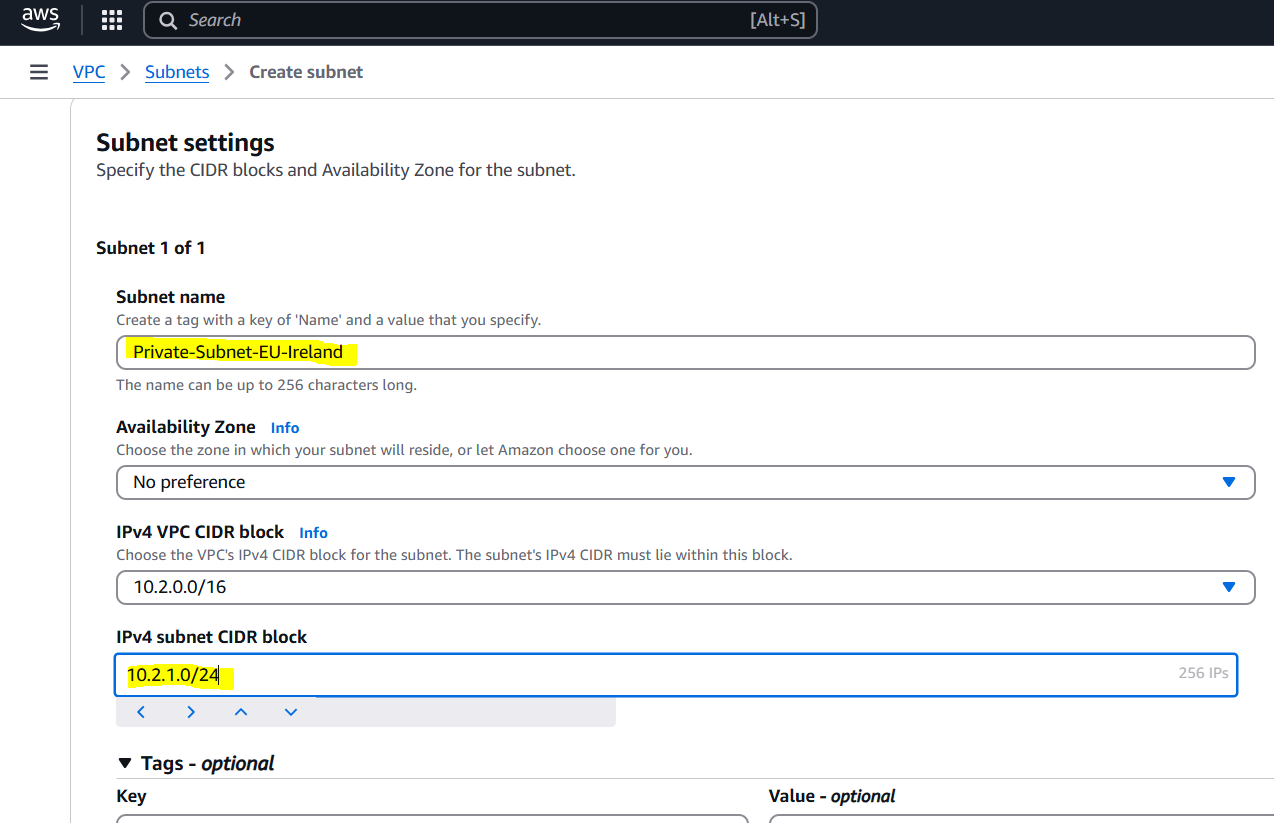
1. In each VPC, create one subnet:
   * VPC-US-East: Create a public subnet named Public-Subnet-US-East with CIDR 10.0.1.0/24



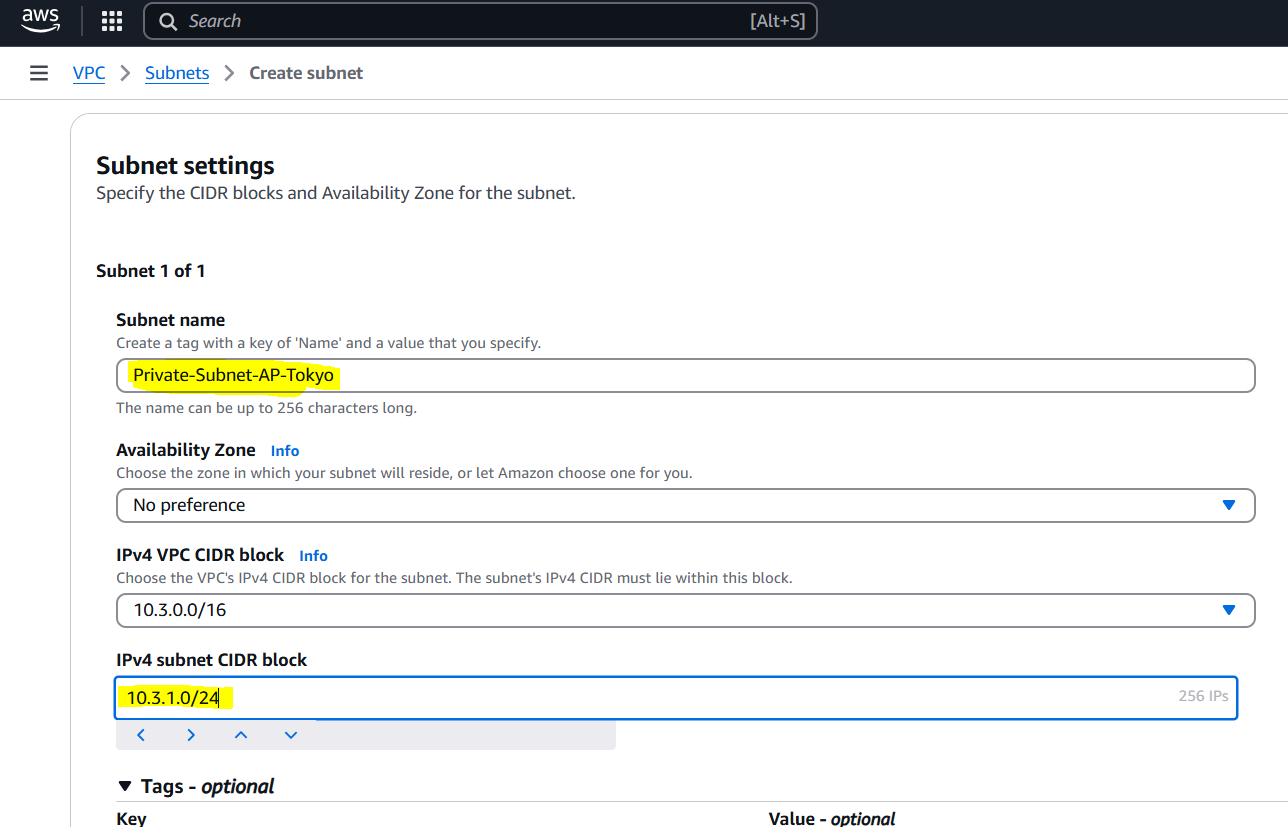
* + VPC-US-West: Create a private subnet named Private-Subnet-US-West with CIDR 10.1.1.0/24



* + VPC-EU-Ireland: Create a private subnet named Private-Subnet-EU-Ireland with CIDR 10.2.1.0/24

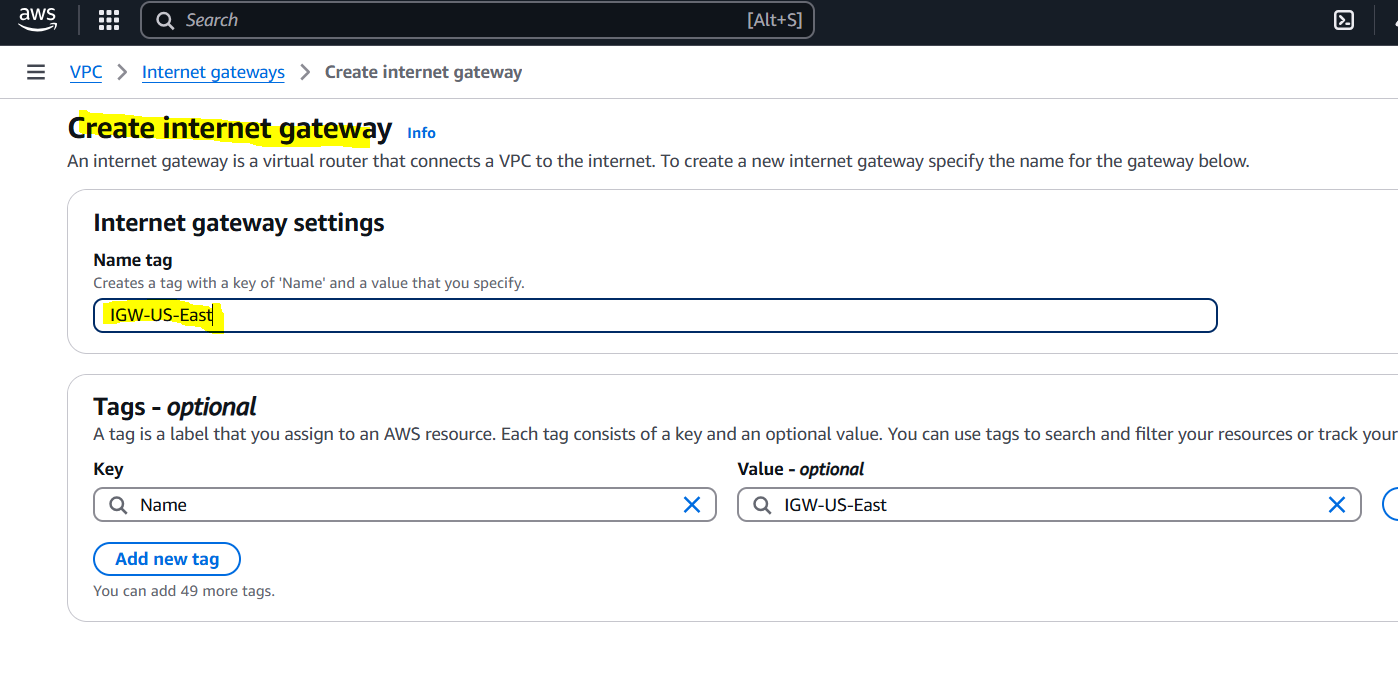


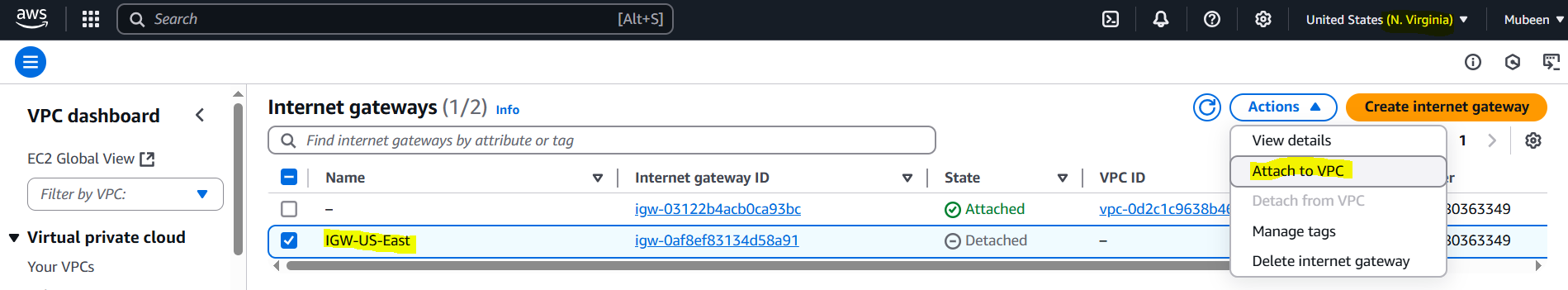
* + VPC-AP-Tokyo: Create a private subnet named Private-Subnet-AP-Tokyo with CIDR 10.3.1.0/24

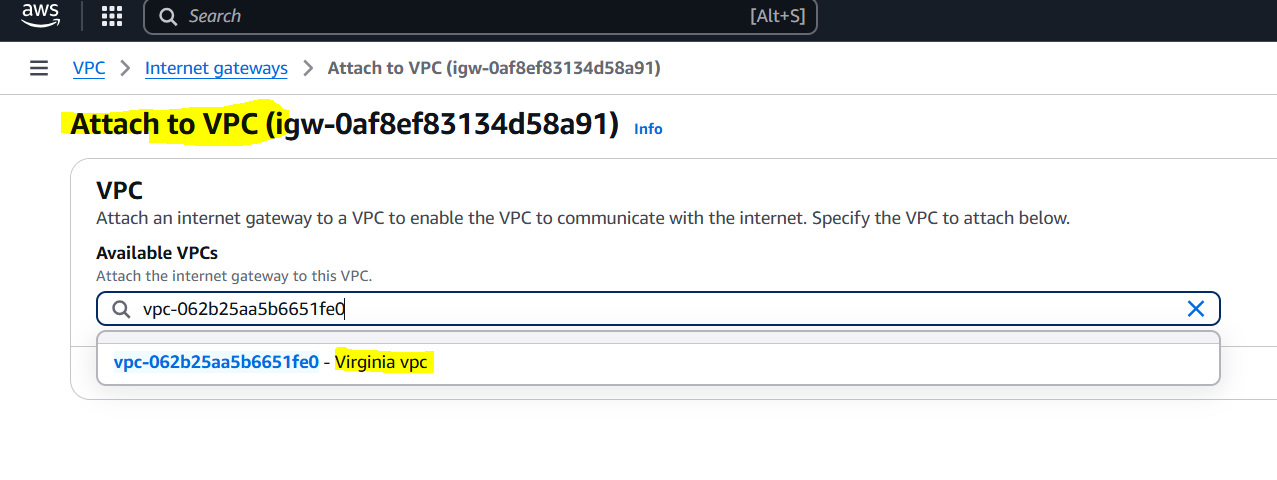


**Step 3: Create and Attach Internet Gateway (IGW)**

1. In the VPC-US-East, go to Internet Gateways and create a new IGW named IGW-US-East.
2. Attach the IGW to VPC-US-East.

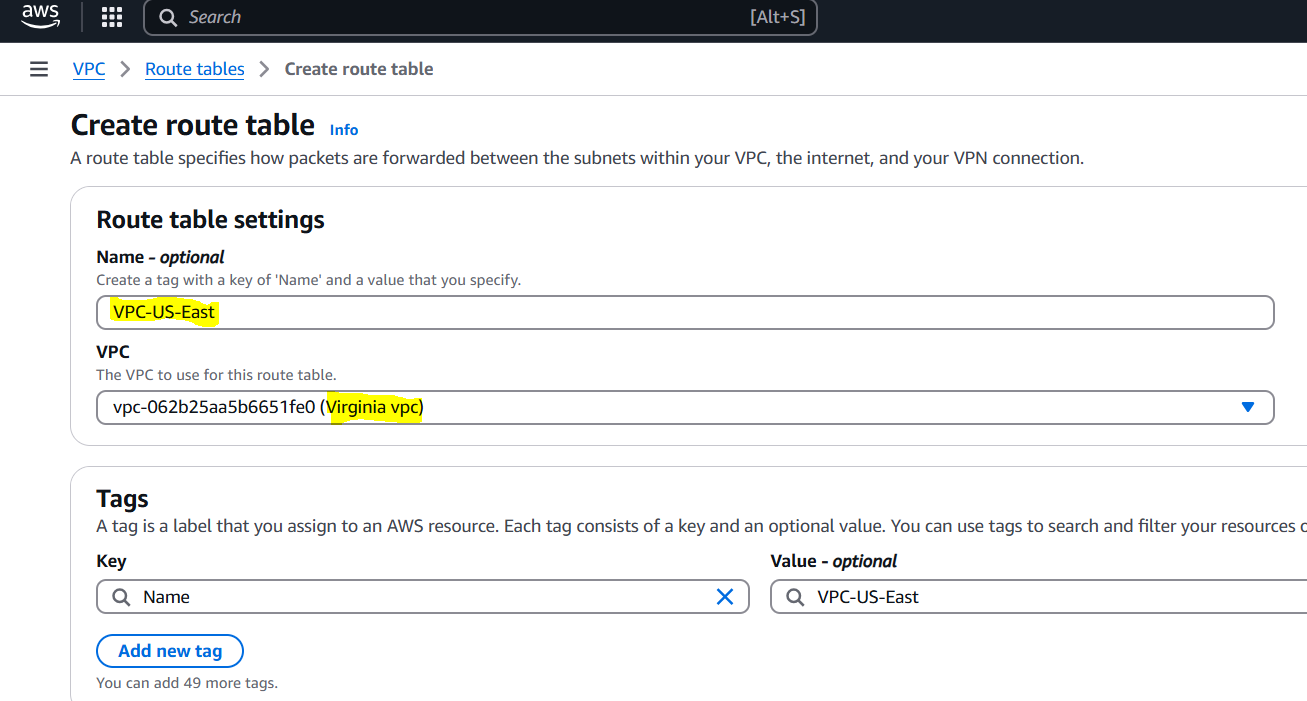


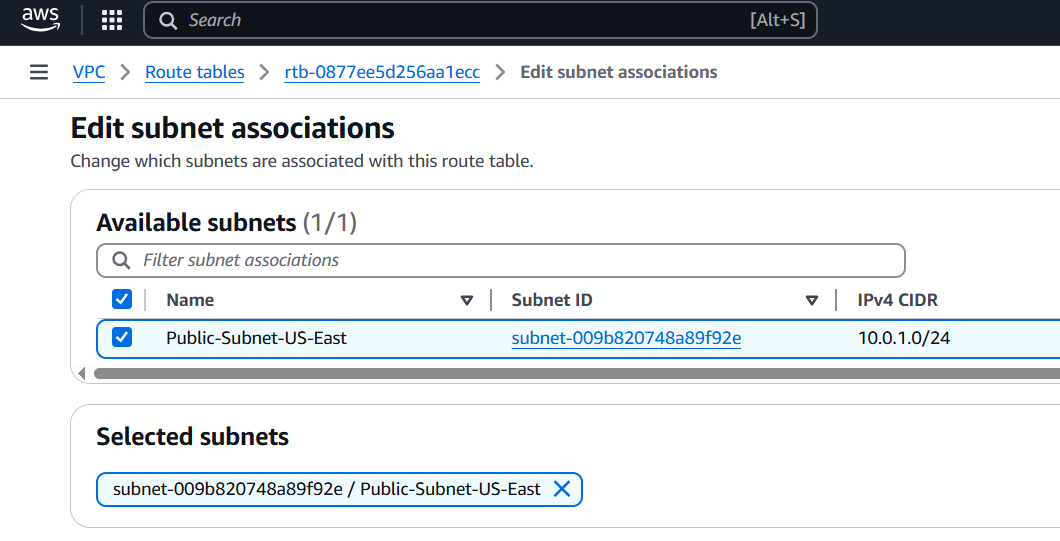


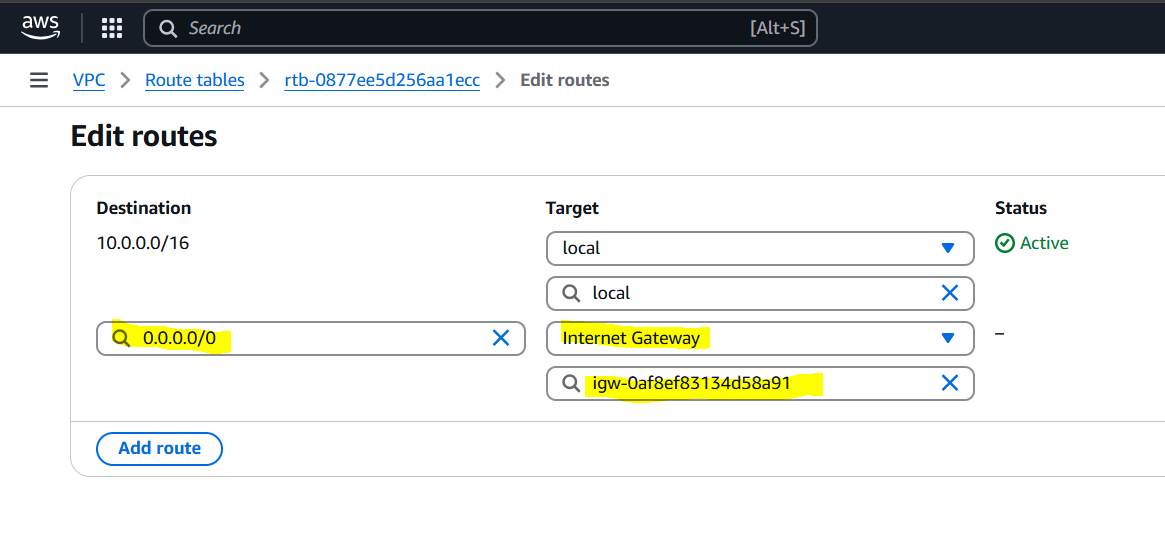


**Step 4: Create Route Tables**

1. Create a public route table for the public subnet in VPC-US-East:
   * Associate it with Public-Subnet-US-East
   * Add route: Destination 0.0.0.0/0 → Target IGW-US-East

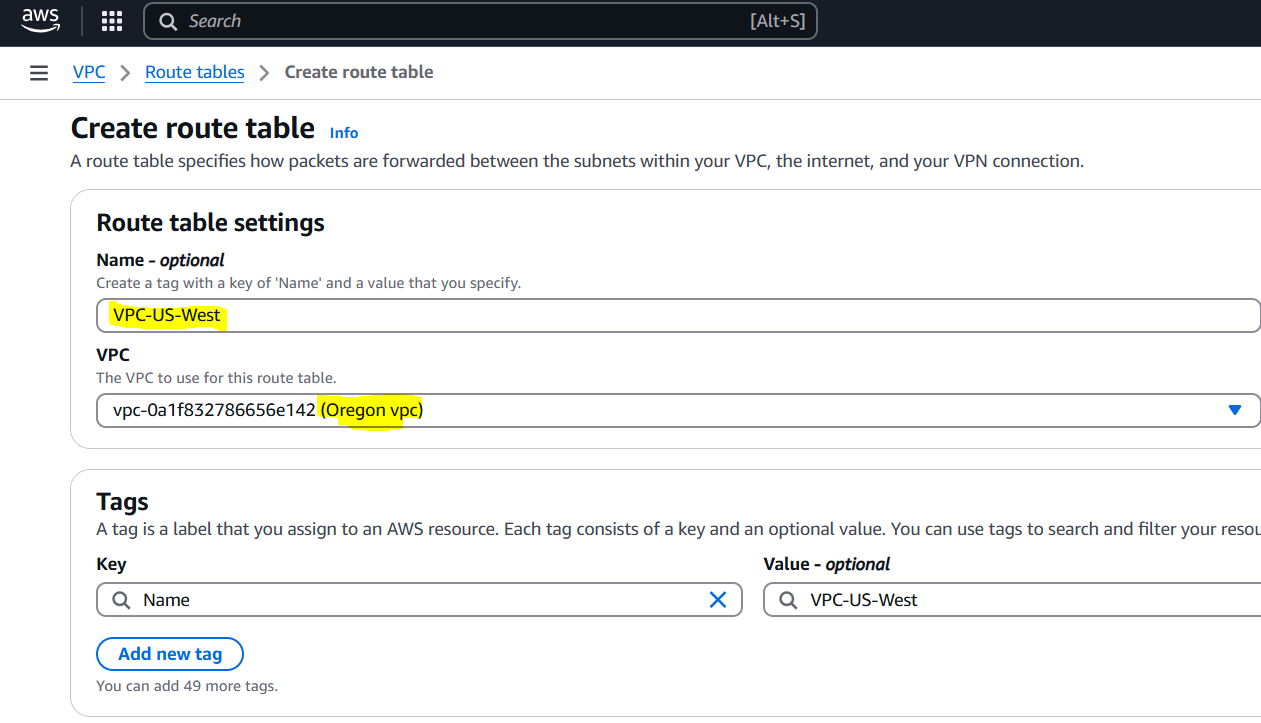


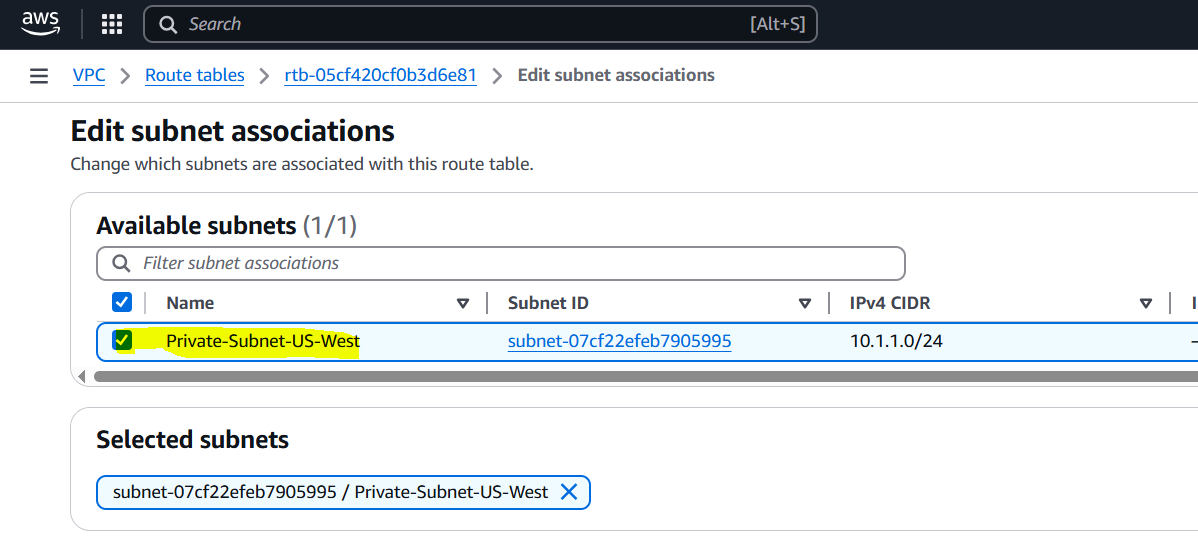




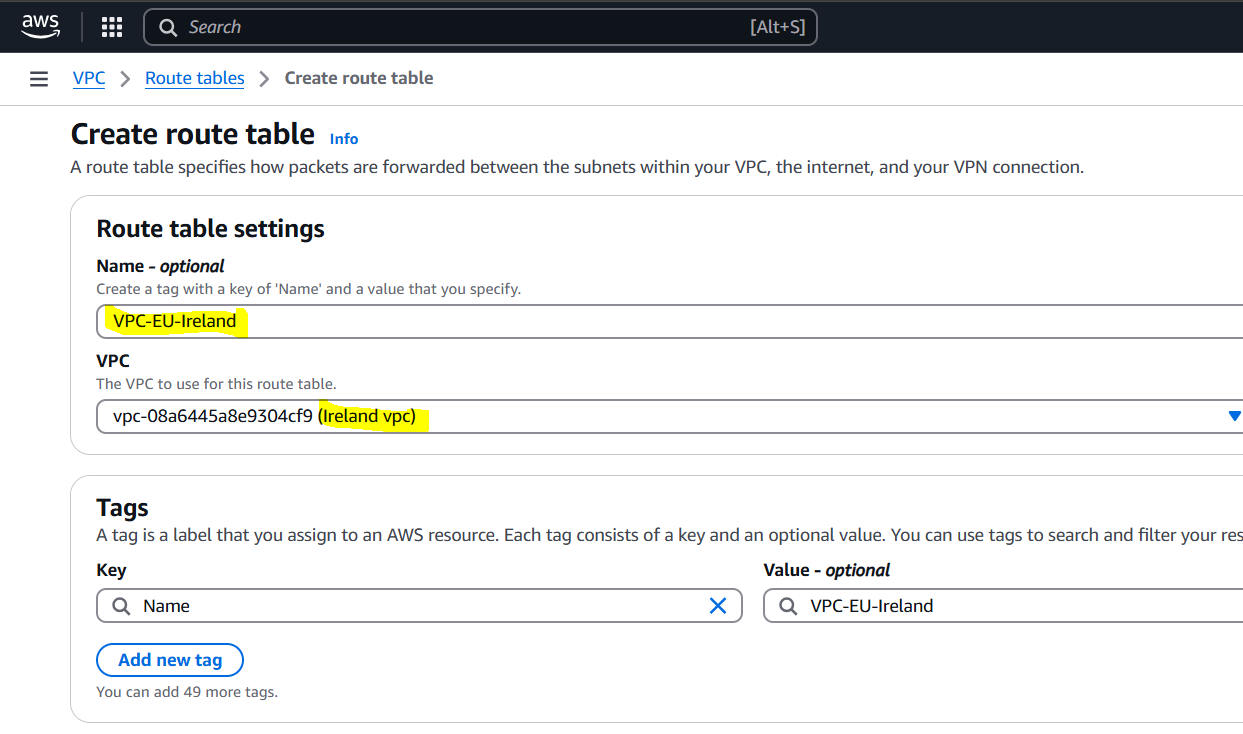
1. For the private subnets in VPC-US-West, VPC-EU-Ireland, and VPC-AP-Tokyo:
   * Create a route table for each
   * Associate each route table with the respective private subnet
   * Leave the route table default for now (we will update with TGW later)

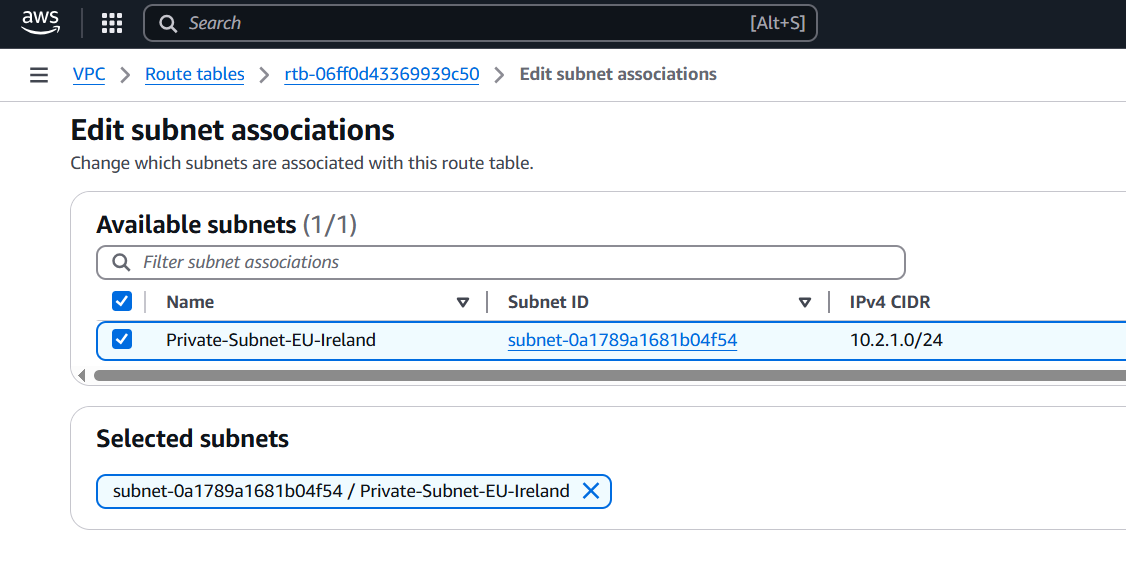
**VPC-EU-WEST**



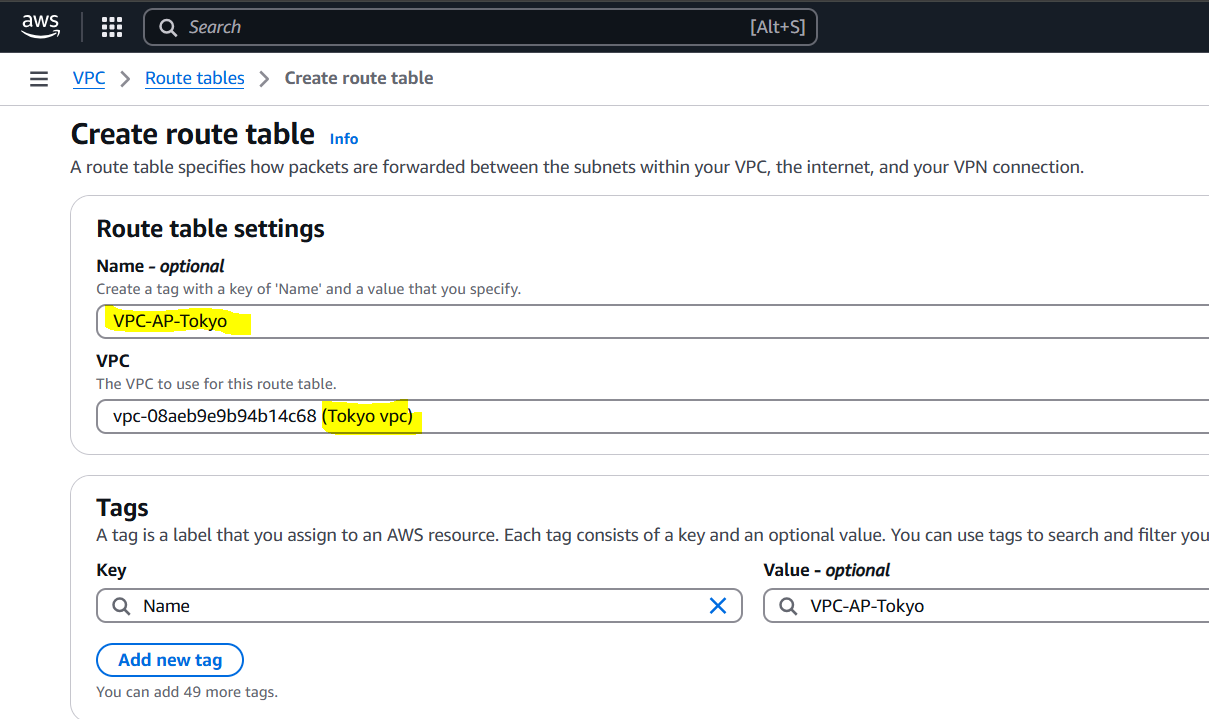


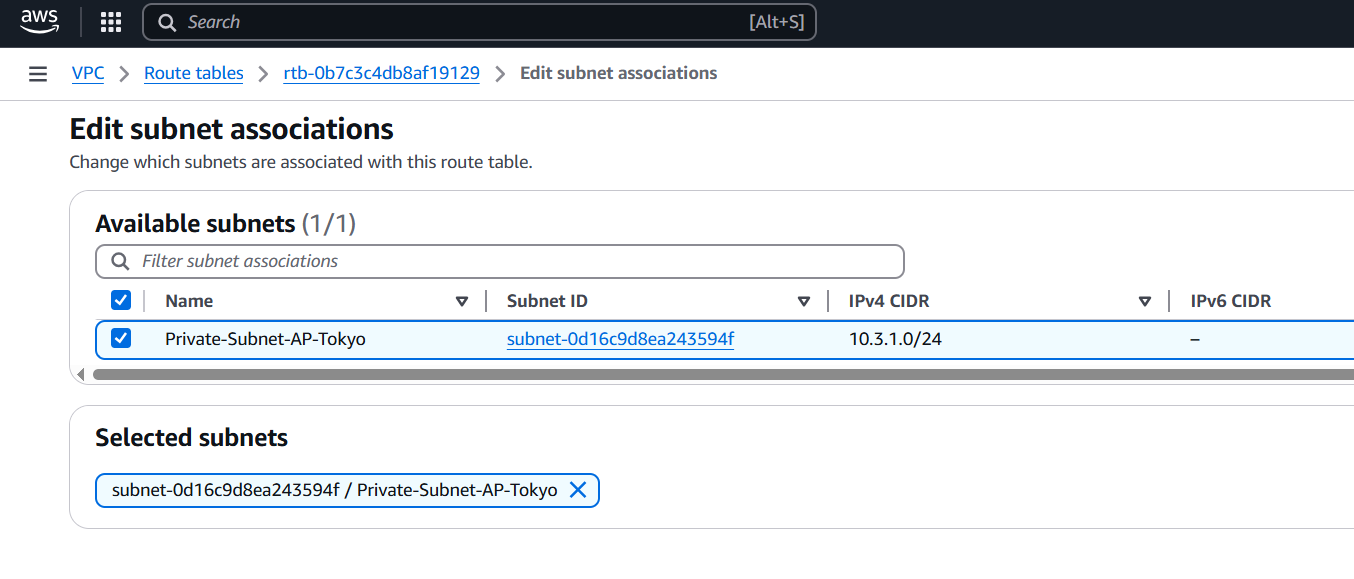
**VPC-EU-Ireland**





**VPC-AP-Tokyo**

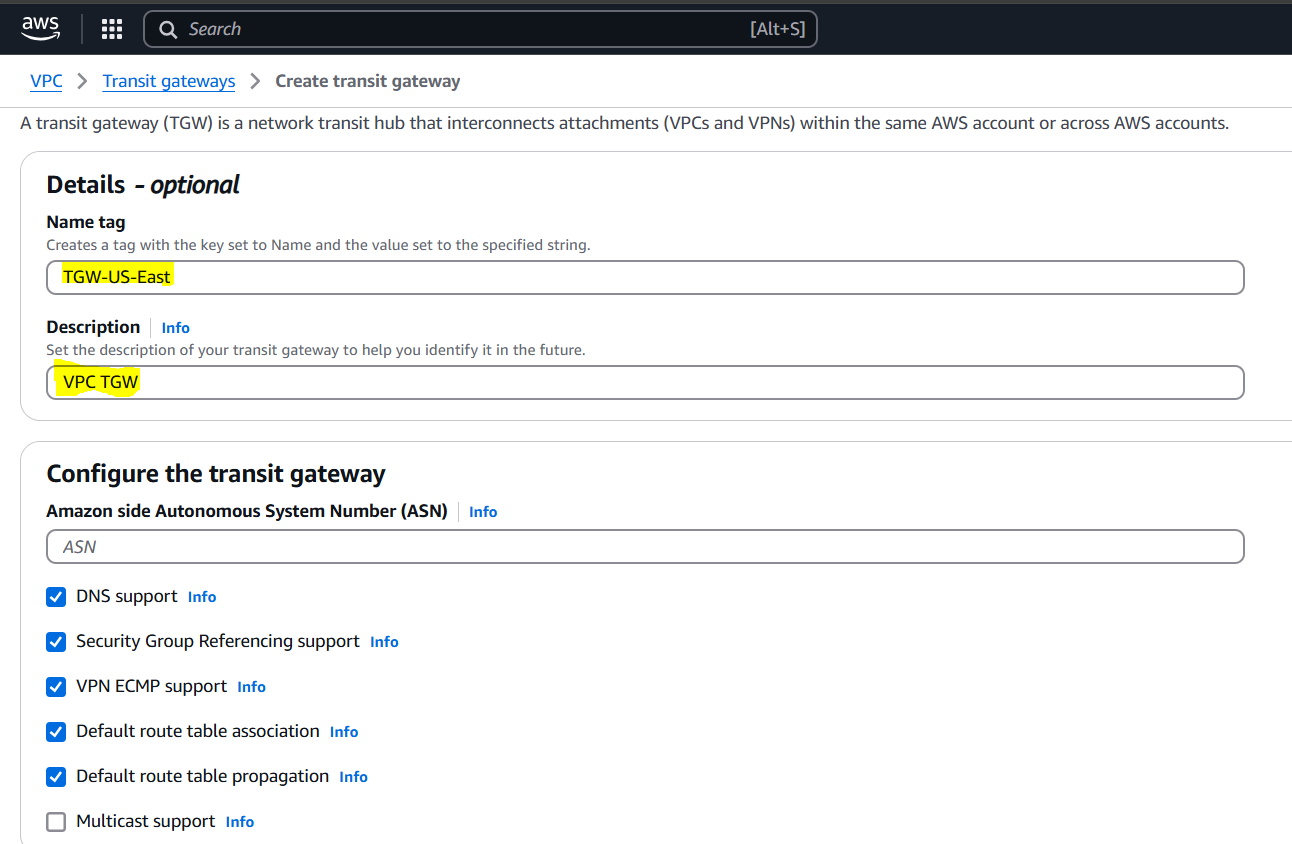




**Step 5: Create Transit Gateways in Each Region**

1. In each region, go to Transit Gateway and click "Create Transit Gateway".
2. Name them appropriately, TGW-US-East, TGW-US-West, TGW-EU-Ireland, TGW-AP-Tokyo.
3. Use default settings and allow default route table creation.

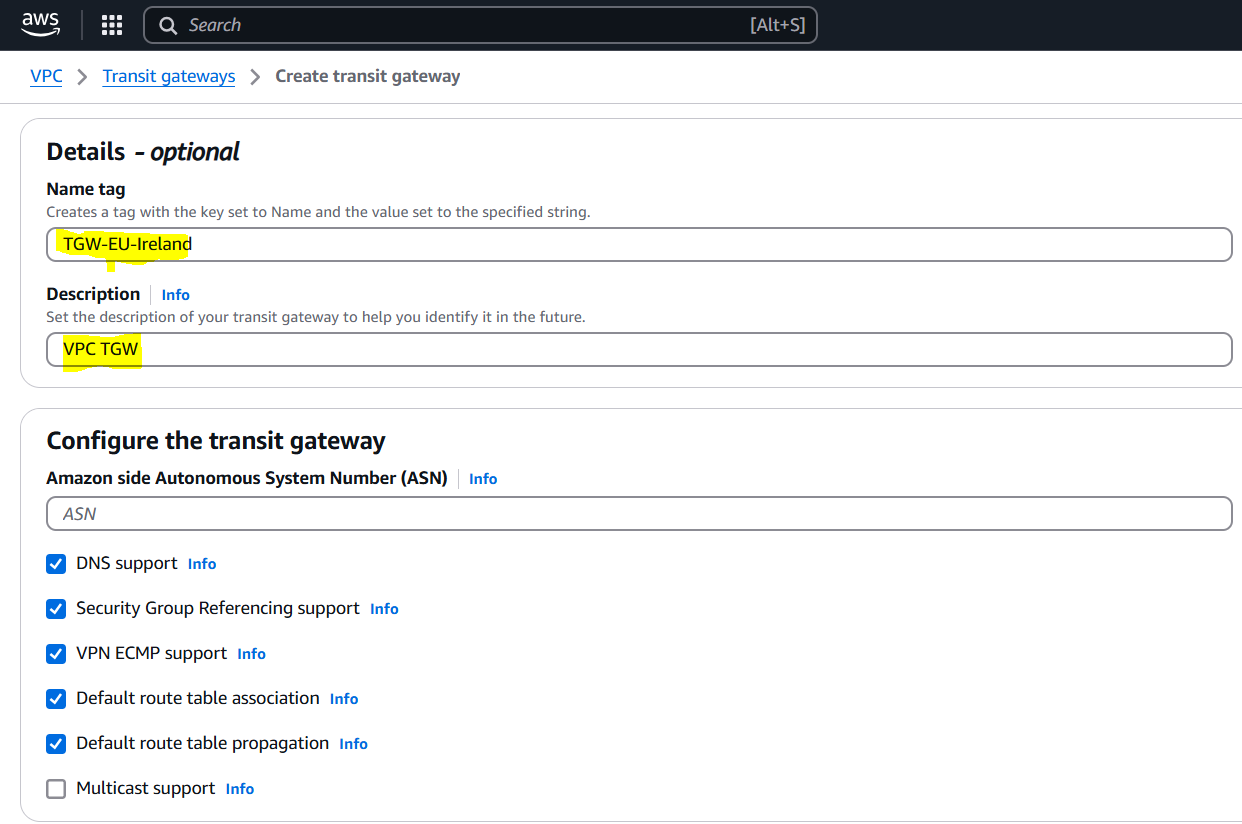
**TGW-US-East**



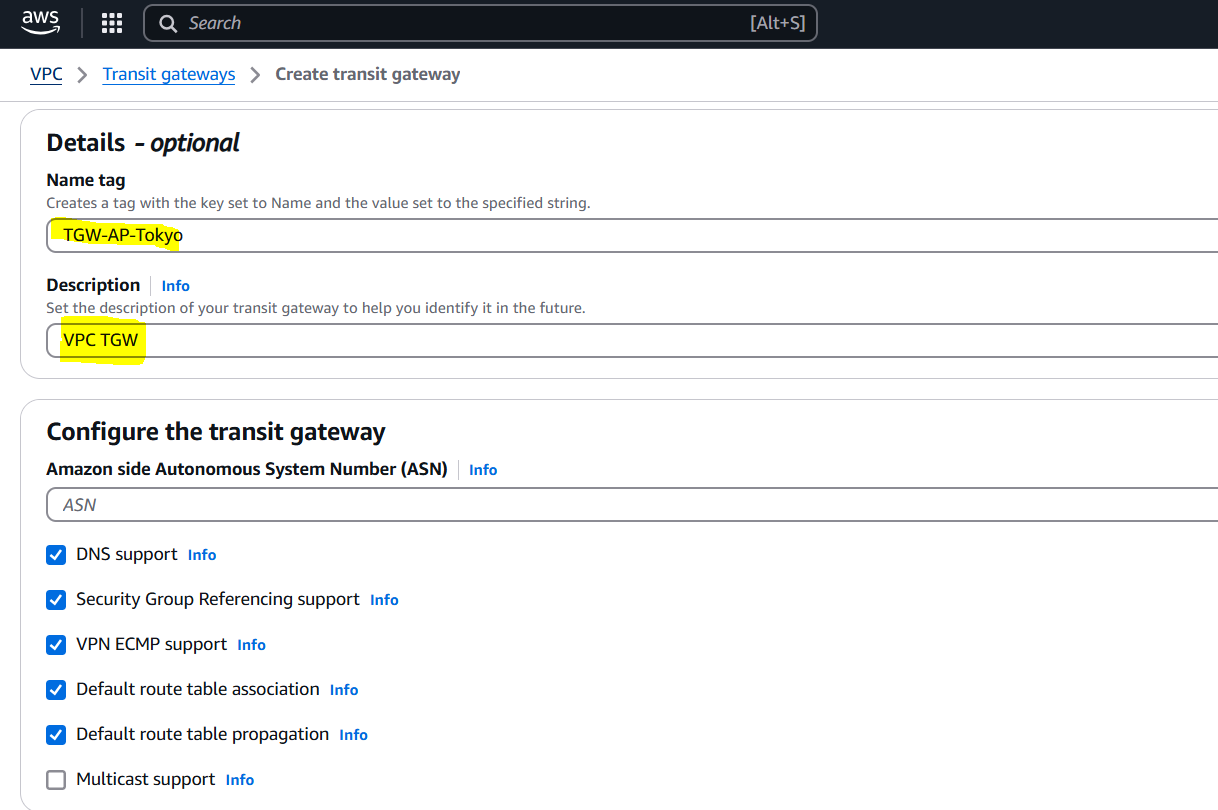
**TGW-US-West**



**TGW-EU-Ireland**

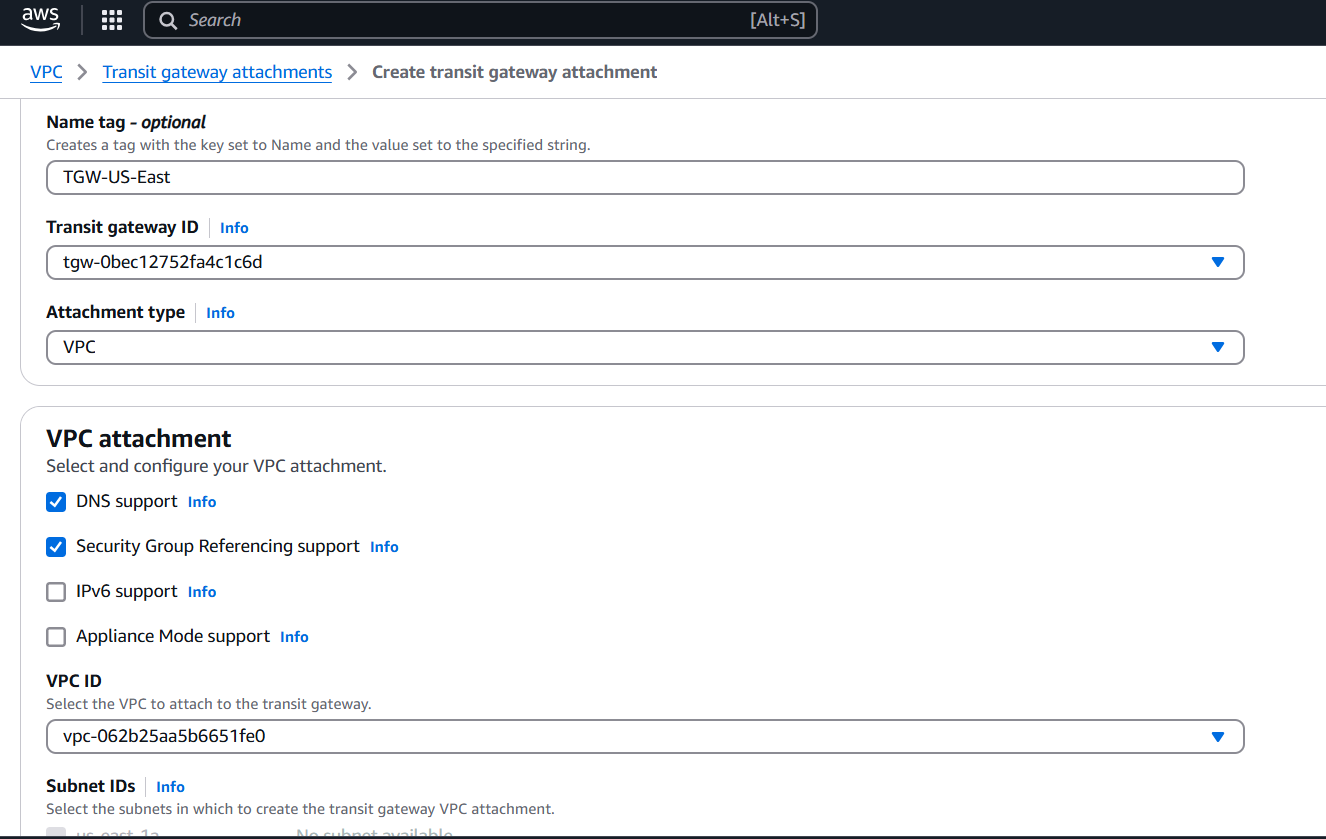


**TGW-AP-Tokyo**

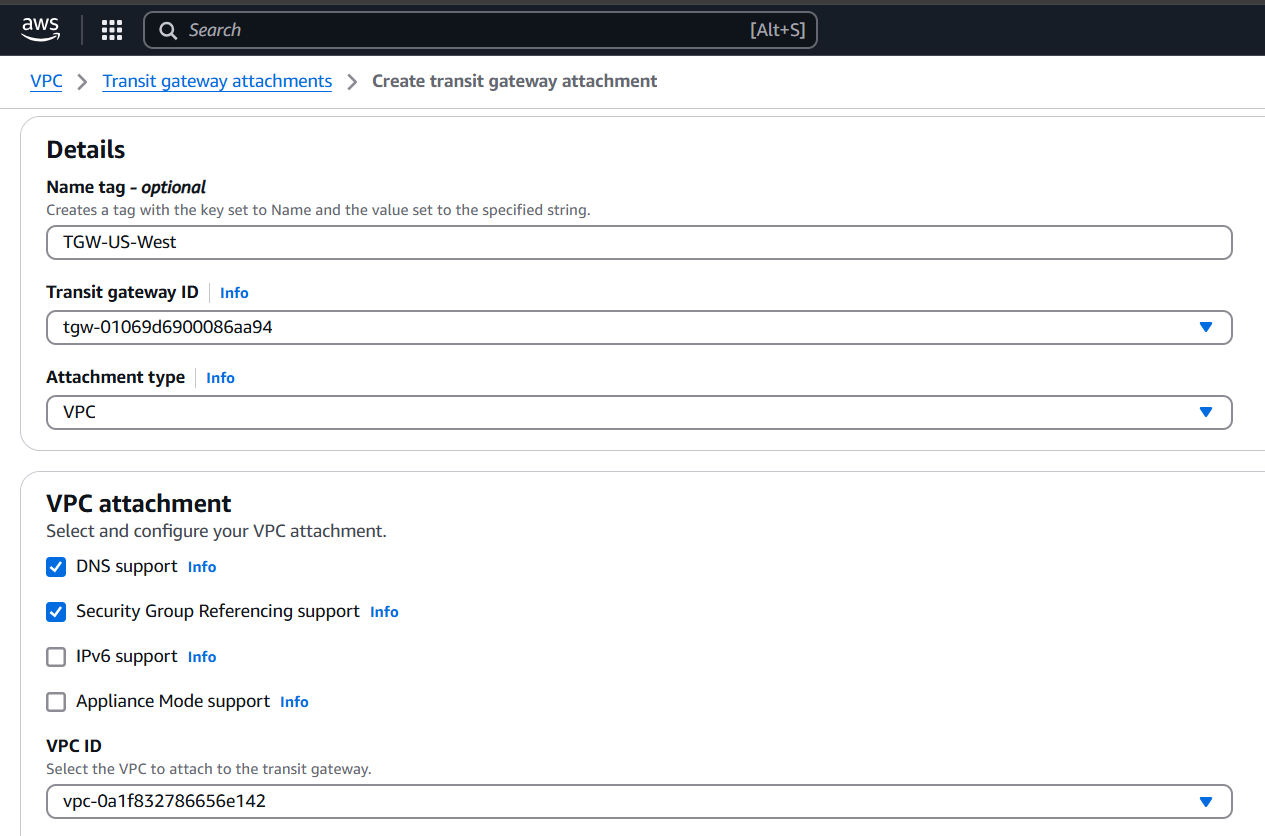


**Step 6: Create Transit Gateway Attachments**

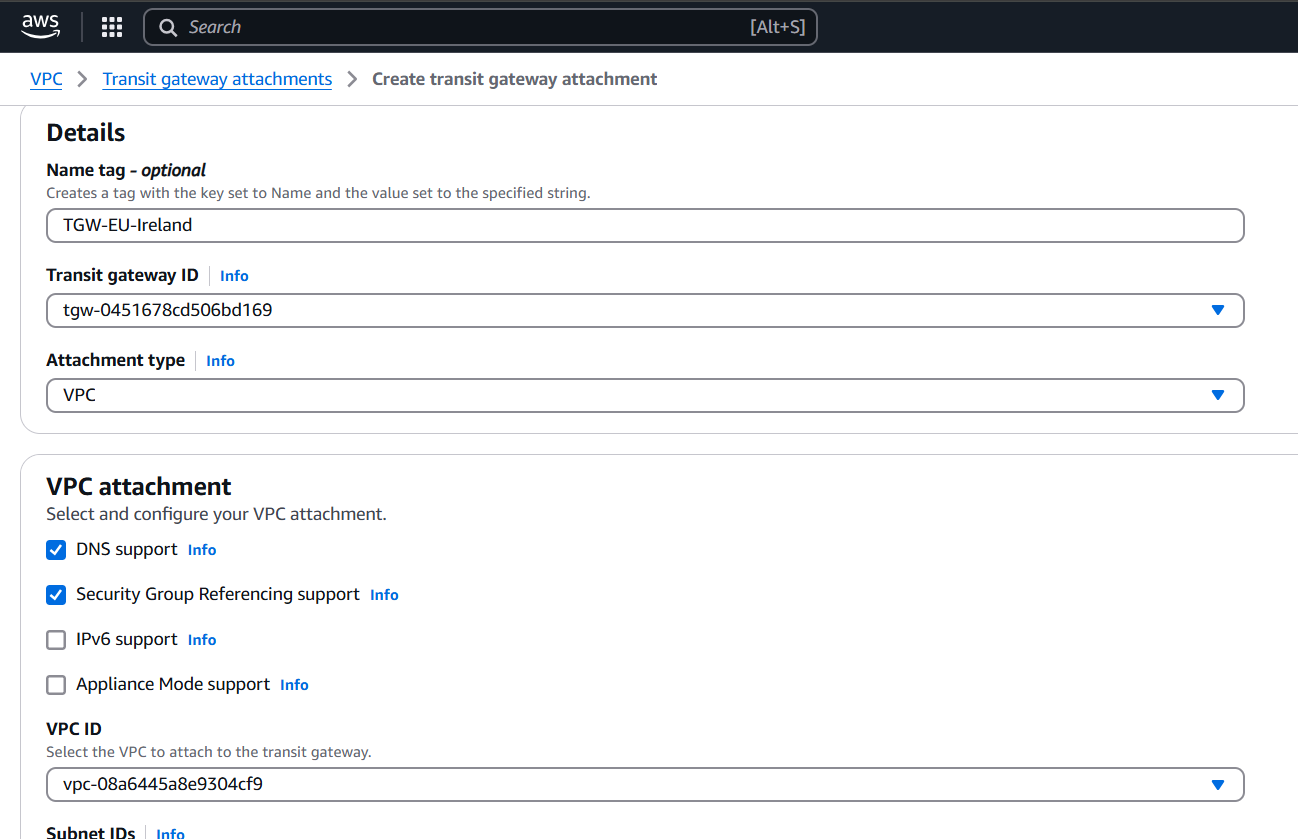
1. For each VPC, go to Transit Gateway → Attachments → Create Attachment.
2. Select the TGW in the region and attach the corresponding VPC and subnet.
3. Example:
   * Attach VPC-US-East to TGW-US-East using Public-Subnet-US-East



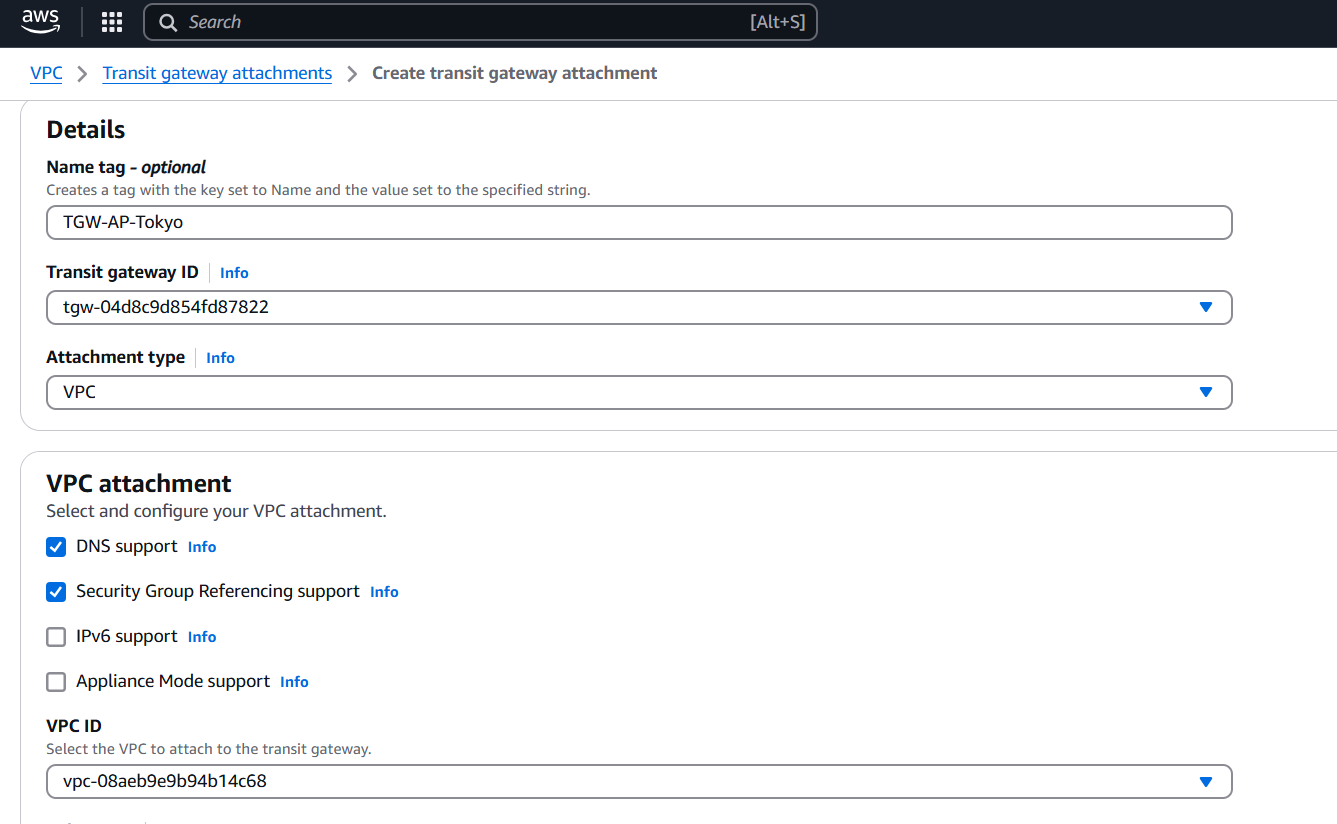
* + Attach VPC-US-West to TGW-US-West using Private-Subnet-US-West



* + Attach VPC-EU-Ireland to TGW-EU-Ireland using Private-Subnet-EU-Ireland

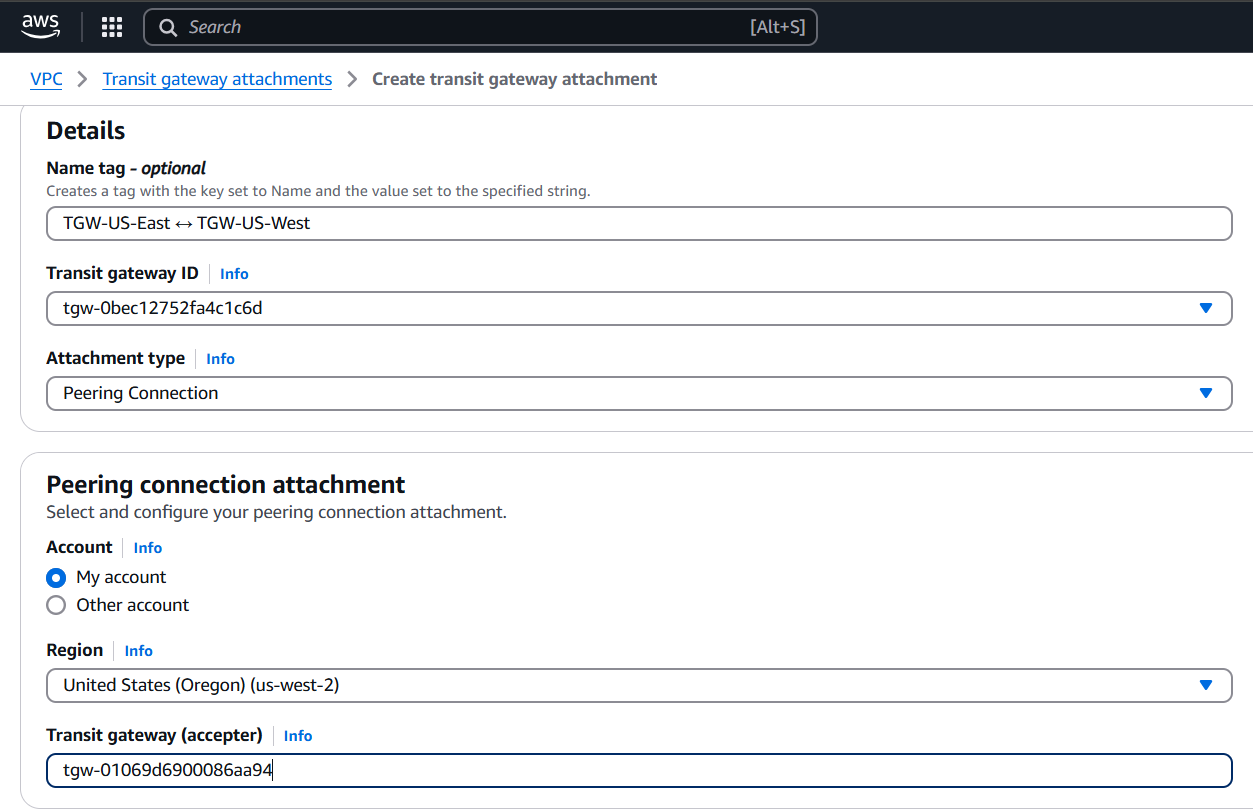


* + Attach VPC-AP-Tokyo to TGW-AP-Tokyo using Private-Subnet-AP-Tokyo

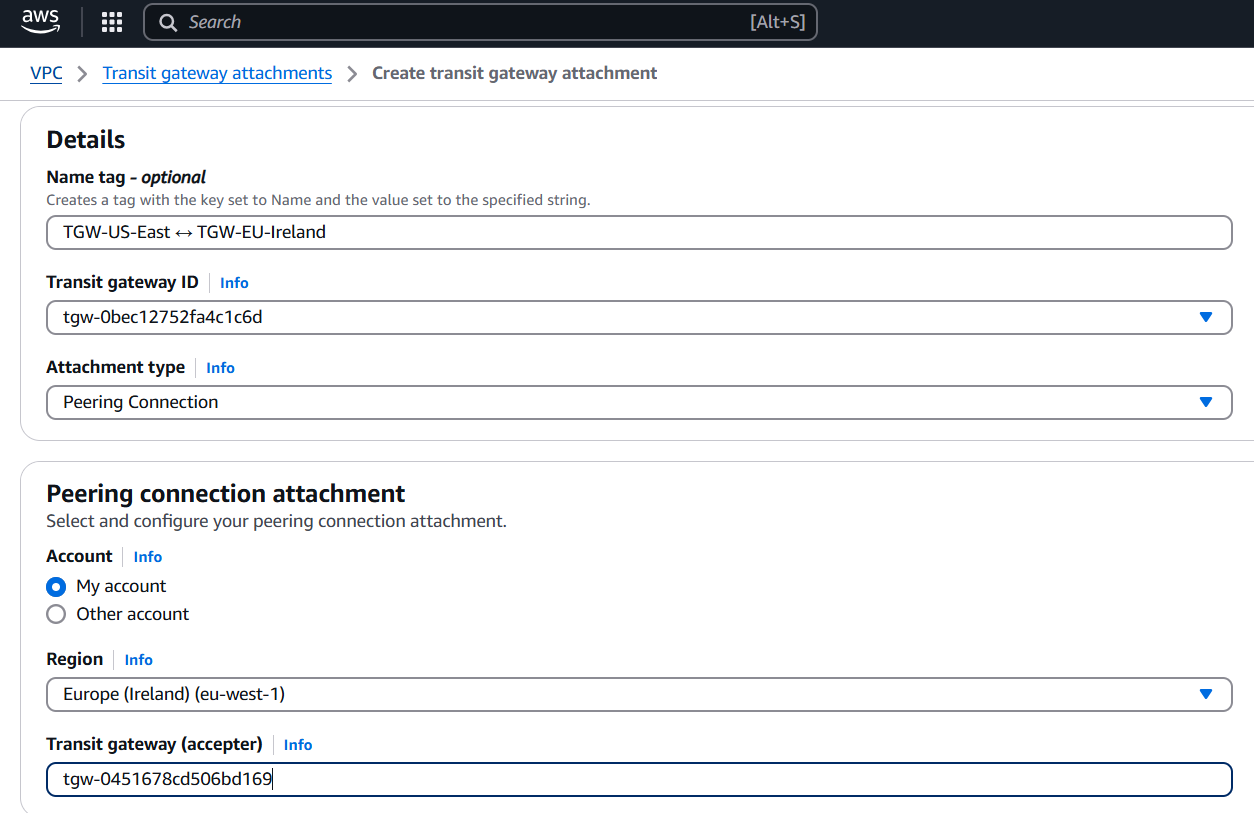


**Step 7: Create Transit Gateway Peering**

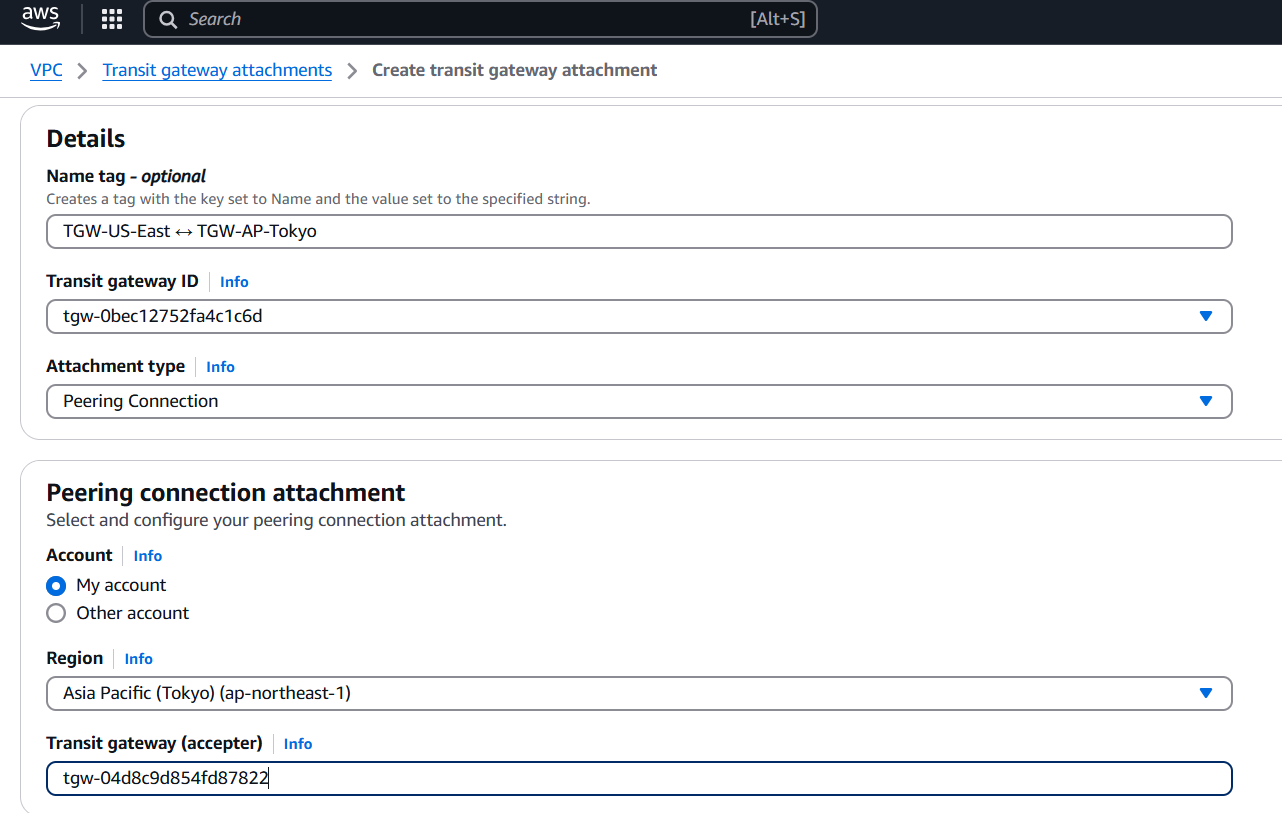
1. In one region, go to Transit Gateway → Peering Attachments → Create Peering.
2. Create peering between:
   * TGW-US-East ↔ TGW-US-West



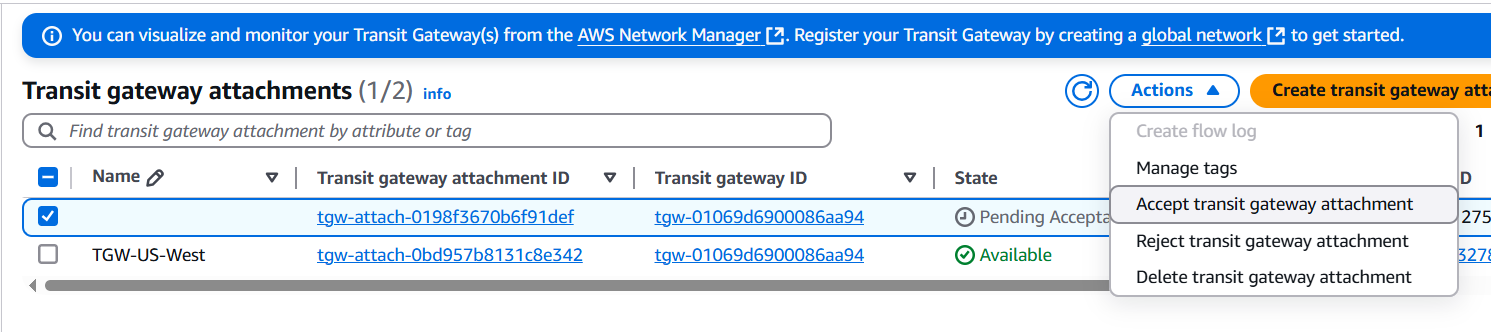
* + TGW-US-East ↔ TGW-EU-Ireland

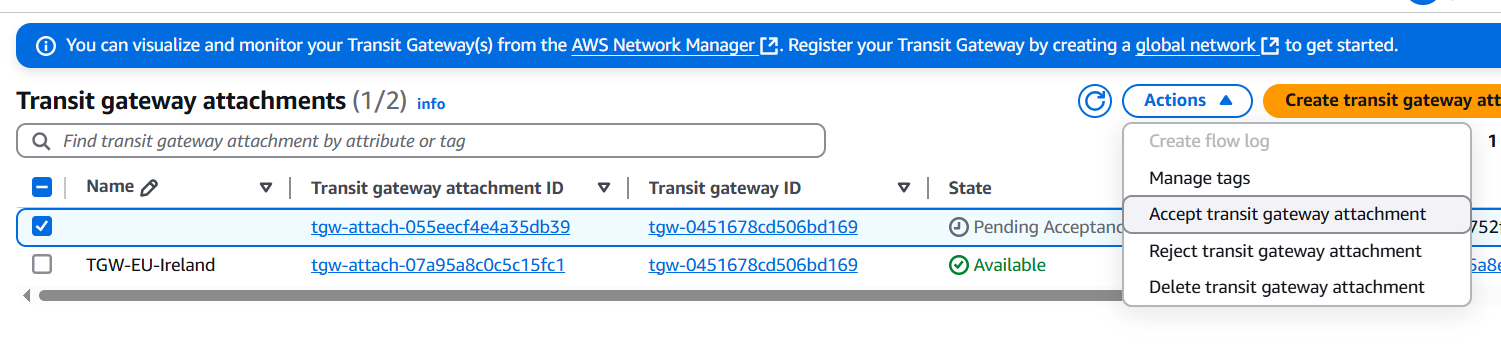


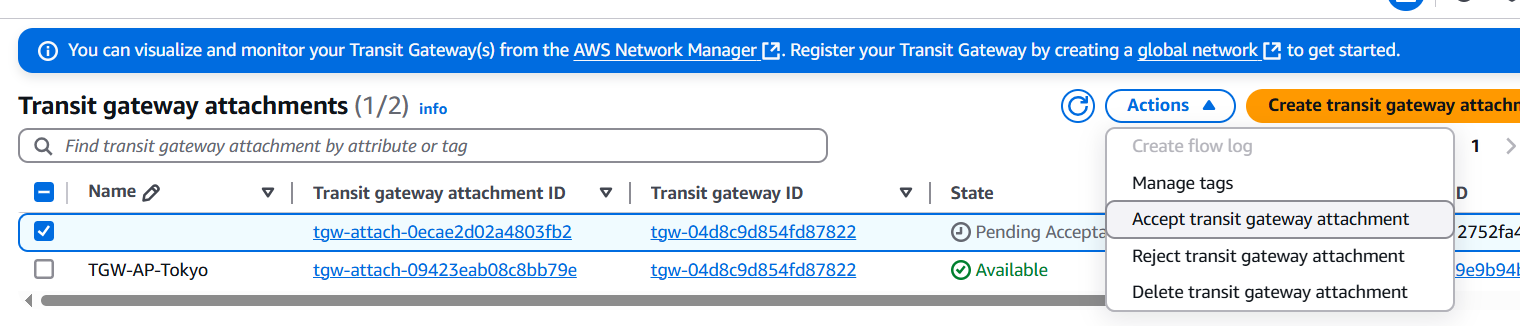
* + TGW-US-East ↔ TGW-AP-Tokyo



1. Accept the peering attachments in the other regions.

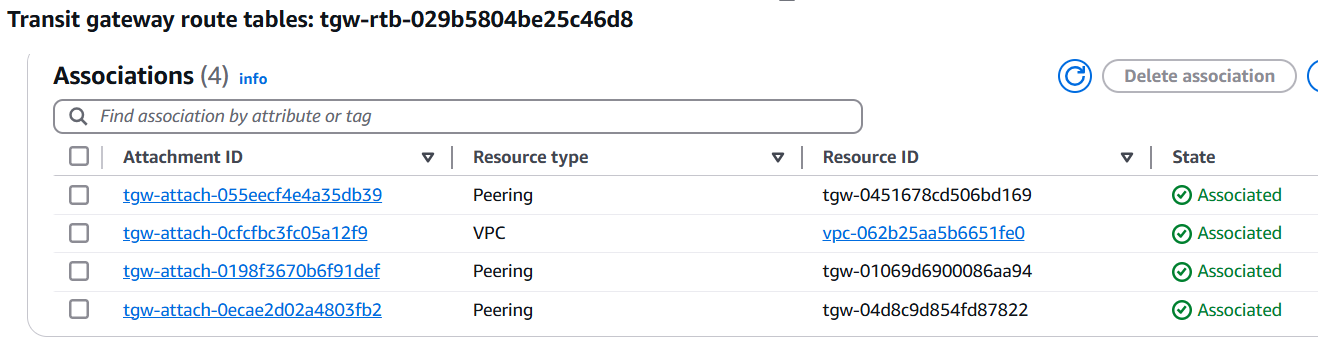


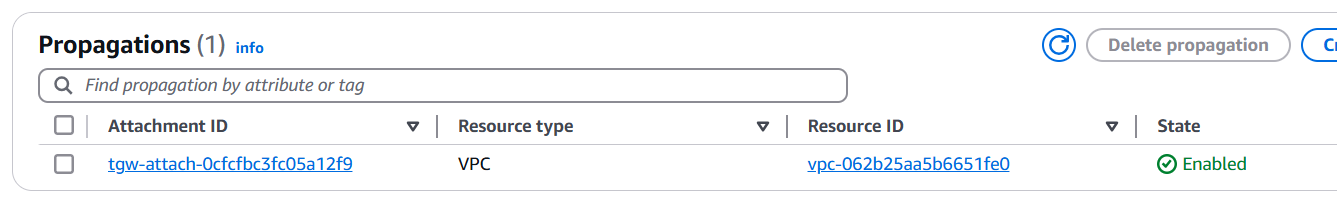


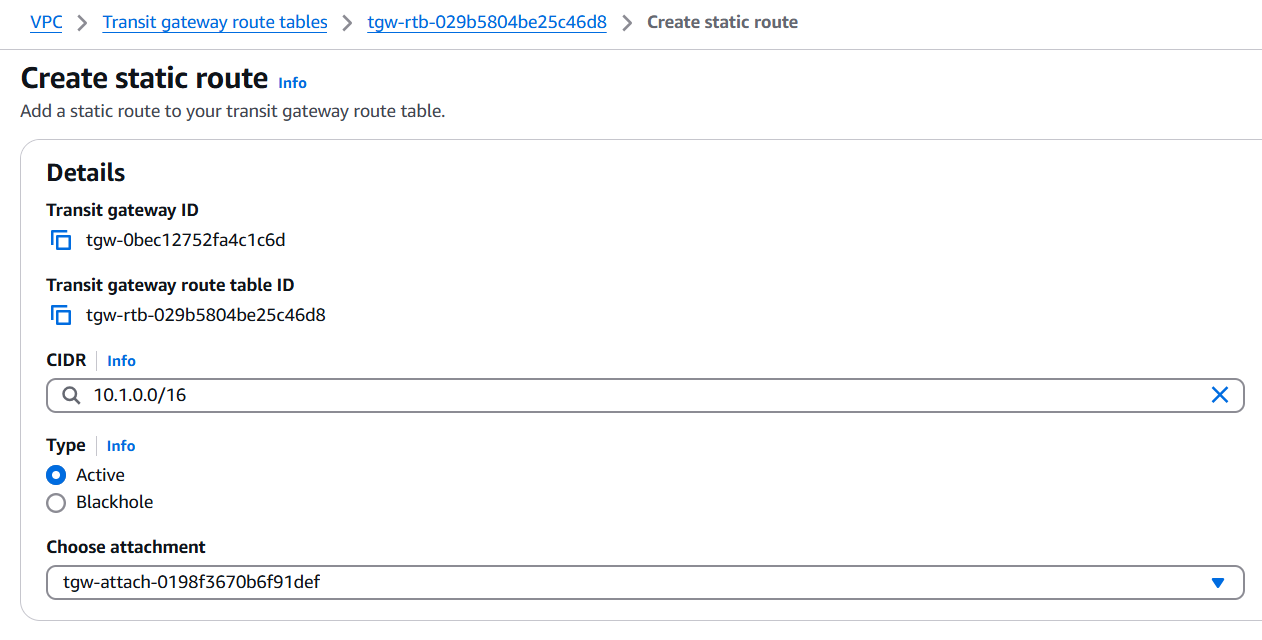


**Step 8: Update Transit Gateway Route Tables**

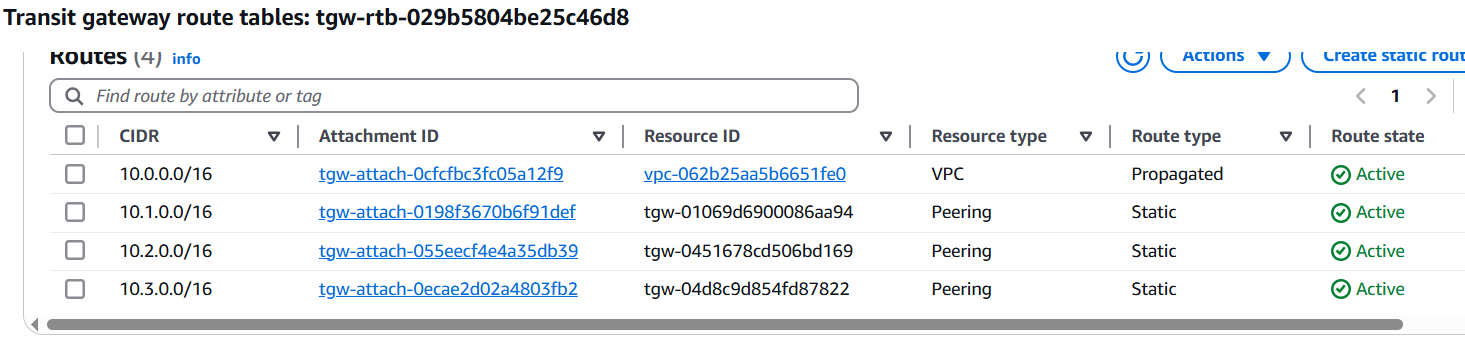
1. In each TGW, go to Route Tables.
2. Associate each VPC attachment to the TGW route table.
3. Enable route propagation for all VPC attachments.
4. Add static routes to remote VPCs via peering attachments.
   * For example, in TGW-US-East:
     + Add route to 10.1.0.0/16 via TGW-US-West peering
     + Add route to 10.2.0.0/16 via TGW-EU-Ireland peering
     + Add route to 10.3.0.0/16 via TGW-AP-Tokyo peering





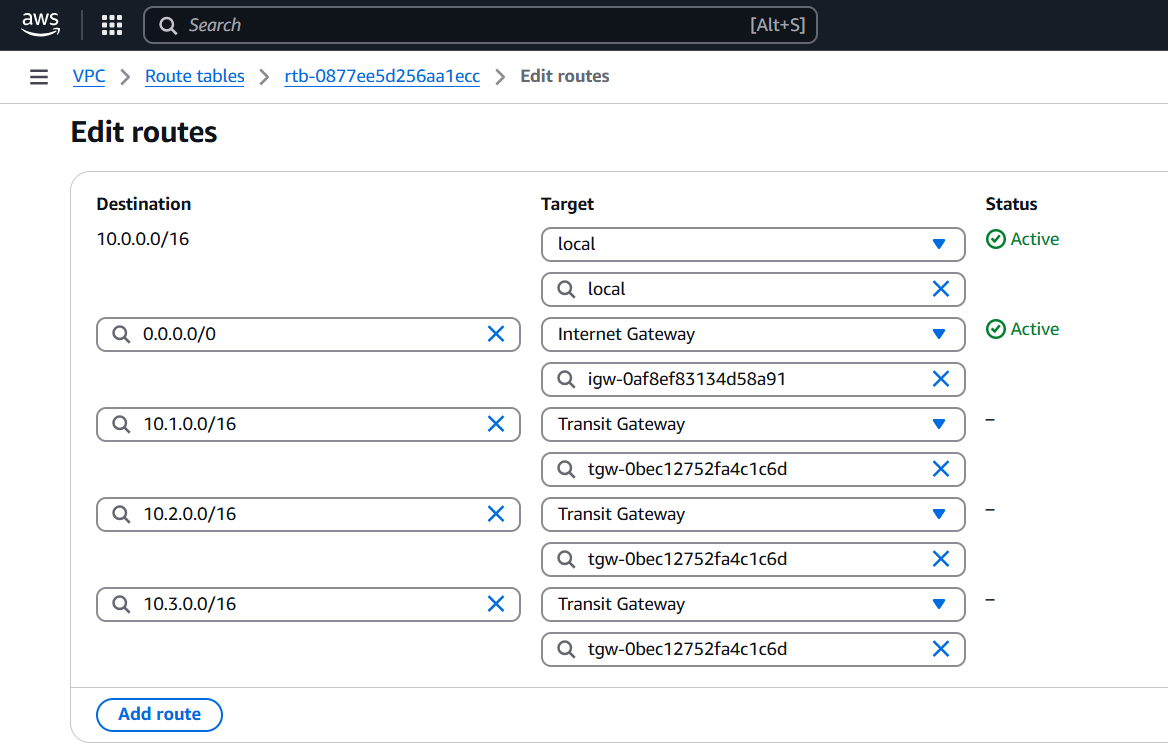


Similarly, we can add all and to other regions also.



**Step 9: Update Route Tables for Private Subnets**

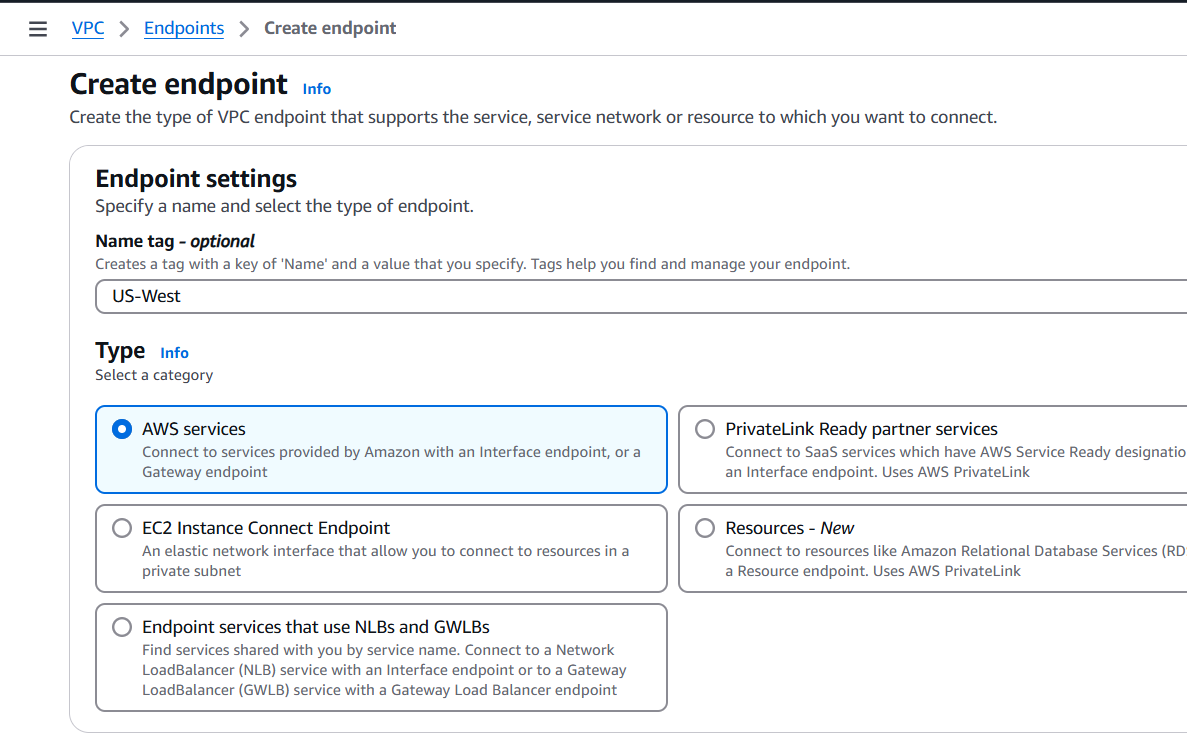
1. In the private route table of each region (US-West, EU-Ireland, AP-Tokyo), add the following routes:
   * `` → Transit Gateway (for US-East)
   * Add routes to other VPCs as needed (based on peering)

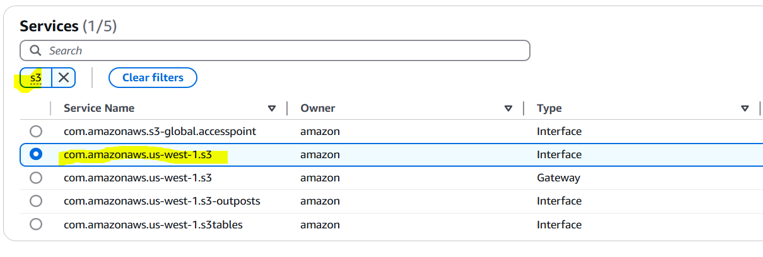


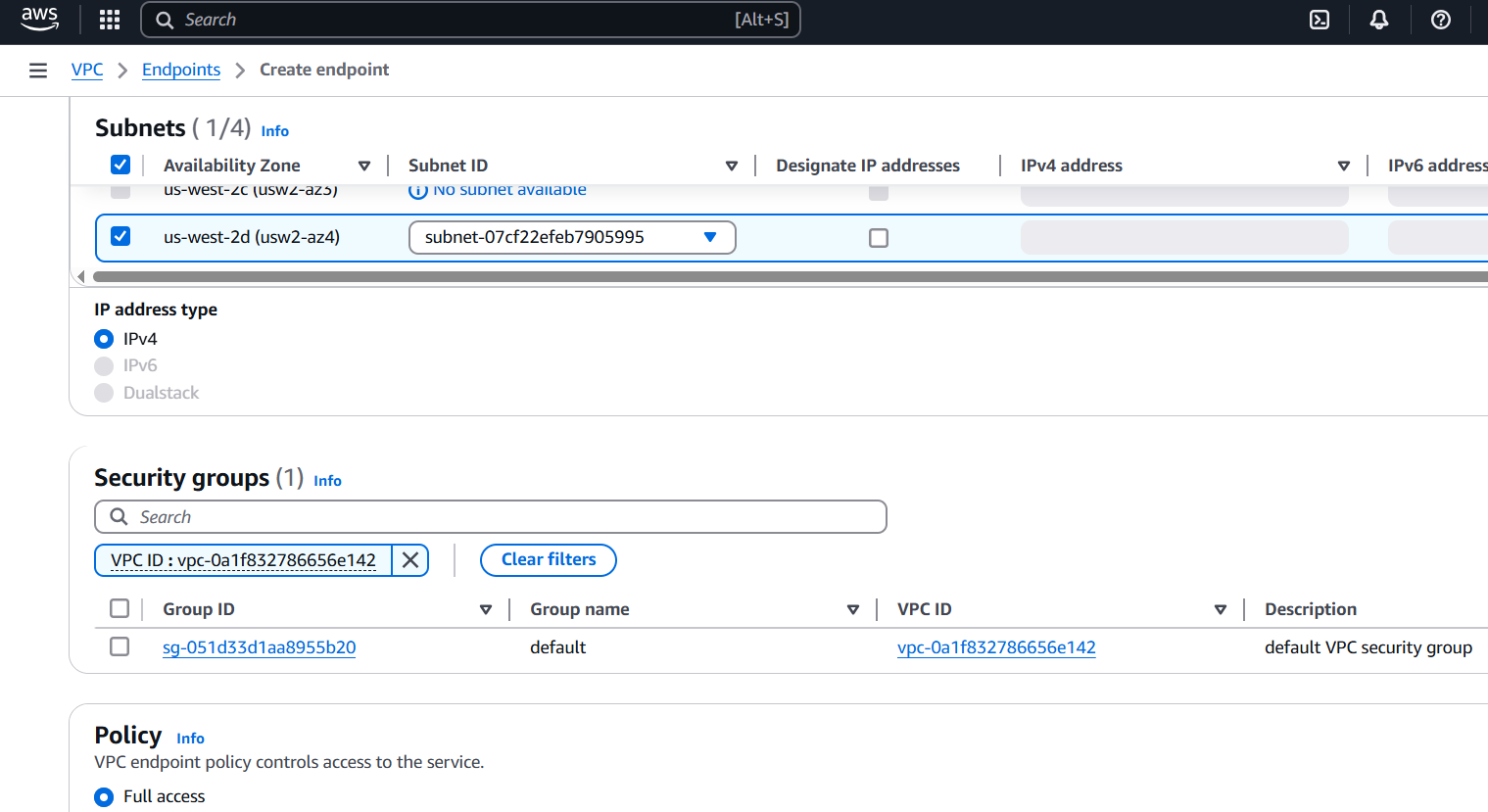
Similarly, add to all private vpc’s.

**Step 10: Create VPC Endpoints**

In each private VPC, go to VPC Endpoints and create Endpoints for s3.



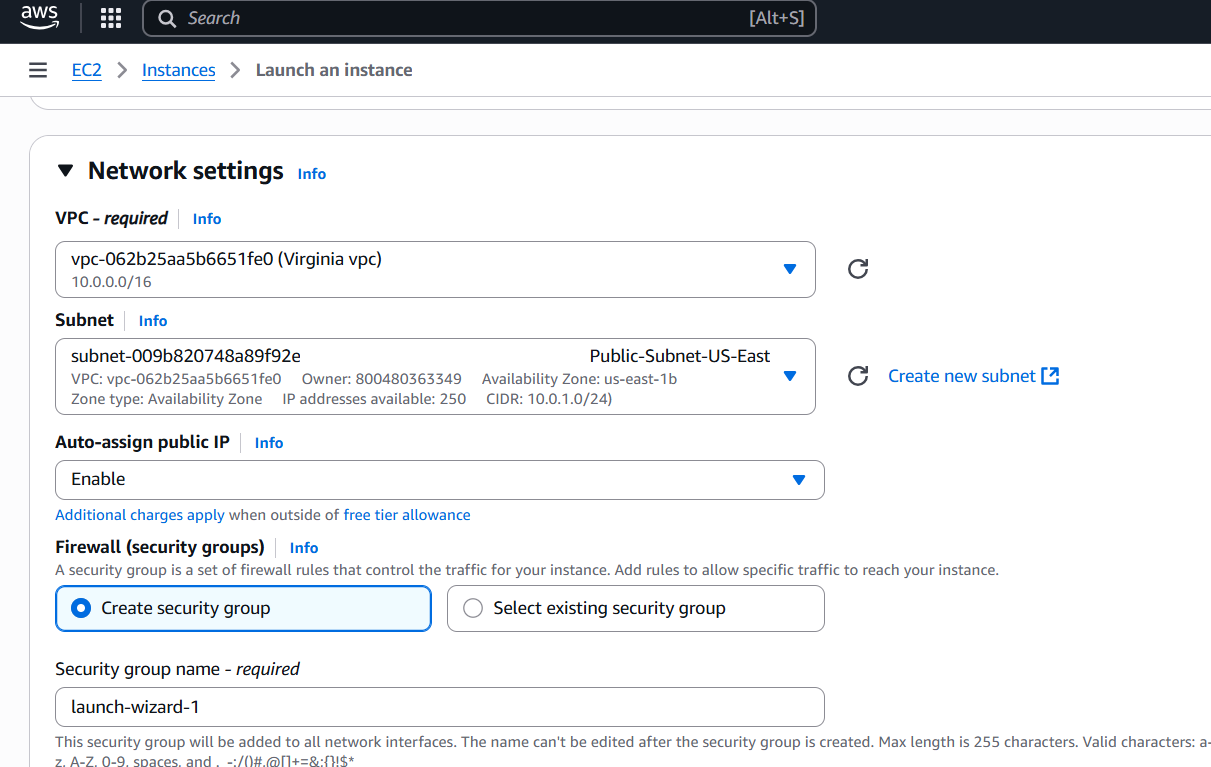


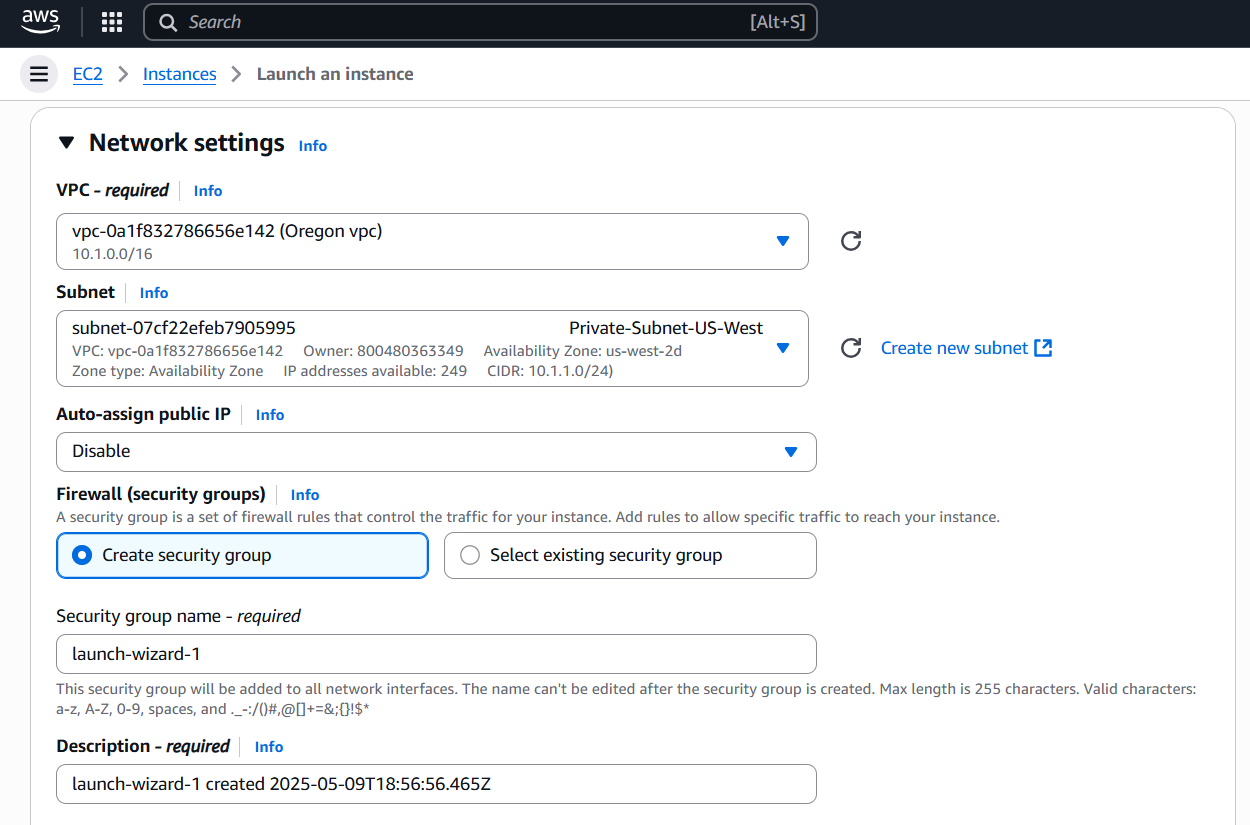


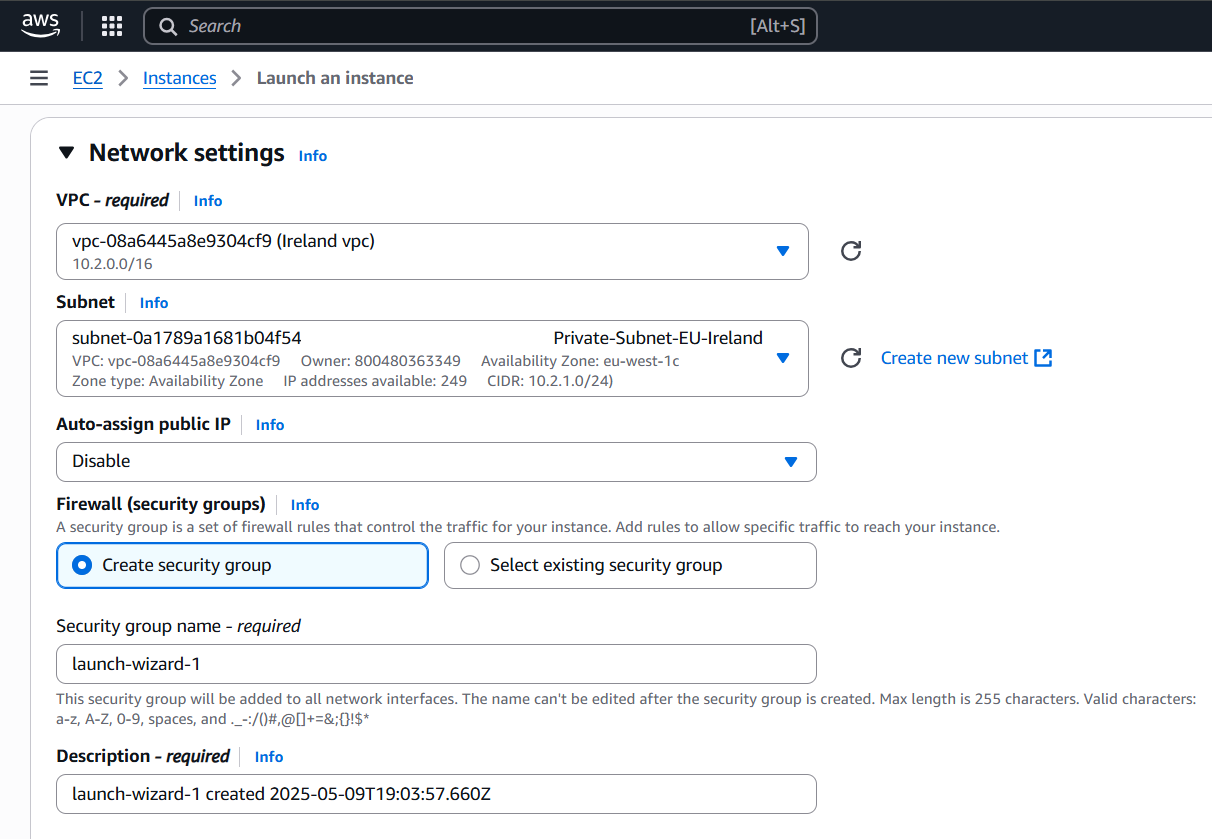
Similarly, make more 2 more end points for private subnets.

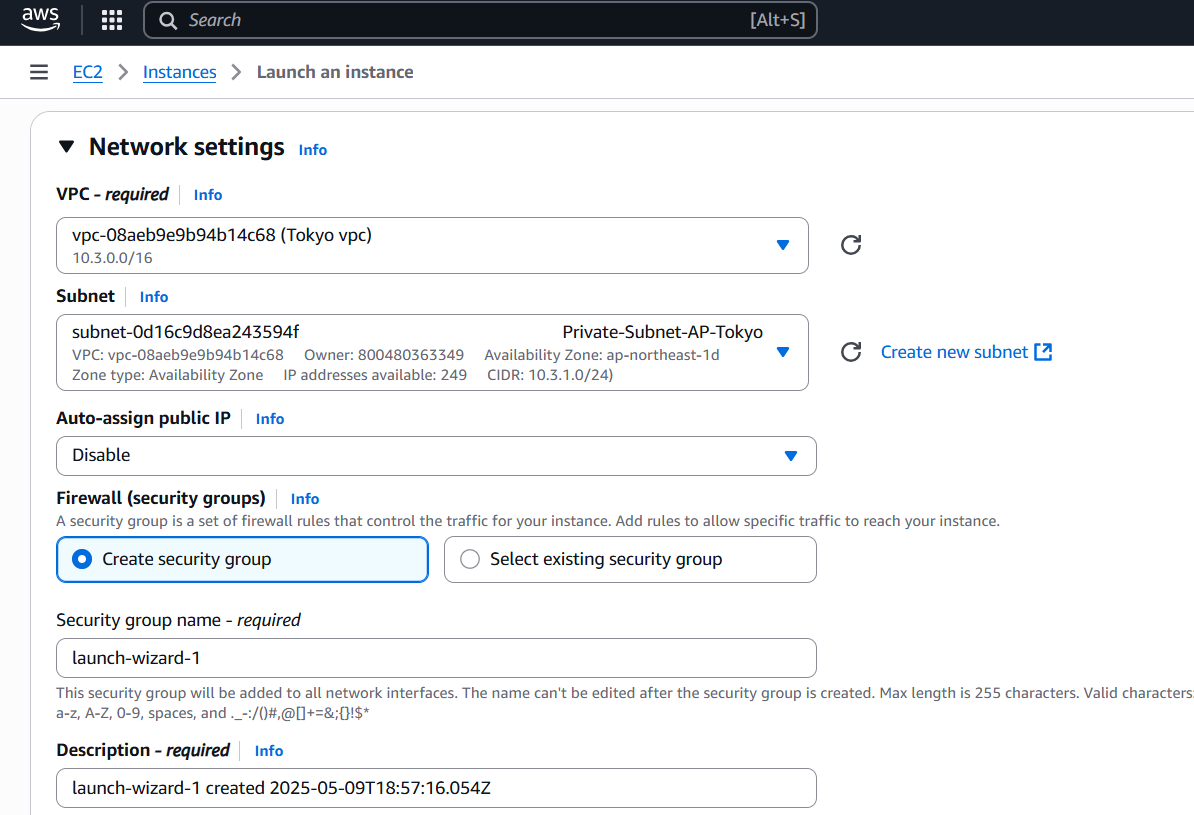
**Step 11: Launch EC2 Instances**

1. Launch one EC2 instance in Public-Subnet-US-East with a public IP.
2. Launch one EC2 instance in each private subnet (US-West, EU-Ireland, AP-Tokyo).
3. Use Amazon Linux 2 and enable SSM.
4. Make sure security groups allow ICMP (ping) and/or SSH from the correct CIDRs.



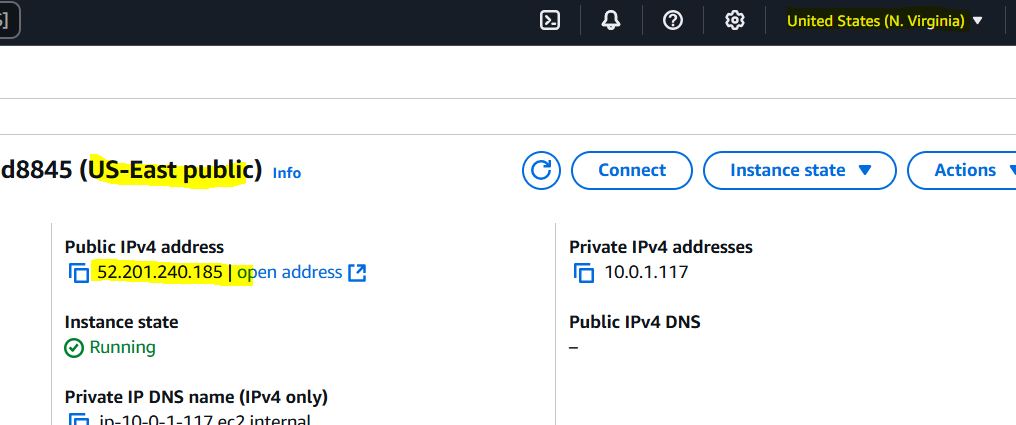


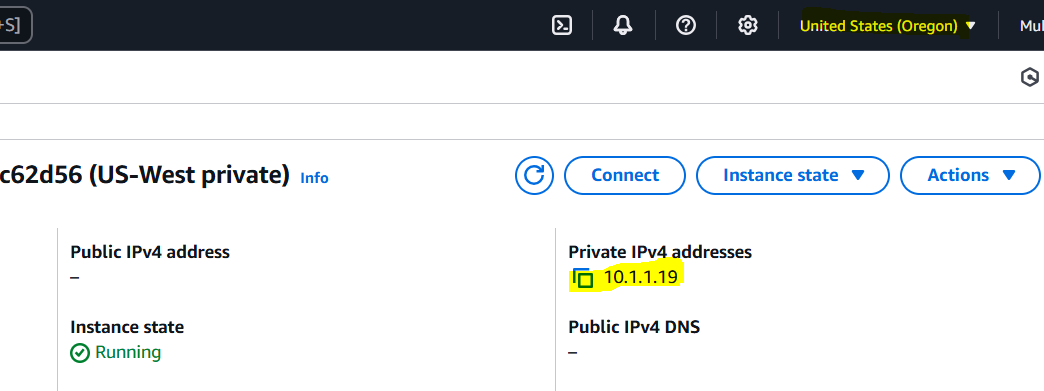


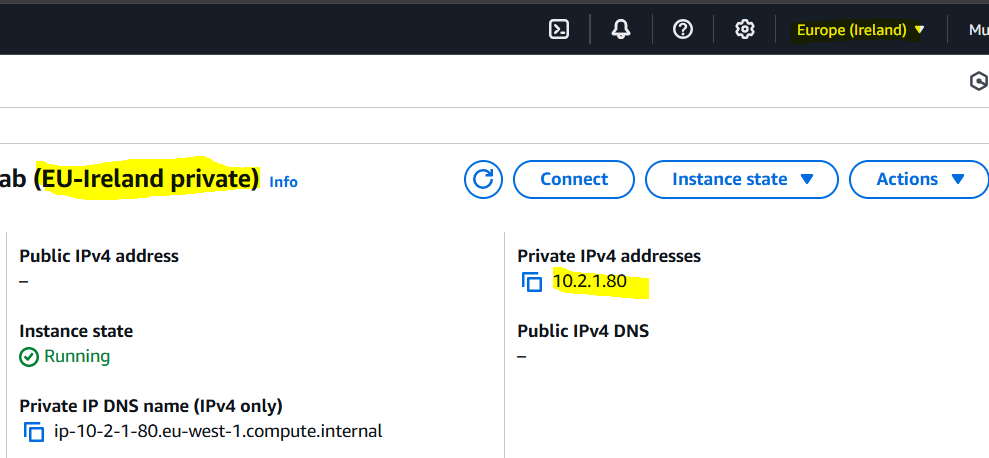


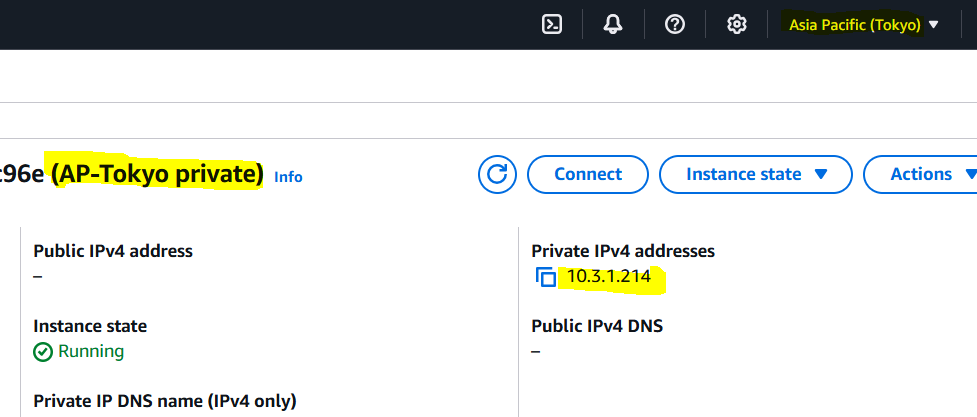
**Step 12: Test**

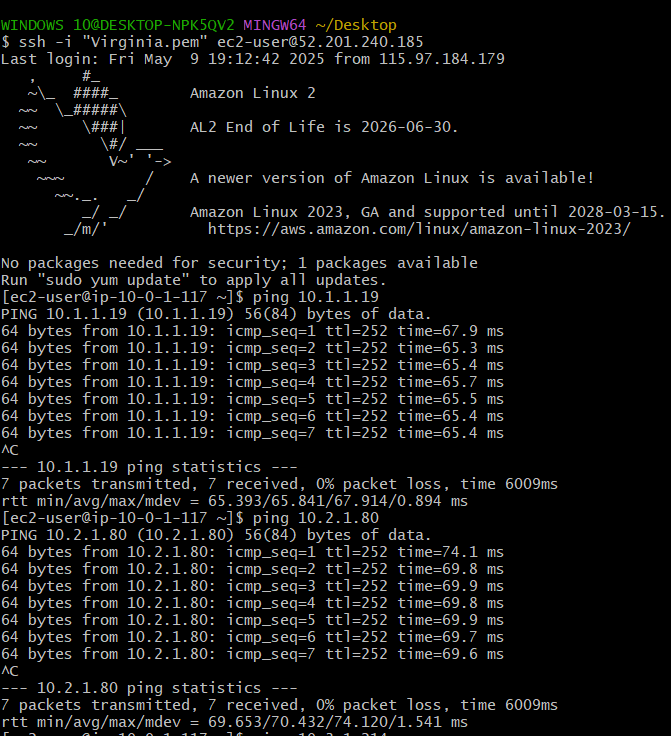
Test ping between private instances from 1 public instance.

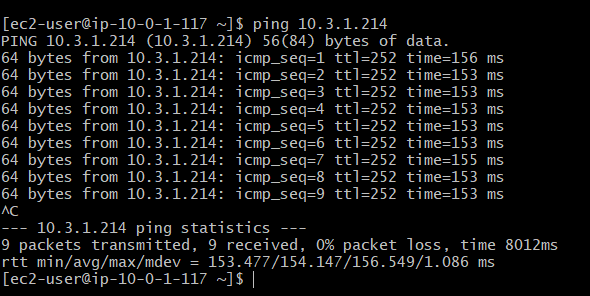




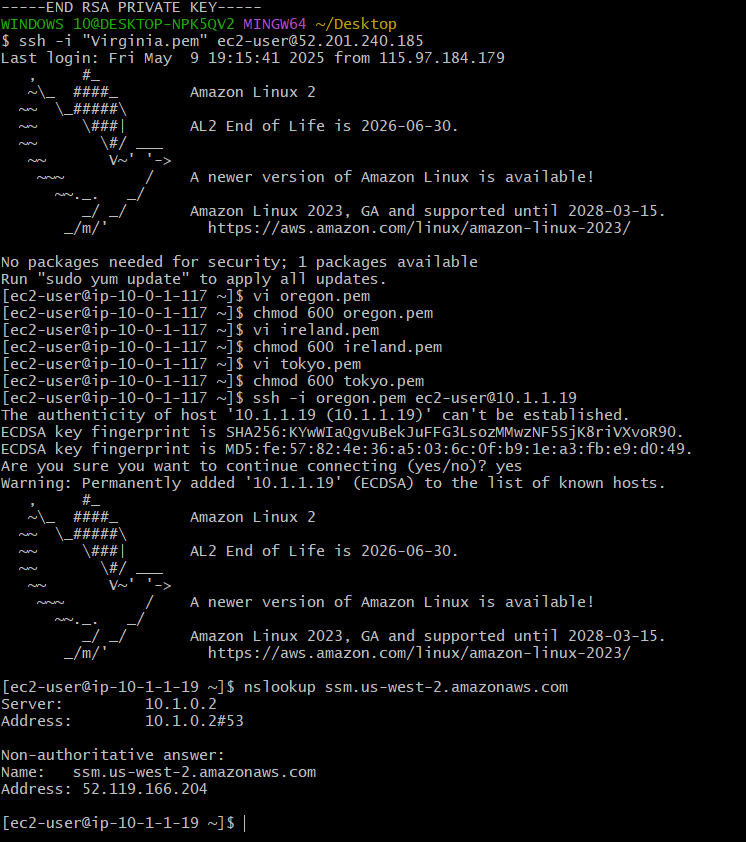


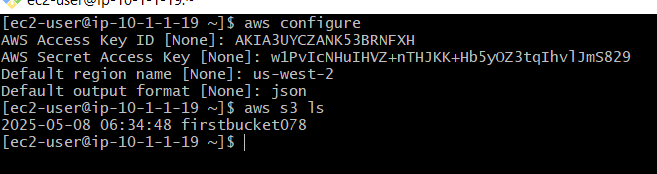


**Ping for private servers,**



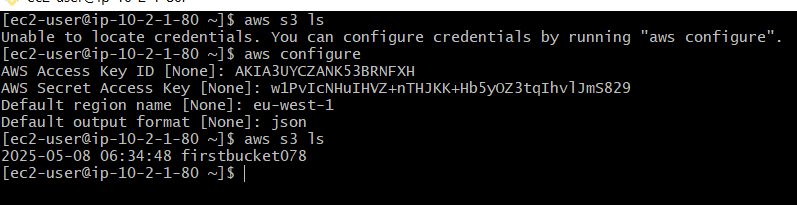
Connected to **Oregon region private** **server** through **public server of virginia** and checking Endpoint with s3 bucket.





Now connecting to **Ireland private server** from **Oregon private server** and checking Endpoint with s3 bucket.





**The-End**