**1)Create one VPC with 1 Public subnet and 1 Private subnet**

**1. Create VPC**

1. Login → AWS Console
2. Search → **VPC**
3. Click → **Create VPC**
4. Select → **VPC only**
5. Enter:

* Name: MyVPC
* IPv4 CIDR: 198.162.0.0/16

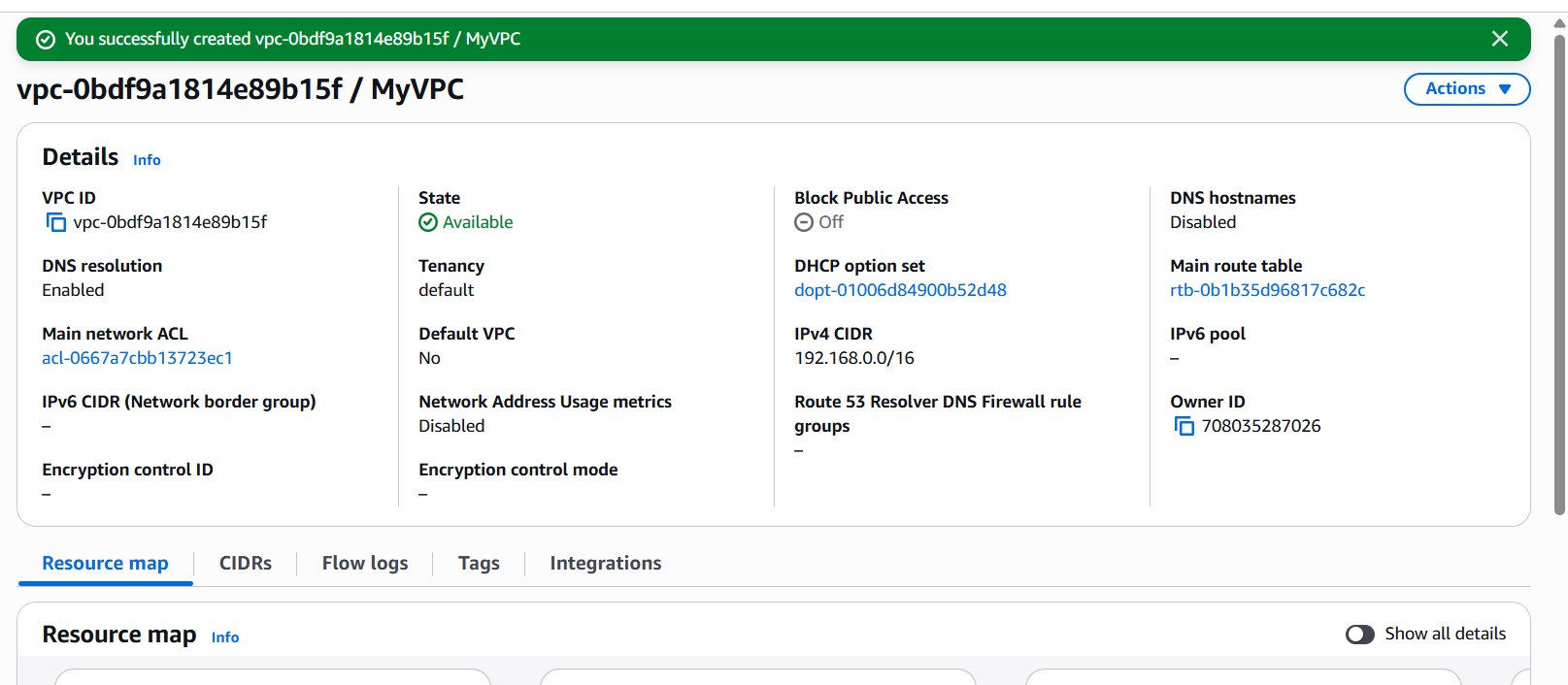
1. Click → **Create VPC**

Name: MyVPC

IPv4 CIDR block: 192.168.0.0/16

Now your VPC IP range will be:

192.168.0.0 → 192.168.255.255



**STEP 2: Create Public Subnet**

1. Go → Subnets → Create subnet
2. Select → MyVPC
3. Enter:

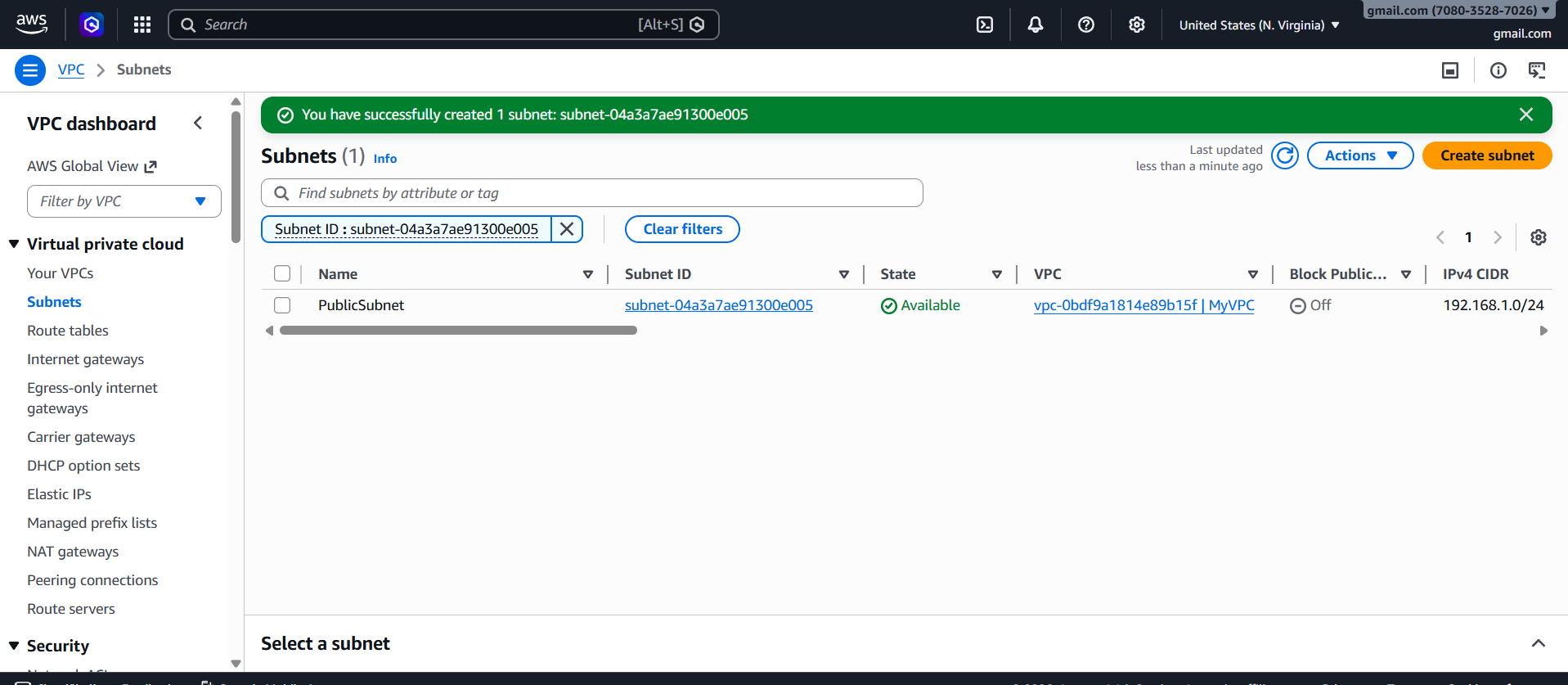
Name: PublicSubnet

CIDR block: 192.168.1.0/24

Click Create

This subnet range:

192.168.1.0 → 192.168.1.255



**STEP 3: Create Private Subnet**

1. Click → Create subnet
2. Enter:

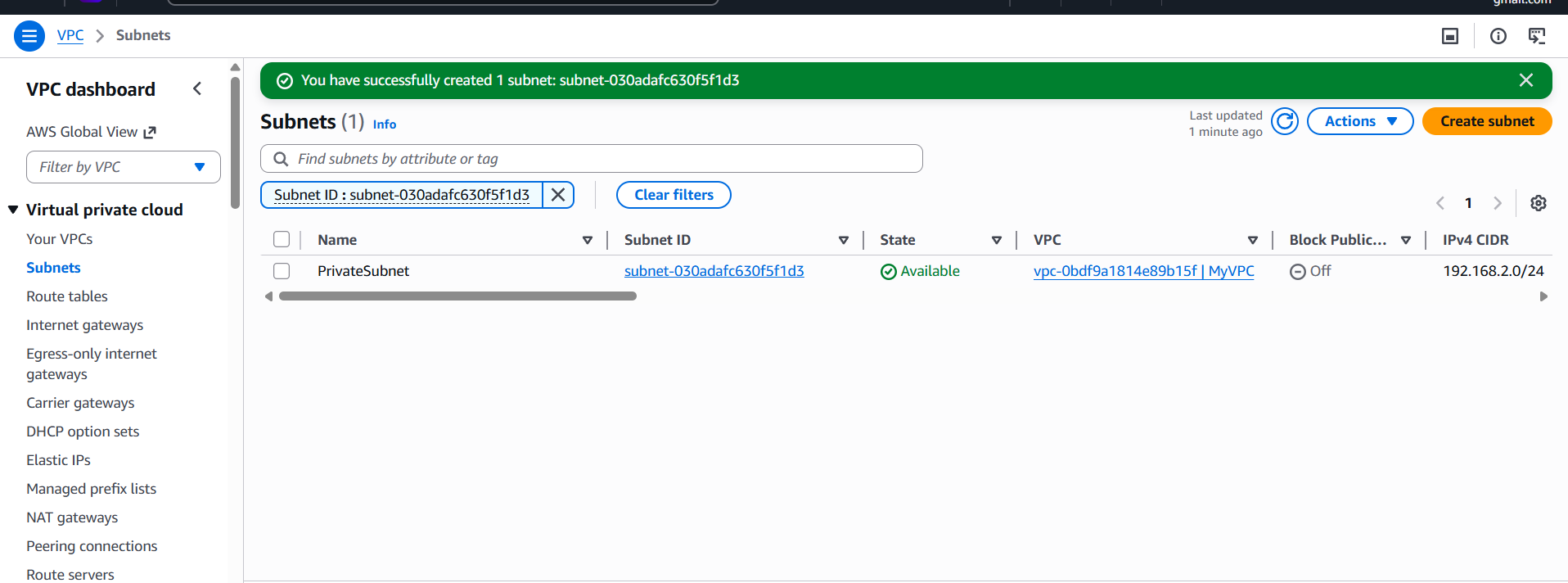
Name: PrivateSubnet

CIDR block: 192.168.2.0/24

Click Create

This subnet range:

192.168.2.0 → 192.168.2.255



**FINAL STRUCTURE**

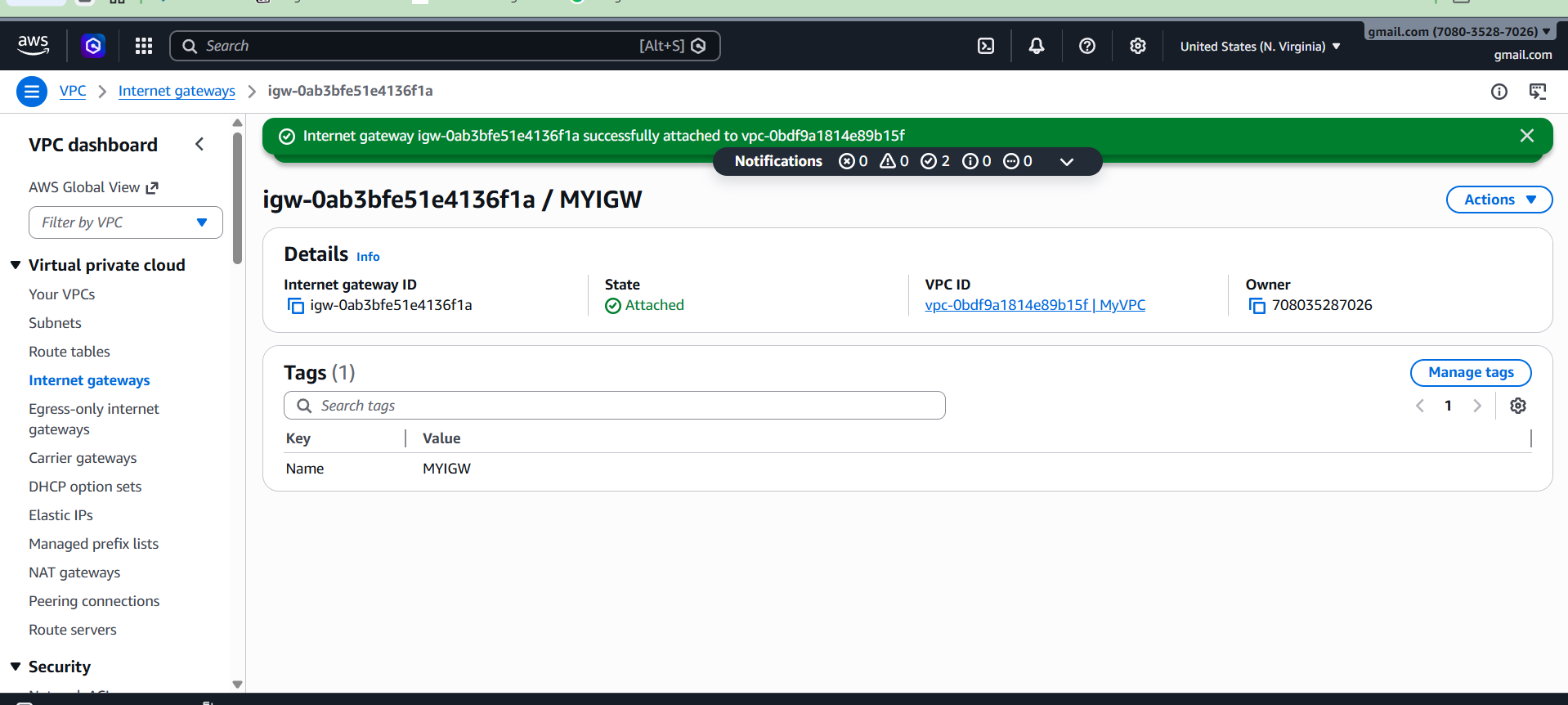
VPC: 192.168.0.0/16

Public Subnet: 192.168.1.0/24

Private Subnet: 192.168.2.0/24

**4. Create Internet Gateway**

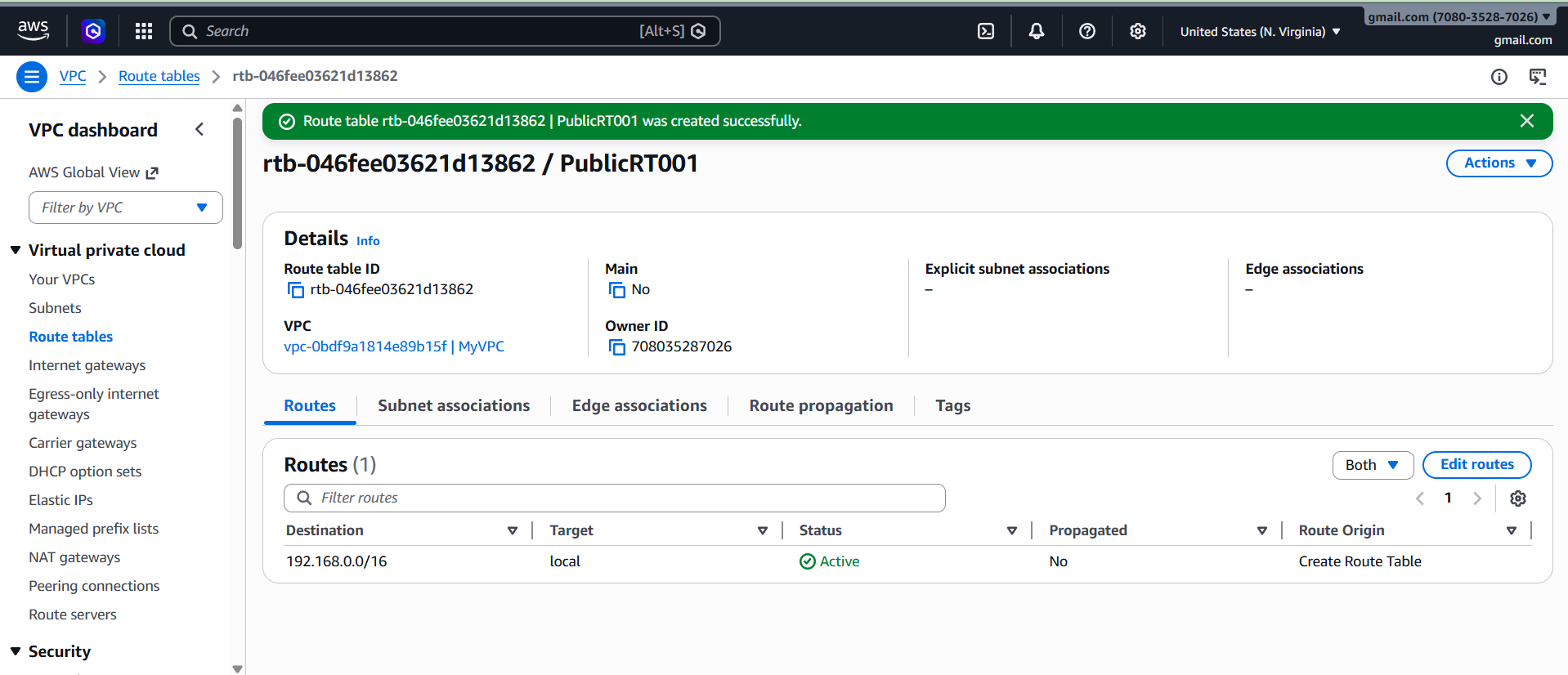
1. Go → Internet Gateway
2. Click → Create
3. Name: MyIGW
4. Click → Create
5. Click → Attach to VPC → select MyVPC



**5. Create Public Route Table**

1. Go → Route Tables
2. Click → Create
3. Name: PublicRT
4. Select VPC: MyVPC

Click Create



**6. Add Internet Route**

1. Select PublicRT
2. Click → Routes → Edit routes
3. Add:

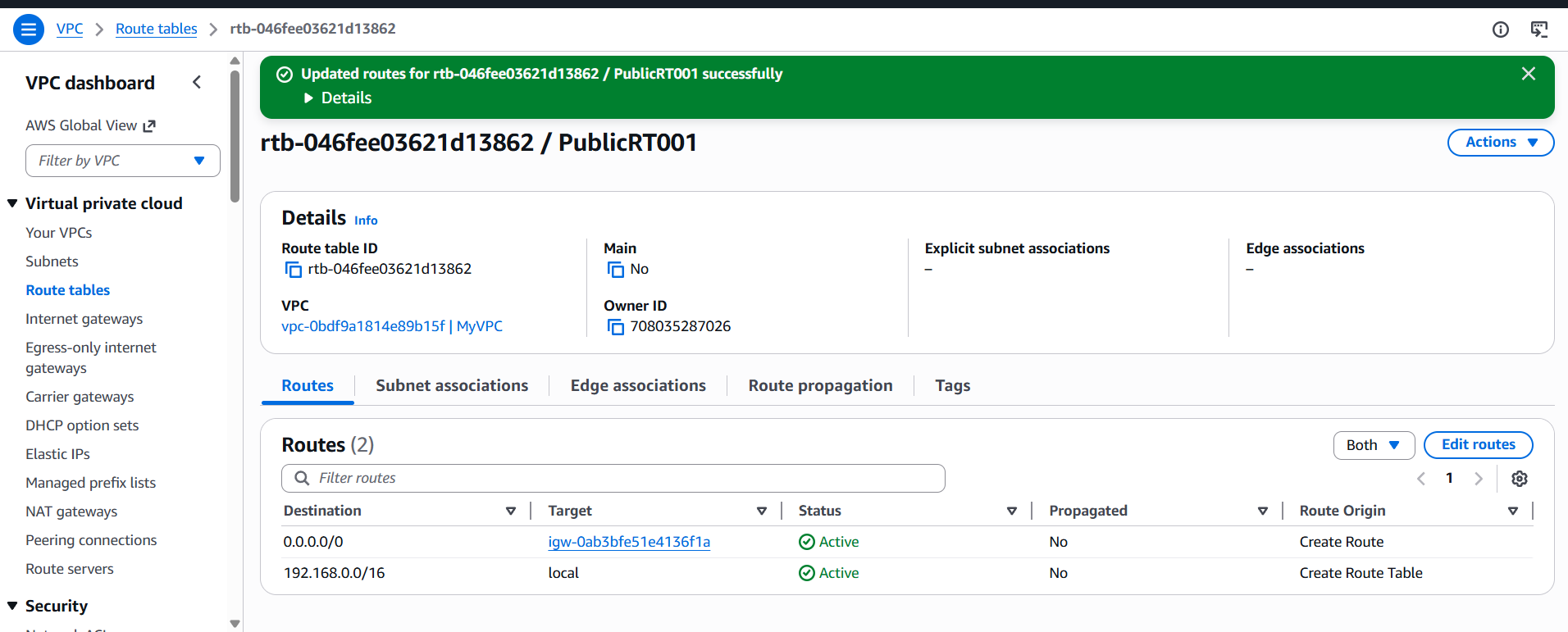
Destination:

0.0.0.0/0

Target:

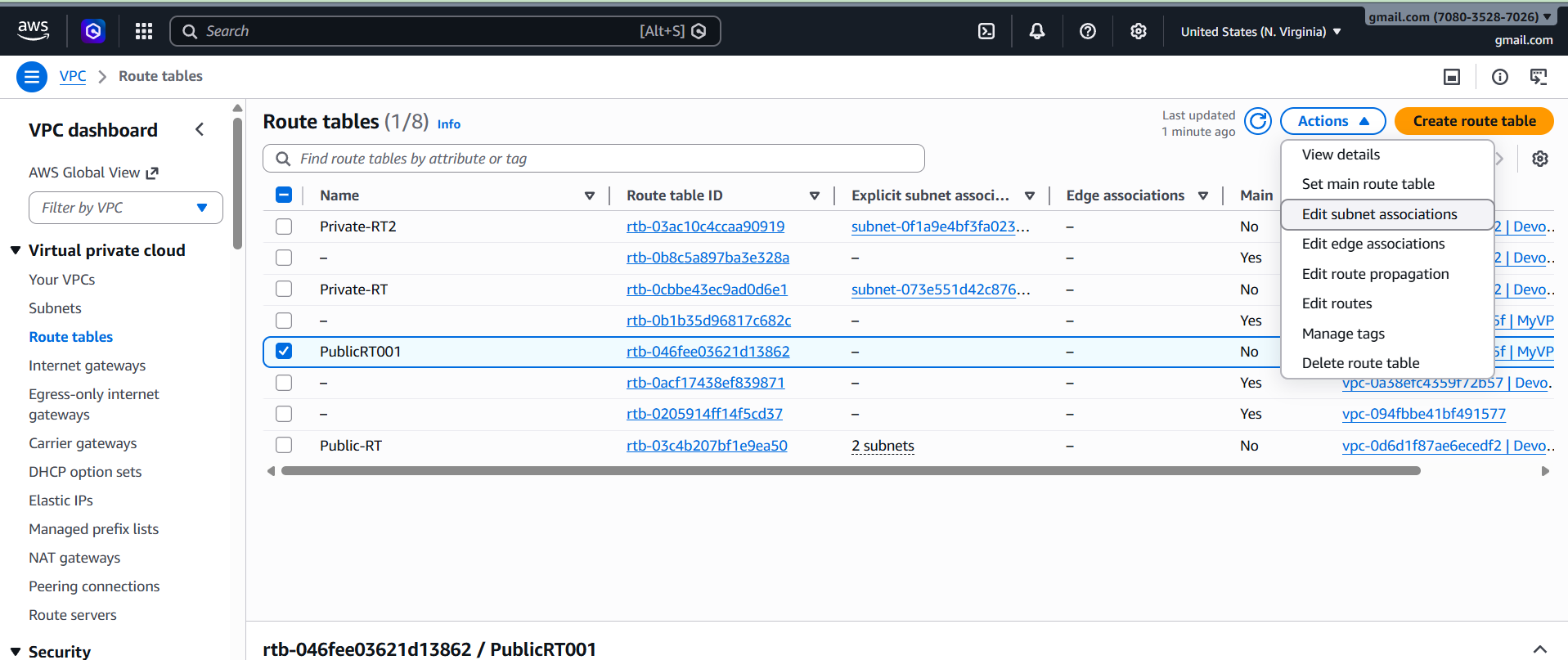
Internet Gateway → MyIGW

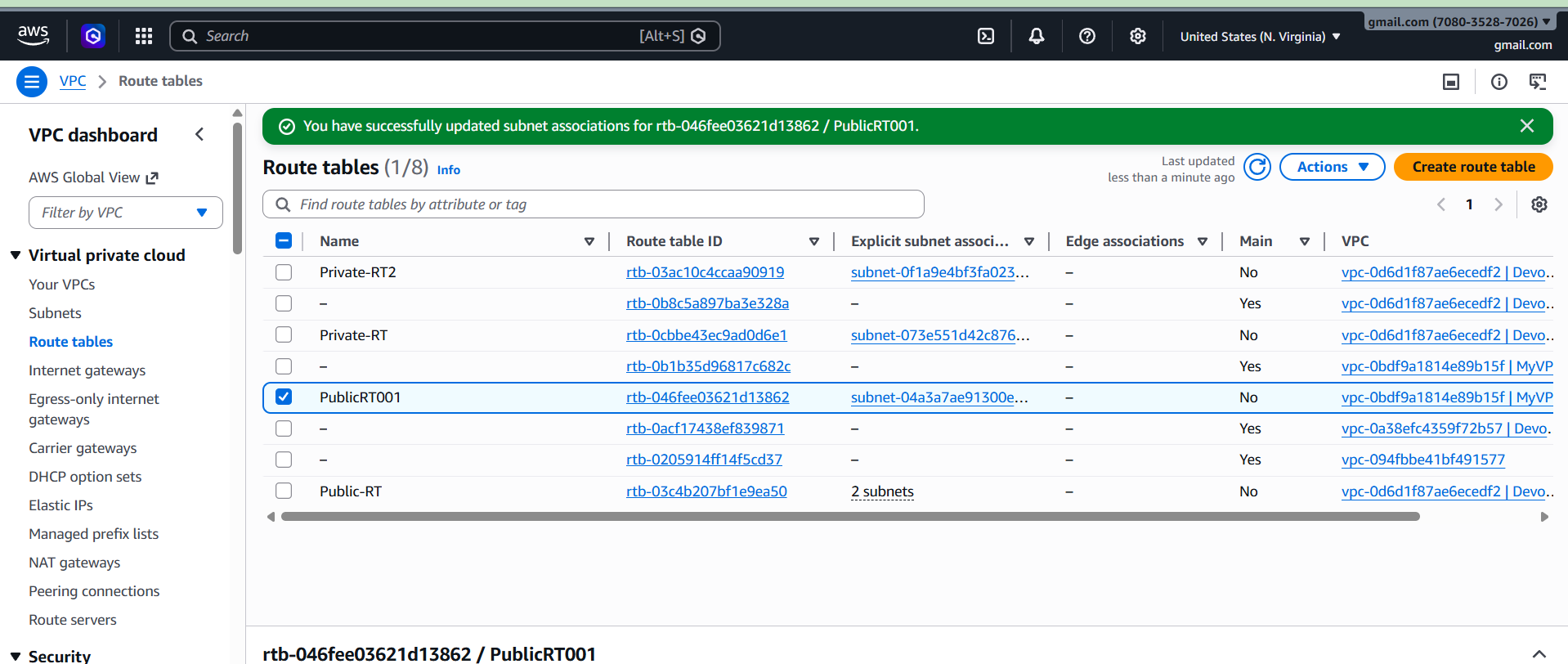
Save



**7. Associate Public Subnet**

1. Click → Subnet associations
2. Select → PublicSubnet
3. Save

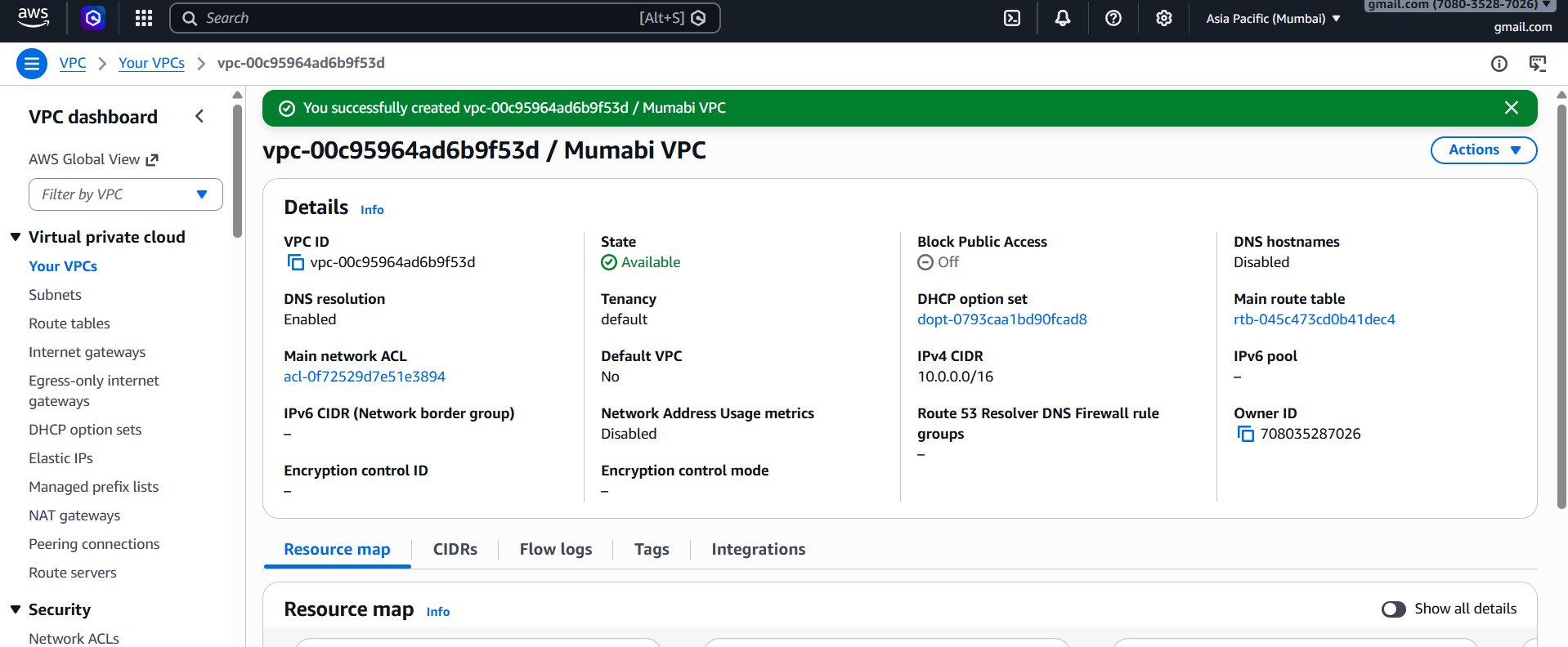




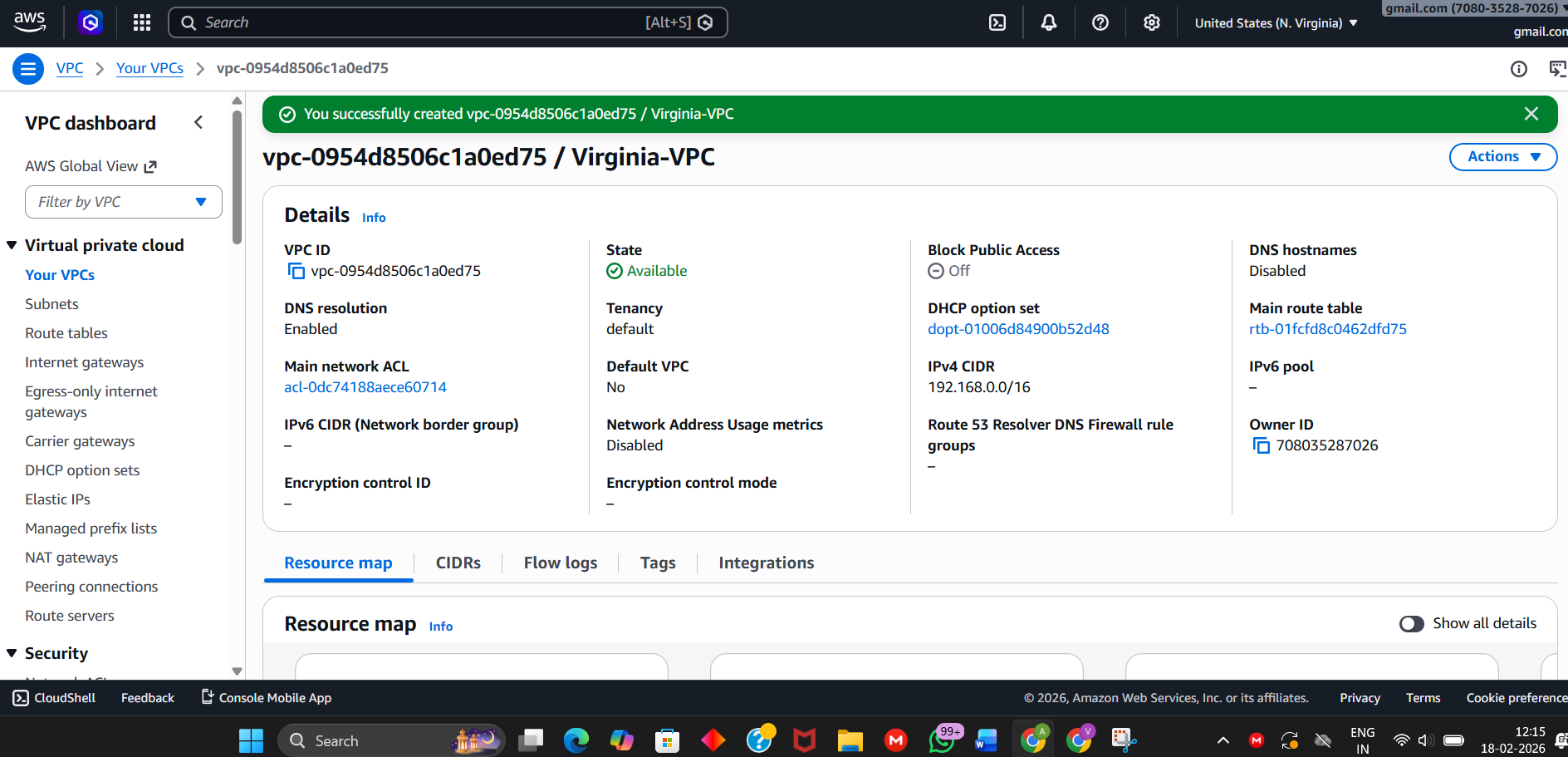
**2: Enable VPC Peering (Cross-Region)**

STEP 1: Create VPC in Mumbai Region  
 1. Login to AWS Console  
 2. Select Region: ap-south-1 (Mumbai)  
 3. Go to VPC Dashboard  
 4. Click Create VPC  
 5. Select VPC only  
 6. Name: Mumbai-VPC  
 7. CIDR: 10.0.0.0/16  
 8. Click Create

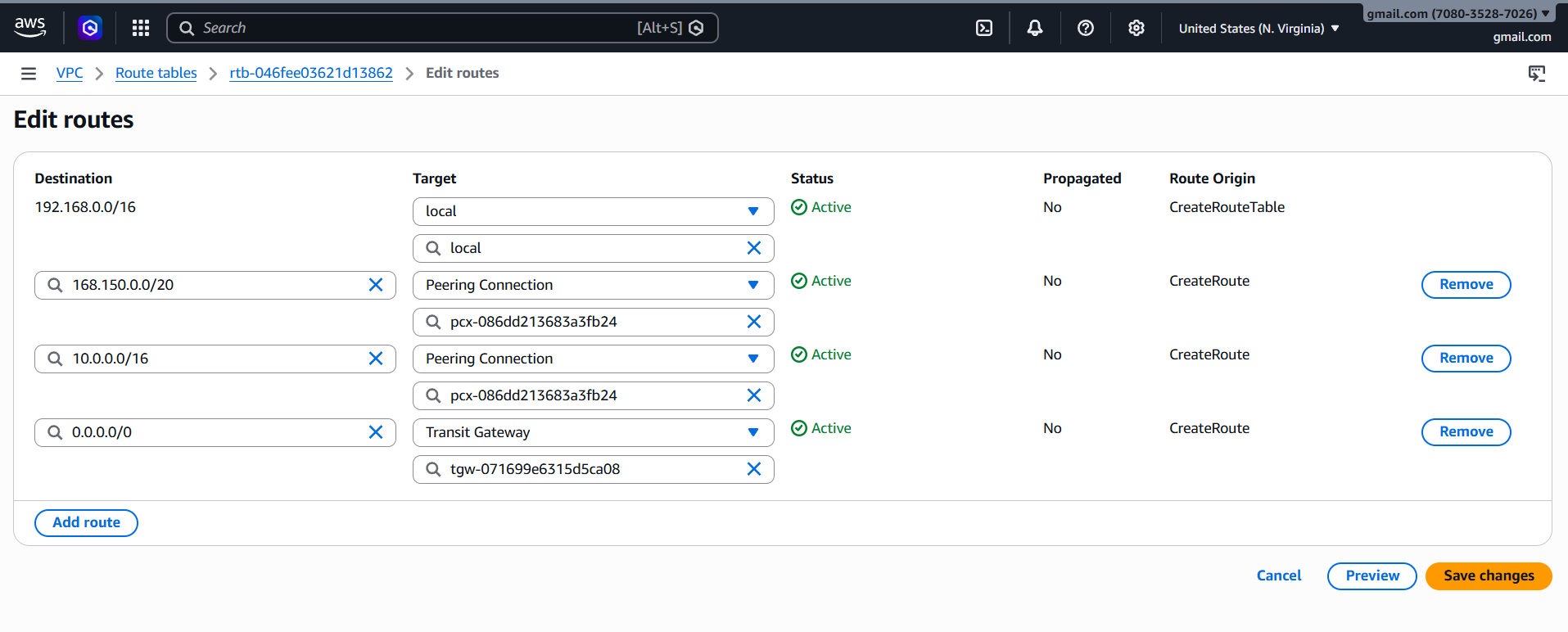
**VPC → Create VPC**

****

**STEP 2: Create VPC in Virginia Region**  
1. Change Region to us-east-1 (Virginia)  
2. Go to VPC Dashboard  
3. Click Create VPC  
4. Name: Virginia-VPC  
5. CIDR: 192.168.0.0/16  
6. Click Create

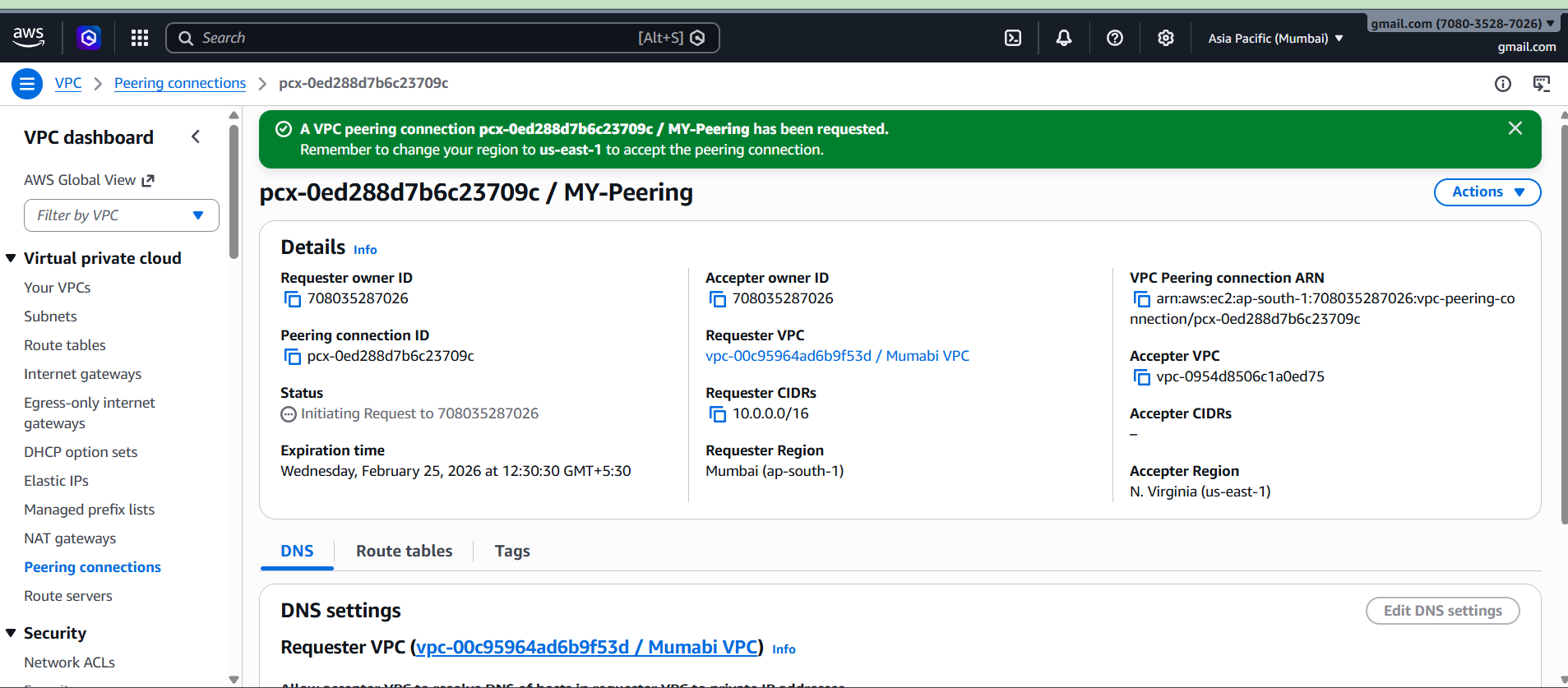
****





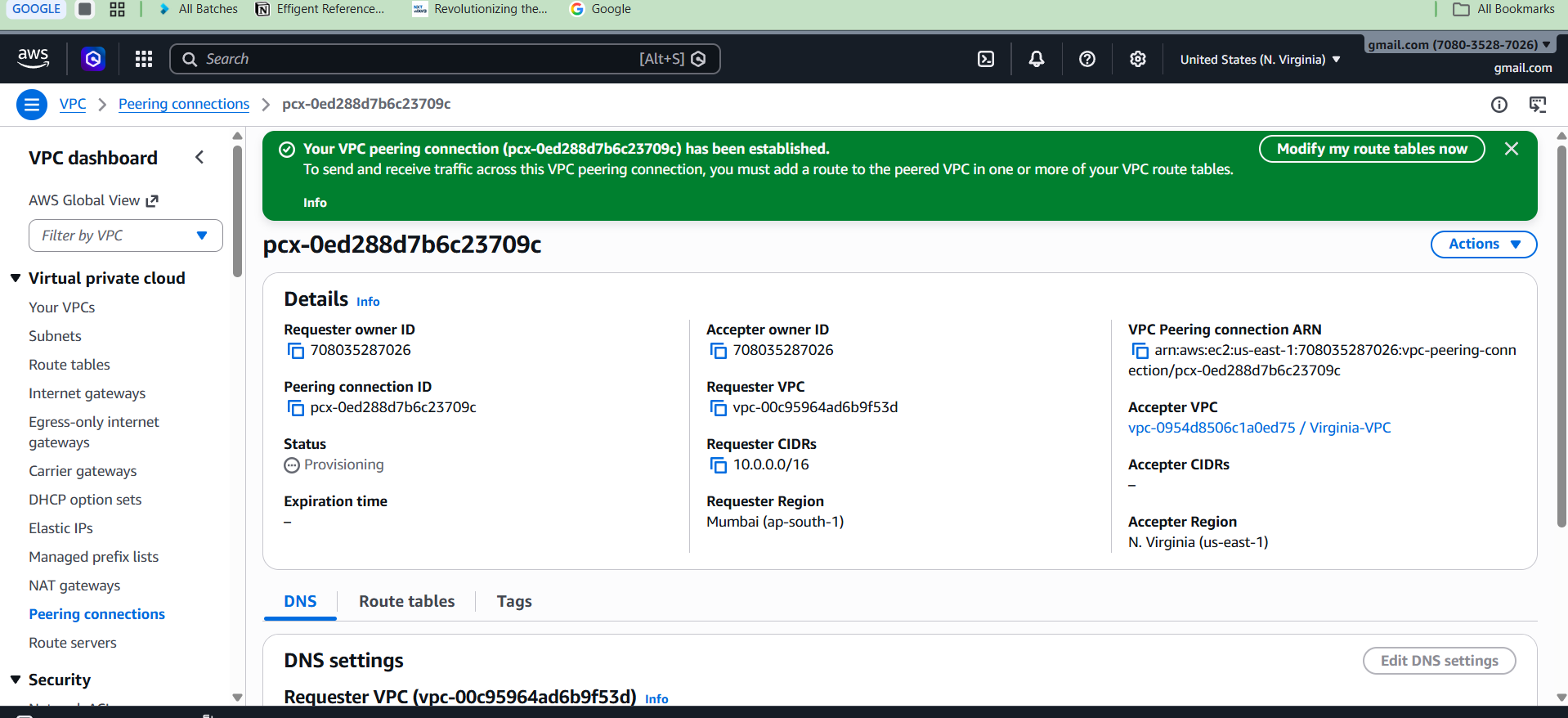
**STEP 3: Create Peering Connection**

* 1. Go to Mumbai Region  
     2. Open VPC Dashboard  
     3. Click Peering Connections  
     4. Click Create Peering Connection  
     5. Name: Cross-Region-Peering  
     6. Select Requester VPC: Mumbai-VPC  
     7. Select Another Region  
     8. Choose us-east-1  
     9. Select Virginia-VPC ID  
     10. Click Create





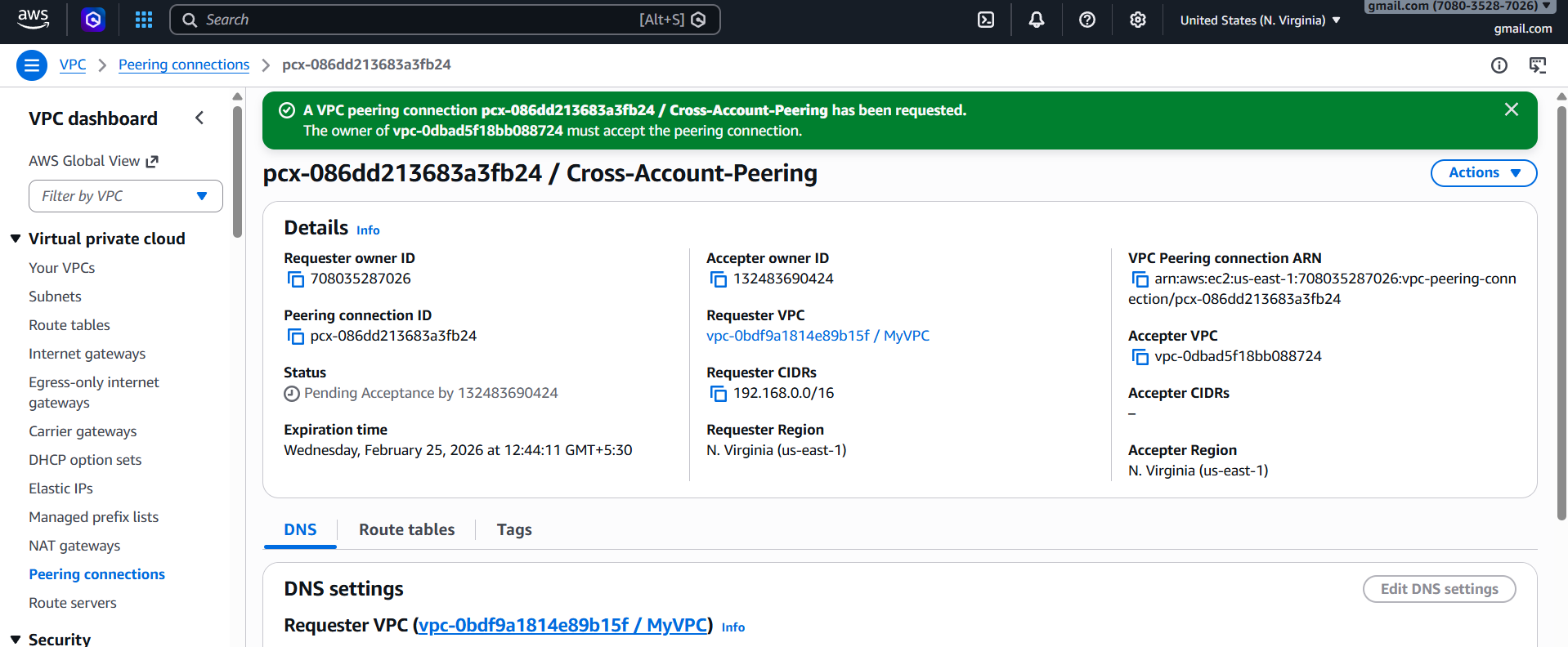
**STEP 4: Accept Peering Connection**  
1. Switch to Virginia Region  
2. Open VPC Dashboard  
3. Click Peering Connections  
4. Select Cross-Region-Peering  
5. Click Actions  
6. Click Accept Request  
7. Status becomes Active



**3) AWS Cross-Account VPC Peering**

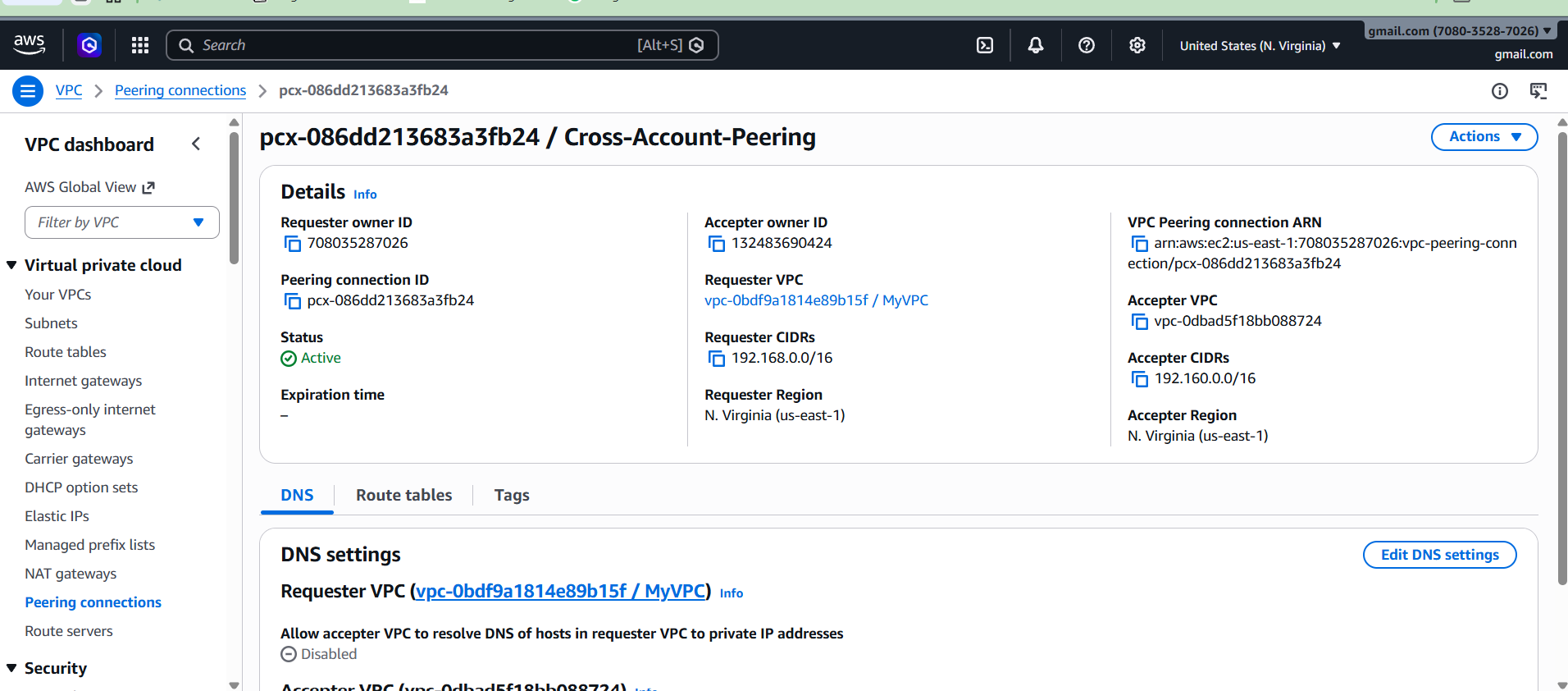
Cross-Account VPC Peering allows two VPCs from different AWS accounts to communicate privately.  
  
**EXAMPLE USED**  
Your Account ID: 111111111111  
Friend Account ID: 222222222222  
  
**Your VPC:**  
Region: ap-south-1  
CIDR: 10.0.0.0/16  
VPC ID: vpc-xxxx  
 **Friend VPC:**  
Region: us-east-1  
CIDR: 192.168.0.0/16  
VPC ID: vpc-yyyy  
 **STEP 1: Get Friend Details**

Ask friend to provide:  
• Account ID  
• VPC ID  
• Region  
• CIDR block  
  
**STEP 2: Create Peering from Your Account**  
1. Login to AWS Console  
2. Go to VPC Dashboard  
3. Click Peering Connections  
4. Click Create Peering Connection  
5. Name: Cross-Account-Peering  
6. Requester VPC: Select your VPC  
7. Account: Select Another account  
8. Enter Friend Account ID  
9. Select Friend Region  
10. Enter Friend VPC ID (vpc-xxxx)  
11. Click Create

****

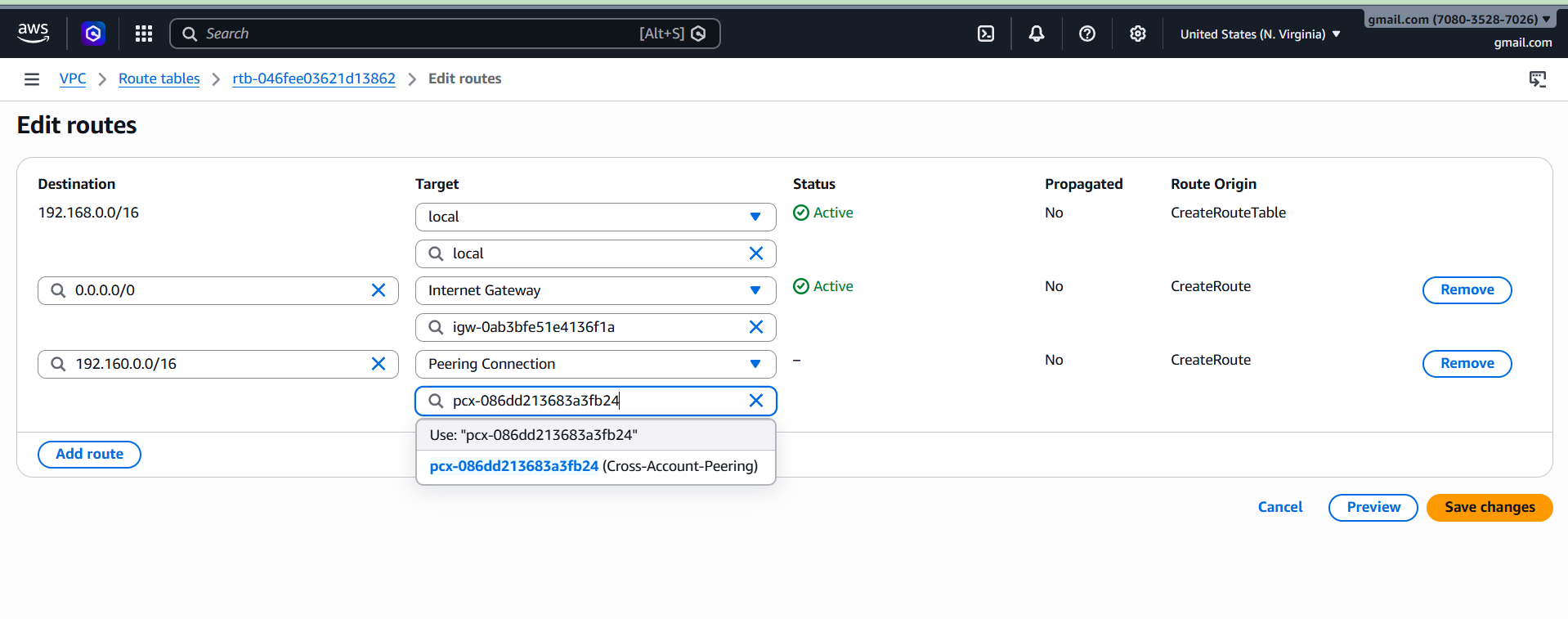
**STEP 3: Friend Accept Peering**

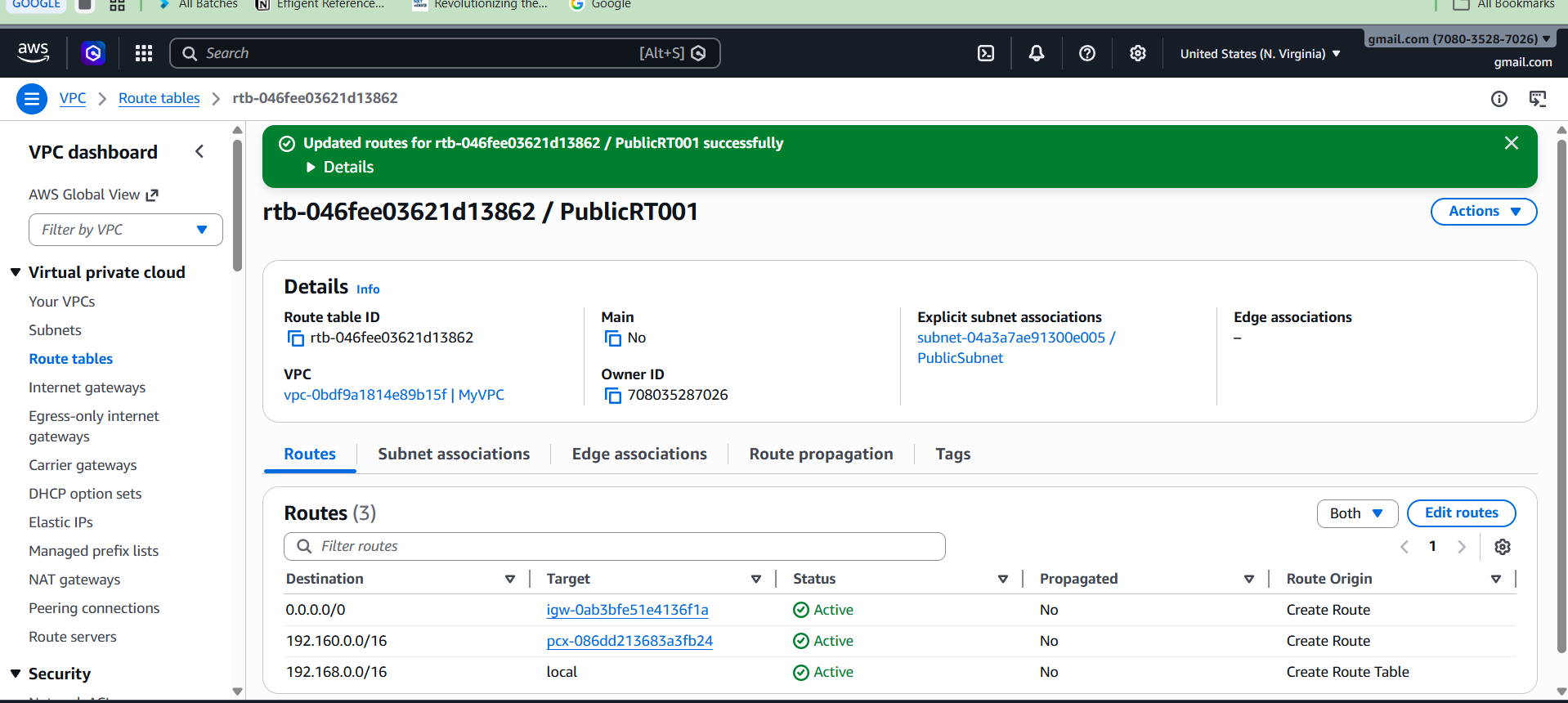
Friend must:  
1. Login to AWS Console  
2. Go to VPC Dashboard  
3. Click Peering Connections  
4. Select Cross-Account-Peering  
5. Click Actions  
6. Click Accept Request  
7. Status becomes Active

****

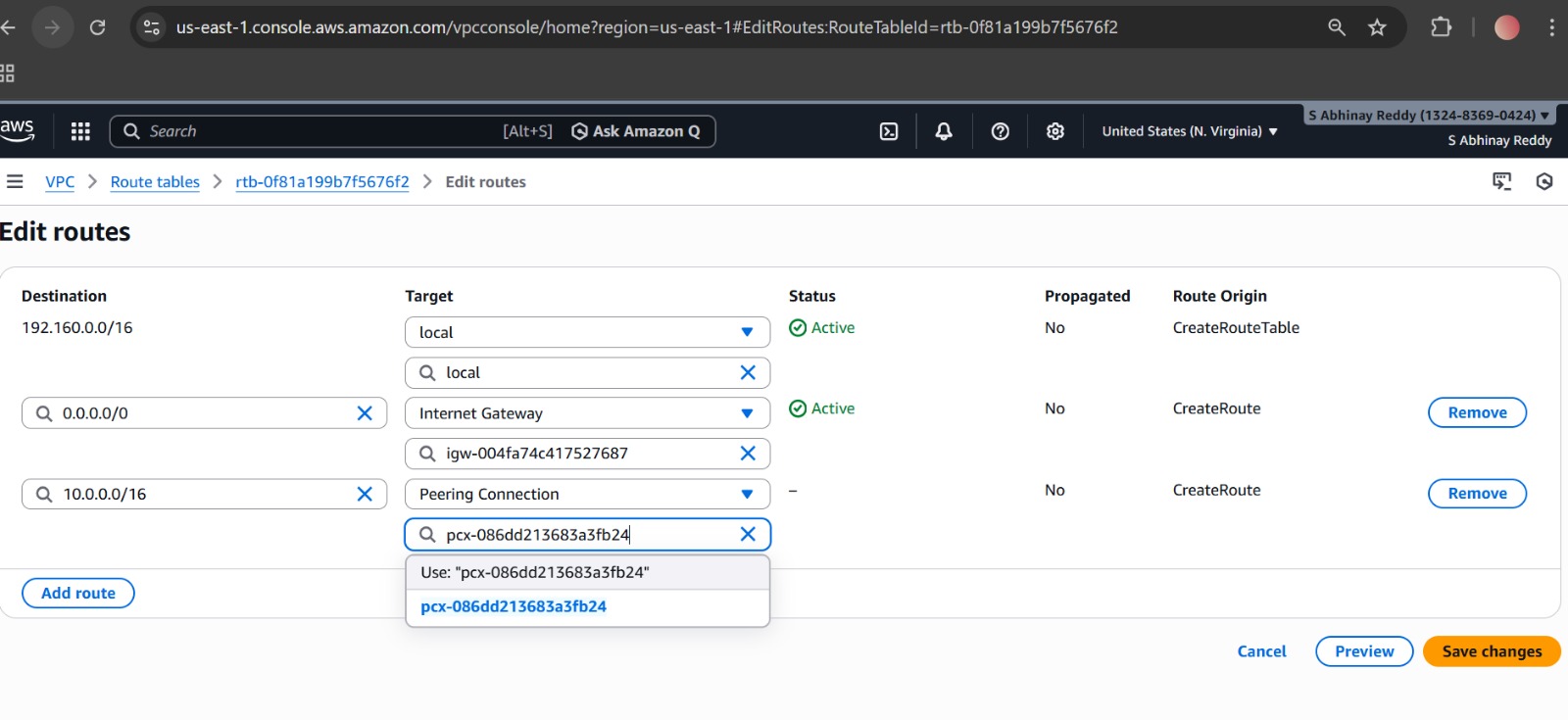
**STEP 4: Update Route Table in Your Account**

1. Go to VPC Dashboard  
2. Click Route Tables  
3. Select Route Table  
4. Click Edit Routes  
5. Click Add Route  
6. Destination: Friend CIDR (192.168.0.0/16)  
7. Target: Peering Connection  
8. Click Save

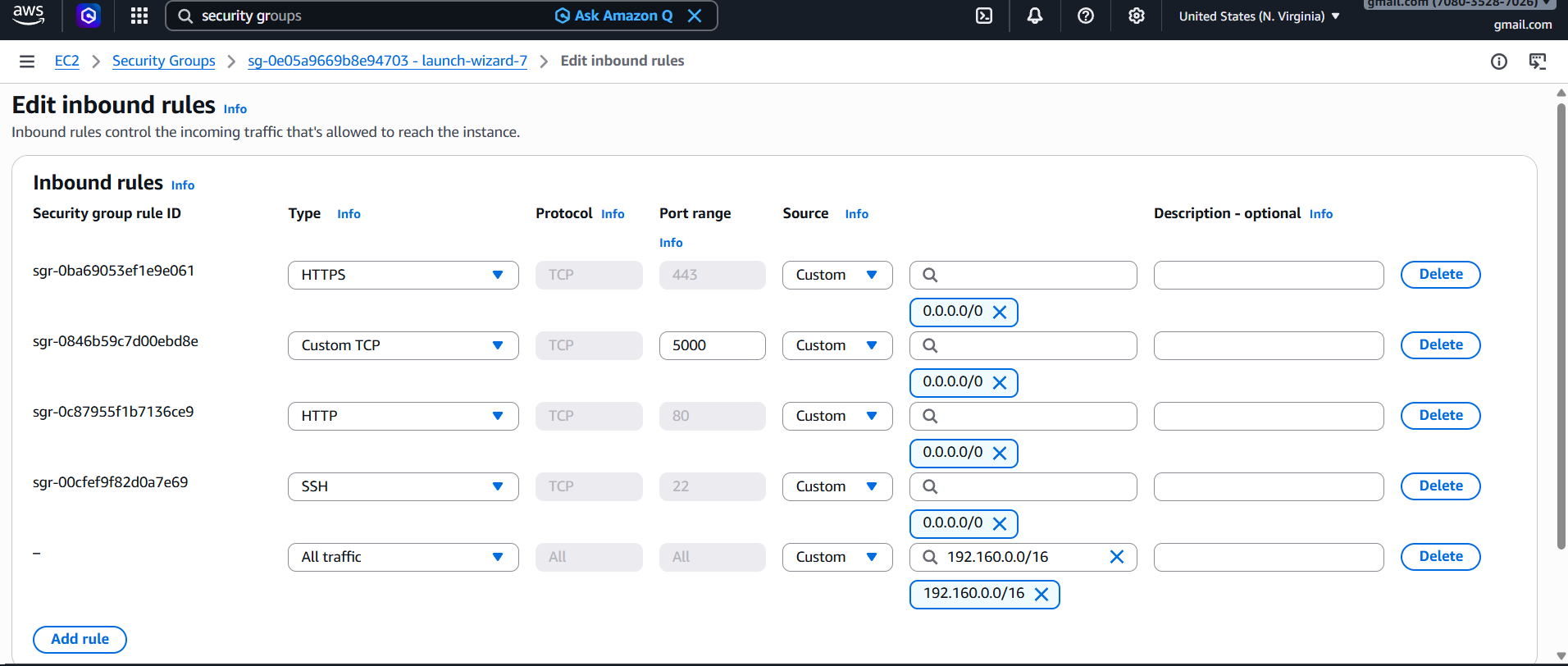




**STEP 5: Update Route Table in Friend Account**  
1. Go to Route Tables  
2. Click Edit Routes  
3. Click Add Route  
4. Destination: Your CIDR (10.0.0.0/16)  
5. Target: Peering Connection  
6. Click Save

****

**STEP 6: Update Security Groups**  
Allow inbound traffic from other VPC CIDR



**STEP 7: Test Connection**  
Launch EC2 in both VPCs and test ping using private IP.

**RESULT**  
Both AWS accounts VPCs can communicate privately successfully

**4: Set up a VPC Transit Gateway** step-by-step in **AWS Console**

**Step 1: Open Transit Gateway**

1. Login to AWS Console
2. Search → **VPC**
3. Left side → click **Transit Gateways**
4. Click **Create transit gateway**

**Step 2: Create Transit Gateway**

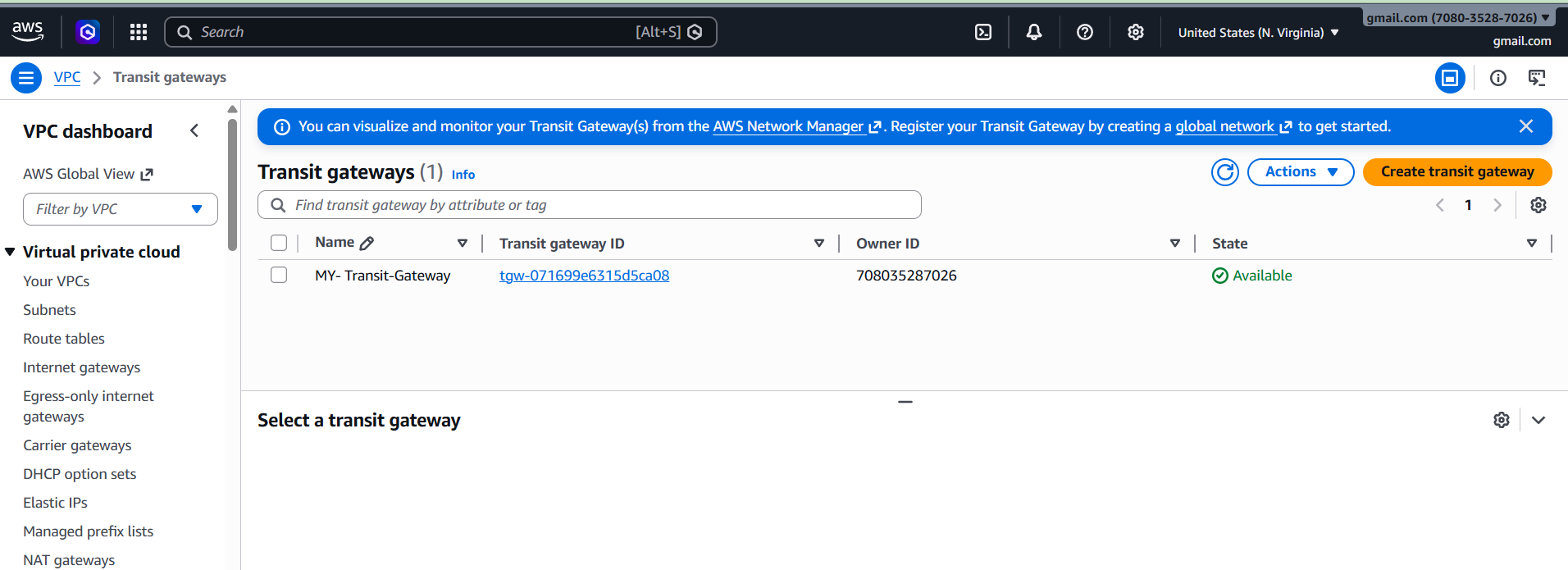
1. Enter Name tag: My-Transit-Gateway

2. Enter Description: TGW for VPC connection

3. Leave other settings as default.

4. Click 'Create transit gateway'.

5. Wait until status becomes 'Available'

****

**Step 3: Create Transit Gateway Attachment**

1. Click 'Transit Gateway Attachments'.

2. Click 'Create transit gateway attachment'.

3. Select your Transit Gateway.

4. Select Attachment type: VPC

5. Select your VPC.

6. Select subnet (public Id).

7. Click 'Create attachment'.

8. Wait until status becomes 'Available'.

****

**Step 4: Update Route Table**

1. Go to VPC → Route Tables.

2. Select your route table.

3. Click 'Edit routes'.

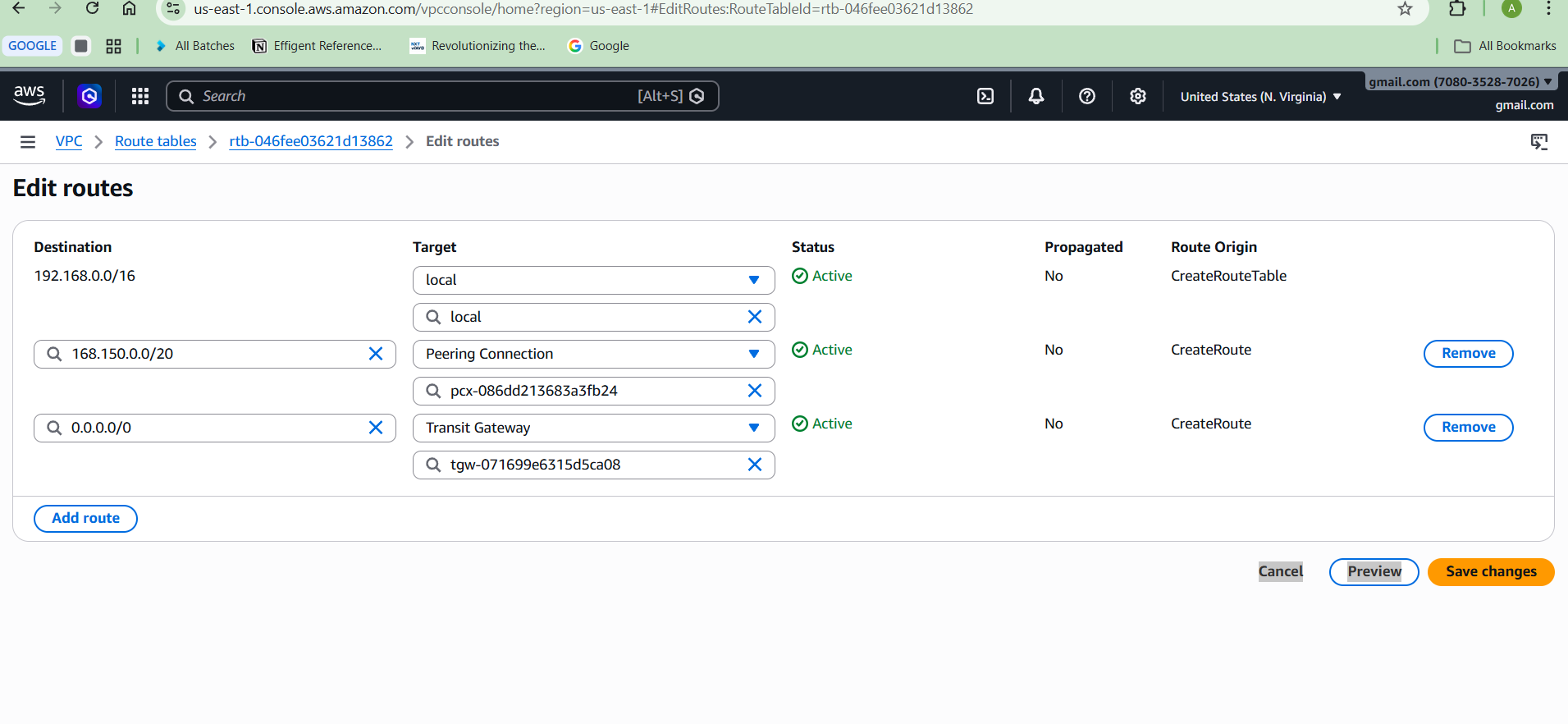
4. Click 'Add route'.

5. Destination: 0.0.0.0/0

6. Target: Transit Gateway

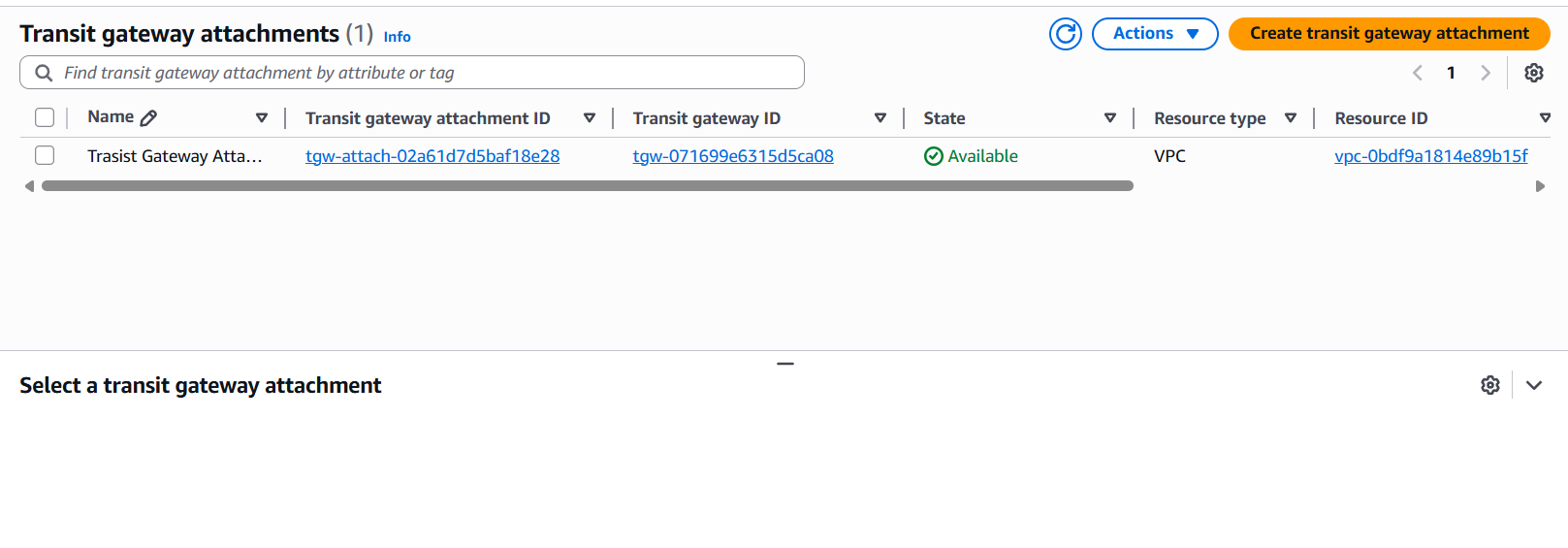
7. Select Transit Gateway ID.

8. Click 'Save routes'.

****

**Verification**

Go to Transit Gateway Attachments and confirm status shows 'Available'.

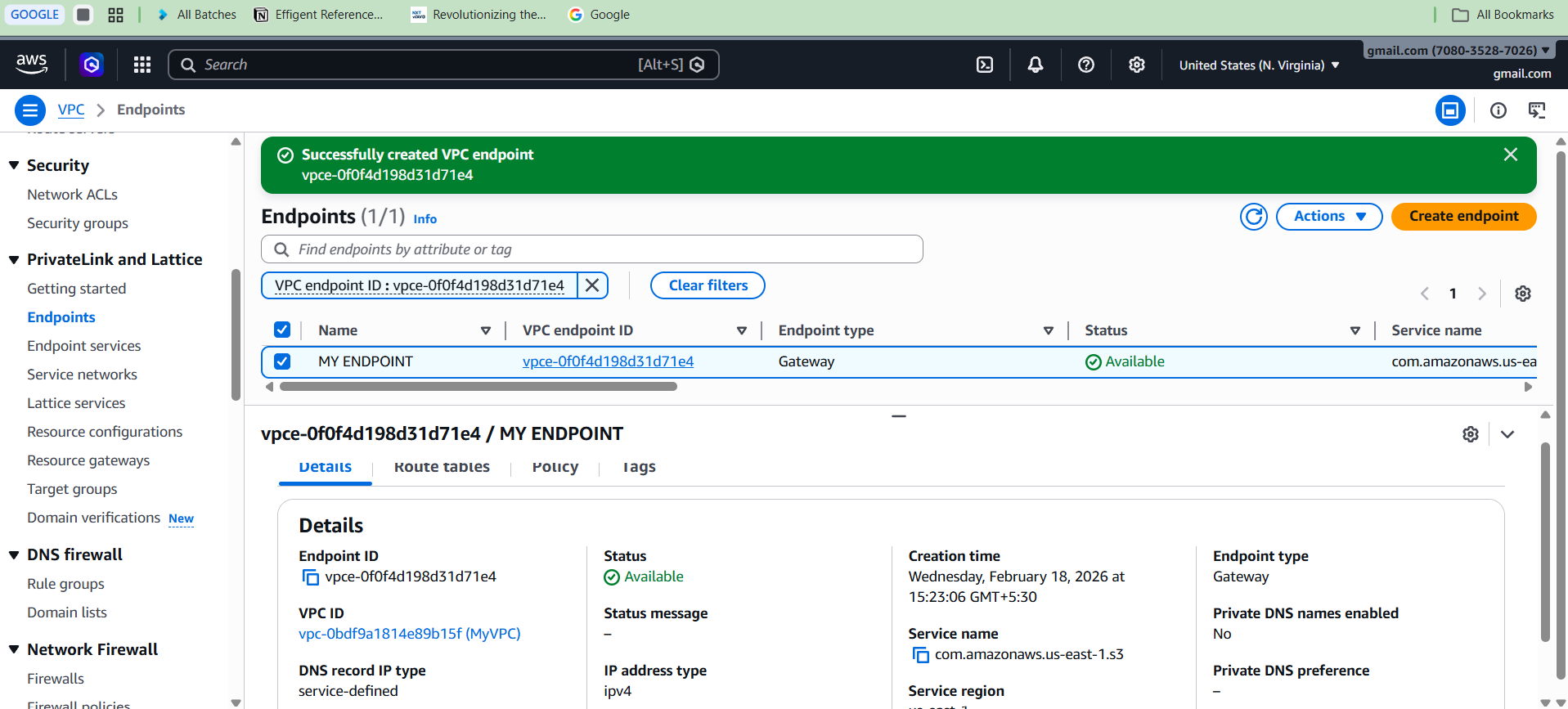
****

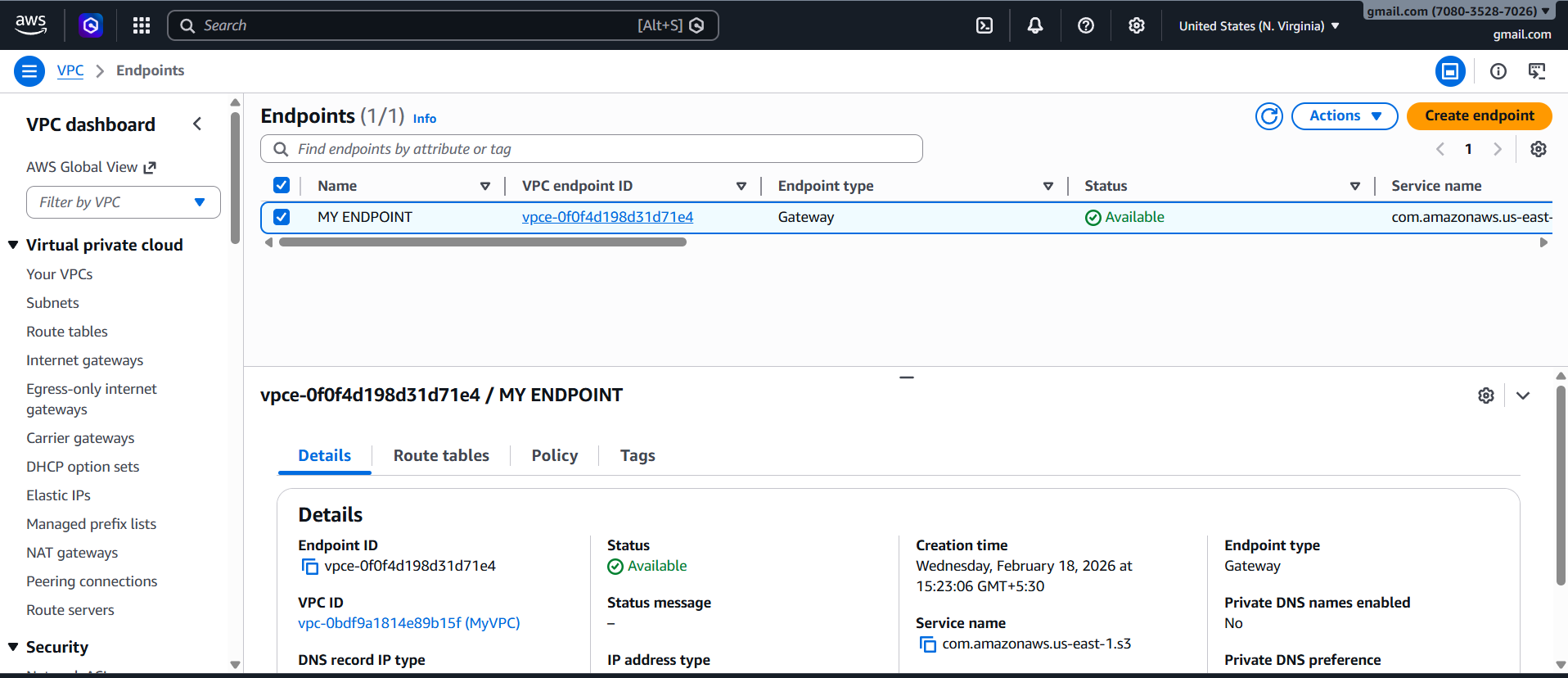
**Conclusion**

Transit Gateway is now successfully connected to your VPC. You can connect multiple VPCs using the same Transit Gateway.

**5: Set up a VPC Endpoint**

Step 1: Go to VPC → Endpoints  
Step 2: Click Create Endpoint  
Step 3: Select AWS Services  
Step 4: Select Service: com.amazonaws.ap-south-1.s3  
Step 5: Select VPC  
Step 6: Select Private Route Table  
Step 7: Click Create Endpoint

****

****

**Result:**  
Private EC2 can access S3 without Internet.