Lab 15: VPC Peering Across Two Regions (ap-south-1 and ap-south-2)

Objective:

Establish a VPC peering connection between two AWS regions and demonstrate communication between EC2 instances via private IP using `curl`. Each instance will serve a webpage using HTTPD.

Estimated Duration: 90 minutes

Prerequisites:

- AWS Free Tier account
- IAM user with VPC and EC2 full access

Part A: Create VPCs in Two Regions

1. In Region: ap-south-1 (Mumbai)

- Go to **VPC > Your VPCs > Create VPC**
    - Name: `VPC-Mumbai`
    - IPv4 CIDR block: **`10.0.0.0/16`**
    - Tenancy: Default
    - Click **Create VPC**

2. In Region: ap-south-2 (Hyderabad)

- Go to **VPC > Your VPCs > Create VPC**
    - Name: `VPC-Hyderabad`
    - IPv4 CIDR block: **`10.1.0.0/16`**
    - Tenancy: Default
    - Click **Create VPC**

Part B: Launch EC2 Instances in Each VPC

1. In ap-south-1 (Mumbai)

- Launch EC2 instance:
    - Name: `Web-Mumbai`

- AMI: Amazon Linux 2
- Instance Type: t2.micro
- Network: `VPC-Mumbai`
- Subnet: any public subnet
- Security Group: allow **SSH (22)** and **HTTP (80)**
- Key pair: create/select a key - optional

2. In ap-south-2 (Hyderabad)

- Launch EC2 instance:
  - Name: `Web-Hyderabad`
  - Same configurations as above, but under `VPC-Hyderabad`

Part C: Install HTTPD and Create Web Pages

Run the following commands on both instances:

```
sudo yum update -y
sudo yum install httpd -y
sudo systemctl start httpd
sudo systemctl enable httpd
echo "<h1>This is <Region> Web Page</h1>" | sudo tee /var/www/html/index.html
```

- Replace `<Region>` with `Mumbai` or `Hyderabad`
- Use `curl http://localhost` to verify

Part D: Create VPC Peering Connection

1. In Region: ap-south-1

- Go to **VPC** > **Peering Connections** > **Create Peering Connection**
  - Name: `Mumbai-to-Hyderabad`
  - VPC Requester: `VPC-Mumbai`
  - VPC Accepter: **Another account or region** > Select `ap-south-2` and `VPC-Hyderabad`
  - Click **Create Peering Connection**

2. Accept the Peering Connection

- Switch to Region: `ap-south-2`
- Go to **VPC** > **Peering Connections**
- Select the pending request and **Accept**

Part E: Update Route Tables

1. In Region: ap-south-1

- Go to **Route Tables** > **Select RT for VPC-Mumbai**
- Edit Routes > Add route:
    - Destination: `10.1.0.0/16`
    - Target: VPC Peering Connection ID

2. In Region: ap-south-2

- Go to **Route Tables** > **Select RT for VPC-Hyderabad**
- Edit Routes > Add route:
    - Destination: `10.0.0.0/16`
    - Target: VPC Peering Connection ID

Part F: Modify Security Groups for Internal Access

On both instances' security groups:

- Inbound Rules:
    - Add custom rule:
        - Type: HTTP
        - Source: **CIDR of peer VPC**

Example:

- For `Web-Mumbai`, add source `10.1.0.0/16`
- For `Web-Hyderabad`, add source `10.0.0.0/16`

Part G: Test Private Connectivity

From Mumbai instance:

```
curl http://<Private-IP-of-Hyderabad-Instance>
```

From Hyderabad instance:

```
curl http://<Private-IP-of-Mumbai-Instance>
```

You should see each other’s webpage HTML content.

Cleanup (Optional)

- Terminate EC2 instances
- Delete VPC peering connection
- Delete custom routes and VPCs

Student Assignment

- Add a third VPC and peer it with both regions
- Use ICMP (ping) to test basic connectivity
- Try automating peering using AWS CLI