VPC Setup

STEP-BY-STEP GUIDE TO SET UP A BASIC AWS VPC

🛠️ **Prerequisites**

- An **AWS account**
- **IAM permissions** to create VPCs, subnets, EC2 instances, and related networking components

---

STEP 1: Create a New VPC

1. Go to the AWS Console → **VPC Dashboard**
2. Click **"Create VPC"**
3. Select **"VPC only"**
4. Enter:
    - **Name**: MyCustomVPC
    - **IPv4 CIDR block**: 10.0.0.0/16
    - Leave IPv6 disabled (for now)
5. Click **Create VPC**

STEP 2: Create Subnets

✅ **2.1 Public Subnet**

1. Go to **Subnets** → **Create subnet**
2. Choose:
    - **VPC**: MyCustomVPC
    - **Availability Zone**: Choose one (e.g., us-east-1a)

- **Name**: PublicSubnet
- **IPv4 CIDR block**: 10.0.1.0/24

3. Click **Create subnet**

✅ **2.2 Private Subnet**

1. Repeat the steps to create another subnet:
   - **Name**: PrivateSubnet
   - **IPv4 CIDR block**: 10.0.2.0/24
2. Click **Create subnet**

STEP 3: Create and Attach an Internet Gateway

1. Go to **Internet Gateways** → **Create Internet Gateway**
2. Name it MyIGW
3. Click **Create**, then **Attach to VPC** → Choose MyCustomVPC

STEP 4: Create a Route Table for the Public Subnet

1. Go to **Route Tables** → **Create route table**
2. Name: PublicRouteTable
3. Select VPC: MyCustomVPC
4. Click **Create**

Add Route to Internet:

1. Select PublicRouteTable → **Routes** tab → **Edit routes**
2. Add route:
   - **Destination**: 0.0.0.0/0
   - **Target**: Internet Gateway → select MyIGW
3. Click **Save**

Associate the Route Table:

1. Go to **Subnet Associations** tab → **Edit subnet associations**
2. Select: PublicSubnet
3. Save

STEP 5: Create NAT Gateway for Private Subnet Access

✅ **5.1 Create Elastic IP**

1. Go to **Elastic IPs** → Allocate a new one

✅ **5.2 Create NAT Gateway**

1. Go to **NAT Gateways** → **Create NAT Gateway**
2. Choose:
    - **Subnet**: PublicSubnet
    - **Elastic IP**: Select the one you just created
3. Click **Create NAT Gateway**

STEP 6: Create Route Table for Private Subnet

1. Go to **Route Tables** → **Create route table**
2. Name: PrivateRouteTable
3. Select VPC: MyCustomVPC
4. Click **Create**

Add NAT Route:

1. Select PrivateRouteTable → **Routes** → **Edit routes**
2. Add route:
    - **Destination**: 0.0.0.0/0
    - **Target**: NAT Gateway → select your NAT Gateway
3. Save

Associate with Private Subnet:

1. Go to **Subnet Associations** → **Edit subnet associations**
2. Select PrivateSubnet

STEP 7: Launch EC2 Instances

✅ **7.1 Public EC2 Instance (Web Server)**

1. Go to **EC2** → Launch Instance
2. Name: WebServer
3. Choose:
    - **Amazon Linux 2 AMI**
    - **Instance Type**: t2.micro (Free tier eligible)
4. **Network settings**:
    - VPC: MyCustomVPC
    - Subnet: PublicSubnet
    - Auto-assign public IP: **Enable**
    - Security group: Allow SSH (port 22) and HTTP (port 80)
5. Launch instance

✅ **7.2 Private EC2 Instance (Backend)**

1. Repeat above, but:
    - Name: AppServer
    - Subnet: PrivateSubnet
    - Auto-assign public IP: **Disable**
    - Security group: Allow SSH only from WebServer's security group (not from internet)

✅ **Final Test**

- SSH into the **WebServer** using its public IP.

- From WebServer, SSH into the **AppServer** using its private IP (test internal communication).
- AppServer can access the internet (e.g., ping google.com) via **NAT Gateway**, but can’t be accessed from the internet directly.