VPC Public Private Subnets

create vpc and 1 public 1 private subnet in us-east1 using terraform

Here's a Terraform script to create a **VPC** with **one public and one private subnet** in the **us-east-1** region. The public subnet will have internet access via an **Internet Gateway (IGW)**, and the private subnet will have **NAT Gateway** for outbound internet access.

Terraform Code for VPC with Public and Private Subnets

```hcl
provider "aws" {
  region = "us-east-1"
}

# VPC
resource "aws_vpc" "my_vpc" {
  cidr_block           = "10.0.0.0/16"
  enable_dns_support   = true
  enable_dns_hostnames = true

  tags = {
    Name = "MyVPC"
  }
}

# Public Subnet
resource "aws_subnet" "public_subnet" {
  vpc_id                  = aws_vpc.my_vpc.id
  cidr_block              = "10.0.1.0/24"
  map_public_ip_on_launch = true
  availability_zone       = "us-east-1a"

  tags = {
    Name = "PublicSubnet"
  }
}

# Private Subnet
resource "aws_subnet" "private_subnet" {
  vpc_id            = aws_vpc.my_vpc.id
  cidr_block        = "10.0.2.0/24"
  availability_zone = "us-east-1b"

  tags = {
    Name = "PrivateSubnet"
  }
}

# Internet Gateway for Public Subnet
resource "aws_internet_gateway" "igw" {
  vpc_id = aws_vpc.my_vpc.id

  tags = {
    Name = "MyIGW"
  }
}

# Public Route Table
resource "aws_route_table" "public_rt" {
  vpc_id = aws_vpc.my_vpc.id

  tags = {
    Name = "PublicRouteTable"
  }
}

# Route for Public Subnet to Internet Gateway
resource "aws_route" "public_internet_access" {
  route_table_id         = aws_route_table.public_rt.id
  destination_cidr_block = "0.0.0.0/0"
  gateway_id             = aws_internet_gateway.igw.id
}

# Associate Public Subnet with Public Route Table
resource "aws_route_table_association" "public_assoc" {
  subnet_id      = aws_subnet.public_subnet.id
  route_table_id = aws_route_table.public_rt.id
}

# Elastic IP for NAT Gateway
resource "aws_eip" "nat_eip" {
  domain = "vpc"
}

# NAT Gateway for Private Subnet
resource "aws_nat_gateway" "nat_gw" {
  allocation_id = aws_eip.nat_eip.id
  subnet_id     = aws_subnet.public_subnet.id

  tags = {
    Name = "MyNATGateway"
  }
}

# Private Route Table
resource "aws_route_table" "private_rt" {
  vpc_id = aws_vpc.my_vpc.id

  tags = {
    Name = "PrivateRouteTable"
  }
}

# Route for Private Subnet to NAT Gateway
resource "aws_route" "private_nat_access" {
  route_table_id         = aws_route_table.private_rt.id
  destination_cidr_block = "0.0.0.0/0"
  nat_gateway_id         = aws_nat_gateway.nat_gw.id
}

# Associate Private Subnet with Private Route Table
resource "aws_route_table_association" "private_assoc" {
  subnet_id      = aws_subnet.private_subnet.id
  route_table_id = aws_route_table.private_rt.id
}
```

Explanation

1. **VPC (10.0.0.0/16)**: Creates a new Virtual Private Cloud.
2. **Public Subnet (10.0.1.0/24)**:
   - Has **map_public_ip_on_launch** set to `` `true` `` for auto-assigning public IPs.
   - Internet access via **Internet Gateway (IGW)**.
3. **Private Subnet (10.0.2.0/24)**:
   - No public IPs assigned.
   - Uses a **NAT Gateway** in the public subnet for internet access.
4. **Internet Gateway**: Enables internet access for resources in the public subnet.

5. **NAT Gateway**: Allows private subnet instances to access the internet securely.
6. **Route Tables**:
    - **Public Subnet** → Internet via IGW.
    - **Private Subnet** → Internet via NAT Gateway.

How to Apply?

```sh
terraform init
terraform plan
terraform apply -auto-approve
```

Let me know if you need modifications! 🚀