Terraform Demo Project by Prasad Pawar

Terraform Project Explanation

This Terraform project provisions a basic AWS infrastructure consisting of:

1. **VPC**: A custom Virtual Private Cloud (VPC) with CIDR block `10.10.0.0/16`.
2. **Subnet**: A public subnet within the VPC with CIDR block `10.10.0.0/24`, mapped to availability zone `ap-south-1a`. It has `map_public_ip_on_launch = true` so that EC2 instances launched within it get public IPs.
3. **Internet Gateway (IGW)**: Allows the subnet to communicate with the internet.
4. **Route Table**: Routes traffic destined for the internet (`0.0.0.0/0`) to the IGW.
5. **Route Table Association**: Associates the route table with the subnet.
6. **Security Group**: Allows inbound HTTP traffic on port 80 and all outbound traffic.
7. **EC2 Instance**: Launches a t2.micro Amazon Linux 2 instance using the specified AMI. The instance installs and runs Apache (httpd), and displays a custom message on the homepage using `user_data`.

This infrastructure allows users to access a basic HTTP server hosted on the EC2 instance from the internet.

Terraform Code:

Terraform Configuration

```
terraform {
  required_providers {
    aws = {
      source = "hashicorp/aws"
      version = "5.99.1"
    }
  }
}
```

Terraform Demo Project by Prasad Pawar

```
provider "aws" {
  access_key = "AKIA..."
  secret_key = "5rUW..."
  region     = "ap-south-1"
}

resource "aws_vpc" "my_vpc" {
  cidr_block = "10.10.0.0/16"
}

resource "aws_subnet" "my-sub" {
  vpc_id                  = aws_vpc.my_vpc.id
  cidr_block              = "10.10.0.0/24"
  availability_zone       = "ap-south-1a"
  map_public_ip_on_launch = true
}

resource "aws_internet_gateway" "gw" {
  vpc_id = aws_vpc.my_vpc.id
}

resource "aws_route_table" "pub_route" {
  vpc_id = aws_vpc.my_vpc.id

  route {
    cidr_block = "0.0.0.0/0"
    gateway_id = aws_internet_gateway.gw.id
  }
}

resource "aws_route_table_association" "a1" {
  route_table_id = aws_route_table.pub_route.id
  subnet_id      = aws_subnet.my-sub.id
}

resource "aws_security_group" "sg" {
  name   = "sg"
  vpc_id = aws_vpc.my_vpc.id

  ingress {
    from_port   = 80
    to_port     = 80
    protocol    = "tcp"
    cidr_blocks = ["0.0.0.0/0"]
  }

  egress {
    from_port   = 0
    to_port     = 0
    protocol    = "-1"
    cidr_blocks = ["0.0.0.0/0"]
  }

  tags = {
    Name = "my-sg"
  }
}

resource "aws_instance" "hello" {
  ami                         = "ami-018046b953a698135"
  instance_type               = "t2.micro"
  subnet_id                   = aws_subnet.my-sub.id
  vpc_security_group_ids      = [aws_security_group.sg.id]
  associate_public_ip_address = true

  user_data = <<EOF
#!/bin/bash
yum update -y
yum install -y httpd
systemctl start httpd
systemctl enable httpd
echo "<h1> Hello Prasad </h1>" > /var/www/html/index.html
EOF
}
```