### Activity 2: Creating a VPC Using Terraform

**Objective**: The objective of this activity is to introduce students to Terraform by creating an AWS Virtual Private Cloud (VPC) with public and private subnets. Students will also create an Internet Gateway (IGW) and configure routing for the public subnet.

# Instructions:

1. **Prerequisites**: Ensure you have access to an AWS account with the necessary permissions to create VPCs, subnets, route tables, and Internet Gateways. For that, create an IAM user with the policy mentioned in the file iam_terraform_policy.json present in the labDirectory folder. Keep the **Access Key ID** and **Secret Access Key** of the IAM user for future reference.

2. **Specification**:

a. VPC Specification:

- Create a VPC with a CIDR block **10.0.0.0/16**

b. Subnet Specification:

- Create a public subnet within the VPC with CIDR block **10.0.1.0/24**. Availability Zone should be **us-east-1b**
- Create a private subnet within the VPC with CIDR block **10.0.2.0/24**, Availability Zone should be **us-east-1b**

c. Internet Gateway:

- Create an Internet Gateway and attach it to the VPC.

d. Route Table Configuration:

- Create a route table for the public subnet.
- Add a route to direct all external traffic **(0.0.0.0/0)** to the Internet Gateway.
- Associate the public subnet with this route table.

3. **Activity Setup:**

Students need to write the content for the following Terraform files:

a. main.tf: Define the VPC, subnets, Internet Gateway, and route table.

b. terraform.tfvars: Assign values to the variables defined in variables.tf.

c. outputs.tf: Define outputs to display the VPC ID, subnet IDs, Internet Gateway ID, and route table ID after deployment. For

A. VPC ID use variable name “"vpc_id"”
B. Public Subnet ID use variable name “public_subnet_id”
C. Private Subnet ID use variable name “private_subnet_id”
D. Internet Gateway ID use variable name “internet_gateway_id”
E. Routing Table ID use variable name “public_route_table_id”

d. provider.tf: Configure the AWS provider with the appropriate region and credentials.

e. variables.tf: Define all the variables required for the setup. For

A. Access key ID use variable name “access_key_value”
B. Secret Access Key use variable name “secret_key_value”
C. Region use variable name “region_value”
D. CIDR block of VPC use “vpc_cidr_block”
E. CIDR block of Public Subnet use “public_subnet_cidr_block”
F. CIDR block of Private Subnet use “private_subnet_cidr_block”
G. For Availability Zone use “availability_zone”

**Steps to Complete the Lab**:

a. Initialize Terraform: Run the command **terraform init** in your project directory to initialize Terraform.

b. Plan the Terraform deployment: Execute **terraform plan** to preview the changes that Terraform will make.

c. Apply the Terraform deployment: Run **terraform apply** and confirm the action when prompted. This will create the VPC, subnets, Internet Gateway, and route table.

**Verification**: Once the VPC is created, verify the following:

- Go to the AWS Management Console and navigate to the VPC dashboard.
- Check the created VPC, subnets, Internet Gateway, and route table.
- Ensure that the public subnet is associated with the correct route table.

**Clean-Up**: To avoid incurring charges after completing the lab:

- Run **terraform destroy** in your project directory and confirm the action.

**Why Terraform when we have CloudFront in AWS?**

**CloudFront** : There is no built-in tracking of resources or their relationships. If you forget a resource you created, it could remain active and incur charges unnecessarily.

**Understanding Terraform State:**

Terraform maintains a state file (`terraform.tfstate`) that tracks all resources it creates, their configurations, and dependencies. This allows Terraform to identify and manage changes (e.g., updates, deletions) efficiently.

One such example is **terraform destroy**. When you run **terraform destroy**, Terraform references this state file to identify the resources it created and deletes them accordingly.

You can verify the deletion of the Virtual Private Cloud (VPC) with public and private subnets, Internet Gateway (IGW) and Routing Tables by checking the AWS Management Console after running **terraform destroy**.

The state file is critical because it acts as the source of truth for Terraform to track and manage resources efficiently.

**Evaluation**: After **terraform destroy**, click on Evaluate. It may take 1-2 minutes for evaluation.