**AWS Vpc Creation in Terraform**

**Create terraform template for VPC Creation with following components**

1. EC2 - ami-024e6efaf93d85776
2. region - us-east-2
3. Security Group - rtp03-sg [ inbound : 22,80 outbound : all ]
4. Public Subnet - rtp03-public\_subent\_01
5. Internet gateway - rtp03-igw
6. Route Table - rtp03-public-rt
7. VPC id - aws\_vpc.rtp03-vpc.id

This Terraform script is used to create infrastructure resources on Amazon Web Services (AWS) using the Infrastructure as Code (IaC) approach. It provides an AWS EC2 instance, VPC, subnet, security group, internet gateway, route table, and associates resources together.

**Source Code - vpc.tf**

**//Provider Name and EC2 resource creation**

provider "aws" {

region = "us-east-2"

profile = "default"

}

resource "aws\_instance" "ec2" {

ami = "ami-024e6efaf93d85776"

instance\_type = "t2.micro"

key\_name = "prd01"

//vpc\_security\_group\_ids = ["${aws\_security\_group.rtp03-sg.id}"]

subnet\_id = "${aws\_subnet.rtp03-public\_subent\_01.id}"

}

resource "aws\_security\_group" "rtp03-sg" {

name = "rtp03-sg"

vpc\_id = "${aws\_vpc.rtp03-vpc.id}"

ingress {

from\_port = 22

to\_port = 22

protocol = "tcp"

cidr\_blocks = ["0.0.0.0/0"]

}

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 = "ssh-sg"

}

}

**//creating a VPC**

resource "aws\_vpc" "rtp03-vpc" {

cidr\_block = "10.1.0.0/16"

tags = {

Name = "rpt03-vpc"

}

}

**// Creating a Subnet**

resource "aws\_subnet" "rtp03-public\_subent\_01" {

vpc\_id = “${aws\_vpc.rtp03-vpc.id}”

cidr\_block = "10.1.1.0/24"

map\_public\_ip\_on\_launch = "true"

availability\_zone = "us-east-2a"

tags = {

Name = "rtp03-public\_subent\_01"

}

}

//**Creating a Internet Gateway**

resource "aws\_internet\_gateway" "rtp03-igw" {

vpc\_id = "${aws\_vpc.rtp03-vpc.id}"

tags = {

Name = "rtp03-igw"

}

}

// **Create a route table**

resource "aws\_route\_table" "rtp03-public-rt" {

vpc\_id = "${aws\_vpc.rtp03-vpc.id}"

route {

cidr\_block = "0.0.0.0/0"

gateway\_id = "${aws\_internet\_gateway.rtp03-igw.id}"

}

tags = {

Name = "rtp03-public-rt"

}

}

**// Associate subnet with routetable**

resource "aws\_route\_table\_association" "rtp03-rta-public-subent-1" {

subnet\_id = "${aws\_subnet.rtp03-public\_subent\_01.id}"

route\_table\_id = "${aws\_route\_table.rtp03-public-rt.id}"

}

—-----------

Vpc-2

Main.tf, vars.tf, output.tf

Vi Main.tf

# main.tf

# Define AWS as the provider

provider "aws" {

region = "us-east-1" # Replace with your desired region

}

# VPC

resource "aws\_vpc" "my\_vpc" {

cidr\_block = "10.0.0.0/16" # Replace with your desired CIDR block

tags = {

Name = "MyVPC"

}

}

# Public Subnet

resource "aws\_subnet" "public\_subnet" {

vpc\_id = aws\_vpc.my\_vpc.id

cidr\_block = var.public\_subnet\_cidr\_block # Replace with your desired CIDR block for the public subnet

availability\_zone = "us-east-1a" # Replace with your desired availability zone

map\_public\_ip\_on\_launch = true

tags = {

Name = "PublicSubnet"

}

}

# Private Subnet

resource "aws\_subnet" "private\_subnet" {

vpc\_id = aws\_vpc.my\_vpc.id

cidr\_block = "10.0.2.0/24" # Replace with your desired CIDR block for the private subnet

availability\_zone = "us-east-1b" # Replace with your desired availability zone

tags = {

Name = "PrivateSubnet"

}

}

# Internet Gateway

resource "aws\_internet\_gateway" "my\_igw" {

vpc\_id = aws\_vpc.my\_vpc.id

tags = {

Name = "MyInternetGateway"

}

}

# Route Table for Public Subnet

resource "aws\_route\_table" "public\_route\_table" {

vpc\_id = aws\_vpc.my\_vpc.id

route {

cidr\_block = "0.0.0.0/0"

gateway\_id = aws\_internet\_gateway.my\_igw.id

}

tags = {

Name = "PublicRouteTable"

}

}

# Associate Public Subnet with Public Route Table

resource "aws\_route\_table\_association" "public\_subnet\_association" {

subnet\_id = aws\_subnet.public\_subnet.id

route\_table\_id = aws\_route\_table.public\_route\_table.id

}

# NAT Gateway for Private Subnet

resource "aws\_nat\_gateway" "my\_nat\_gateway" {

allocation\_id = aws\_eip.my\_eip.id

subnet\_id = aws\_subnet.public\_subnet.id

tags = {

Name = "MyNATGateway"

}

}

output.tf

# Output VPC ID

output "vpc\_id" {

value = aws\_vpc.my\_vpc.id

}

# Output Public Subnet ID

output "public\_subnet\_id" {

value = aws\_subnet.public\_subnet.id

}

# Output Private Subnet ID

output "private\_subnet\_id" {

value = aws\_subnet.private\_subnet.id

}

Vars.tf

variable “public\_subnet\_cidr\_block” {

Description = “CIDR block for public subnet”

Default = “10.0.1.0/24”

}