

CLOUD DEVOPS (EPAM) 2022-23 EVEN SEMESTER

LAB3

Name: D V D Bhargava

ID:2000031248

Q) Provision an apache web server in AWS using Terraform

Here we need to install an apache web server in an EC2 instance using terraform. Copy the given below code in main.tf file.

```
provider "aws" {
  region = "us-east-2"
  access_key = "*****"
  secret_key = "*****"
}

resource "aws_instance" "web_server" {
  ami = "ami-05bfbece1ed5beb54"
  instance_type = "t2.micro"
  tags = {
    Name = "Apache Web Server1"
  }
  user_data = <<-EOF
    #!/bin/bash
    echo "*** Installing apache2"
    sudo apt update -y
    sudo apt install apache2 -y
    echo "*** Completed Installing apache2"
  EOF
}

resource "aws_security_group" "web_server_sg" {
  name = "web_server_sg1"
  description = "Allow HTTP traffic"
  ingress {
    from_port = 80
    to_port = 80
    protocol = "tcp"
    cidr_blocks = ["0.0.0.0/0"]
  }
  ingress {
    description = "SSH from VPC"
    from_port = 22
    to_port = 22
    protocol = "tcp"
    cidr_blocks = ["0.0.0.0/0"]
  }
}
```

```

    ipv6_cidr_blocks = [ "::/0" ]
  }

  egress {
    from_port      = 0
    to_port        = 0
    protocol       = "-1"
    cidr_blocks    = [ "0.0.0.0/0" ]
    ipv6_cidr_blocks = [ "::/0" ]
  }
}

resource "aws_ebs_volume" "web_server_data" {
  availability_zone = "${data.aws_availability_zones.available.names[0]}"
  size = 10
}

resource "aws_volume_attachment" "web_server_data_attachment" {
  device_name = "/dev/xvdf"
  volume_id   = "${aws_ebs_volume.web_server_data.id}"
  instance_id = "${aws_instance.web_server.id}"
}

data "aws_availability_zones" "available" {}

```

Here the userdata will install the apache server while creating the ec2 instance

Run the file using Terraform init,plan,apply commands.

The screenshot shows the AWS Management Console interface for an EC2 instance. The left sidebar contains navigation links for EC2 Dashboard, EC2 Global View, Events, Tags, Limits, Instances, Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Scheduled Instances, Capacity Reservations, Images, AMIs, AMI Catalog, and Elastic Block Store. The main content area displays the 'Instance summary for i-0f081be8f1ace1f2a (Apache Web Server)'. The instance is in a 'Running' state. The console shows various configuration details including IP addresses, DNS names, instance type (t2.micro), VPC ID, and Subnet ID. The 'Instance details' tab is selected, showing the platform as 'Amazon Linux (Inferred)' and the AMI ID as 'ami-06e85d4c3149db26a'.

Instance summary for i-0f081be8f1ace1f2a (Apache Web Server)		
Instance ID i-0f081be8f1ace1f2a (Apache Web Server)	Public IPv4 address 35.161.220.208 open address	Private IPv4 addresses 172.31.30.5
IPv6 address -	Instance state Running	Public IPv4 DNS ec2-35-161-220-208.us-west-2.compute.amazonaws.com open address
Hostname type IP name: ip-172-31-30-5-us-west-2.compute.internal	Private IP DNS name (IPv4 only) ip-172-31-30-5-us-west-2.compute.internal	Elastic IP addresses -
Answer private resource DNS name -	Instance type t2.micro	AWS Compute Optimizer finding Opt-in to AWS Compute Optimizer for recommendations. Learn more
Auto-assigned IP address 35.161.220.208 [Public IP]	VPC ID vpc-00ee90cb0240ed9fe	Auto Scaling Group name -
IAM Role -	Subnet ID subnet-06ee9421cf338242a	

Instance details		
Platform Amazon Linux (Inferred)	AMI ID ami-06e85d4c3149db26a	Monitoring disabled