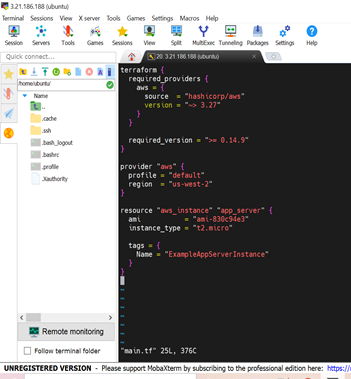
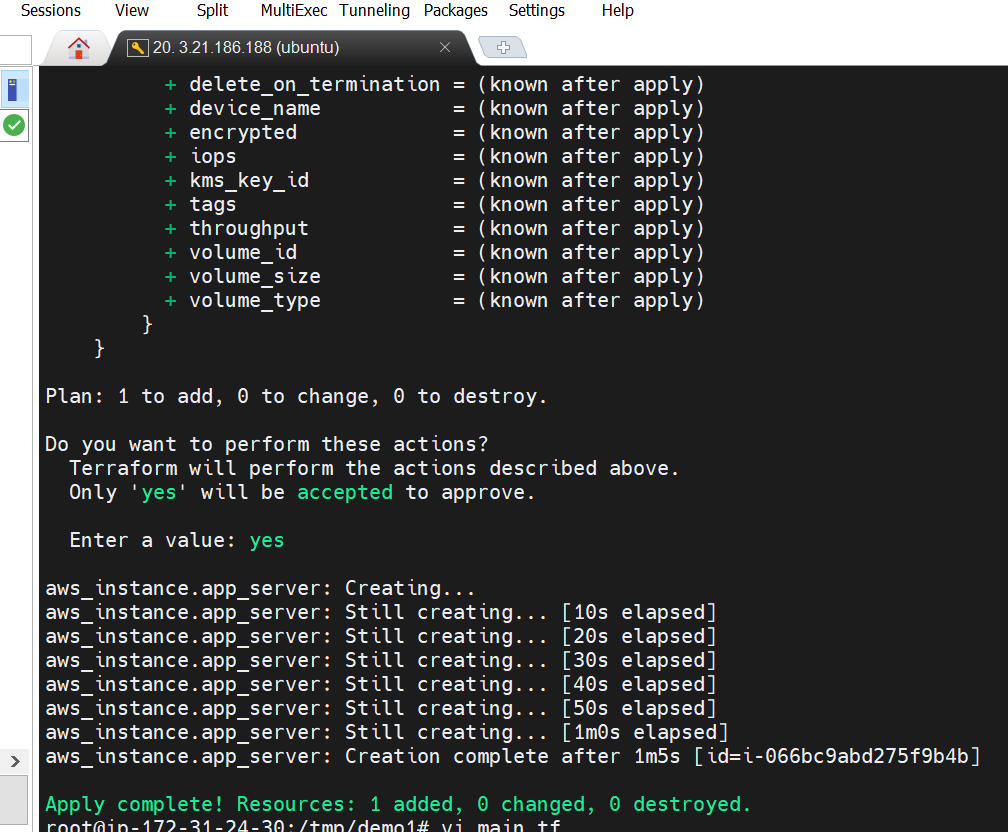
**Terraform Assignments**

🡪Create a VPC with required document components using Terraform

Deploy an EC2 instance inside VPC





🡪Create a script to install apache2

Run the script on the newly created instance

Print the IP address

**Main.tf**

terraform {

required\_providers {

aws = {

source = "hashicorp/aws"

version = "~> 3.0"

}

}

}

# Configure the AWS Provider

provider "aws" {

region = var.region

shared\_credentials\_file = "var.creds\_path.var.creds\_file"

profile = "default"

}

# Create a VPC

resource "aws\_vpc" "app\_vpc" {

cidr\_block = var.vpc\_cidr

tags = {

Name = "app-vpc"

}

}

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

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

tags = {

Name = "vpc\_igw"

}

}

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

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

cidr\_block = var.public\_subnet\_cidr

map\_public\_ip\_on\_launch = true

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

tags = {

Name = "public-subnet"

}

}

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

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

route {

cidr\_block = "0.0.0.0/0"

gateway\_id = aws\_internet\_gateway.igw.id

}

tags = {

Name = "public\_rt"

}

}

resource "aws\_route\_table\_association" "public\_rt\_asso" {

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

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

}

resource "aws\_instance" "web" {

ami = "ami-005e54dee72cc1d00"

instance\_type = var.instance\_type

key\_name = var.instance\_key

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

security\_groups = [aws\_security\_group.sg.id]

user\_data = <<-EOF

#!/bin/bash

echo "\*\*\* Installing apache2"

sudo apt update -y

sudo apt install apache2 -y

echo "\*\*\* Completed Installing apache2"

EOF

tags = {

Name = "web\_instance"

}

volume\_tags = {

Name = "web\_instance"

}

}

