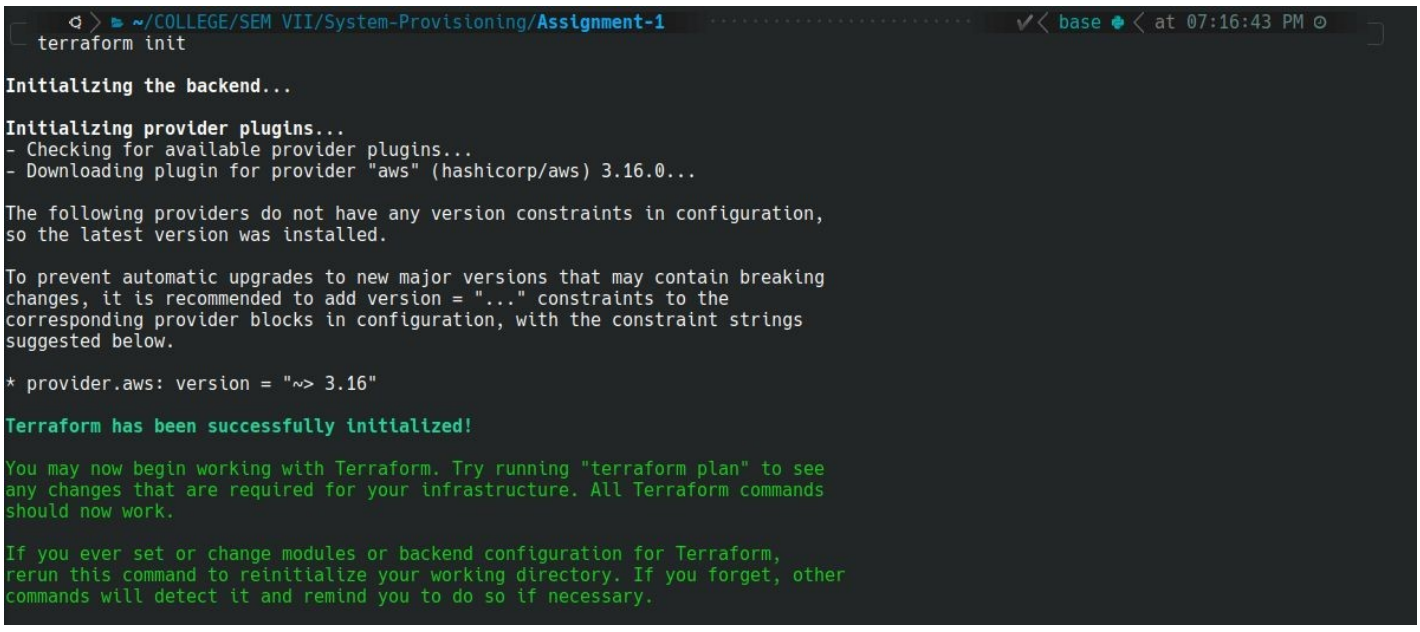
SYSTEM PROVISIONING AND CONFIGURATION MANAGEMENT
(ASSIGNMENT 1)
Prajjawal Banati

Terraform scripts to perform following tasks on AWS cloud Platform
1. Creating two T2 micro ec2 instances
2. Creating a VPN on AWS
3. Creating a S3 bucket

What is Terraform?
Terraform is an open source tool for infrastructure provisioning created by HashiCorp. It provides Infrastructure as code allowing you to automate and manage your infrastructure, platform and your services that run on the platform. Terraform can manage existing and popular service providers(aws, azure, GCP etc).
You do not have to prepare infrastructure like private network space, ec2 server instances, installing docker and other tools and security. Terraform does all that for you by preparing the whole infrastructure using terraform scripts. Thus, it is a software tool that provides Infrastructure as code. Terraform is declarative which means you define what you want.

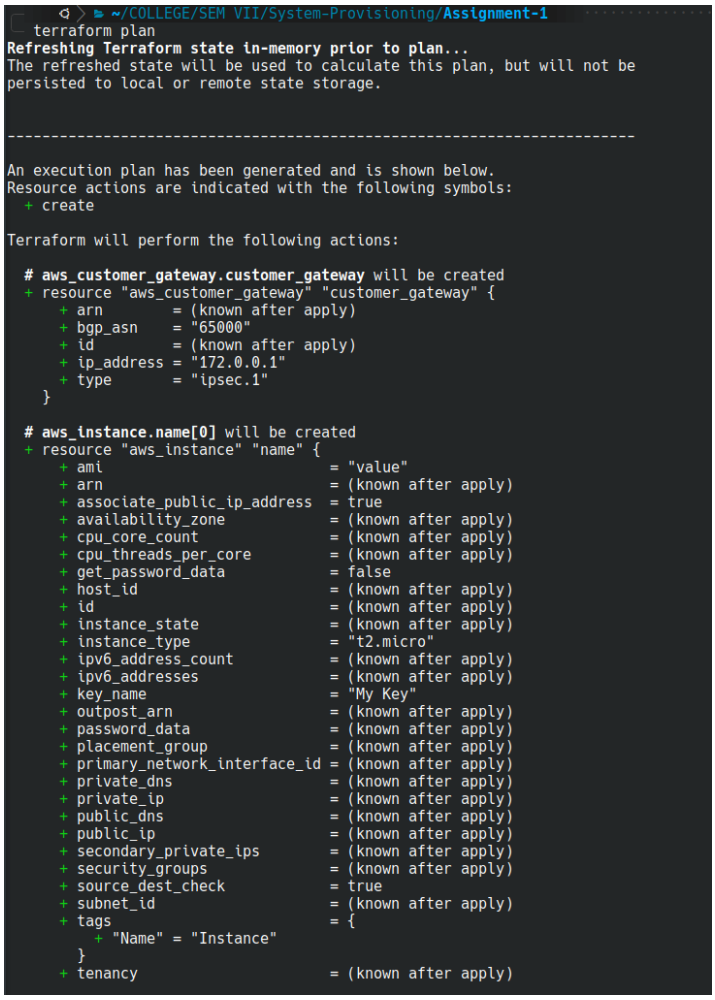
Run the following commands

 # terraform init
Initializes working directory containing terraform configuration files. It is safe to run this command multiple times.

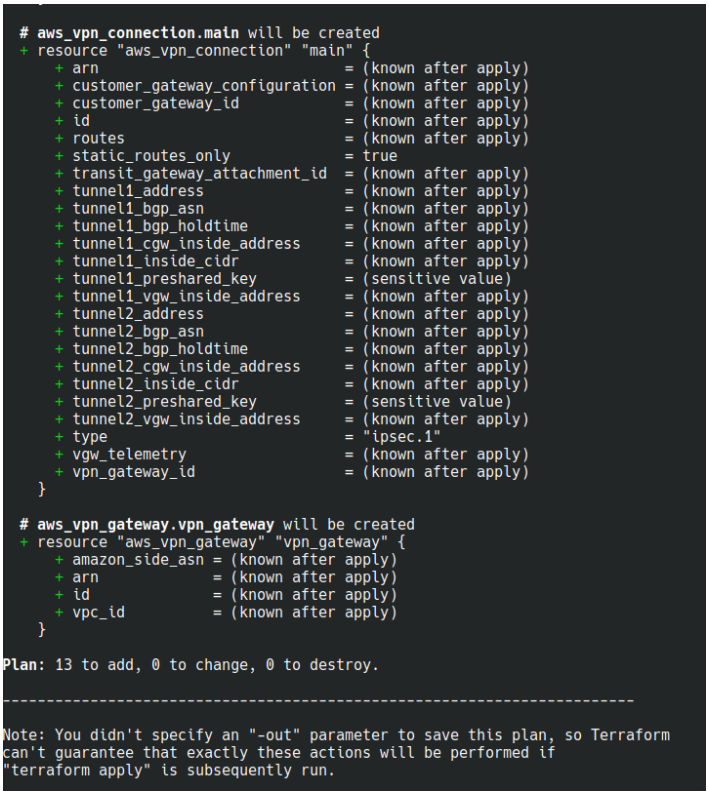


 # terraform plan

To create execution plan that helps you check whether execution plan matches your expectations.



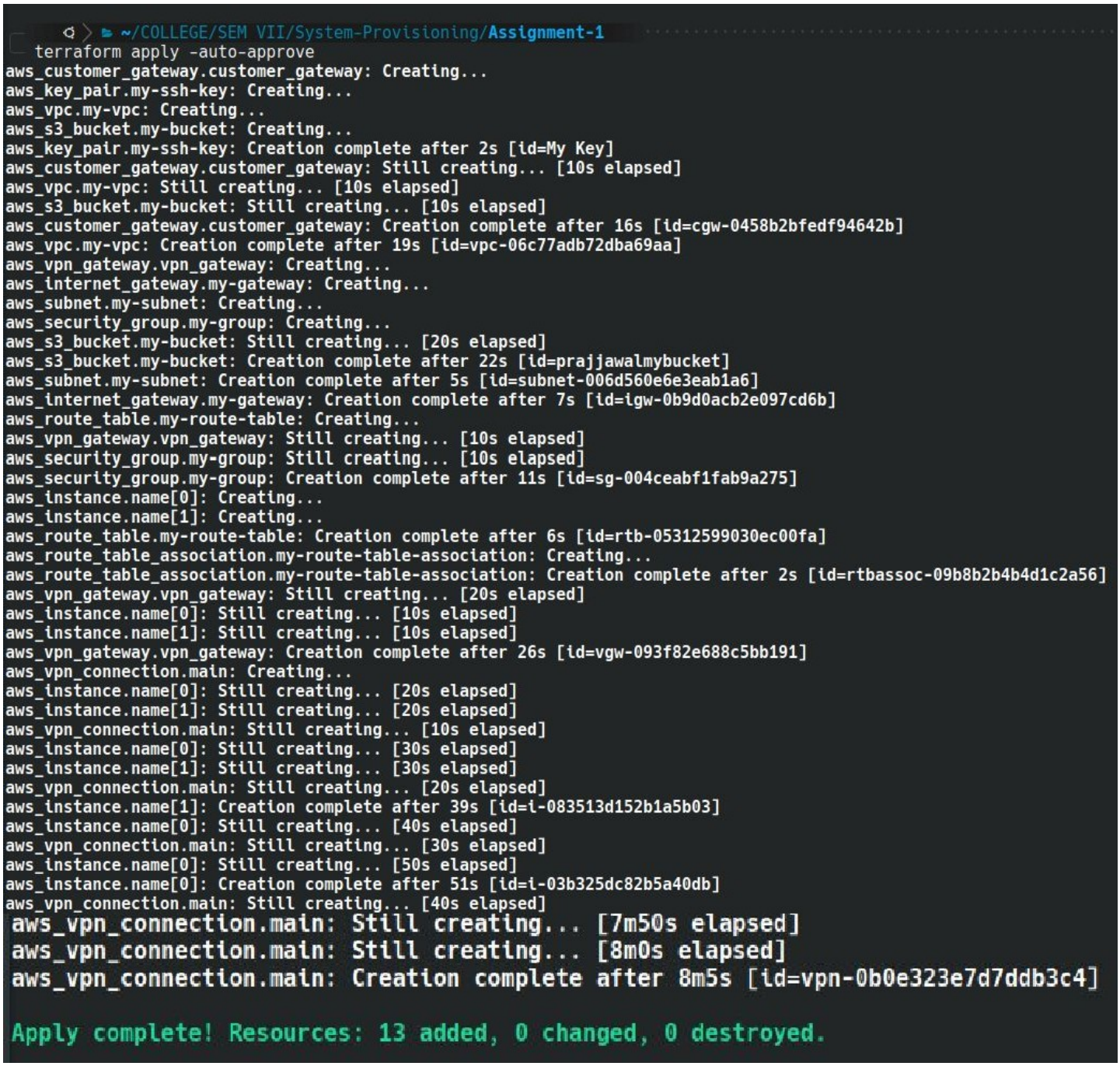
```
~/COLLEGE/SEM VII/System-Provisioning/Assignment-1
terraform plan
Refreshing Terraform state in-memory prior to plan...
The refreshed state will be used to calculate this plan, but will not be
persisted to local or remote state storage.


------------------------------------------------------------------------

An execution plan has been generated and is shown below.
Resource actions are indicated with the following symbols:
  + create

Terraform will perform the following actions:

  # aws_customer_gateway.customer_gateway will be created
  + resource "aws_customer_gateway" "customer_gateway" {
      + arn        = (known after apply)
      + bgp_asn    = "65000"
      + id         = (known after apply)
      + ip_address = "172.0.0.1"
      + type       = "ipsec.1"
    }

  # aws_instance.name[0] will be created
  + resource "aws_instance" "name" {
      + ami                          = "value"
      + arn                          = (known after apply)
      + associate_public_ip_address  = true
      + availability_zone            = (known after apply)
      + cpu_core_count               = (known after apply)
      + cpu_threads_per_core         = (known after apply)
      + get_password_data            = false
      + host_id                      = (known after apply)
      + id                           = (known after apply)
      + instance_state               = (known after apply)
      + instance_type                = "t2.micro"
      + ipv6_address_count           = (known after apply)
      + ipv6_addresses               = (known after apply)
      + key_name                     = "My Key"
      + outpost_arn                  = (known after apply)
      + password_data                = (known after apply)
      + placement_group              = (known after apply)
      + primary_network_interface_id = (known after apply)
      + private_dns                  = (known after apply)
      + private_ip                   = (known after apply)
      + public_dns                   = (known after apply)
      + public_ip                    = (known after apply)
      + secondary_private_ips        = (known after apply)
      + security_groups              = (known after apply)
      + source_dest_check            = true
      + subnet_id                    = (known after apply)
      + tags                         = {
          + "Name" = "Instance"
        }
      + tenancy                      = (known after apply)
```



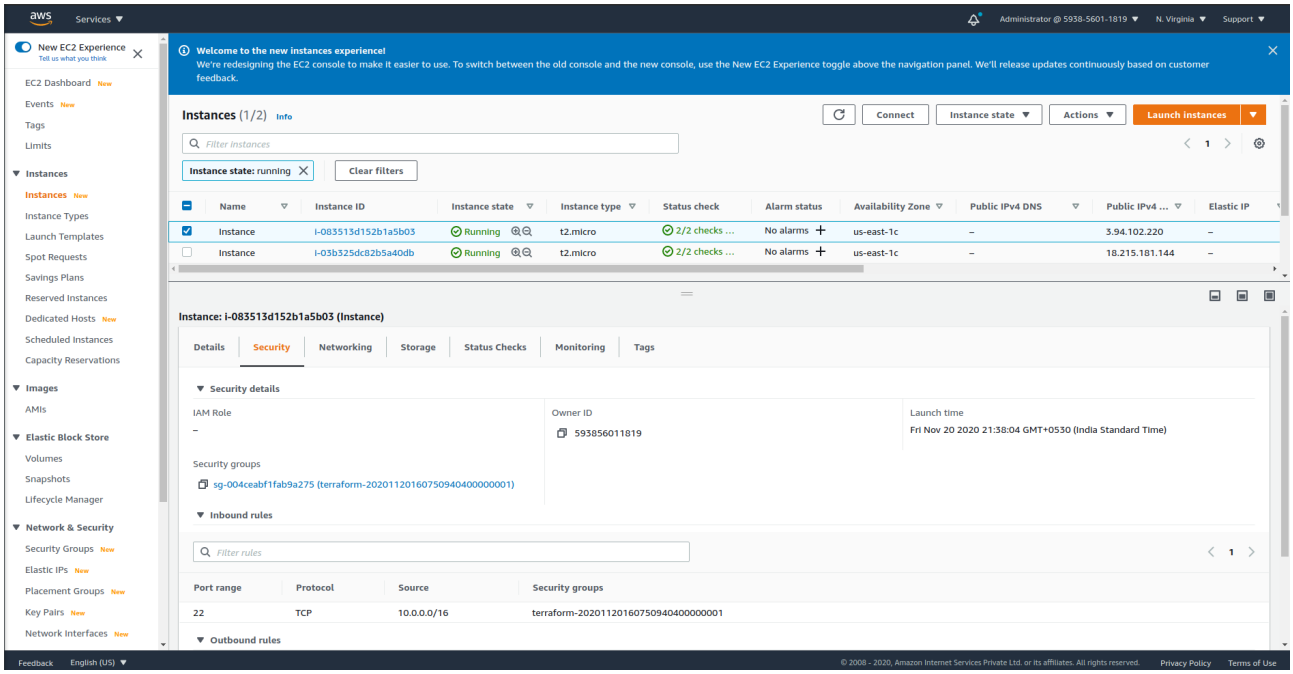
```
  # aws_vpn_connection.main will be created
  + resource "aws_vpn_connection" "main" {
      + arn                           = (known after apply)
      + customer_gateway_configuration = (known after apply)
      + customer_gateway_id           = (known after apply)
      + id                            = (known after apply)
      + routes                        = (known after apply)
      + static_routes_only            = true
      + transit_gateway_attachment_id  = (known after apply)
      + tunnel1_address               = (known after apply)
      + tunnel1_bgp_asn               = (known after apply)
      + tunnel1_bgp_holdtime          = (known after apply)
      + tunnel1_cgw_inside_address    = (known after apply)
      + tunnel1_inside_cidr           = (known after apply)
      + tunnel1_preshared_key         = (sensitive value)
      + tunnel1_vgw_inside_address    = (known after apply)
      + tunnel2_address               = (known after apply)
      + tunnel2_bgp_asn               = (known after apply)
      + tunnel2_bgp_holdtime          = (known after apply)
      + tunnel2_cgw_inside_address    = (known after apply)
      + tunnel2_inside_cidr           = (known after apply)
      + tunnel2_preshared_key         = (sensitive value)
      + tunnel2_vgw_inside_address    = (known after apply)
      + type                          = "ipsec.1"
      + vgw_telemetry                 = (known after apply)
      + vpn_gateway_id                = (known after apply)
    }

  # aws_vpn_gateway.vpn_gateway will be created
  + resource "aws_vpn_gateway" "vpn_gateway" {
      + amazon_side_asn = (known after apply)
      + arn             = (known after apply)
      + id              = (known after apply)
      + vpc_id          = (known after apply)
    }

Plan: 13 to add, 0 to change, 0 to destroy.

------------------------------------------------------------------------

Note: You didn't specify an "-out" parameter to save this plan, so Terraform
can't guarantee that exactly these actions will be performed if
"terraform apply" is subsequently run.
```

 # terraform apply -auto-approve

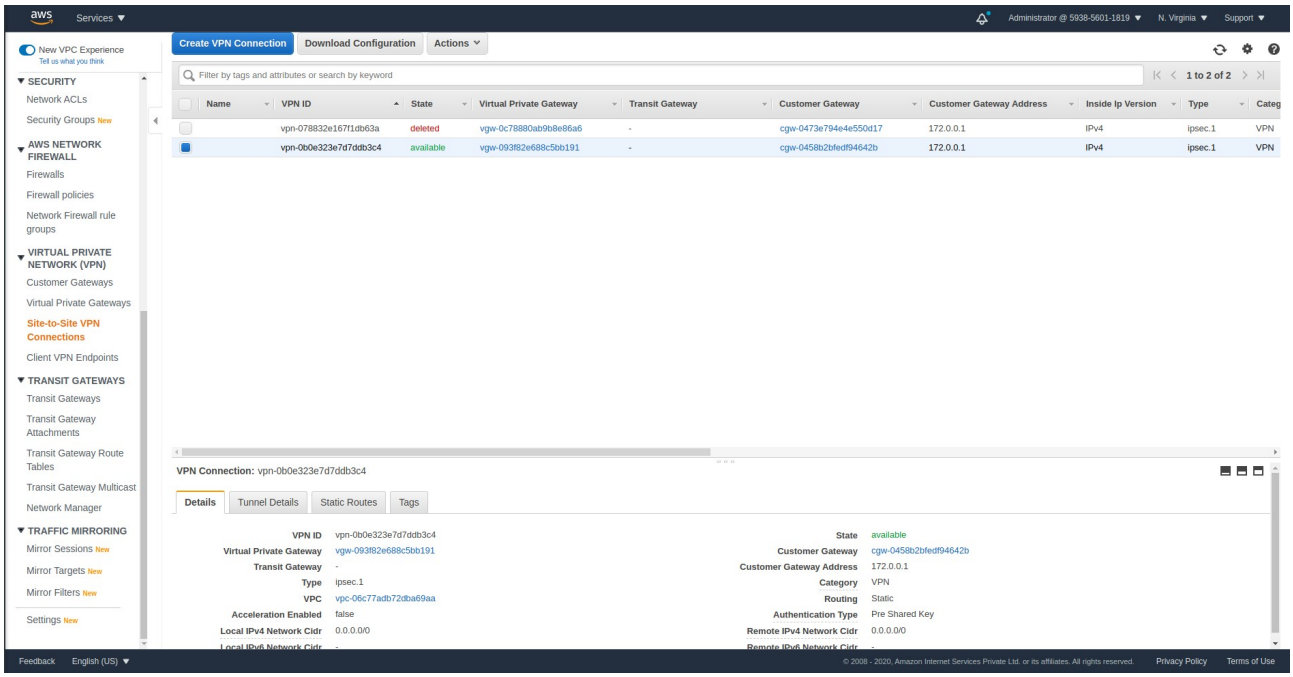
To apply the changes to reach the desired state of the configuration



Now you can check the instances, VPN and S3 bucket have been created on your AWS cloud.Both t2-micro ec2-instances are created in N. Virginia region.



VPN



Prajjawal Banati R171217044 B2

S3 Bucket