

# ASSIGNMENT-1

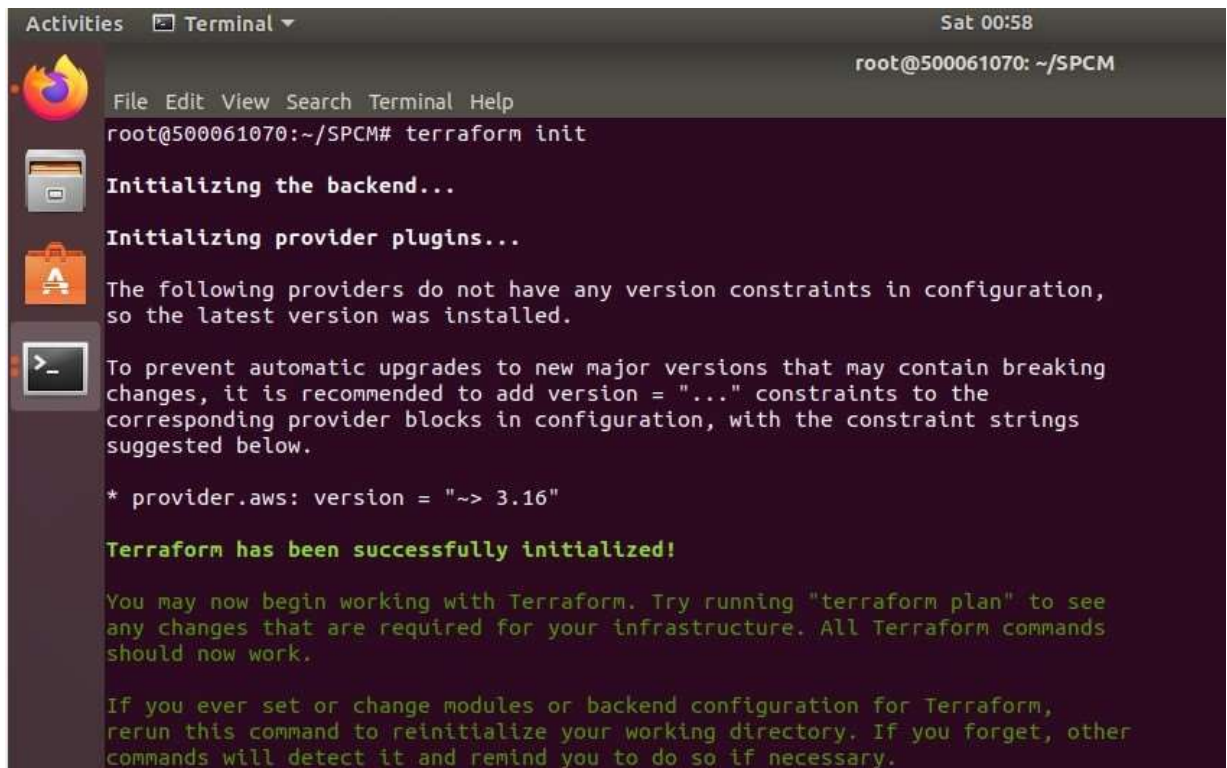
## System Provisioning & Configuration Management

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

STEPS:

- 1) Initialize Terraform in an Empty directory



```
Activities Terminal Sat 00:58
root@500061070: ~/SPCM
File Edit View Search Terminal Help
root@500061070:~/SPCM# terraform init
Initializing the backend...
Initializing provider plugins...
The following providers do not have any version constraints in configuration,
so the latest version was installed.

To prevent automatic upgrades to new major versions that may contain breaking
changes, it is recommended to add version = "..." constraints to the
corresponding provider blocks in configuration, with the constraint strings
suggested below.

* provider.aws: version = "~> 3.16"

Terraform has been successfully initialized!

You may now begin working with Terraform. Try running "terraform plan" to see
any changes that are required for your infrastructure. All Terraform commands
should now work.

If you ever set or change modules or backend configuration for Terraform,
rerun this command to reinitialize your working directory. If you forget, other
commands will detect it and remind you to do so if necessary.
```

## 2) Write the Script in main.tf

### Script:-

```
provider "aws" {  
  region= "ap-south-1"  
  access_key="AKIA5R5X36NAOL7XF74X"  
  secret_key=" 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 "  
}
```

```
resource "aws_instance" "RishabhInstance" {  
  ami      = "ami-0db0b3ab7df22e366"  
  count=2  
  key_name = "SPCM"  
  instance_type = "t2.micro"  
  security_groups= [ "secgp"]  
  tags= {  
    Name = "secgp"  
  }  
}
```

```
resource "aws_s3_bucket" "rishabhs3" {  
  bucket = "rishabhs3"  
}
```

```
resource "aws_vpc" "vpc" {
```

```
    cidr_block = "10.0.0.0/16"
}
```

```
resource "aws_vpn_gateway" "vpn_gateway" {
    vpc_id = aws_vpc.vpc.id
}
```

```
resource "aws_customer_gateway" "customer_gateway" {
    bgp_asn    = 65000
    ip_address = "172.0.0.1"
    type       = "ipsec.1"
}
```

```
resource "aws_vpn_connection" "main" {
    vpn_gateway_id      = aws_vpn_gateway.vpn_gateway.id
    customer_gateway_id = aws_customer_gateway.customer_gateway.id
    type                = "ipsec.1"
    static_routes_only = true
}
```

```
resource "aws_security_group" "secgp" {
    name        = "secgp"
    description = "none"

    ingress {
```

```
from_port = 8080
to_port   = 8080
protocol  = "tcp"
cidr_blocks = ["0.0.0.0/0"]
}

ingress {
    from_port = 22
    to_port   = 22
    protocol  = "tcp"
    cidr_blocks = ["0.0.0.0/0"]
}

egress {
    from_port = 0
    to_port   = 65535
    protocol  = "tcp"
    cidr_blocks = ["0.0.0.0/0"]
}

tags= {
    Name = "secgp"
}
}
```

### 3) Terraform plan

```
root@500061070:~/SPCM# 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.rishabhInstance[0] will be created
+ resource "aws_instance" "rishabhInstance" {
  + ami                  = "ami-0db0b3ab7df22e366"
  + arn                  = (known after apply)
  + associate_public_ip_address = (known after apply)
  + availability_zone     = (known after apply)
  + cpu_core_count       = (known after apply)
  + cpu_threads_per_core = (known after apply)
  + get_password_data     = false
}
```

#### 4) Terraform Apply

```

root@500061070:~/SPCM# terraform apply

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.rishabhInstance[0] will be created
+ resource "aws_instance" "rishabhInstance" {
  + ami                  = "ami-0db0b3ab7df22e366"
  + arn                  = (known after apply)
  + associate_public_ip_address = (known after apply)
  + 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             = "SPCM"
}

Plan: 8 to add, 0 to change, 0 to destroy.

Do you want to perform these actions?
  Terraform will perform the actions described above.
  Only 'yes' will be accepted to approve.

Enter a value: yes

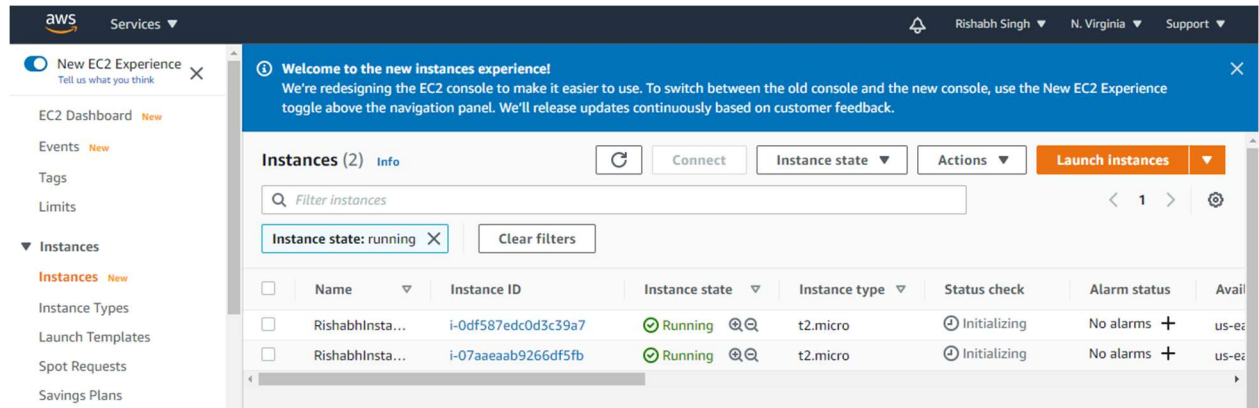
aws_instance.rishabhInstance[0]: Creating...
aws_customer_gateway.customer_gateway: Creating...
aws_vpc.vpc: Creating...
aws_instance.rishabhInstance[1]: Creating...
aws_s3_bucket.tf_course: Creating...
aws_security_group.secgp: Creating...
aws_vpc.vpc: Still creating... [10s elapsed]
aws_security_group.secgp: Still creating... [10s elapsed]

```



## 5) Check the Created Services

### EC2 Instances: -



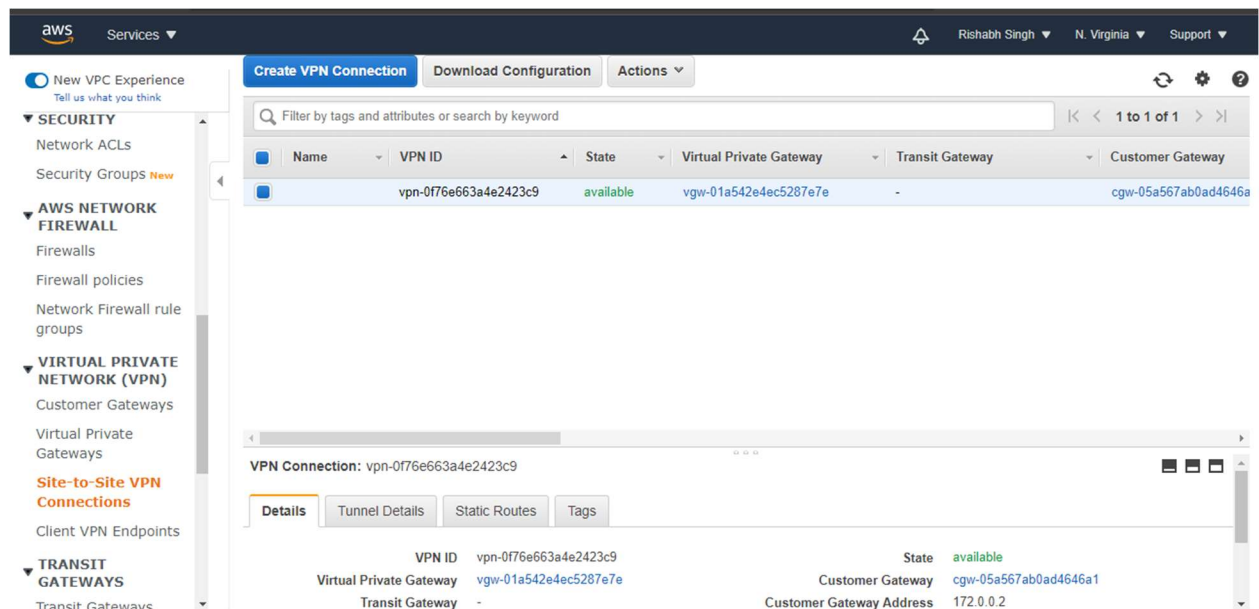
**Instances (2)** Info

Filter instances

Instance state: running X Clear filters

	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone
<input type="checkbox"/>	RishabhInsta...	i-0df587edc0d3c39a7	Running	t2.micro	Initializing	No alarms	us-east-1
<input type="checkbox"/>	RishabhInsta...	i-07aaeaab9266df5fb	Running	t2.micro	Initializing	No alarms	us-east-1

### VPN: -



**Create VPN Connection** Download Configuration Actions

Filter by tags and attributes or search by keyword

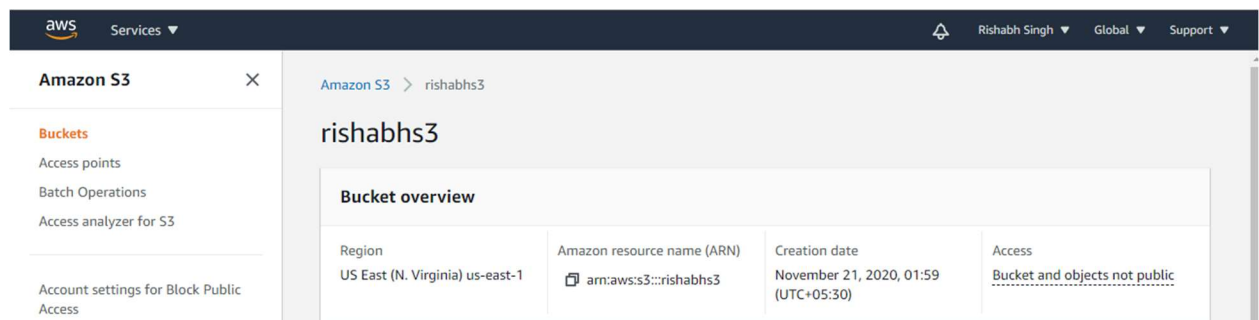
	Name	VPN ID	State	Virtual Private Gateway	Transit Gateway	Customer Gateway
<input type="checkbox"/>	vpn-0f76e663a4e2423c9	vpn-0f76e663a4e2423c9	available	vgw-01a542e4ec5287e7e	-	cgw-05a567ab0ad4646a1

VPN Connection: vpn-0f76e663a4e2423c9

Details Tunnel Details Static Routes Tags

VPN ID	vpn-0f76e663a4e2423c9	State	available
Virtual Private Gateway	vgw-01a542e4ec5287e7e	Customer Gateway	cgw-05a567ab0ad4646a1
Transit Gateway	-	Customer Gateway Address	172.0.0.2

### S3 Bucket: -



**Amazon S3** X

Amazon S3 > rishabhs3

### rishabhs3

**Bucket overview**

Region	Amazon resource name (ARN)	Creation date	Access
US East (N. Virginia) us-east-1	arn:aws:s3:::rishabhs3	November 21, 2020, 01:59 (UTC+05:30)	Bucket and objects not public