

SPCM - LAB

6th Sem

Submitted To:

Dr. Hitesh Kumar Sharma

School of Computer Science

Submitted By:

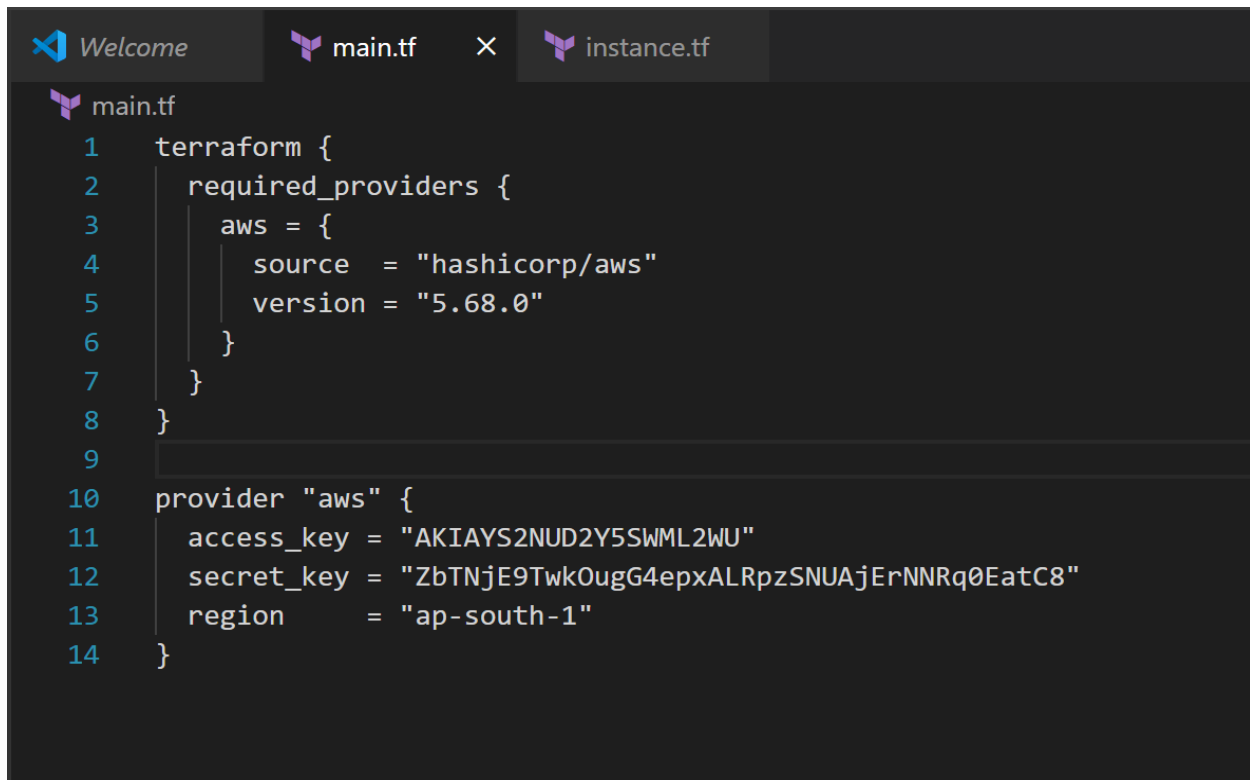
Samar Kumar Singh

R2142220283

500106348

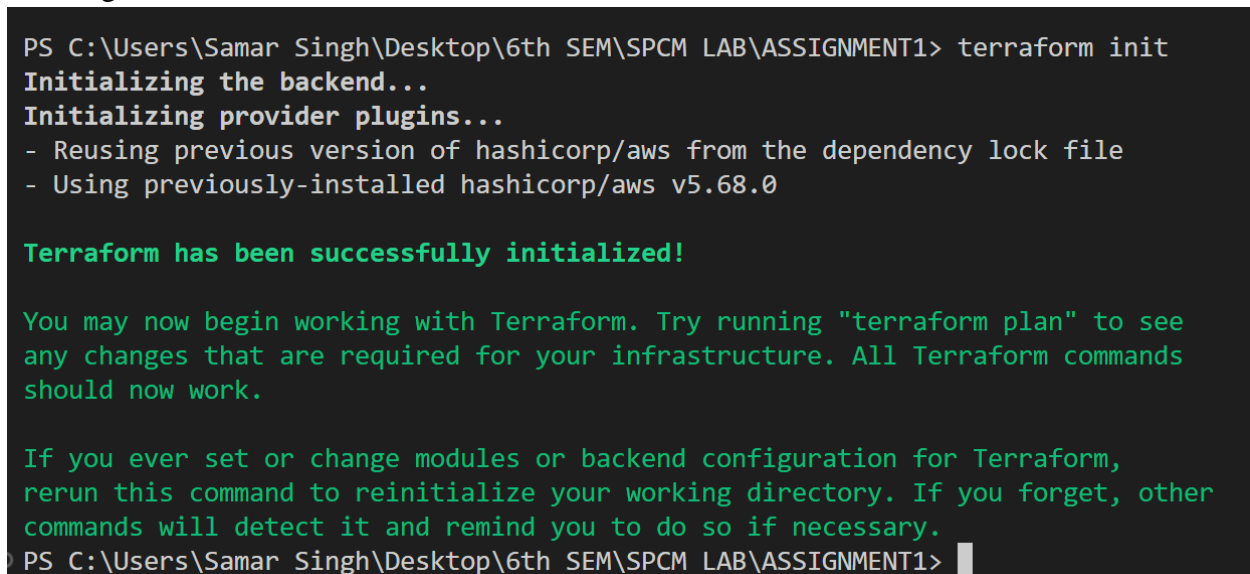
Batch 1-NH

Writing Terraform Scripts to perform the following task 2 ec2 Instances, VPN and S3 main.tf



```
1 terraform {
2   required_providers {
3     aws = {
4       source = "hashicorp/aws"
5       version = "5.68.0"
6     }
7   }
8 }
9
10 provider "aws" {
11   access_key = "AKIAYS2NUD2Y5SWML2WU"
12   secret_key = "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"
13   region     = "ap-south-1"
14 }
```

Running terraform init



```
PS C:\Users\Samar Singh\Desktop\6th SEM\SPCM LAB\ASSIGNMENT1> terraform init
Initializing the backend...
Initializing provider plugins...
- Reusing previous version of hashicorp/aws from the dependency lock file
- Using previously-installed hashicorp/aws v5.68.0

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.
PS C:\Users\Samar Singh\Desktop\6th SEM\SPCM LAB\ASSIGNMENT1>
```

Terraform init to initialize the terraform folder which will have the aws provider plugin installed

instance.tf

```
1  resource "aws_instance" "ins1" {
2      ami          = "ami-0e35ddab05955cf57"
3      instance_type = "t2.micro"
4
5      tags = {
6          Name = "Instance1"
7      }
8  }
9
10 resource "aws_instance" "ins2" {
11     ami          = "ami-0e35ddab05955cf57"
12     instance_type = "t2.micro"
13
14     tags = {
15         Name = "Instance2"
16     }
17 }
```

This file holds the iac code to make 2 instances - t2-micro ec2 machines

resource.tf

```
resource.tf
1  resource "aws_vpc" "main"{
2    cidr_block = "10.0.0.0/16"
3    tags = {
4      Name = "SamarVPC"
5    }
6  }
7  resource "aws_vpn_gateway" "example" {
8    vpc_id = aws_vpc.main.id
9
10   tags = {
11     Name = "MyVPNGateway"
12   }
13 }
14 resource "aws_customer_gateway" "example" {
15   bgp_asn      = 65000
16   ip_address   = "203.0.113.1" # Replace with actual IP
17   type         = "ipsec.1"
18
19   tags = {
20     Name = "MyCustomerGateway"
21   }
22 }
23 resource "aws_vpn_connection" "example" {
24   customer_gateway_id = aws_customer_gateway.example.id
25   vpn_gateway_id      = aws_vpn_gateway.example.id
26   type                 = "ipsec.1"
27   static_routes_only  = true
28 }
```

This resource.tf hold the iac code to create vpc , the customer gateway and the vpn connection.

s3.tf

```
1
2  ✓ resource "aws_s3_bucket" "example" {
3  ✓   bucket = "Samar-singh-upes-s3"
4     acl    = "private"
5  }
```

The s3.tf hold the code to create a s3 bucket which has a unique name

Outputs:

terraform plan

```
    + tunnel2_log_options (known after apply)
  }

# aws_vpn_gateway.example will be created
+ resource "aws_vpn_gateway" "example" {
  + amazon_side_asn = (known after apply)
  + arn              = (known after apply)
  + id               = (known after apply)
  + tags             = {
    + "Name" = "MyVPNGateway"
  }
  + tags_all         = {
    + "Name" = "MyVPNGateway"
  }
  + vpc_id           = (known after apply)
}
```

Terraform apply

```
aws_vpn_connection.example: Still creating... [5m0s elapsed]
aws_vpn_connection.example: Creation complete after 5m7s [id=vpn-029da44f6ca1c9be9]

Warning: Argument is deprecated

  with aws_s3_bucket.example,
  on s3.tf line 4, in resource "aws_s3_bucket" "example":
    4:     acl     = "private"

Use the aws_s3_bucket_acl resource instead

Apply complete! Resources: 7 added, 0 changed, 0 destroyed.
(base) ➜ assignment1
```

Customer Gateway

Customer gateways (1) info

Find resource by attribute or tag

Name	Customer gateway ID	State	BGP ASN	IP address	Type	Certificate ARN
MyCustomerGateway	cgw-0db8bed0020c682a5	Available	65000	203.0.113.1	ipsec.1	-

Vpc

Your VPCs (2) info

Find VPCs by attribute or tag

Name	VPC ID	State	Block Public...	IPv4 CIDR	IPv6 CIDR	DHCP option set	Main route table
SamarVPC	vpc-0c34a7c6ebfc0a060	Available	Off	10.0.0.0/16	-	dopt-0b00b9b8f9deec8...	rtb-09ec57a6b70
-	vpc-0aa7c2c8ba69a7678	Available	Off	172.31.0.0/16	-	dopt-0b00b9b8f9deec8...	rtb-0c63f5161e7...

S3

General purpose buckets | Directory buckets

General purpose buckets (1) info All AWS Regions

Buckets are containers for data stored in S3.

Find buckets by name

Name	AWS Region	IAM Access Analyzer	Creation date
samar-upes-s3	Asia Pacific (Mumbai) ap-south-1	View analyzer for ap-south-1	April 22, 2025, 11:01:46 (UTC+05:30)

Instances

Instances (2) info

Find Instance by attribute or tag (case-sensitive)

All states

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Pu
Instance1	i-0d47a9f307d7ccc3b	Running	t2.micro	2/2 checks passed	View alarms +	ap-south-1b	ec2-35-154-107-95.ap-...	35
Instance2	i-081605682eca104d5	Running	t2.micro	2/2 checks passed	View alarms +	ap-south-1b	ec2-65-0-105-63.ap-so...	65