

ASSIGNMENT - 1

System Provisioning & Config. Management

Write Terraform script to do perform following tasks on AWS cloud Platform

Step 1: Create two T2 Micro EC2 Instances.

Step2: Create a VPN on AWS

Step 3: Create a S3 Bucket

Step 4: Write the code for step 1,2 and 3 in a IaC terraform file and run terraform commands to execute these steps.

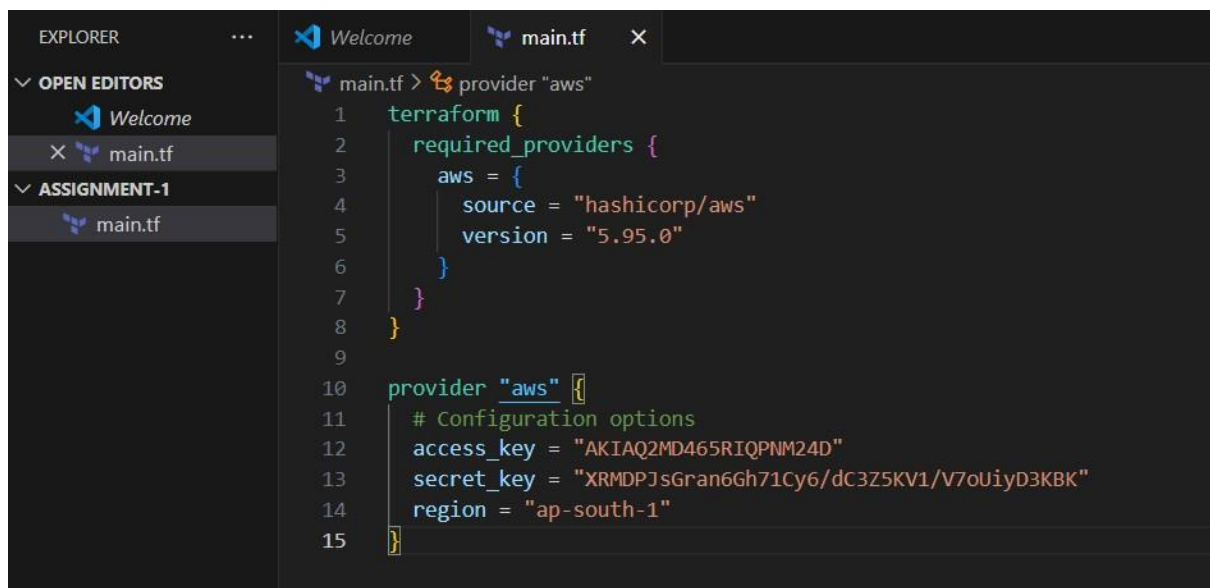
Create a folder mkdir

assignment-1 cd

assignment-1

```
C:\Users\laugh>mkdir assignment-1  
C:\Users\laugh>cd assignment-1  
C:\Users\laugh\assignment-1>code . |
```

First create main.tf

A screenshot of the Visual Studio Code editor interface. The Explorer sidebar on the left shows a project named 'ASSIGNMENT-1' with a file 'main.tf' selected. The main editor area displays the content of 'main.tf'. The file starts with a Terraform provider configuration for AWS, including required providers and provider options like access_key, secret_key, and region. The code is as follows:

```
1 terraform {  
2   required_providers {  
3     aws = {  
4       source = "hashicorp/aws"  
5       version = "5.95.0"  
6     }  
7   }  
8 }  
9  
10 provider "aws" {  
11   # Configuration options  
12   access_key = "AKIAQ2MD465RIQPNM24D"  
13   secret_key = "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"  
14   region = "ap-south-1"  
15 }
```

```
terraform {  
required_providers {  
aws = {
```

```

    source = "hashicorp/aws"
version = "5.95.0"
  }
}
} provider
"aws" {
  # Configuration options  access_key =
"AKIAQ2MD465RIQPNM24D"  secret_key =
"XRMDPJ5Gran6Gh71Cy6/dC3Z5KV1/V7oUiD3KKBK"  region =
"ap-south-1"
}

```

Step 1: Create two T2 Micro EC2 Instances. Create

ec2.tf

```

resource "aws_instance" "example1" {
ami          = "ami-0f58b397bc5c1f2e8"
instance_type = "t2.micro"
  tags = {
    Name = "EC2_Instance_1"
  }
} resource "aws_instance" "example2" {
ami          = "ami-0f58b397bc5c1f2e8"
instance_type = "t2.micro"
  tags = {
    Name = "EC2_Instance_2"
  }
}

```

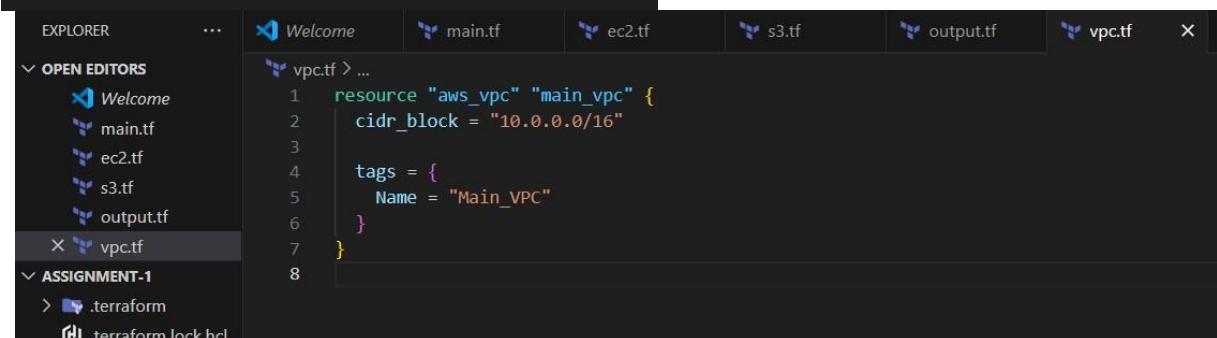
Step2: Create a VPN on AWS

Create vpc.tf

```

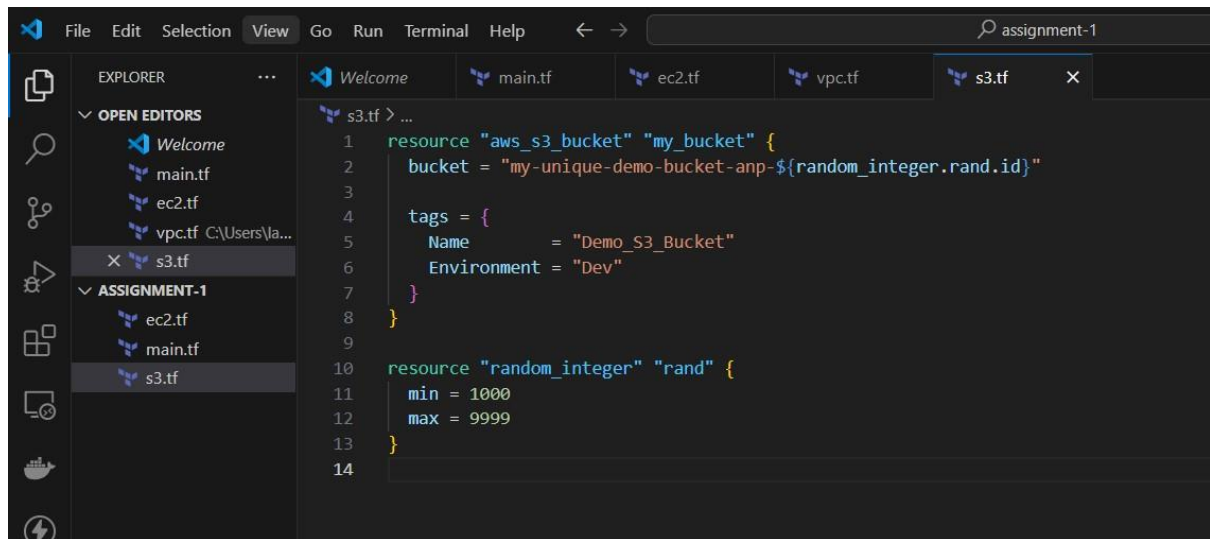
resource "aws_vpc" "main_vpc" {
cidr_block = "10.0.0.0/16"
  tags = {
    Name = "Main_VPC"
  }
}

```



Step 3: Create a S3 Bucket

Create s3.tf



```
resource "aws_s3_bucket" "my_bucket" {  bucket = "my-unique-  
demo-bucket-anp-${random_integer.rand.id}"  
  
  tags = {  
    Name       = "Demo_S3_Bucket"  
    Environment = "Dev"  
  }  
}  
resource "random_integer" "rand"  
{  min = 1000  max = 9999  
}
```

Run terraform init

This initializes the Terraform working directory by downloading necessary provider plugins and setting up the backend configuration. It must be run before any other commands

```
C:\Users\laugh\assignment-1>terraform init
Initializing the backend...
Initializing provider plugins...
- Finding latest version of hashicorp/random...
- Finding hashicorp/aws versions matching "5.95.0"...
- Installing hashicorp/random v3.7.2...
- Installed hashicorp/random v3.7.2 (signed by HashiCorp)
- Installing hashicorp/aws v5.95.0...
- Installed hashicorp/aws v5.95.0 (signed by HashiCorp)
Terraform has created a lock file .terraform.lock.hcl to record the provider
selections it made above. Include this file in your version control repository
so that Terraform can guarantee to make the same selections by default when
you run "terraform init" in the future.
```

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.

Now **terraform plan** : It's like a dry run to review changes before applying them

```
# random_integer.rand will be created
+ resource "random_integer" "rand" {
+   id       = (known after apply)
+   max      = 9999
+   min      = 1000
+   result   = (known after apply)
+ }

Plan: 5 to add, 0 to change, 0 to destroy.

Changes to Outputs:
+ ec2_instance_1_public_ip = (known after apply)
+ ec2_instance_2_public_ip = (known after apply)
+ s3_bucket_name           = (known after apply)
+ vpc_id                   = (known after apply)

Note: You didn't use the -out option to save this plan, so Terraform can't guarantee to take exactly these actions if
you run "terraform apply" now.

C:\Users\laugh\assignment-1>
```

terraform apply : Applies the changes required to reach the desired state as defined in the configuration files. It provisions or updates resources on the cloud platform

```

Enter a value: yes

random_integer.rand: Creating...
random_integer.rand: Creation complete after 0s [id=3188]
aws_vpc.main_vpc: Creating...
aws_s3_bucket.my_bucket: Creating...
aws_instance.example1: Creating...
aws_instance.example2: Creating...
aws_vpc.main_vpc: Creation complete after 2s [id=vpc-01520a7617053ccf8]
aws_s3_bucket.my_bucket: Creation complete after 2s [id=my-unique-demo-bucket-anp-3188]
aws_instance.example1: Still creating... [10s elapsed]
aws_instance.example2: Still creating... [10s elapsed]
aws_instance.example1: Creation complete after 12s [id=i-08db651130ce6bc7c]
aws_instance.example2: Creation complete after 13s [id=i-02b4cb0aa6d06aca4]

Apply complete! Resources: 5 added, 0 changed, 0 destroyed.

Outputs:

ec2_instance_1_public_ip = "13.127.23.251"
ec2_instance_2_public_ip = "13.203.200.21"
s3_bucket_name = "my-unique-demo-bucket-anp-3188"
vpc_id = "vpc-01520a7617053ccf8"

C:\Users\laugh\assignment-1>

```

Instances (5) [Info](#) Last updated less than a minute ago [Connect](#) [Instance state](#) [Actions](#) [Launch instances](#)

Find Instance by attribute or tag (case-sensitive) [All states](#)

<input type="checkbox"/>	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone
<input type="checkbox"/>	Ansible Server	i-03f0d726080db462e	Stopped	t2.micro	-	View alarms +	ap-south-1b
<input type="checkbox"/>	Node 1	i-08656b9b9793e4b61	Stopped	t2.micro	-	View alarms +	ap-south-1b
<input type="checkbox"/>	Node 2	i-01b45819c8d108860	Stopped	t2.micro	-	View alarms +	ap-south-1b
<input type="checkbox"/>	EC2_Instance_2	i-02b4cb0aa6d06aca4	Running	t2.micro	Initializing	View alarms +	ap-south-1b
<input type="checkbox"/>	EC2_Instance_1	i-08db651130ce6bc7c	Running	t2.micro	Initializing	View alarms +	ap-south-1b

General purpose buckets (1) [Info](#) [All AWS Regions](#) [Copy ARN](#) [Empty](#) [Delete](#) [Create bucket](#)

Buckets are containers for data stored in S3.

Find buckets by name [All states](#)

<input type="radio"/>	Name	AWS Region	IAM Access Analyzer	Creation date
<input type="radio"/>	my-unique-demo-bucket-anp-3188	Asia Pacific (Mumbai) ap-south-1	View analyzer for ap-south-1	April 25, 2025, 23:55:14 (UTC+05:30)

Your VPCs (2) [Info](#) Last updated less than a minute ago [Actions](#) [Create VPC](#)

Find VPCs by attribute or tag [All states](#)

<input type="checkbox"/>	Name	VPC ID	State	Block Public...	IPv4 CIDR	IPv6 CIDR
<input type="checkbox"/>	Main_VPC	vpc-01520a7617053ccf8	Available	Off	10.0.0.0/16	-
<input type="checkbox"/>	-	vpc-008f69ac4dbef893d	Available	Off	172.31.0.0/16	-

Clean up using **terraform destroy**

```
aws_vpc.main_vpc: Destroying... [id=vpc-01520a7617053ccf8]
aws_s3_bucket.my_bucket: Destroying... [id=my-unique-demo-bucket-anp-3188]
aws_instance.example1: Destroying... [id=i-08db651130ce6bc7c]
aws_instance.example2: Destroying... [id=i-02b4cb0aa6d06aca4]
aws_s3_bucket.my_bucket: Destruction complete after 0s
random_integer.rand: Destroying... [id=3188]
random_integer.rand: Destruction complete after 0s
aws_vpc.main_vpc: Destruction complete after 1s
aws_instance.example2: Still destroying... [id=i-02b4cb0aa6d06aca4, 10s elapsed]
aws_instance.example1: Still destroying... [id=i-08db651130ce6bc7c, 10s elapsed]
aws_instance.example1: Still destroying... [id=i-08db651130ce6bc7c, 20s elapsed]
aws_instance.example2: Still destroying... [id=i-02b4cb0aa6d06aca4, 20s elapsed]
aws_instance.example2: Still destroying... [id=i-02b4cb0aa6d06aca4, 30s elapsed]
aws_instance.example1: Still destroying... [id=i-08db651130ce6bc7c, 30s elapsed]
aws_instance.example2: Destruction complete after 30s
aws_instance.example1: Still destroying... [id=i-08db651130ce6bc7c, 40s elapsed]
aws_instance.example1: Still destroying... [id=i-08db651130ce6bc7c, 50s elapsed]
aws_instance.example1: Still destroying... [id=i-08db651130ce6bc7c, 1m0s elapsed]
aws_instance.example1: Destruction complete after 1m1s
```

Destroy complete! Resources: 5 destroyed.

C:\Users\laugh\assignment-1>