

ASSIGNMENT 1

NAME : Simranjeet Kour

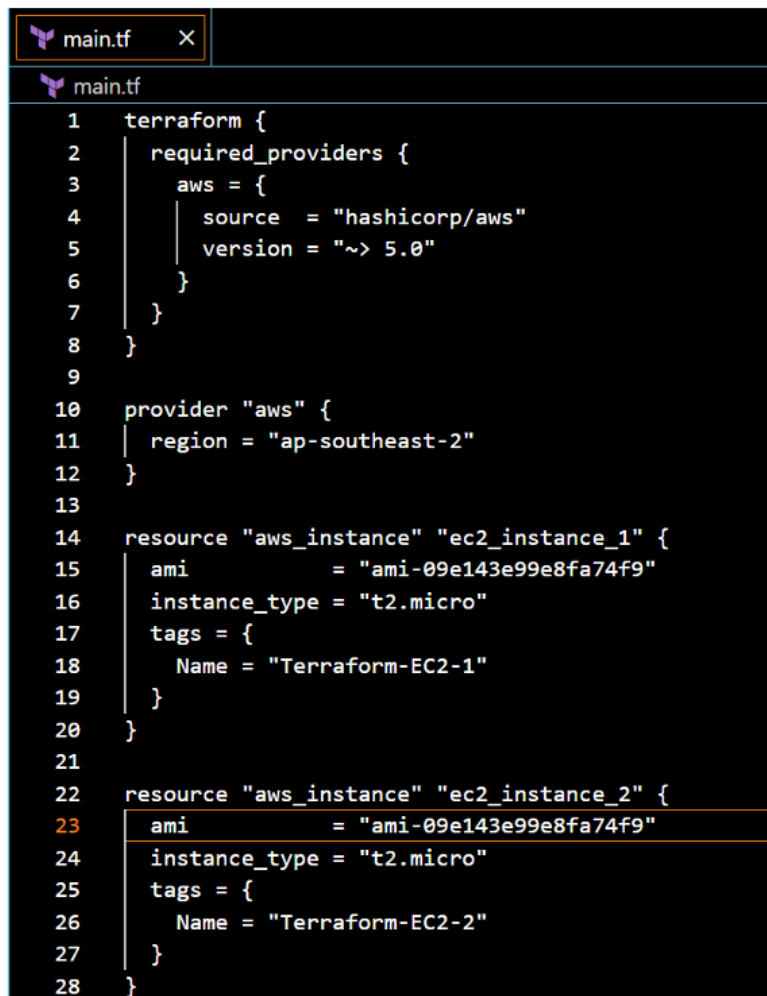
SAP-ID : 500110713

BATCH : B-2

ROLL NO. : R2142221495

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

Step 1: Create two T2 Micro EC2 Instances.

A screenshot of a code editor window titled 'main.tf'. The editor contains a Terraform script for creating two EC2 instances. The script is as follows:

```
1 terraform {
2   required_providers {
3     aws = {
4       source = "hashicorp/aws"
5       version = "~> 5.0"
6     }
7   }
8 }
9
10 provider "aws" {
11   region = "ap-southeast-2"
12 }
13
14 resource "aws_instance" "ec2_instance_1" {
15   ami           = "ami-09e143e99e8fa74f9"
16   instance_type = "t2.micro"
17   tags = {
18     Name = "Terraform-EC2-1"
19   }
20 }
21
22 resource "aws_instance" "ec2_instance_2" {
23   ami           = "ami-09e143e99e8fa74f9"
24   instance_type = "t2.micro"
25   tags = {
26     Name = "Terraform-EC2-2"
27   }
28 }
```

Step2: Create a VPN on AWS

```
main.tf x
main.tf
30 resource "aws_vpc" "main" {
31   cidr_block = "10.0.0.0/16"
32   tags = {
33     Name = "Terraform-VPC"
34   }
35 }
36
37 resource "aws_subnet" "public_subnet" {
38   vpc_id      = aws_vpc.main.id
39   cidr_block   = "10.0.1.0/24"
40   availability_zone = "${data.aws_availability_zones.available.names[0]}"
41   map_public_ip_on_launch = true
42   tags = {
43     Name = "Terraform-Public-Subnet"
44   }
45 }
46
47 resource "aws_internet_gateway" "gw" {
48   vpc_id = aws_vpc.main.id
49   tags = {
50     Name = "Terraform-Internet-Gateway"
51   }
52 }
53
54 resource "aws_route_table" "public_rt" {
55   vpc_id = aws_vpc.main.id
56   route {
57     cidr_block = "0.0.0.0/0"
58     gateway_id = aws_internet_gateway.gw.id
59   }
60   tags = {
61     Name = "Terraform-Public-RouteTable"
62   }
63 }
```

Step 3: Create a S3 Bucket

```
main.tf
main.tf
64
65 resource "aws_s3_bucket" "my_bucket" {
66   bucket = "terraform-ass1-${random_id.bucket_id.hex}"
67   tags = {
68     Name      = "TerraformExampleBucket"
69     Environment = "Dev"
70   }
71 }
72
73 resource "aws_s3_bucket_acl" "my_bucket_acl" {
74   bucket = aws_s3_bucket.my_bucket.id
75   acl    = "private"
76 }
77
78 resource "random_id" "bucket_id" {
79   byte_length = 8
80 }
81
82 data "aws_availability_zones" "available" {}
```

Step 4: Write the code for step 1,2 and 3 in a IaC terraform file and run terraform

Terraform init:

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS D:\College\Sem-6\System Provisioning\Lab\terraform-ass1> C:\terraform init
Initializing the backend...
Initializing provider plugins...
- Finding hashicorp/aws versions matching "~> 5.0"...
- Finding latest version of hashicorp/random...
- Installing hashicorp/aws v5.94.1...
- Installed hashicorp/aws v5.94.1 (signed by HashiCorp)
- Installing hashicorp/random v3.7.1...
- Installed hashicorp/random v3.7.1 (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.
```

Terraform plan:

```
PS D:\College\Sem-6\System Provisioning\Lab\terraform-ass1> C:\terraform plan
data.aws_availability_zones.available: Reading...
data.aws_availability_zones.available: Read complete after 1s [id=ap-southeast-2]

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the
following symbols:
+ create

Terraform will perform the following actions:

# aws_instance.ec2_instance_1 will be created
+ resource "aws_instance" "ec2_instance_1" {
  + ami                        = "ami-09e143e99e8fa74f9"
  + 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)
  + disable_api_stop          = (known after apply)
  + disable_api_termination   = (known after apply)
  + ebs_optimized              = (known after apply)
  + enable_primary_ipv6       = (known after apply)
  + get_password_data          = false
  + host_id                   = (known after apply)
  + host_resource_group_arn    = (known after apply)
  + iam_instance_profile       = (known after apply)
  + id                        = (known after apply)
  + instance_initiated_shutdown_behavior = (known after apply)
```

Terraform plan:

```
+ id = (known after apply)
+ instance_tenancy = "default"
+ ipv6_association_id = (known after apply)
+ ipv6_cidr_block = (known after apply)
+ ipv6_cidr_block_network_border_group = (known after apply)
+ main_route_table_id = (known after apply)
+ owner_id = (known after apply)
+ tags = {
  + "Name" = "Terraform-VPC"
}
+ tags_all = {
  + "Name" = "Terraform-VPC"
}
}

# random_id.bucket_id will be created
+ resource "random_id" "bucket_id" {
  + b64_std = (known after apply)
  + b64_url = (known after apply)
  + byte_length = 8
  + dec = (known after apply)
  + hex = (known after apply)
  + id = (known after apply)
}
```

Plan: 9 to add, 0 to change, 0 to destroy.

Terraform apply:

```
PS D:\College\Sem-6\System Provisioning\Lab\terraform-ass1> C:\terraform apply
random_id.bucket_id: Refreshing state... [id=RMnQU3_FvdI]
data.aws_availability_zones.available: Reading...
aws_vpc.main: Refreshing state... [id=vpc-066125ae4a8302e52]
aws_s3_bucket.my_bucket: Refreshing state... [id=terraform-ass1-44c9d0537fc5bdd2]
aws_instance.ec2_instance_1: Refreshing state... [id=i-0c613319867b09f06]
aws_instance.ec2_instance_2: Refreshing state... [id=i-035169ef2ad81d1bd]
data.aws_availability_zones.available: Read complete after 1s [id=ap-southeast-2]
aws_internet_gateway.gw: Refreshing state... [id=igw-01c8e32c2e2e0eb8d]
aws_subnet.public_subnet: Refreshing state... [id=subnet-05ffef25f915b4a02]
aws_route_table.public_rt: Refreshing state... [id=rtb-0cd6a048ea5e1feca]

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the
following symbols:
  ~ update in-place

Terraform will perform the following actions:
```

```
    }
    ~ tags_all = {
      ~ "Name" = "TerraformExampleBucket" -> "TerraformAss1Bucket"
        # (1 unchanged element hidden)
    }
    # (12 unchanged attributes hidden)

    # (3 unchanged blocks hidden)
  }

Plan: 0 to add, 1 to change, 0 to destroy.

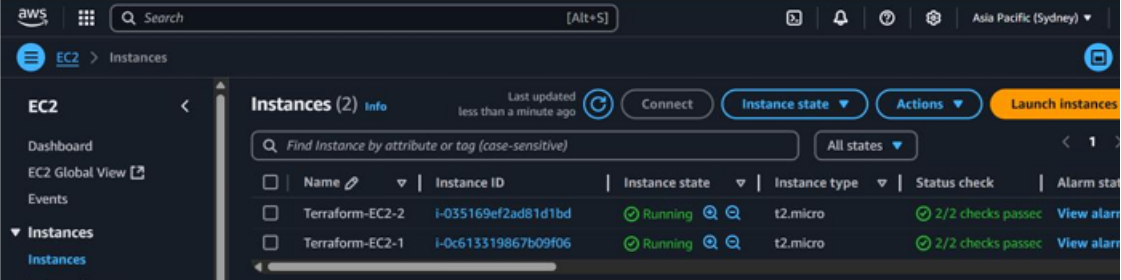
Do you want to perform these actions?
  Terraform will perform the actions described above.
  Only 'yes' will be accepted to approve.

  Enter a value: yes

aws_s3_bucket.my_bucket: Modifying... [id=terraform-ass1-44c9d0537fc5bdd2]
aws_s3_bucket.my_bucket: Modifications complete after 4s [id=terraform-ass1-44c9d0537fc5bdd2]

Apply complete! Resources: 0 added, 1 changed, 0 destroyed.
```

AWS Console Output: Instance-



	Name	Instance ID	Instance state	Instance type	Status check	Alarm state
<input type="checkbox"/>	Terraform-EC2-2	i-035169ef2ad81d1bd	Running	t2.micro	2/2 checks passed	View alarm
<input type="checkbox"/>	Terraform-EC2-1	i-0c613319867b09f06	Running	t2.micro	2/2 checks passed	View alarm

VPN-

VPCs ▶ See all regions	Asia Pacific 2	NAT Gateways ▶ See all regions	Asia Pacific 0
Subnets ▶ See all regions	Asia Pacific 4	VPC Peering Connections ▶ See all regions	Asia Pacific 0
Route Tables ▶ See all regions	Asia Pacific 3	Network ACLs ▶ See all regions	Asia Pacific 2
Internet Gateways ▶ See all regions	Asia Pacific 2	Security Groups ▶ See all regions	Asia Pacific 13
Egress-only Internet Gateways ▶ See all regions	Asia Pacific 0	Customer Gateways ▶ See all regions	Asia Pacific 0
DHCP option sets ▶ See all regions	Asia Pacific 1	Virtual Private Gateways ▶ See all regions	Asia Pacific 0

This screenshot shows the AWS VPC dashboard in the Asia Pacific (Sydney) region. The left sidebar contains navigation links for 'VPC dashboard', 'EC2 Global View', 'Virtual private cloud', 'Your VPCs', and 'Subnets'. The main content area is titled 'Your VPCs (2)' and includes a search bar and a table of VPCs. The table lists two VPCs: one with ID 'vpc-07ab47408653afc65' and another named 'Terraform-VPC' with ID 'vpc-066125ae4a8302e52'. Both are in an 'Available' state with 'Block Public Access' set to 'Off'.

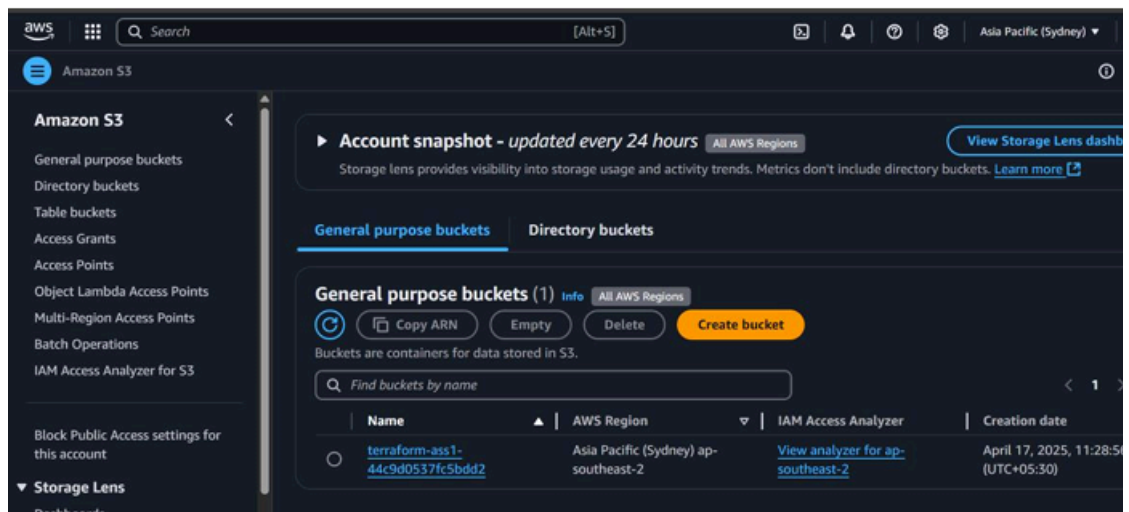
Name	VPC ID	State	Block Public...	IPv4 CIDR
-	vpc-07ab47408653afc65	Available	Off	172.31.0.0/16
Terraform-VPC	vpc-066125ae4a8302e52	Available	Off	10.0.0.0/16

This screenshot shows the 'Subnets (4)' section of the AWS VPC dashboard. It features a search bar and a table listing four subnets. All subnets are in an 'Available' state and are associated with the same VPC ID: 'vpc-07ab47408653afc65'. The subnets are: 'Terraform-Public-Subnet' (ID: subnet-05ffe725f915b4a02), and three unnamed subnets with IDs subnet-0aa9f8f7d7b0abe72, subnet-038da8992db8c8ca4, and subnet-0d1beb62e57a2af053.

Name	Subnet ID	State	VPC
-	subnet-0aa9f8f7d7b0abe72	Available	vpc-07ab47408653afc65
Terraform-Public-Subnet	subnet-05ffe725f915b4a02	Available	vpc-066125ae4a8302e52
-	subnet-038da8992db8c8ca4	Available	vpc-07ab47408653afc65
-	subnet-0d1beb62e57a2af053	Available	vpc-07ab47408653afc65



S3 Bucket-



Terraform destroy:

```
PS D:\College\Sem-6\System Provisioning\Lab\terraform-ass1> C:\terraform destroy
random_id.bucket_id: Refreshing state... [id=RMnQU3_FvdI]
aws_vpc.main: Refreshing state... [id=vpc-066125ae4a8302e52]
data.aws_availability_zones.available: Reading...
aws_s3_bucket.my_bucket: Refreshing state... [id=terraform-ass1-44c9d0537fc5bdd2]
aws_instance.ec2_instance_2: Refreshing state... [id=i-035169ef2ad81d1bd]
aws_instance.ec2_instance_1: Refreshing state... [id=i-0c613319867b09f06]
data.aws_availability_zones.available: Read complete after 1s [id=ap-southeast-2]
aws_internet_gateway.gw: Refreshing state... [id=igw-01c8e32c2e2e0eb8d]
aws_subnet.public_subnet: Refreshing state... [id=subnet-05ffef25f915b4a02]
aws_route_table.public_rt: Refreshing state... [id=rtb-0cd6a048ea5e1feca]
```

```
aws_instance.ec2_instance_1: Destruction complete after 1m12s
Destroy complete! Resources: 8 destroyed.
PS D:\College\Sem-6\System Provisioning\Lab\terraform-ass1> |
```