# **SPCM LAB**

Gopika Jhanwar 500093662 R2142210318 Btech cse devops B2

# Lab Exercise 3–Provisioning an EC2 Instance on AWS

Step 1: Create a New Directory: mkdir aws-terraform-demo cd aws-terraform-demo Step 2: Create Terraform Configuration File (main.tf)

```
aws-terraform-demo >  main.tf

1   terraform {
2   required_providers {
3   aws = {
4   source = "hashicorp/aws"
5   version = "5.31.0"
6  }
7  }
8  }
9  provider "aws" {
10  region = "ap-south-1"
11  access_key = "AKIAYS2NV47DL6IMWZUT"
12  secret_key = "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"
13 }
```

Step 3: Initialize Terraform: terraform ini

```
PS D:\6 th sem\SPCM\SPCM LAB\teraform lab files> cd aws-terraform-demo
PS D:\6 th sem\SPCM\SPCM LAB\tert orm lab files\aws-terraform-demo>
PS D:\6 th sem\SPCM\SPCM LAB\teraform lab files\aws-terraform-demo> terraform init
Initializing the backend...
Initializing provider plugins...
- Finding hashicorp/aws versions matching "5.31.0"...

    Installing hashicorp/aws v5.31.0...
    Installed hashicorp/aws v5.31.0 (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.
PS D:\6 th sem\SPCM\SPCM LAB\teraform lab files\aws-terraform-demo>
```

Step 4: Create Terraform Configuration File for EC2 instance (instance.tf):

```
aws-terraform-demo > instance.tf

1    resource "aws_instance" "My-instance" {
2    instance_type = "t2.micro"
3    ami = "ami-0e670eb768a5fc3d4"
4    count = 1
5    tags = {
6    Name = "UPES-EC2-Instnace"
7    }
8 }
```

Step 5: Review Plan: terraform plan

```
PS D:\6 th sem\SPCM\SPCM LAB\teraform lab files\aws-terraform-demo> terraform plan
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:
 \mbox{\tt\#} \mbox{\tt aws\_instance.My-instance[0]} will be created
 + resource "aws instance"
                       "My-instance"
                                        "ami-0e670eb768a5fc3d4"
    + ami
     + arn
                                      = (known after apply)
     + associate public ip address
                                        (known after apply)
                                        (known after apply)
     + availability zone
                                        (known after apply)
     + cpu_core_count
     + cpu threads per core
                                        (known after apply)
                                        (known after apply)
    + disable_api_stop
     + disable_api_termination
                                        (known after apply)
     + ebs_optimized
                                        (known after apply)
     + get_password_data
                                      = false
                                      = (known after apply)
     + host_id
     + host_resource_group_arn
                                        (known after apply)
     + iam instance_profile
                                      = (known after apply)
                                      = (known after apply)
     + id
     + instance_initiated_shutdown_behavior = (known after apply)
     + instance_lifecycle_
                                        (known after apply)
     + instance_state
+ instance_type
                                      = (known after apply)
= "t2.micro"
      ipv6_address_count
                                      = (known after apply)
     + ipv6_addresses
                                      = (known after apply)
                                        (known after apply)
     + key_name
     + monitoring
                                      = (known after apply)
     + outpost arn
                                        (known after apply)
      password data
                                        (known after apply)
        ipv6 addresses
                                                              (known after apply)
      + key_name
                                                              (known after apply)
      + monitoring
                                                              (known after apply)
                                                             (known after apply)
      + outpost arn
                                                             (known after apply)

    password data

                                                             (known after apply)

    placement group

                                                             (known after apply)

    placement partition number

      + primary_network_interface_id
                                                             (known after apply)
        private_dns
                                                             (known after apply)
                                                             (known after apply)
        private_ip
      + 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
      + spot instance request id
                                                             (known after apply)
                                                             (known after apply)
      + subnet id
        tags
             "Name" = "UPES-EC2-Instnace"
        tags_all
                                                           = {
             "Name" = "UPES-EC2-Instnace"
      + tenancy
                                                           = (known after apply)
      + user data
                                                           = (known after apply)

    user data base64

                                                           = (known after apply)

    user data replace on change

                                                           = false

    vpc security group ids

                                                           = (known after apply)
```

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

## Step 6: Apply Changes:

#### terraform apply

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

Do you want to perform these actions?

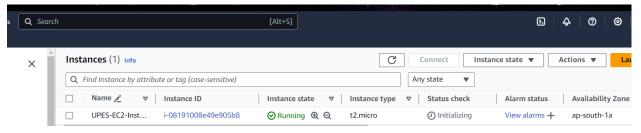
Terraform will perform the actions described above.
Only 'yes' will be accepted to approve.

Enter a value: yes

aws_instance.My-instance[0]: Creating...
aws_instance.My-instance[0]: Still creating... [10s elapsed]
aws_instance.My-instance[0]: Still creating... [20s elapsed]
aws_instance.My-instance[0]: Still creating... [30s elapsed]
aws_instance.My-instance[0]: Creation complete after 32s [id=i-08191008e49e905b8]

Apply complete! Resources: 1 added, 0 changed, 0 destroyed.
PS D:\6 th sem\SPCM\SPCM LAB\teraform lab files\aws-terraform-demo>
```

## Step 7: Verify Resources:



# Step 8: Cleanup Resources: terraform destroy

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

Do you really want to destroy all resources?

Terraform will destroy all your managed infrastructure, as shown above.

There is no undo. Only 'yes' will be accepted to confirm.

Enter a value: yes

aws_instance.My-instance[0]: Destroying... [id=i-08191008e49e905b8]

aws_instance.My-instance[0]: Still destroying... [id=i-08191008e49e905b8, 10s elapsed]

aws_instance.My-instance[0]: Still destroying... [id=i-08191008e49e905b8, 20s elapsed]

aws_instance.My-instance[0]: Still destroying... [id=i-08191008e49e905b8, 30s elapsed]

aws_instance.My-instance[0]: Destruction complete after 30s

Destroy complete! Resources: 1 destroyed.

PS D:\6 th sem\SPCM\SPCM LAB\teraform lab files\aws-terraform-demo>
```