



System Provisioning and Configuration Management Lab

Siddhant Singh

500095375

Batch - 3

Experiment 2

Prerequisites

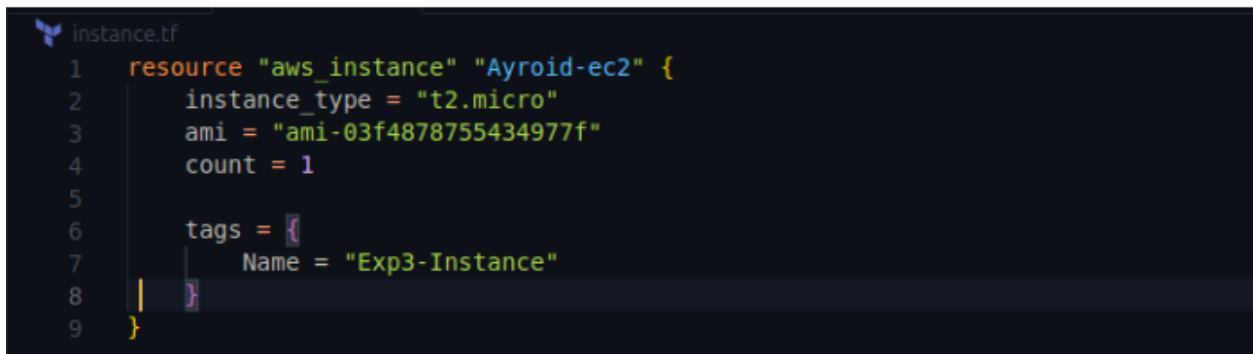
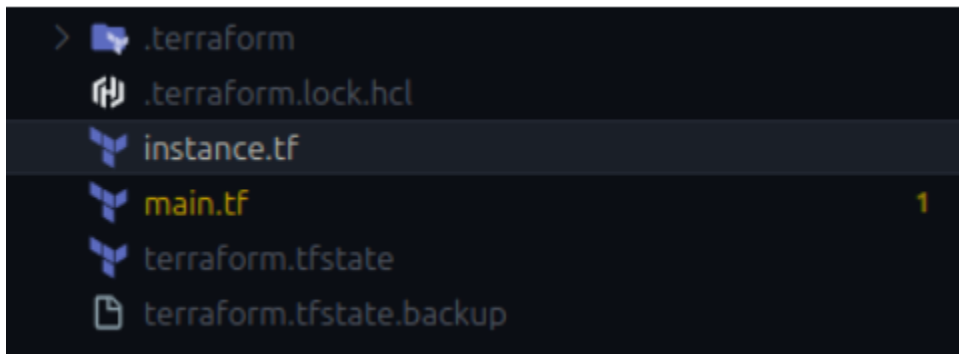
Terraform Installed & AWS Credentials.

Aim

Provisioning an EC2 Instance on AWS

Steps

1. Create a terraform configuration file for EC2 Instance (instance.tf)



2. Validate the configuration



3. Review Plan

```
● → Exp2 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:

# aws_instance.Ayroid-ec2[0] will be created
+ resource "aws_instance" "Ayroid-ec2" {
```

4. Terraform Apply

```
● → Exp2 terraform apply

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.Ayroid-ec2[0] will be created
+ resource "aws_instance" "Ayroid-ec2" {
  + ami                                = "ami-03f4878755434977f"
```

```
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.Ayroid-ec2[0]: Creating...
aws_instance.Ayroid-ec2[0]: Still creating... [10s elapsed]
aws_instance.Ayroid-ec2[0]: Still creating... [20s elapsed]
aws_instance.Ayroid-ec2[0]: Still creating... [30s elapsed]
aws_instance.Ayroid-ec2[0]: Creation complete after 34s [id=i-0688471057f5dbedd]
```

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

5. Verify Resources

Instances (1) Info								
<input type="text" value="Find Instance by attribute or tag (case-sensitive)"/>								
<input type="checkbox"/>	Name 	Instance ID	Instance state 	Instance type 	Status check	Alarm status	Availability	
<input type="checkbox"/>	Exp3-Instance	i-0688471057f5dbedd	 Pending  	t2.micro	-	View alarms 	ap-south-1a	

6. Cleanup Resources

```
• → Exp2 terraform destroy
aws_instance.Ayroid-ec2[0]: Refreshing state... [id=i-0688471057f5dbedd]

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

Terraform will perform the following actions:

# aws_instance.Ayroid-ec2[0] will be destroyed
- resource "aws_instance" "Ayroid-ec2" {
  - ami              = "ami-03f4878755434977f" -> null
  - arn              = "arn:aws:ec2:ap-south-1:851931352354:instance/i-0688471057f5dbedd"
  - availability_zone = "ap-south-1a"
  - ebs_optimized     = false
  - enable_terraform  = false
  - instance_profile   = "aws-ec2-instance-profile"
  - key_name           = "ayroid-ec2-key"
  - monitoring         = false
  - subnet             = "subnet-03000000"
  - tags               = {}
  - vpc_security_group_ids = ["sg-03000000"]
}

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.Ayroid-ec2[0]: Destroying... [id=i-0688471057f5dbedd]
aws_instance.Ayroid-ec2[0]: Still destroying... [id=i-0688471057f5dbedd, 10s elapsed]
aws_instance.Ayroid-ec2[0]: Still destroying... [id=i-0688471057f5dbedd, 20s elapsed]
aws_instance.Ayroid-ec2[0]: Still destroying... [id=i-0688471057f5dbedd, 30s elapsed]
aws_instance.Ayroid-ec2[0]: Destruction complete after 31s

Destroy complete! Resources: 1 destroyed.
○ → Exp2 █
```