Lesson 07 Demo 8

Create an S3 Bucket Using Terraform

Objective: To create a S3 bucket using Terraform

Pre-requisites: You need to have Terraform installed in order to proceed with this demo. If you don't have it installed, refer to demo 1 of lesson 7.

Tools required: Terraform

Steps to be performed

1. Set up Terraform components
2. Create Terraform execution plan

Step 1: Set up Terraform components

1.1 Run the below commands in the given sequence to set up the Terraform component:

pip install awscli
sudo apt-get update

1.2 Create a new file to execute this project.

mkdir s3back
cd s3back

Step 2: Create a Terraform execution plan

2.1 Create **creds.tf** under **s3back** and add the code given below:

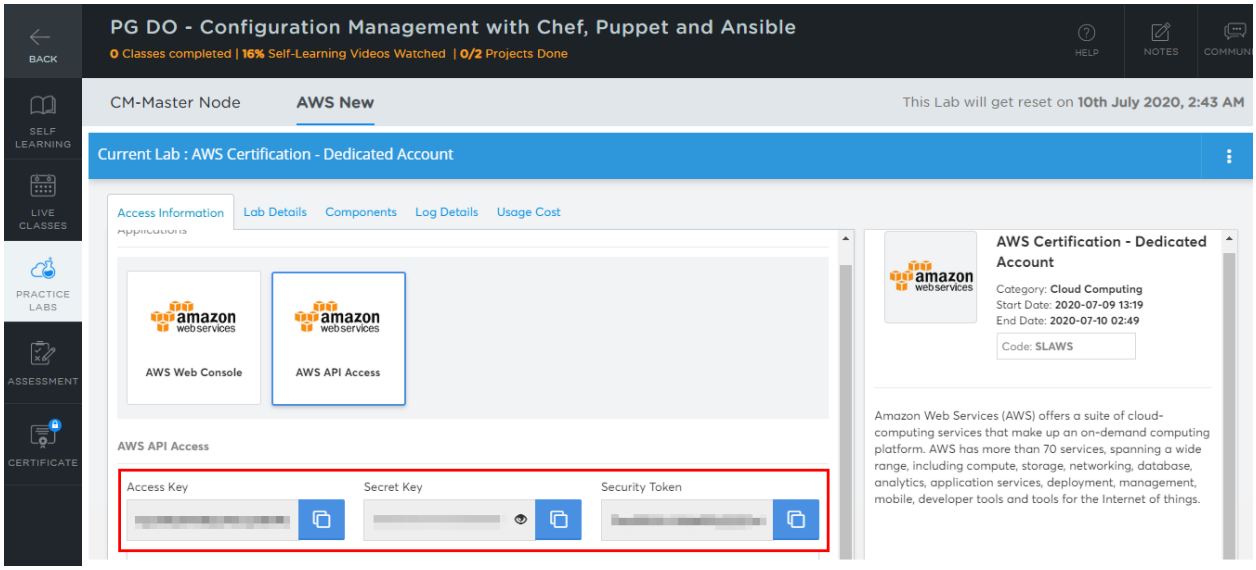
nano creds.tf

2.2 Paste the below code:

```
provider "aws" {
  access_key = ""
  secret_key = ""
  token = ""
  region = "us-east-1"
}
```

```
root@ip-172-31-66-87:/home/anjanasinghsimp# cd s3back
root@ip-172-31-66-87:/home/anjanasinghsimp/s3back# nano creds.tf
root@ip-172-31-66-87:/home/anjanasinghsimp/s3back# nano creds.tf
root@ip-172-31-66-87:/home/anjanasinghsimp/s3back# nano main.tf
```

Note: Use the AWS access credentials provided in the AWS API Access tab in your LMS in your PRACTICE LAB tab as shown in the screenshot below:



AWS access credentials will change when the AWS Lab session expires, every four hours.

2.3 Create **main.tf** under **s3back** and run the code given below:

nano main.tf

2.4 Paste the below code:

```
resource "aws_s3_bucket" "b" {
  bucket = "my-tf-test-bucket"
```

```
  acl    = "private"

  tags = {
    Name        = "My bucket"
    Environment = "Dev"
  }
}
```

Note: Bucket name entered here should be unique globally otherwise it may throw an error while executing the script.

2.3 Run the below commands in the given sequence to add the AWS providers:
terraform init

```
root@ip-172-31-66-87:/home/anjanasinghsimp/s3back# terraform init

Initializing the backend...

Initializing provider plugins...
- Finding latest version of hashicorp/aws...
- Installing hashicorp/aws v3.35.0...
- Installed hashicorp/aws v3.35.0 (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.
```

2.4 Run the command given below to commit TF state:

terraform plan

```
root@ip-172-31-66-87:/home/anjanasinghsimp/s3back# terraform plan

An execution plan has been generated and is shown below.
Resource actions are indicated with the following symbols:
  + create

Terraform will perform the following actions:

  # aws_s3_bucket.bcdefghijklmnopq will be created
  + resource "aws_s3_bucket" "bcdefghijklmnopq" {
      + acceleration_status         = (known after apply)
      + acl                         = "private"
      + arn                         = (known after apply)
      + bucket                      = "my-tf-test-bcdefghijklucket"
      + bucket_domain_name          = (known after apply)
      + bucket_regional_domain_name = (known after apply)
      + force_destroy               = false
```

terraform apply

```
root@ip-172-31-66-87:/home/anjanasinghsimp/s3back# terraform apply

An execution plan has been generated and is shown below.
Resource actions are indicated with the following symbols:
  + create

Terraform will perform the following actions:

  # aws_s3_bucket.bcdefghijklmnopq will be created
  + resource "aws_s3_bucket" "bcdefghijklmnopq" {
      + acceleration_status         = (known after apply)
      + acl                         = "private"
      + arn                         = (known after apply)
      + bucket                      = "my-tf-test-bcdefghijklucket"
      + bucket_domain_name          = (known after apply)
      + bucket_regional_domain_name = (known after apply)
      + force_destroy               = false
      + hosted_zone_id              = (known after apply)
      + id                          = (known after apply)
      + region                      = (known after apply)
```

Enter a value: Yes

```
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_s3_bucket.bcdefghijklmnopq: Creating...
aws_s3_bucket.bcdefghijklmnopq: Creation complete after 1s [id=my-tf-test-bcdefghijklucket]

Apply complete! Resources: 1 added, 0 changed, 0 destroyed.
root@ip-172-31-66-87:/home/anjanasinghsimp/s3back#
```

2.5 Verify the creation of S3 bucket in the AWS Management console