

---

# Lab Exercise 5—Provisioning an S3 Bucket on AWS

---

## Exercise Steps:

### Step 1: Create a New Directory:

Create a new directory to store your Terraform configuration:

```
mkdir Terraform-S3-Demo
cd Terraform-S3-Demo
```

### Step 2: Create the Terraform Configuration File (main.tf):

Create a file named main.tf with the following content:

```
terraform {
  required_providers {
    aws = {
      source = "hashicorp/aws"
      version = "5.31.0"
    }
  }
}

provider "aws" {
  region    = "us-east-1" # Replace with your preferred region
  access_key = "your IAM access key" # Replace with your Access Key
  secret_key = "your secret access key" # Replace with your Secret Key
}
```

This file sets up the Terraform AWS provider.

---

### Step 3: Create a Terraform Configuration File for the S3 Bucket (s3.tf):

---

Create another file named `s3.tf` with the following content:

```
resource "aws_s3_bucket" "my_bucket" {  
  bucket = "my-demo-s3-bucket"  
  tags = {  
    Name      = "Terraform-S3-Bucket"  
  }  
}
```

This file provisions an S3 bucket with a unique name using a random string suffix.

---

#### Step 4: Initialize Terraform:

Run the following command to initialize your Terraform working directory:

```
terraform init
```

```
● mohdanas@Mohds-MacBook-Air LAB_14 % terraform init  
  Initializing the backend...  
  Initializing provider plugins...  
    - Reusing previous version of hashicorp/aws from the dependency lock file  
    - Using previously-installed hashicorp/aws v5.31.0  
  
  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.  
○ mohdanas@Mohds-MacBook-Air LAB_14 % █
```

---

#### Step 5: Review the Plan:

Preview the changes Terraform will make:

```
terraform plan
```

Review the output to ensure it meets your expectations.

## Step 6: Apply the Changes:

Create the resources:

```
terraform apply
```

When prompted, type yes to confirm.

## Step 7: Verify Resources:

1. Log in to your AWS Management Console.
2. Navigate to the **S3** dashboard.
3. Verify that the S3 bucket has been created with the specified configuration.

The screenshot displays the AWS Management Console interface for the Amazon S3 service. The top navigation bar includes the AWS logo, a search bar, and utility icons. The left-hand navigation pane is titled 'Amazon S3' and contains a list of service features. The main content area is divided into two tabs: 'General purpose buckets' (selected) and 'Directory buckets'. Under the 'General purpose buckets' tab, there is a section titled 'General purpose buckets (1)' with an 'Info' link. This section includes buttons for 'Copy ARN', 'Empty', 'Delete', and a prominent orange 'Create bucket' button. Below these buttons, a text box states 'Buckets are containers for data stored in S3.' and a search bar prompts 'Find buckets by name'. A table below the search bar lists the available buckets. The table has columns for 'Name', 'AWS Region', and 'Creation date'. One bucket is listed: 'my-demo-s3-bucket-mohdanas-20250903' in the 'Europe (Ireland) eu-west-1' region, created on 'September 3, 2025, 18:05:47 (UTC+05:30)'.

Name	AWS Region	Creation date
<a href="#">my-demo-s3-bucket-mohdanas-20250903</a>	Europe (Ireland) eu-west-1	September 3, 2025, 18:05:47 (UTC+05:30)

---

## Step 8: Cleanup Resources:

To remove the resources created, run the following command:

```
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_s3_bucket.my_bucket: Destroying... [id=my-demo-s3-bucket-mohdanas-20250903]
aws_s3_bucket.my_bucket: Destruction complete after 2s

Destroy complete! Resources: 1 destroyed.
```

When prompted, type yes to confirm.

---