

Hosting a Static Website with Terraform Automation




Overview of the Project


Hosting a static website on AWS S3 using Terraform provides an automated and efficient way to deploy and manage your web content.

This project leverages Terraform, an infrastructure as code (IaC) tool, to create and configure AWS S3 buckets for hosting static website files.

Install Terraform



 Terraform Install Tutorials Documentation Registry Try Cloud

[< Terraform Home](#)
[Install Terraform](#)
Operating Systems
macOS
Windows
Linux
FreeBSD
OpenBSD
Solaris
Release information
Next steps
Resources

Developer / Terraform
 **Install Terraform** 1.9.8
macOS
Package manager

```
brew tap hashicorp/tap  
brew install hashicorp/tap/terraform
```


Binary download

AMD64 Version: 1.9.8 Download 	ARM64 Version: 1.9.8 Download 
--	--

Create & Configure AWS Provider

provider.tf > provider "aws" > access_key

```
1  terraform {  
8  }  
9  
10 provider "aws" {  
11     region = "ap-south-1"  
12     access_key = "Yourkey"  
13     secret_key = "Yourkey"  
14  
15 }
```

Create a Variable File

provider.tf

.terraform.lock.hcl

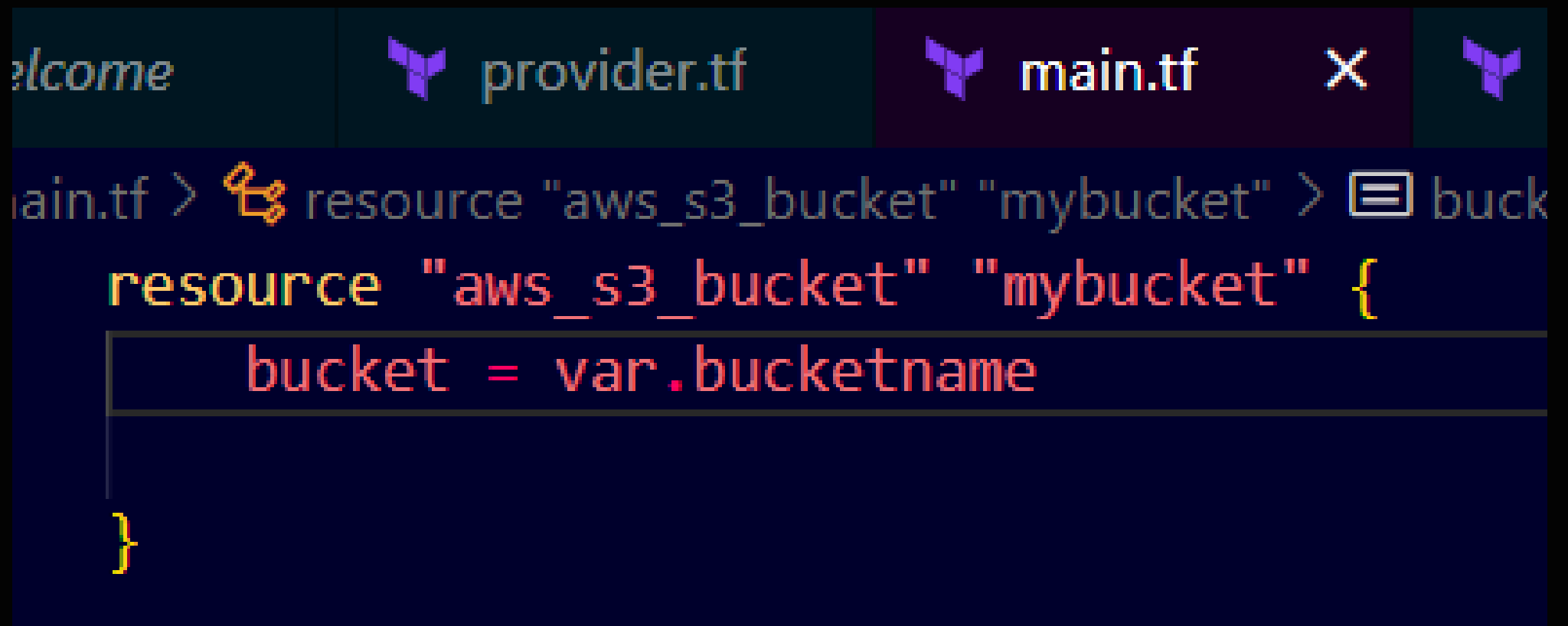
main.tf

5


variable.tf > variable "bucketname"

```
1  variable "bucketname" {  
2    default = "terraform-aws-staticweb-project"  
3  }
```

Creation of S3 Bucket with the help of Variable



The screenshot shows a code editor with a dark blue background. At the top, there are four tabs: 'welcome', 'provider.tf', 'main.tf' (which is active and highlighted in purple), and an empty tab. Below the tabs, the text 'ain.tf >' is visible. The main content is a Terraform configuration snippet for an S3 bucket, using a resource named 'aws_s3_bucket' with the identifier 'mybucket'. The bucket name is assigned to a variable 'var.bucketname'.

```
ain.tf > resource "aws_s3_bucket" "mybucket" >  buck  
resource "aws_s3_bucket" "mybucket" {  
    bucket = var.bucketname  
  
}
```

Initialize Terraform & Validate it

so that Terraform can guarantee to make the same selection you run "terraform init" in the future.

Terraform has been successfully initialized!

You may now begin working with Terraform. Try running "terraform apply" to make any changes that are required for your infrastructure. Terraform should now work.

If you ever set or change modules or backend configuration, rerun this command to reinitialize your working directory. Terraform commands will detect it and remind you to do so if necessary.

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\kkrish\Vsc\t-final-static-web> terraform plan
```

```
Terraform used the selected providers to generate the following execution plan:
+ create
```

```
Terraform will perform the following actions:
```

```
# aws_s3_bucket.mybucket will be created
+ resource "aws_s3_bucket" "mybucket" {
  + acceleration_status    = (known after apply)
  + acl                    = (known after apply)
  + arn                     = (known after apply)
  + bucket                  = "terraform-aws-project"
```

Apply the Configuration

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\kkrish\Vsc\t-final-static-web> terraform apply -auto-approve
```

```
Terraform used the selected providers to generate the following execution plan  
+ create
```

```
Terraform will perform the following actions:
```

```
# aws_s3_bucket.mybucket will be created  
+ resource "aws_s3_bucket" "mybucket" {  
  + acceleration status = (known after apply)
```

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

```
aws_s3_bucket.mybucket: Creating...
```

```
aws_s3_bucket.mybucket: Creation complete after 1s [id=terraform-aws-staticweb
```

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


S3 Bucket

Amazon S3

► **Account snapshot** - *updated every 24 hours* All AWS Regions

Storage lens provides visibility into storage usage and activity trends. [Learn more](#)

General purpose buckets

Directory buckets

General purpose buckets (1) Info All AWS Regions

Buckets are containers for data stored in S3.

Find buckets by name

	Name ▲	AWS Region ▼	IAM Access Analyzer
<input type="radio"/>	terraform-aws-staticweb-project	Asia Pacific (Mumbai) ap-south-1	View analyzer for ap-south-1

Disable ACL

```
provider.tf  terraform.tfstate  main.tf  ×  variable.tf
main.tf > ...
1  resource "aws_s3_bucket" "mybucket" {
4  }
5
6  resource "aws_s3_bucket_ownership_controls" "example" {
7      bucket = aws_s3_bucket.mybucket.id
8      rule {
9          object_ownership = "BucketOwnerPreferred"
10     }
11 }
12
13 resource "aws_s3_bucket_public_access_block" "example" {
14     bucket = aws_s3_bucket.mybucket.id
15
16     block_public_acls       = false
17     block_public_policy     = false
18     ignore_public_acls     = false
19     restrict_public_buckets = false
20 }
21
22 resource "aws_s3_bucket_acl" "example" {
23     depends_on = [
24         aws_s3_bucket_ownership_controls.example,
25         aws_s3_bucket_public_access_block.example,
26     ]
27
28     bucket = aws_s3_bucket.mybucket.id
29     acl    = "public-read"
30 }
31
```

Website Code

```
provider.tf  .terraform.lock.hcl  main.tf  error.html  index.html X  variable.tf

index.html > html > head > style > body
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>Weather App</title>
7      <style>
8          body {
9              font-family: Arial, sans-serif;
10             background-color: #f0f0f0;
11             display: flex;
12             flex-direction: column;
13             align-items: center;
14             justify-content: center;
15             height: 100vh;
16             margin: 0;
17         }
18         .weather-container {
19             background: #fff;
20             padding: 20px;
21             border-radius: 8px;
22             box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
23             text-align: center;
24         }
```

Setup S3 objects of the Website code file



```
main.tf > resource "aws_s3_bucket_website_configuration" "example"

31
32 resource "aws_s3_object" "index" {
33     bucket = aws_s3_bucket.mybucket.id
34     key    = "indexx.html"
35     source = "indexx.html"
36     acl    = "public-read"
37     content_type = "text/html"
38 }
39
40
41 resource "aws_s3_object" "error" {
42     bucket = aws_s3_bucket.mybucket.id
43     key    = "error.html"
44     source = "error.html"
45     acl    = "public-read"
46     content_type = "text/html"
47 }
48
```

Upload S3 object using Terraform Apply

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

C:\Users\kkrish\Vsc\t-final-static-web> terraform apply -auto-approve
aws_s3_bucket.mybucket: Refreshing state... [id=terraform-aws-staticweb-project]
aws_s3_bucket_ownership_controls.example: Refreshing state... [id=terraform-aws-staticweb-project]
aws_s3_bucket_public_access_block.example: Refreshing state... [id=terraform-aws-staticweb-project]
aws_s3_bucket_acl.example: Refreshing state... [id=terraform-aws-staticweb-project,public-read]

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

```
+ source           = indexx.html
+ storage_class    = (known after apply)
+ tags_all         = (known after apply)
+ version_id       = (known after apply)
}

Plan: 1 to add, 0 to change, 0 to destroy.
aws_s3_object.index: Creating...
aws_s3_object.index: Creation complete after 0s [id=indexx.html]

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


S3 Bucket with the Website Code file

[Amazon S3](#) > [Buckets](#) > terraform-aws-staticweb-project

terraform-aws-staticweb-project [Info](#)

[Objects](#)

[Properties](#)

[Permissions](#)

[Metrics](#)

[Management](#)



[Access Points](#)

Objects (2) [Info](#)

  Copy S3 URI  Copy URL  Download  Open  Delete  Actions  Create folder 

Objects are the fundamental entities stored in Amazon S3. You can use [Amazon S3 inventory](#) to get a list of all objects in your bucket. For others to access your objects, you'll need to grant them permissions. [Learn more](#)

 Find objects by prefix

<input type="checkbox"/>	Name	Type	Last modified	Size	Storage
<input type="checkbox"/>	 error.html	html	November 15, 2024, 00:45:44 (UTC+05:30)	0 B	Standard
<input type="checkbox"/>	 indexx.html	html	November 15, 2024, 00:40:43 (UTC+05:30)	1.5 KB	Standard

Setup Website Configuration for Web Static Hosting

```
resource "aws_s3_bucket_website_configuration" "example" {  
  bucket = aws_s3_bucket.mybucket.id  
  
  index_document {  
    suffix = "index.html"  
  }  
  
  error_document {  
    key = "error.html"  
  }  
  
  routing_rule {  
    condition {  
      key_prefix_equals = "docs/"  
    }  
    redirect {  
      replace_key_prefix_with = "documents/"  
    }  
  }  
}
```

Apply Website Configuration

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\kkrish\Vsc\t-final-static-web> terraform apply -auto-approve
aws_s3_bucket.mybucket: Refreshing state... [id=terraform-aws-staticweb-project]
aws_s3_bucket_public_access_block.example: Refreshing state... [id=terraform-aws-staticweb-project]
aws_s3_bucket_ownership_controls.example: Refreshing state... [id=terraform-aws-staticweb-project]
aws_s3_object.index: Refreshing state... [id=indexx.html]
aws_s3_bucket_acl.example: Refreshing state... [id=terraform-aws-staticweb-project,public-read]
```

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_s3_bucket_website_configuration.example will be created
+ resource "aws_s3_bucket_website_configuration" "example" {
  + bucket          = "terraform-aws-staticweb-project"
  + id              = (known after apply)
  + routing_rules   = (known after apply)
  + website_domain  = (known after apply)
```

```
aws_s3_bucket_website_configuration.example: Creation complete after 1s [id=terraform-aws-staticweb-project]
```

```
Apply complete! Resources: 2 added, 0 changed, 0 destroyed.
```



Enabled Static Website hosting using Automation

Static website hosting

Use this bucket to host a website or redirect requests. [Learn more](#) 



We recommend using AWS Amplify Hosting for static website hosting

Deploy a fast, secure, and reliable website quickly with AWS Amplify Hosting. Learn more about [Amplify Hosting](#)  or [View your existing Amplify apps](#) 


S3 static website hosting

Enabled

Hosting type

Bucket hosting

Bucket website endpoint

When you configure your bucket as a static website, the website is available at the AWS Region-specific website endpoint of the bucket. [Learn more](#) 

 <http://terraform-aws-staticweb-project.s3-website.ap-south-1.amazonaws.com> 

Result & Outcome of the Project

