

Project 3: Creating a Static Website Using S3

What is Amazon S3?

- Amazon Simple Storage Service is storage for the internet. It is designed to make web-scale computing easier.
- It has a simple web services interface that you can use to store and retrieve any amount of data, at any time, from anywhere on the web.

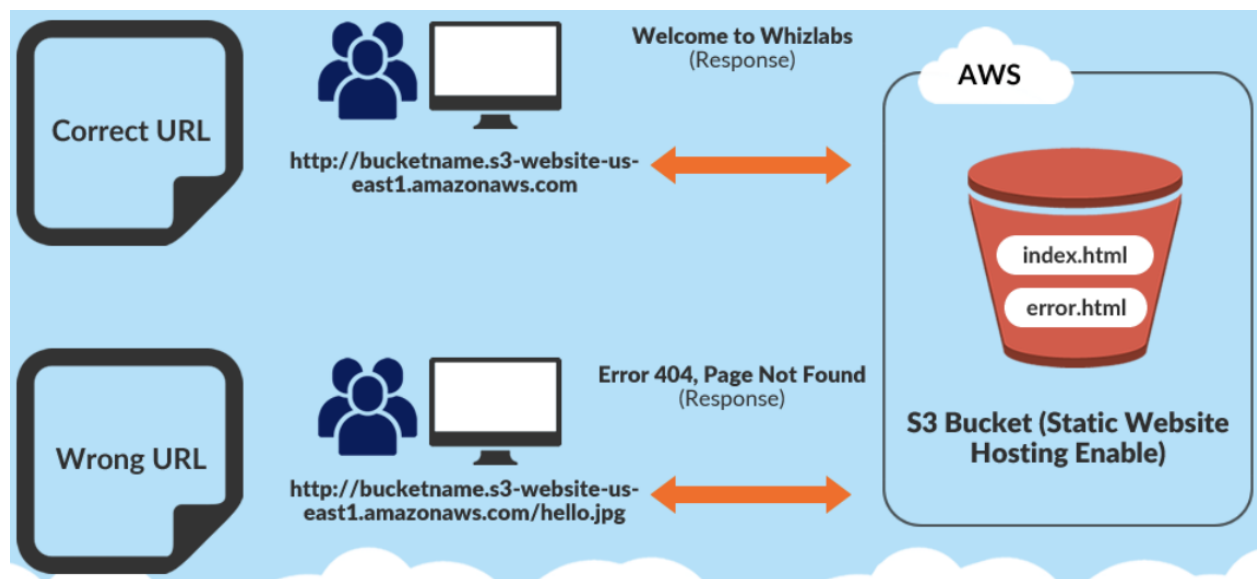
What is a Static Website?

- A static web page is a web page that is delivered to the user's web browser exactly as stored. It holds fixed content, where each page is coded in HTML, and displays the same information to every visitor.

Objective of this Project:

- We will be creating and launching a static website using S3.

Architecture Diagram



Step 1: Create a Bucket

- We start with creating an S3 bucket with a bucket name that is supposed to be unique. We also make sure it is available and accessible publicly.

General configuration

Bucket name

satinbucket123

Bucket name must be unique and must not contain spaces or uppercase letters. [See rules for bucket naming](#) 

AWS Region

US East (N. Virginia) us-east-1

Copy settings from existing bucket - *optional*

Only the bucket settings in the following configuration are copied.

Choose bucket

☐ Block all public access

Turning this setting on is the same as turning on all four settings below. Each of the following settings are independent of one another.

☐ Block public access to buckets and objects granted through *new* access control lists (ACLs)

S3 will block public access permissions applied to newly added buckets or objects, and prevent the creation of new public access ACLs for existing buckets and objects. This setting doesn't change any existing permissions that allow public access to S3 resources using ACLs.

☐ Block public access to buckets and objects granted through *any* access control lists (ACLs)

S3 will ignore all ACLs that grant public access to buckets and objects.

☐ Block public access to buckets and objects granted through *new* public bucket or access point policies

S3 will block new bucket and access point policies that grant public access to buckets and objects. This setting doesn't change any existing policies that allow public access to S3 resources.


☐ Block public and cross-account access to buckets and objects through *any* public bucket or access point policies

S3 will ignore public and cross-account access for buckets or access points with policies that grant public access to buckets and objects.

Step 2: Enable Static Website Hosting

- We start with allowing the S3 to enable static web hosting. And then we specify the default page of the website which can be index.html. If an error occurs, we specify what the website should return if an error were to occur.



Static website hosting

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

Static website hosting

- ☐ Disable
- ☒ Enable

Hosting type

- ☒ Host a static website
Use the bucket endpoint as the web address. [Learn more](#) 
- ☐ Redirect requests for an object
Redirect requests to another bucket or domain. [Learn more](#) 

Index document

Specify the home or default page of the website.

index.html



Error document - *optional*

This is returned when an error occurs.

error.html

Here's the endpoint: <http://satinbucket123.s3-website-us-east-1.amazonaws.com>

We upload the error and index files on to the S3 bucket:

Name ▲	Type ▼	Last modified ▼	Size ▼	Storage class ▼
 error.html	html	June 24, 2021, 07:52:21 (UTC-07:00)	157.0 B	Standard
 index.html	html	June 24, 2021, 07:52:22 (UTC-07:00)	155.0 B	Standard

We then edit the bucket policy:

Policy

```
1 {  
2  
3   "Id": "Policy1",  
4  
5   "Version": "2012-10-17",  
6  
7   "Statement": [  
8  
9     {  
10  
11       "Sid": "Stmt1",  
12  
13       "Action": [  
14  
15         "s3:GetObject"  
16  
17       ],  
18  
19       "Effect": "Allow",  
20  
21       "Resource": "arn:aws:s3:::satinbucket123/*",  
22  
23       "Principal": "*"   
24  
25     }  
26   ]  
27 }
```

Step 3: Testing The Website

We copy the website URL and run it on the browser. Since we see the contents of the index.html file, it is shown that the website is working.

Welcome to Whizlabs

Its a sample index HTML page of static website sample on AWS S3

Step 4: Testing the Error Page

We test the error page by typing a bunch of random characters on the URL and we see that the error.html file has popped up.

Error 404, Page Not Found!

Its a Sample Error Page of static website sample on AWS S3