

AWS training - Serverless

Hands On #8 - S3 - Static website



Overview

- In this hands-on you'll configure Amazon Simple Storage Service (S3) to host the static resources for your web application.
- All of your static web content including HTML, JavaScript, images and other files will be stored in Amazon S3.
- You will then access your site using the public website URL exposed by Amazon S3. You don't need to run any web servers or use other services in order to make your site available.
- For the purposes of this module you'll use the Amazon S3 website endpoint URL that we supply. It takes the form `http://{your-bucket-name}.s3-website.{region}.amazonaws.com`.

Let's go!

1. Select the region US EAST (N. Virginia)

N. Virginia ▾

2. Create a S3 bucket

3. Make public

4. Upload content, the website is in the following zip :

<https://s3.amazonaws.com/awstacktraining-serverless-resources/code-templates/dist.zip>

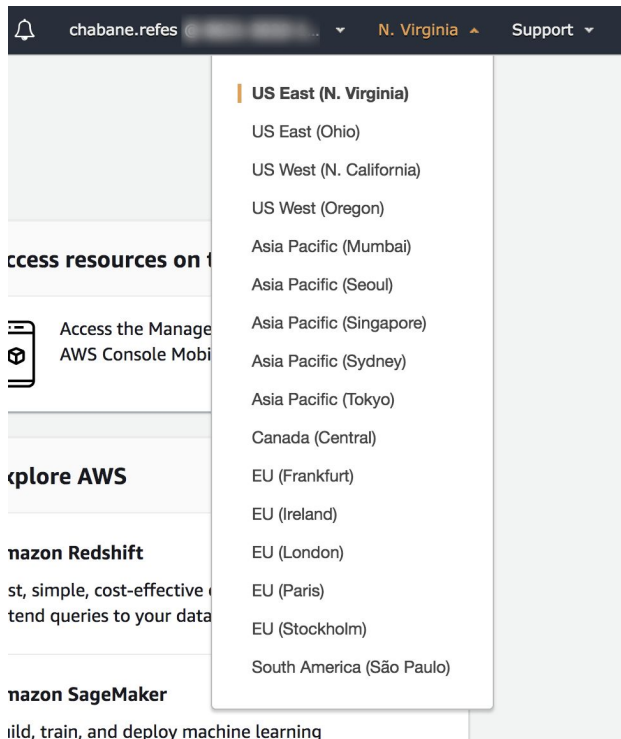
5. Add a Bucket Policy to Allow Public Reads

6. Enable Website Hosting

7. Add Cors

8. Validate your implementation

Hint #1 - Select a region



Hint #2 - Create a S3 bucket

1. In the AWS Management Console choose Services then select S3 under Storage.
2. Choose + Create Bucket
3. Provide a globally unique name for your bucket such as `serverless-training-website-xxx`. If you get an error that your bucket name already exists, try adding additional numbers or characters until you find an unused name.
4. Choose Create in the lower left of the dialog without selecting a bucket to copy settings from.

Hint #3 - Upload Content

1. Download an archive of this repository using this link:
<https://s3.amazonaws.com/awstacktraining-serverless-resources/code-templates/dist.zip>
2. Unzip the archive you downloaded on your local machine.
3. Update config.json with the proper values.
4. Open the AWS Management Console in Chrome. Choose Services then select S3 under Storage.
5. Select the bucket you created in the previous step and ensure you are viewing the Objects tab.
6. Open either Windows File Explorer or macOS Finder and browse to the expanded contents of the zip file you downloaded in the first step.
7. Browse to the unzipped directory on your local machine.

Hint #3 - Upload Content

7. Select all of the files and subdirectories under the website directory. Ensure that the website directory itself is not selected.
8. Drag and drop the selected files from your local filesystem to the content under the Objects tab in the S3 console.
9. Choose Upload in the lower left of the dialog box that appears.
10. Wait for the upload to complete, and ensure you see the contents of the website directory listed in the S3 console. If you only see a single website directory, you should delete it from your bucket and follow these instructions again ensuring that you select only the contents of the directory before dragging and dropping into the S3 console.

Config.json

```
{
  "api": {
    "preSignedEndpoint": "https://b46qfj0h8e.execute-api.us-east-1.amazonaws.com/dev/upload",
    "resultEndpoint": "https://b46qfj0h8e.execute-api.us-east-1.amazonaws.com/dev/results"
  },
  "bucket": {
    "thumbnail": "https://s3.amazonaws.com/serverless-training-thumb"
  }
}
```


Hint #4 - Add a Bucket Policy to Allow Public Reads

1. In the S3 console, select the name of the bucket you created in **Hint #1**.
2. Choose the Permissions tab, then choose Bucket Policy.
3. Enter the following policy document into the bucket policy editor replacing `[YOUR_BUCKET_NAME]` with the name of the bucket you created in section 1:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": "*",
      "Action": "s3:GetObject",
      "Resource": "arn:aws:s3:::[YOUR_BUCKET_NAME]/*"
    }
  ]
}
```

4. Choose Save to apply the new policy.

Hint #5 - Enable Website Hosting

1. From the bucket detail page in the S3 console, choose the Properties tab.
2. Choose the Static website hosting card.
3. Select Use this bucket to host a website and enter `index.html` for the Index document. Leave the other fields blank.
4. Note the Endpoint URL at the top of the dialog before choosing Save. You will use this URL throughout the rest of the hands-on to view your web application. From here on this URL will be referred to as your website's base URL.
5. Click Save to save your changes.

Hint #5 - Add CORS

```
<?xml version="1.0" encoding="UTF-8"?>
<CORSConfiguration xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <CORSRule>
    <AllowedOrigin>*</AllowedOrigin>
    <AllowedMethod>GET</AllowedMethod>
    <AllowedMethod>PUT</AllowedMethod>
    <AllowedMethod>POST</AllowedMethod>
    <MaxAgeSeconds>3000</MaxAgeSeconds>
    <AllowedHeader>*</AllowedHeader>
  </CORSRule>
</CORSConfiguration>
```

Hint #6 - Validate your implementation

After completing these implementation steps you should be able to access your static website by visiting the website endpoint URL for your S3 bucket.

Références

<https://aws.amazon.com/getting-started/projects/build-serverless-web-app-lambda-apigateway-s3-dynamodb-cognito/module-1/>