Advanced Cloud DevOps

Nanodegree Program

Prepared by

Aya RABIH Mostafa

INFORMATION FOR USER

- o NAME: Aya Rabih Mostafa

- o Project : Deploy Static Website on AWS

- o USER: 598824656916

- o LINK PROJECT: https://d3g7c4ah6f8c9c.cloudfront.net/

PROJECT OVERVIEW

The cloud is perfect for hosting static websites that only include HTML, CSS, and JavaScript files that require no server-side processing. The whole project has two major intentions to implement:

Hosting a static website on S3 and

Accessing the cached website pages using Cloud Front content delivery network (CDN) service. Recall that Cloud Front offers low latency and high transfer speeds during website rendering.

Note that Static website hosting essentially requires a public bucket, whereas the Cloud Front can work with public and private buckets.

➢ In this project, you will deploy a static website to AWS by performing the following steps:

1. You will create a public S3 bucket and upload the website files to your bucket.

2. You will configure the bucket for website hosting and secure it using IAM policies.

3. You will speed up content delivery using AWS's content distribution network service, Cloud Front.

4. You will access your website in a browser using the unique Cloud Front endpoint.

1) <u>You will create a public S3 bucket and upload the website files to your bucket.</u>

➢ Click create bucket

➢ Click create bucket name

➢ Click create Region you will create your bucket there

➢ Click create Acls disabled

Amazon S3 > Buckets > Create bucket

Create bucket Info
Buckets are containers for data stored in S3. Learn more ⤢

General configuration

Bucket name

my-598824656916 bucket

Bucket name must be globally unique and must not contain spaces or uppercase letters. See rules for bucket naming ⤢

AWS Region

US East (Ohio) us-east-2 ▼

Copy settings from existing bucket - *optional*
Only the bucket settings in the following configuration are copied.

Choose bucket

Object Ownership Info
Control ownership of objects written to this bucket from other AWS accounts and the use of access control lists (ACLs). Object ownership determines who can specify access to objects.

◉ ACLs disabled (recommended)
All objects in this bucket are owned by this account. Access to this bucket and its objects is specified using only policies.

○ ACLs enabled
Objects in this bucket can be owned by other AWS accounts. Access to this bucket and its objects can be specified using ACLs.

Object Ownership
Bucket owner enforced

➢ Click to remove block all public access

➢ Click to sure that you want be public

Block Public Access settings for this bucket

Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access to this bucket and its objects is blocked, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to this bucket or objects within, you can customize the individual settings below to suit your specific storage use cases. Learn more ↗

☐ **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.

⚠ **Turning off block all public access might result in this bucket and the objects within becoming public**
AWS recommends that you turn on block all public access, unless public access is required for specific and verified use cases such as static website hosting.

☑ I acknowledge that the current settings might result in this bucket and the objects within becoming public.

➤ Click create bucket

➢ Now we create bucket

2. You will configure the bucket for website hosting and secure it using IAM policies

- ➢ Upload files to S3 Bucket
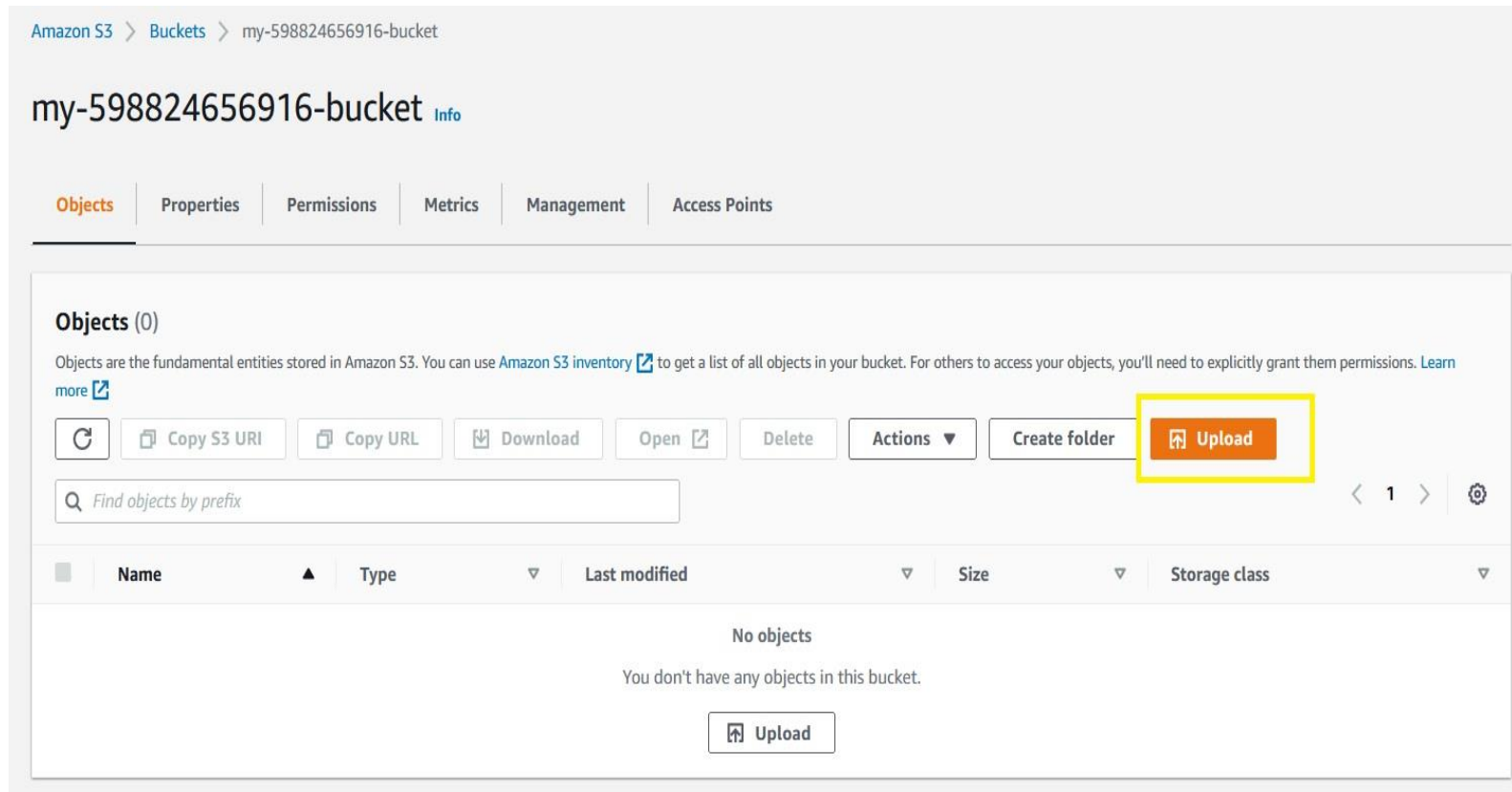
➢ Click "Add files" to upload the index.html file, and click "Add folder" to upload the css, img, and vendor folders.

Upload Info

Add the files and folders you want to upload to S3. To upload a file larger than 160GB, use the AWS CLI, AWS SDK or Amazon S3 REST API. Learn more ⤷

Drag and drop files and folders you want to upload here, or choose **Add files**, or **Add folders**.

Files and folders (0)   Remove   **Add files**   **Add folder**
All files and folders in this table will be uploaded.

🔍 Find by name   ‹ 1 ›

☐	Name ▲	Folder ▽	Type ▽	Size ▽

No files or folders
You have not chosen any files or folders to upload.

حجم الملف	تاريخ آخر تعديل	الاسم	
KB 342	-	MACOSX__	📁
KB 15	-	css	📁
KB 418	-	img	📁
MB 14	-	vendor	📁
KB 6	2019/05/18	index.html	📄
382 بايت	2019/05/18	README.txt	📄

➢ Now update files to S3 Bucket

➢ Now complete update files to S3 Bucket

1. Click on the "Permissions" tab.
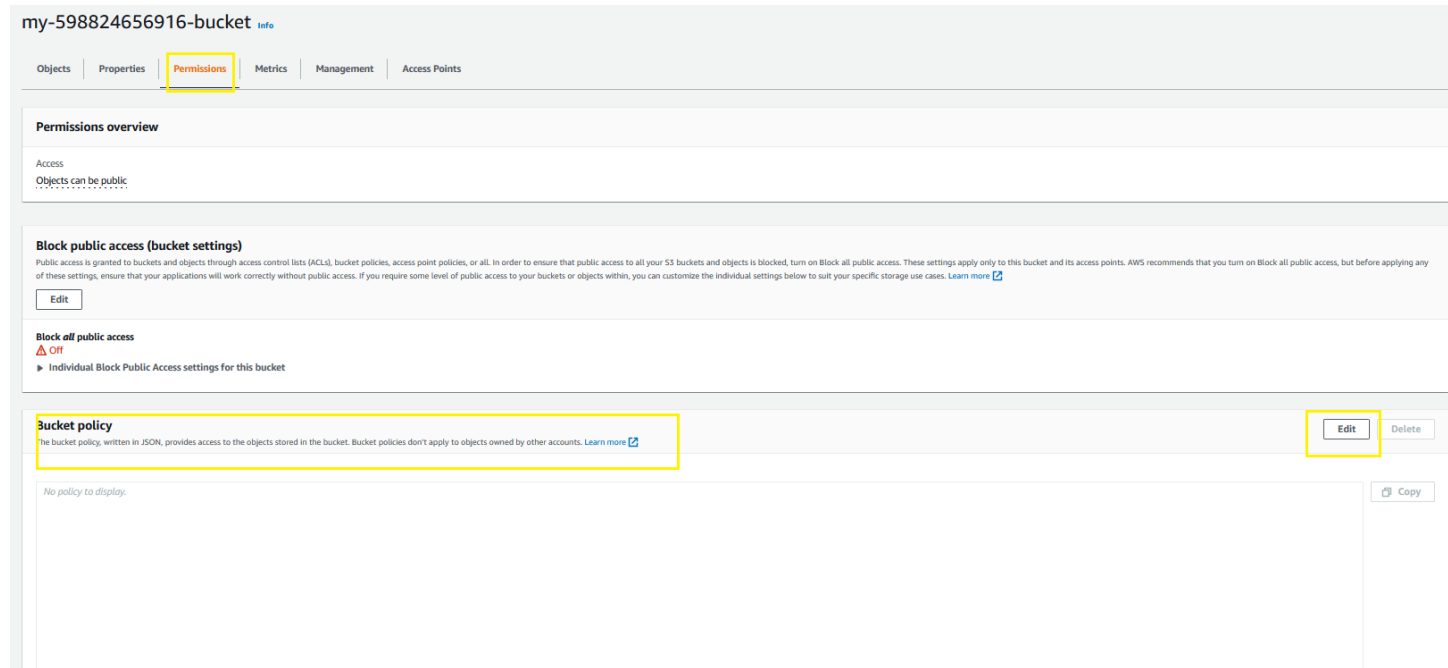
2. The "Bucket Policy" section shows it is empty. Click on the Edit button.

```
{
"Version":"2012-10-17",
"Statement":[
  {
    "Sid":"AddPerm",
    "Effect":"Allow",
    "Principal": "*",
    "Action":["s3:GetObject"],
    "Resource":["arn:aws:s3::: my-598824656916-bucket/*"]
  }
]
}
```

3- Enter the following bucket policy replacing my-598824656916-bucketwith the name of your bucket and click "Save".

Bucket ARN

arn:aws:s3:::my-598824656916-bucket

Policy

```json
1  {
2    "Version":"2012-10-17",
3    "Statement":[
4      {
5        "Sid":"AddPerm",
6        "Effect":"Allow",
7        "Principal": "*",
8        "Action":["s3:GetObject"],
9        "Resource":["arn:aws:s3:::my-598824656916-bucket/*"]
10     }
11   ]
12 }
```

Edit statement

Select a statement

Select an existing statement in the policy or add a new statement.

+ Add new statement

+ Add new statement

JSON   Ln 12, Col 2

Cancel   Save changes

CONFIGURE S3 BUCKET

1. Go to the Properties tab and then scroll down to edit the Static website hosting section
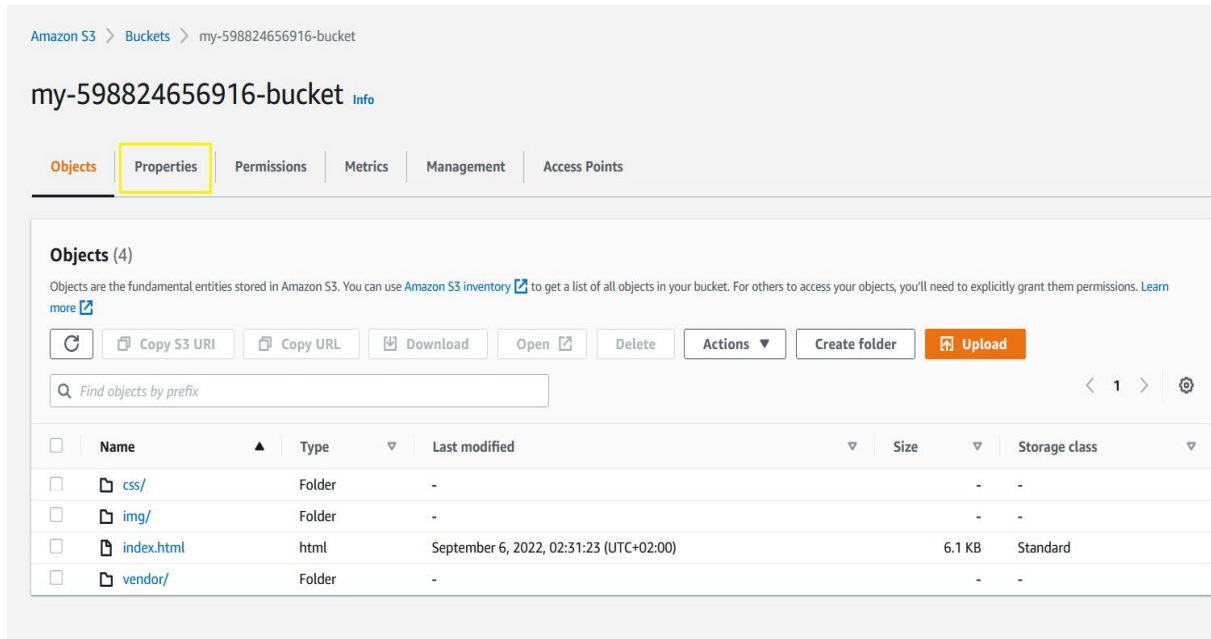
2. Click on the "Edit" button to see the Edit static website hosting screen.
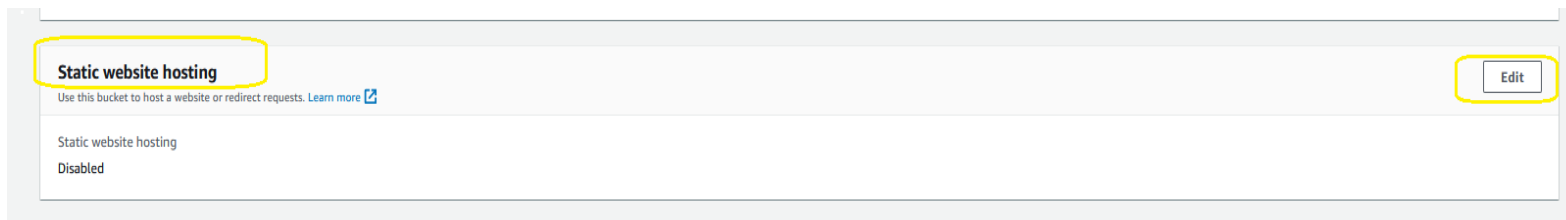
3. Now, enable the Static website hosting, and provide the default home page and error page for your website.

4. Check the **Static website hosting** section again under the **Properties** tab. You must now be able to view the website endpoint as shown below:

5. Copy the website endpoint for future use (WE GIT IT)
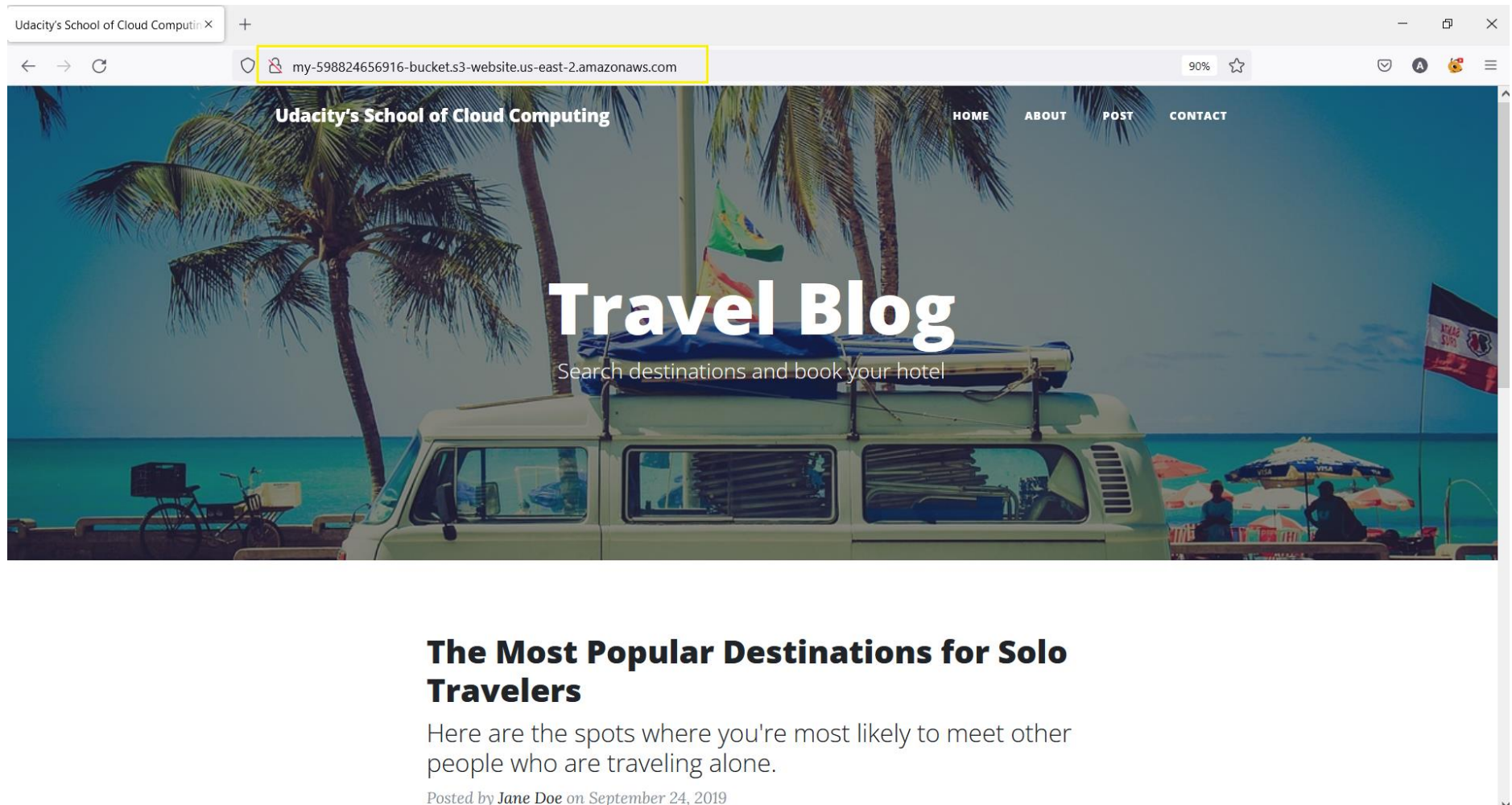
Distribute Website via CloudFront

1. Select "cloud front"
2. Click "Create Distribution".

3. Configure distribution

Default cache behavior

Path pattern   Info

Default (*)

Compress objects automatically   Info

○ No

● Yes

Viewer

Viewer protocol policy

○ HTTP and HTTPS

● Redirect HTTP to HTTPS

○ HTTPS only

Allowed HTTP methods

● GET, HEAD

○ GET, HEAD, OPTIONS

○ GET, HEAD, OPTIONS, PUT, POST, PATCH, DELETE

Restrict viewer access

If you restrict viewer access, viewers must use CloudFront signed URLs or signed cookies to access your content.

● No

○ Yes

Cache key and origin requests

We recommend using a cache policy and origin request policy to control the cache key and origin requests.

● Cache policy and origin request policy (recommended)

○ Legacy cache settings

┌─ Cache policy
│ Choose an existing cache policy or create a new one.
│
│ CachingOptimized
│ Default policy when CF compression is enabled   ▼   ↻
│
│ Create policy 🗗   View policy 🗗
│
└─ Origin request policy - *optional*
   Choose an existing origin request policy or create a new one.

5. Click distribution

6. now all things is perfect distribution
7. get link cope and past in new tap

✓ Successfully created new distribution.

ⓘ Do you have 30 seconds to share your experience? Help us improve CloudFront.

CloudFront > Distributions > E2LTLHC6W1WDFR

E2LTLHC6W1WDFR

General   Origins   Behaviors   Error pages   Geographic restrictions   Invalidations   Tags

Details

Distribution domain name	ARN	Last modified
🗐 d3g7c4ah6f8c9c.cloudfront.net	🗐 arn:aws:cloudfront::598824656916:distribution/E2LTLHC6W1WDFR	September 6, 2022 at 1:04:59 AM UTC

Settings   Edit

Description	Alternate domain names	Standard logging
-	-	Off
Price class		Cookie logging
Use all edge locations (best performance)		Off
Supported HTTP versions		Default root object
HTTP/2, HTTP/1.1, HTTP/1.0		-
AWS WAF		
-		

✓ NOW JOB DONE

https://d3g7c4ah6f8c9c.cloudfront.net/