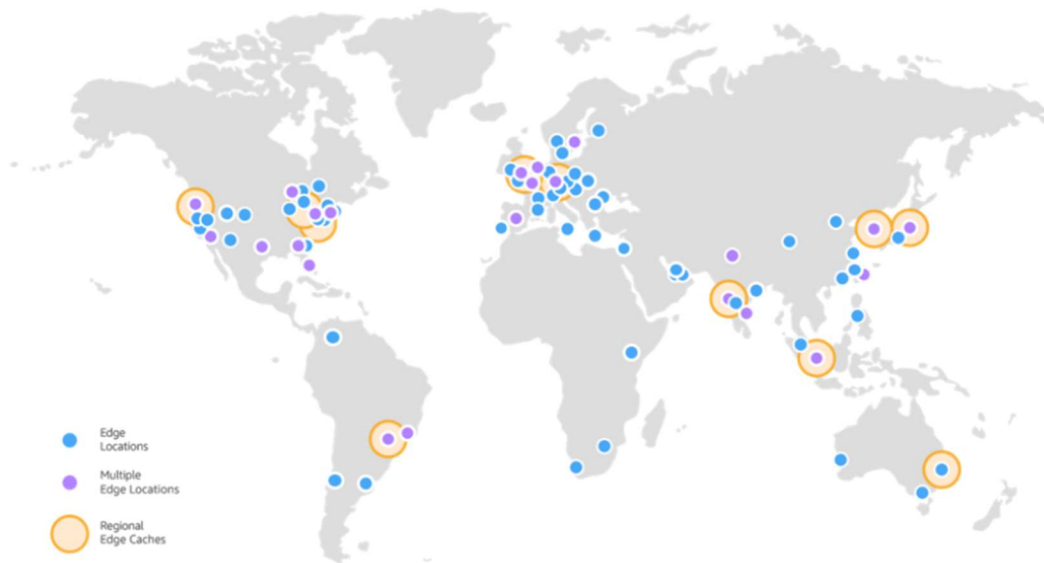


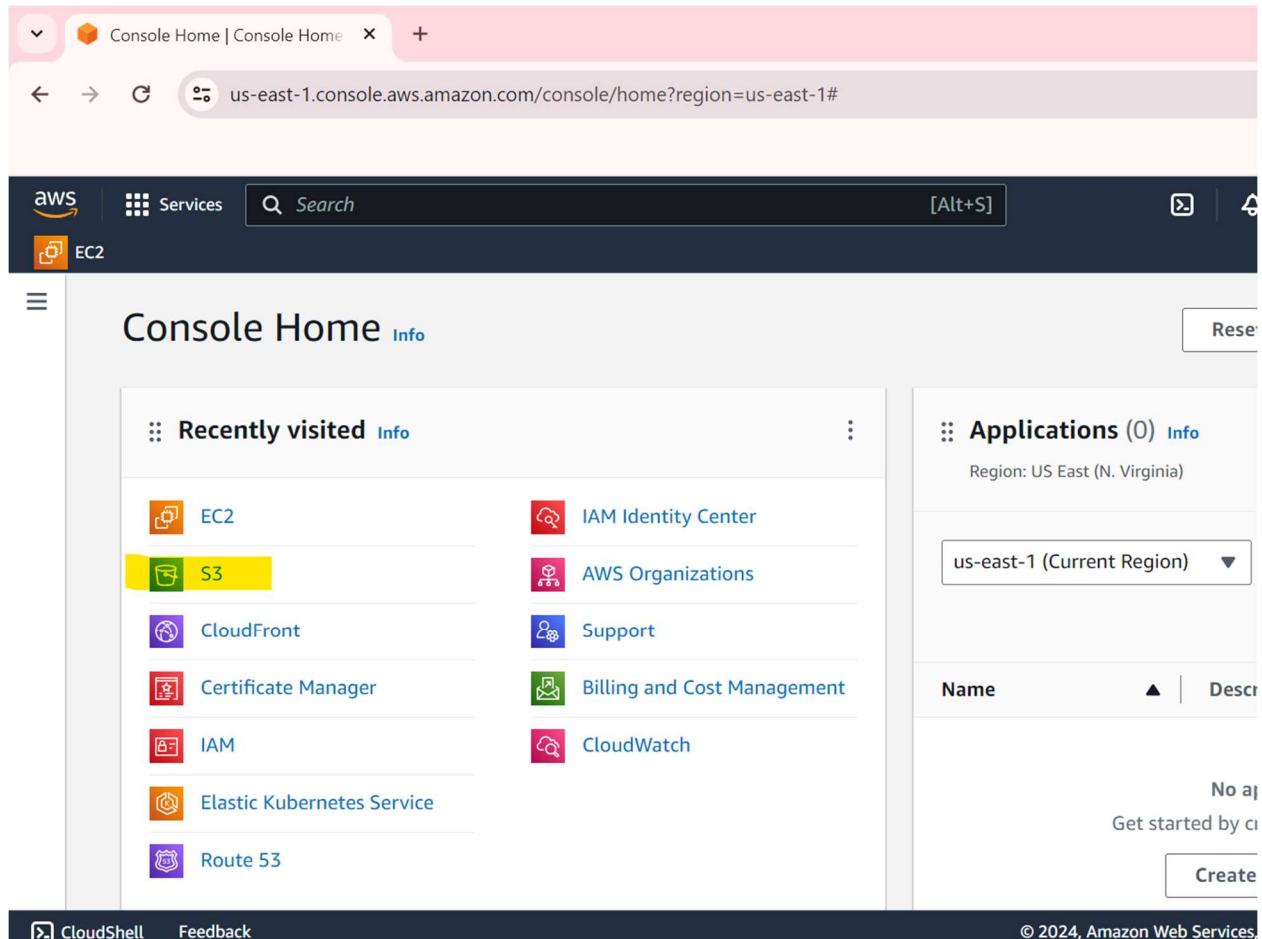
INTRODUCTION:

Today, I quickly reviewed S3 Bucket and CloudFront by deploying a website on AWS in just 20 minutes. I'd like to share the step-by-step process, so if you want to do a quick deployment and get an overview as well, I've detailed everything here. I've also created a small website that you can clone from my GitHub repository at <https://github.com/Sara951/Quick-Project-to-Deploy-Cloudfront-S3.git> and follow all the steps in the following Doc. The documentation covers setting up a static website on an S3 bucket for hosting, configuring CloudFront as your CDN, ensuring HTTPS with ACM, and setting up DNS records.

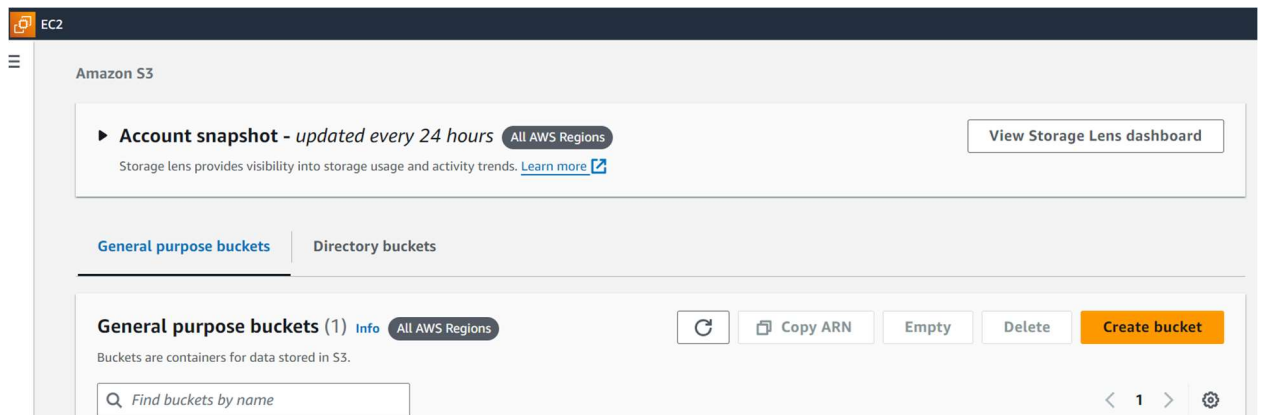
Ready to dive in and get your site live today? Let's get started!



Login to the IAM user of your aws



Then click on create a bucket



I gave it a the same as my website name, and default region which is N.Virginia

AWS Region

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

Bucket type [Info](#)

☒ General purpose

Recommended for most use cases and access patterns. General purpose buckets are the original S3 bucket type. They allow a mix of storage classes that redundantly store objects across multiple Availability Zones.

☐ Directory - *New*

Recommended for low-latency use cases. These buckets use only the S3 Express One Zone storage class, which provides faster processing of data within a single Availability Zone.

Bucket name [Info](#)

www.parsdesign.xyz

Bucket name must be unique within the global namespace and follow the bucket naming rules. [See rules for bucket naming](#)

Copy settings from existing bucket - *optional*

Only the bucket settings in the following configuration are copied.

[Choose bucket](#)

Format: s3://bucket/prefix

Ownership: ACLs disabled (default setting)

Block all public access(default) for security reasons

You can leave everything on its own default settings and click on “Create bucket” at the bottom

► **Advanced settings**

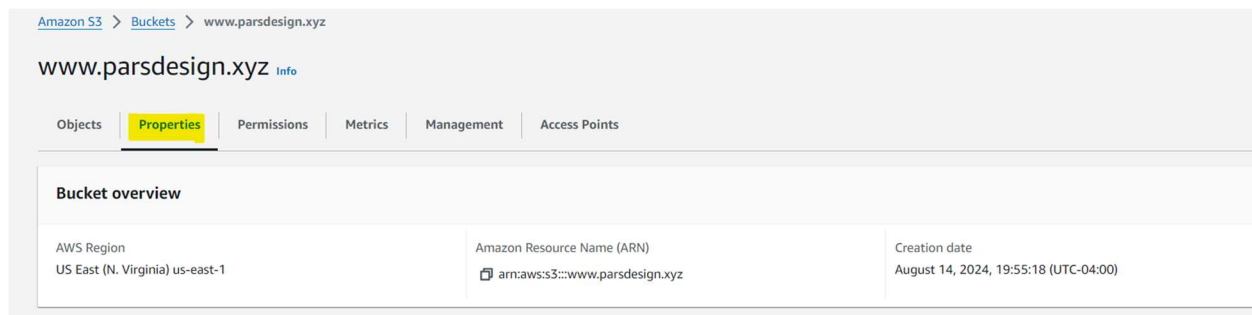
After creating the bucket, you can upload files and folders to the bucket, and configure additional bucket settings.

Cancel [Create bucket](#)

And then here we are, we have the bucket

General purpose buckets				Directory buckets	
General purpose buckets (1) Info All AWS Regions				Refresh Copy ARN Empty Delete Create bucket	
Buckets are containers for data stored in S3.					
Find buckets by name				Previous 1 Next Settings	
Name	▲	AWS Region	▼	IAM Access Analyzer	Creation date
www.parsdesign.xyz		US East (N. Virginia) us-east-1		View analyzer for us-east-1	August 14, 2024, 19:55:18 (UTC-04:00)

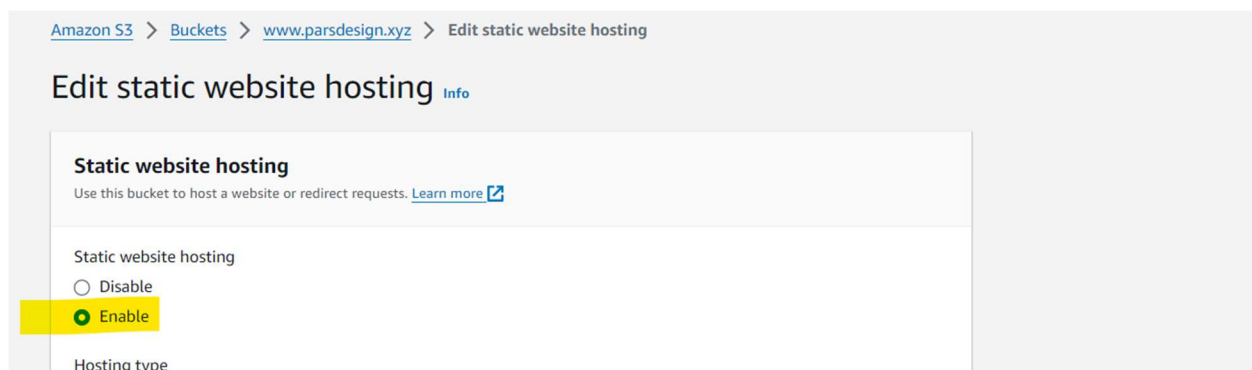
Now open the bucket you just created! And go to the properties tab



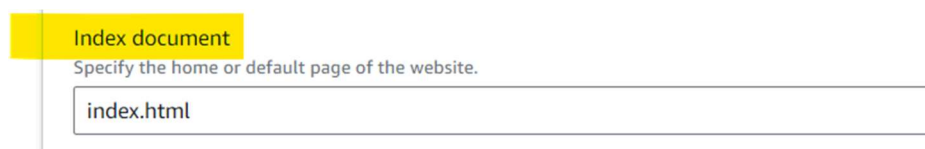
Scroll down to see Static website and click on edit and in static website hosting click on enable



Then



Then fill out index document field



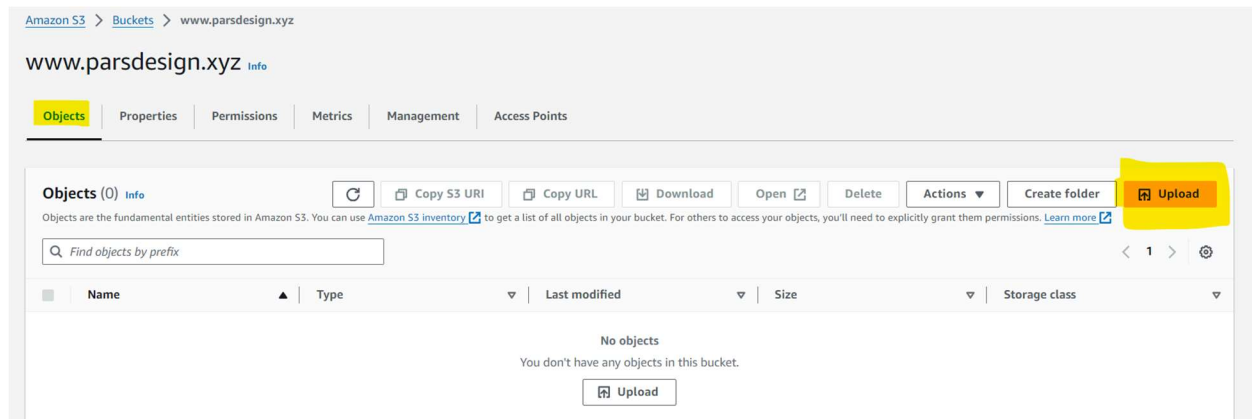
And save changes

After all these things you can go to object section and upload the files. I quickly created a simple website for this purpose, You can use my website and download it from my git

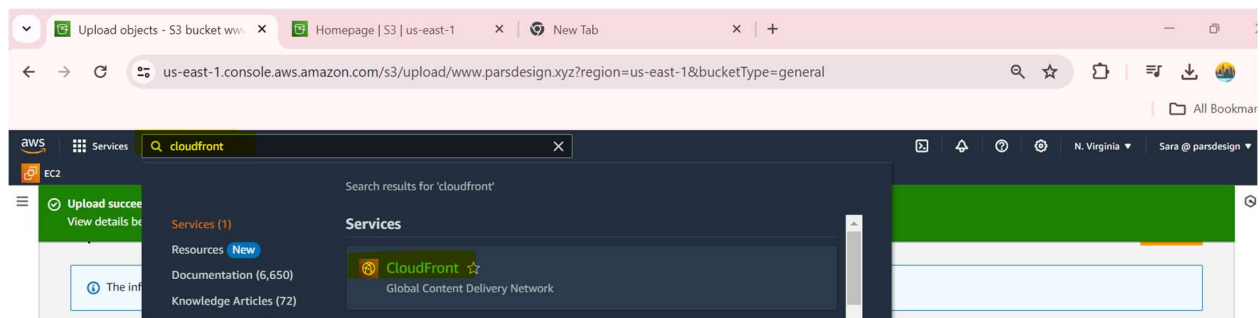
<https://github.com/Sara951/Quick-Project-to-Deploy-Cloudfront-S3>

<https://github.com/Sara951/Quick-Project-to-Deploy-Cloudfront-S3.git>

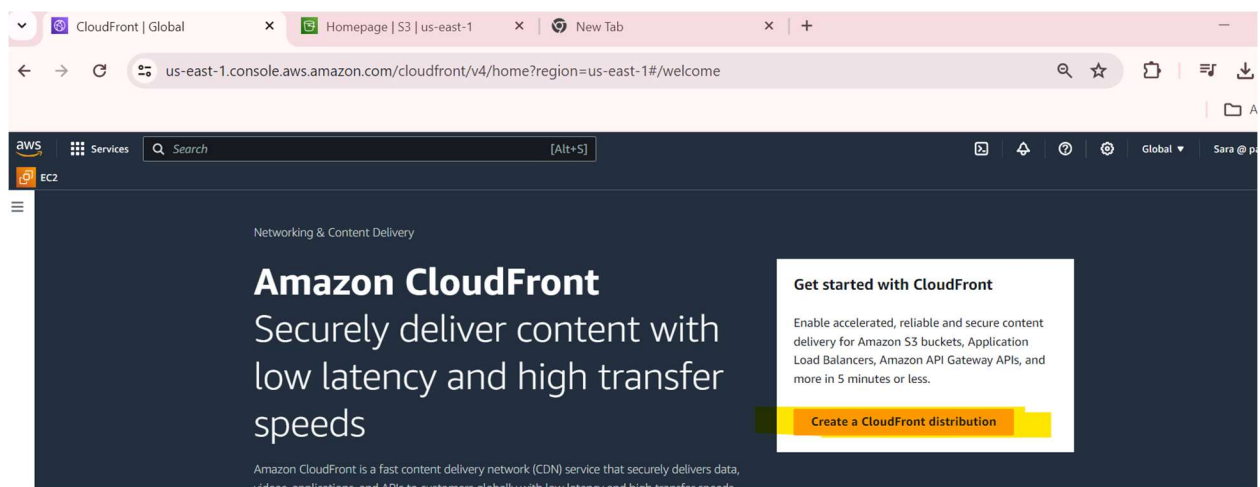
and upload it here



After uploading the files go to cloud front section



And create a cloud front distribution



Choose origin domain which is the bucket and will be shown there

Origin

Origin domain

Choose an AWS origin, or enter your origin's domain name.

Q Choose origin domain

Amazon S3

www.parsdesign.xyz.s3.amazonaws.com

Elastic Load Balancer

If you scroll the Name section is also populated with the same thing,

But for the Origin Access we need to set it up in Legacy access identities to be used to securely grant CloudFront access to your S3 bucket without making the bucket publicly accessible

Name

Enter a name for this origin.

www.parsdesign.xyz.s3.us-east-1.amazonaws.com

Origin access [Info](#)

☐ Public

Bucket must allow public access.

☐ Origin access control settings (recommended)

Bucket can restrict access to only CloudFront.

☒ Legacy access identities

Use a CloudFront origin access identity (OAI) to access the S3 bucket.

And then we click on create new OAI (origin access identity) and we select again the one we created for OAI.

Origin access identity

Select an existing origin access identity (recommended) or create a new identity.

www.parsdesign.xyz.s3.us-east-1.amazonaws.com

Create new OAI

And we select update the “Bucket Policy”

Bucket policy

Update the S3 bucket policy to allow read access to the OAI.

☐ No, I will update the bucket policy

☒ Yes, update the bucket policy

Then we change the “Viewer Protocol Policy” to:

Viewer

Viewer protocol policy

- ☐ HTTP and HTTPS
- ☒ Redirect HTTP to HTTPS
- ☐ HTTPS only

Also for “ Allowed HTTP methods”

Allowed HTTP methods

- ☐ GET, HEAD
- ☐ GET, HEAD, OPTIONS
- ☒ GET, HEAD, OPTIONS, PUT, POST, PATCH, DELETE

And in the “Alternate Domain Name” again your domain name:

Settings

Price class

[Info](#)

Choose the price class associated with the maximum price that you want to pay.

- ☒ Use all edge locations (best performance)
- ☐ Use only North America and Europe
- ☐ Use North America, Europe, Asia, Middle East, and Africa

Alternate domain name (CNAME) - optional

Add the custom domain names that you use in URLs for the files served by this distribution.

www.parsdesign.xyz

Remove

Add item

Then we need to activate WAF

Web Application Firewall (WAF) [Info](#)

☒ Enable security protections

Keep your application secure from the most common web threats and security vulnerabilities using AWS WAF. Blocked requests are stopped before they reach your web servers.

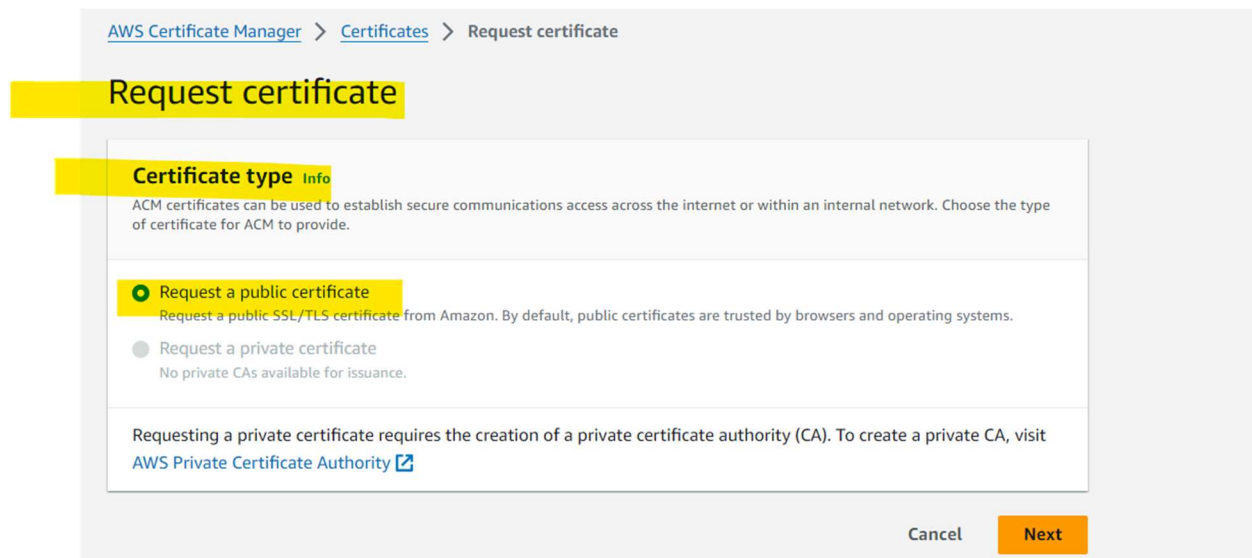
☐ Do not enable security protections

Select this option if your application does not need security protections from AWS WAF.

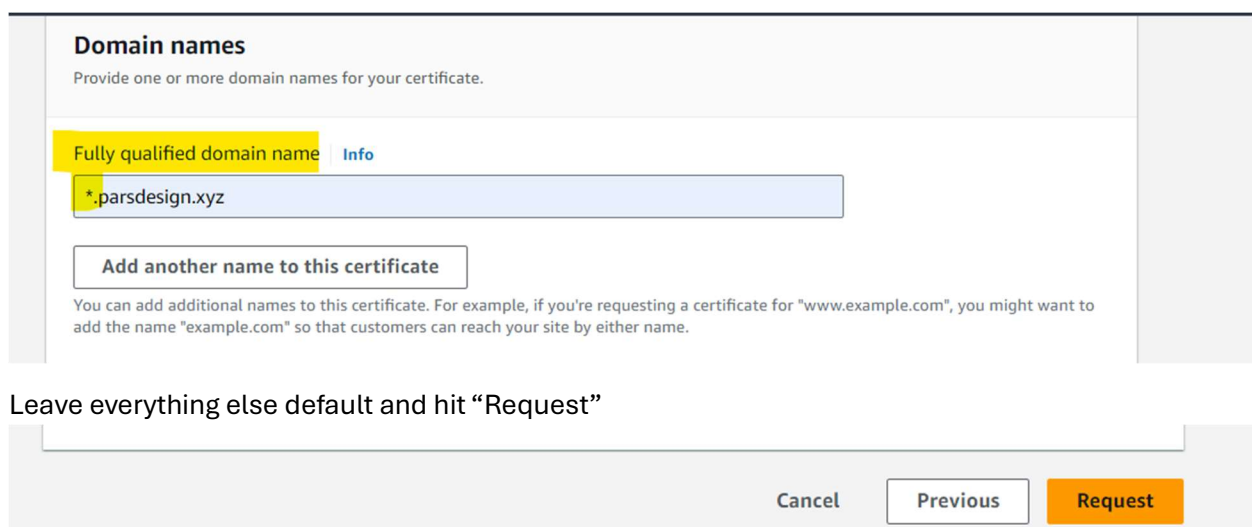
And now it is the time to create a certificate and It is important to read the highlighted section, here it says “Associate a certificate from AWS Certificate Manager. The certificate must be in the US East (N. Virginia) Region (us-east-1).”



After clicking on create certificate on the above pic, you will see the following, select the option and hit next



Then type *.domain name as below



Then we will have a pending verification certificate, so we need to go to the DNS management of our website and add a CNAME record, for the name of the cname do not copy the website name and for the value omit the last "dot" . I tried my best to show what I mean in the below pic.

Domains (1)

Create records in Route 53 [Export to CSV](#)

Domain	Status	Renewal status	Type	CNAME name	CNAME value
.parsdesign.xyz	Pending validation	-	CNAME	9c1c5b6f72a7-1f862948cd1-90-.parsdesign.xyz	2516baade4-38033cd483d5c-5dec6-3dgate-4f-*.acm-validations.aws

My DNS management look like this so I added above info here , read again previous requirements in previous step

New Records

[CNAME records](#) are a type of subdomain, or alias, that points to another domain name.

Type *

Name *

Value *

TTL

[Add More Records](#) [Save](#) [Cancel](#)

After a few min (most of the time) it will be issued, for me it took only few sec!

Domains (1)

Create records in Route 53 [Export to CSV](#)

Domain	Status	Renewal status	Type	CNAME name	CNAME value
.parsdesign.xyz	Success	-	CNAME	9c1c5b6f72a7-1f862948cd1-90-.parsdesign.xyz	2516baade4-38033cd483d5c-5dec6-3dgate-4f-*.acm-validations.aws

Now we go back again to cloud front, the place we were before creating the certificate,

And we select the certificate we just created, if you don't see that create on refresh button beside it

Custom SSL certificate - optional

Associate a certificate from AWS Certificate Manager. The certificate must be in the US East (N. Virginia) Region (us-east-1).

*.parsdesign.xyz (e6dd0b8-1467-409c-7e9a0c72e31-) [Refresh](#)

☒ *.parsdesign.xyz [Request certificate](#)

The root default again is index.html

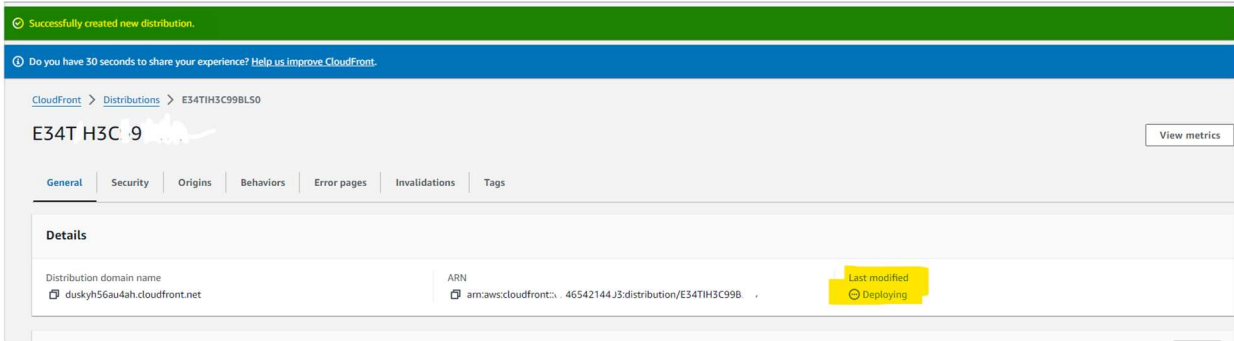
Default root object - optional

The object (file name) to return when a viewer requests the root URL (/) instead of a specific object.

Everything else on default and then click on "create distribution"

Description - optional

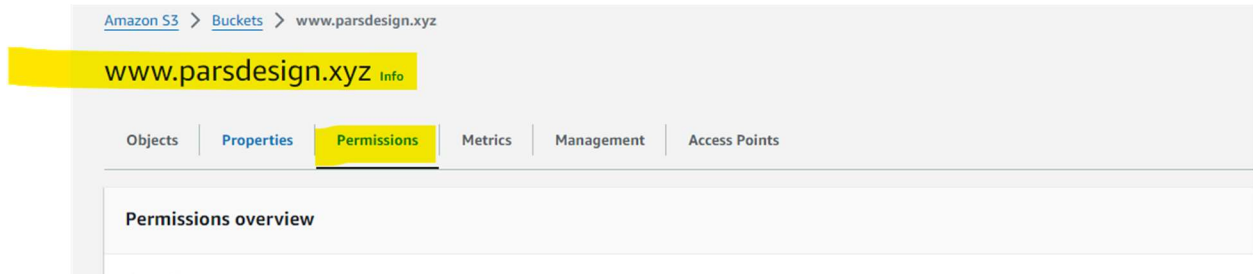
[Cancel](#) [Create distribution](#)



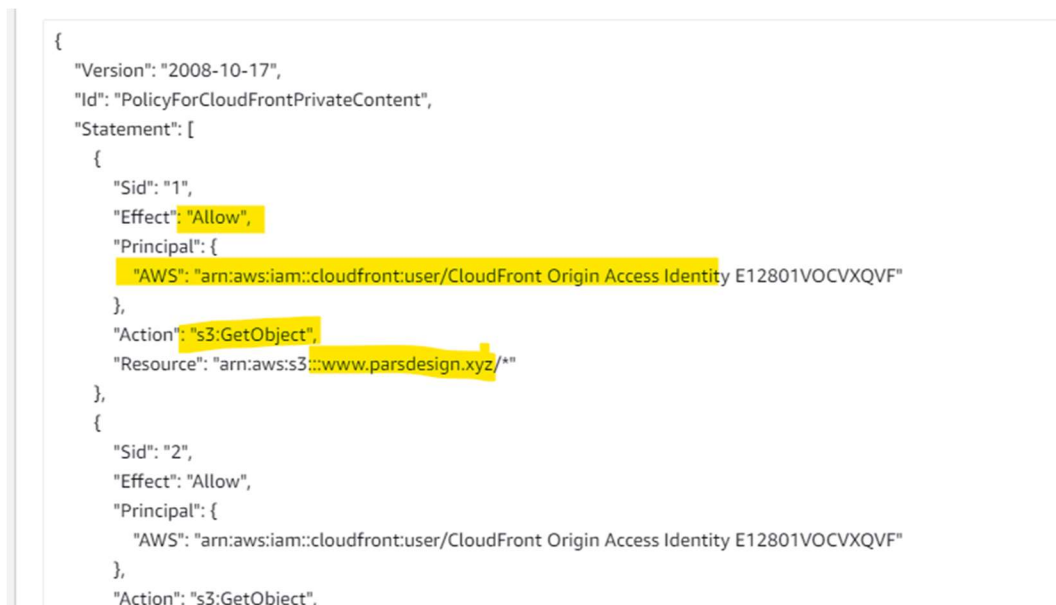
Then you need to wait for 10-15 min to deploy the app

And after that you will see that it will be deployed

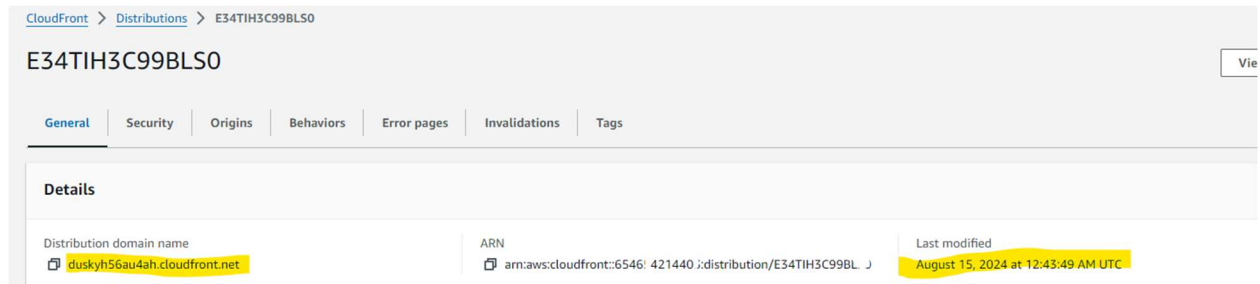
Also if you go to s3 buckets permission,



we selected update bucket policy When we enabled the CloudFront distribution, an option to 'Update bucket policy' was selected. This automatically created a bucket policy that grants access solely to the CloudFront origin access identity (OAI). This policy ensures that the S3 bucket is only accessible through CloudFront, thereby securing the content and preventing direct public access. And this policy is created automatically! Amazing, isn't it?



And after a few min waiting the website will be shown and ready to use, you can use the distribution name indicated below or your website name,



The screenshot shows the AWS CloudFront console for a distribution named E34TIH3C99BLS0. The 'Details' tab is selected, showing the following information:

Property	Value
Distribution domain name	duskyh56au4ah.cloudfront.net
ARN	arn:aws:cloudfront::6546:421440::distribution/E34TIH3C99BL...
Last modified	August 15, 2024 at 12:43:49 AM UTC

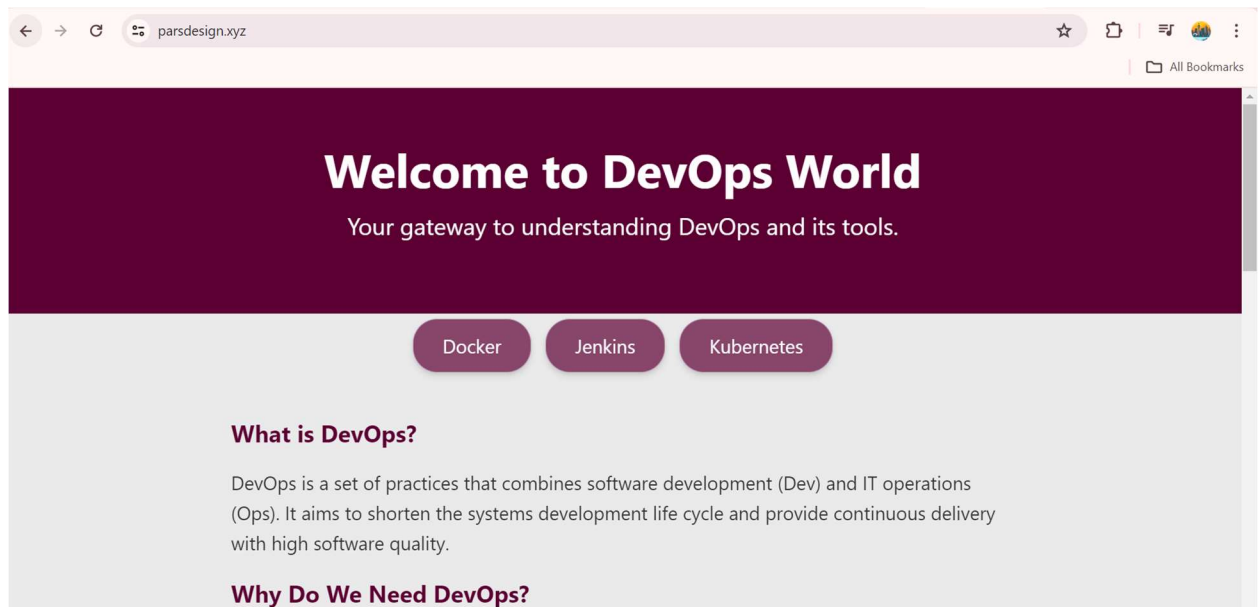
Note, if it didn't show up by the website name, you need to add a CNAME to this which the value name is above distribution name

Example Configuration:

Type	Host/Name	Value/Target
CNAME	www	duskyh56au4ah.cloudfront.net

And then Voila

The website is Up



The screenshot shows a web browser displaying the website 'parsdesign.xyz'. The website has a dark purple header with the text 'Welcome to DevOps World' and 'Your gateway to understanding DevOps and its tools.' Below the header, there are three buttons: 'Docker', 'Jenkins', and 'Kubernetes'. The main content area is light gray and contains the text 'What is DevOps?' and 'Why Do We Need DevOps?'.