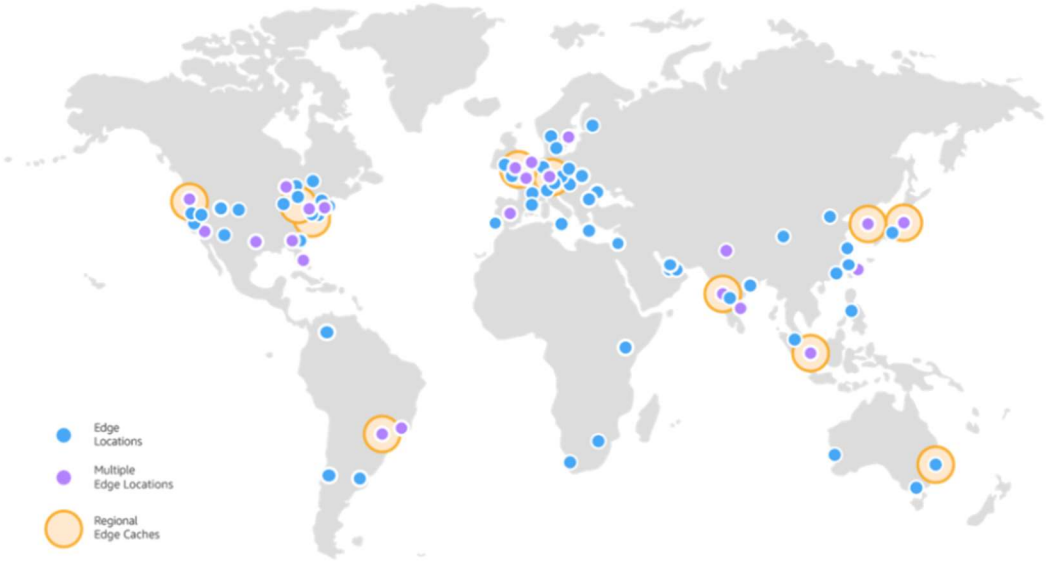
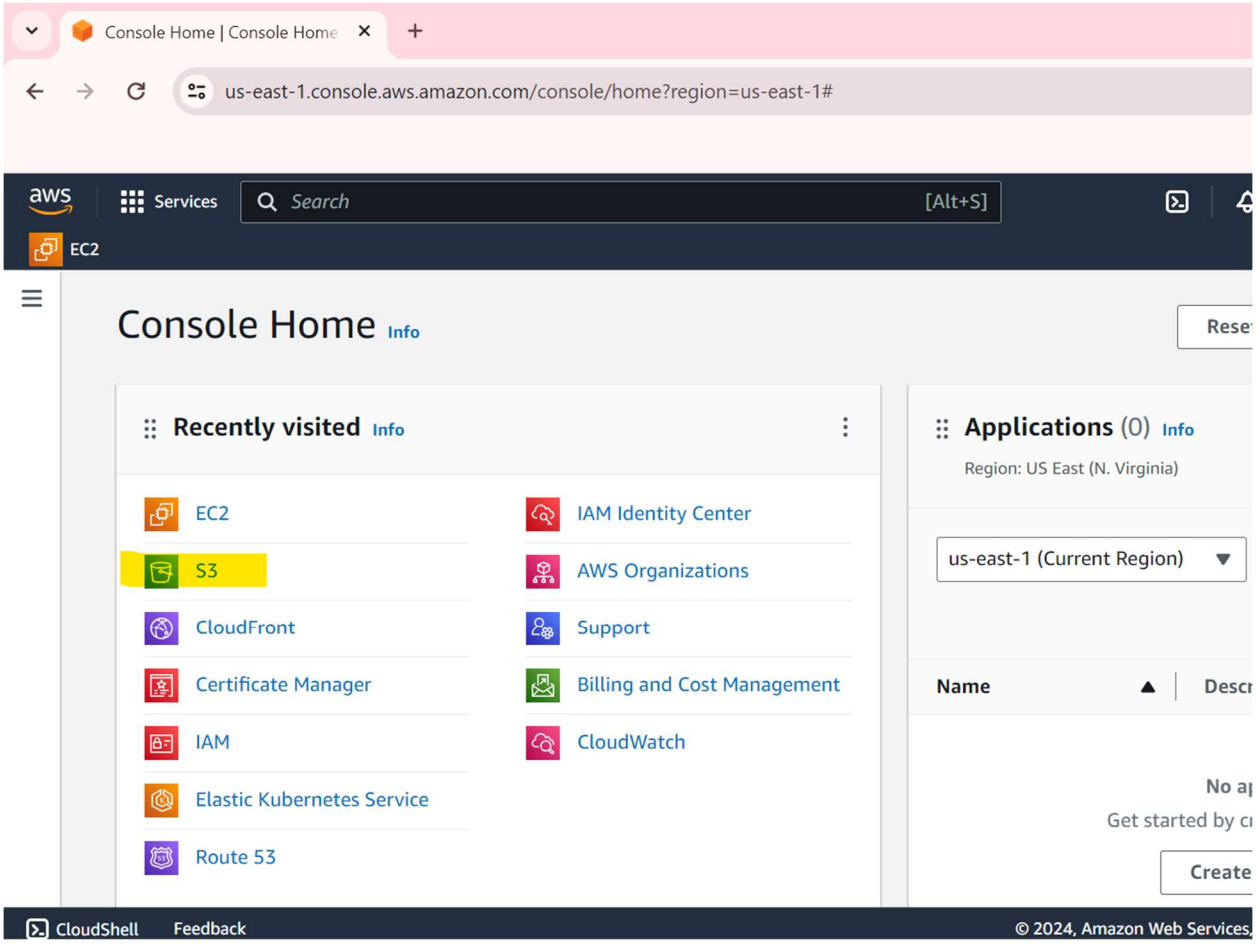
INTRODUCTION:

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.

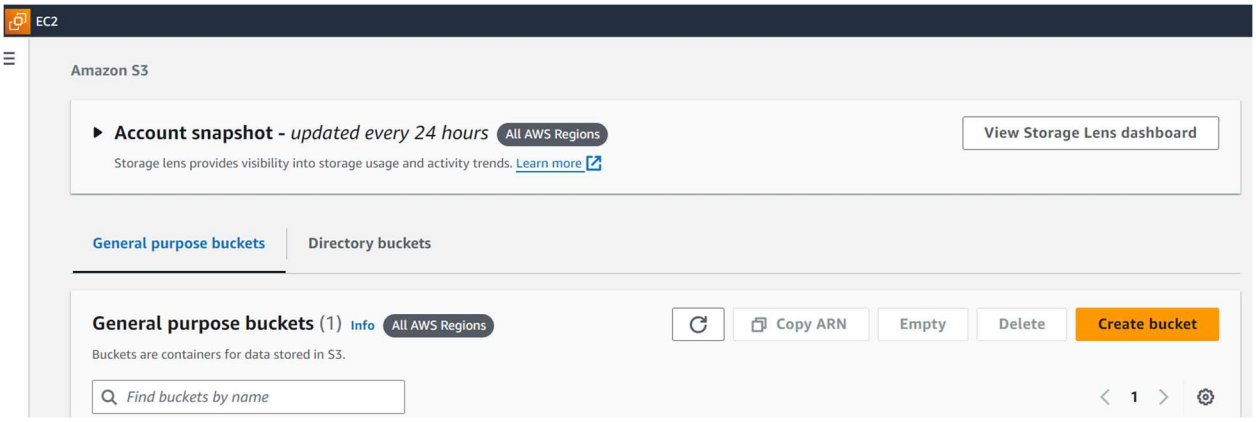
Ready to dive in and get your site live today? Let's get started!



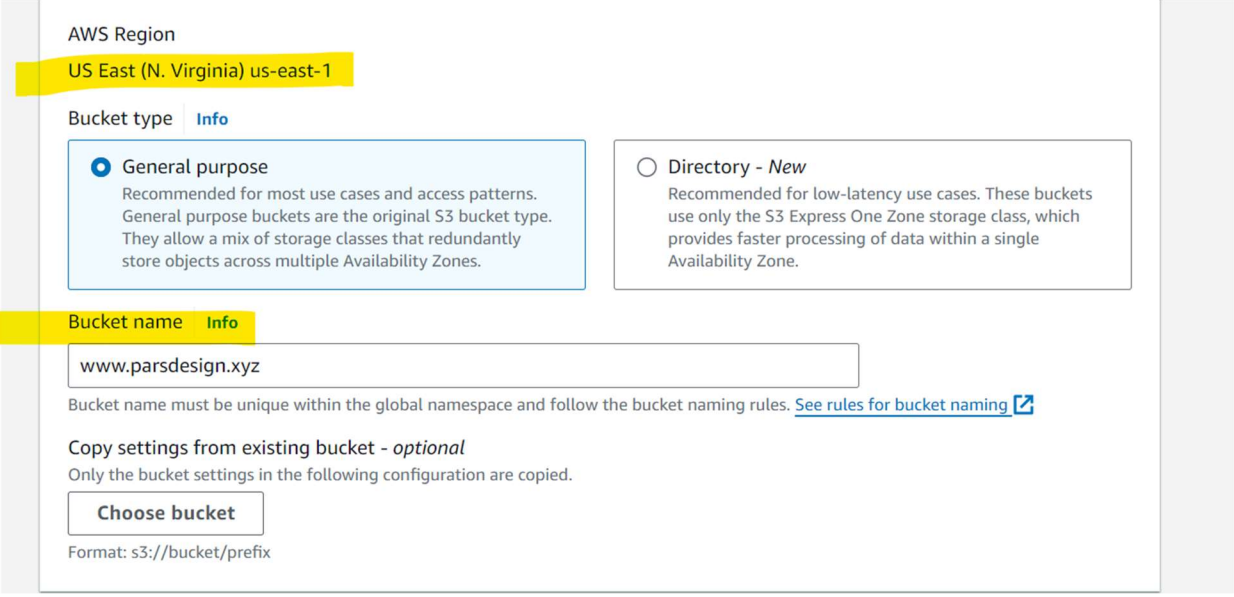
Login to the IAM user of your aws



Then click on create a bucket



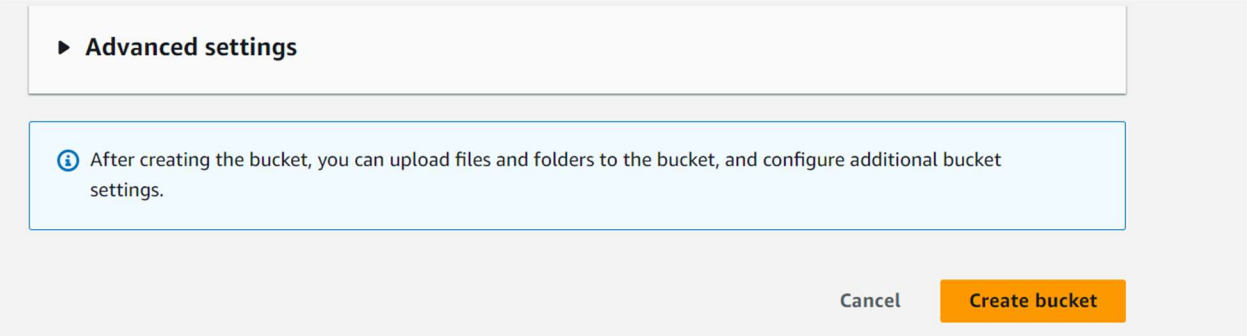
I gave it a the same as my website name, and default region which is N.Virginia



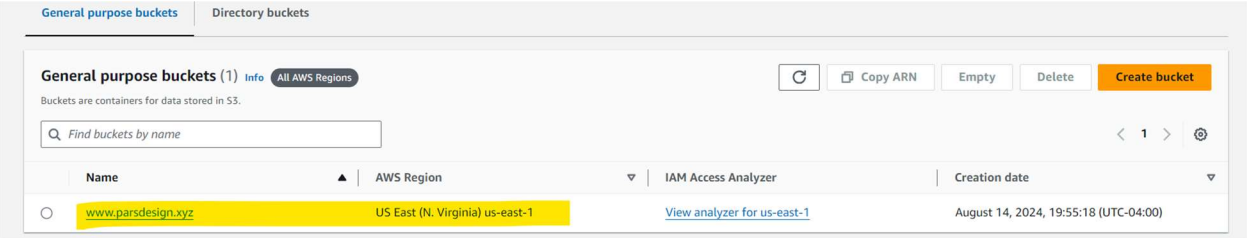
Ownership: ACLs disabled (default setting)

Block all public access(default) for security reasons

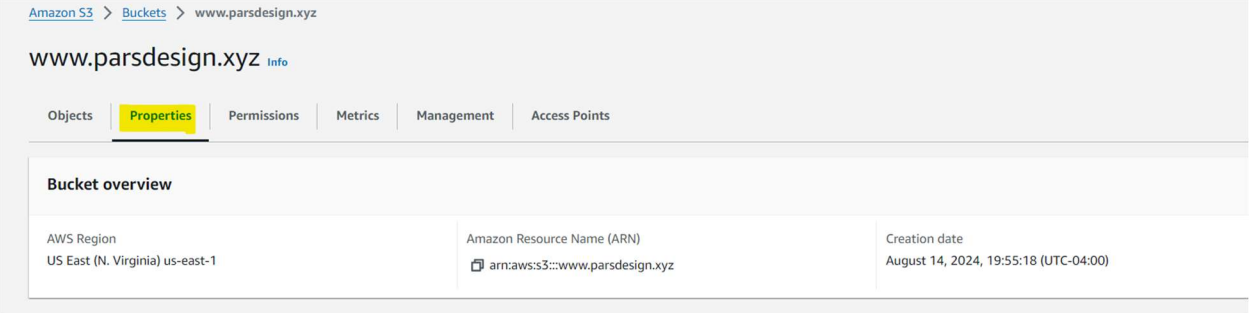
You can leave everything on its own default settings and click on “Create bucket” at the bottom



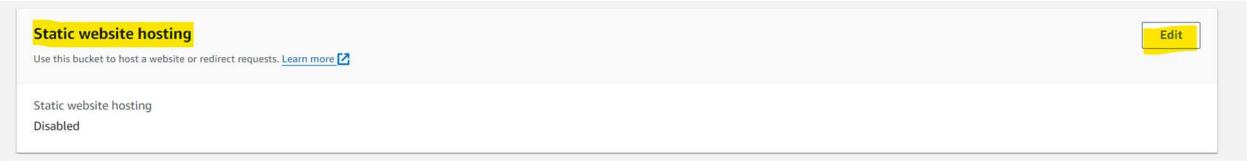
And then here we are, we have the bucket



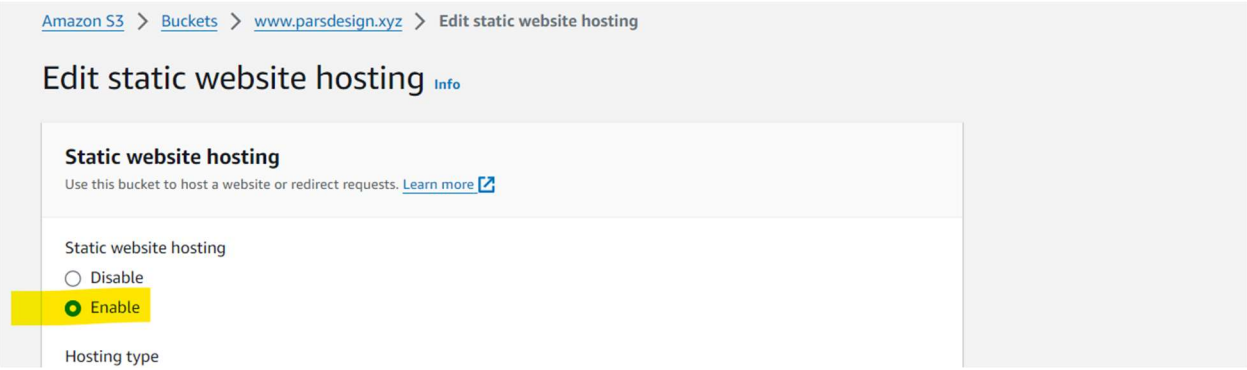
Now open the bucket you just created! And go to the properties tab



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



Then



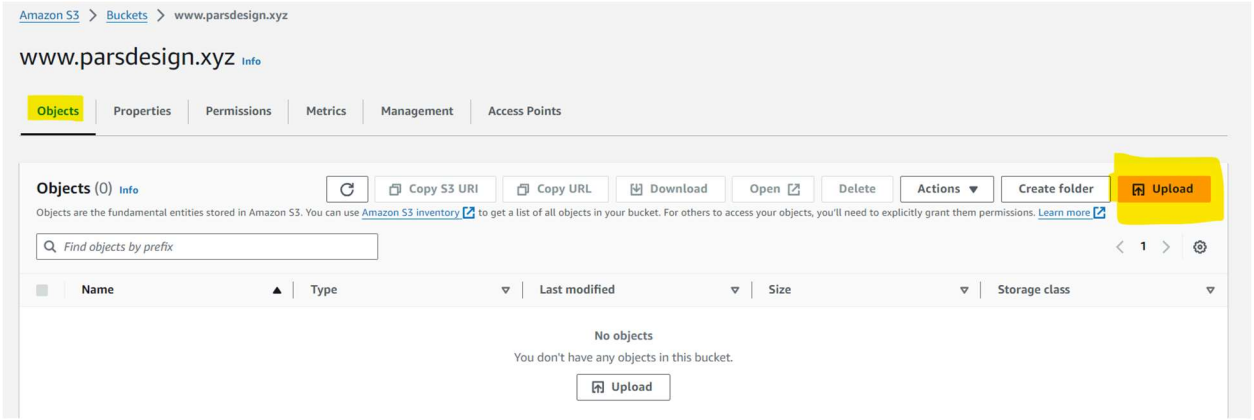
Then fill out index document field

And save changes

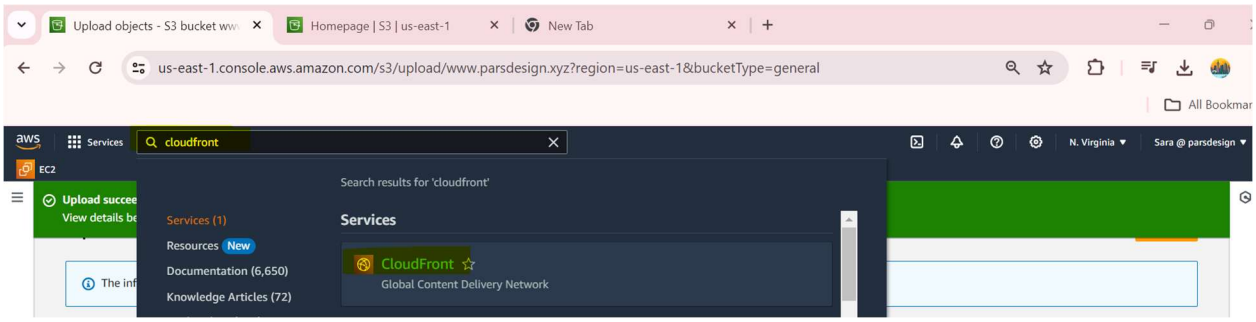
Afte all these things you can go to object section and upload the files. I quickly created a simple website for this purpose, You can use my website and download it from my git https://github.com/Sara951/Quick-Project-to-Deploy-Cloudfront-S3

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

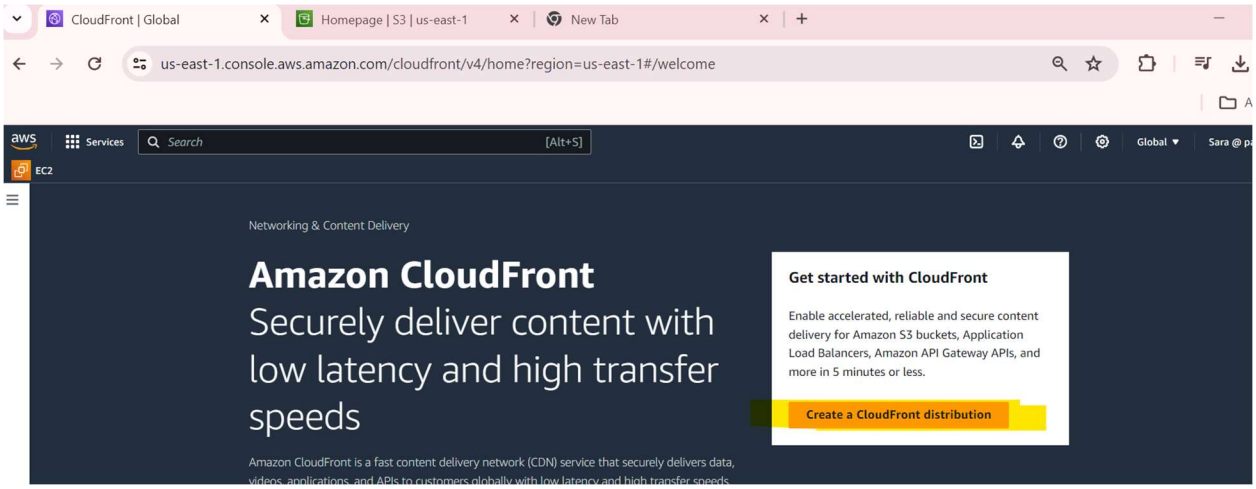
and upload it here



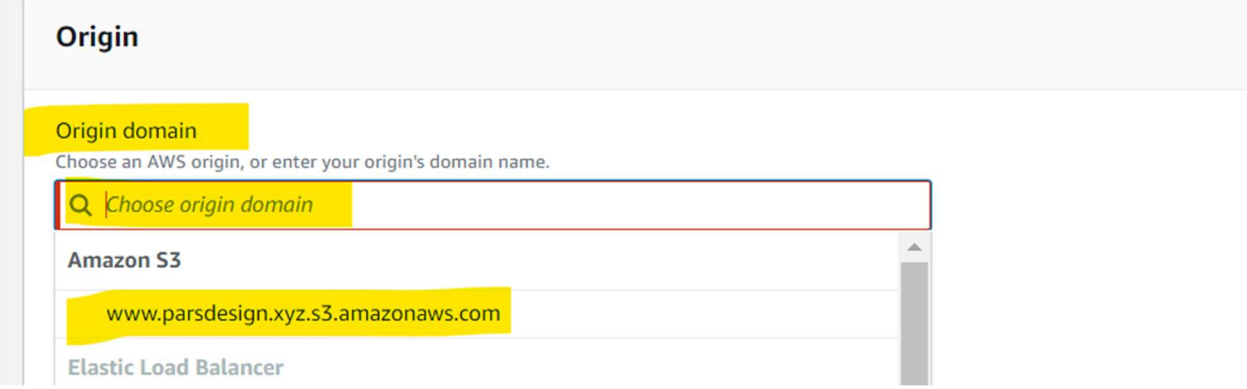
After uploading the files go to cloud front section



And create a cloud front distribution

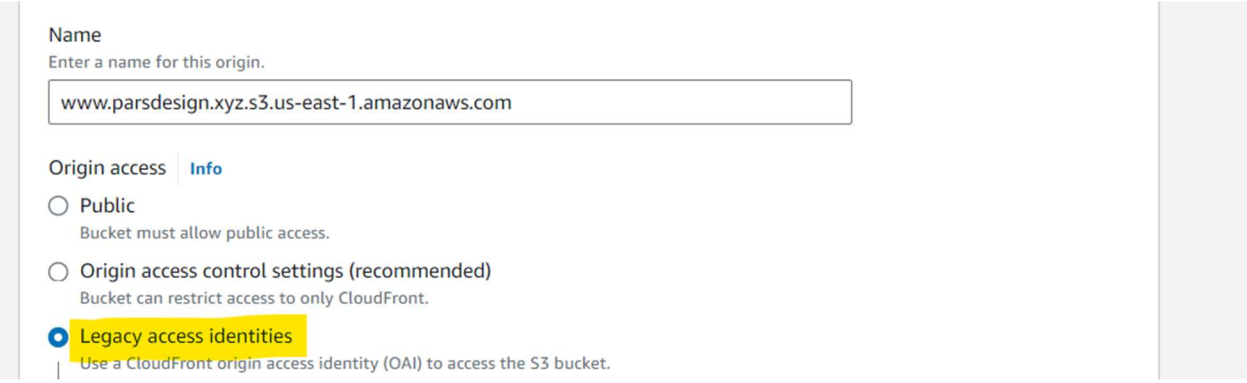


Choose origin domain which is the bucket and will be shown there



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



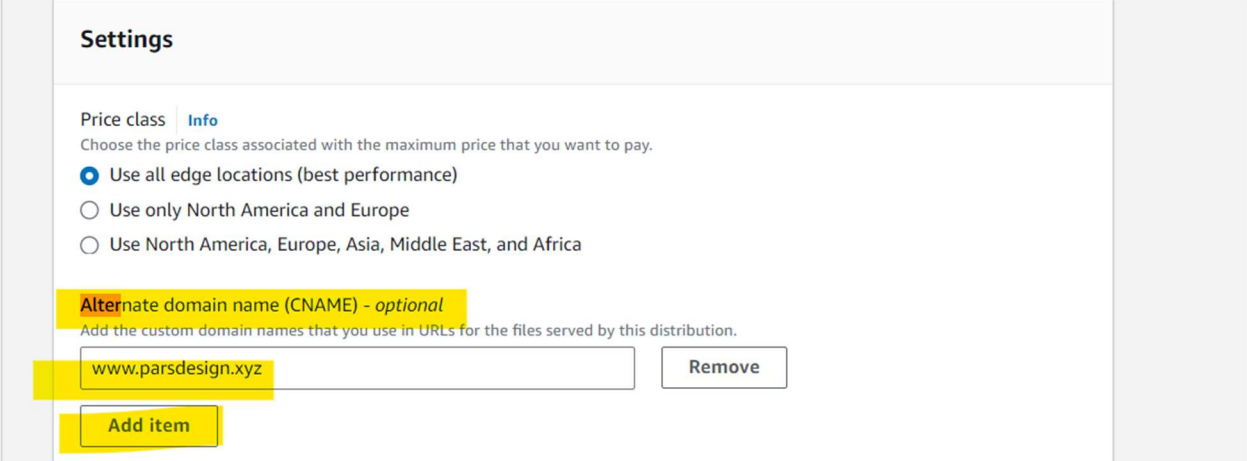
And then we click on create new OAI (origin access identity) and we select again the one we created for OAI.

And we select update the “Bucket Policy”

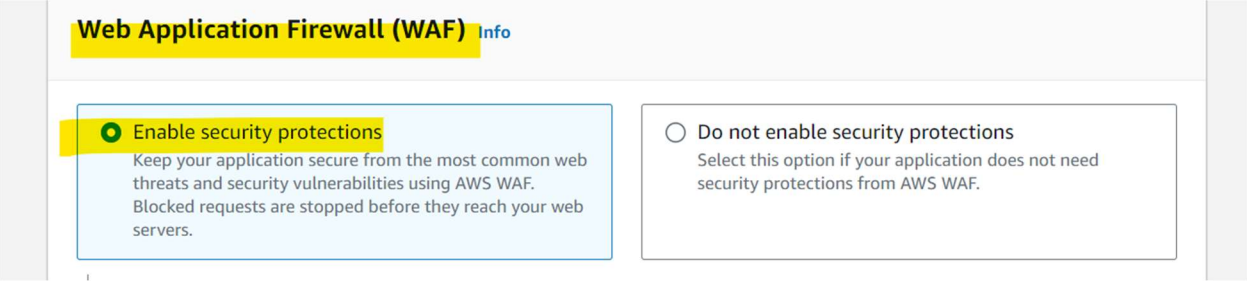
Then we change the “Viewer Protocol Policy” to:

Also for " Allowed HTTP methods"

And in the "Alternate Domain Name" again your domain name:

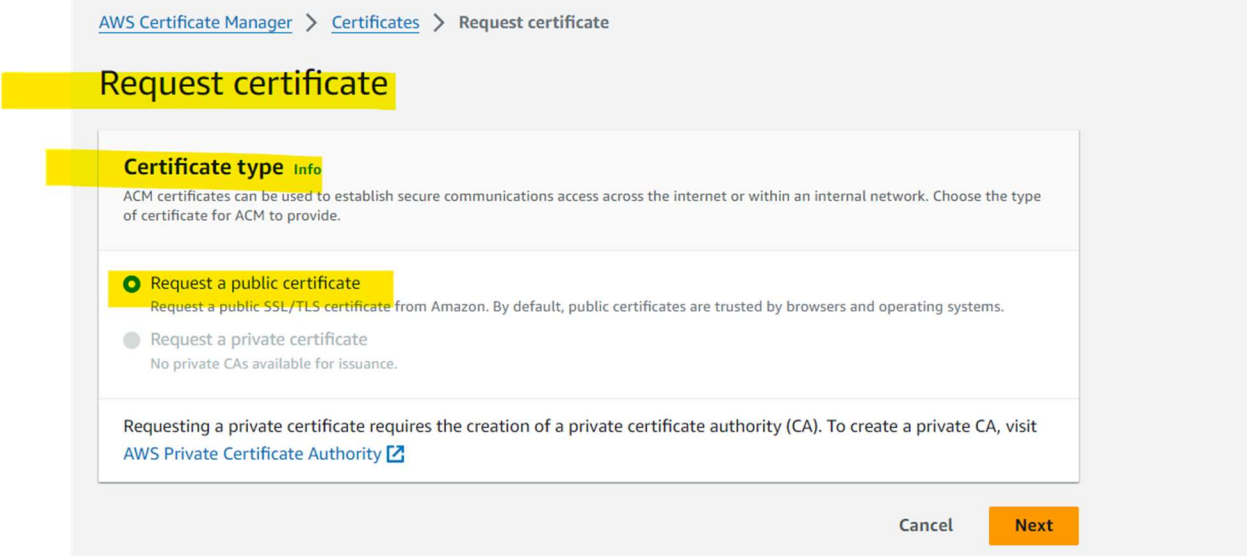


Then we need to activate WAF

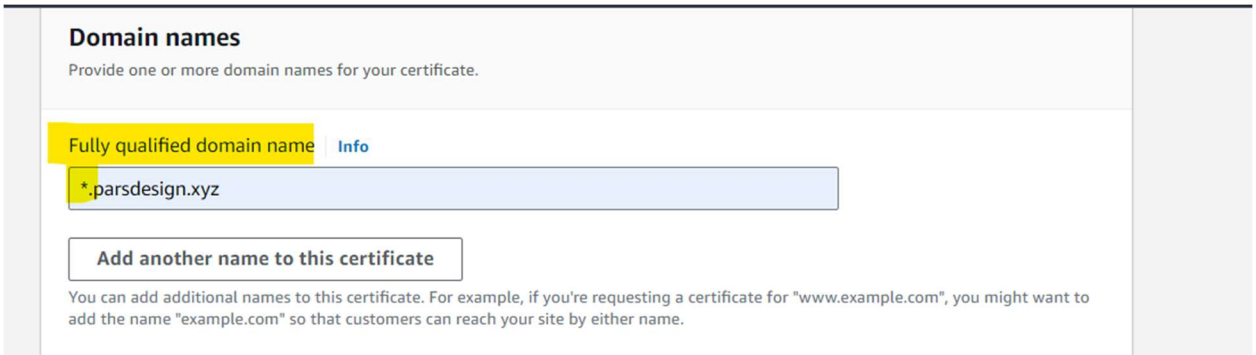


And now it is the time to create a certificate and It is important to read the highlighted section, here it says *"Associate a certificate from AWS Certificate Manager. The certificate must be in the US East (N. Virginia) Region (us-east-1)."*

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



Leave everything else default and hit “Request”

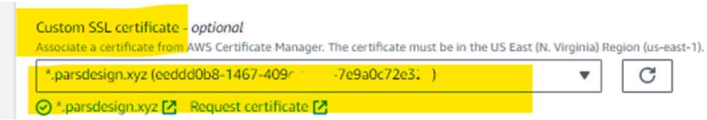
Then we will have a pending verification certificate, so we need to go to the DNS management of or website and add a CNAME record, for the name of the cname do not copy the website name and for the value omit the last “dot” . I tried my best to show what I mean in the below pic.

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

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

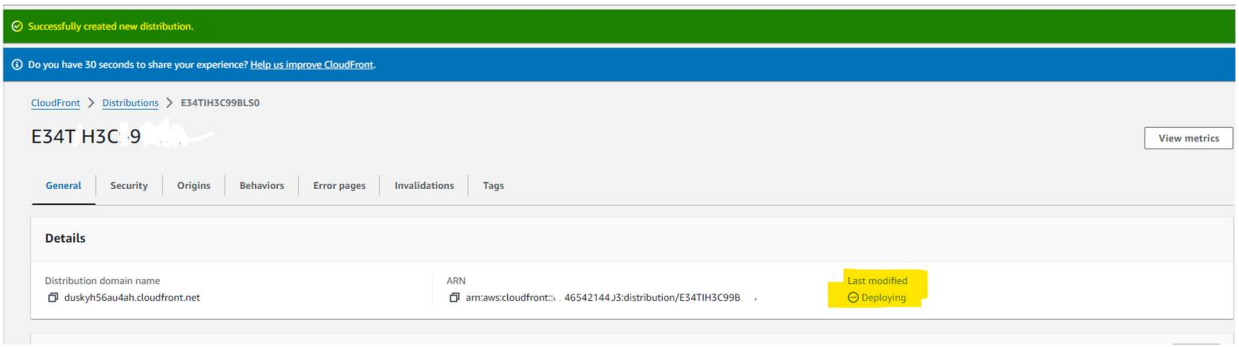
Now we go back again to cloud front, the place we were before creating the certificate,

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



The root default again is index.html

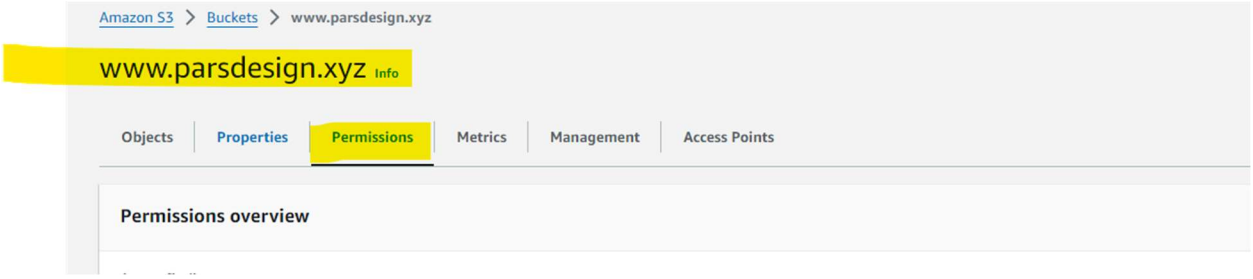
Everything else on default and then click on "create distribution"



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

And after that you will see that it will be deployed

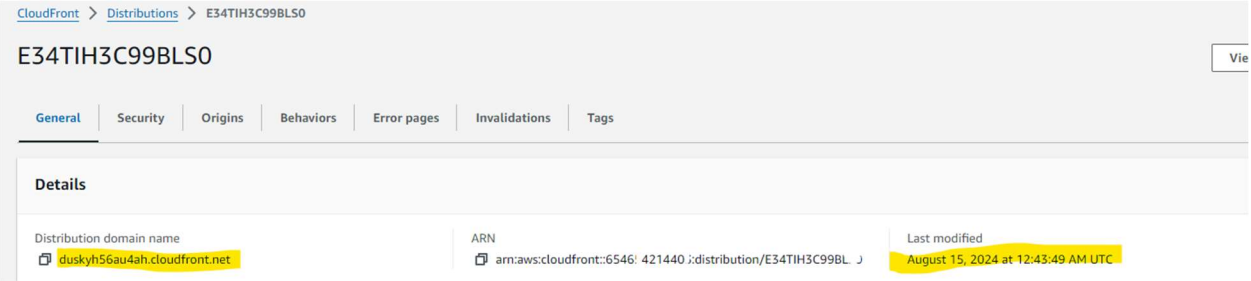
Also if you go to s3 buckets permission,



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?

```
{
  "Version": "2008-10-17",
  "Id": "PolicyForCloudFrontPrivateContent",
  "Statement": [
    {
      "Sid": "1",
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::cloudfront:user/CloudFront Origin Access Identity E12801VOCVXQVF"
      },
      "Action": "s3:GetObject",
      "Resource": "arn:aws:s3:::www.parsdesign.xyz/*"
    },
    {
      "Sid": "2",
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::cloudfront:user/CloudFront Origin Access Identity E12801VOCVXQVF"
      },
      "Action": "s3:GetObject",
```

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,



Note, if it didn't shown 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	duskyh56au-. ..h.cloudfront.net

And then Voila

The website is Up