**Project 1: Hosting a Static Website using Amazon S3, CloudFront, ACM, Route 53**

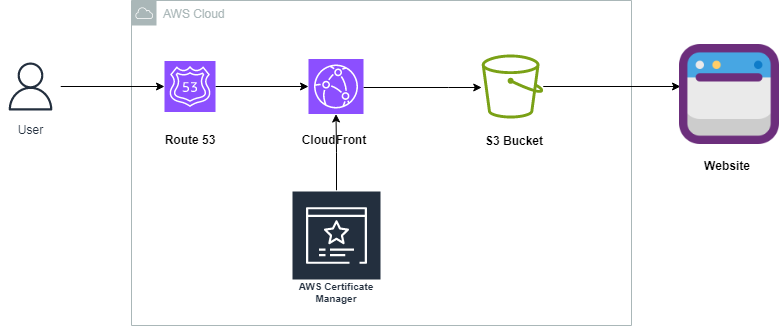
The key concept of this project is

1. **Object Storage**: Learn how cloud storage works and how files are managed in a cloud environment.
2. **Scalability**: Understand how your static website can handle varying levels of traffic without manual intervention.
3. **Cost Efficiency**: Discover the pay-as-you-go model of cloud services, making it affordable for personal and small-scale projects.
4. **Global Reach**: Gain insights into content delivery networks (CDNs) and how your website can be accessed quickly from anywhere in the world. Route 53 connects user requests to infrastructure in AWS and outside of AWS & managing DNS records for domain names.

**Steps to host a static website with custom domain**

1. Create a custom domain name using Amazon Route 53.
2. Use an Amazon S3 bucket to host a sample website.
3. Enable static website hosting for the S3 bucket.
4. Create a certificate for the website and add the record to Amazon Route 53.
5. Create a CloudFront distribution and map it to S3 bucket.

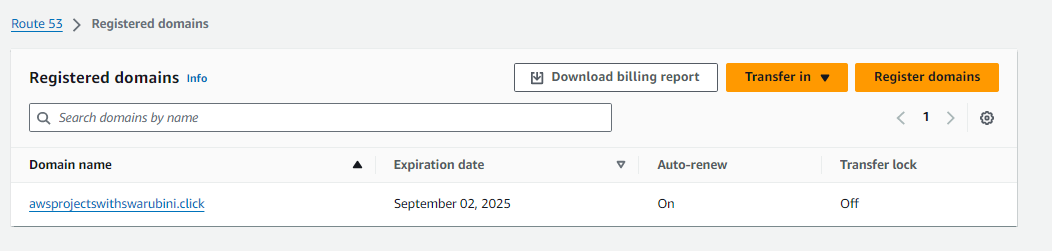
**Architecture: Hosting a Static Website using Amazon S3**



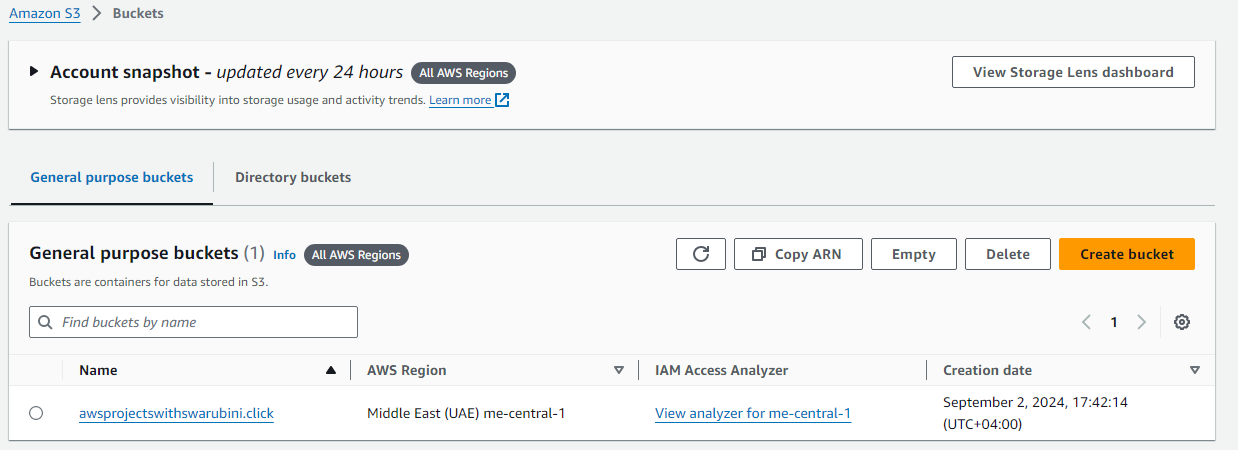
**AWS Services used: Amazon S3, Amazon Route 53, Amazon CloudFront, AWS Certificate Manager**

**Process:**

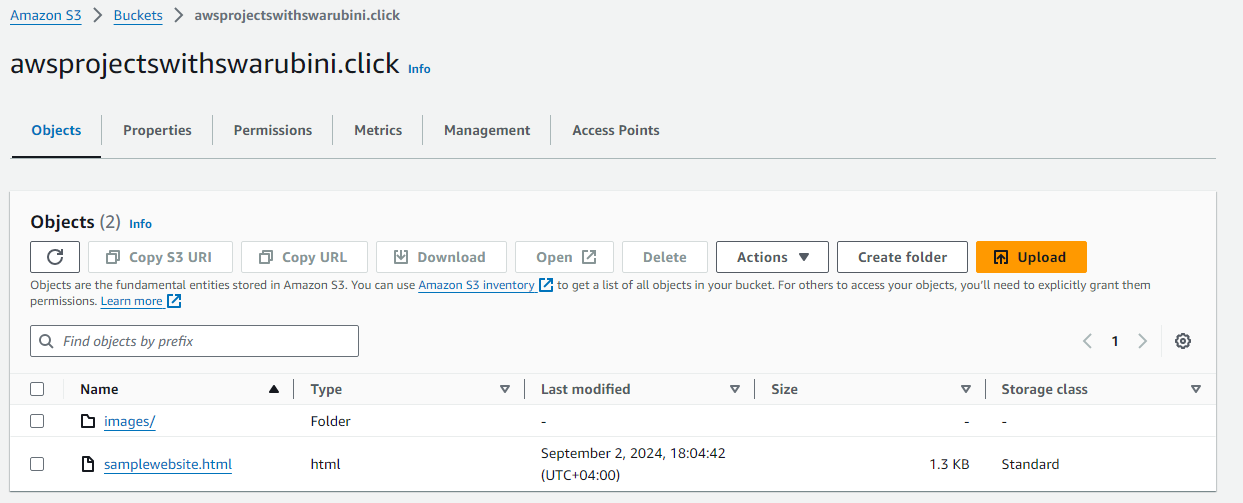
1. Created a custom domain as **awsprojectswithswarubini.click** using Amazon Route 53.



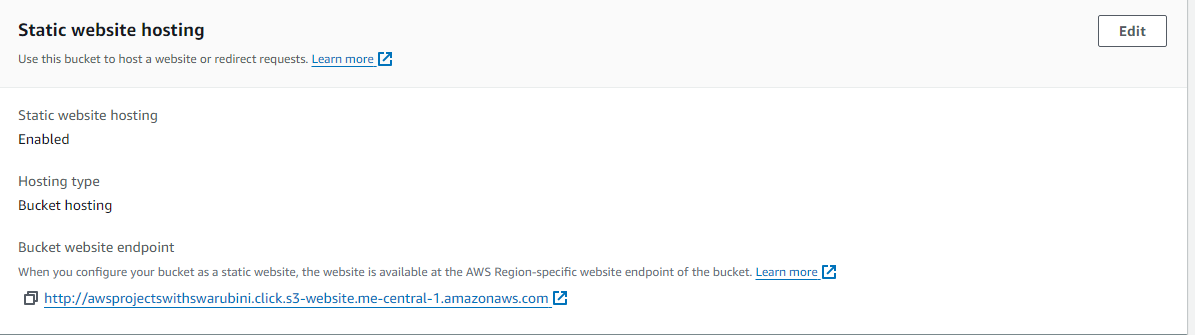
1. Created a S3 bucket as **awsprojectswithswarubini.click** and provided public access to it.



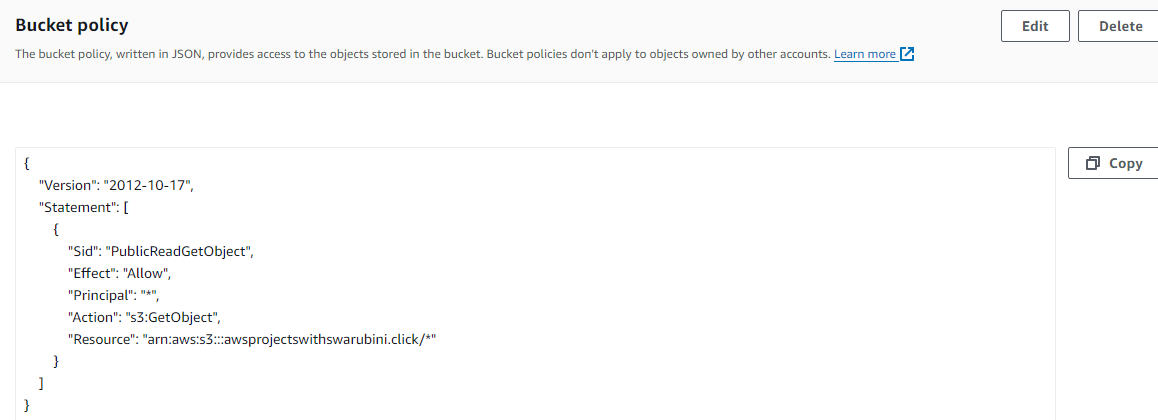
1. Uploaded a file called samplewebsite.html.



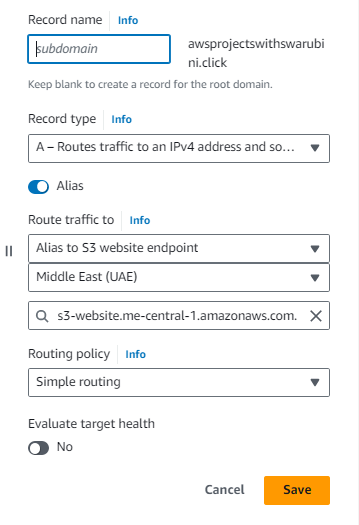
1. Enabled a static website hosting option for the S3 bucket.



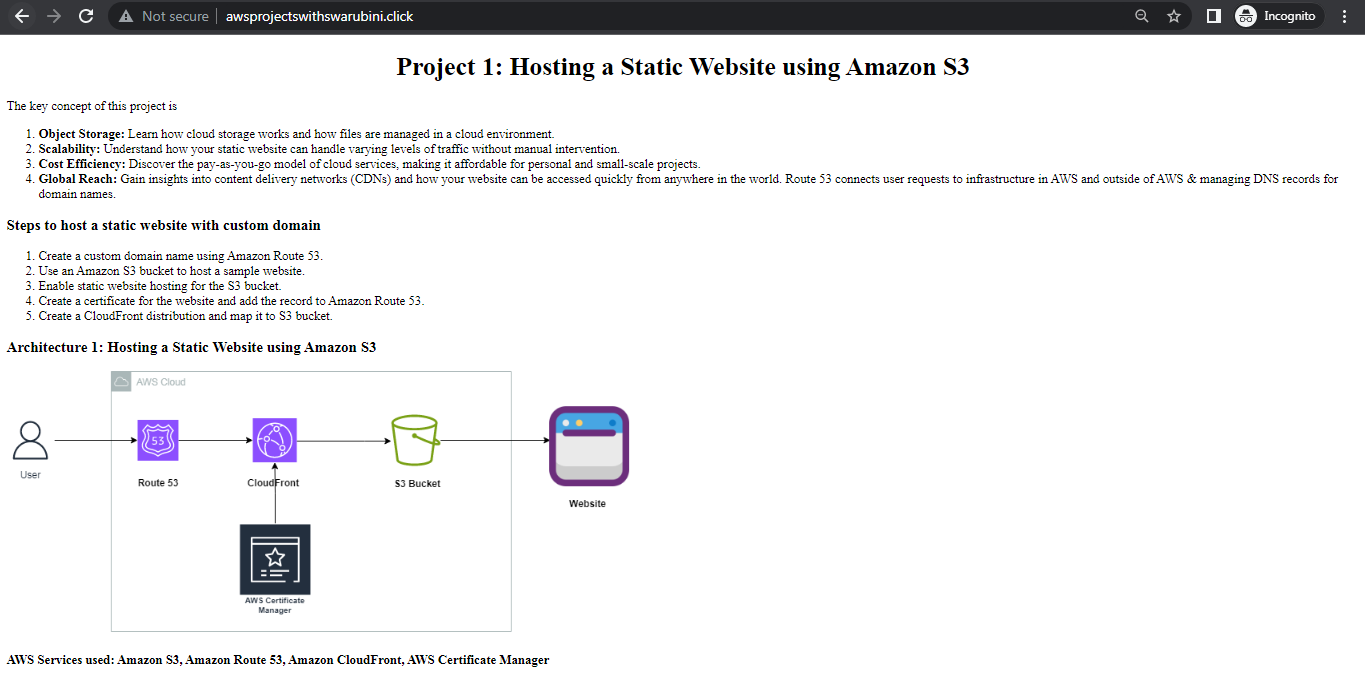
1. Attached a bucket policy to s3.GetObject for retrieves an Object from Amazon S3.



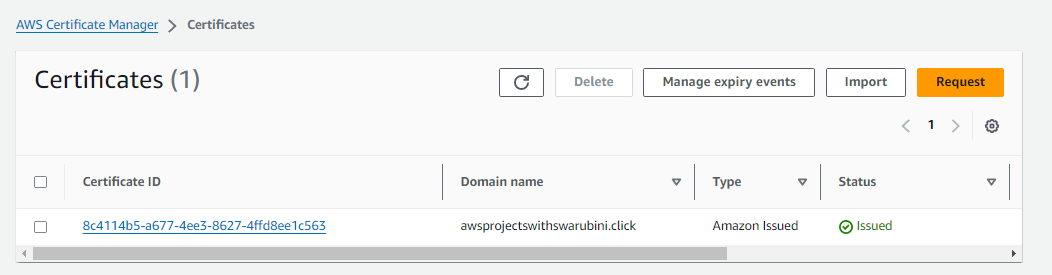
1. Now the Website URL is provided by AWS but we want to use [**http://awsprojectswithswarubini.click/**](http://awsprojectswithswarubini.click/)
2. Therefore, I created A type record and route traffic to s3 endpoint in Hosted Zones under Route 53.



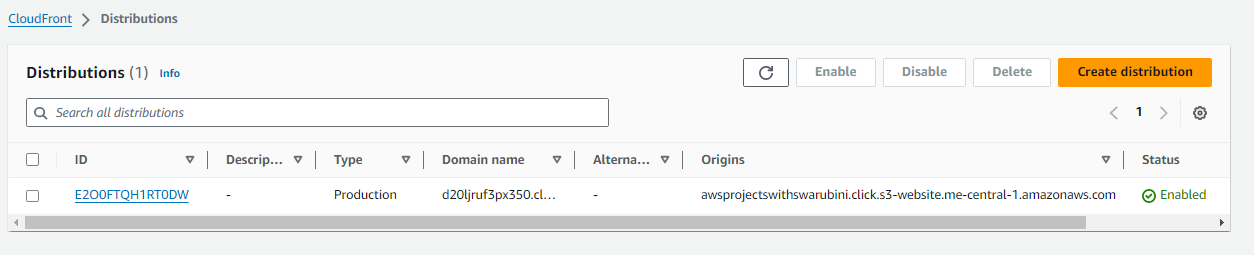
1. Now I am able to access my sample website using <http://awsprojectswithswarubini.click/>.



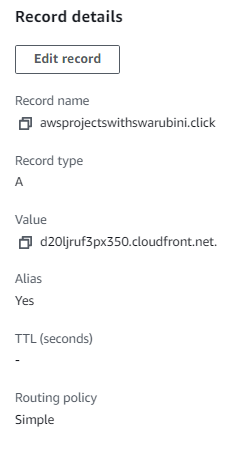
1. Currently the website is not secured and accessed via http. So we have created and requested for TLS certificate using AWS Certificate Manager. From there, we have created a CNAME record in Route 53.



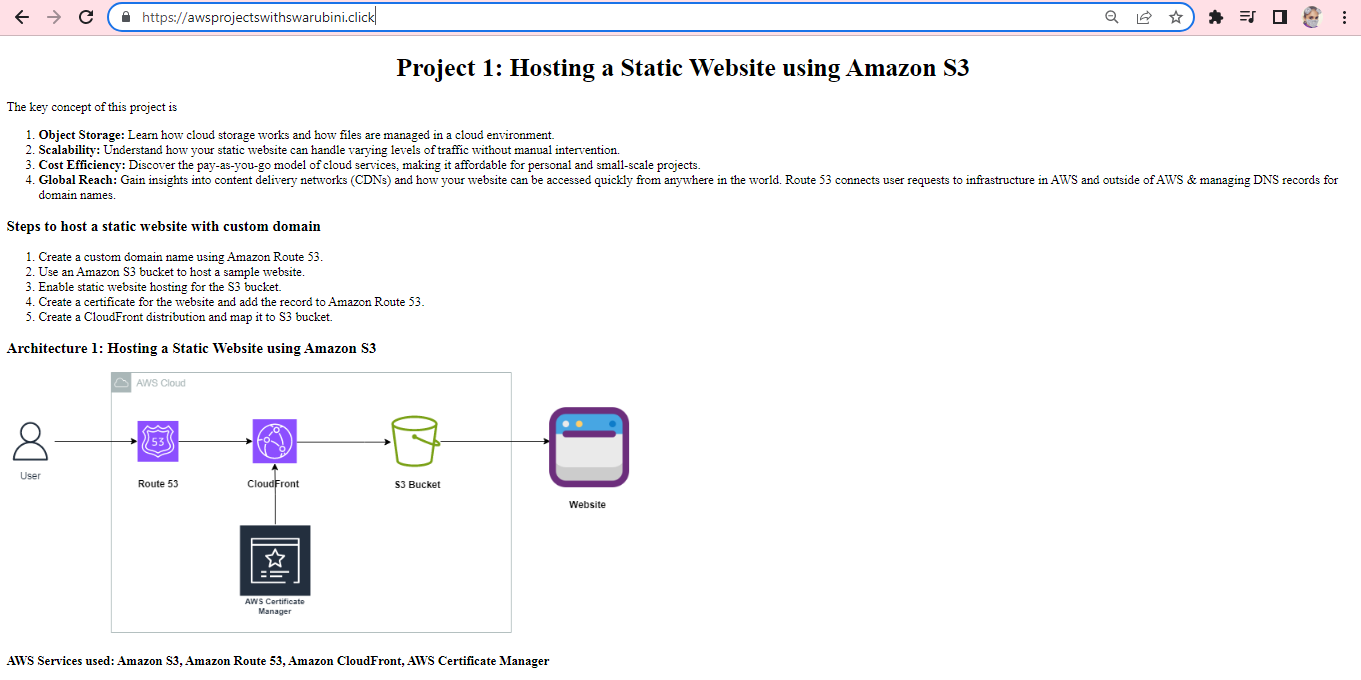
1. Created a CloudFront distribution and added an Origin with Bucket Website Endpoint. Then redirected the traffic from HTTP to HTTPS.



1. Deleted the previous A record type which is pointed to S3 bucket. Now we have created a new A type record which then pointing out to CloudFront.



1. Now we were able to run the website in secure and fast way.



**Conclusion**

Amazon S3, Amazon CloudFront and Amazon Route53 offers a powerful combination of services for Hosting, Caching, and DNS management, making hosting your website on AWS reliable and cost-effective.