Week 9

HTTPS Setup using AWS Cloud Services

Name: Abubakari-Sadique Hamidu

Email: abubakarisadiquehamidu@gmail.com

Introduction

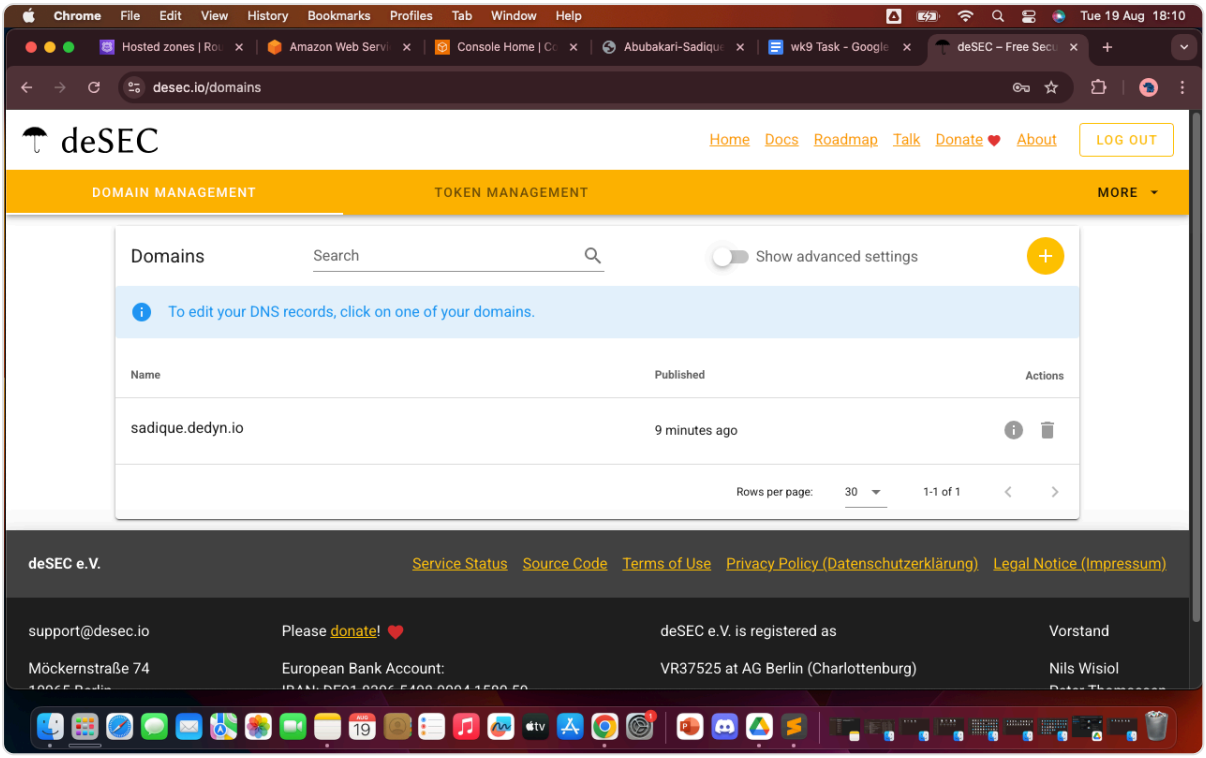
The goal of this project is to set up a secure static website hosted on AWS with HTTPS enabled. Several AWS services were used, each playing a critical role:

- **Amazon S3**: Used to host the static website. Without it, we would not have a place to store and serve the website files.
- **AWS Route 53**: Provides DNS management and connects the domain name to AWS services. Without it, users cannot easily access the site using a custom domain.
- **AWS Certificate Manager (ACM)**: Issues and manages SSL/TLS certificates for encryption. Without ACM, the site would not be secure and would show as "Not Secure" in browsers.
- **Amazon CloudFront**: A Content Delivery Network (CDN) that speeds up content delivery and adds HTTPS support. Without it, the website would load slower, especially for global users, and would not have SSL termination.

> **Note:** Since a free Freenom domain was not available, I used **sadique.dedyn.io** from DeSEC for this project.

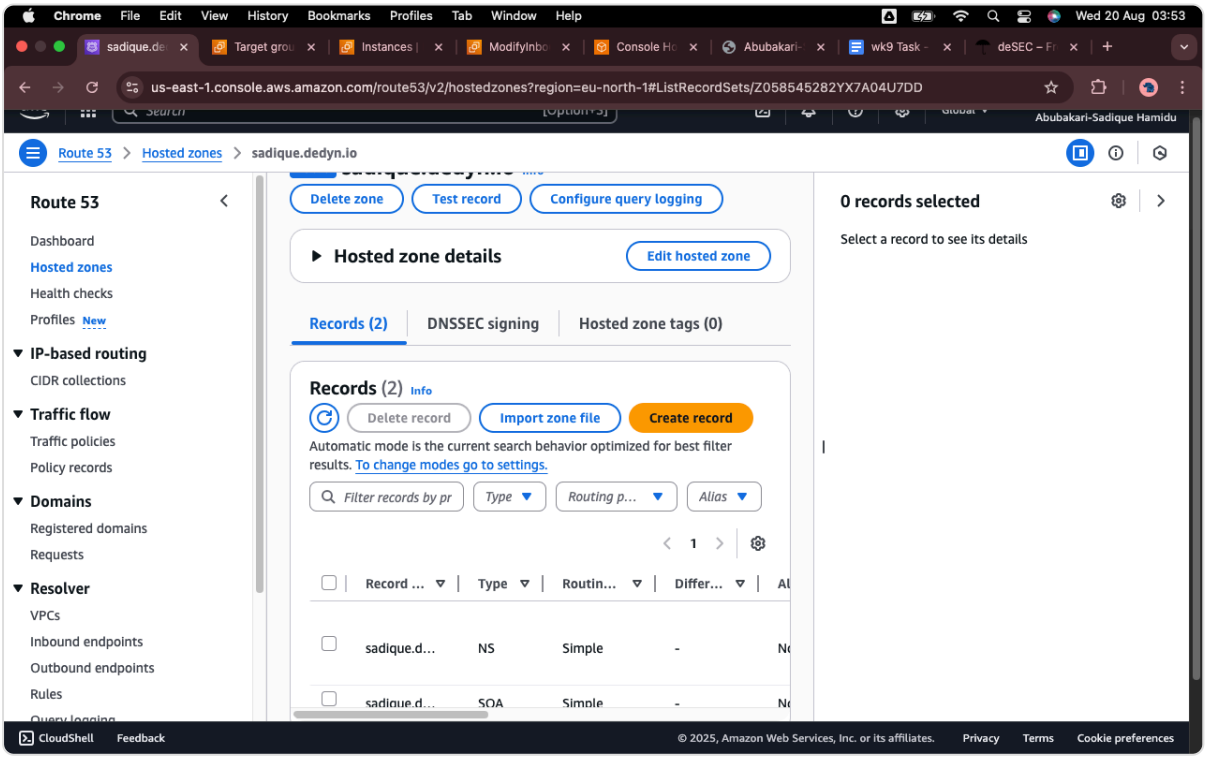
Step 1: Domain Registration

I registered the domain **sadique.dedyn.io** using DeSEC. This domain was later connected with AWS Route 53 for DNS management.



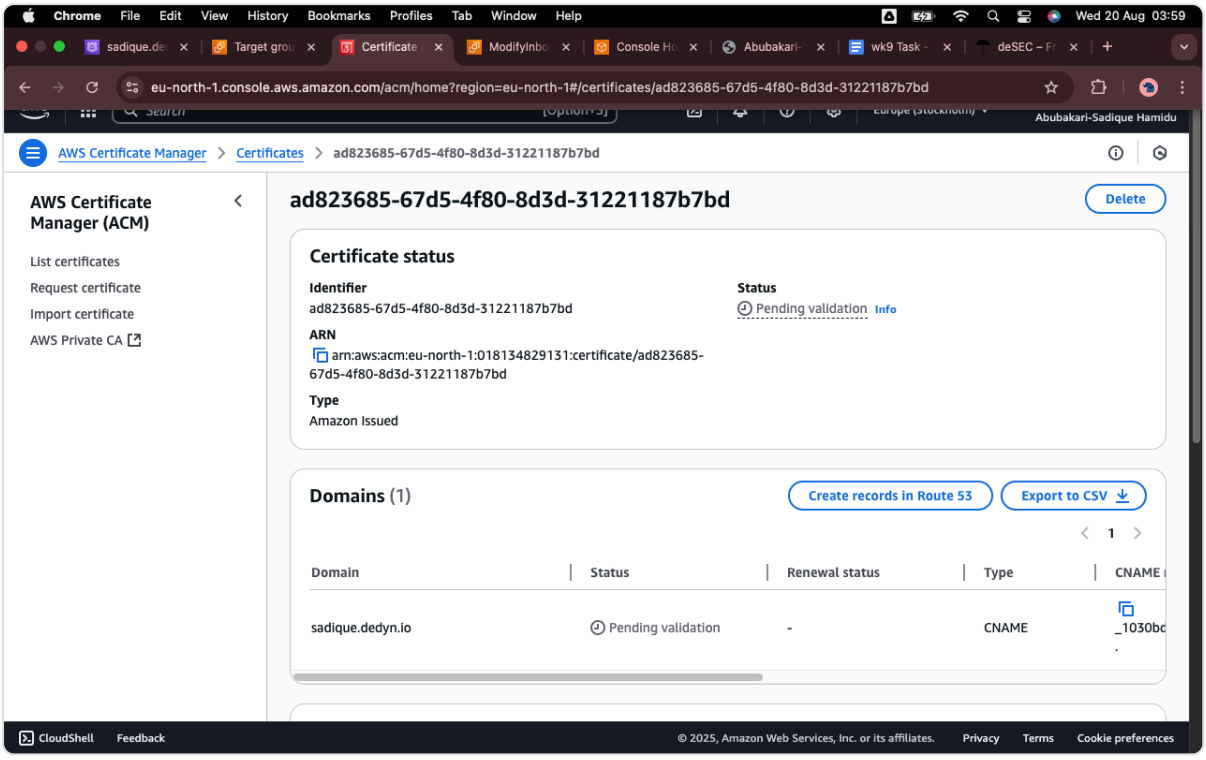
Step 2: Hosted Zone in Route 53

A hosted zone was created in Route 53 to manage DNS records for **sadique.dedyn.io**. The nameservers provided by AWS were updated at the domain registrar.



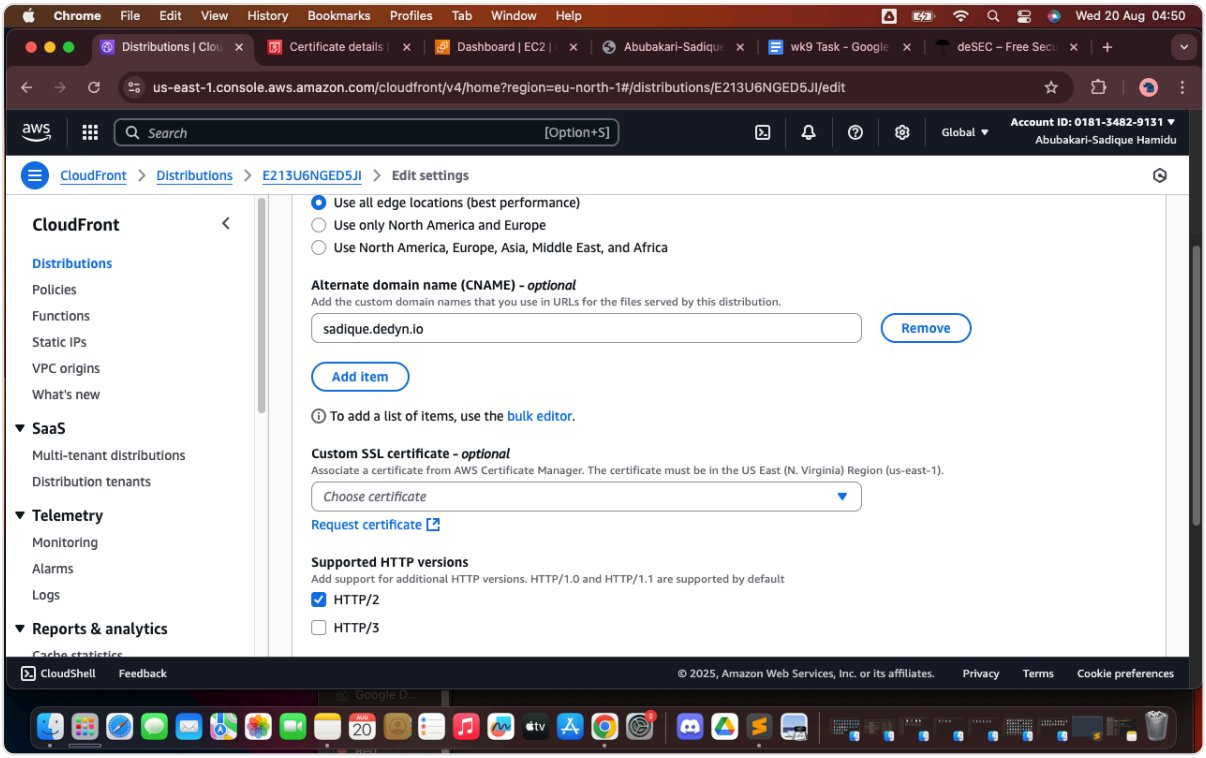
Step 3: SSL Certificate Request

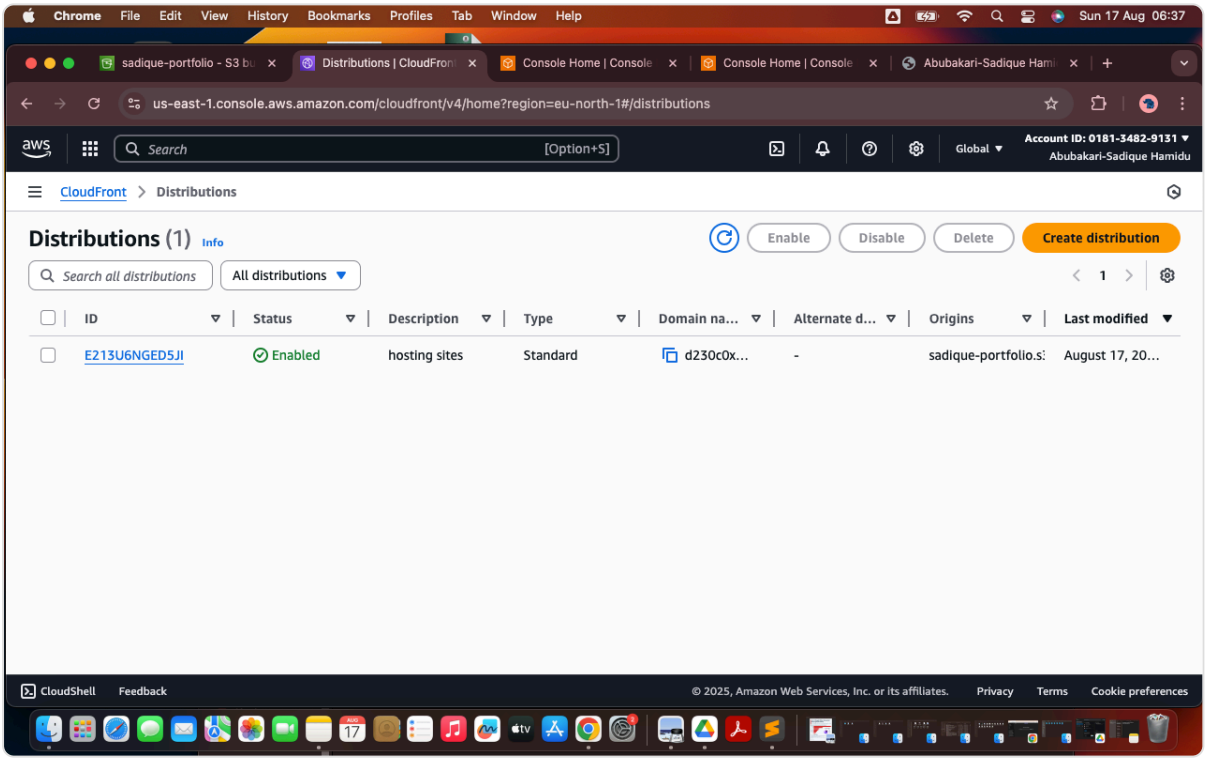
An SSL certificate was requested using AWS Certificate Manager (ACM) with DNS validation. Once validated, this certificate ensures all traffic to the website is secure.



Step 4: CloudFront Distribution

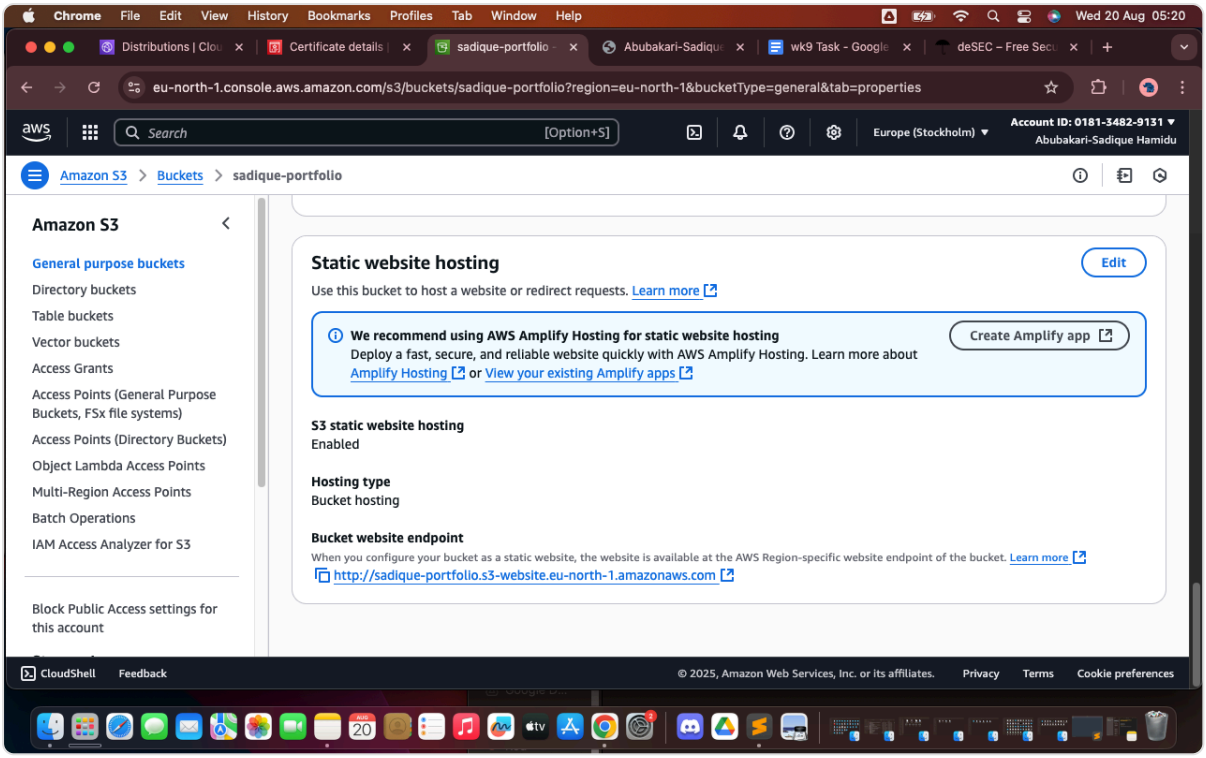
A CloudFront distribution was created to serve the website from the S3 bucket. The SSL certificate from ACM was attached to enable HTTPS on the custom domain.





Step 5: S3 Bucket

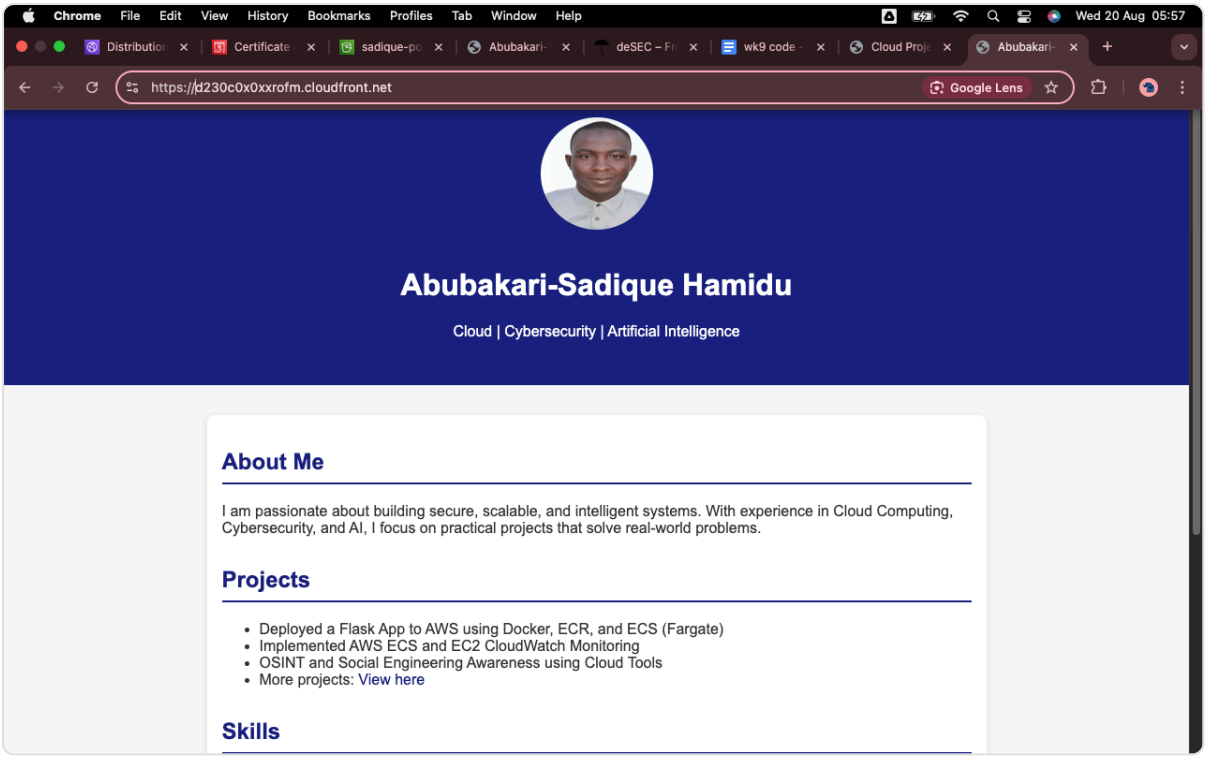
The static website files were uploaded to an S3 bucket. Static website hosting was enabled in the bucket properties, allowing CloudFront to fetch content from it.



Step 6: Live Website

Finally, the website was tested using the CloudFront-provided domain. The site loaded successfully over HTTPS, proving that the configuration was correct.

If the SSL certificate had been successfully validated in ACM, this website would load over **HTTPS** with the custom domain `sadique.dedyn.io` instead of the default CloudFront URL.



Prepared by: **Abubakari-Sadique Hamidu**

Email: abubakarisadiquehamidu@gmail.com