

# 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.

The screenshot shows the deSEC domain management interface. At the top, there are tabs for 'DOMAIN MANAGEMENT' and 'TOKEN MANAGEMENT'. Below this, a table lists domains. One entry is visible: 'sadique.dedyn.io' under 'Name', 'Published' at '9 minutes ago', and actions represented by a blue info icon and a grey trash bin icon.

## 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.

The screenshot shows the AWS Route 53 console. On the left, a sidebar menu includes 'Route 53', 'Dashboard', 'Hosted zones' (which is selected), 'Health checks', 'Profiles', 'IP-based routing', 'Traffic flow', 'Domains', 'Resolver', and 'Query logging'. The main content area shows 'Hosted zone details' for 'sadique.dedyn.io'. It displays 'Records (2)' with two entries: 'sadique.d... NS Simple -' and 'sadique.d... SOA Simple -'. There are buttons for 'Delete zone', 'Test record', 'Configure query logging', 'Edit hosted zone', 'Import zone file', and 'Create record'. A message at the bottom says 'Automatic mode is the current search behavior optimized for best filter results. To change modes go to settings.' The right side of the screen shows a message '0 records selected' and 'Select a record to see its details'.

## 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.

The screenshot shows the AWS Certificate Manager console in a web browser. The URL is [eu-north-1.console.aws.amazon.com/acm/home?region=eu-north-1#/certificates/ad823685-67d5-4f80-8d3d-31221187b7bd](https://eu-north-1.console.aws.amazon.com/acm/home?region=eu-north-1#/certificates/ad823685-67d5-4f80-8d3d-31221187b7bd). The certificate details are as follows:

Identifier	Status
ad823685-67d5-4f80-8d3d-31221187b7bd	Pending validation

**ARN:** arn:aws:acm:eu-north-1:018134829131:certificate/ad823685-67d5-4f80-8d3d-31221187b7bd

**Type:** Amazon Issued

**Domains (1):**

Domain	Status	Renewal status	Type	CNAME
sadique.dedyn.io	Pending validation	-	CNAME	_1030bc

## 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.

The screenshot shows the AWS S3 console in a web browser. The left sidebar is titled 'Amazon S3' and includes a 'General purpose buckets' section with links to 'Directory buckets', 'Table buckets', 'Vector buckets', 'Access Grants', 'Access Points (General Purpose Buckets, FSx file systems)', 'Access Points (Directory Buckets)', 'Object Lambda Access Points', 'Multi-Region Access Points', 'Batch Operations', and 'IAM Access Analyzer for S3'. Below this is a link to 'Block Public Access settings for this account'. The main content area is titled 'Static website hosting' and contains a note: 'Use this bucket to host a website or redirect requests.' with a 'Learn more' link. A callout box says 'We recommend using AWS Amplify Hosting for static website hosting' with a 'Create Amplify app' button. It also lists 'S3 static website hosting' (Enabled), 'Hosting type' (Bucket hosting), and 'Bucket website endpoint' (http://sadique-portfolio.s3-website.eu-north-1.amazonaws.com). The top of the browser window shows tabs for 'Distributions | CloudFront', 'Certificate details', 'sadique-portfolio', 'Abubakari-Sadique', 'wk9 Task - Google', 'deSEC - Free Secu...', and a new tab. The status bar at the bottom shows 'Wed 20 Aug 05:20'.

## 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.

The screenshot shows a resume page for Abubakari-Sadique Hamidu. The page has a dark blue header with a circular profile picture of the author. Below the header, the name "Abubakari-Sadique Hamidu" is displayed in bold, followed by the text "Cloud | Cybersecurity | Artificial Intelligence". The main content area is white and contains three sections: "About Me", "Projects", and "Skills". The "About Me" section includes a short bio: "I am passionate about building secure, scalable, and intelligent systems. With experience in Cloud Computing, Cybersecurity, and AI, I focus on practical projects that solve real-world problems." The "Projects" section lists three items: "Deployed a Flask App to AWS using Docker, ECR, and ECS (Fargate)", "Implemented AWS ECS and EC2 CloudWatch Monitoring", and "OSINT and Social Engineering Awareness using Cloud Tools". It also includes a link "More projects: [View here](#)". The "Skills" section is currently empty.

Prepared by: **Abubakari-Sadique Hamidu**

Email: [abubakarisadiquehamidu@gmail.com](mailto:abubakarisadiquehamidu@gmail.com)

© 2025