

# Project Documentation

## Project Title: AWS Static Website Architecture with Security & Performance Optimization

---

### Objective

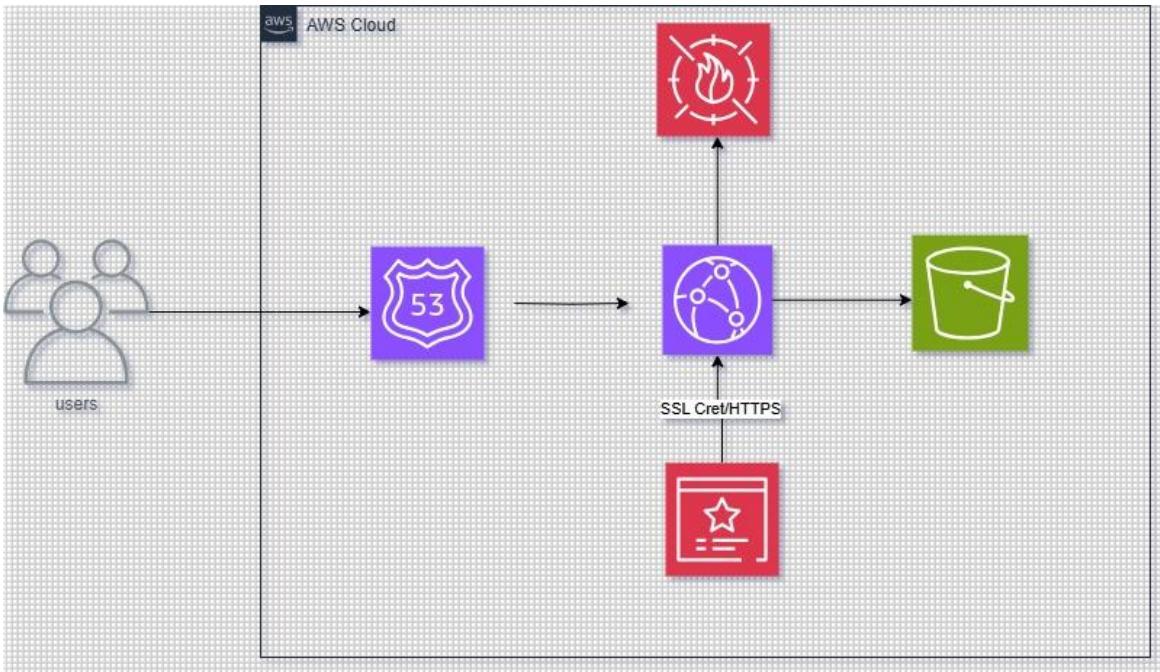
To design and deploy a secure, scalable, and high-performing static website hosted on Amazon S3, integrated with CloudFront for content delivery, AWS WAF for threat protection, and Route 53 for domain-level DNS routing. The architecture also ensures HTTPS encryption via AWS Certificate Manager (ACM), delivering a professional-grade solution for static content hosting.

### Architecture Overview

#### Services Used:

- **Amazon S3** (Static Website Hosting)
- **Amazon CloudFront** (Global Content Delivery Network)
- **AWS WAF** (Web Application Firewall)
- **Amazon Route 53** (DNS Service)
- **AWS Certificate Manager (ACM)** (SSL/TLS Management)

#### Architecture Flow



## Security Measures

- **AWS WAF** is attached to CloudFront to block common web exploits (e.g., SQL injection, XSS).
- **S3 Bucket** is configured with public access blocked. Only CloudFront is allowed using Origin Access Control (OAC).
- **HTTPS enabled** using a validated SSL certificate from **ACM** to ensure encrypted data transmission.

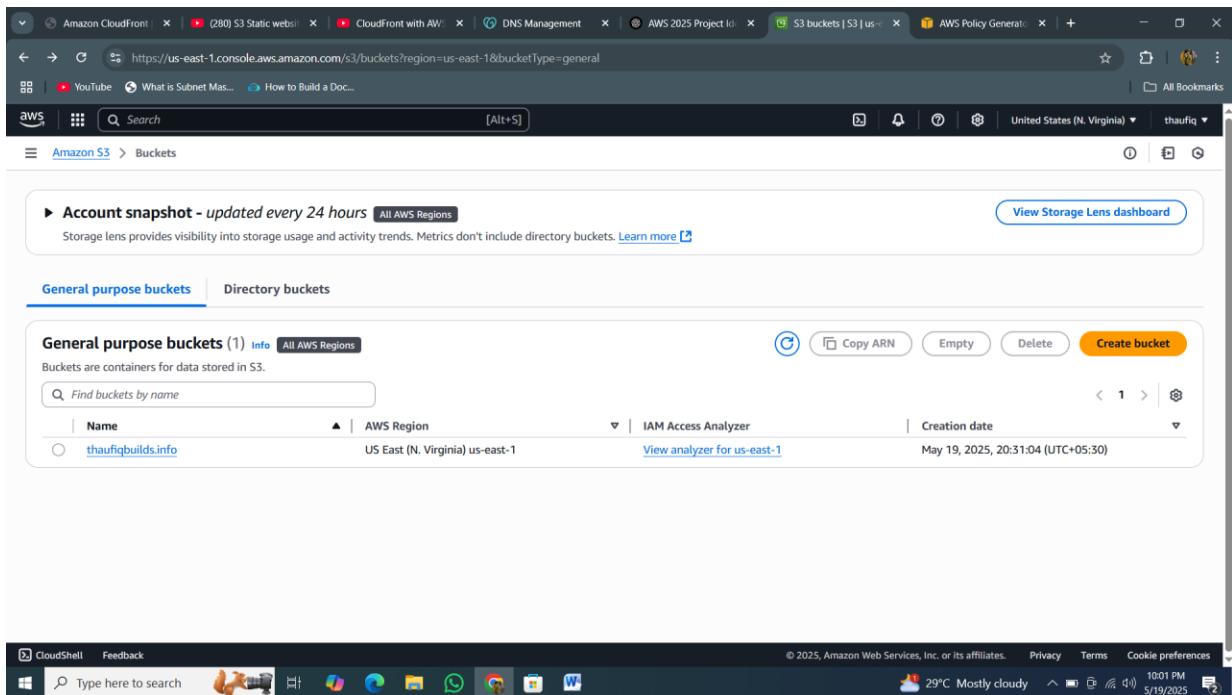
## Performance Optimizations

- **CloudFront caching** reduces latency by serving cached content from global edge locations.
- **Route 53 latency-based routing** (if enabled) ensures users are routed to the lowest-latency CloudFront edge.

# Step 1: S3 Static Website Hosting

Service: Amazon S3

- Created a bucket named `thaufiqbuilds.info`
- Enabled static website hosting
- Uploaded your HTML, CSS, JS files
- Unblocked public access
- Added a bucket policy to allow public read
- Confirmed site accessible via S3 static website endpoint



The screenshot shows the AWS S3 console interface. At the top, there's a navigation bar with tabs like 'Amazon CloudFront', 'CloudFront with AWS', 'DNS Management', 'AWS 2025 Project Id', 'S3 buckets | S3 | us-east-1', and 'AWS Policy Generator'. Below the navigation bar, the URL in the address bar is `https://us-east-1.console.aws.amazon.com/s3/buckets?region=us-east-1&bucketType=general`. The main content area has a header 'Account snapshot - updated every 24 hours [All AWS Regions]' with a link to 'View Storage Lens dashboard'. Below this, there are two tabs: 'General purpose buckets' (selected) and 'Directory buckets'. Under 'General purpose buckets', there's a table with one row:

Name	AWS Region	IAM Access Analyzer	Creation date
<a href="#">thaufiqbuilds.info</a>	US East (N. Virginia) us-east-1	<a href="#">View analyzer for us-east-1</a>	May 19, 2025, 20:31:04 (UTC+05:30)

At the bottom of the page, there's a dark footer bar with icons for CloudShell, Feedback, a search bar, and various system status indicators like battery level, temperature (29°C), and date/time (May 19, 2025).

**Objects** (3)

Name	Type	Last modified	Size	Storage class
index.html	html	May 19, 2025, 20:35:25 (UTC+05:30)	543.0 B	Standard
projects.html	html	May 19, 2025, 20:35:24 (UTC+05:30)	891.0 B	Standard
style.css	css	May 19, 2025, 20:35:23 (UTC+05:30)	969.0 B	Standard

**Requester pays**

When enabled, the requester pays for requests and data transfer costs, and anonymous access to this bucket is disabled. [Learn more](#)

**Requester pays**  
Disabled

**Static website hosting**

Use this bucket to host a website or redirect requests. [Learn more](#)

We recommend using AWS Amplify Hosting for static website hosting  
Deploy a fast, secure, and reliable website quickly with AWS Amplify Hosting. Learn more about [Amplify Hosting](#) or [View your existing Amplify apps](#)

**S3 static website hosting**  
Enabled

**Hosting type**  
Bucket hosting

**Bucket website endpoint**  
When you configure your bucket as a static website, the website is available at the AWS Region-specific website endpoint of the bucket. [Learn more](#)  
<http://thaufiqbuilds.info.s3-website-us-east-1.amazonaws.com>

The screenshot shows the AWS S3 Bucket policy configuration page for the bucket 'thauiqbuilds.info'. The left sidebar shows navigation options like 'Amazon S3', 'General purpose buckets', and 'Storage Lens'. The main content area displays a JSON-based bucket policy:

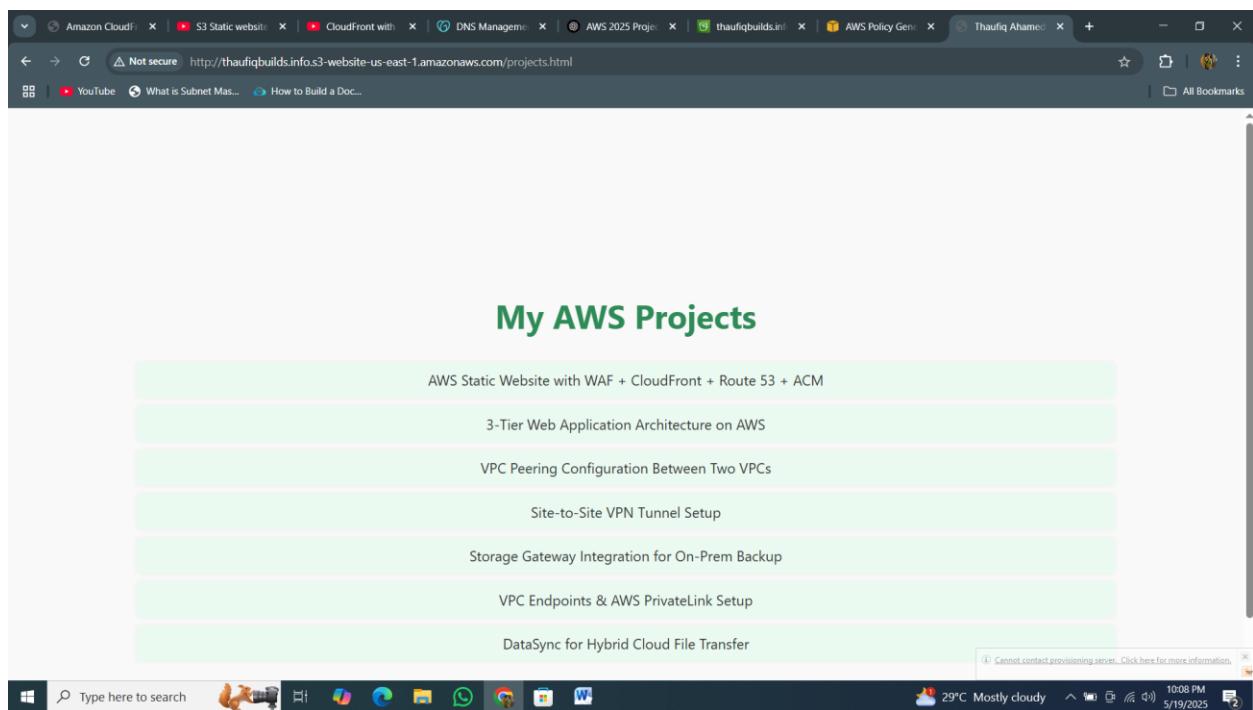
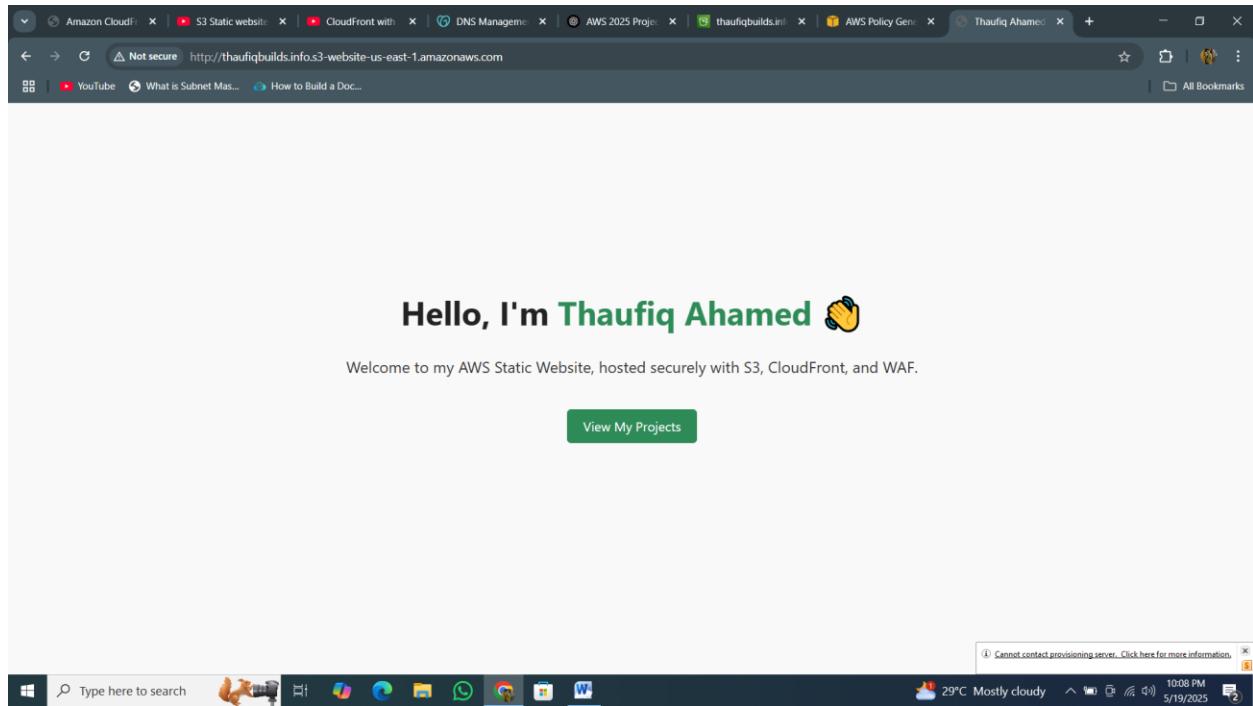
```
{
  "Version": "2012-10-17",
  "Id": "Policy1747667715704",
  "Statement": [
    {
      "Sid": "Stmt1747667713374",
      "Effect": "Allow",
      "Principal": "*",
      "Action": "s3:GetObject",
      "Resource": "arn:aws:s3:::thauiqbuilds.info/*"
    }
  ]
}
```

Buttons for 'Edit' and 'Delete' are visible at the top right, and a 'Copy' button is located on the right side of the policy text.

The screenshot shows a web browser displaying the Google search results for the URL 'http://thauiqbuilds.info.s3-website-us-east-1.amazonaws.com'. The search bar at the top contains the query 'Search Google or type a URL'. Below the search bar, the Google logo is displayed. A dropdown menu shows the following results:

- http://thauiqbuilds.info.s3-website-us-east-1.amazonaws.com
- http://thauiqbuilds.info.s3-website-us-east-1.amazonaws.com
- http://thauiqbuilds.info.s3-website-us-east-1.amazonaws.com - Google Search

The browser interface includes a toolbar with icons for CloudShell, Feedback, and various applications. The status bar at the bottom shows the date and time as 5/19/2025, 10:08 PM, and the weather as 29°C Mostly cloudy.



## Step 2: Create a Public Hosted Zone in Route 53

### Service: Route 53

1. Go to Route 53 → Hosted Zones → **Create Hosted Zone**
2. Enter your domain: `thaufiqbuilds.info`
3. Copy the 4 **NameServer (NS) records**

The screenshot shows the 'Get started' section of the AWS Route 53 console. It features a grid of six cards, each with an icon and a brief description:

- Register a domain**: Register the name, such as example.com, that your users use to access your application. (Icon: shield with '53' and a laptop)
- Transfer domain**: You can transfer domain names to Route 53 that you registered with another domain registrar. (Icon: shield with '53' and a laptop)
- Create hosted zones**: A hosted zone tells Route 53 how to respond to DNS queries for a domain such as example.com. (Icon: three shields with '53')
- Configure health checks**: Health checks monitor your applications and web resources, and direct DNS queries to healthy resources. (Icon: heart rate monitor with '53')
- Configure traffic flow**: A visual tool that lets you easily create policies for multiple endpoints in complex configurations. (Icon: network diagram with '53' and endpoints)
- Configure resolvers**: A regional service that lets you route DNS queries between your VPCs and your network. (Icon: server racks and cloud with '53')

At the bottom right of the grid are 'Cancel' and 'Get started' buttons. The 'Get started' button is highlighted with a yellow background.

At the very bottom of the screen, the Windows taskbar is visible with icons for CloudShell, Feedback, Start, Task View, File Explorer, Microsoft Edge, WhatsApp, Google Chrome, Microsoft Word, and Microsoft Excel. The system tray shows the date (5/19/2025), time (10:16 PM), battery level (29%), and signal strength.

Screenshot of the AWS Route 53 console showing the creation of a new hosted zone.

The URL is <https://us-east-1.console.aws.amazon.com/route53/v2/hostedzones?region=us-east-1#CreateHostedZone>.

**Domain name:** `thaufiqbuilds.info`

**Description - optional:** `The hosted zone is used for...`

**Type:**  **Public hosted zone** (A public hosted zone determines how traffic is routed on the internet.)

**Tags:** `Add tag`

No tags associated with the resource.

Bottom navigation bar: CloudShell, Feedback, Search, AWS logo, Global dropdown, and a date/time indicator: 10:16 PM 5/19/2025.

Screenshot of the AWS Route 53 console showing the details of a newly created hosted zone.

The URL is <https://us-east-1.console.aws.amazon.com/route53/v2/hostedzones?region=us-east-1#ListRecordSets?hostedZoneId=Z077244127ZK4G2U6FTAD>.

**Route 53:** Hosted zones (selected), IP-based routing, Traffic flow, Domains, Resolver.

**Hosted zones:** `thaufiqbuilds.info was successfully created.`

**Records (2):**

Record Type	Name	Type	TTL	Value/Route traffic to
NS	thaufiqbu...	Simple	-	No ns-639.awsdns-15.net. ns-1323.awsdns-37.org. ns-1704.awsdns-21.co.uk. ns-297.awsdns-37.com
SOA	thaufiqbu...	Simple	-	No ns-639.awsdns-15.net.awsd...

Bottom navigation bar: CloudShell, Feedback, Search, AWS logo, Global dropdown, and a date/time indicator: 10:17 PM 5/19/2025.

## Step 3: Update GoDaddy Nameservers to Point to Route 53

### Outside AWS (GoDaddy)

1. Go to GoDaddy → My Products → DNS
2. Find your domain (`thaufiqbuilds.info`)
3. Change nameservers to **Custom**
4. Paste the 4 NS records from Route 53

The screenshot shows the GoDaddy DNS Management interface. The left sidebar has sections for Domains, Portfolio, DNS, Transfers, Services, and Settings. The main area is titled "DNS Management" and shows the domain "thaufiqbuilds.info". Below the domain name are tabs for DNS Records, Forwarding, Nameservers, Premium DNS, Hostnames, DNSSEC, and Crypto Wallet. The "Nameservers" tab is selected. A note says "Nameservers determine where your DNS is hosted and where you add, edit or delete your DNS records." It shows "Using default nameservers" and a "Change Nameservers" button. At the bottom, there's a Windows taskbar with various icons and system status information.

The screenshot shows a web browser window with multiple tabs open, including 'DNS Management', 'AWS 2025 Project Ideas', and 'thauqibuilds.info - S3 bucket'. The main content area displays the 'Edit nameservers' dialog box for the domain 'thauqibuilds.info'. The dialog box has the heading 'Edit nameservers' and the sub-instruction 'Choose nameservers for thauqibuilds.info'. It includes two radio button options: 'GoDaddy Nameservers (recommended)' and 'I'll use my own nameservers' (which is selected). Below this are two input fields labeled 'Nameserver 1' and 'Nameserver 2', both currently empty. At the bottom of the dialog are 'Save' and 'Cancel' buttons. The background of the browser shows a sidebar with 'Domains', 'Portfolio', 'DNS', 'Transfers', 'Services', 'Settings', and 'Activity Log'. A note at the bottom right of the dialog box says 'facebook.thauqibuilds.info to link to your social media page'.

This screenshot is identical to the one above, but it shows four nameservers listed in the 'Nameserver 1' field of the 'Edit nameservers' dialog box. The nameservers are: 'ns-639.awsdns-15.net', 'ns-1323.awsdns-37.org', 'ns-1704.awsdns-21.co.uk', and 'ns-297.awsdns-37.com'. Each name is preceded by a trash can icon for deletion. The rest of the interface, including the sidebar and the note about social media linking, remains the same.

The screenshot shows a web browser window with multiple tabs open. The active tab is 'https://dcc.godaddy.com/control/dnsmanagement?domainName=thaufiqbuilds.info&subtab=nameservers'. The page displays a modal dialog titled 'Edit nameservers' for the domain 'thaufiqbuilds.info'. The dialog contains a warning message: 'By clicking Continue, you consent to updating the nameservers for the selected domain(s). Changing nameservers is risky, and could potentially lead to your website disappearing from public view.' Below the message are two buttons: 'Continue' (dark gray) and 'Cancel' (light gray). The background of the main page shows a sidebar with 'DNS', 'Transfers', 'Services', and 'Settings' options, and a central area with a 'Nameservers' section. A sub-section titled 'Using default nameservers' is visible. A 'Change Nameservers' button is located in the top right corner of the main content area. At the bottom of the screen, there is a Windows taskbar with various pinned icons and a weather widget showing '29°C Mostly cloudy'.

This screenshot shows the same GoDaddy DNS Management interface as the previous one, but the 'Nameservers' section is now populated with a list of custom nameservers. The list includes:

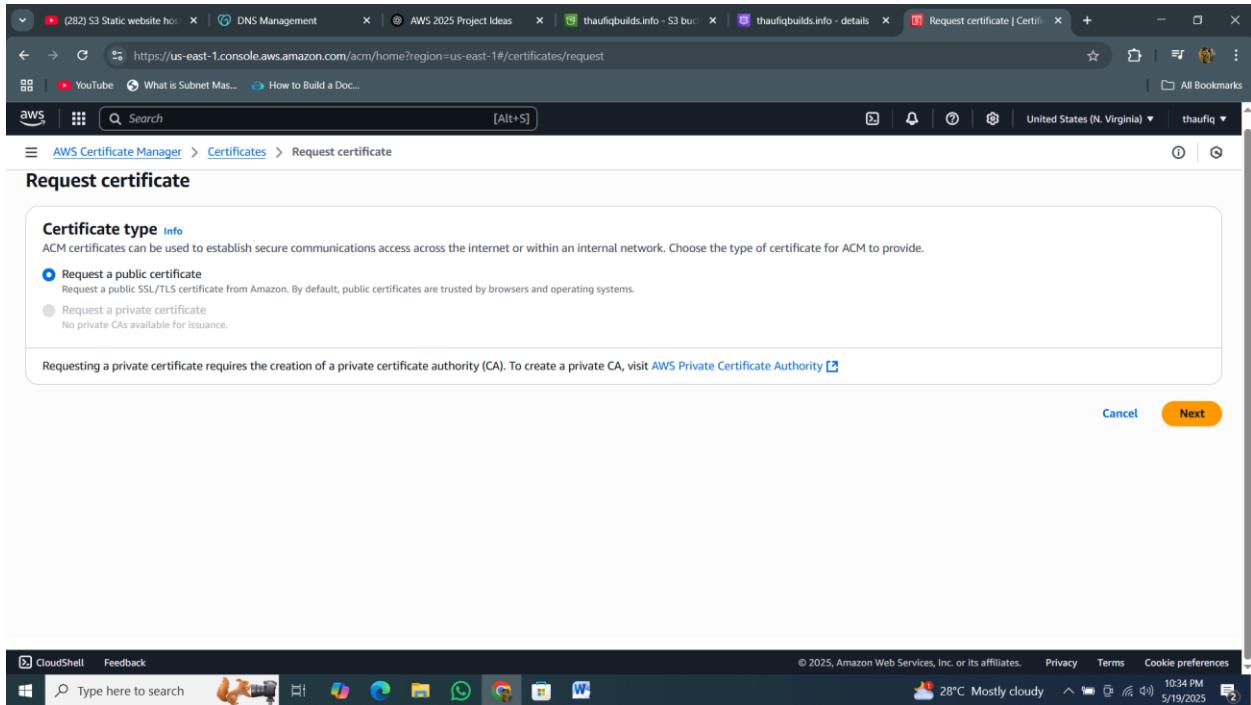
- ns-639.awsdns-15.net
- ns-1323.awsdns-37.org
- ns-1704.awsdns-21.co.uk
- ns-297.awsdns-37.com

The interface includes a sidebar with icons for Portfolio, Transfers, Services, and Settings. A 'Change Nameservers' button is at the top right. A cookie consent banner at the bottom states: 'We serve cookies. We use tools, such as cookies, to enable essential services and functionality on our site and to collect data on how visitors interact with our site, products and services. By clicking Accept, you agree to our use of these tools for advertising, analytics and support.' It offers 'Manage', 'Decline', and 'Accept' buttons. The Windows taskbar at the bottom is identical to the one in the first screenshot.

## Step 4: Request SSL Certificate in AWS ACM

Service: AWS Certificate Manager (ACM)

1. Go to ACM → Request certificate → **Public certificate**
2. Enter domain name: `thaufiqbuilds.info` (also add `www.thaufiqbuilds.info` if needed)
3. Choose **DNS validation**
4. ACM will show a CNAME record → Go to Route 53 → Add that as a **new record**
5. Wait until ACM shows "**Issued**"



The screenshot shows the 'Request public certificate' page in the AWS Certificate Manager. In the 'Domain names' section, the 'Fully qualified domain name' field contains 'thaufiqbuilds.info'. Below it, there's a link to 'Add another name to this certificate'. In the 'Validation method' section, the 'DNS validation - recommended' option is selected. The status message indicates 'Choose this option if you are authorized to modify the DNS configuration for the domains in your certificate request.' In the 'Key algorithm' section, 'RSA 2048' is selected. The status message says 'Select an encryption algorithm. Some algorithms may not be supported by all AWS services.' At the bottom right, there are 'Cancel', 'Previous', and 'Request' buttons.

This screenshot continues from the previous one, showing the 'Request public certificate' page. In the 'Validation method' section, the 'Email validation' option is selected, with the status message 'Choose this option if you do not have permission or cannot obtain permission to modify the DNS configuration for the domains in your certificate request.' In the 'Key algorithm' section, 'RSA 2048' is selected again. The status message is identical to the previous section. At the bottom right, there are 'Cancel', 'Previous', and 'Request' buttons.

CLICK on create record in route 53 and automatically a record is created route53 then only ssl certificate is issued

The screenshot shows the AWS Certificate Manager (ACM) interface. On the left, a sidebar lists options like 'List certificates', 'Request certificate', 'Import certificate', and 'AWS Private CA'. The main area displays the 'Certificate status' for a certificate with Identifier '84b58563-a718-4152-a611-8a92ebaf888d'. It shows the ARN 'arn:aws:acm:us-east-1:688567302802:certificate/84b58563-a718-4152-a611-8a92ebaf888d' and Type 'Amazon Issued'. The status is 'Issued'. Below this, a table titled 'Domains (1)' lists the domain 'thaufiqbuilds.info' with a status of 'Success' and type 'CNAME'. A CNAME value is partially visible: '\_b63b4d165fc0bb13a5ada2696596f9.thaufiq...'. At the bottom, there's a 'Details' section with columns for In use, Serial number, Requested at, and Renewal eligibility.

## Step 5: Create CloudFront Distribution

Service: Amazon CloudFront

1. Go to the CloudFront console and click **Create Distribution**.
2. For **Origin domain**, enter your **S3 bucket REST endpoint** (not the static website endpoint).  
Example: `thaufiqbuilds.info.s3.<region>.amazonaws.com`
3. Under **Origin Access**, enable **Origin Access Identity (OAI)**:
  - o Create a new OAI or choose an existing one.
  - o This allows CloudFront to securely access your private S3 bucket.
4. Update your **S3 bucket policy** to allow access only from the OAI and block all public access.
5. For **Viewer protocol policy**, select **Redirect HTTP to HTTPS** to enforce secure access.

6. In **Alternate domain names (CNAMEs)**, add your custom domain: `thaufiqbuilds.info`.
7. For **SSL Certificate**, choose **Custom SSL Certificate** (for example, from ACM) and select your certificate.
8. Leave other settings as default and click **Create Distribution**.

**NOTE :** The S3 static website endpoint is a public endpoint designed for hosting static content and does not support private access mechanisms like OAI. If the bucket is public, anyone can access the website URL directly, bypassing CloudFront.

#### . Solution / Best Practice:

To restrict access **only via CloudFront**, use the **S3 bucket REST endpoint** (e.g., `thaufiqbuilds.info.s3.<region>.amazonaws.com`) as the CloudFront origin and enable **Origin Access Identity (OAI)**. Then, make the bucket private by blocking public access. This ensures only CloudFront can fetch the content and users must access via CloudFront

The screenshot shows the 'Edit Block public access (bucket settings)' page for the 'thaufiqbuilds.info' bucket. On the left, there's a sidebar with navigation links for Amazon S3, General purpose buckets, Storage Lens, and a Feature spotlight section. The main content area is titled 'Edit Block public access (bucket settings)'. It contains a section titled 'Block public access (bucket settings)' with a detailed description of what it does. Below this, there are four checkboxes under the heading 'Block all public access':

- Block all public access**: Turning this setting on is the same as turning on all four settings below. Each of the following settings are independent of one another.
- Block public access to buckets and objects granted through new access control lists (ACLs)**: S3 will block public access permissions applied to newly added buckets or objects, and prevent the creation of new public access ACLs for existing buckets and objects. This setting doesn't change any existing permissions that allow public access to S3 resources using ACLs.
- Block public access to buckets and objects granted through any access control lists (ACLs)**: S3 will ignore all ACLs that grant public access to buckets and objects.
- Block public access to buckets and objects granted through new public bucket or access point policies**: S3 will block new bucket and access point policies that grant public access to buckets and objects. This setting doesn't change any existing policies that allow public access to S3 resources.
- Block public and cross-account access to buckets and objects through any public bucket or access point policies**: S3 will ignore public and cross-account access for buckets or access points with policies that grant public access to buckets and objects.

At the bottom right of the page are 'Cancel' and 'Save changes' buttons. The browser's address bar shows the URL: `https://us-east-1.console.aws.amazon.com/s3/bucket/thaufiqbuilds.info/property/bpa/edit?region=us-east-1&bucketType=general`.

Even though we made the buck private the user can access the the bucket via cloudfront right ( through the bucket policy)

## Let's create CloudFront distribution

The screenshot shows the 'Create distribution' wizard on the AWS CloudFront console. In the first step, 'Distribution options', the 'Single website or app' option is selected, indicating it's for a single app or website. The 'Multi-tenant architecture - New' option is also available for multiple domains.

**Origin**  
The 'Origin domain' field contains 'thaufiqbuilds.info.s3.us-east-1.amazonaws.com'. A note states: 'This S3 bucket has static web hosting enabled. If you plan to use this distribution as a website, we recommend using the S3 website endpoint rather than the bucket endpoint.' A 'Use website endpoint' button is present.

**Origin path - optional**  
The 'Enter the origin path' field is empty.

**Name**  
The 'Name' field contains 'thaufiqbuilds.info.s3.us-east-1.amazonaws.com'. The 'Origin access' section includes options for 'Public', 'Origin access control settings (recommended)', and 'Legacy access identities'. The 'Origin access control' section lists 'thaufiqbuilds.info.s3.us-east-1.amazonaws.com'. A note says: 'You must update the S3 bucket policy. CloudFront will provide you with the policy statement after creation.' The 'Add custom header - optional' section has an 'Add header' button. The 'Enable Origin Shield' section has a 'No' option selected.

The 'Create new OAC' dialog box is open. It requires a 'Name' (entered as 'thaufiqbuilds.info.s3.us-east-1.amazonaws.com') and a 'Description - optional' (left empty). Under 'Signing behavior', 'Sign requests (recommended)' is selected. Under 'Origin type', 'S3' is chosen. At the bottom are 'Cancel' and 'Create' buttons.

**Origin path - optional**  
Enter a URL path to append to the origin domain name for origin requests.

**Name**  
Enter a name for this origin.

**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.

**Origin access control**  
Select an existing origin access control (recommended) or create a new control.  
 [Create new OAC](#)

**⚠ You must update the S3 bucket policy**  
CloudFront will provide you with the policy statement after creating the distribution.

**Add custom header - optional**  
CloudFront includes this header in all requests that it sends to your origin.

**Path pattern** | [Info](#)

**Compress objects automatically** | [Info](#)  
 No  
 Yes

**Viewer**

**Viewer protocol policy**  
 HTTP and HTTPS  
 Redirect HTTP to HTTPS  
 HTTPS only

**Allowed HTTP methods**  
 GET, HEAD  
 GET, HEAD, OPTIONS  
 GET, HEAD, OPTIONS, PUT, POST, PATCH, DELETE

**Restrict viewer access**  
If you restrict viewer access, viewers must use CloudFront signed URLs or signed cookies to access your content.  
 No  
 Yes

**Cache key and origin requests**  
We recommend using a cache policy and origin request policy to control the cache key and origin requests.

**Restrict viewer access**  
If you restrict viewer access, viewers must use CloudFront signed URLs or signed cookies to access your content.  
 No  
 Yes

**Cache key and origin requests**  
We recommend using a cache policy and origin request policy to control the cache key and origin requests.

The screenshot shows the 'Create Distribution' wizard in the AWS CloudFront console. The first step, 'Select distribution settings', is completed. The second step, 'Configure distribution', is currently being worked on.

**Alternate domain name (CNAME) - optional**  
Add the custom domain names that you use in URLs for the files served by this distribution.

**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).

**Legacy clients support - \$600/month prorated charge applies. Most customers do not need this.**  
CloudFront allocates dedicated IP addresses at each CloudFront edge location to serve your content over HTTPS.  
 Enabled

**Security policy**  
The security policy determines the SSL or TLS protocol and the specific ciphers that CloudFront uses for HTTPS connections with viewers (clients).  
 TLSv1\_2\_2021 (recommended)  
 TLSv1\_2\_2019  
 TLSv1\_2\_2018  
 TLSv1\_1\_2016  
 TLSv1\_2016  
 TLSv1

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences 28°C Mostly cloudy 11:45 PM 5/19/2025

The screenshot shows the 'Edit settings' page for distribution E108HRE2FO661Z. The 'TLSv1\_2\_2021 (recommended)' security policy is selected.

**Supported HTTP versions**  
Add support for additional HTTP versions. HTTP/1.0 and HTTP/1.1 are supported by default  
 HTTP/2  
 HTTP/3

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

**IPv6**  
 Off  
 On

**Description - optional**

Cancel Save changes

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences 28°C Mostly cloudy 12:00 AM 5/20/2025

The screenshot shows the AWS CloudFront distribution settings page for distribution E108HRE2FO661Z. A green success message at the top states "Successfully updated distribution settings." The distribution domain name is d36m1bssi435ij.cloudfront.net, and the ARN is arn:aws:cloudfront::688567302802:distribution/E108HRE2FO661Z. The last modified status is "Deploying". The "Settings" section includes fields for Description (empty), Alternate domain names (thauqibuilds.info), Standard logging (Off), Price class (Use all edge locations (best performance)), Custom SSL certificate (thauqibuilds.info), Cookie logging (Off), Supported HTTP versions (HTTP/2, HTTP/1.1, HTTP/1.0), Security policy (TLSv1.2\_2021), and Default root object (empty). A "Create staging distribution" button is present.



The screenshot shows the AWS CloudFront distribution settings page for distribution E108HRE2FO661Z. A yellow warning message at the top states "The S3 bucket policy needs to be updated" and "Complete distribution configuration by allowing read access to CloudFront origin access control in your policy statement. Go to S3 bucket permissions to update policy." The distribution domain name is d36m1bssi435ij.cloudfront.net, and the ARN is arn:aws:cloudfront::688567302802:distribution/E108HRE2FO661Z. The last modified status is "Deploying". The "Settings" section includes fields for Description (empty), Alternate domain names (empty), Standard logging (Off), Price class (Use all edge locations (best performance)), Custom SSL certificate (thauqibuilds.info), Cookie logging (Off), Supported HTTP versions (HTTP/2, HTTP/1.1, HTTP/1.0), Security policy (TLSv1.2\_2021), and Default root object (empty). A "Create staging distribution" button is present.



Copy the bucket policy and edit the bucket policy

```

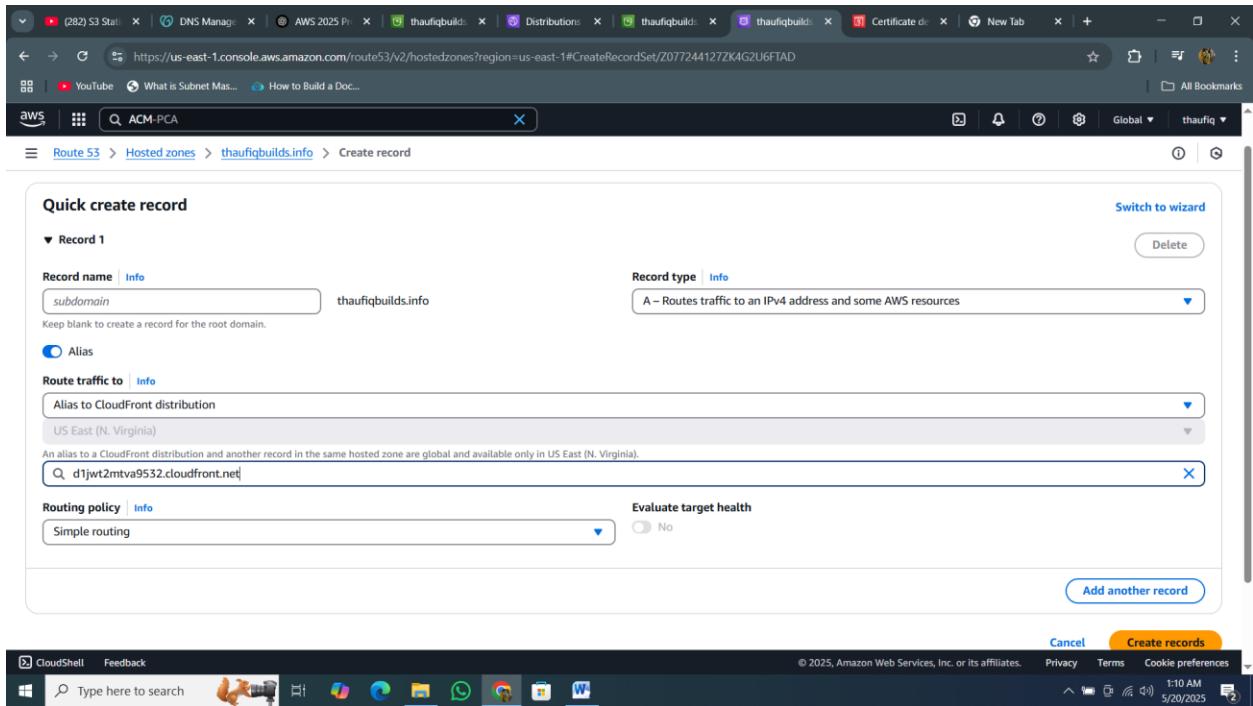
{
    "Version": "2008-10-17",
    "Id": "PolicyForCloudFrontPrivateContent",
    "Statement": [
        {
            "Sid": "AllowCloudFrontServicePrincipal",
            "Effect": "Allow",
            "Principal": [
                "Service": "cloudfront.amazonaws.com"
            ],
            "Action": "s3:GetObject",
            "Resource": "arn:aws:s3:::thaufiqbuilds.info/*",
            "Condition": {
                "StringEquals": {
                    "AWS:SourceArn": "arn:aws:cloudfront::688567302802:distribution/E37XN210NTFX3A"
                }
            }
        }
    ]
}

```

## Step 6: Point Domain to CloudFront in Route 53

### Service: Route 53

1. Go to Hosted Zone → thaufiqbuilds.info
2. Click **Create Record**
  - o Record type: A (IPv4)
  - o Alias: Yes
  - o Alias target: Select your **CloudFront distribution**
3. Save



## Step 7: Add AWS WAF (Web Application Firewall)

### 1. Go to WAF

- Open AWS Console → Search "WAF & Shield" → Open it.

### 2. Create Web ACL

- Click "Web ACLs" on the left → Click "Create web ACL".

### 3. Web ACL Settings

- **Name:** S3Website-WAF
- **Region:**  Select **CloudFront (Global)** (important — WAF for CloudFront is global).
- **Resource type:** Leave as default.
- Click **Next**.

## 4. Add Rules (you can customize later)

You can:

- Add **Managed Rules** (recommended):
  - Click **Add managed rule groups** → Choose:
    - AWS-AWSManagedRulesCommonRuleSet (blocks known bad requests)
    - AWS-AWSManagedRulesAmazonIpReputationList
  - Click **Add rule groups**.
- You can also add your own custom rules later (for IP blocking, geo restrictions, etc.).

Click **Next**.

## 5. Default Action

- Choose: **Allow** (block only if rules match).

## 6. Logging & Metrics (Optional)

- You can enable logging to an S3 bucket or CloudWatch logs (recommended for production).

Click **Next**.

The screenshot shows the AWS WAF & Shield console with the 'Web ACLs' page. A green success message at the top states: 'Success' and 'You successfully deleted web ACL S3Website-WAF.' On the left, a sidebar menu includes sections for AWS WAF (Getting started, Web ACLs, Bot control dashboard, Application integration, IP sets, Regex pattern sets, Rule groups, Add-on protections) and AWS Shield (Getting started, Overview, Protected resources). The main content area displays a table titled 'Web ACLs (0)' with one entry: 'No web ACLs found'. Below the table, a note says: 'You don't have any web ACLs in the Global (CloudFront) Region created with this version of AWS WAF.' and 'Resources created under AWS WAF Classic aren't compatible with the new AWS WAF.' A 'Create web ACL' button is located at the bottom of the table area. The browser's address bar shows the URL: https://us-east-1.console.aws.amazon.com/wafv2/homev2/web-acls?region=global.

Step 1  
Describe web ACL and associate it to AWS resources

Step 2  
Add rules and rule groups

Step 3  
Set rule priority

Step 4  
Configure metrics

Step 5  
Review and create web ACL

Resource type  
Choose the type of resource to associate with this web ACL. Changing this setting will reset the page.  
 Global resources (CloudFront Distributions, CloudFront Distribution Tenants and AWS Amplify Applications)  
 Regional resources (Application Load Balancers, Amazon API Gateway REST APIs and AWS AppSync APIs)

Name  
S3Website-WAF

Description - optional

CloudWatch metric name  
S3Website-WAF

CloudFront metric name

Add AWS resources

Resource type  
Select the resource type and then select the resource you want to associate with this web ACL.

CloudFront Distributions  CloudFront Distribution Tenants  AWS Amplify

Resources (1)  
Select the resource you want to associate with the web ACL.

Find AWS resources to associate

Name  
E37XN210NTFX3A - thaufiqbuilds.info

Cancel Add

CloudFront distribution and AWS Amplify  
Default  
16 KB

The screenshot shows the AWS WAF console at <https://us-east-1.console.aws.amazon.com/wafv2/homev2/web-acls/new?region=global>. The page is titled "Create web ACL".

**CloudWatch metric name:** S3Website-WAF

**Associated AWS resources - optional (1)**

Name	Resource type	Region
E37XN210NTFX3A - thaufiqbuilds.info	CloudFront Distribution	Global (CloudFront)

**Web request body inspection - optional**

By default, rules that inspect the web request body are limited to the first 16 KB of content. You can increase this size for additional costs. [AWS WAF Pricing](#)

**Body size limit:** The AWS WAF default limit is 16 KB. Settings over 16 KB incur additional costs. [Learn more](#)

**CloudFront distribution and AWS Amplify:**

Default  
 16 KB  
 32 KB  
 48 KB

© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

The screenshot shows the AWS WAF console at <https://us-east-1.console.aws.amazon.com/wafv2/homev2/web-acls/new?region=global>. The page is titled "Create web ACL".

**Step 1: Describe web ACL and associate it to AWS resources**

**Add rules and rule groups**

A rule defines attack patterns to look for in web requests and the action to take when a request matches the patterns. Rule groups are reusable collections of rules. You can use managed rule groups offered by AWS and AWS Marketplace sellers. You can also write your own rules and use your own rule groups.

**Rules (0)**

If a request matches a rule, take the corresponding action. The rules are prioritized in order they appear.

Name	Capacity	Action
No rules. You don't have any rules added.		

**Web ACL capacity units (WCUs) used by your web ACL**

The WCUs used by the web ACL will be less than or equal to the sum of the capacities for all of the rules in the web ACL.

The total WCUs for a web ACL can't exceed 5000. Using over 1500 WCUs affects your costs. [AWS WAF Pricing](#)

0/5000 WCUs

© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

https://us-east-1.console.aws.amazon.com/wafv2/homev2/web-acls/new?region=global

AWS WAF Services Search [Alt+S]

**Add managed rule groups** Info

Managed rule groups are created and maintained for you by AWS and AWS Marketplace sellers. Any fees that a managed rule group provider charges for using a managed rule group are in addition to the standard service charges for AWS WAF. [AWS WAF Pricing](#)

Step 1 [Describe web ACL and associate it to AWS resources](#)

Step 2 [Add managed rule groups](#)

Step 3 [Set rule priority](#)

Step 4 [Configure metrics](#)

Step 5 [Review and create web ACL](#)

**AWS managed rule groups**

**Paid rule groups**

AWS WAF charges subscription and usage fees for paid managed rule groups. These are in addition to the standard service charges for AWS WAF. [AWS WAF Pricing](#)

Name	Capacity	Additional fees	Action
<b>Account creation fraud prevention</b> - new	50	<ul style="list-style-type: none"> <li>\$10 per month (prorated hourly).</li> <li>Tiered fee model for requests analyzed <a href="#">AWS WAF Pricing</a></li> </ul>	<input type="radio"/> Add to web ACL
<b>Account takeover prevention</b> Provides protection for your login page against stolen credentials, credential stuffing			

© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences 2:04 AM 5/20/2025

https://us-east-1.console.aws.amazon.com/wafv2/homev2/web-acls/new?region=global

AWS WAF Services Search [Alt+S]

**Admin protection**

Contains rules that allow you to block external access to exposed admin pages. This may be useful if you are running third-party software or would like to reduce the risk of a malicious actor gaining administrative access to your application. [Learn More](#)

100  Add to web ACL

**Amazon IP reputation list**

This group contains rules that are based on Amazon threat intelligence. This is useful if you would like to block sources associated with bots or other threats. [Learn More](#)

25  Add to web ACL

**Anonymous IP list**

This group contains rules that allow you to block requests from services that allow obfuscation of viewer identity. This can include request originating from VPN, proxies, Tor nodes, and hosting providers. This is useful if you want to filter out viewers that may be trying to hide their identity from your application. [Learn More](#)

50  Add to web ACL

**Core rule set**

Contains rules that are generally applicable to web applications. This provides protection against exploitation of a wide range of vulnerabilities, including those described in OWASP publications. [Learn More](#)

700  Add to web ACL

**Known bad inputs**

Contains rules that allow you to block request patterns that are known to be invalid and are associated with exploitation or discovery of vulnerabilities. This can help reduce the risk of a malicious actor discovering a vulnerable application. [Learn More](#)

200  Add to web ACL

**Linux operating system**

Contains rules that block request patterns associated with exploitation of vulnerabilities specific to Linux, including LFI attacks. This can help prevent attacks that expose file contents or execute code for which the attacker should not have had access. [Learn More](#)

200  Add to web ACL

**PHP application**

Contains rules that block request patterns associated with exploiting vulnerabilities specific to the use of PHP, including injection of unsafe PHP functions. This can help

100  Add to web ACL

© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences 2:04 AM 5/20/2025 29°C Mostly cloudy

The screenshot shows the 'Configure metrics' step of the AWS WAF 'Create web ACL' wizard. On the left, a sidebar lists steps 1 through 5: 'Describe web ACL and associate it to AWS resources', 'Add rules and rule groups', 'Set rule priority', and 'Configure metrics'. The current step is 'Configure metrics'. The main area is titled 'Configure metrics' and contains two sections: 'Amazon CloudWatch metrics' and 'Request sampling options'. In the 'Amazon CloudWatch metrics' section, under 'Rules', there is a checkbox for 'AWS-AWSManagedRulesCommonRuleSet' which is checked, and a text input field containing 'AWS-AWSManagedRulesCommonRuleSet'. In the 'Request sampling options' section, there are three radio button options: 'Enable sampled requests' (selected), 'Disable sampled requests', and 'Enable sampled requests with exclusions'. At the bottom right are 'Cancel', 'Previous', and 'Next' buttons.

The screenshot shows the 'S3Website-WAF' configuration page in the AWS WAF & Shield console. The left sidebar under 'WAF & Shield' has 'AWS WAF' expanded, showing 'Getting started', 'Web ACLs' (which is selected and highlighted in blue), 'Bot control dashboard', 'Application integration', 'IP sets', 'Regex pattern sets', 'Rule groups', and 'Add-on protections'. Below that is a link to 'Switch to AWS WAF Classic'. The 'AWS Shield' section is also present with 'Getting started', 'Overview', and 'Protected resources'. The main content area is titled 'S3Website-WAF' and shows an 'Associated AWS resources' section with the ARN 'arn:aws:wafv2:us-east-1:688567302802:global/webacl/S3Website-WAF/f4516d37-f76e-4671-9ff1-c994be1d58bd'. It includes tabs for 'Traffic overview' (which is selected), 'Rules', 'Associated AWS resources', 'Custom response bodies', 'Logging and metrics', 'Sampled requests', and 'CloudWatch Log Insights'. A message box says 'Top insights are now available with CloudWatch logs' and 'You can access the new top insights dashboard section by enabling a CloudWatch logging destination in your logging configuration.' There is a 'Download web ACL as JSON' button. Below this is a 'Data filters' section with a 'Traffic type' dropdown set to 'All traffic' and a 'Terminating rule actions' dropdown with options 'Blocked', 'Allowed', 'Captcha', and 'Challenge'. At the bottom, there is a link to 'Actions available for Amazon CloudWatch Metrics' and a 'Logs' tab. The status bar at the bottom indicates the URL 'https://us-east-1.console.aws.amazon.com/wafv2/homev2/web-acl/S3Website-WAF/f4516d37-f76e-4671-9ff1-c994be1d58bd/loggingconfiguration?region=global' and the date '5/20/2025'.

**Logging destination**  
Select a destination for your web ACL traffic logs.

CloudWatch Logs log group  
 Amazon Data Firehose stream  
 S3 bucket

**Amazon CloudWatch Logs log group**  
Select a log group in your account that begins with 'aws-waf-logs-' or create one in the Amazon CloudWatch console. You must use a log group that's associated with your account.

aws-waf-logs-001

**Redacted fields**  
Select the data fields that you want to omit from the logs.

Redacted fields

HTTP method  
 Query string  
 URI path  
 Single header

**Filter logs**  
Add filters to control which web requests are logged. If you add multiple filters, AWS WAF evaluates them starting from the top.

No filters

**Cancel** **Save**

Click save

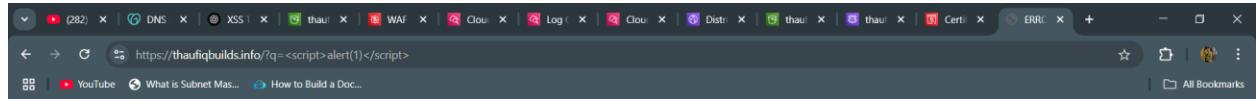
## Core Rule Set (CRS) — SELECT THIS

- Protects against OWASP Top 10 (SQLi, XSS, LFI, etc.).
- Covers general app-layer threats.
- **Must-have for all websites.**

Lets test it WAF will block these and return a 403 Forbidden if it's working correctly 

Test pattern

XSS test – Append to URL: [https://thaufiqbuilds.info/?q=<script>alert\(1\)</script>](https://thaufiqbuilds.info/?q=<script>alert(1)</script>)



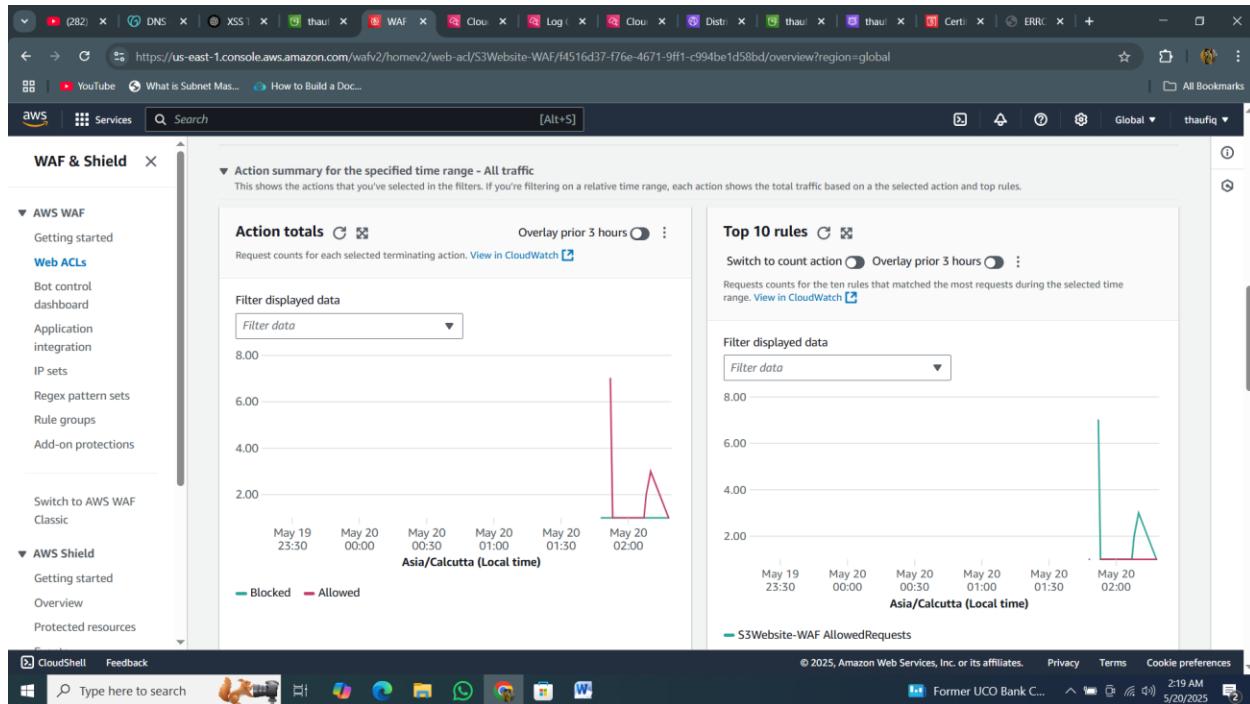
**403 ERROR**

The request could not be satisfied.

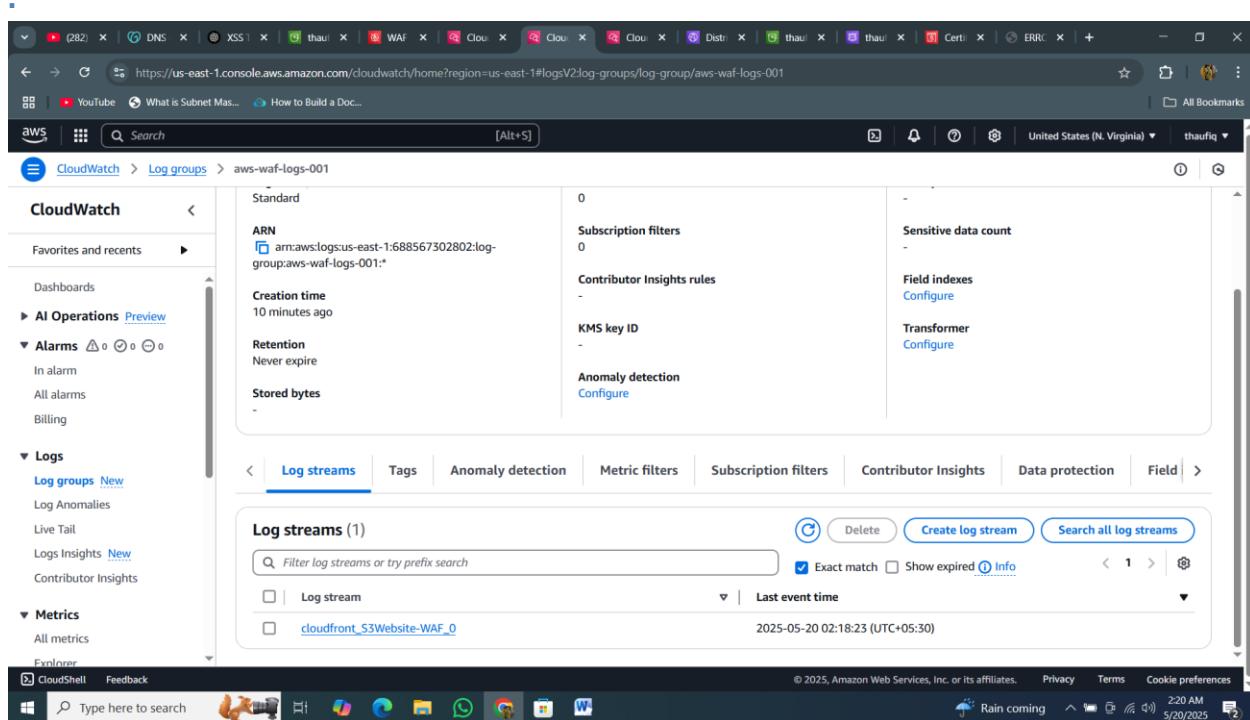
Request blocked. We can't connect to the server for this app or website at this time. There might be too much traffic or a configuration error. Try again later, or contact the app or website owner.  
If you provide content to customers through CloudFront, you can find steps to troubleshoot and help prevent this error by reviewing the CloudFront documentation.

Generated by cloudfront (CloudFront)  
Request ID: aW58w-QM5i16vZ0tRKpAlqufdzSIXFYVeL1neVD1Z5xGz990w1orw==





## Cloud watch log



The screenshot shows the AWS CloudWatch Log Events interface. The left sidebar navigation includes CloudWatch, Favorites and recents, Dashboards, AI Operations (Preview), Alarms, Logs (Log groups New), Metrics, and an Explorer section. The main content area displays 'Log events' with a search bar, time range selector (Clear, 1m, 30m, 1h, 12h, Custom, Local timezone), and a 'Display' dropdown. A message indicates 'No older events at this moment. Retry'. Below this, two log entries are shown:

```
2025-05-20T02:18:23.470+05:30 {"timestamp":1747687703470,"formatVersion":1,"webacId":"arn:aws:wafv2:us-east-1:688567302802:global/webac1/S3Website-WAF/f4516d37-..."}  
2025-05-20T02:18:23.670+05:30 {"timestamp":1747687703670,"formatVersion":1,"webacId":"arn:aws:wafv2:us-east-1:688567302802:global/webac1/S3Website-WAF/f4516d37-..."}
```

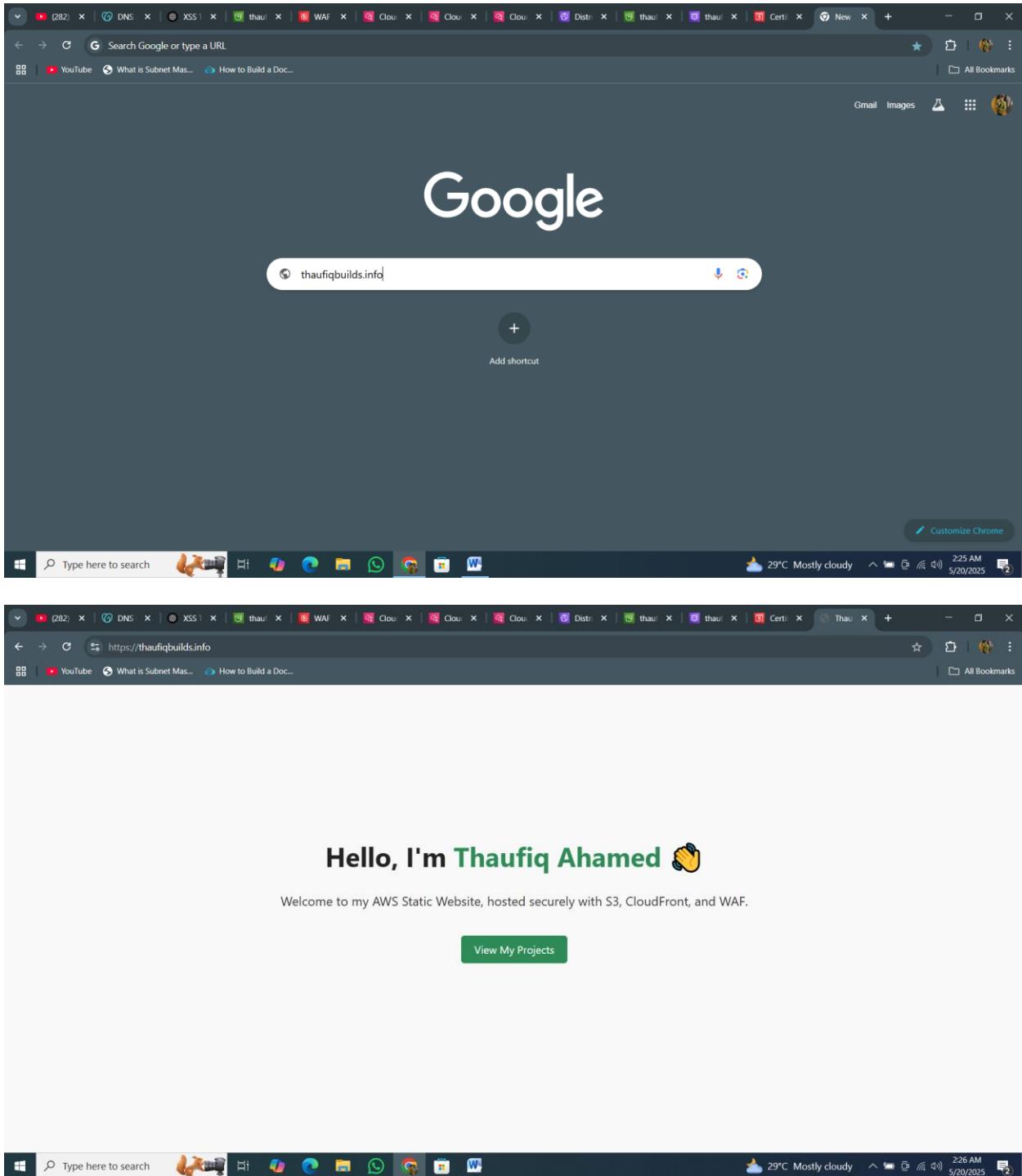
No newer events at this moment. Auto retry paused. Resume

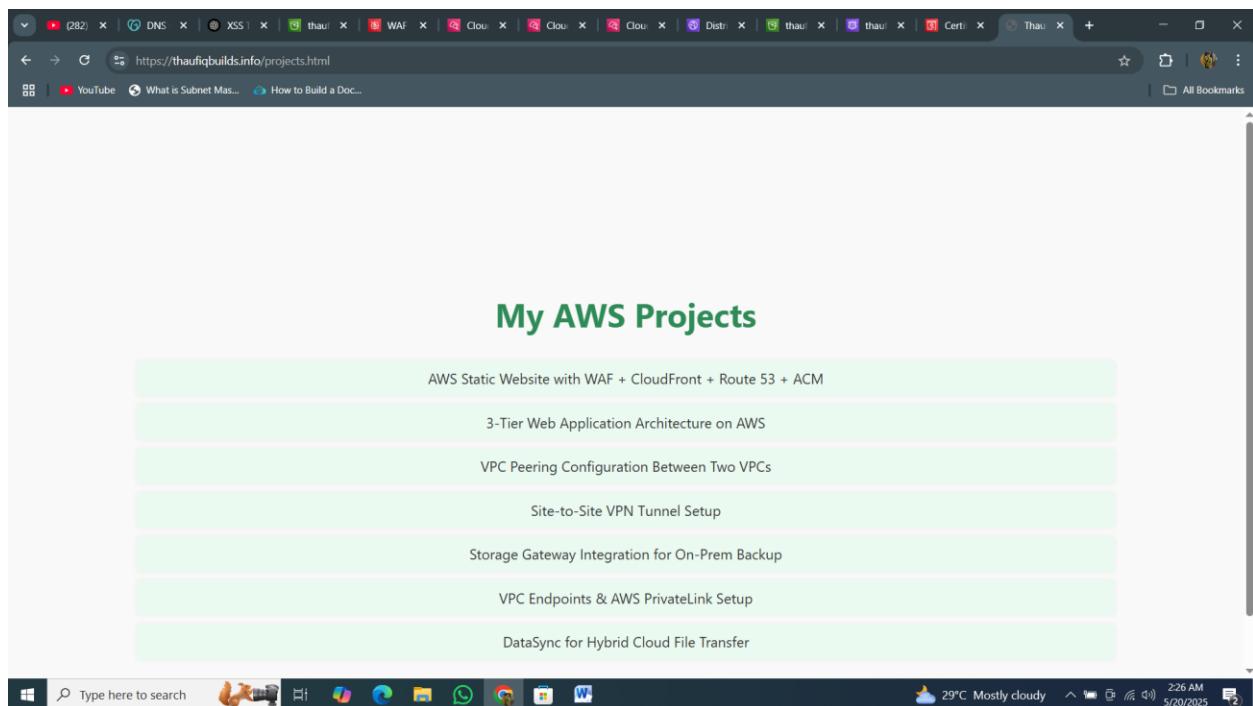
This screenshot is identical to the one above, but the second log entry is expanded to show its full JSON structure. The expanded entry includes fields like timestamp, formatVersion, webacId, and a detailed 'terminatingRuleMatchDetails' array.

```
{
  "timestamp": 1747687703470,
  "formatVersion": 1,
  "webacId": "arn:aws:wafv2:us-east-1:688567302802:global/webac1/S3Website-WAF/f4516d37-f7be-4671-9ff1-c994be1d58bd",
  "terminatingRuleId": "AWS-AWSManagedRulesCommonRuleSet",
  "terminatingRuleType": "MANAGED_RULE_GROUP",
  "action": "BLOCK",
  "terminatingRuleMatchDetails": [
    {
      "conditionType": "XSS",
      "location": "ALL_QUERY_ARGS",
      "matchedData": [
        "c",
        "script"
      ],
      "matchedFieldName": "q"
    }
  ]
}
```

OK done waf it perfectly working

**Know we try the normal with your domine we see the https also the brower**





## Final Result:

You now have:

A secure, fast, global static site at:  
**https://thaufiqbuilds.info**

Protected with:

- **HTTPS**
- **CDN performance**
- **Custom domain**
- **Web firewall**

## Conclusion

This project successfully demonstrates the deployment of a **secure, performant, and highly available static website** using core AWS services. By integrating **Amazon S3**, **Route 53**, **Amazon CloudFront**, **AWS Certificate Manager (ACM)**, and **AWS Web Application Firewall (WAF)**, we have created a modern web hosting architecture that not only delivers content at low latency but also ensures **robust security and domain-level branding**.

Key highlights of this project include:

- **HTTPS-secured access** using ACM-integrated SSL certificates for encrypted communication.

- **Custom domain name** resolution via Route 53 for professional web presence.
- **Global content delivery** powered by CloudFront for reduced latency and caching.
- **WAF protection** using managed rules (like Core Rule Set, IP Reputation, and Anonymous IP List) to block common threats including XSS, injection attempts, and traffic from suspicious IPs.
- **Tested WAF rules** with real-world malicious patterns to confirm rule effectiveness and response.

This architecture follows **AWS Well-Architected Framework** principles, specifically around **security, performance efficiency, and reliability**, making it an ideal choice for hosting modern static websites that require enterprise-grade protection without server-side complexity.