

AWS ROUTE53: To manage DNS & website hosting using S3

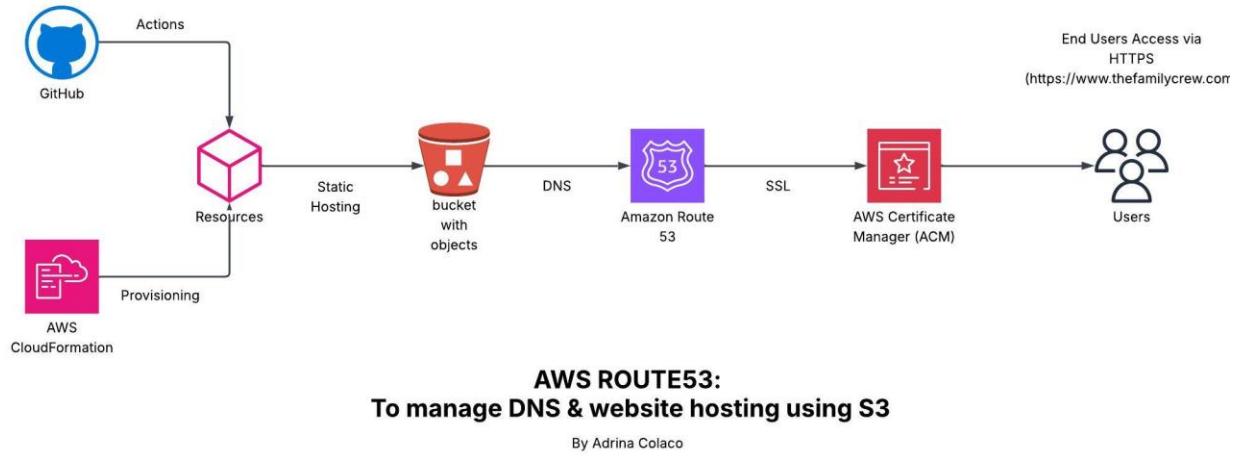
1. Project Overview

The objective is to design, deploy, and maintain a static website for *The Family Crew Company*, a small business specializing in handmade soy candles, bath bombs, crochet items, and hair accessories. The website will be the primary digital platform to showcase products, share the brand story, and facilitate customer engagement. Leveraging AWS services ensures scalability, reliability, and cost-effectiveness, while integrating GitHub enables streamlined content updates.

2. Objective

- **Static Website Hosting:** Host the website on Amazon S3, ensuring high availability and durability.
- **Infrastructure as Code:** Utilize AWS CloudFormation to automate the provisioning and management of AWS resources, promoting consistency and ease of maintenance.
- **Custom Domain Integration:** Use Amazon Route 53 to manage DNS settings, enabling the website to be accessible via the custom domain *thefamilycrew.com*.
- **Secure Communication:** Implement HTTPS using AWS Certificate Manager (ACM) to provide SSL/TLS certificates, ensuring secure data transmission.
- **Continuous Deployment:** Integrate GitHub with AWS to automatically deploy website updates to S3 whenever changes are committed to the repository.

3. Architecture



4. Component Breakdown

- **GitHub Repository:** This repository contains the website's source code and static assets (HTML, CSS, JS).
- **GitHub Actions:** This tool automates the build and deployment process. Every commit to the main branch triggers workflows that package the website and deploy it to the S3 bucket.
- **AWS CloudFormation:** Manages infrastructure as code, provisioning resources like the S3 bucket, Route 53 hosted zone, and ACM certificates.
- **Amazon S3 Bucket:** Hosts the static website files. Configured for static website hosting, it serves the website content directly to users.
- **AWS Certificate Manager (ACM):** Provides SSL/TLS certificates to enable HTTPS for the custom domain.
- **Amazon Route 53:** Manages DNS records, routing traffic from the custom domain to the S3 bucket.
- **End Users:** Access the website securely via HTTPS through the custom domain.

5. Deliverables

- **Amazon S3 Bucket:** Configured for static website hosting with appropriate permissions and policies.
- **CloudFormation Templates:** Scripts to automate the creation and configuration of AWS resources, including S3, IAM.
- **Route 53 Hosted Zone:** Set up to manage DNS records for *thefamilycrew.com*, pointing to the S3 website endpoint.
- **SSL/TLS Certificate:** Provisioned via ACM and associated with the domain to enable HTTPS.
- **GitHub Integration:** Set up GitHub Actions workflows to automatically deploy website content to S3 upon commits to the repository.

6. Technical Requirements

- **AWS Services:** Amazon S3, AWS CloudFormation, Amazon Route 53, AWS Certificate Manager.
- **Domain Registration:** Ensure that *thefamilycrew.com* is registered and accessible.
- **Security:** Implement appropriate bucket policies and access controls to protect website content.
- **Scalability:** Design the infrastructure to accommodate future enhancements, such as integrating a content delivery network (CDN) or adding dynamic content.

7. Team Size

One Cloud Engineer with HTML and CSS knowledge

8. Implementation Steps

Method 1: Automated with CloudFormation Stack

1. Use the attached YAML script to create resources


AWS
CloudFormation to cr

- a. S3 bucket(Bucket Name Must Match the Domain Name) with public access enabled, and add the respective bucket policy.

- b. IAM role created with an access key for programmatic access to the S3 bucket to be used for GitHub synchronization.

CloudFormation Stack Created and executed successfully:

The screenshot shows the AWS CloudFormation console interface. On the left, a sidebar menu includes 'Stack details', 'Drifts', 'StackSets', 'Exports', 'Infrastructure Composer', 'IaC generator', 'Hooks overview', 'Hooks', 'Registry', 'Public extensions', 'Activated extensions', 'Publisher', and 'Spotlight'. The main area displays a list of stacks under 'Stacks (1)'. One stack is listed: 'Thefamilycrew' (Status: CREATE_COMPLETE). To the right, the 'Events' tab is selected for the 'Thefamilycrew' stack, showing 14 events. The first event is 'CREATE_COMPLETE' at 2025-05-16 15:57:34 UTC+0530. Subsequent events show the creation of an 'S3UserAccessKey' resource, with one entry labeled 'CREATE_IN_PROGRESS' and another labeled 'CREATE_COMPLETE'. The 'Events' table has columns for Timestamp, Logical ID, Status, Detailed status, and Status reason.

CloudFormation Outputs:

The screenshot shows the AWS CloudFormation console interface. The left sidebar is identical to the previous screenshot. The main area shows the 'Outputs' tab for the 'Thefamilycrew' stack. There are three outputs listed: 'AccessKeyId' with value 'AKIAUQ4L3FDQ37IIJH7' and description 'Access Key ID for the user'; 'BucketName' with value 'www.thefamilycrew.com' and description 'S3 Bucket Name'; and 'SecretAccessKey' with value '3Qo8oaC+cu+wUPtjfSPjQUkAdwYOsnsOjt5rPVmL' and description 'Secret Access Key for the user'. The 'Outputs' table has columns for Key, Value, Description, and Export name.

S3 Bucket Created:

2 S3 buckets created – One for the website code and other for the YAML file.

The screenshot shows the AWS S3 console with the 'General purpose buckets' tab selected. There are two buckets listed:

Name	AWS Region	IAM Access Analyzer	Creation date
cf-templates-deh83tgqw5a8-us-east-1	US East (N. Virginia) us-east-1	View analyzer for us-east-1	May 16, 2025, 15:56:26 (UTC+05:30)
www.thefamilycrew.com	US East (N. Virginia) us-east-1	View analyzer for us-east-1	May 16, 2025, 15:56:58 (UTC+05:30)

S3 bucket containing YAML:

The screenshot shows the AWS S3 console with the 'Objects' tab selected for the 'cf-templates-deh83tgqw5a8-us-east-1' bucket. There is one object listed:

Name	Type	Last modified	Size	Storage class
2025-05-16T102626.069Z13a-AWSCloudFormationCreate_StackWebsitehostedonS3.yaml	yaml	May 16, 2025, 15:56:28 (UTC+05:30)	3.6 KB	Standard

Empty S3 created, GitHub will sync the code automatically to this S3 bucket.

The screenshot shows the AWS S3 console interface. In the top navigation bar, the URL is 'Buckets > www.thefamilycrew.com'. The left sidebar has a tree view with 'Amazon S3' expanded, showing 'General purpose buckets' with 'www.thefamilycrew.com' selected. Other items in the sidebar include 'Table buckets', 'Access Grants', 'Access Points', 'Object Lambda Access Points', 'Multi-Region Access Points', 'Batch Operations', 'IAM Access Analyzer for S3', 'Block Public Access settings for this account', 'Storage Lens' (with 'Dashboards', 'Storage Lens groups', and 'AWS Organizations settings'), 'Feature spotlight' (with 11 items), and 'AWS Marketplace for S3'. At the bottom of the sidebar are links for 'CloudShell' and 'Feedback'. The main content area is titled 'www.thefamilycrew.com Info' and shows the 'Objects' tab selected. It displays a message: 'Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions.' Below this is a search bar with 'Find objects by prefix' and a table header with columns 'Name', 'Type', 'Last modified', 'Size', and 'Storage class'. A message 'No objects' is displayed, followed by 'You don't have any objects in this bucket.' and a large blue 'Upload' button. The top right of the page shows the AWS logo, a search bar, and the text 'United States (N. Virginia) Adrina Colaco'. The bottom right corner includes links for 'Privacy', 'Terms', and 'Cookie preferences'.

S3 bucket policy updated and public access being setup

The screenshot shows the AWS S3 Bucket Permissions Overview page for the bucket `www.thefamilycrew.com`. The top navigation bar includes tabs for Objects, Metadata, Properties, **Permissions**, Metrics, Management, and Access Points. The **Permissions overview** section contains a sub-section for **Access finding**, which provides information about IAM external access analyzers and links to learn more. Below this is the **Block public access (bucket settings)** section, which is currently off. It lists four options for managing public access:

- Block public access to buckets and objects granted through **new** access control lists (ACLs)
- Block public access to buckets and objects granted through **any** access control lists (ACLs)
- Block public access to buckets and objects granted through **new** public bucket or access point policies
- Block public and cross-account access to buckets and objects through **any** public bucket or access point policies

 The **Bucket policy** section displays a JSON policy document:

```
{ "Version": "2012-10-17", "Statement": [ { "Effect": "Allow", "Principal": "*", "Action": "s3:GetObject", "Resource": "arn:aws:s3:::www.thefamilycrew.com/*" } ] }
```

 Buttons for Edit, Delete, and Copy are available. The **Object Ownership** section indicates that control of objects in the bucket is from the owner's AWS account. The **Access control list (ACL)** section shows grants for the bucket owner and everyone (public access). The **Cross-origin resource sharing (CORS)** section indicates no configurations are displayed.

S3 Static Website Configurations updated:

Bucket overview

Amazon Resource Name (ARN)
arn:aws:s3:::www.thefamilycrew.com

Creation date
May 16, 2025, 15:56:58 (UTC+05:30)

Bucket Versioning

Versioning is a means of keeping multiple variants of an object in the same bucket. You can use versioning to preserve, retrieve, and restore every version of every object stored in your Amazon S3 bucket. With versioning, you can easily recover from both unintended user actions and application failures. [Learn more](#)

Bucket Versioning
Disabled

Multi-factor authentication (MFA) delete

An additional layer of security that requires multi-factor authentication for changing Bucket Versioning settings and permanently deleting object versions. To modify MFA delete settings, use the AWS CLI, AWS SDK, or the Amazon S3 REST API. [Learn more](#)

Tags (3)

You can use bucket tags to track storage costs and organize buckets. [Learn more](#)

Key	Value
awscloudformation:stack-id	amawscloudformationus-east-1:311141542113:stack/Thefamilycrew/48850be0-3240-11f0-bcdd-0affe30c0943
awscloudformation:stack-name	Thefamilycrew
awscloudformation:logical-id	thefamilycrewwebsitebucket

Default encryption

Server-side encryption is automatically applied to new objects stored in this bucket.

Encryption type [Info](#)
Server-side encryption with Amazon S3 managed keys (SSE-S3)

Bucket Key

When KMS encryption is used to encrypt new objects in this bucket, the bucket key reduces encryption costs by lowering calls to AWS KMS. [Learn more](#)

Intelligent-Tiering Archive configurations (0)

Enable objects stored in the Intelligent-Tiering storage class to tier-down to the Archive Access tier or the Deep Archive Access tier which are optimized for objects that will be rarely accessed for long periods of time. [Learn more](#)

Server access logging

Log requests for access to your bucket. Use [CloudWatch](#) to check the health of your server access logging. [Learn more](#)

Server access logging
Disabled

AWS CloudTrail data events (0) [Info](#)

Configure CloudTrail data events to log Amazon S3 object-level API operations in the CloudTrail console. [Learn more](#)

Event notifications (0)

Send a notification when specific events occur in your bucket. [Learn more](#)

Amazon EventBridge

For additional capabilities, use Amazon EventBridge to build event-driven applications at scale using S3 event notifications. [Learn more](#) or see [EventBridge pricing](#)

Transfer acceleration

Use an accelerated endpoint for faster data transfers. [Learn more](#)

Object Lock

Store objects using a write-once-read-many (WORM) model to help you prevent objects from being deleted or overwritten for a fixed amount of time or indefinitely. Object Lock works only in versioned buckets. [Learn more](#)

Requester pays

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

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](#)

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://www.thefamilycrew.com.s3-website-us-east-1.amazonaws.com>

IAM User Created and Policy assigned:

S3User [Info](#) [Delete](#)

Summary

ARN arn:aws:iam::311141542113:user/MyIAMUsers/S3User	Console access Disabled	Access key 1 AKIAUQ4L3FDQ4PM5UGPA - Active <small>Never used. Created today.</small>
Created May 16, 2025, 12:03 (UTC+05:30)	Last console sign-in -	Access key 2 Create access key

Permissions [Groups](#) [Tags](#) [Security credentials](#) [Last Accessed](#)

Permissions policies (1)

Permissions are defined by policies attached to the user directly or through groups.

Filter by Type [Search](#) [All types](#) [Attached via](#)

Policy name	Type	Attached via
S3Access	Customer inline	Inline

S3Access

```
1 ~ [{
2     "Version": "2012-10-17",
3     "Statement": [
4         {
5             "Action": [
6                 "s3:GetObject",
7                 "s3:PutObject",
8                 "s3>ListBucket"
9             ],
10            "Resource": [
11                "arn:aws:s3:::www.thefamilycrew.com",
12                "arn:aws:s3:::www.thefamilycrew.com/*"
13            ],
14            "Effect": "Allow"
15        }
16    ]
17}]
```

[Copy JSON](#) [Edit](#)

► Permissions boundary (not set)

▼ Generate policy based on CloudTrail events

You can generate a new policy based on the access activity for this user, then customize, create, and attach it to this role. AWS uses your CloudTrail events to identify the services and actions used and generate a policy. [Learn more](#)

[Generate policy](#)

No requests to generate a policy in the past 7 days.

Access Key created for IAM User:

S3User Info

Summary		Access key 1																													
ARN arn:aws:siam::311141542113:user/MyIAMUsers/S3User	Console access Disabled	AKIAUQ4L3FDQ37IIJXH7 - Active	Never used. Created today.																												
Created May 16, 2025, 12:05 (UTC+05:30)	Last console sign-in -	Access key 2 Create access key																													
Permissions Groups Tags Security credentials Last Accessed																															
Console sign-in <div style="display: flex; justify-content: space-between;"> <div> Console sign-in link https://311141542113.signin.aws.amazon.com/console </div> <div> Console password Not enabled </div> <div> Enable console access </div> </div>																															
Multi-factor authentication (MFA) (0) <div style="display: flex; justify-content: space-between;"> <div> Remove </div> <div> Resync </div> <div> Assign MFA device </div> </div> <p>Use MFA to increase the security of your AWS environment. Signing in with MFA requires an authentication code from an MFA device. Each user can have a maximum of 8 MFA devices assigned. Learn more</p> <table border="1"> <thead> <tr> <th>Type</th> <th>Identifier</th> <th>Certifications</th> <th>Created on</th> </tr> </thead> <tbody> <tr> <td colspan="4">No MFA devices. Assign an MFA device to improve the security of your AWS environment</td> </tr> <tr> <td colspan="4"> Assign MFA device </td> </tr> </tbody> </table>				Type	Identifier	Certifications	Created on	No MFA devices. Assign an MFA device to improve the security of your AWS environment				Assign MFA device																			
Type	Identifier	Certifications	Created on																												
No MFA devices. Assign an MFA device to improve the security of your AWS environment																															
Assign MFA device																															
Access keys (1) <div style="display: flex; justify-content: space-between;"> <div> Create access key </div> </div> <p>Use access keys to send programmatic calls to AWS from the AWS CLI, AWS Tools for PowerShell, AWS SDKs, or direct AWS API calls. You can have a maximum of two access keys (active or inactive) at a time. Learn more</p> <table border="1"> <thead> <tr> <th colspan="2">AKIAUQ4L3FDQ37IIJXH7</th> <th colspan="2">Actions</th> </tr> <tr> <th>Description</th> <th>Status</th> <th colspan="2"></th> </tr> </thead> <tbody> <tr> <td>-</td> <td>Active</td> <td colspan="2"> Actions </td> </tr> <tr> <td>Last used</td> <td>Created</td> <td colspan="2"> Actions </td> </tr> <tr> <td>None</td> <td>13 minutes ago</td> <td colspan="2"> Actions </td> </tr> <tr> <td>Last used region</td> <td>Last used service</td> <td colspan="2"> Actions </td> </tr> <tr> <td>N/A</td> <td>N/A</td> <td colspan="2"> Actions </td> </tr> </tbody> </table>				AKIAUQ4L3FDQ37IIJXH7		Actions		Description	Status			-	Active	Actions		Last used	Created	Actions		None	13 minutes ago	Actions		Last used region	Last used service	Actions		N/A	N/A	Actions	
AKIAUQ4L3FDQ37IIJXH7		Actions																													
Description	Status																														
-	Active	Actions																													
Last used	Created	Actions																													
None	13 minutes ago	Actions																													
Last used region	Last used service	Actions																													
N/A	N/A	Actions																													
SSH public keys for AWS CodeCommit (0) <div style="display: flex; justify-content: space-between;"> <div> Actions </div> <div> Upload SSH public key </div> </div> <p>User SSH public keys to authenticate access to AWS CodeCommit repositories. You can have a maximum of five SSH public keys (active or inactive) at a time. Learn more</p> <table border="1"> <thead> <tr> <th>SSH Key ID</th> <th>Uploaded</th> <th>Status</th> </tr> </thead> <tbody> <tr> <td colspan="3">No SSH public keys</td> </tr> <tr> <td colspan="3"> Upload SSH public key </td> </tr> </tbody> </table>				SSH Key ID	Uploaded	Status	No SSH public keys			Upload SSH public key																					
SSH Key ID	Uploaded	Status																													
No SSH public keys																															
Upload SSH public key																															
HTTPS Git credentials for AWS CodeCommit (0) <div style="display: flex; justify-content: space-between;"> <div> Actions </div> <div> Generate credentials </div> </div> <p>Generate a user name and password you can use to authenticate HTTPS connections to AWS CodeCommit repositories. You can have a maximum of 2 sets of credentials (active or inactive) at a time. Learn more</p> <table border="1"> <thead> <tr> <th>User name</th> <th>Created</th> <th>Status</th> </tr> </thead> <tbody> <tr> <td colspan="3">No credentials</td> </tr> <tr> <td colspan="3"> Generate credentials </td> </tr> </tbody> </table>				User name	Created	Status	No credentials			Generate credentials																					
User name	Created	Status																													
No credentials																															
Generate credentials																															
Credentials for Amazon Keyspaces (for Apache Cassandra) (0) <div style="display: flex; justify-content: space-between;"> <div> Actions </div> <div> Generate credentials </div> </div> <p>Generate a user name and password you can use to authenticate to Amazon Keyspaces. You can have a maximum of two sets of credentials (active or inactive) at a time. Learn more</p> <table border="1"> <thead> <tr> <th>User name</th> <th>Created</th> <th>Status</th> </tr> </thead> <tbody> <tr> <td colspan="3">No credentials</td> </tr> <tr> <td colspan="3"> Generate credentials </td> </tr> </tbody> </table>				User name	Created	Status	No credentials			Generate credentials																					
User name	Created	Status																													
No credentials																															
Generate credentials																															
X.509 Signing certificates (0) <div style="display: flex; justify-content: space-between;"> <div> Actions </div> <div> Upload </div> <div> Create X.509 certificate </div> </div> <p>Use X.509 certificates to make secure SOAP-protocol requests to some AWS services. You can have a maximum of two X.509 certificates (active or inactive) at a time. Learn more</p> <table border="1"> <thead> <tr> <th>Creation time</th> <th>Thumbprint</th> <th>Status</th> </tr> </thead> <tbody> <tr> <td colspan="3">No X.509 certificates</td> </tr> <tr> <td colspan="3"> Create X.509 certificate </td> </tr> </tbody> </table>				Creation time	Thumbprint	Status	No X.509 certificates			Create X.509 certificate																					
Creation time	Thumbprint	Status																													
No X.509 certificates																															
Create X.509 certificate																															

2. Configure GitHub repo Create with action workflow file to sync on Push.
 - a. On your GitHub Repo Click on settings.
 - b. Select Secrets from the left sidebar
 - c. Create three Secret Keys for AWS_S3_BUCKET, AWS_ACCESS_KEY_ID and AWS_SECRET_ACCESS_KEY

Actions secrets and variables

Secrets and variables allow you to manage reusable configuration data. Secrets are encrypted and are used for sensitive data. [Learn more about encrypted secrets](#). Variables are shown as plain text and are used for non-sensitive data. [Learn more about variables](#).

Anyone with collaborator access to this repository can use these secrets and variables for actions. They are not passed to workflows that are triggered by a pull request from a fork.

Repository secrets

Name	Last updated	Action
AWS_ACCESS_KEY_ID	now	Edit Delete
AWS_S3_BUCKET	now	Edit Delete
AWS_SECRET_ACCESS_KEY	now	Edit Delete

New repository secret

- d. On your GitHub repo click on Actions and a workflow file by clicking on the New Workflow Button.

e. Paste the code below in the action workflow file

```
name: uploadToS3

on:
  push:
    branches: [ "main" ]

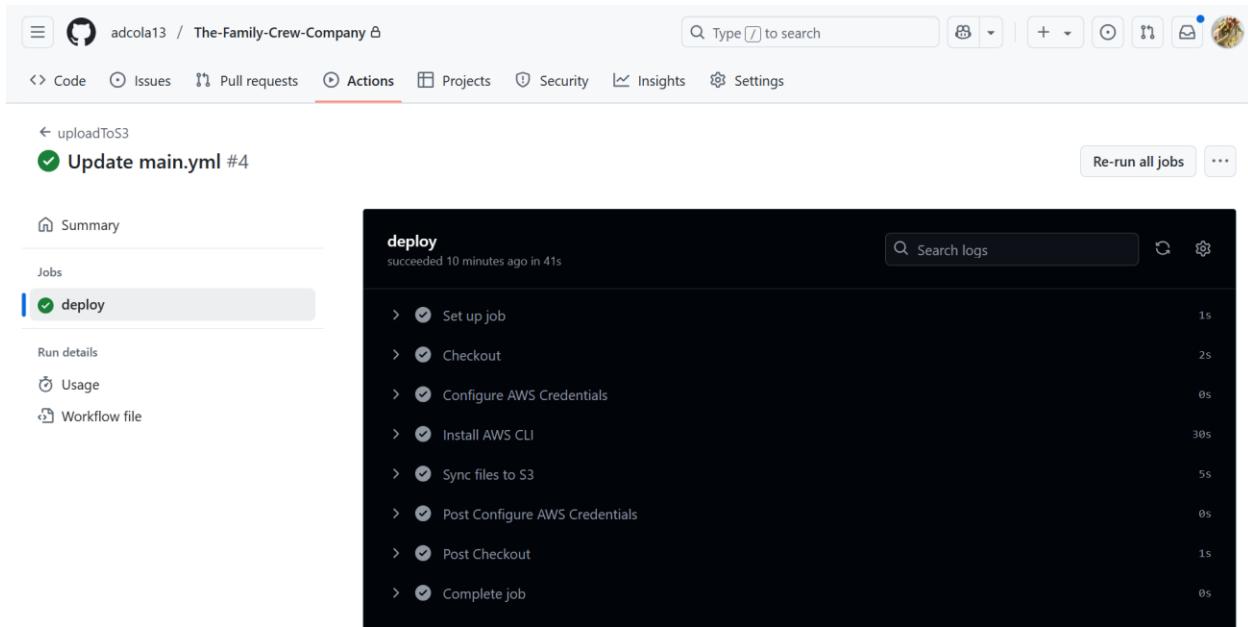
jobs:
  deploy:
    runs-on: ubuntu-latest
    steps:
      - name: Checkout
        uses: actions/checkout@v3

      - name: Configure AWS Credentials
        uses: aws-actions/configure-aws-credentials@v2
        with:
          aws-access-key-id: ${{ secrets.AWS_ACCESS_KEY_ID }}
          aws-secret-access-key: ${{ secrets.AWS_SECRET_ACCESS_KEY }}
          aws-region: 'us-east-1'

      - name: Install AWS CLI
        run:
          sudo apt-get update
          sudo snap install aws-cli --classic

      - name: Sync files to S3
        run:
          aws s3 sync ./ s3://${{ secrets.AWS_S3_BUCKET }} --delete
```

We are done. GitHub will upload the files to your S3 Bucket every time you push your update to GitHub.



The screenshot shows a GitHub Actions workflow named 'deploy' for a repository 'The-Family-Crew-Company'. The workflow has completed successfully 10 minutes ago in 41 seconds. The steps listed are:

- > Set up job (1s)
- > Checkout (25s)
- > Configure AWS Credentials (0s)
- > Install AWS CLI (30s)
- > Sync files to S3 (5s)
- > Post Configure AWS Credentials (0s)
- > Post Checkout (1s)
- > Complete job (0s)

On the left sidebar, there are links for 'Summary', 'Jobs' (which is selected), 'Run details', 'Usage', and 'Workflow file'.

The screenshot shows the GitHub Actions interface for a workflow named 'uploadToS3'. The 'Actions' tab is selected. A green checkmark indicates the workflow has passed. The 'Workflow file for this run' section displays the following YAML code:

```

1  name: uploadToS3
2
3  on:
4    push:
5      branches: [ "main" ]
6
7  jobs:
8    deploy:
9      runs-on: ubuntu-latest
10     steps:
11       - name: Checkout
12         uses: actions/checkout@v3
13
14       - name: Configure AWS Credentials
15         uses: aws-actions/configure-aws-credentials@v2
16         with:
17           aws-access-key-id: ${{ secrets.AWS_ACCESS_KEY_ID }}
18           aws-secret-access-key: ${{ secrets.AWS_SECRET_ACCESS_KEY }}
19           aws-region: 'us-east-1'
20
21       - name: Install AWS CLI
22         run: |
23           sudo apt-get update
24           sudo snap install aws-cli --classic
25
26       - name: Sync files to S3
27         run: |
28           aws s3 sync ./ s3://${{ secrets.AWS_S3_BUCKET }} --delete

```

You may Rerun if needed:

The screenshot shows the GitHub Actions interface for the same workflow. The 'Actions' tab is selected. A green checkmark indicates the workflow has passed. The 'Summary' section shows the latest run was triggered 2 minutes ago by 'adcola13' on branch 'main', with a status of 'Success' and a total duration of '37s'. The 'Workflow file' section shows the 'main.yml' file content, which includes a 'deploy' job. A tooltip for the 'Latest attempt #2' shows it succeeded on May 16 by 'adcola13'. Another tooltip for 'Attempt #1' also shows it succeeded on May 16 by 'adcola13'.

S3 bucket synced with the latest website code from GitHub:

www.thefamilycrew.com [Info](#)

Objects (10)

Actions [Copy S3 URI](#) [Copy URL](#) [Download](#) [Open](#) [Delete](#) [Actions](#) [Create folder](#) [Upload](#)

Find objects by prefix

Name	Type	Last modified	Size	Storage class
.git/	Folder	-	-	-
.github/	Folder	-	-	-
css/	Folder	-	-	-
error.html	html	May 16, 2025, 16:22:24 (UTC+05:30)	133.0 B	Standard
images/	Folder	-	-	-
index.html	html	May 16, 2025, 16:22:25 (UTC+05:30)	69.6 KB	Standard
js/	Folder	-	-	-
README.md	md	May 16, 2025, 16:22:23 (UTC+05:30)	331.0 B	Standard
style.css	css	May 16, 2025, 16:22:25 (UTC+05:30)	32.7 KB	Standard
Website_Code/	Folder	-	-	-

3. Verify the Static Website created by S3. Run the “Bucket website endpoint” URL found under Static website hosting section in a new browser window.

TheFamilyCrewCo - Bootstrap

Not secure thefamilycrew.com.s3-website-us-east-1.amazonaws.com

HOME SHOP BLOG PAGES CONTACT

WISHLIST (0) CART (0)

New Collections

Scelerisque duis aliquam qui lorem ipsum

PILLAR CANDLES

Mango Papaya

6:58 PM 5/16/2025

4. Currently, the www.thefamilycrew.com URL does not point to this new website.

It still forwards to the below URL

<https://the-family-crew-company.square.site/s/order#most-popular>

The screenshot shows a web browser window displaying a product catalog. The title bar says "The Family Crew Company". The address bar shows the URL "the-family-crew-company.square.site/s/order#most-popular". The main content area is titled "Most popular" and displays four items in a grid:

- Hawaiian Plumeria Candle** | Homemade | Natural Eco Friendly | Wedding | Bridal | Baby Shower | Valentines | Gift for men
This candle oozes out the freshness of a tropical island. Experience the fragrance of a sweet and...
CAD\$24.99
- Mango Exotic Bath Bomb** | Homemade | Mothers Day | Wedding Gift | Bridal Shower | Baby Shower | Self Love
Mango exotic bath bomb is made with premium quality essential oil. It boosts your mood. This w...
CAD\$8.00
- Lemongrass Bath Bomb** | Homemade | Mothers Day | Wedding Gift | Bridal Shower | Baby Shower | Self Love
Lemongrass is a tropical grassy plant used in cooking and herbal medicine. The place where I...
CAD\$8.00
- Pineapple** | Coconut Soy Wax Candle | Homemade | Natural Eco Friendly | Eco Home Decor | Mothers day gift| Wedding decor | Bridal
Bask into the tangy fragrance of our Pineapple candle any day. Imagine sun bathing at a pool a...
CAD\$25.00
Low stock

5. Configure Route 53 Hosted Zone: (Use Classic view)

- a. Navigate to Route 53 and select the hosted zone for your domain.
- b. Click "Create record" and choose "Simple routing".
- c. Define the record:
- d. Enter your subdomain (e.g., "www.yourdomain.com").
- e. Select "Alias to S3 website endpoint".
- f. Choose the same region as your S3 bucket.
- g. Your S3 endpoint should automatically populate.
- h. Click "Define simple record" and then "Create records".
- i. Wait for the record to propagate (this can take a few minutes).

Route 53

Hosted zones

Records (3)

Record name	Type	Routing policy	Differentiation	Alias	Value/Route traffic to	TTL (seconds)
www.thefamilycrew.com	NS	Simple	-	No	ns-1864.awsdns-41.co.uk. ns-545.awsdns-04.net. ns-319.awsdns-39.com. ns-1045.awsdns-02.org.	172800
www.thefamilycrew.com	SOA	Simple	-	No	ns-1864.awsdns-41.co.uk. a...	900

Create record

Define simple record

Record name: www.thefamilycrew.com

Record type: A - Routes traffic to an IPv4 address and some AWS resources

Value/Route traffic to: s3-website-us-east-1.amazonaws.com

Evaluate target health: Yes

The screenshot shows the AWS Route 53 'Configure records' step 2 interface. The top navigation bar includes the AWS logo, search bar, and user information (Adrina Colaco). The breadcrumb trail is 'Route 53 > Hosted zones > www.thefamilycrew.com > Create record'. On the left, a sidebar shows 'Step 1 Choose routing policy' and 'Step 2 Configure records' (selected). The main content area is titled 'Configure records' with a 'Simple routing records to add to www.thefamilycrew.com' section. It explains that you can create multiple records at once with the same routing policy. A table lists a single record: 'www.thefamilycrew.com' (Type: A) pointing to 's3-website-us-east-1.amazonaws.com'. Buttons for 'Edit', 'Delete', and 'Define simple record' are available. Below the table, a link to 'Existing records' is shown. At the bottom are 'Cancel', 'Previous', and 'Create records' buttons.

aws Search [Alt+5] Global Adrina Colaco

Route 53 > Hosted zones > www.thefamilycrew.com > Test record

Test record Info

Test records to simulate the values that Route 53 returns in response to DNS queries. This tool displays the standard values that Route 53 provides based on the settings in the hosted zone. The tool doesn't send actual DNS queries.

Record to test

Hosted zone
www.thefamilycrew.com

Record name - optional Info
To check a record that has the same name as the hosted zone www.thefamilycrew.com, leave this field blank. To check the record for a subdomain, enter the subdomain name excluding the domain name.

Record type Info
The DNS type of the record determines the format of the value that Route 53 returns in response to DNS queries.

Settings to simulate DNS queries - optional
Simulate the response that Route 53 returns to a specific IP address. This is useful for testing geolocation and latency records.

Resolver IP address Info
The IP address that the tool uses to simulate the location of the DNS resolver that a client uses to make requests. If you omit this value, the tool uses the IP address of a DNS resolver in the AWS US East (N. Virginia) Region.

▼ Additional configuration

If the resolver supports EDNS0, specify an IP address and subnet mask for the client.

EDNS0 client subnet IP Info
The client subnet IP for an IP address in the applicable location. For example, 192.0.2.0.

Subnet mask - optional Info
If you specified a value for EDNS0 client subnet IP, optionally enter the number of bits in the IP address that you want to include in the DNS requests. For example, if you specify 192.0.2.44 for EDNS0 client subnet IP and 24 for Subnet mask, this tool will simulate a request from 192.0.2.0/24. The default value is 24 bits for IPv4 addresses and 64 bits for IPv6 addresses.

Response returned by Route 53
Response from Route 53 based on the following options.

Hosted zone
www.thefamilycrew.com

Record name
www

Record type
A

DNS response code
 No Error

Protocol
UDP

Response returned by Route 53

```
16.182.32.229
52.217.136.205
3.5.12.133
52.217.97.91
54.231.170.61
3.5.12.36
16.15.176.66
16.15.193.244
```

Cancel **Get response**

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

6. Add the DNS name servers from AWS into GoDaddy or any other Hosting server.

Hosted zone details

- Hosted zone name: www.thefamilycrew.com
- Hosted zone ID: Z06784272AG2UST3TAGBE
- Description: The hosted Zone is for The family crew company website
- Record count: 3
- Name servers:
 - ns-1864.awsdns-41.co.uk
 - ns-545.awsdns-04.net
 - ns-319.awsdns-39.com
 - ns-1045.awsdns-02.org

Records (3)

Record name	Type	Routing policy	Alias	Value/Route traffic to	TTL (s)	Health
www.thefamilycr...	A	Simple	-	Yes s3-website-us-east-1.amazonaws.com	-	-
www.thefamilycr...	NS	Simple	-	No ns-1864.awsdns-41.co.uk. ns-545.awsdns-04.net. ns-319.awsdns-39.com. ns-1045.awsdns-02.org	172800	-



Nameservers determine where your DNS is hosted and where you add, edit or delete your DNS records.

Using custom
nameservers

Change Nameservers

Nameservers ?

ns-1864.awsdns-41.co.uk

ns-545.awsdns-04.net

ns-319.awsdns-39.com

7. Configure ACM- AWS Certificate manager to create SSL Certificate:

Request public certificate

Domain names
Provide one or more domain names for your certificate.

Fully qualified domain name [Info](#)

Add another name to this certificate
You can add additional names to this certificate. For example, if you're requesting a certificate for "www.example.com", you might want to add the name "example.com" so that customers can reach your site by either name.

Validation method [Info](#)
Select a method for validating domain ownership.

DNS validation - recommended
Choose this option if you are authorized to modify the DNS configuration for the domains in your certificate request.

Email validation
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.

Key algorithm [Info](#)
Select an encryption algorithm. Some algorithms may not be supported by all AWS services.

RSA 2048
RSA is the most widely used key type.

ECDSA P 256
Equivalent in cryptographic strength to RSA 3072.

ECDSA P 384
Equivalent in cryptographic strength to RSA 7680.

Tags [Info](#)
No tags associated with the resource.

[Add new tag](#)
You can add up to 50 tags.

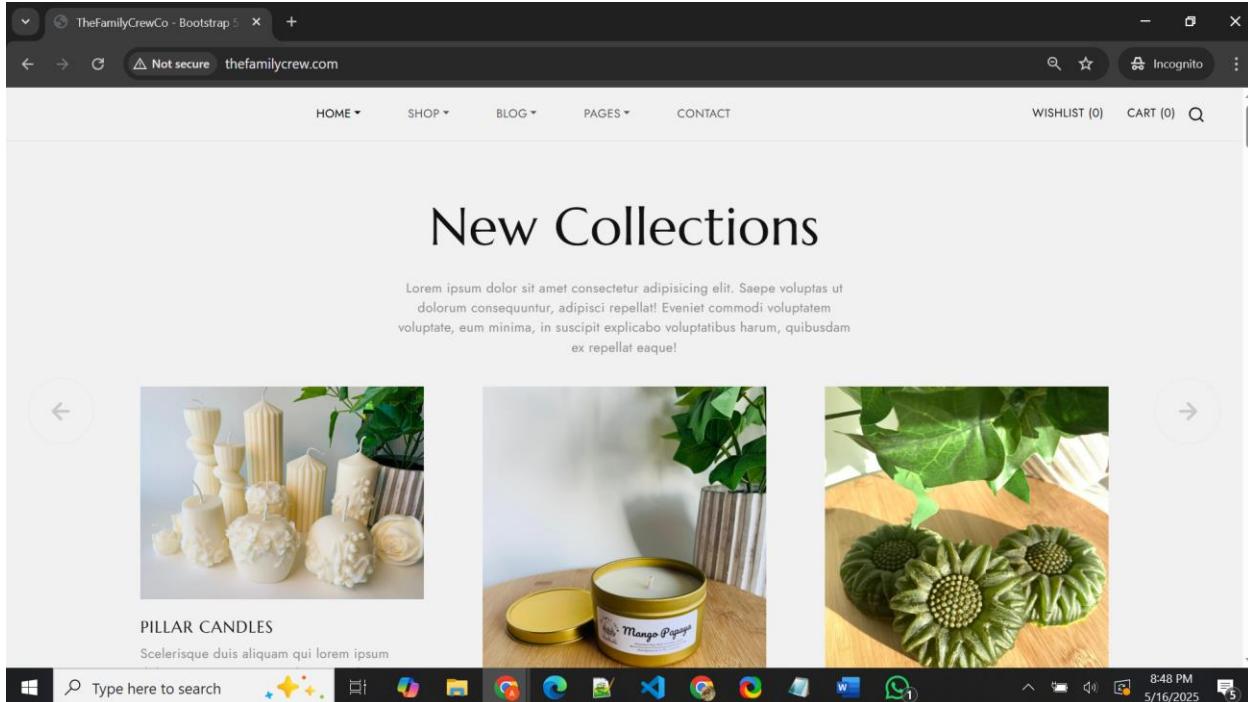
[Cancel](#) [Previous](#) [Request](#)

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

Certificates (1)

<input type="checkbox"/> Certificate ID	Domain name	Type	Status	In use	Renewal eligibility
<input type="checkbox"/> Obb4abd5-c0bc-43aa-b102-534e5cc05b16	www.thefamilycrew.com	Amazon Issued	Pending validation	No	Ineligible

8. Verify Website:
- Navigate to your domain URL in a web browser. <http://www.thefamilycrew.com>
 - Your website should now be accessible.
 - You may test the HTTPS link as well.



Method 2: Manual Creation of Records in AWS Management Console

Step 1: Create S3 bucket

To create a new bucket, you can navigate to the AWS S3 console, and click on the 'Create Bucket' button.

- Click on the properties tab of the newly created bucket and enable the static website option.
- Click on the permission tab and make it public by unchecking "Block all public access"
- On the Bucket policy section paste this policy and save.

{

"Version": "2012-10-17",

"Statement": [

```
{
    "Sid": "PublicReadGetObject",
    "Effect": "Allow",
    "Principal": "*",
    "Action": "s3:GetObject",
    "Resource": "arn:aws:s3:::bucket-name/*"
}
]
```

The screenshot shows the AWS S3 console for the bucket 'the-family-crew-website'. The 'Objects' tab is selected, displaying 0 objects. The interface includes standard S3 actions like Copy S3 URI, Copy URL, Download, Open, Delete, Actions, Create folder, and Upload. A search bar at the top allows finding objects by prefix. The bottom of the screen shows a Windows taskbar with various icons and system status.

Screenshot of the AWS S3 Bucket Properties page for 'the-family-crew-website'.

Bucket overview

- AWS Region: US East (N. Virginia) us-east-1
- Amazon Resource Name (ARN): arnaws3::the-family-crew-website
- Creation date: May 16, 2025, 12:03:43 (UTC+05:30)

Bucket Versioning

Versioning is a means of keeping multiple variants of an object in the same bucket. You can use versioning to preserve, retrieve, and restore every version of every object stored in your Amazon S3 bucket. With versioning, you can easily recover from both unintended user actions and application failures. [Learn more](#)

Bucket Versioning
Disabled

Multi-factor authentication (MFA) delete
An additional layer of security that requires multi-factor authentication for changing Bucket Versioning settings and permanently deleting object versions. To modify MFA delete settings, use the AWS CLI, AWS SDK, or the Amazon S3 REST API. [Learn more](#)

Tags (3)
You can use bucket tags to track storage costs and organize buckets. [Learn more](#)

Key	Value
awscloudformation:stack-id	arnawscloudformationus-east-1:311141542115stack/Thefamilycrew/b5091fe0-321f-11f0-8adf-0eae98b316b5
awscloudformation:stack-name	Thefamilycrew
awscloudformation:logical-id	thefamilycrewwebsitebucket

Default encryption [Info](#)
Server-side encryption is automatically applied to new objects stored in this bucket.

Encryption type [Info](#)
Server-side encryption with Amazon S3 managed keys (SSE-S3)

Bucket Key
When KMS encryption is used to encrypt new objects in this bucket, the bucket key reduces encryption costs by lowering calls to AWS KMS. [Learn more](#)

Disabled

Intelligent-Tiering Archive configurations (0)
Enable objects stored in the Intelligent-Tiering storage class to tier-down to the Archive Access tier or the Deep Archive Access tier which are optimized for objects that will be rarely accessed for long periods of time. [Learn more](#)

Find Intelligent-Tiering Archive configurations

Name	Status	Scope	Days until transition to Archive Access ...	Days until transition to Deep Archive A...
No archive configurations No configurations to display.				

Create configuration

Server access logging
Log requests for access to your bucket. Use [CloudWatch](#) to check the health of your server access logging. [Learn more](#)

Server access logging
Disabled

AWS CloudTrail data events (0) [Info](#)
Configure CloudTrail data events to log Amazon S3 object-level API operations in the CloudTrail console. [Learn more](#)

Name

Name	Access
No data events No data events to display.	

Configure in CloudTrail

Create event notification

Event notifications (0)
Send a notification when specific events occur in your bucket. [Learn more](#)

Name

Name	Event types	Filters	Destination type	Destination
No event notifications Choose Create event notification to be notified when a specific event occurs.				

Create event notification

Amazon EventBridge
For additional capabilities, use Amazon EventBridge to build event-driven applications at scale using S3 event notifications. [Learn more](#) or see [EventBridge pricing](#)

Send notifications to Amazon EventBridge for all events in this bucket
Off

Transfer acceleration
Use an accelerated endpoint for faster data transfers. [Learn more](#)

Transfer acceleration
Disabled

Object Lock
Store objects using a write-once-read-many (WORM) model to help you prevent objects from being deleted or overwritten for a fixed amount of time or indefinitely. Object Lock works only in versioned buckets. [Learn more](#)

Object Lock
Disabled

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](#)

Create Amplify app

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://the-family-crew-website.s3-website-us-east-1.amazonaws.com>

Manually, Upload all the website resources into the S3 bucket.

Name	Type	Last modified	Size	Storage class
.git/	Folder	-	-	-
.github/	Folder	-	-	-
css/	Folder	-	-	-
error.html	html	May 16, 2025, 12:51:09 (UTC+05:30)	133.0 B	Standard
images/	Folder	-	-	-
index.html	html	May 16, 2025, 12:51:10 (UTC+05:30)	69.6 KB	Standard
js/	Folder	-	-	-
README.md	md	May 16, 2025, 12:51:09 (UTC+05:30)	331.0 B	Standard
style.css	css	May 16, 2025, 12:51:10 (UTC+05:30)	32.7 KB	Standard
Website_Code/	Folder	-	-	-

Step 2: Create an IAM user and generate an Access Key

- From AWS IAM create a new user
- In the Set Permission step select "Attach policies directly"
- From the list of Permission policies search and find "AmazonS3FullAccess" and select it and hit the Create user button.
- Click Create access keys for the user and download the access key CSV file or copy and save it somewhere.

IAM Role Created and respective policy assigned

S3User [Info](#) [Delete](#)

Summary

ARN arn:aws:iam::311141542113:user/MyIAMUsers/S3User	Console access Disabled	Access key 1 AKIAUQ4L3FDQ4PM5UGPA - Active <small>Never used. Created today.</small>
Created May 16, 2025, 12:03 (UTC+05:30)	Last console sign-in -	Access key 2 Create access key

Permissions [Groups](#) [Tags](#) [Security credentials](#) [Last Accessed](#)

Permissions policies (1)

Permissions are defined by policies attached to the user directly or through groups.

Filter by Type

Policy name	Type	Attached via
S3Access	Customer inline	Inline

S3Access

```
1 * [{  
2     "Version": "2012-10-17",  
3     "Statement": [  
4         {  
5             "Action": [  
6                 "s3:GetObject",  
7                 "s3:PutObject",  
8                 "s3>ListBucket"  
9             ],  
10            "Resource": [  
11                "arn:aws:s3:::the-family-crew-website",  
12                "arn:aws:s3:::the-family-crew-website/*"  
13            ],  
14            "Effect": "Allow"  
15        }  
16    ]  
17}]
```

[Copy JSON](#) [Edit](#)

► Permissions boundary (not set)

▼ Generate policy based on CloudTrail events

You can generate a new policy based on the access activity for this user, then customize, create, and attach it to this role. AWS uses your CloudTrail events to identify the services and actions used and generate a policy. [Learn more](#)

[Generate policy](#)

No requests to generate a policy in the past 7 days.

IAM Access Key updated in the role.

S3User [Info](#) [Delete](#)

Summary

ARN arn:aws:iam::311141542113:user/MyIAMUsers/S3User	Console access Disabled	Access key 1 AKIAUQ4L3FDQ4PM5UGPA - Active Never used. Created today.
Created May 16, 2025, 12:03 (UTC+05:30)	Last console sign-in -	Access key 2 Create access key

Permissions | Groups | Tags | **Security credentials** | Last Accessed

Console sign-in

Console sign-in link https://311141542113.signin.aws.amazon.com/console	Console password Not enabled	Enable console access
---	---------------------------------	---------------------------------------

Multi-factor authentication (MFA) (0)

Use MFA to increase the security of your AWS environment. Signing in with MFA requires an authentication code from an MFA device. Each user can have a maximum of 8 MFA devices assigned. [Learn more](#)

Type	Identifier	Certifications	Created on
No MFA devices. Assign an MFA device to improve the security of your AWS environment			
Assign MFA device			

Access keys (1)

Use access keys to send programmatic calls to AWS from the AWS CLI, AWS Tools for PowerShell, AWS SDKs, or direct AWS API calls. You can have a maximum of two access keys (active or inactive) at a time. [Learn more](#)

AKIAUQ4L3FDQ4PM5UGPA	Description -	Status Active	Actions ▾ Create access key
Last used 7 minutes ago	Created 54 minutes ago	Last used service s3	

SSH public keys for AWS CodeCommit (0)

User SSH public keys to authenticate access to AWS CodeCommit repositories. You can have a maximum of five SSH public keys (active or inactive) at a time. [Learn more](#)

SSH Key ID	Uploaded	Status
No SSH public keys		
Upload SSH public key		

HTTPS Git credentials for AWS CodeCommit (0)

Generate a user name and password you can use to authenticate HTTPS connections to AWS CodeCommit repositories. You can have a maximum of 2 sets of credentials (active or inactive) at a time. [Learn more](#)

User name	Created	Status
No credentials		
Generate credentials		

Credentials for Amazon Keyspaces (for Apache Cassandra) (0)

Generate a user name and password you can use to authenticate to Amazon Keyspaces. You can have a maximum of two sets of credentials (active or inactive) at a time. [Learn more](#)

User name	Created	Status
No credentials		
Generate credentials		

X.509 Signing certificates (0)

Use X.509 certificates to make secure SOAP-protocol requests to some AWS services. You can have a maximum of two X.509 certificates (active or inactive) at a time. [Learn more](#)

Creation time	Thumbprint	Status
No X.509 certificates		
Create X.509 certificate		

Perform the remaining steps from Step 3-8 from Method 1.

9. Health Checks & Query Logging

- Configure Health Checks in Route53 to monitor the health of your domain

The screenshot shows the 'Create health check' configuration page. In the 'Name - optional' field, 'Primary-Website-Health-TheFamilyCrew' is entered. Under 'Specify endpoint by', 'Domain name' is selected, and 'HTTP' is chosen from the dropdown. The path is set to 'www.thefamilycrew.com'. A note indicates that the path can return an HTTP status code of 2xx or 3xx when the endpoint is healthy. There is an 'Advanced configuration' link. In the 'Tags - optional' section, there is an 'Add new tags' button. At the bottom right, there are 'Cancel' and 'Create health check' buttons.

The screenshot shows the 'Health checks (1)' list page. It displays one health check entry: 'Primary-Website-Health-TheFamilyCrew' with the URL 'http://www.thefamilycrew.com:80/'. The status is listed as 'Healthy'. The left sidebar shows navigation links for Route 53: Dashboard, Hosted zones, Health checks (selected), Profiles, IP-based routing, and Traffic flow.

- Enable “Query Logging” for CloudWatch logs.

Configure query logging Info

Query logging configuration name

Name
A friendly name lets you find a Resolver query logging configuration in the dashboard.
 The name can have up to 64 characters. Valid characters: A-Z, a-z, 0-9, space, _ (underscore) and - (hyphen)

Query logs destination Info
Resolver can save logs in CloudWatch Logs, in an S3 bucket, or in Kinesis Data Streams.

Destination for query logs
Choose where you want Resolver to publish query logs. Standard storage charges apply.

CloudWatch Logs log group
You can analyze logs with Log Insights and create metrics and alarms.

S3 bucket
An S3 bucket is economical for long-term log archiving. Latency is typically higher.

Amazon Data Firehose delivery stream
You can stream logs in real time to Elasticsearch, Redshift, or other applications.

CloudWatch Logs log groups
You can either choose a CloudWatch Logs log group that was created by the current account, or choose to create a log group for this query logging configuration.

[Create log group](#)

New log group name

VPCs to log queries for - optional (0) Info
Resolver logs DNS queries that originate in the VPCs that you choose here. If you don't choose any VPCs, Resolver doesn't log any queries.

[Remove](#) [Add VPC](#)

<input type="button" value="Find resource"/>	VPC ID	VPC name	Status	IPv4 CIDR	IPv6 CIDR	Owner
No resources You don't have any resources.						
Add VPC						

Tags - optional Info
No tags associated with the resource.

[Add tag](#)

[Cancel](#) [Configure query logging](#)

Route 53

- Dashboard
- Hosted zones
- Health checks
- Profiles New
- IP-based routing
- Traffic flow
- Domains
- Resolver
- Query logging

[cloudShell](#) [Feedback](#)

© 2025, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

Route 53 > Resolver > Query logging

You successfully created the log group /aws/Route53/thefamilycrew.com.

You successfully created the query logging configuration thefamilycrewlogs.

You are signed in to the following Region: us-east-1 (N. Virginia)
To change your Region, use the Region selector in the upper-right corner.

Query logging configurations (1) Info

<input type="button" value="Search"/>	Name	ID	Status	Destination type	Destination ARN	VPC count	Sharing status	Creation time (UTC)
<input type="radio"/>	thefamilycrewlogs	rqlc-4594521917904235	Created	CloudWatch Logs	arn:aws:logs:us-east-1:311141542113:log-group:/aws/Route53/thefamilycrew.com	0	Not shared	2025-05-17T04:45:07.267Z 3868Z

[View details](#) [Delete](#) [Configure query logging](#)

[cloudShell](#) [Feedback](#)

© 2025, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

10. Future Considerations

- **Content Delivery Network (CDN):** Integrate Amazon CloudFront to enhance website performance and security.
- **Continuous Deployment Enhancements:** Implement more advanced CI/CD pipelines using AWS CodePipeline or other tools for streamlined updates and maintenance.
- **Analytics Integration:** Incorporate tools like Amazon CloudWatch or third-party services to monitor website traffic and user behavior.

11. References

GitHub: <https://github.com/adcola13>

<https://docs.aws.amazon.com/AmazonS3/latest/userguide/website-hosting-custom-domain-walkthrough.html>

<https://docs.aws.amazon.com/AmazonCloudFront/latest/DeveloperGuide/getting-started-secure-static-website-cloudformation-template.html>