

AWS S3 Static Website Hosting Project Report

Submitted by: Chidapha Chaitharin
Student ID: 64692

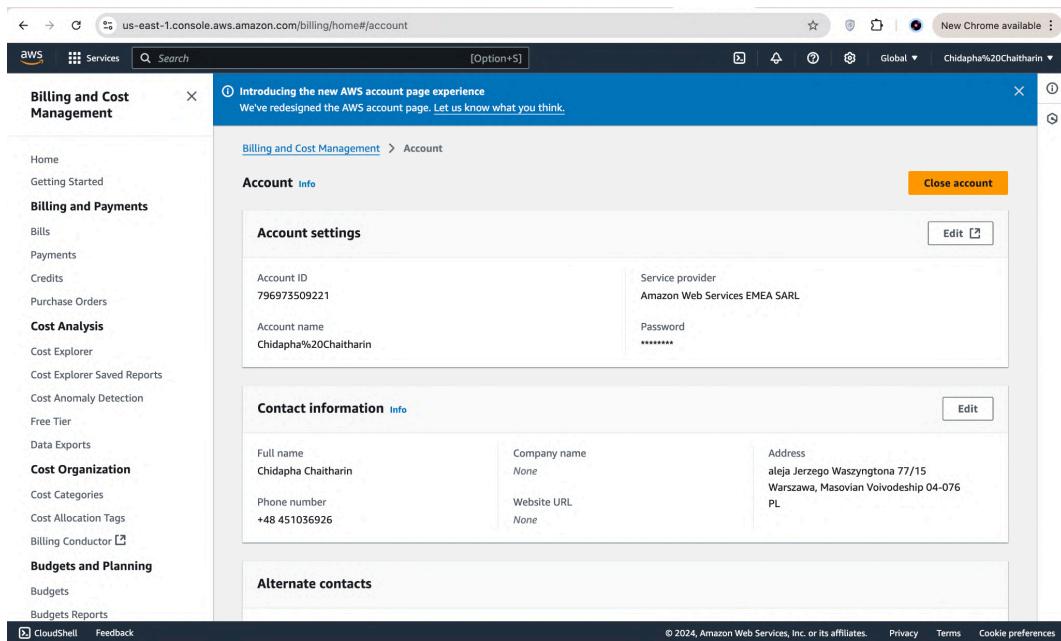
Introduction

This project introduces students to Amazon Web Services (AWS) by guiding them through the deployment of a static website using AWS S3 (Simple Storage Service). Students will gain hands-on experience with S3, learning to configure it for static website hosting, manage files, and enable features like logging and versioning. The project also explores advanced AWS options such as CORS and CloudFront, enhancing practical skills in cloud computing and resource management.

Step-by-Step Documentation

Step 1: Create an AWS Account

- Action Taken:** Created a new AWS account using my student email and enabled Free Tier access.



Step 2: Navigate to AWS S3

- **Action Taken:** Logged into the AWS Management Console and navigated to the S3 service.

The screenshot shows the Amazon S3 landing page. On the left, there's a sidebar with links like 'Buckets', 'Access Grants', 'Access Points', etc. The main content area features the heading 'Amazon S3' and the sub-headline 'Store and retrieve any amount of data from anywhere'. It includes a brief description of S3 as an object storage service and a large 'Create a bucket' button. To the right, there are sections for 'Pricing' (mentioning no minimum fees) and 'Resources' (with a link to 'View pricing details'). At the bottom, there's a video thumbnail titled 'Introduction to Amazon S3' and a 'CloudShell' button.

Step 3: Create a New S3 Bucket

- **Action Taken:** Created a new S3 bucket with a globally unique name.
 - Bucket Name: aws-s3-project-chidapha
 - Region: us-east-1

The screenshot shows the 'Buckets' page under the 'General purpose buckets' tab. It displays a single bucket named 'aws-s3-project-chidapha' located in 'US East (N. Virginia) us-east-1'. The page includes a search bar, filter buttons for 'General purpose buckets' and 'Directory buckets', and buttons for 'Copy ARN', 'Empty', 'Delete', and 'Create bucket'. There's also a 'View Storage Lens dashboard' button and a note about account snapshots.

- **Configuration:**
 - Selected the Properties tab and chose Static website hosting.

- Set the index document to index.html and the error document to error.html.

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

Static website hosting

Disable
 Enable

Hosting type

Host a static website
Use the bucket endpoint as the web address. [Learn more](#)

Redirect requests for an object
Redirect requests to another bucket or domain. [Learn more](#)

Index document
Specify the home or default page of the website.
`index.html`

Error document - optional
This is returned when an error occurs.
`error.html`

For your customers to access content at the website endpoint, you must make all your content publicly readable. To do so, you can edit the S3 Block Public Access settings for the bucket. For more information, see [Using Amazon S3 Block Public Access](#).

Step 4: Upload Website Files

- **Action Taken:** Created a simple static website (index.html, styles.css, script.js, error.html) using Visual Studio Code, and uploaded the files and other assets to the S3 bucket.

Upload succeeded
View details below.

The information below will no longer be available after you navigate away from this page.

Summary		
Destination	Succeeded	Failed
<code>s3://aws-s3-project-chidapha</code>	<input checked="" type="radio"/> 5 files, 58.2 KB (100.00%)	<input type="radio"/> 0 files, 0 B (0%)

Files and folders | Configuration

Files and folders (5 Total, 58.2 KB)

Files and folders (5 Total, 58.2 KB)						
<input type="text"/> Find by name						
Name	Folder	Type	Size	Status	Error	
error.html	-	text/html	564.0 B	<input checked="" type="radio"/> Succeeded	-	
index.html	-	text/html	2.8 KB	<input checked="" type="radio"/> Succeeded	-	
script.js	-	text/javascript	190.0 B	<input checked="" type="radio"/> Succeeded	-	
style.css	-	text/css	3.5 KB	<input checked="" type="radio"/> Succeeded	-	
hero-image.j...	images/	image/jpeg	51.1 KB	<input checked="" type="radio"/> Succeeded	-	

- **Configuration:** Select the uploaded files, and in the Actions menu, choose Make public.

Screenshot of the AWS S3 console showing the contents of the 'aws-s3-project-chidapha' bucket.

The 'Objects' tab is selected, displaying 5 items:

Name	Type	Last modified	Size	Storage class
error.html	html	November 2, 2024, 02:09:01 (UTC+01:00)	564.0 B	Standard
images/	Folder	-	-	-
index.html	html	November 2, 2024, 02:09:01 (UTC+01:00)	2.8 KB	Standard
script.js	js	November 2, 2024, 02:09:01 (UTC+01:00)	190.0 B	Standard

A context menu is open over the first four objects (excluding the folder). The 'Actions' dropdown shows several options:

- Download as
- Share with a presigned URL
- Calculate total size
- Copy
- Move
- Initiate restore
- Query with S3 Select
- Edit actions
- Rename object
- Edit storage class
- Edit server-side encryption
- Edit metadata
- Edit tags
- Make public using ACL

At the bottom of the page, there are links for CloudShell, Feedback, and Copyright information.

Screenshot of the AWS S3 console showing the contents of the 'images/' folder in the 'aws-s3-project-chidapha' bucket.

The 'Objects' tab is selected, displaying 1 item:

Name	Type	Last modified	Size
hero-image.jpg	jpg	November 2, 2024, 02:09:02 (UTC+01:00)	3.5 KB

A context menu is open over the single object. The 'Actions' dropdown shows several options:

- Download as
- Share with a presigned URL
- Calculate total size
- Copy
- Move
- Initiate restore
- Query with S3 Select
- Edit actions
- Rename object
- Edit storage class
- Edit server-side encryption
- Edit metadata
- Edit tags
- Make public using ACL

At the bottom of the page, there are links for CloudShell, Feedback, and Copyright information.

The screenshot shows the AWS S3 console with a green success banner at the top stating "Successfully edited public access". Below it, a modal window titled "Make public: status" displays a summary table with one row: "Source" (s3://aws-s3-project-chidapha), "Status" (Successfully edited public access), and "Objects" (4 objects, 7.1 KB). A link below the table says "The information below will no longer be available after you navigate away from this page." The "Failed to edit public access" tab is selected, showing a table with zero rows. The bottom of the screen includes standard AWS navigation links like CloudShell, Feedback, and cookie preferences.

Step 5: Configure Static Website Hosting

- Configuration:** In the Static website hosting section, note the provided endpoint URL.

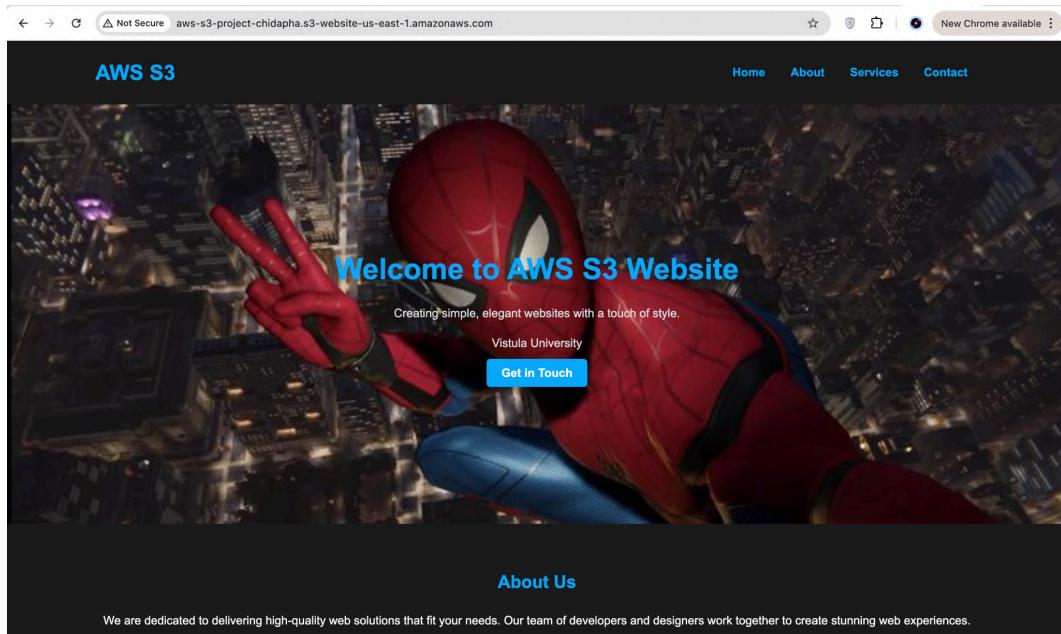
The screenshot shows the AWS S3 bucket properties page for "aws-s3-project-chidapha". Under the "Static website hosting" section, it is set to "Enabled" and "Bucket hosting". A note recommends using AWS Amplify Hosting, with a "Create Amplify app" button. The "Bucket website endpoint" field shows the URL <http://aws-s3-project-chidapha.s3-website-us-east-1.amazonaws.com>. The bottom of the screen includes standard AWS navigation links like CloudShell, Feedback, and cookie preferences.

Step 6: Test the Website

- Action Taken:** Opened the provided endpoint URL in a web browser to verify that the website is live.

- **Endpoint URL:**

<http://aws-s3-project-chidapha.s3-website-us-east-1.amazonaws.com/>



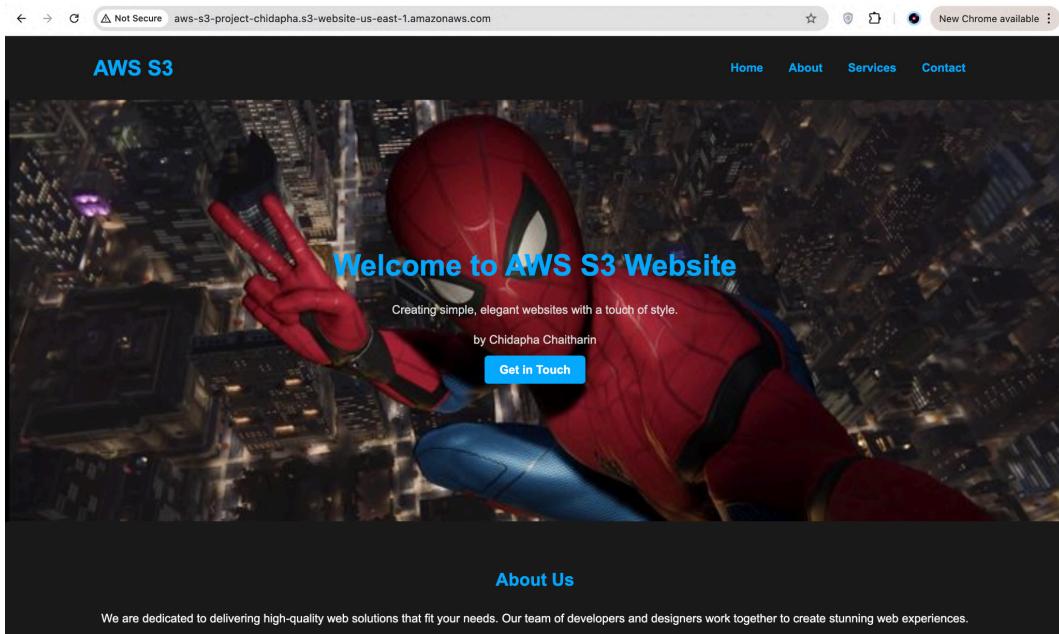
Step 7: Enable Versioning

- **Action Taken:**

- Enabled Versioning in the bucket's Properties tab.

The screenshot shows the AWS S3 Bucket Properties page for 'aws-s3-project-chidapha'. At the top, there is a green success message: 'Successfully edited Bucket Versioning. To transition, archive, or delete older object versions, configure lifecycle rules for this bucket.' Below this, the 'Properties' tab is selected in the navigation bar. The 'Bucket overview' section displays basic information: AWS Region (US East (N. Virginia) us-east-1), Amazon Resource Name (ARN) (arn:aws:s3:::aws-s3-project-chidapha), and Creation date (November 2, 2024, 01:57:42 (UTC+01:00)). The 'Bucket Versioning' section shows that versioning is enabled. The 'Multi-factor authentication (MFA) delete' section indicates that MFA delete is disabled. The bottom of the page includes standard AWS footer links: CloudShell, Feedback, © 2024, Amazon Web Services, Inc. or its affiliates., Privacy, Terms, and Cookie preferences.

- Re-uploaded a modified index.html to observe version control.



Summary					
Destination	Succeeded	Failed			
s3://aws-s3-project-chidapha	1 file, 2.8 KB (100.00%)	0 files, 0 B (0%)			

Files and folders (1 Total, 2.8 KB)

Name	Folder	Type	Size	Status	Error
index.html	-	text/html	2.8 KB	Succeeded	-

- **Configuration:**
 - Observed version history displayed in the Objects tab.
 - Checked that older versions were retained in the bucket.

The screenshot shows the AWS S3 console interface. At the top, there's a navigation bar with 'Amazon S3 > Buckets > aws-s3-project-chidapha'. Below it is a header for 'aws-s3-project-chidapha' with tabs for 'Objects', 'Properties', 'Permissions', 'Metrics', 'Management', and 'Access Points'. The 'Objects' tab is selected. A toolbar below the header includes buttons for 'Copy S3 URI', 'Copy URL', 'Download', 'Open', 'Delete', 'Actions', 'Create folder', and 'Upload'. A search bar and a 'Show versions' checkbox are also present. The main area displays a table of objects:

Name	Type	Version ID	Last modified	Size	Storage class
mWHJ2Et2H 5r3aRve28.hE w2sNW2lgbGE	html		November 2, 2024, 02:09:01 (UTC+01:00)	564.0 B	Standard
-	Folder		-	-	-
f4bQTySm2G SKlg9zOEtr WOETDT.d3xhi	html		November 2, 2024, 02:38:43 (UTC+01:00)	2.8 KB	Standard
TZjtp2XhZF 9Txp0Qroy kLPXGUdYn	html		November 2, 2024, 02:09:01 (UTC+01:00)	2.8 KB	Standard
BmleFV040B dkz5Cz26sC	html		November 2, 2024, 02:09:01	100.0 B	Standard

At the bottom of the page, there are links for 'CloudShell', 'Feedback', and copyright information: '© 2024, Amazon Web Services, Inc. or its affiliates.' followed by 'Privacy', 'Terms', and 'Cookie preferences'.

Step 8: Set Up Logging

- Action Taken:** In the S3 bucket properties, enabled Server Access Logging and created a separate S3 bucket to store logs.
 - Logging Bucket Name: logging-bucket-chidapha.

The screenshot shows the 'Properties' tab of the AWS S3 console for the 'aws-s3-project-chidapha' bucket. A green success message box at the top says 'Successfully edited server access logging.' Below it, under 'BUCK8t Key', it says 'Enabled'. The main area shows 'Intelligent-Tiering Archive configurations (0)' and 'Server access logging'. Under 'Server access logging', it says 'Log requests for access to your bucket. Use CloudWatch to check the health of your server access logging.' It shows a configuration for 'Destination bucket' set to 's3://logging-bucket-chidapha'. There are buttons for 'Edit' and 'Create configuration'.

- Reviewed and interpreted the logs in the logging bucket.

Amazon S3

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

Dashboards

Storage Lens groups

AWS Organizations settings

Feature spotlight

AWS Marketplace for S3

Objects (18) Info

Name	Type	Last modified	Size	Storage class
2024-11-02-02-16-34-14E51BA3FD97860D	-	November 2, 2024, 03:16:35 (UTC+01:00)	652.0 B	Standard
2024-11-02-02-18-58-198BD858D58A71DC	-	November 2, 2024, 03:18:59 (UTC+01:00)	1.2 KB	Standard
2024-11-02-02-20-37-D8BAC44D2C7C9DBB	-	November 2, 2024, 03:20:38 (UTC+01:00)	1.3 KB	Standard
2024-11-02-02-20-56-FE3CG6F38A5E6C02	-	November 2, 2024, 03:20:57 (UTC+01:00)	1.2 KB	Standard
2024-11-02-02-22-00-85245DEC289C0F77	-	November 2, 2024, 03:22:01 (UTC+01:00)	3.0 KB	Standard

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

```
f9838ecf6391cd6e3816969fb3c82c707daed3c52457d060edbc057bd33aa aws-s3-project-chidapha [02/Nov/2024:01:45:38 +0000] 94.172.62.222 "GET /aws-s3-project-chidapha?notification HTTP/1.1" 200 - 115 - 19 - "https://us-east-1.console.aws.amazon.com/" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/129.0.0.0 Safari/157.36" - 414C+0xRbbr=<LHIC4j>/VX5Tm+hSkfGUmc1lnTRQo0/f'sCeemsInWoCpGH+arzsXfejgIrItyLvgAnl7n+zRyzaTehmedhirkHTQ7ArM= SigV4 TLS_AES_128_GCM_SHA256 AuthHeader s3.amazonaws.com TLSv1.3 - -
```

- **Key Components:**

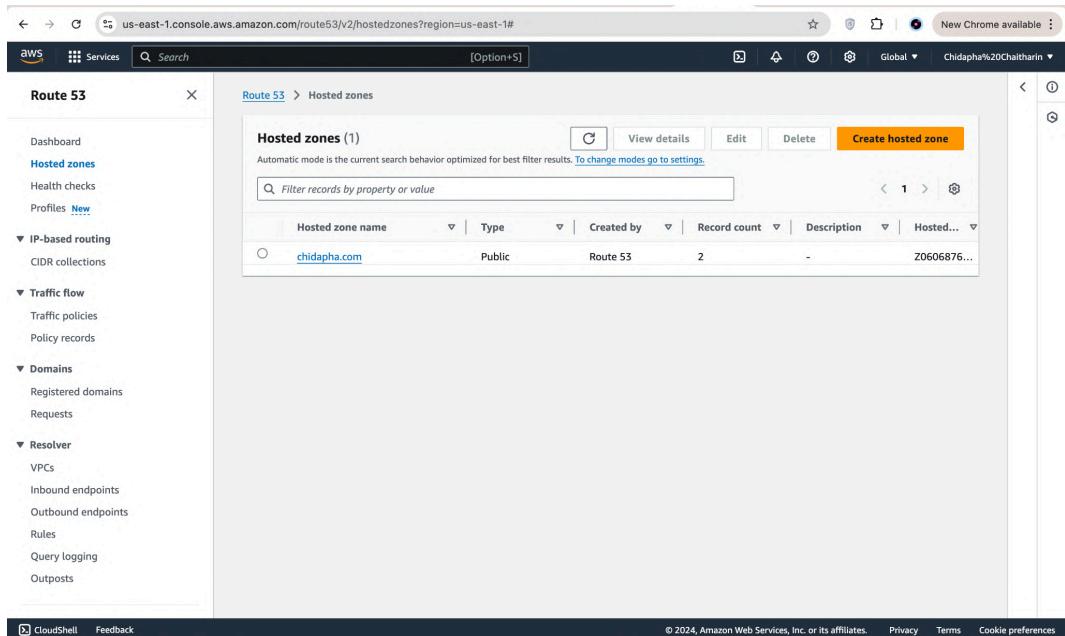
- **Bucket Name:** aws-s3-project-chidapha
- **Request Time:** [02/Nov/2024:01:45:38 +0000]
- **IP Address:** 94.172.62.222
- **Request Type:** REST.GET.NOTIFICATION
- **HTTP Method:** "GET /aws-s3-project-chidapha?notification HTTP/1.1"
- **Status Code:** 200 (success)
- **Bytes Sent:** 115 (size of the requested object)
- **User-Agent:** "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7)... " (browser info)
- **Encryption:** TLSv1.3 (connection security)

- **Summary:** This log shows a successful GET request to the S3 bucket from a specific IP, with a 200 status confirming correct processing. It includes browser and security details, useful for traffic analysis and troubleshooting.

Step 9: Add Custom Domain

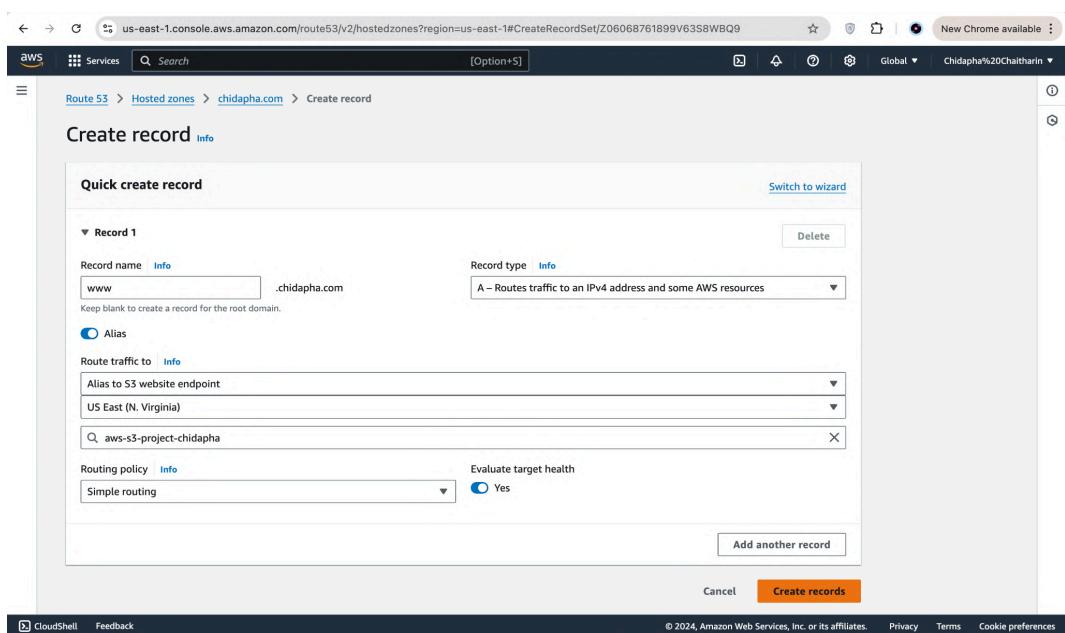
Note: I currently do not have a registered domain, but I simulated the process of configuring a custom domain using AWS Route 53.

- **Action Taken:**
 - Explored AWS Route 53 to create a hosted zone.



The screenshot shows the AWS Route 53 service dashboard. On the left, there's a navigation sidebar with options like Dashboard, Hosted zones (which is selected), Health checks, Profiles, IP-based routing, Traffic flow, Domains, Resolver, VPCs, Inbound endpoints, Outbound endpoints, Rules, Query logging, and Outposts. The main content area is titled "Route 53 > Hosted zones" and shows a table of "Hosted zones (1)". The table has columns for Hosted zone name, Type, Created by, Record count, Description, and Hosted zone ID. A single row is listed: "chidapha.com" (Type: Public, Created by: Route 53, Record count: 2, Hosted zone ID: Z0606876...). At the top right of the table, there are buttons for "Create hosted zone", "View details", "Edit", and "Delete". Below the table is a search bar labeled "Filter records by property or value". The bottom of the page includes standard AWS footer links for CloudShell, Feedback, Privacy, Terms, and Cookie preferences.

- Updated DNS records to point to the S3 bucket.



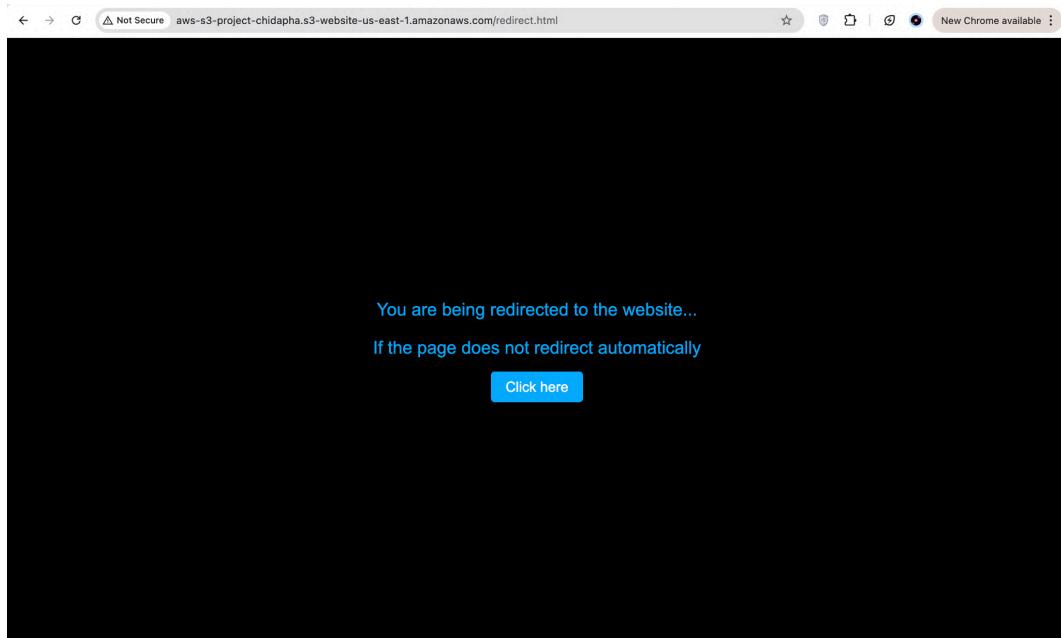
The screenshot shows the "Create record" wizard for the "chidapha.com" hosted zone. The title bar says "Route 53 > Hosted zones > chidapha.com > Create record". The main form is titled "Quick create record" and contains the following fields:

- Record name:** www.chidapha.com
- Record type:** A – Routes traffic to an IPv4 address and some AWS resources
- Alias:** Selected (radio button)
- Route traffic to:**
 - Alias to S3 website endpoint
 - US East (N. Virginia)
 - aws-s3-project-chidapha
- Routing policy:** Simple routing
- Evaluate target health:** Yes (radio button)

 At the bottom of the form are "Add another record" and "Create records" buttons. The footer includes CloudShell, Feedback, Privacy, Terms, and Cookie preferences links.

Step 10: Implement Redirects

- **Action Taken:** Created a new HTML file (redirect.html) with a meta refresh tag to automatically redirect users to another page.



- **Configuration:** Uploaded this file to the S3 bucket and configured a redirect rule in the S3 properties.

A screenshot of the AWS S3 console. The top navigation bar shows the URL "us-east-1.console.aws.amazon.com/s3/upload/aws-s3-project-chidapha?region=us-east-1&bucketType=general". A green success message box is visible, stating "Upload succeeded" and "View details below.". Below this, a modal window titled "Upload: status" is open. It contains a summary table:

Destination	Succeeded	Failed
s3://aws-s3-project-chidapha	1 file, 371.0 B (100.00%)	0 files, 0 B (0%)

Below the summary, there are tabs for "Files and folders" and "Configuration". Under "Files and folders", a table lists one item: "redirect.html" (1 Total, 371.0 B). The table includes columns for Name, Folder, Type, Size, Status, and Error. The status for "redirect.html" is listed as "Succeeded". At the bottom of the page, there are links for "CloudShell", "Feedback", "© 2024, Amazon Web Services, Inc. or its affiliates.", "Privacy", "Terms", and "Cookie preferences".

```

1 {
2   "Condition": {
3     "KeyPrefixEquals": "index.html"
4   },
5   "Redirect": {
6     "ReplaceKeyWith": "redirect.html"
7   }
8 }
9
10

```

Redirection rules – optional
Redirection rules, written in JSON, automatically redirect webpage requests for specific content. [Learn more](#)

Cancel **Save changes**

- **Tested the redirect by opening:**
<http://aws-s3-project-chidapha.s3-website-us-east-1.amazonaws.com/index.html>

Step 11: Explore S3 Object Metadata

- **Action Taken:** Added metadata to S3 objects.

Key	Value
No tags associated with this resource.	

Type	Key	Value
System defined	Content-Language	English
System defined	Content-Type	application/pdf

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](#)

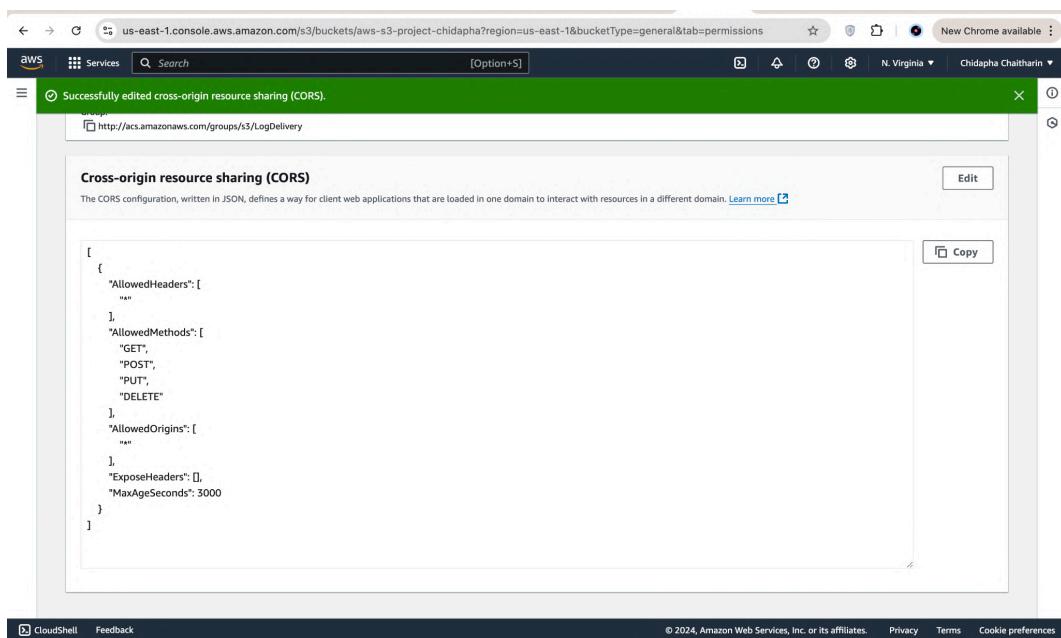
Object Lock
Disabled

Amazon S3 currently does not support enabling Object Lock after a bucket has been created. To enable Object Lock for this bucket, contact [Customer Support](#).

- **File Organization:**
 - **Categorization:** Adding metadata, such as Category or Content-Language, allows for easy filtering and searching of objects in S3, enhancing file management.

Step 12: Enable Cross-Origin Resource Sharing (CORS)

- **Action Taken:** Configured CORS to allow or restrict web applications hosted on different domains.



- **CORS Configuration:**
 - **AllowedHeaders: ["*"]**
Allows all headers in requests to the S3 bucket.
 - **AllowedMethods: ["GET", "POST", "PUT", "DELETE"]**
Permits the specified HTTP methods for requests to the bucket.
 - **AllowedOrigins: ["*"]**
Allows requests from any domain. This is broad access and can be restricted for security.
 - **ExposeHeaders: []**
No additional response headers are exposed to the client.
 - **MaxAgeSeconds: 3000**
Specifies the caching duration for preflight request results (in seconds).
- **Summary:** The CORS configuration allows all headers and permits GET, POST, PUT, and DELETE methods from any domain, with broad access that can be restricted for security. No additional response headers are exposed, and preflight requests are cached for 3,000 seconds.

Step 13: Implement a Content Delivery Network (CDN)

- **Action Taken:** Set up AWS CloudFront to distribute the S3 bucket content globally.

The screenshot shows the AWS CloudFront Settings page for a distribution named 'aws-s3-project-chidapha.s3-website-us-east-1.amazonaws.com'. The 'Protocol' section is set to 'HTTP only'. The 'HTTP port' is set to 80. The 'Origin path - optional' field contains '/'. The 'Name' field is set to 'aws-s3-project-chidapha.s3-website-us-east-1.amazonaws.com'. There is also a 'Add custom header - optional' section.

- **Configuration:** Configured the CloudFront distribution.
- **CloudFront URL:** <https://d2fpff6mfi9r0t.cloudfront.net>

The screenshot shows the AWS CloudFront Distribution Details page for distribution 'E21T3WOFKLUW4H'. The distribution domain name is 'd2fpff6mfi9r0t.cloudfront.net'. The ARN is 'arnaws:cloudfront::796973509221:distribution/E21T3WOFKLUW4H'. The last modified status is 'Deploying'. The 'General' tab is selected, showing settings like 'Description' (empty), 'Price class' (empty), 'Alternate domain names' (empty), 'Standard logging' (Off), 'Cookie logging' (Off), and 'Default root object' (empty).

Step 14: Implement Serverless Functionality with AWS Lambda

- **Action Taken:** Created a simple AWS Lambda function and integrated it with the S3 bucket.

us-west-1.console.aws.amazon.com/lambda/home?region=us-west-1#/create/function

Create function [Info](#)

Choose one of the following options to create your function.

- Author from scratch**
Start with a simple Hello World example.
- Use a blueprint**
Build a Lambda application from sample code and configuration presets for common use cases.
- Container image**
Select a container image to deploy for your function.

Basic information

Function name
Enter a name that describes the purpose of your function.
S3Function

Function name must be 1 to 64 characters, must be unique to the Region, and can't include spaces. Valid characters are a-z, A-Z, 0-9, hyphens (-), and underscores (_).

Runtime [Info](#)
Choose the language to use to write your function. Note that the console code editor supports only Node.js, Python, and Ruby.
Python 3.12

Architecture [Info](#)
Choose the instruction set architecture you want for your function code.
 x86_64
 arm64

Permissions [Info](#)
By default, Lambda will create an execution role with permissions to upload logs to Amazon CloudWatch Logs. You can customize this default role later when adding triggers.

CloudShell Feedback

us-east-1.console.aws.amazon.com/lambda/home?region=us-east-1#/functions/S3Function?newFunction=true&tab=code

Successfully created the function **S3Function**. You can now change its code and configuration. To invoke your function with a test event, choose "Test".

S3Function

Function overview [Info](#)

Diagram **Template**

S3Function

Layers (0)

+ Add trigger + Add destination

Throttle Copy ARN Actions

Description
-

Last modified
1 minute ago

Function ARN
arn:aws:lambda:us-east-1:796973509221:function:S3Function

Function URL [Info](#)
-

Code Test Monitor Configuration Aliases Versions

Code source [Info](#)

Upload from

CloudShell Feedback

```

    You are using the new console editor.

    lambda_function.py
    1 import json
    2
    3 def lambda_handler(event, context):
    4     # Log the event for debugging
    5     print("Received event: " + json.dumps(event, indent=2))
    6
    7     # Process the S3 object key
    8     bucket = event['Records'][0]['s3']['bucket']['name']
    9     key = event['Records'][0]['s3']['object']['key']
    10
    11     print(f"Bucket: {bucket}, Key: {key}")
    12     # You can add more processing logic here
    13
    14
    15     return {
    16         'statusCode': 200,
    17         'body': json.dumps('Processing completed!')
    18     }

```

The screenshot shows the AWS Lambda console interface. On the left, there's an 'EXPLORER' sidebar with a tree view showing a single function named 'S3FUNCTION' containing a file 'lambda_function.py'. The main area is titled 'S3Function' and contains the provided Python code. Below the code editor, there are buttons for 'Deploy (D+U)' and 'Test (T+R)'. A message at the bottom says 'You haven't created any test events.' and has a 'Create test event (C+M)' button. The bottom navigation bar includes links for CloudShell, Feedback, Privacy, Terms, and Cookie preferences.

- **Configuration:** Set up an event trigger to automatically execute the function when new objects are uploaded to the bucket, enabling efficient serverless computations and data processing.

Add trigger

Trigger configuration [Info](#)

Bucket: s3 aws asynchronous storage

Event types: All object create events

Prefix - optional: e.g. images/

Suffix - optional: e.g. jpg

The screenshot shows the 'Add trigger' configuration page. It's set to trigger on 'All object create events' from the 's3' bucket 's3/aws-s3-project-chidapha'. The 'Bucket' dropdown is set to 's3 aws asynchronous storage'. The 'Event types' section shows 'All object create events' selected. There are also fields for 'Prefix - optional' (containing 'e.g. images/') and 'Suffix - optional' (containing 'e.g. jpg'). The bottom navigation bar includes links for CloudShell, Feedback, Privacy, Terms, and Cookie preferences.

Step 15: Explore S3 Transfer Acceleration

- Action Taken:** Enabled Transfer Acceleration for the S3 bucket to speed up uploads.

- Configuration:** Uploaded a 16.3 MB file in just 2 seconds, indicating that S3 Transfer Acceleration is likely working well.

The screenshot shows the AWS S3 console after a file upload. The top navigation bar includes the AWS logo, services dropdown, search bar, and user information (N. Virginia, Chidapha Chaitharin). A green banner at the top indicates 'Upload succeeded' with a link to 'View details below.' Below this, a modal window titled 'Upload: status' displays summary information: Destination (s3://aws-s3-project-chidapha) with 1 file (Succeeded, 16.3 MB), and Failed (0 files, 0 B). The 'Files and folders' tab is selected, showing a table with one item: OneDrive-20... (application/zip, 16.3 MB, Succeeded). The bottom of the page includes CloudShell, Feedback, and copyright information.

Step 16: Implement Versioned Lifecycle Policies

- Action Taken:** Configured lifecycle policies for versioned objects.
- Configuration:** Set policies to transition older versions to Amazon Glacier.

The screenshot shows the AWS S3 management console for a lifecycle rule. The top navigation bar is identical to the previous screenshot. The main area is titled 'Lifecycle rule configuration' for a rule named 'Transition to Glacier'. It shows the rule is Enabled, applies to the entire bucket, and has no prefix or object tags. Under 'Review transition and expiration actions', it lists actions for Day 0 (upload objects), Day 30 (move to Standard-IA), and Day 365 (objects expire). The bottom of the page includes CloudShell, Feedback, and copyright information.

The screenshot shows the AWS S3 Lifecycle configuration page. At the top, a green banner indicates that a lifecycle rule named "Transition to Glacier" has been successfully added. Below the banner, the page title is "Lifecycle configuration". A sub-header explains that lifecycle configurations define actions for objects throughout their lifetime. The main content area is titled "Lifecycle rules (1)". It shows a single rule named "Transition to Glacier" which is enabled and applies to the entire bucket. The rule's status is "Enabled" and its scope is "Entire bucket". The transition action is set to "Transition to Glacier".

Step 17: Implement Cross-Region Replication

- **Action Taken:** Created a new S3 bucket in a different region.

The screenshot shows the AWS S3 Buckets page. A green banner at the top indicates that a bucket named "destination-cross-region-chidapha" has been successfully created. Below the banner, the page title is "Amazon S3 > Buckets". A sub-header provides an account snapshot. The main content area shows a table of general purpose buckets. The table includes columns for Name, AWS Region, IAM Access Analyzer, and Creation date. Three buckets are listed: "aws-s3-project-chidapha" (US East, November 2, 2024), "destination-cross-region-chidapha" (Europe Frankfurt, November 2, 2024), and "logging-bucket-chidapha" (US East, November 2, 2024).

Name	AWS Region	IAM Access Analyzer	Creation date
aws-s3-project-chidapha	US East (N. Virginia) us-east-1	View analyzer for us-east-1	November 2, 2024, 01:57:42 (UTC+01:00)
destination-cross-region-chidapha	Europe (Frankfurt) eu-central-1	View analyzer for eu-central-1	November 2, 2024, 16:40:46 (UTC+01:00)
logging-bucket-chidapha	US East (N. Virginia) us-east-1	View analyzer for us-east-1	November 2, 2024, 02:46:41 (UTC+01:00)

- **Configuration:** Configured cross-region replication to automatically replicate objects from the original bucket to the new one.

Step 18: Implement Access Control Policies

- Action Taken:** Explored IAM to create policies controlling access to the S3 bucket.

```

1 - {
2   "Version": "2012-10-17",
3   "Statement": [
4     {
5       "Effect": "Allow",
6       "Action": "s3:GetObject",
7       "Resource": "arn:aws:s3:::aws-s3-project-chidapha/*"
8     }
9   ]
10 }
  
```

The screenshot shows the AWS IAM Policies page. The left sidebar includes options like Dashboard, Access management (User groups, Users, Roles, Policies), Access reports (Access Analyzer, External access, Unused access, Analyzer settings, Credential report, Organization activity, Service control policies), and CloudShell/Feedback. The main content area displays a table of policies with columns for Policy name, Type, Use..., and Description. One policy, 'allow', is selected and highlighted.

Policy name	Type	Use...	Description
allow	Customer managed	None	-
AlexaForBusiness...	AWS managed	None	This policy enables Alexa for Business to perform automated tasks sch...
AlexaForBusinessP...	AWS managed	None	Provide access to Poly AVS devices
AlexaForBusinessR...	AWS managed	None	Provide read only access to AlexaForBusiness services
AmazonAPIGatewa...	AWS managed	None	Provides full access to create/edit/delete APIs in Amazon API Gateway ...
AmazonAPIGatewa...	AWS managed	None	Provides full access to invoke APIs in Amazon API Gateway.
AmazonAPIGatewa...	AWS managed	None	Allows API Gateway to push logs to user's account.
AmazonAppFlowF...	AWS managed	None	Provides full access to Amazon AppFlow and access to AWS services su...
AmazonAppFlowR...	AWS managed	None	Provides read only access to Amazon Appflow flows
AmazonAppStrea...	AWS managed	None	Provides full access to Amazon AppStream via the AWS Management ...
AmazonAppStrea...	AWS managed	None	Amazon AppStream 2.0 access to AWS Certificate Manager Private CA ...
AmazonAppStrea...	AWS managed	None	Provides read only access to Amazon AppStream via the AWS Manage...

- **Configuration:** Defined policies for users or roles.

The screenshot shows the AWS IAM User details page for 'Chidapha-Chaitharin'. The left sidebar includes options like Dashboard, Access management (User groups, Users, Roles, Policies), Access reports (Access Analyzer, External access, Unused access, Analyzer settings, Credential report, Organization activity, Service control policies), and CloudShell/Feedback. The main content area shows a 'Summary' section with ARN, Console access status, and access keys. Below it is a 'Permissions' section showing one policy attached: 'allow' (Customer managed, Directly).

ARN	Console access	Access key 1
arn:aws:iam::796973509221:user/Chidapha-Chaitharin	Disabled	Create access key

Permissions policies (1)		
allow	Customer managed	Directly

Step 19: Review Billing and Cost Management

- **Action Taken:** Reviewed usage and costs in the AWS Billing and Cost Management dashboard.

The screenshot shows the AWS Billing and Cost Management home page. It includes sections for Cost summary, Cost monitor, Cost breakdown, and Recommended actions.

Cost summary:

- Month-to-date cost: \$0.00
- Last month's cost for same time period: \$0.00
- Total forecasted cost for current month: Data unavailable
- Last month's total cost: \$0.41

Cost monitor:

- Budgets status: Setup required (No budget created)
- Cost anomalies status (MTD): None detected (1 monitor(s) active)

Cost breakdown:

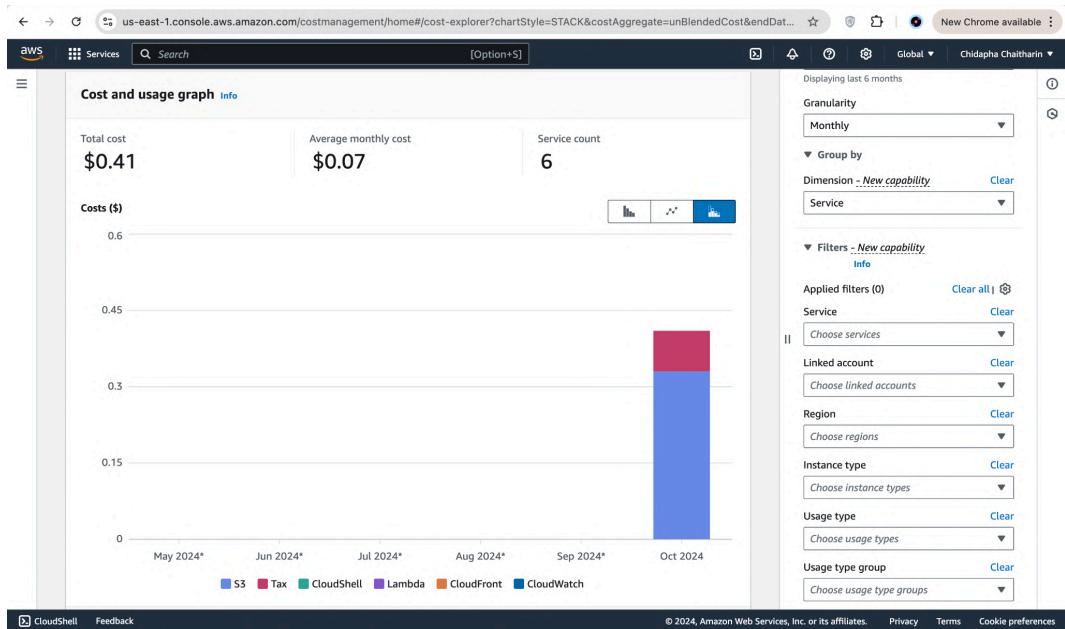
Group costs by Service. A bar chart shows costs from June 2024 to Oct 2024, with a significant blue segment for Oct 2024.

Recommended actions:

- Free Tier usage:** You have exceeded 85% of the Free Tier usage limit for 1 services. (View details)
- Getting started:** Create a cost budget to receive alerts when your costs and usage exceed your budgeted amounts. (Create budget)

Footer: © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

- **Configuration:** Reviewed usage and costs related to S3, ensuring that it stayed within the Free Tier limits.



Summary

Month-to-date usage: 6 service offers in use

Service offerings at or above usage limits: 1 of 6

Service	AWS Free Tier usage limit	Current usage	Forecasted usage	MTD actual usage %	MTD forecasted usage %
Amazon Simple Storage Service	2000.0 Requests for free for 12 months as part of AWS Free Usage Tier (Global-Requests-Tier1)	2,000 Requests	60,000 Requests	100.00%	3000
AWS Lambda	1000000.0 Request are always free per month as part of AWS Free Usage Tier (Global-Request)	2,417 Request	72,510 Request	0.24%	7
AmazonCloudWatch	5.0 GB are always free per month as part of AWS Free Usage Tier (Global-DataProcessing-Bytes)	0 GB	0 GB	0.11%	3

Step 20: Clean Up Resource Selectively

- Action Taken:** Deleted resource except for the S3 bucket, which I can keep active for the purpose of maintaining the endpoint link.

Delete bucket Info

⚠ Deleting a bucket cannot be undone.

Bucket names are unique. If you delete a bucket, another AWS user can use the name.

If this bucket is used with a Multi-Region Access Point in an external account, initiate failover before deleting the bucket.

If this bucket is used with an access point in an external account, the requests made through those access points will fail after you delete this bucket.

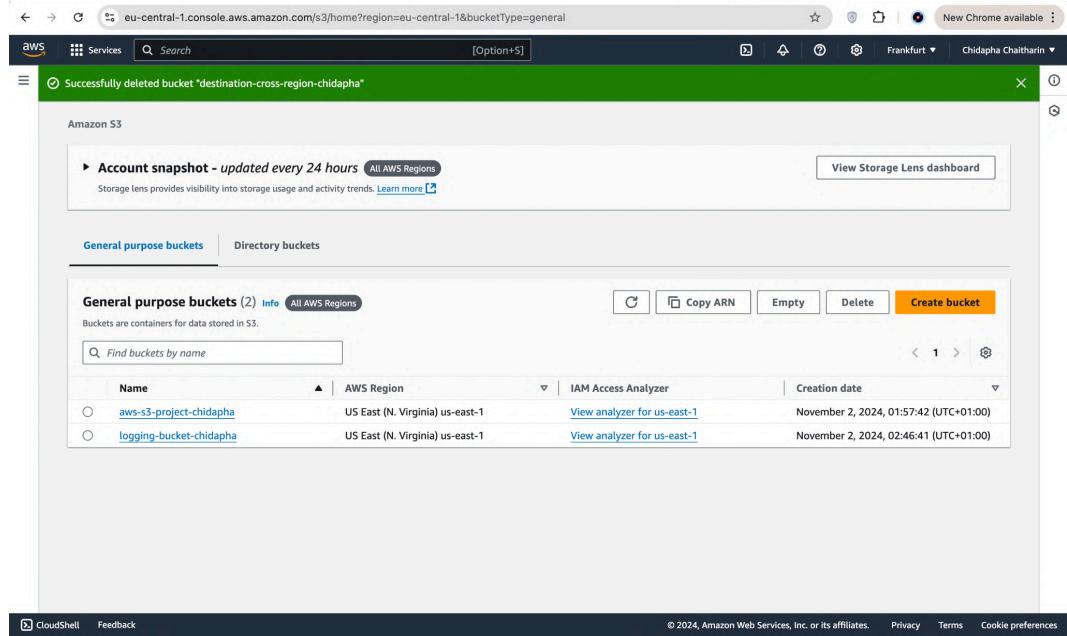
[Learn more](#)

Delete bucket "destination-cross-region-chidapha"?

To confirm deletion, enter the name of the bucket in the text input field.

destination-cross-region-chidapha

[Cancel](#) [Delete bucket](#)



Conclusion

This project successfully demonstrated how to create a static website using AWS S3 and explore various AWS features. The detailed documentation and screenshots provide a comprehensive overview of each step, ensuring clarity and understanding. This project not only familiarised students with AWS S3 but also enhanced their skills in cloud computing and resource management.

Insights and Reflection

This project offered valuable insights into AWS S3's capabilities for hosting and managing a static website, with a focus on security, performance, and scalability.

- **Challenges:** Configuring secure public access was challenging, especially when balancing access permissions with security for CORS and CloudFront.
- **Key Learnings:** AWS S3's advanced features like CDN with CloudFront and Cross-Region Replication are essential for enhancing site performance and availability. Serverless functions with Lambda provided an efficient way to automate processes.
- **Future Applications:** The skills gained in managing permissions, cost optimization, and scalable infrastructure will be highly useful for future cloud-based projects.

Website Endpoint URL

- URL: <http://aws-s3-project-chidapha.s3-website-us-east-1.amazonaws.com/>
-