

AWS S3 Security Project

By Syeda Umaina Abeer

July 2025

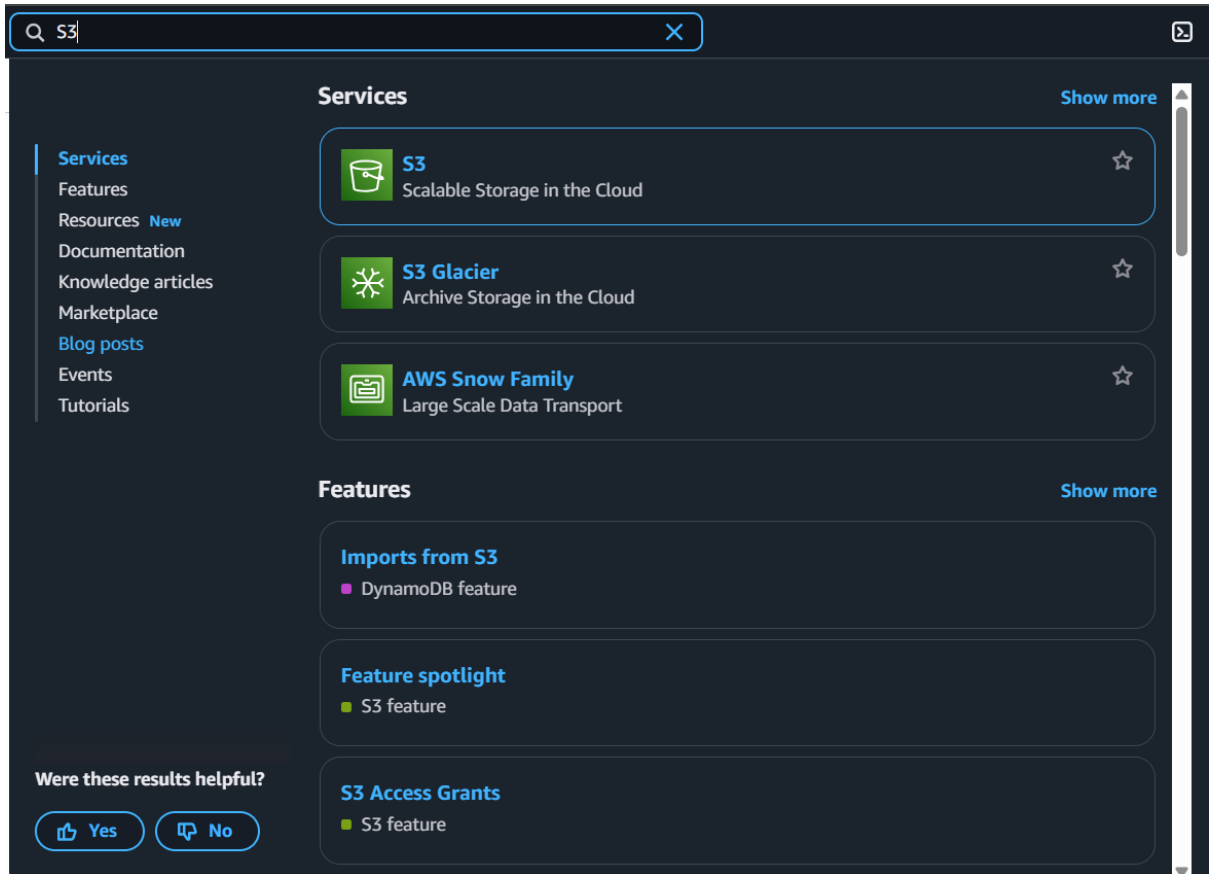
This project demonstrates how to securely configure an Amazon S3 bucket using AWS Identity and Access Management (IAM), bucket policies, server-side encryption, and logging. The goal was to apply core cloud security principles like least privilege access, data encryption at rest, version control, and access monitoring using AWS native tools.

Table of Contents (Optional)

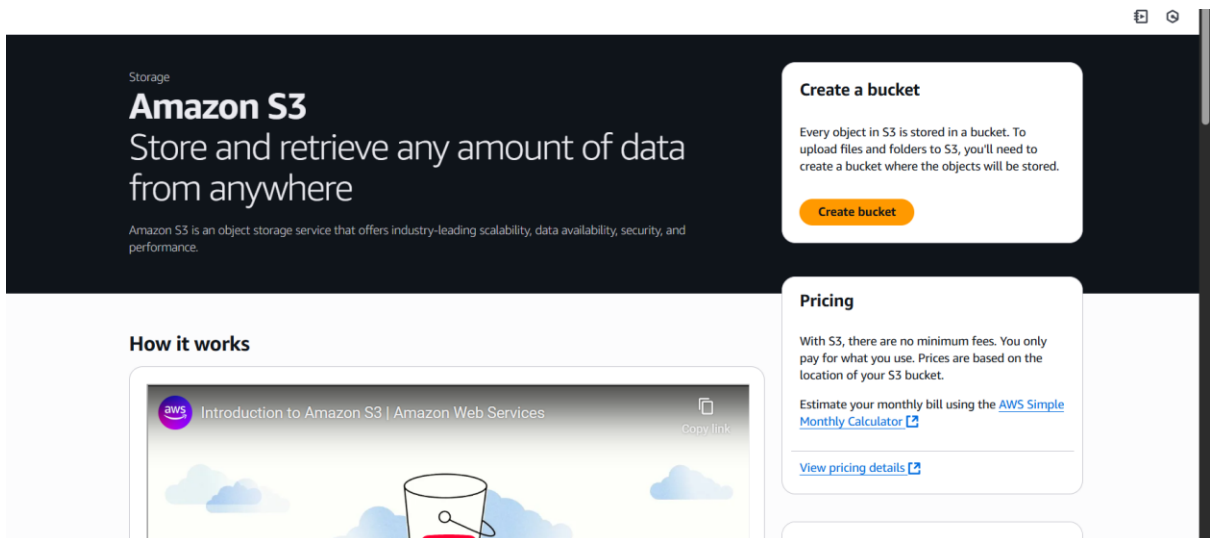
- Step 1: Create S3 Bucket
- Step 2: Block Public Access
- Step 3: IAM User Setup
- Step 4: Bucket Policy
- Step 5: Encryption & Versioning
- Step 6: Access Logging
- Screenshots

➔ Create a New S3 Bucket

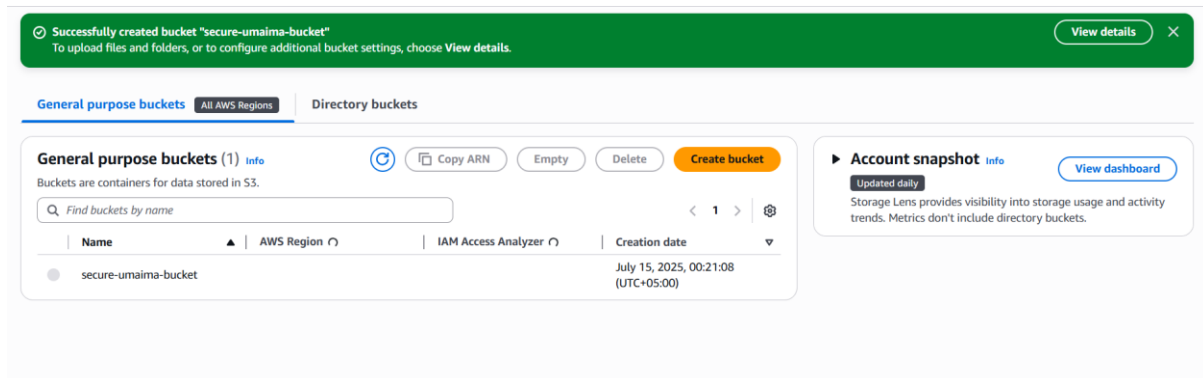
Opened the S3 service from the AWS Management Console to start bucket creation.



Named the bucket and selected region. Disabled public access to ensure security.

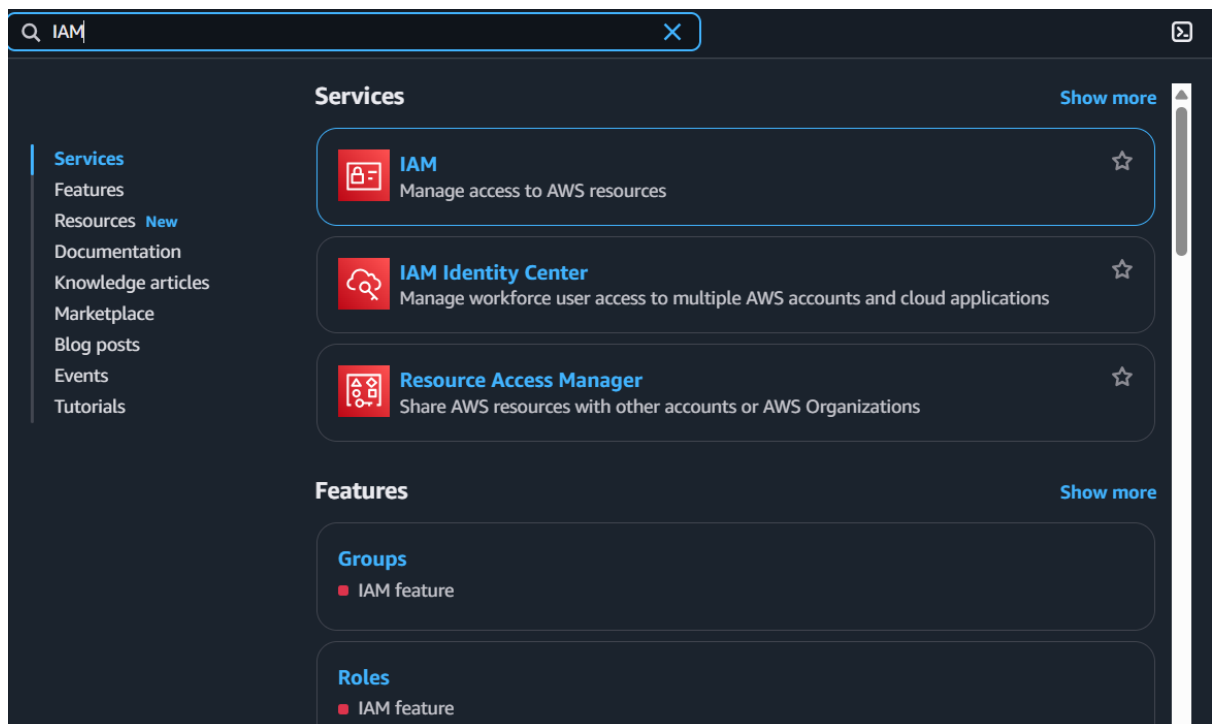


The secure S3 bucket was created and is now listed in the console.



Create IAM User with Limited Permissions

Opened the AWS IAM service from the console to create a new user.



Created a new IAM user with programmatic access for S3 usage.

New access analyzers available
Access Analyzer now analyzes internal access patterns to your critical resources within a single account or across your entire organization.

Create new analyzer

IAM Dashboard

Info

Security recommendations

1

⚠️ Add MFA for root user

Add MFA for root user - Enable multi-factor authentication (MFA) for the root user to improve security for this account.

✅ Root user has no active access keys

Using access keys attached to an IAM user instead of the root user improves security.

Add MFA

IAM resources

Resources in this AWS Account

User groups	Users	Roles	Policies	Identity providers
0	0	2	0	0

AWS Account

Account ID

547900304698

Account Alias

Create

Sign-in URL for IAM users in this account

https://547900304698.signin.aws.amazon.com/console

Quick Links

My security credentials

Manage your access keys, multi-factor authentication (MFA) and other credentials.

Created a new IAM user with programmatic access to securely access S3 buckets.

Create user

ails

Specify user details

User details

User name

s3-access-user

The user name can have up to 64 characters. Valid characters: A-Z, a-z, 0-9, and + = , . @ _ - (hyphen)

☐ Provide user access to the AWS Management Console - optional

If you're providing console access to a person, it's a [best practice](#) to manage their access in IAM Identity Center.

ⓘ If you are creating programmatic access through access keys or service-specific credentials for AWS CodeCommit or Amazon Keyspaces, you can generate them after you create this IAM user. [Learn more](#)

Cancel

Next

Attached AmazonS3ReadOnlyAccess managed policy to the IAM user for limited S3 access.

Set permissions

Add user to an existing group or create a new one. Using groups is a best-practice way to manage user's permissions by job functions. [Learn more](#)

Permissions options

☐ Add user to group

Add user to an existing group, or create a new group. We recommend using groups to manage user permissions by job function.

☐ Copy permissions

Copy all group memberships, attached managed policies, and inline policies from an existing user.

☒ Attach policies directly

Attach a managed policy directly to a user. As a best practice, we recommend attaching policies to a group instead. Then, add the user to the appropriate group.

Permissions policies (1/1372)

Choose one or more policies to attach to your new user.



[Create policy](#)

Filter by Type			
<input type="text" value="AmazonS3ReadOnlyAccess"/>		<input type="button" value="X"/>	
<input type="button" value="All types"/>		1 match	<input type="button" value="1"/>
<input checked="" type="checkbox"/>	Policy name	Type	Attached entities
<input checked="" type="checkbox"/>	AmazonS3ReadOnlyAccess	AWS managed	0

► Set permissions boundary - *optional*

Reviewed IAM user details and attached policy before final creation.

Review your choices. After you create the user, you can view and download the autogenerated password, if enabled.

User details

User name
s3-access-user

Console password type
None

Require password reset
No

Permissions summary

Name	Type	Used as
AmazonS3ReadOnlyAccess	AWS managed	Permissions policy

Tags - optional

Tags are key-value pairs you can add to AWS resources to help identify, organize, or search for resources. Choose any tags you want to associate with this user.

No tags associated with the resource.

[Add new tag](#)

You can add up to 50 more tags.

[Cancel](#)

[Previous](#)

[Create user](#)

IAM user created successfully with secure programmatic access to Amazon S3.

✔ User created successfully

View user

✕

You can view and download the user's password and email instructions for signing in to the AWS Management Console.

Users (1) [Info](#)

Ⓢ

Delete

Create user

An IAM user is an identity with long-term credentials that is used to interact with AWS in an account.

🔍 Search

< 1 > ⚙️

<input type="checkbox"/>	User name	Path	Group	Last activity	MFA	Password age	Console last sign-in	A
<input type="checkbox"/>	s3-access-user	/	0	-	-	-	-	-

Add a Bucket Policy to Control Access

Opened the bucket policy editor to define user-level access to objects in the bucket.

Bucket policy

Edit

Delete

The bucket policy, written in JSON, provides access to the objects stored in the bucket. Bucket policies don't apply to objects owned by other accounts. [Learn more](#)

No policy to display.

Copy

Added a JSON policy to allow read-only access to a specific IAM user.

Bucket ARN
arn:aws:s3:::secure-umaima-bucket

Policy

```
1 {  
2   "Version": "2012-10-17",  
3   "Statement": [  
4     {  
5       "Effect": "Allow",  
6       "Principal": {  
7         "AWS": "arn:aws:iam::ACCOUNT-ID:user/s3-access-user"  
8       },  
9       "Action": "s3:GetObject",  
10      "Resource": "arn:aws:s3:::secure-umaima-bucket/*"  
11    }  
12  ]  
13 }  
14
```

Edit statement [Remove](#)

Add actions

Choose a service

Included
S3

Available
[AI Operations](#)
[AMP](#)
[API Gateway](#)

I change Account-ID by my account ID

✓ Successfully edited bucket policy.

Bucket policy [Edit](#) [Delete](#)

The bucket policy written in JSON provides access to the objects stored in the bucket. Bucket policies don't apply to objects owned by other accounts. [Learn more](#)

Enable Server-Side Encryption (SSE)

Enabled server-side encryption to protect data at rest using Amazon S3-managed keys.

Edit default encryption [Info](#)

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)
- ☐ Server-side encryption with AWS Key Management Service keys (SSE-KMS)
- ☐ Dual-layer server-side encryption with AWS Key Management Service keys (DSSE-KMS)
Secure your objects with two separate layers of encryption. For details on pricing, see DSSE-KMS pricing on the [Storage](#) tab of the [Amazon S3 pricing page](#). [↗](#)

Bucket Key

Using an S3 Bucket Key for SSE-KMS reduces encryption costs by lowering calls to AWS KMS. S3 Bucket Keys aren't supported for DSSE-KMS. [Learn more](#) [↗](#)

- ☐ Disable
- ☒ Enable

[Cancel](#)

[Save changes](#)

Turned on versioning to retain old versions of files for better data recovery and protection.

Edit Bucket Versioning [Info](#)

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

- ☐ Suspend
This suspends the creation of object versions for all operations but preserves any existing object versions.
- ☒ Enable

[i](#) After enabling Bucket Versioning, you might need to update your lifecycle rules to manage previous versions of objects.

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 using the Amazon S3 REST API. [Learn more](#) [↗](#)

Disabled

Edit Bucket Versioning [Info](#)

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

☐ Suspend
This suspends the creation of object versions for all operations but preserves any existing object versions.

☒ Enable

[i](#) After enabling Bucket Versioning, you might need to update your lifecycle rules to manage previous versions of objects.

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 using the Amazon S3 REST API. [Learn more](#) [↗](#)

Disabled

Enable Server Access Logging

Enabled access logging for monitoring all access events on the secure S3 bucket.

✔ Successfully created bucket "umaima-logs-bucket"
To upload files and folders, or to configure additional bucket settings, choose **View details**.

General purpose buckets

All AWS Regions

Directory buckets

General purpose buckets (2) [Info](#)



Copy ARN

Empty

Buckets are containers for data stored in S3.

Find buckets by name

	Name	AWS Region	IAM Access Analyzer
<input type="radio"/>	umaima-logs-bucket	Asia Pacific (Mumbai) ap-south-1	View analyzer for ap-south-1
<input type="radio"/>	secure-umaima-bucket	Asia Pacific (Mumbai) ap-south-1	View analyzer for ap-south-1

Created a dedicated bucket to receive access logs from the main secure S3 bucket.

Enabled Access Logging to Target Bucket: Configured logging to send all access events to umaima-logs-bucket with prefix logs/

☒ Enable

Bucket policy will be updated

When you enable server access logging, the S3 console automatically updates your bucket policy to include access to the S3 log delivery group.

Destination

Specify a destination bucket in the Asia Pacific (Mumbai) ap-south-1 Region. To store your logs under a particular prefix, make sure that you include a slash (/) after the name of the prefix. Otherwise, the prefix will be added to the name of your log files.

s3://umaima-logs-bucket/logs/

[Browse S3](#)

Format: s3://<bucket>/<optional-prefix-with-path>

Destination Region

Asia Pacific (Mumbai) ap-south-1

Destination bucket name

umaima-logs-bucket

Destination prefix

logs/

Log object key format

☒ [DestinationPrefix][YYYY]-[MM]-[DD]-[hh]-[mm]-[ss]-[UniqueString]

☐ [DestinationPrefix][SourceAccountid]/[SourceRegion]/[SourceBucket]/[YYYY]/[MM]/[DD]/[YYYY]-[MM]-[DD]-[hh]-[mm]-[ss]-[UniqueString]

To speed up analytics and query applications, use this format.

Log object key example

logs/2025-07-01-10-12-56-[UniqueString]

Final Summary

The AWS S3 Security Project was completed by implementing a secure and private S3 bucket with strict IAM-based access, custom bucket policies, encryption at rest using SSE-S3, object versioning, and full access logging. This project helped solidify key concepts in cloud security and gave practical experience in AWS console-based configurations.