

Secure Cloud Storage & Backup Management System

Introduction

This project demonstrates a secure cloud-based storage and backup solution. It ensures data availability, controlled access, and recovery from failures using cloud storage services.

Problem Statement

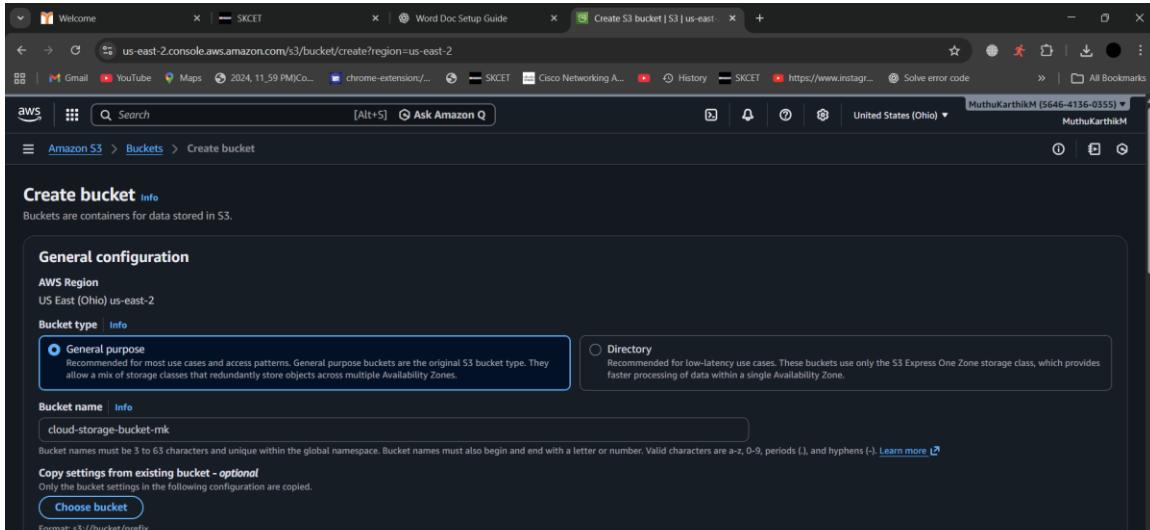
An organization has lost critical internal documents due to accidental deletion and disk failures. There is no backup policy, no retention strategy, and no centralized access control. Management demands a secure cloud-based storage and backup solution that ensures data availability and controlled access.

Objectives

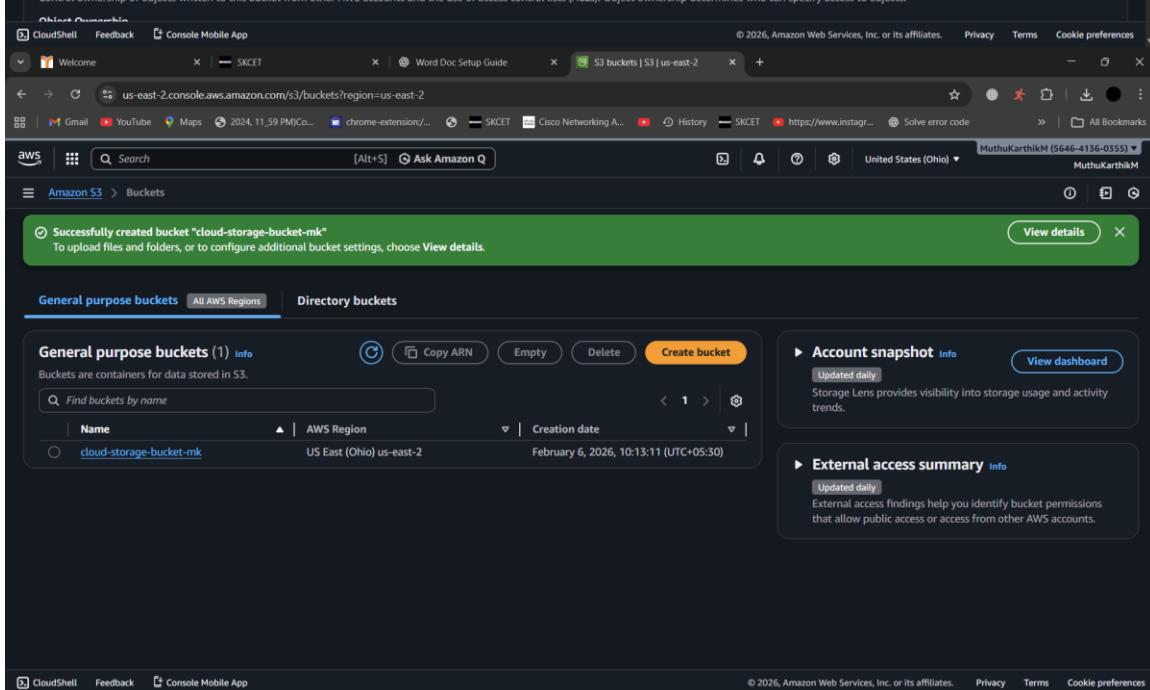
- Store business files in cloud storage
- Take periodic backups
- Restore snapshots in case of failure
- Restrict access based on roles (admin/user)
- Understand storage performance (IOPS, throughput)

Architecture

The system uses cloud object or block storage. Files are uploaded to cloud storage, snapshots are taken periodically, and access is controlled using IAM roles.



The screenshot shows the 'Create bucket' page in the AWS S3 console. Under 'General configuration', the 'AWS Region' is set to 'US East (Ohio) us-east-2'. The 'Bucket type' dropdown has 'General purpose' selected (indicated by a blue outline). Other options include 'Directory', which is described as recommended for low-latency use cases. The 'Bucket name' field contains 'cloud-storage-bucket-mk'. Below it, a note specifies bucket names must be 3 to 63 characters and unique. A 'Copy settings from existing bucket - optional' section is present, with a 'Choose bucket' button and a note about copied settings. The 'Format' is set to 's3://bucket/prefix'.



The screenshot shows the 'Buckets' page in the AWS S3 console. It lists one bucket: 'cloud-storage-bucket-mk' (General purpose, US East (Ohio) us-east-2). The bucket was created on February 6, 2026, at 10:13:11 (UTC+05:30). A green success message at the top states 'Successfully created bucket "cloud-storage-bucket-mk"'. Below the table, there are sections for 'Account snapshot' (Storage Lens provides visibility into storage usage and activity trends) and 'External access summary' (External access findings help identify bucket permissions).

Implementation Steps

Step 1: Enable Backup/Snapshot Feature

The screenshot shows the AWS S3 Bucket Properties page for 'cloud-storage-bucket-mk'. The 'Properties' tab is selected. In the 'Bucket Versioning' section, there is a green success message: 'Successfully edited Bucket Versioning. To transition, archive, or delete older object versions, configure lifecycle rules for this bucket.' The 'Bucket Versioning' status is now 'Enabled'. Other sections like 'Multi-factor authentication (MFA) delete' and 'Bucket ABAC' are also visible.

Step 2: Upload Files

The screenshot shows the AWS S3 console interface. The top navigation bar includes tabs for Welcome, SKCET, Word Doc Setup Guide, and cloud-storage-bucket-mk - S3. The main navigation bar shows the path: Amazon S3 > Buckets > cloud-storage-bucket-mk. Below this, the bucket name "cloud-storage-bucket-mk" is displayed with a "Info" link. A horizontal menu bar offers options like Objects, Metadata, Properties, Permissions, Metrics, Management, and Access Points. The main content area is titled "Objects (1)". It lists a single file: "Screenshot 2026-02-06 101357.png", which is a PNG file from February 6, 2026, at 10:14:28 UTC+05:30, with a size of 178.8 KB and a storage class of Standard. There are buttons for Actions (Copy S3 URI, Copy URL, Download, Open in new tab, Delete), Create folder, and Upload.

Step 3: Configure Access Control (Admin/User)

The screenshot shows the "Create user" wizard in the AWS IAM console. The top navigation bar includes tabs for Welcome, SKCET, Word Doc Setup Guide, and Create user | IAM | Global. The main navigation bar shows the path: IAM > Users > Create user. On the left, a sidebar shows steps: Step 1 (Specify user details, currently selected), Step 2 (Set permissions), and Step 3 (Review and create). The main content area is titled "Specify user details". It has a section for "User details" where the "User name" is set to "storage_user". A note states: "The user name can have up to 64 characters. Valid characters: A-Z, a-z, 0-9, and + . @ _ - (hyphen)". An optional checkbox "Provide user access to the AWS Management Console - optional" is checked, with a note: "In addition to console access, users with SigninLocalDevelopmentAccess permissions can use the same console credentials for programmatic access without the need for access keys." A callout box provides information about generating access keys for AWS CodeCommit or Amazon Keyspaces. At the bottom right are "Cancel" and "Next" buttons.

Step 1

- Specify user details
- Set permissions
- Review and create

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 (1/1445)

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

| Filter by Type | |
|--|-------------|
| <input type="text" value="amazonss3rea"/> | All types |
| <input checked="" type="checkbox"/> Policy name | Type |
| <input checked="" type="checkbox"/>  AmazonS3ReadOnlyAccess | AWS managed |
| 0 | |

Set permissions boundary - optional

User created successfully

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

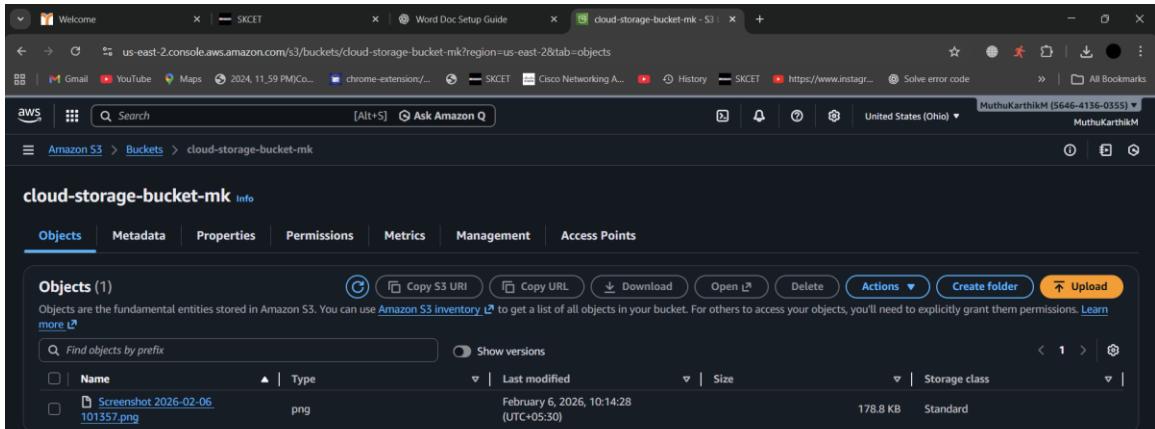
[View user](#)

Users (2) Info

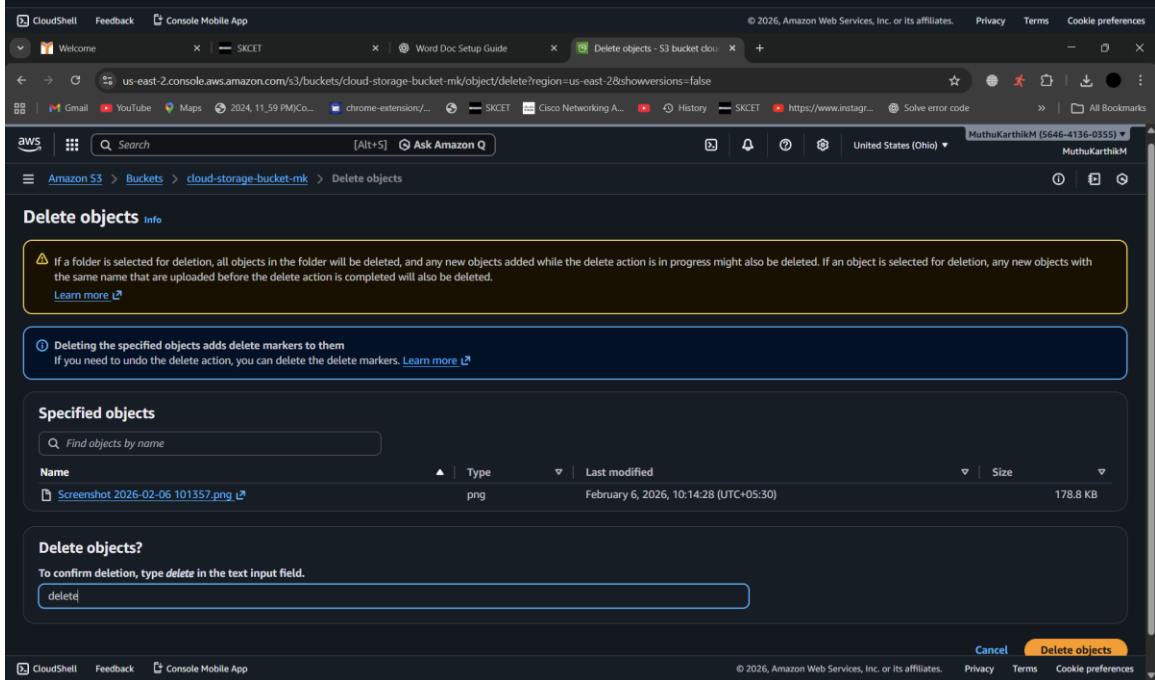
An IAM user is an identity with long-term credentials that is used to interact with AWS in an account.

| User name | Path | Group | Last activity | MFA | Password age | Console last sign-in | Acc |
|--------------|------|-------|---------------|-----|--------------|----------------------|-----|
| storage_user | / | 0 | - | - | - | - | - |
| user1 | / | 0 | 8 days ago | - | 8 days | 8 days ago | - |

Step 4: Simulate Failure and Restore Data



The screenshot shows the AWS S3 console interface. The top navigation bar includes tabs for Welcome, SKCET, Word Doc Setup Guide, and cloud-storage-bucket-mk - S3. The main content area displays the 'cloud-storage-bucket-mk' bucket. The 'Objects' tab is selected, showing a single object: 'Screenshot 2026-02-06 101357.png'. The object details are: Name: Screenshot 2026-02-06 101357.png, Type: png, Last modified: February 6, 2026, 10:14:28 (UTC+05:30), Size: 178.8 KB, Storage class: Standard.



The screenshot shows the 'Delete objects' confirmation dialog. The top navigation bar includes tabs for CloudShell, Feedback, Console Mobile App, Welcome, SKCET, Word Doc Setup Guide, and Delete objects - S3 bucket cloud-storage-bucket-mk. The main content area displays the 'Delete objects' dialog. It contains two sections: 'Delete objects?' and 'Specified objects'. The 'Delete objects?' section has a note: 'To confirm deletion, type delete in the text input field.' Below it is a text input field with the word 'delete'. The 'Specified objects' section shows the same object as the previous screenshot: 'Screenshot 2026-02-06 101357.png'. The object details are: Name: Screenshot 2026-02-06 101357.png, Type: png, Last modified: February 6, 2026, 10:14:28 (UTC+05:30), Size: 178.8 KB.

The screenshot shows two consecutive screenshots of the AWS S3 console.

Screenshot 1: Delete objects - S3 bucket status

This screen shows the results of a delete operation:

- Successfully deleted:** 1 object, 178.8 KB
- Failed to delete:** 0 objects

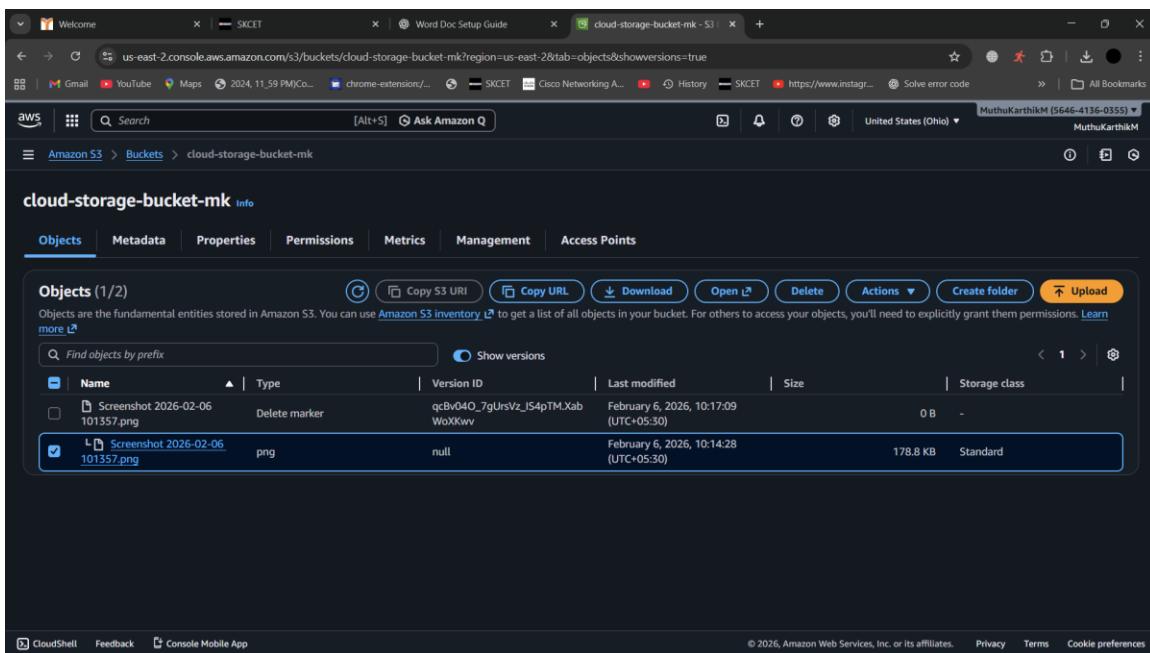
A message at the bottom states: "After you navigate away from this page, the following information is no longer available."

Screenshot 2: cloud-storage-bucket-mk Objects

This screen shows the contents of the bucket:

- Objects (2):**
 - Screenshot 2026-02-06 101357.png**: Type: Delete marker, Version ID: qEBv04O_7gUrsVz_lS4pTM.Kab.WoXkwv, Last modified: February 6, 2026, 10:17:09 (UTC+05:30), Size: 0 B, Storage class: Standard.
 - L Screenshot 2026-02-06 101357.png**: Type: png, Version ID: null, Last modified: February 6, 2026, 10:14:28 (UTC+05:30), Size: 178.8 KB, Storage class: Standard.

Actions buttons include: Copy S3 URI, Copy URL, Download, Open, Delete, Actions, Create folder, and Upload.



Security Features

- Encryption at rest
- Encryption in transit
- Role-based access control
- Backup retention policy

Recovery Process

In case of accidental deletion or disk failure, the admin restores data using snapshots or backups.

Conclusion

This project ensures reliable and secure cloud storage with proper backup and recovery mechanisms.