

Secure File Sharing Portal with Automated Monitoring & Notification On AWS

Abstract:

This project is all about making sharing files safe and easy for businesses. It uses Amazon Web Services (AWS) to create a secure online portal where people can upload files, share them with others through special secure links, and get automatic alerts when something happens with their files. The system relies on AWS tools like S3 for storage, IAM for access control, CloudWatch for monitoring, SNS for notifications, and Python for programming. Everything is designed to keep files protected and the process smooth, without needing complicated tech setups.

Introduction:

File sharing plays a big role in how businesses get work done, but old-school methods can leave sensitive information vulnerable to unauthorized access or make it hard to track who sees what. This project takes a modern approach, using cloud tools from Amazon Web Services (AWS) to keep shared files safe, make monitoring easy, and help organizations save money.

With this system, anyone can:

- **Easily upload files** to a highly secure online storage space, so documents are protected from prying eyes.
- **Share files safely** by giving others special links that work only for a limited time—keeping control in the right hands.
- **Keep track of activity automatically:** every time a file is uploaded, AWS CloudWatch logs the event and helps catch any suspicious behaviour.
- **Stay in the loop with instant notifications:** whenever something important happens (like a new file upload), the system can send alerts via email or text message using AWS SNS.

Objective:

1. Create a Secure File-Sharing Portal on AWS

Build a safe, scalable platform for uploading and sharing files with strong security.

Ensure reliability, compliance, and protection of sensitive data

2. Automate Monitoring of Uploads

Enable automated checks to track file uploads and detect risks.

Reduce manual effort while ensuring accurate activity monitoring.

3. Send Real-Time Notifications with Amazon SNS

Keep users updated with instant alerts on uploads or issues.

Deliver notifications across email, SMS, or applications.

4. Protect Files with Pre-Signed URLs

Provide secure, time-limited access links to files.

Allow only authorized users to view or download content.

Methodology:

1. AWS Resource Setup

- **IAM User:** Created with limited access (S3 + SNS).
- **S3 Bucket:** Configured as private for secure storage.
- **CloudWatch:** Set up to track upload events.
- **SNS:** Configured to send alerts to subscribers.

2. Python Integration

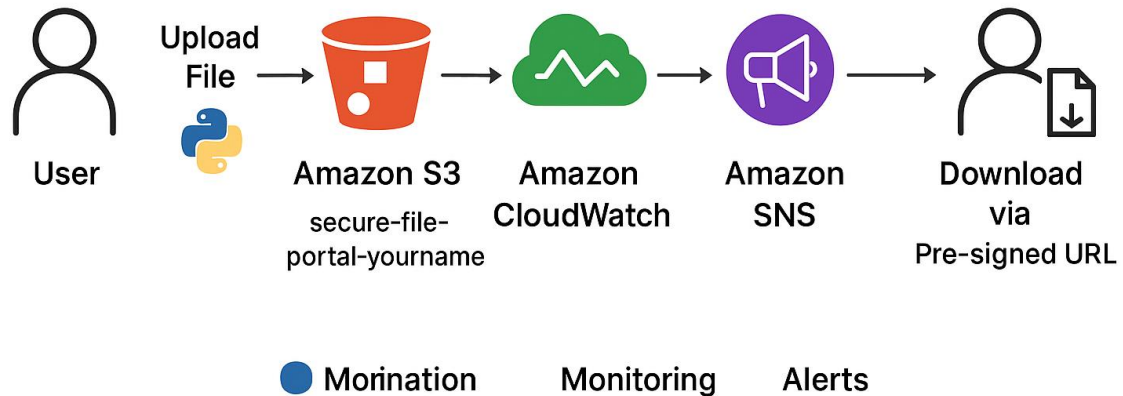
A Python script using **boto3** was developed to:

- Upload files to the S3 bucket.
- Generate pre-signed URLs with expiry time.

3. Workflow

- User runs the script and uploads a file.
- File is stored securely in the private bucket.
- A **secure link** (valid for 1 hour) is generated.
- CloudWatch logs the event, and SNS sends a notification.
- Recipient downloads the file using the link.

Secure File Sharing Portal with Automated Monitoring & Notifications



Architecture Workflow:

- 1. User → Upload File (via Python Script / VS Code CLI)**
 - The user runs your `upload_file.py` script.
 - File is uploaded into the S3 bucket (secure-file-portal-yourname).
 - After upload, an S3 pre-signed URL is generated for secure sharing.
- 2. Amazon S3 → Storage & Access Control**
 - Files are stored securely in the bucket.
 - IAM policies ensure only authorized users/apps can upload.
 - Pre-signed URLs ensure controlled access (temporary, expiring links).
- 3. Amazon CloudWatch / EventBridge → Monitoring**
 - Any file upload triggers an S3 Event Notification.
 - CloudWatch monitors logs (success, failure, access attempts).
- 4. Amazon SNS (Simple Notification Service) → Alerts**
 - When an upload or access happens, SNS sends email/SMS alerts to admins.
- 5. End User → Download via Pre-signed URL**
 - Recipient clicks the URL and securely downloads the file.
 - Link expires after a set time → prevents misuse.

Advantages:

- Secure and private file sharing.
- Automated event tracking.
- Notifications ensure real-time updates.
- Cost-efficient serverless solution.

Result:

The system successfully uploaded files into the S3 bucket. Pre-signed URLs were generated to provide secure, time-limited access. Using these links, recipients could safely download the files. Meanwhile, CloudWatch logged the events and SNS delivered automated notifications.

Conclusion:

The **Secure File Sharing Portal** successfully demonstrates how AWS cloud services can be used to build a **secure, automated, and efficient file sharing system**. With S3 for storage, CloudWatch for monitoring, SNS for notifications, and Python for automation, the project ensures confidentiality, reliability, and ease of access.