Secure File Sharing System - Complete Step-by-Step Guide

Name: HARSHIKA HARESH GANGAN

Cyber Security Intern

Project Title: Secure File Sharing using Flask and AES Encryption

Internship Task: Cyber Security Task 3 - FutureSkills Prime

- Project Overview:

This project allows a user to securely upload and download files via a web interface.
All uploaded files are encrypted using AES (Advanced Encryption Standard) and stored.
During download, the files are decrypted and returned to the user.

- Technologies & Libraries Used:

- Python 3
- Flask (for web server)
- PyCryptodome (for AES encryption)
- HTML (for the upload form)

- Project Structure:

secure-file-share/

- app.py - Main Flask application
- templates/index.html - Upload form UI (HTML)
- uploads/ - Stores encrypted and decrypted files

- AES Encryption:

- AES Mode: CFB (Cipher Feedback Mode)
- Key Length: 16 bytes (AES-128)
- Library: pycryptodome
- Both encryption and decryption use a fixed key and IV

Example used:
KEY = b'ThisIsASecretKey1'
IV = b'ThisIsInitVector'

- Step-by-Step Setup Instructions:

Step 1: Install Python dependencies
pip install flask pycryptodome

Step 2: Create Folder Structure
Create a folder named 'secure-file-share' and inside it:
- Place 'app.py'
- Create 'templates' folder - inside it place 'index.html'
- Create an empty folder named 'uploads'

Step 3: Write the HTML File
Basic upload form for the frontend.

Step 4: Write the Flask App
Flask routes for upload, encrypt, and download decrypt.

Step 5: Run the Application
python app.py
Visit the app at: http://127.0.0.1:5000/

Step 6: Upload a Test File
Upload a test file; it will be encrypted and stored.

Step 7: Download the Decrypted File
Download the file in its original form from the encrypted storage.

- Additional Notes:

- All encrypted files are stored with a .enc extension.
- The AES key and IV are hardcoded for demo purposes.
- Use secure key management for production deployments.

- Submission Checklist:

- [x] Flask File Upload
- [x] AES Encryption
- [x] HTML Upload Page
- [x] Decryption and Download
- [x] Documentation
- [x] Ready for Submission

You have successfully completed Task 3.