INTERNSHIP PROJECT REPORT

Project Title : Secure File Storage System (AES-256)

Submitted to: Elevate Lab

Submitted by: LEKHA SRI G

Date: 27/10/2025

Abstract

The Secure File Storage System (SFS) is a Python-based application that focuses on securing user files through AES-256 encryption. The main goal of this project is to protect sensitive information from unauthorized access and data breaches. The system allows users to easily encrypt files before storing them and decrypt them when needed, ensuring confidentiality and integrity. The project provides both a Command-Line Interface (CLI) and a Graphical User Interface (GUI), making it flexible for users with different levels of technical expertise. Through this project, users can gain hands-on experience in implementing real-world data security mechanisms using Python.

Introduction

As the amount of digital data grows, so does the need for secure file management systems. Unauthorized access, hacking, and accidental sharing of private information are common threats in today's world. The Secure File Storage System (AES-256) provides an effective and user-friendly solution for storing and retrieving files safely. It uses AES-256, an advanced encryption standard widely used across the world for secure data transmission. The project's modular design allows users to work with files through both command-line and graphical interfaces, making it suitable for beginners and professionals alike.

Tools Used

- Python 3:The main programming language used for development.
- Tkinter: Used for designing the graphical user interface (GUI).
- Cryptography Library: Handles AES-256 encryption and decryption operations.
- OS and Sys Modules: Used for handling files and directories securely.
- Pytest: For testing and validating all modules of the system.

Steps Involved in Building the Project

- **Environment Setup:**

Installed Python and all required dependencies using pip, ensuring compatibility and proper configuration of the development environment.

- **Core Module Development:**

Implemented encryption and decryption functionalities using the AES-256 algorithm to ensure strong data confidentiality.

- **Interface Design:**

Designed both Command Line Interface (CLI) and Graphical User Interface (GUI) components to facilitate user interaction and improve accessibility.

- **File Storage Logic:**

Developed secure mechanisms for file saving, retrieval, and deletion, ensuring encrypted data is handled safely throughout its lifecycle.

- **Error Handling:**

Incorporated robust validation and error-checking routines to manage invalid file inputs, encryption failures, and other potential exceptions.

- **Testing**:

Conducted extensive testing using Pytest to validate module performance, functionality, and overall system stability.

- **Final Integration:**

Integrated all modules into a cohesive system, ensuring seamless operation and smooth user experience across interfaces.

Conclusion

The Secure File Storage System (AES-256) project successfully demonstrates the application of advanced encryption techniques in a user-friendly and practical setting. It provides users with a secure platform to store personal or professional files, safeguarding them against unauthorized access.