Data Encryption & Decryption tool with added functionality

The domain of the Project:

Cybersecurity

Team Mentors:

Derick Johnson

Team Members:

Mr. Akshay S. Sannakki
Ms. Ainala Nikhitarchana

Period of the project

November 2025 to December 2025

Declaration

The project titled “Data Encryption & Decryption tool with added functionality” has been mentored by **Derick Johnson**, organised by SURE Trust, from June 2025 to December 2025, for the benefit of the educated unemployed rural youth for gaining hands-on experience in working on industry relevant projects that would take them closer to the prospective employer. I declare that to the best of my knowledge the members of the team mentioned below, have worked on it successfully and enhanced their practical knowledge in the domain.

Team Members :

Mr. Akshay S. Sannakki
Ms. Ainala nikhitarchana

Mentor’s Name : Derick Johnson

Prof. Radhakumari
Executive Director & Founder
SURE Trust

Table of contents

Executive Summary

The **Blowfish Secure File Encryption Tool** is a comprehensive security-oriented software solution developed to address the growing need for protecting sensitive digital data. With the rapid increase in digital file usage across individuals, organizations, and institutions, ensuring confidentiality, integrity, and controlled access to information has become critical. This project aims to provide a reliable and user-friendly system that safeguards files against unauthorized access, tampering, and data leakage.

The project utilizes the **Blowfish symmetric key cryptographic algorithm**, chosen for its speed, efficiency, and strong security characteristics. The system supports file-level encryption and decryption, ensuring that sensitive data remains protected even if files are accessed by unauthorized entities. In addition, cryptographic hashing is implemented to verify file integrity, ensuring that encrypted or decrypted data has not been altered during storage or transmission.

A key highlight of the project is the inclusion of **secure file deletion**, which overwrites file contents before deletion, significantly reducing the risk of data recovery using forensic techniques. The application provides both a **Graphical User Interface (GUI)** for ease of use and a **Command Line Interface (CLI)** for flexibility, testing, and automation.

The project successfully demonstrates the practical implementation of cryptographic principles, secure file handling, and modular software architecture. It serves as an effective educational tool while also offering real-world applicability in environments requiring data protection. The results indicate that the system is functional, reliable, and extensible, with strong potential for future enhancements and industry adoption.

Introduction

1. Background and Context of the Project

With the rapid growth of digital data, **data security and privacy** have become critical concerns across industries and individuals. Sensitive information stored in files is vulnerable to theft, tampering, and unauthorized access if not properly protected. Cryptographic techniques play a vital role in ensuring data confidentiality and integrity.

Blowfish is a symmetric-key block cipher known for its **speed, simplicity, and effectiveness**, making it suitable for file encryption applications.

2. Problem Statement / Goals of the Project

Many existing encryption tools are either:

- Too complex for non-technical users, or
- Lack features such as secure deletion and integrity verification.

The goal of this project is to build a **simple yet secure file encryption system** that:

- Protects files using strong cryptography
- Ensures data integrity
- Prevents easy file recovery after deletion
- Provides an intuitive interface

3. Scope and Limitations of the Project

Scope:

- File encryption and decryption using Blowfish
- Hash-based integrity verification
- Secure deletion of files
- GUI and CLI-based usage

Limitations:

- Uses symmetric key encryption (key management is user-dependent)
- Not optimized for very large files with parallel processing
- Not yet packaged as a standalone executable

Project Objectives

The primary objective of this project is to design and develop a secure, efficient, and user-friendly file encryption system that ensures confidentiality, integrity, and controlled access to digital data.

Key Objectives:

- **To implement Blowfish-based symmetric encryption for secure file storage and transmission**
- **To enable reliable decryption of files using authorized secret keys**
- **To implement cryptographic hashing mechanisms for verifying file integrity**
- **To provide a secure deletion mechanism that minimizes the possibility of file recovery**
- **To develop an intuitive GUI-based application for non-technical users**
- **To provide a CLI-based interface for advanced users, testing, and automation**
- **To design the system using a modular and scalable architecture**
- **To gain hands-on experience in applying cryptography concepts in real-world software development**

Expected Outcomes and Deliverables:

- **A fully functional file encryption and decryption tool**
- **Secure handling of binary and text-based files**
- **User-friendly graphical interface**
- **Well-structured and documented source code**
- **A GitHub repository for version control and collaboration**
- **A project report documenting methodology, results, and learning outcomes**

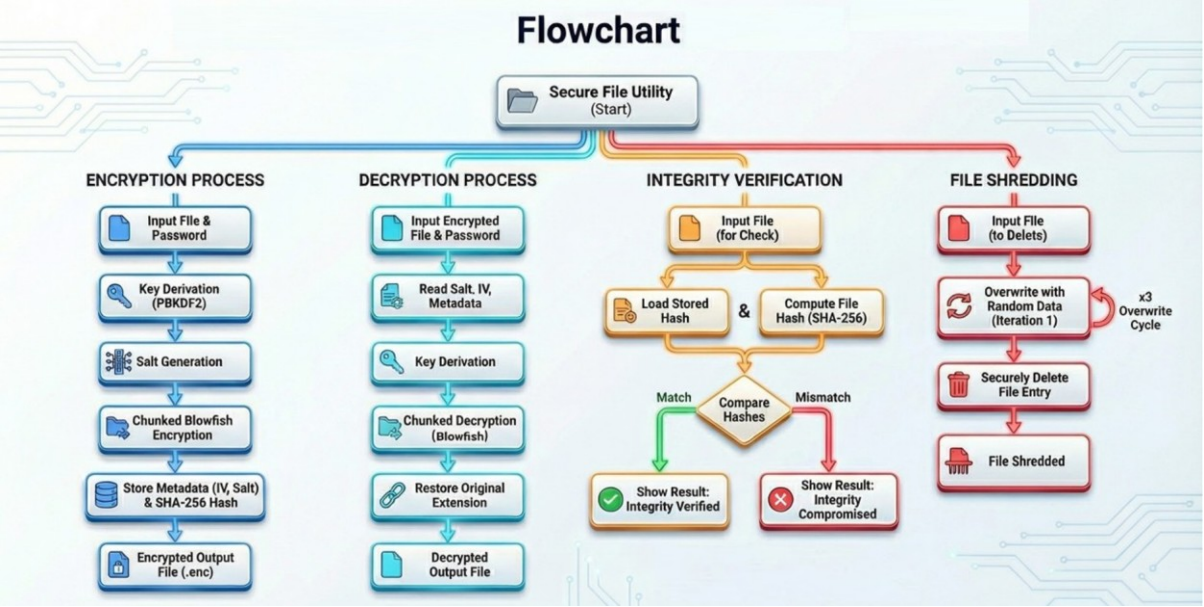
Methodology and Results

1. **Methods / Technology Used :**
    - **Cryptography :** Blowfish symmetric-key algorithm
    - **Hashing :** SHA-based cryptographic hashing
    - **Encryption Approach :** File-based binary encryption
2. **Tools / Software Used :**
    - Python 3.x
    - Tkinter (for GUI)
    - PyCryptodome library (for cryptographic operations)
3. **Project Architecture :**
    a) **GUI Layer**
        - Handles user interaction
        - File selection and input validation
        - Displays operation status

    b) **Core Cryptographic Layer**
        - Implements Blowfish encryption and decryption
        - Handles padding and binary data processing
        - Performs hashing and integrity checks

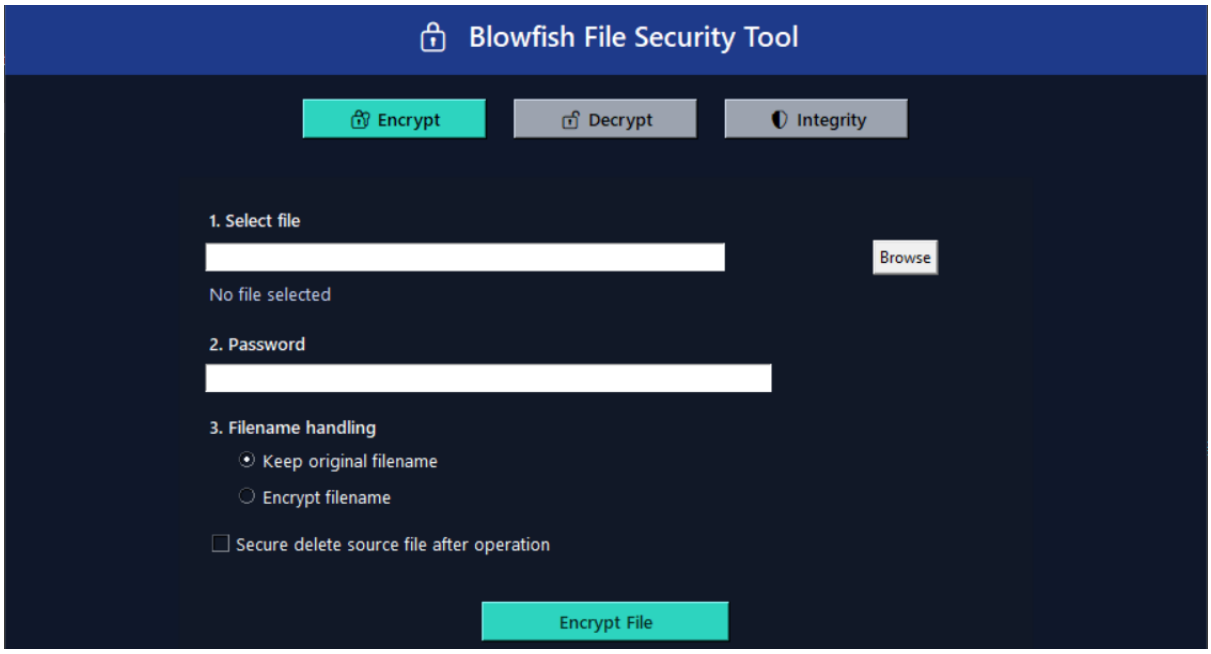
    c) **Utility Layer**
        - Secure file deletion using overwrite methods

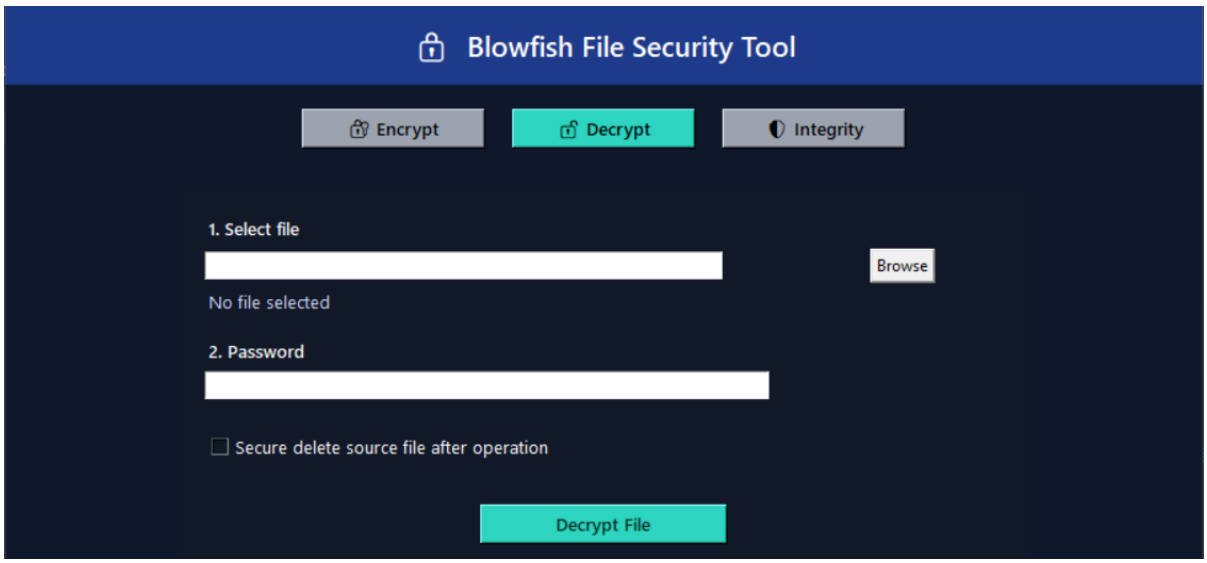
    This separation improves maintainability, debugging, and scalability.

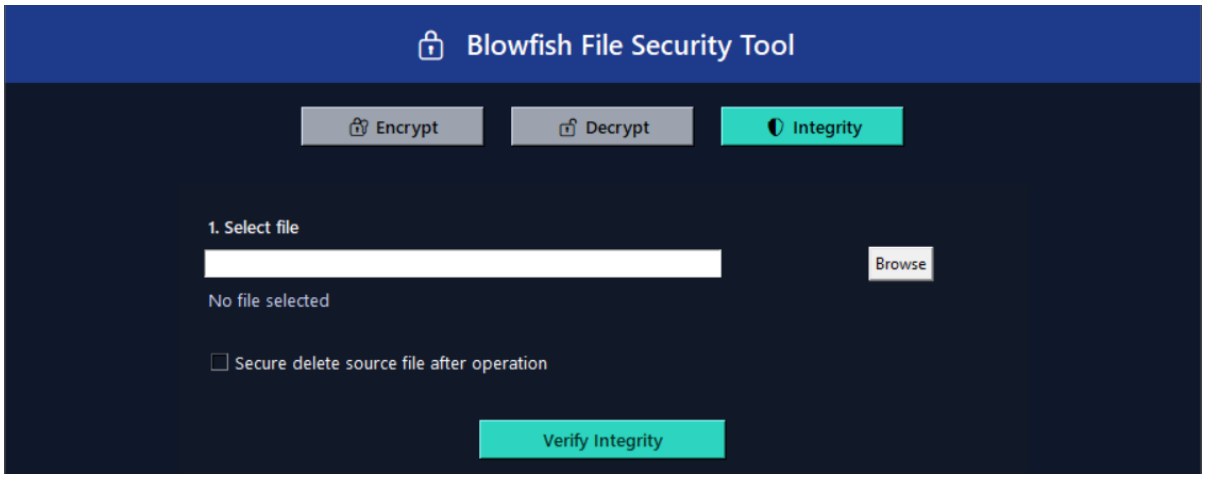


5. Final Project Working :

- GUI allows users to select files and choose encryption or decryption
- Users provide a secret key for cryptographic operations
- Encrypted files are generated securely
- Integrity verification ensures file authenticity
- Secure deletion prevents recovery of sensitive files




Blowfish File Security Tool
Encrypt
Decrypt
Integrity
1. Select file
Browse
No file selected
2. Password
Secure delete source file after operation
Decrypt File


Blowfish File Security Tool
Encrypt
Decrypt
Integrity
1. Select file
Browse
No file selected
Secure delete source file after operation
Verify Integrity

Learning and Reflection

Technical Learnings :

- **Practical understanding of symmetric key cryptography, specifically Blowfish**
- **File-level encryption and decryption techniques**
- **Binary-safe file handling and padding mechanisms**
- **Implementation of cryptographic hashing for integrity verification**
- **Secure deletion techniques using file overwrite strategies**
- **GUI development using Tkinter**
- **Modular programming and separation of concerns**
- **Debugging and performance analysis**

Management and Professional Learnings :

- **Team coordination and task allocation**
- **Time management and milestone-based development**
- **Problem-solving through iterative testing**
- **Documentation and reporting standards**
- **Use of Git and GitHub for version control and collaboration**

Conclusion and Future Scope

Future Scope of the Project :

- Implementation of multi-threaded and parallel encryption to improve performance
- Optimization for large file sizes and high-speed processing
- Advanced key management and key-strength validation
- Support for additional encryption algorithms (AES, RSA, etc.)
- Cloud storage integration for secure uploads and downloads
- Packaging the application as a standalone executable
- Logging, auditing, and user authentication features

Conclusion :

In conclusion, the Blowfish Secure File Encryption Tool is a successful demonstration of applied cryptography and secure software development. The project highlights the importance of data protection in modern computing environments and provides a strong foundation for future innovation. With further enhancements, the system can evolve into a robust, industry-ready security solution.