DATA STRUCTURES CSE228 INITIAL PROJECT REPORT

Topic: File Encryption/Decryption Tool

Submitted By:

Aman Deep Yadav

K22UR

Roll.No: 01

Submitted To:

Shubham Sharma

UID: 64339

File Encryption/Decryption Tool Report

Introduction

In today's digital age, securing sensitive information is of paramount importance. To ensure the confidentiality and integrity of data, encryption is a fundamental security measure. This report outlines the development and functionality of a console application, the File Encryption/Decryption Tool, which empowers users to encrypt and decrypt files using various encryption algorithms.

Purpose

The primary purpose of the File Encryption/Decryption Tool is to provide a user-friendly and versatile solution for protecting sensitive data stored in files. This tool aims to meet the following objectives:

1. Enable users to encrypt files with different encryption algorithms.
2. Provide the ability to decrypt encrypted files.

3. Offer a command-line interface for user convenience.
4. Enhance data security through strong encryption methods.

Features

1. Encryption Algorithms
   The File Encryption/Decryption Tool offers a selection of encryption algorithms, including:

   AES (Advanced Encryption Standard): A widely accepted symmetric encryption algorithm known for its strong security.
   RSA (Rivest–Shamir–Adleman): A popular asymmetric encryption algorithm that uses a pair of keys for encryption and decryption.
   DES (Data Encryption Standard): A symmetric encryption algorithm that was widely used in the past.

2. Command-Line Interface
   The application provides a command-line interface, making it accessible for users who prefer to interact with the tool through their terminals. Users can specify the encryption algorithm, the file to encrypt/decrypt, and the destination for the output file.

3. File Selection
   Users can specify the file they wish to encrypt or decrypt, and the application ensures that the operation is performed securely.

4. Key Management

For asymmetric encryption algorithms like RSA, the application manages the generation and storage of keys securely.

Approach

The development and implementation of the File Encryption/Decryption Tool followed a structured approach to ensure its effectiveness and security:

1. Requirements Analysis
   The project began with a thorough analysis of user requirements, including the need for multiple encryption algorithms, command-line interface, and secure key management. Clear and well-defined requirements were established to guide the development process.

2. Algorithm Selection
   The choice of encryption algorithms was critical. The development team selected widely recognized and strong algorithms, such as AES, RSA, and DES, to ensure robust data security.

3. Secure Key Management
   For asymmetric encryption (RSA), the tool incorporates secure key generation and management. Keys are generated using strong randomization methods and stored in a manner that prevents unauthorized access.

4. Development and Testing
   The development team created the tool, implementing the chosen algorithms, the command-line interface, and other features. Extensive testing was conducted to identify and rectify any vulnerabilities or issues.

5. Security Measures
   Security was a top priority throughout the development process. The tool incorporates secure password protection, strong random number generation, error handling, and thorough algorithm testing to enhance data security.

6. Future-Proofing
   The development process considered future enhancements, including the potential for supporting additional encryption algorithms, a graphical user interface (GUI), cloud integration, and secure key storage solutions.

Usage

The File Encryption/Decryption Tool is designed to be user-friendly and easy to use. Users can interact with the application through the command line by following these basic commands:

To encrypt a file: encrypt -algorithm <algorithm> -input <input_file> -output <output_file>

To decrypt a file: decrypt -algorithm <algorithm> -input <input_file> -output <output_file>

The user must specify the encryption algorithm, input file, and output file. For asymmetric encryption (RSA), the tool handles key generation and management.

Conclusion

The File Encryption/Decryption Tool is a versatile and user-friendly solution for protecting sensitive data through encryption. It provides users with a command-line interface, support for multiple encryption algorithms, and robust security measures. In an age where data security is paramount, this tool is a valuable addition to any user's digital security arsenal.

By continuously improving and expanding its capabilities, the File Encryption/Decryption Tool will continue to meet the ever-evolving security needs of users in an increasingly digital world.