Software Requirements Specification (SRS)

# Project Title: Secure Storage and Controlled Access of Files within a Directory

Version: 1.0

Date: 22-05-2025

Developed by (Team – 7), 2nd Semester, 'P' Section:  
- Yadunandana Reddy M  
- Yashas Sadananda  
- Vidushi Singh  
- Yashwanth Aditya

# 1. Introduction

## 1.1 Purpose

This project simulates a secure digital vault that uses puzzles and delays to create an illusion of security. It does not offer real protection or encryption and is intended as a novelty or satirical tool.  
  
The system includes:  
- Password-based authentication  
- Identity verification through security questions  
- Puzzle challenge as an added authentication layer  
- File listing and viewing within a secure directory  
  
Limitations:  
- No encryption  
- No multi-user support  
- No graphical user interface (GUI)  
  
Note: Configuration data, including authentication credentials, is stored in a configuration file.

## 1.2 Definitions, Acronyms, and Abbreviations

|  |  |
| --- | --- |
| Term | Definition |
| SRS | Software Requirements Specification |
| Vault | Protected storage system for files |
| CLI | Command Line Interface |
| Puzzle Authentication | Challenge-based access mechanism |

# 2. Overall Description

## 2.1 Product Perspective

The Secure Vault is a standalone, CLI-based application with no external dependencies (e.g., databases or network access).

## 2.2 User Classes and Characteristics

|  |  |
| --- | --- |
| User Type | Description |
| General User | Basic knowledge of file systems and command-line usage |

## 2.3 Operating Environment

- OS: Windows 10 or higher  
- Language: C  
- Compiler: GCC (via MinGW / Code::Blocks)

## 2.4 Design and Implementation Constraints

- Windows-only support  
- Requires password, security questions, and puzzle for access  
- Files must reside within the 'vault\_files' directory  
- Authentication and configuration data are stored in a configuration file

## 2.5 Assumptions and Dependencies

- Users should not manually rename or move files in the vault  
- Single-user environment is assumed

# 3. Specific Requirements

## 3.1 Functional Requirements

* FR1: System prompts user for authentication
* FR2: Files are stored securely in the vault
* FR3: Puzzle must be solved before access
* FR4: Security questions verify identity
* FR5: Lists available files in the vault
* FR6: Allows viewing of file content
* FR7: Maintains configuration settings in a file

## 3.2 Non-Functional Requirements

* NFR1: Validates user input at every step
* NFR2: Responds to commands within 1 second
* NFR3: Runs on systems with ≤ 4GB RAM
* NFR4: Masks sensitive inputs (planned enhancement)

## 3.3 External Interface Requirements

- User Interface: CLI-based  
- Hardware Interface: Keyboard and monitor  
- Software Interface: Windows OS and file system

# 4. System Features

|  |  |
| --- | --- |
| Feature | Description |
| Multi-step Authentication | Password, puzzle, and security question verification, with ability to reset credentials |
| File Listing | Displays vault file list |
| File Viewing | Opens and displays file content |
| Configuration Management | Securely stores credentials in a config file |

# 5. Appendices

## Appendix A: Future Enhancements

* Hashed password storage
* File encryption
* GUI support

## Appendix B: Sample Output

Welcome to the Secure Vault!  
1. Unlock Vault  
2. Reset Vault  
3. Exit

# 6. Team Contributions

Member 1 - Name: Yadunandana Reddy M SRN : PES2UG24CS605

**Contribution:**

Took primary responsibility for structuring and integrating the complete codebase. Played a key role in coordinating individual modules, particularly contributing to the list\_vault\_files\_and\_select and unlock functions, and ensured the final solution was cohesive and fully functional.

Member 2 - Name: Yashas Sadananda SRN : PES2UG24CS610

Contribution:

Focused on developing robust file handling mechanisms within the vault system, including file content display, the file menu interface, configuration file management, and general file access logic. Ensured secure, stable, and user-friendly operations for listing and interacting with stored vault files.

Member 3 - Name: Vidushi Singh SRN : PES2UG24CS581

Contribution:

Developed the command-line interface and implemented the security authentication flow including password verification, puzzle challenge, and security questions.

Member 4 - Name: Yashwanth Aditya SRN : PES2UG24AM190

Contribution:

Handled the password setup, reset functionality, and managing configuration file interactions for storing and retrieving vault credentials.

# 7. References

Source Code Repository: https://github.com/yxshas565/Secure-Digital-Vault