International Journal of Computer Science Engineering Techniques (IJCSE) Volume 10 Issue 2 ,Mar-Apr 2026

ISSN : 2455-135X

https://www.ijcsejournal.org/ *Type of Article (Review Article)*

A LIGHTWEIGHT PRIVACY-PRESERVING FILE PROTECTION SYSTEM FRAMEWORK FOR SECURE DATA STORAGE

N. Sakthivelu[1], Dr. S. Saraswathi[2]

[1]UG Student

Department of Computer Science and Data Science,

Nehru Arts and Science College, Coimbatore, Tamilnadu, India.

[2]Associate Professor,

Department of Computer Science and Data Science,

Nehru Arts and Science College, Coimbatore,Tamilnadu, India.

[1]sakthivelun655@gmail.com, [2]saraswathisubbian@gmail.com.

ABSTRACT:

In recent years, the rapid growth of digital storage has increased the demand for reliable and practical file protection mechanisms. Individuals and organizations frequently store sensitive information such as personal documents, images, academic records, and confidential files on local systems, where they remain vulnerable to unauthorized access, accidental modification, and data misuse. Although several security solutions are available, many require advanced technical expertise or depend heavily on cloud-based services, which may introduce additional privacy concerns. To address these challenges, this paper proposes a Lightweight Privacy-Preserving File Protection Framework for Secure Data Storage. The proposed system is designed to provide a simple, efficient, and locally operated security solution that ensures data confidentiality without adding unnecessary system complexity. The framework integrates user authentication and password-based symmetric encryption to securely encrypt and decrypt files and images. All operations are executed within the local environment, eliminating reliance on third-party servers and enhancing user privacy. Furthermore, the system maintains a local encryption activity log to record file protection operations, thereby improving transparency and user awareness. The overall framework achieves a balanced integration of confidentiality, controlled access, usability, and computational efficiency. The proposed solution offers a practical and user-friendly approach to secure local file storage and establishes a foundation for future enhancements such as advanced access control and secure file-sharing capabilities.

KEYWORDS:

File Encryption, Data Confidentiality, User Authentication, Cryptographic Security, Secure Storage Framework, Privacy-Preserving System.

1.INTRODUCTION:

In today's digital world, the use of electronic storage has become an essential part of daily life. Individuals and organizations store a large amount of sensitive information such as personal documents, images, academic records, and confidential files on their computers and local devices. While digital storage offers convenience and

ISSN : 2455-135X

https://www.ijcsejournal.org/ *Type of Article* (Review Article)

efficiency, it also creates serious security concerns. Without proper protection, important files can be accessed, modified, or misused by unauthorized users. Cryptographic techniques are widely recognized as effective mechanisms for protecting digital data and ensuring confidentiality [1].Many file protection methods are available to secure digital data. Some systems rely on simple password protection, while others use advanced encryption mechanisms based on established cryptographic principles [2], [3]. However, basic methods often fail to provide strong data confidentiality, and advanced solutions may be too complex for general users to understand and operate. In addition, certain systems depend on external cloud services, which may introduce additional privacy risks and reduce user control over personal data [6].To overcome these challenges, this paper proposes a Lightweight Privacy-Preserving File Protection System Framework for Secure Data Storage. The main goal of the proposed system is to provide a secure yet easy-to-use file protection solution that operates locally. The framework allows authenticated users to encrypt and decrypt files using password-based symmetric encryption techniques derived from standard cryptographic practices [3], [5]. It also maintains an encryption history to help users monitor file protection activities. By combining data confidentiality, controlled access, and user-friendly design, the proposed system aims to deliver a practical and reliable solution for secure file storage.

2.MOTIVATIONAL RESEARCH:

The rapid expansion of digital technologies has significantly transformed the way individuals and organizations store and manage information. Today, most personal documents, academic records, project files, multimedia content, and confidential data are stored in digital form on local computers and portable storage devices. While this shift toward digital storage has improved accessibility and efficiency, it has also introduced serious security challenges. Unauthorized access, accidental data exposure, and privacy breaches have become common concerns in everyday computing environments. Many users rely on simple folder locks or operating system-based password protection to secure their files. Although these mechanisms provide a basic level of protection, they often lack strong encryption capabilities and do not guarantee complete confidentiality. In several cases, users are unaware of how securely their files are actually protected. On the other hand, advanced security systems available in the market may require technical knowledge, complex configurations, or dependence on external cloud services. Such solutions may not be practical for students and general users who require simple yet effective protection methods. Another motivating factor behind this research is the increasing awareness of data privacy. Users are becoming more conscious about how and where their data is stored. Cloud-based storage solutions, while convenient, may raise concerns regarding third-party access, data leakage, and lack of user control. This highlights the need for a lightweight and locally operated security framework that allows users to maintain complete control over their files without relying on external servers. The proposed Lightweight Privacy-Preserving File Protection System Framework is motivated by the need to bridge the gap between strong security and ease of use. The aim is to design a system that combines password-based encryption, controlled access, and transparency through encryption history tracking, while maintaining simplicity in implementation and operation. By focusing on local encryption and user-friendly design, this research attempts to provide a practical and accessible solution for secure data storage in everyday computing environments.

ISSN : 2455-135X

https://www.ijcsejournal.org/ *Type of Article* (Review Article)

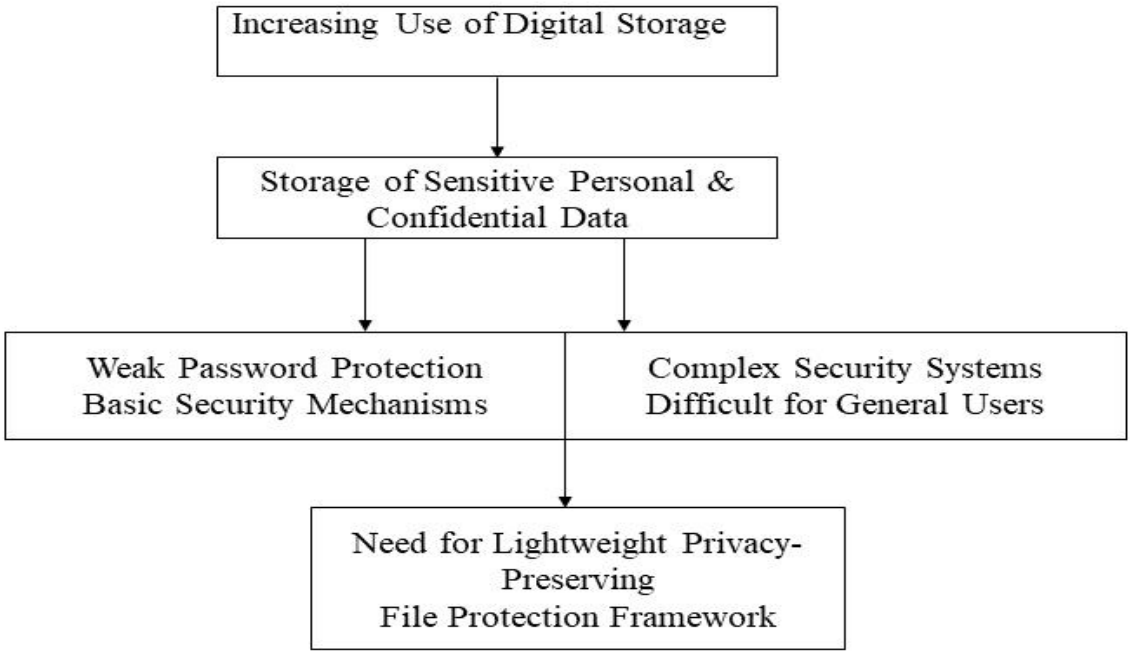

Figure 1: Motivation Behind the Proposed Framework

3.PROBLEM STATEMENT:

The rapid growth of digital storage has made it common for individuals and institutions to store large volumes of sensitive information on local computing systems. Personal documents, academic records, confidential files, multimedia content, and project data are frequently stored on laptops, desktops, and external storage devices. While this digital transformation has improved convenience and accessibility, it has also increased the risk of unauthorized access, data breaches, and privacy violations [1]. Many users rely on basic password protection mechanisms or simple folder-locking tools provided by operating systems. Although these methods offer a minimal level of security, they do not guarantee strong data confidentiality. In certain cases, files may still be exposed due to system vulnerabilities, weak password practices, or improper configurations [2]. Furthermore, most basic protection techniques do not provide encryption-based security or activity monitoring features, limiting user awareness of file protection operations. On the other hand, advanced security solutions often require complex configuration, technical expertise, or dependence on cloud-based infrastructure. While cloud platforms offer accessibility and scalability, they may introduce concerns related to third-party access, data leakage, and reduced control over personal information [3]. Many users prefer a secure solution that operates independently within a local environment. Another critical issue is the lack of transparency in file security processes. Without proper monitoring mechanisms, users have limited visibility into encryption and decryption activities, reducing accountability and operational awareness. Therefore, there is a clear need to design a lightweight, privacy-preserving file protection framework that integrates strong encryption, controlled access, local data processing, and activity tracking in a simple and user-friendly architecture. This research addresses the challenge of developing a secure file protection system that ensures confidentiality while remaining practical and accessible for everyday users.

4. RESEARCH CONTRIBUTION:

This research aims to develop a practical and efficient file protection framework that ensures data confidentiality

ISSN : 2455-135X

https://www.ijcsejournal.org/ *Type of Article* (Review Article)

while maintaining simplicity in design and operation. The focus of this work is not only on implementing encryption but also on creating a balanced system that integrates privacy, usability, and transparency. The major contributions of this research are outlined as follows:

4.1 Development of a Lightweight Privacy-Focused Framework:

The primary contribution of this research is the design and implementation of a lightweight file protection system that operates entirely within a local computing environment. Unlike many modern security solutions that rely on cloud-based infrastructure, the proposed framework ensures that all encryption and decryption processes occur locally. This approach strengthens user privacy by preventing unnecessary exposure of sensitive data to external servers while maintaining full user control over stored files.

4.2 Implementation of Secure Password-Based Encryption:

Another key contribution of this work is the integration of a secure password-based encryption and decryption mechanism. The system converts original files into encrypted formats that can only be accessed using the correct password. This ensures confidentiality and protects data from unauthorized access. The encryption module is designed to be effective while remaining simple enough for practical everyday use.

4.3 Integration of Authentication and Activity Logging:

To enhance security and transparency, the proposed framework includes a user authentication mechanism that restricts access to the application. In addition, an encryption history feature has been incorporated to record file protection activities. This logging mechanism provides users with visibility into encryption and decryption operations, improving accountability and overall system reliability.

4.4 Development of a Standalone User-Friendly Application:

The final contribution of this research is the creation of a standalone executable application with a simple graphical user interface. The system is designed to be accessible to general users without requiring advanced technical knowledge. By combining security features with ease of use, the framework provides a practical solution for secure file management.

5.LITERATURE REVIEW:

The protection of digital files has been an active area of research due to the rapid growth of data storage across personal and institutional computing systems. As individuals increasingly rely on digital platforms to store personal documents, academic records, images, and confidential files, the need for effective file protection mechanisms has become more critical. Researchers have proposed various approaches to secure digital data, ranging from basic password protection techniques to advanced cryptographic frameworks designed to ensure confidentiality and integrity [1], [2]. One of the most widely studied approaches in file security is password-based encryption. Several studies emphasize the role of symmetric encryption algorithms in protecting locally stored data [1], [3]. These methods typically rely on user-defined passwords to generate encryption keys that convert readable files into encrypted formats. While such systems are relatively simple and efficient, research indicates

ISSN : 2455-135X

https://www.ijcsejournal.org/ *Type of Article* (Review Article)

that weak password policies and improper key management can reduce overall security strength [2]. Many existing implementations focus primarily on encryption robustness but do not adequately address usability and monitoring aspects.In addition to password-based systems, modern research has explored privacy-preserving storage frameworks aimed at enhancing data protection while maintaining user control. Cloud-based encryption models have gained attention due to their scalability and accessibility. However, multiple studies highlight concerns related to third-party data exposure, unauthorized server access, and dependency on external service providers [4]. These concerns have motivated researchers to investigate locally operated encryption systems that preserve full data ownership while minimizing external vulnerabilities. Another important aspect discussed in previous research is authentication and access control. Authentication mechanisms are widely implemented to restrict unauthorized entry into secure systems [2]. Although advanced authentication models improve security, they may introduce complexity that is not always suitable for general users. Research suggests the need for simplified authentication approaches that maintain strong protection without increasing operational difficulty. Furthermore, recent studies emphasize the importance of transparency and activity monitoring in security systems. Logging mechanisms and audit trails enable users to track encryption and decryption operations, thereby enhancing accountability. However, many lightweight file protection tools lack integrated monitoring capabilities, focusing primarily on encryption without offering operational visibility. From the existing body of research, it is evident that encryption techniques and privacy-preserving storage models are well established. Nevertheless, there remains a gap in developing a lightweight, locally operated file protection framework that integrates encryption, authentication, and activity monitoring within a simple and user-friendly architecture. The present research attempts to address this gap by proposing a secure file protection system that balances data confidentiality, user control, and practical usability.

6.METHODOLOGY:

The proposed Lightweight Privacy-Preserving File Protection System is designed as a locally operated security framework that integrates authentication, encryption, decryption, and activity logging within a single application. The methodology focuses on structured implementation, secure data handling, and controlled access mechanisms. The system workflow is divided into multiple interconnected modules, which are explained below.

6.1 Overall System Architecture:

The system architecture defines the interaction between the user interface, authentication module, encryption module, and storage components. The framework follows a modular design to ensure maintainability and security separation between components. The process begins when a user launches the application and provides valid authentication credentials. Upon successful login, the user is redirected to the dashboard interface, where encryption and decryption operations can be performed. When a file is selected for encryption, the system applies password-based encryption techniques to convert the original file into an encrypted format. The encrypted file is then securely stored in the local system. Similarly, decryption restores the file only when the correct password is provided. All operations are recorded in an encryption history log for monitoring purposes.

ISSN : 2455-135X

https://www.ijcsejournal.org/ *Type of Article* (Review Article)

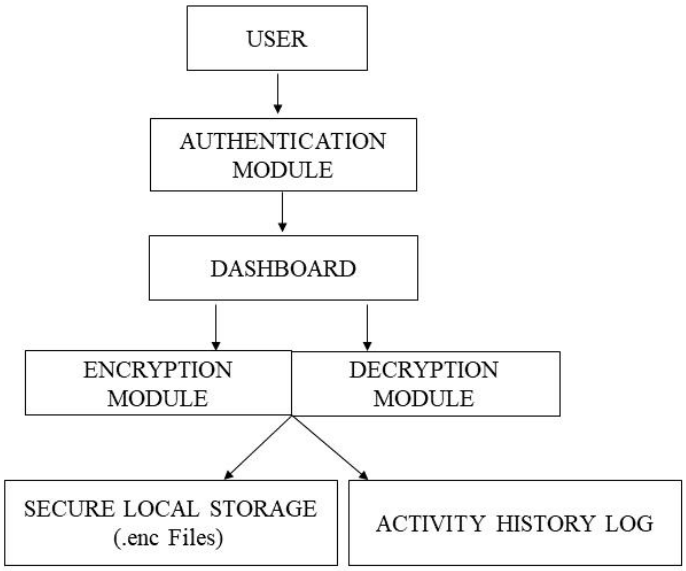

Figure 2: The proposed system uses a modular structure integrating authentication, encryption, and logging for secure file handling.

6.2 Encryption And Decryption Mechanism:

The encryption and decryption mechanism forms the core functionality of the proposed system. The framework implements a password-based encryption approach to ensure data confidentiality and secure file handling. When a user selects a file and provides a password, the system processes the file content through a cryptographic transformation, converting it into an encrypted format. The resulting file is stored with a dedicated extension (.enc), which restricts direct access and protects the data from unauthorized use. During the decryption process, the user must enter the same password that was originally used for encryption. The system validates the password before reconstructing the original file content. If the password does not match, the decryption process is denied. This verification mechanism ensures controlled access and prevents misuse of protected information. The entire encryption and decryption operations are performed locally, without requiring internet connectivity. This design enhances privacy, reduces exposure to external threats, and minimizes dependency on third-party services. Furthermore, the system is structured to avoid unintended overwriting of the original file unless explicitly specified by the user.

ISSN : 2455-135X

https://www.ijcsejournal.org/ *Type of Article* (Review Article)

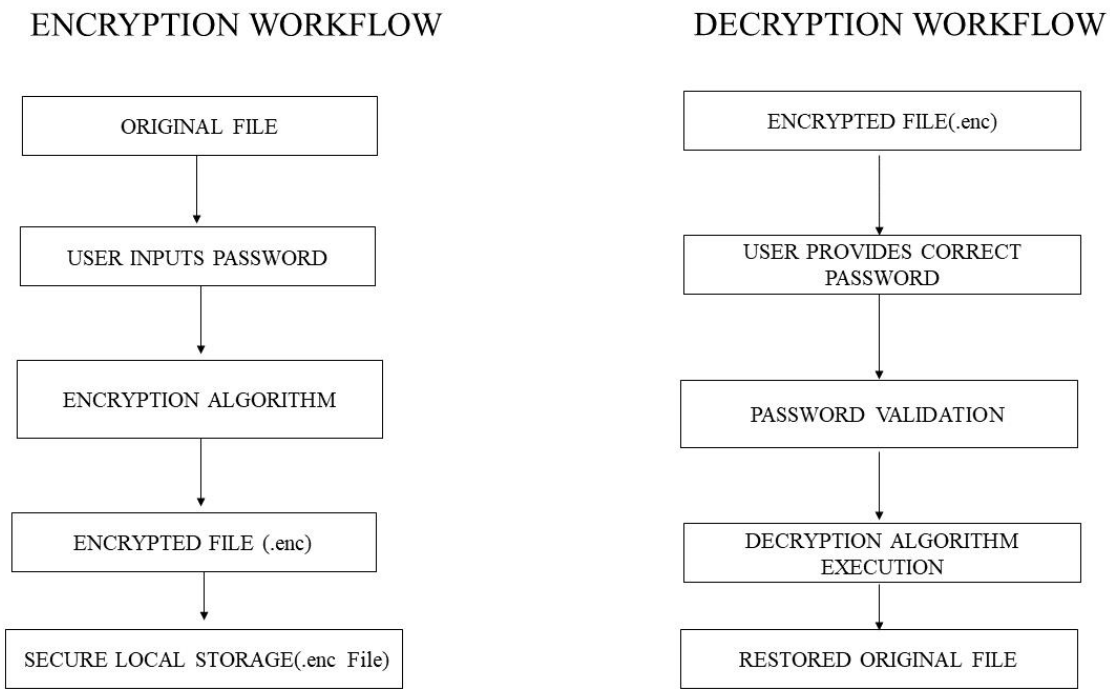

Figure 3: shows the encryption and decryption workflow of the proposed system.

6.3 Authentication and Access Control:

Authentication and access control play a crucial role in ensuring system security. The proposed system implements a login-based authentication mechanism to restrict unauthorized access. Users must provide valid credentials before accessing the dashboard and file handling features. The authentication module verifies the entered username and password against stored records. Only authenticated users are permitted to perform encryption and decryption operations. If invalid credentials are entered, access is denied and the user is prompted to re-enter correct details.

This controlled access mechanism ensures that only authorized users can interact with the system, thereby enhancing data confidentiality and preventing misuse.

6.4 Algorithmic Representation:

The proposed system implements a password-based symmetric encryption mechanism. The encryption algorithm transforms the original file content into an unreadable encrypted format using a derived cryptographic key generated from the user-provided password. The system utilizes symmetric encryption principles, where the same password-derived key is used for both encryption and decryption operations.

1.Encryption Algorithm Steps:

1.Input the original file.

2.Accept password from the user.

3.Generate a secure encryption key from the password.

4.Convert file content into encrypted data using the cryptographic algorithm.

5.Save the encrypted output with a ".enc" extension.

6.Record the activity in the log file.

ISSN : 2455-135X

https://www.ijcsejournal.org/ *Type of Article* (Review Article)

2.Decryption Algorithm Steps:

1. Input the encrypted file (.enc).
2. Accept password from the user.
3. Generate key using the same password.
4. Validate the password.
5. Decrypt the file content.
6. Restore the original file.
7. Update the activity log.

6.5 Mathematical Representation:

The encryption and decryption operations of the proposed system can be represented mathematically using symmetric encryption principles.

Let:

- **F** represent the original file content.
- **P** represent the user-provided password.
- **K** represent the cryptographic key derived from the password.
- **E(.)** represent the encryption function.
- **D(.)** represent the decryption function.
- **C(.)** represent the encrypted file content.

The key generation process is defined as:

K = f(P)

Where f(.) denotes the key derivation function applied to the password.

The encryption process is expressed as:

C =E(F,K)

The decryption process is expressed as:

F =D(C,K)

Successful decryption is possible only when the same key **K**, derived from the original password, is provided. This mathematical representation confirms that the system follows symmetric encryption principles for secure file protection.

7.IMPLEMENTATION:

This section presents the practical implementation of the proposed Privacy-Preserving Secure File Protection System. The system was developed by integrating authentication, encryption, decryption, and activity logging modules into a unified framework. The implementation focuses on secure local file processing, controlled user access, and maintaining activity records to ensure confidentiality and transparency.

7.1 Development Environment

The proposed system was implemented using the Python programming language. The graphical user interface (GUI) was developed using the Tkinter library to provide a user-friendly interaction platform. Cryptographic

ISSN : 2455-135X

https://www.ijcsejournal.org/ *Type of Article* (Review Article)

operations were performed using the Cryptography library to ensure secure encryption and decryption. The system was developed and tested in a Windows-based environment.

7.2 System Modules

The implementation is divided into the following core modules:

1.Authentication Module:

Manages user login verification and restricts unauthorized access to the system.

2.Encryption Module:

Processes selected files using password-based encryption and generates encrypted files with a dedicated extension (.enc).

3.Decryption Module:

Validates user-provided passwords and restores the original file content upon successful verification.

4.Activity Logging Module:

Records encryption and decryption operations to maintain transparency and track system usage.

7.3 File Handling Mechanism:

The system processes files locally without internet dependency. During encryption, the selected file is transformed into an encrypted format and stored securely in local storage. During decryption, the encrypted file is restored to its original format only after password validation. This local processing approach enhances user privacy and reduces exposure to external security risks.

8.SECURITY ANALYSIS:

The security of the proposed system is built on three main aspects: authentication control, password-based encryption, and local file processing. Together, these mechanisms ensure that sensitive files remain protected from unauthorized access. The authentication module restricts access to the system by verifying user credentials before granting permission to use encryption or decryption features. This prevents unauthorized individuals from interacting with protected files. The encryption mechanism uses a password-based symmetric approach. A cryptographic key is generated from the password provided by the user and is used to convert the original file into an unreadable encrypted format. Since the same password-derived key is required for decryption, the file cannot be restored without the correct credentials. All operations are performed locally without relying on internet connectivity or external servers. This reduces exposure to network-based threats and preserves user privacy. Additionally, the system maintains an activity log that records encryption and decryption actions, improving transparency without compromising file content. Overall, the proposed system ensures confidentiality, controlled access, and secure file handling within a self-contained environment.

9.SYSTEM ANALYSIS:

The proposed framework follows a modular design in which each component performs a specific function. The authentication module manages user verification, the encryption and decryption modules handle secure file transformation, and the logging module records system activities. The system operates efficiently in a local environment and does not depend on external infrastructure. This reduces processing delays and enhances reliability. The modular structure also makes the system easier to maintain and extend in the future. From a

ISSN : 2455-135X

https://www.ijcsejournal.org/ *Type of Article* (Review Article)

functional perspective, the system ensures that files are encrypted securely and can only be restored using the correct password. This structured design supports secure, organized, and user-friendly file protection.

10.ADVANTAGES OF THE PROPOSED SYSTEM:

1.Provides secure password-based file protection.

2.Prevents unauthorized access through authentication.

3.Operates without internet dependency.

4.Maintains encryption and decryption history.

5.Simple and easy-to-use interface.

6.Modular and extendable design.

11.RESULTS AND DISCUSSION:

This section presents the experimental results obtained from the implementation of the proposed lightweight privacy-preserving file protection framework. The system was tested under local execution conditions to evaluate authentication control, encryption accuracy, decryption correctness, and activity logging functionality. The objective of the evaluation was to verify secure file transformation and controlled access without relying on external network infrastructure. The implemented prototype, titled “Privacy-Preserving Secure File Protection System,” represents the working application of the proposed lightweight framework. The system demonstrated stable and consistent performance during testing. The authentication module successfully restricted access to authorized users. Only valid credentials allowed access to the dashboard interface, ensuring controlled system usage. Unauthorized attempts were properly rejected. During encryption, selected files were successfully converted into protected encrypted formats with a dedicated “.enc File” extension. The encryption process prevented unauthorized access to file content without the correct password-derived key. In the decryption phase, the system validated the user-provided password before restoring the original file. Incorrect password attempts did not reveal any file data, confirming the effectiveness of the symmetric encryption mechanism. The activity history log accurately recorded encryption and decryption events, improving operational transparency while preserving data confidentiality. Since all processes are executed locally, the framework eliminates risks associated with cloud-based data transmission and external exposure. Overall, the experimental results confirm that the proposed lightweight framework achieves secure file protection, controlled access management, and reliable local data handling in a practical and user-friendly manner.

1.Login Interface:

The system begins with a secure authentication interface that verifies user credentials before granting access.

ISSN : 2455-135X

https://www.ijcsejournal.org/ *Type of Article* (Review Article)

Figure 4: User authentication interface of the implemented of the proposed lightweight framework

The authentication module serves as the first security layer of the system. It validates user credentials before granting access to the dashboard. The system restricts unauthorized login attempts and ensures that only registered users can proceed further. This controlled access mechanism enhances overall system security and prevents misuse of protected file operations.

2.Dashboard and File Processing:

After successful authentication, the user is redirected to the main dashboard containing encryption and decryptionoptions.

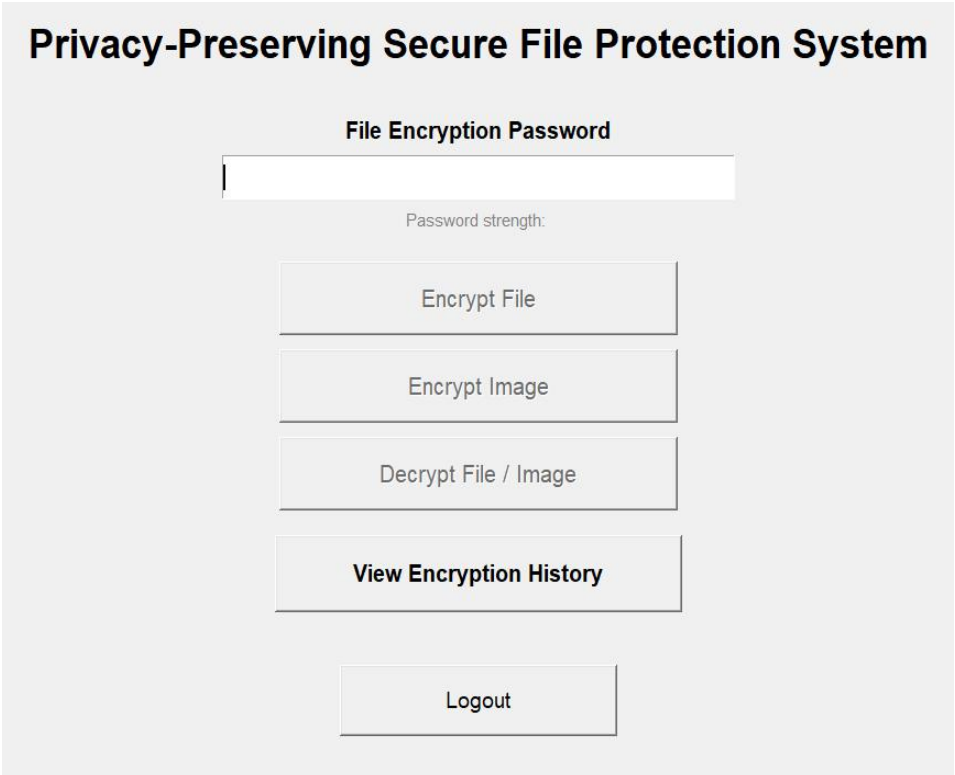

Figure 5: Main dashboard showing encryption and decryption functionalities.

ISSN : 2455-135X

https://www.ijcsejournal.org/ *Type of Article* (Review Article)

After successful authentication, users are redirected to the dashboard, which contains the core functionalities of the framework. The interface allows users to select files, enter encryption passwords, and perform secure encryption or decryption operations. The design focuses on simplicity and usability while maintaining secure processing. The password-based symmetric encryption ensures that file access remains protected.

3.Activity History Log:

In addition to secure file processing, the framework maintains a local logging mechanism to record system activities.

Figure 6: Local Encryption and Decryption Activity History Log.

Figure 6 illustrates the activity history maintained by the system. It records encryption and decryption operations performed by authenticated users. The log stores only operation details without exposing sensitive file content, thereby ensuring transparency while preserving data confidentiality. Overall, the results validate that the proposed lightweight framework effectively integrates authentication, encryption, and logging mechanisms to ensure secure local file protection.

12. CONCLUSION:

This paper introduced a lightweight privacy-preserving file protection framework aimed at securing data in a local environment without relying on cloud services. The proposed system combines user authentication, password-based symmetric encryption, secure file transformation, and activity logging within a simple and modular structure. The implementation and testing confirm that the framework performs reliably under normal operating conditions. The authentication mechanism ensures that only authorized users can access encryption and decryption features. The encryption process protects files by converting them into secure encrypted formats using a password-derived key, while the decryption process restores the original file only when the correct password is provided. The inclusion of a local activity history log further improves transparency by recording operations

ISSN : 2455-135X

https://www.ijcsejournal.org/ *Type of Article* (Review Article)

without exposing sensitive file content. Overall, the results demonstrate that the framework provides a practical and efficient approach to secure local file protection. Its lightweight design makes it suitable for users who require privacy-focused data storage without complex infrastructure. In the future, the framework can be enhanced by integrating multi-factor authentication and role-based access control to further strengthen security. Performance improvements for handling larger files and the addition of secure backup features may also expand its applicability in real-world environments.

REFERENCES:

[1] W. Stallings, Cryptography and Network Security: Principles and Practice, 7th ed., Pearson Education, 2017.

[2] B. Schneier, Applied Cryptography: Protocols, Algorithms, and Source Code in C, 2nd ed., Wiley, 1996.

[3] A. J. Menezes, P. C. van Oorschot, and S. A. Vanstone, Handbook of Applied Cryptography, CRC Press, 1996.

[4] R. Anderson, Security Engineering: A Guide to Building Dependable Distributed Systems, 2nd ed., Wiley, 2008.

[5] C. Paar and J. Pelzl, Understanding Cryptography: A Textbook for Students and Practitioners, Springer, 2010.

[6] M. Bishop, Computer Security: Art and Science, Addison-Wesley, 2018.

[7] National Institute of Standards and Technology (NIST), "Recommendation for Key Management – Part 1: General," NIST Special Publication 800-57, 2016.

[8] W. Diffie and M. Hellman, "New Directions in Cryptography," IEEE Transactions on Information Theory, vol. 22, no. 6, pp. 644–654, 1976.

[9] A. Shamir, "Identity-Based Cryptosystems and Signature Schemes," Advances in Cryptology, Lecture Notes in Computer Science, vol. 196, Springer, 1985.

[10] D. Boneh and X. Boyen, "Short Signatures Without Random Oracles," Advances in Cryptology – EUROCRYPT, Springer, 2004.