2026 Jurnal Kejuruteraan, Teknologi dan Sains Sosial
Journal of Engineering, Technology and Social Science
Volume 12 Issue 2, e-ISSN: 27166848

Enhancing File Security Through Encryption: Development of a User-Friendly System with Threat Identification

Maznita Binti Mohamed Fikri[1] Sharifah Nur Bt Syed Ismail[2] and Azrul Junaidi bin Abd Aziz[3]
Information and Communication Technology, Polytechnic Ungku Omar, Ipoh, Malaysia
Maznita@puo.edu.my

ARTICLE INFO

Article history:
Received
31 July 2025
Received in revised form
22 October 2026
Accepted
04 April 2026
Published online
30 June 2026

Keywords:
Encryption; File security; Virus detection

ABSTRACT

In today's digital world, protecting personal and sensitive data is more critical than ever, as traditional security measures often fail to fully prevent unauthorized access or damage caused by malicious software such as viruses. The objective of this research is to enhance file security by developing a system that primarily uses encryption to safeguard files while also identifying threats that could harm or corrupt data. The method involves designing encryption and decryption algorithms to secure files and implementing virus detection techniques to identify various types of malwares. Additionally, a user-friendly graphical interface was developed to ensure accessibility and ease of use for all users. The system provides real-time notifications to alert users of any potential security breaches. Findings from system testing demonstrate that the solution effectively encrypts files, detects malicious threats promptly, and offers a straightforward user experience. This combination significantly improves data protection and helps maintain the integrity of users' digital files. In conclusion, the developed system provides a reliable and accessible approach to file security by combining robust encryption with proactive virus detection and notification, thereby enhancing user confidence and reducing the risk of data compromise

1. Introduction

In the digital age, data is one of the most valuable assets for individuals, organizations, and governments alike. The widespread use of information and communication technologies has led to a dramatic increase in the volume of digital files stored, processed, and transmitted across various platforms. However, this convenience also comes with significant risks. Files containing sensitive or personal information are frequently targeted by cyber threats such as malware, ransomware, and unauthorized access. Traditional security mechanisms, while useful, often fall short in effectively safeguarding data against evolving attack techniques.

Encryption has long been recognized as a fundamental method for securing data by converting it into a format that is unreadable without proper authorization. Despite its effectiveness, many encryption tools are either too complex for the average user or fail to incorporate real-time threat detection. Therefore, there is a growing need for a comprehensive solution that not only encrypts files but also identifies potential threats before they can cause damage.

This research focuses on the development of a user-friendly system that enhances file security through robust encryption techniques, combined with built-in threat identification capabilities. The aim is to create a practical and accessible tool that empowers users regardless of technical expertise to protect their files from unauthorized access and digital threats, thereby strengthening overall data resilience in an increasingly hostile cyber environment.

2. Materials and Methods

This research adopts the Agile Software Development Model to guide the design and implementation of a file security system that integrates encryption and threat identification. Agile was selected due to its flexibility, iterative nature, and strong emphasis on user involvement, making it well-suited for developing systems that require both functional security mechanisms and user-centred design.

The study began with the identification of key user requirements, collected through structured interviews and questionnaires with potential users. This approach ensured that the system was developed based on actual user needs and expectations. Requirements gathered were translated into development goals that shaped the system's functionality, including data encryption, threat detection, and usability considerations.

Development activities were conducted in iterative cycles, allowing incremental improvements and continuous refinement. Each iteration involved collaborative planning, system coding, testing, and feedback evaluation. By using this agile approach, the research ensured that early prototypes could be assessed and enhanced quickly, reducing the risk of design flaws and increasing system reliability.

To ensure the quality and effectiveness of the system, various testing methods were employed, including functional testing, user acceptance testing, and basic security validation. Feedback collected during testing was analysed and used to enhance the system's performance, especially in terms of ease of use and threat-handling capability.

In line with ethical research practices, all participants involved in the requirement-gathering and evaluation stages provided informed consent, and their data was handled with strict confidentiality.

The Agile methodology not only supported rapid prototyping and system evolution but also aligned with the research objective: to produce a secure, practical, and user-friendly file protection system that meets real-world demands.

3. Results

To evaluate the functionality, reliability, and usability of the developed file security system, a series of structured tests were conducted. These included the execution of the Testing Plan, Integration Testing, and User Acceptance Testing (UAT). Each testing phase served to validate different aspects of the system's performance and alignment with its design objectives.

3.1 Functional Testing Results

The functional testing phase focused on verifying that each feature performed as intended. The core modules tested included file encryption, decryption, malware detection, real-time notification, and the graphical user interface.

Function Tested	Expected Outcome	Actual Outcome	Status
File encryption	File is securely encrypted and unreadable	Achieved	Pass
File decryption	File is accurately restored to original state	Achieved	Pass
Malware detection	Detection of malicious test files	All threats successfully detected	Pass
Notification alert	Real-time popup on threat detection	Alerts displayed as expected	Pass

All functional requirements were met, indicating that the system performs consistently with the intended design specifications.

3.2 Integration Testing Results

Integration testing was conducted to ensure seamless communication between system components—specifically, how the encryption engine, malware detection module, and GUI interface interact.

Key test scenarios included:

Scenario 1: File encryption followed by malware scanning

Outcome: Encrypted files were successfully excluded from scanning to prevent false positives.

Scenario 2: Malware detection triggering notification

Outcome: Detected threats triggered real-time GUI alerts and updated system logs as expected.

2026 Jurnal Kejuruteraan, Teknologi dan Sains Sosial
Journal of Engineering, Technology and Social Science
Volume 12 Issue 2, e-ISSN: 27166848

Scenario 3: User command through GUI executing backend functions

Outcome: GUI commands effectively triggered backend processes without errors.

No integration failures were observed, confirming that the modules communicate and function harmoniously under typical use conditions.

4. User Acceptance Testing (UAT)

UAT was conducted with a sample group of 10 users, including non-technical users, to assess system usability and satisfaction. Participants were asked to perform key tasks: encrypting and decrypting files, scanning for malware, and responding to alerts.

Evaluation Criteria	User Feedback	Summary
Ease of use	90% rated as intuitive and easy to navigate	Ease of use
Interface design	80% found the layout clean and user-friendly	Interface design
System responsiveness	100% noted prompt system reaction	System responsiveness
Overall satisfaction	85% satisfied or highly satisfied	Overall satisfaction

Feedback confirms the system meets user expectations for both functionality and usability, with minor suggestions for enhancement.

5. Discussion

The system has been successfully developed and compiled into an executable (.exe) file, and it is now ready for evaluation. Overall, the project shows several practical strengths. One key advantage is its portability the file size is only 11.4MB, which makes it easy to transfer and install on different devices. The system is also completely free to use, which makes it accessible to a wide range of users, including those with limited budgets. In terms of security, it uses standard algorithms like SHA-256 and AES, which are widely trusted in the industry. This helps ensure that data remains safe and protected. The system also comes with a graphical user interface (GUI) that makes it easier to use, especially for non-technical users. Another benefit is that the software combines three features in one program, offering multiple functions without needing separate tools.

However, there are a few limitations to consider. The virus scanner feature only allows up to 800 scans because it relies on the free version of Cloudmersive's API. This might not be enough for users who need to scan files regularly. Also, the software is currently only available for Windows users, which limits its accessibility for those using macOS or Linux. An internet connection is required for the virus scanning function to work, which could be a problem in areas with poor connectivity. Additionally, while the scanner can detect viruses, it cannot remove them automatically users must delete infected files manually, which may not be easy for everyone. These limitations suggest areas for future improvement, such as supporting more platforms, increasing scan limits, enabling offline use, and adding automated virus removal. Despite these challenges, the system provides a solid foundation and shows promising potential for further development.

6. Conclusion

In summary, the developed software offers a practical and cost-effective solution for data protection, particularly suited for general users. Its lightweight and portable design ensures compatibility across a wide range of laptops and desktop computers, regardless of their generation. Positive feedback from users highlights its usability and efficiency. By integrating three essential functions into a single platform, the software not only simplifies the user experience but also saves time and effort otherwise spent searching for multiple tools. Moreover, offering this software free of charge makes it an accessible alternative to expensive commercial solutions, allowing users to manage their data security needs without financial burden. Based on these strengths, we confidently recommend this software as a reliable and user-friendly tool for everyday data protection.

7. Acknowledgements

This research project was made possible through the support and contributions of many individuals and parties. Sincere appreciation is extended to all who offered guidance, expertise, and encouragement throughout the course of this study.

Gratitude is also given to those who provided constructive feedback and took part in testing the developed system, helping to enhance its functionality and usability. Their involvement contributed significantly to the overall quality of the project.

Appreciation is also extended to the academic community and support staff who provided the resources and environment necessary for the successful completion of this research.
The encouragement and moral support received from various colleagues, peers, and supporters are also gratefully acknowledged.

8. References

Fatima, S., Rehman, T., Fatima, M., Khan, S., & Ali, M. A. (2022). Comparative analysis of AES and RSA algorithms for data security in cloud computing. Engineering Proceedings, 20(1), 14.

Sitraka, R. R., & Malalatiana, R. H. (2024). Evaluations of crypto-system AES using multiple block ciphering modes. Applied Engineering, 8(1), 14–30

El-Hajj, M., Mousawi, H., & Fadlallah, A. (2023). Analysis of lightweight cryptographic algorithms on IoT hardware platforms. Future Internet, 15, 54.

Shrestha, M., Johansen, C., & Johansen, J. (2024). LightSC: The making of a usable security classification tool for DevSecOps. arXiv preprint. Published October 2, 2024.

Díaz Ferreyra, N. E., Khelifi, S., Arachchilage, N. A. G., & Scandariato, R. (2024). The good, the bad, and the (un)usable: A rapid literature review on Privacy as Code. arXiv preprint. Published December 21, 2024.

Cloudmersive. (2025, May 20). Cloud Virus API v10-3-4 [Release notes]. Includes performance optimizations and bug fixes; still maintains free usage tier.