e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:07/Issue:04/April-2025 Impact Factor- 8.187 www.irjmets.com

SECURING SECRET FILES USING CYBER SECURITY

Chavan Prem[*1], Nimse Nikhil[*2], Neharkar Roshan[*3], Pansare Parth[*4], Dumbre S.B[*5]

[*1,2,3,4,5]Department Of Computer Engineering, Jaihind Polytechnic, Kuran, Pune, Maharashtra, India.

ABSTRACT

There are various methods to safeguard data and networks from potential attackers. Firewalls are employed to secure passwords as necessary; however, they often fall short. Consequently, systems and networks remain vulnerable to threats. An Intrusion Detection System (IDS) monitors for unauthorized activities within computer systems that originate from the internet. Hackers may manipulate these systems to access valuable and confidential files. It is important to note that most firewalls and IDS primarily focus on defending against external attacks. Access control is a critical security component that adapts to technological advancements and contemporary social contexts. Numerous access control models emerge from specific application domains, such as healthcare and collaborative enterprises. Nevertheless, effective implementation of these models requires additional management strategies, consideration of human factors, and infringement management tailored to the specific environment. This project discusses a robust mechanism for file protection and authentication utilizing the device's MAC address.

Keywords: File Access Control Mechanism, Device MAC Address Authentication.

I. INTRODUCTION

In today's world, it has become almost commonplace for individuals to maintain an account with one of the numerous online file storage services. These platforms allow us to conveniently access our uploaded files from virtually any location and on a variety of devices. We are approaching a stage where, as noted in the abstract, our dependence on these services for individual productivity and collaborative efforts is growing. The necessity for real-time sharing of documents and information is evident; however, similar to many technological advancements, these advantages come with significant concerns regarding security and privacy. It is crucial to fully comprehend these issues before entrusting our data to these providers.

Like any storage solution, a file storage system should possess certain essential security attributes: confidentiality, integrity, write serializability, and read freshness. These characteristics guarantee that user data remains secure from unauthorized alterations and that the most current versions of the data are accessible when retrieved.

Numerous incidents have highlighted vulnerabilities within these services. A notable example is the recent Dropbox incident, where user data was compromised due to a coding error during an update. In other instances, the inherent design of online storage systems has permitted malicious actors to extract substantial amounts of data before the breaches were identified. Additionally, some service providers place the responsibility of data security entirely on the user, which can lead to accessibility issues if the user forgets their access credentials.

Enterprise users encounter similar challenges, with a particular emphasis on effectively integrating security, privacy, and data availability features. Many enterprise clients are hesitant to utilize remote storage solutions due to concerns over insufficient security measures and a lack of transparency.

II. METHODOLOGY

Problem Statement

This initiative aims to improve file security through MAC address-based authentication. The objective is to guarantee that a secured file can only be accessed from a designated, trusted device. Each device possesses a distinct MAC (Media Access Control) address, which will serve as the basis for access verification. During the initial configuration, the file is associated with the MAC address of the authorized device. If an attempt is made to open the file on a different device, access will be refused. This approach effectively prevents unauthorized access, even if the file is duplicated or shared. It introduces a hardware-level security measure for sensitive or personal information. The solution is straightforward, efficient, and user-friendly.

Motivation

- Limitations of Existing Security Tools.
- Growing Complexity of Access Control Needs.
- Need for Device-Specific Authentication.

III. MODELING AND ANALYSIS

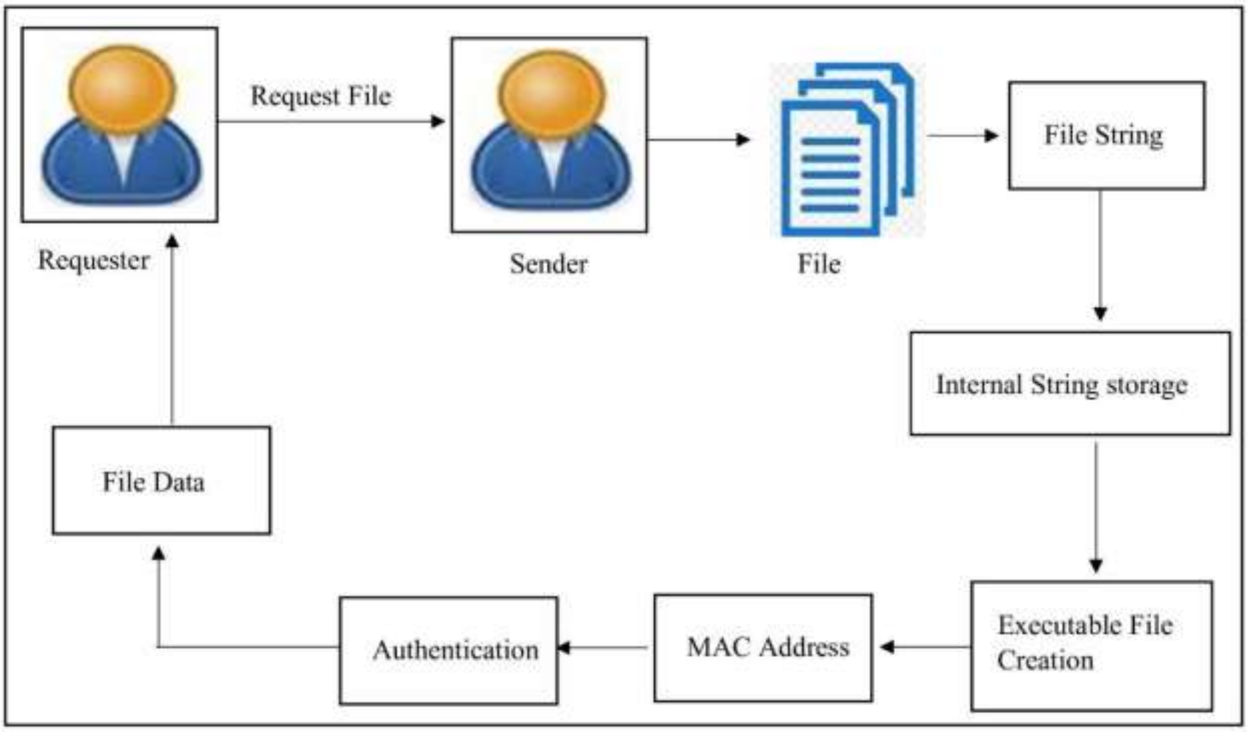


Figure 1: Proposed Work for Remote Desktop monitoring

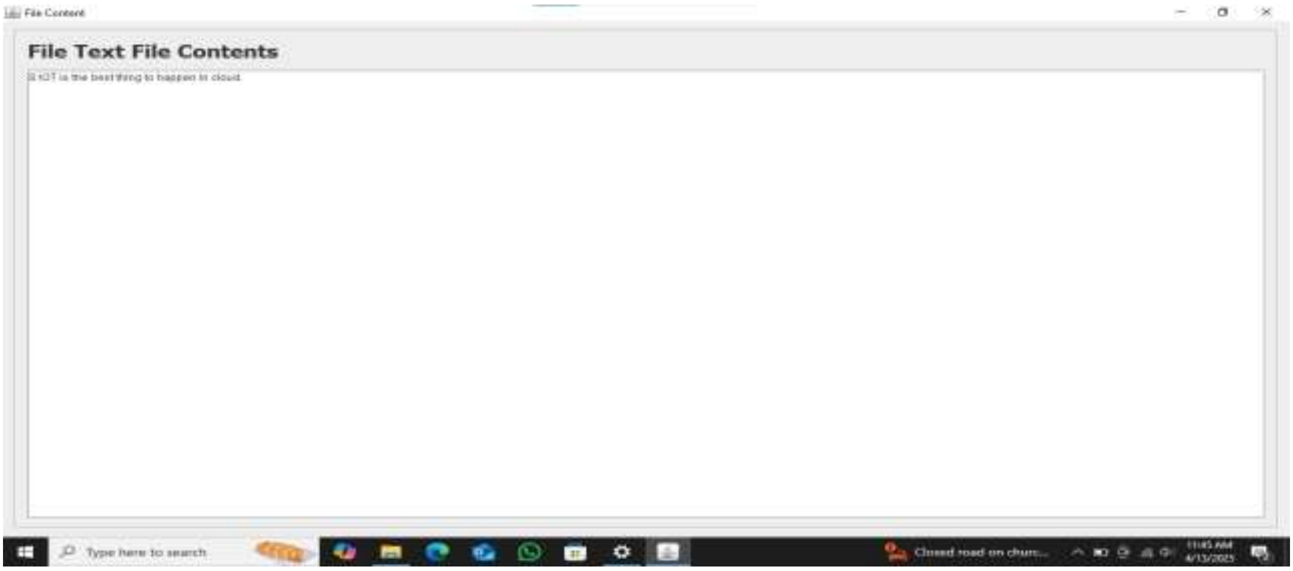
Step 1: File Content Extraction: The process begins when the sender creates a login ID by providing essential details such as a username, password, and email address. Upon successful registration, the user can log into the system by entering the correct username and password. After logging in, the user can select multiple files to share with the recipient. These files are securely stored in a designated location and made available to the file server. The recipient can then access the file server and request a specific file listed there, prompting the sender to receive the request. The content of the file is extracted and read into a string, which is then stored for subsequent processing.

Step 2: Internal File Storage – After the sender submits the request, the recipient accepts it and extracts the file content in the form of a string. This string is effectively utilized to generate a secure file. The system supports input in the form of text files, DOC files, and PDF files, which are processed through the respective JAR files that enable interfacing with the Java code. Once the content is prepared, it is utilized in the next step for the creation of an executable file.

Step 3: Executable File Creation and MAC Address Authentication – Following the extraction of file content, the data is securely stored and embedded within Java code. This procedure is applied to the entire file, allowing the content to be displayed in a text area. The Java code can then be converted into a JAR file using the clean and build option in the NetBeans IDE. Once the JAR file is created, its status in the database is updated to "secured," and the file is ready to be sent to the user. Utilizing socket programming, the file is transmitted to the recipient's IP address. Prior to sending, the file undergoes authentication and authorization through the MAC address of the receiving device, ensuring that it can only be accessed on the designated device.

IV. RESULTS AND DISCUSSION

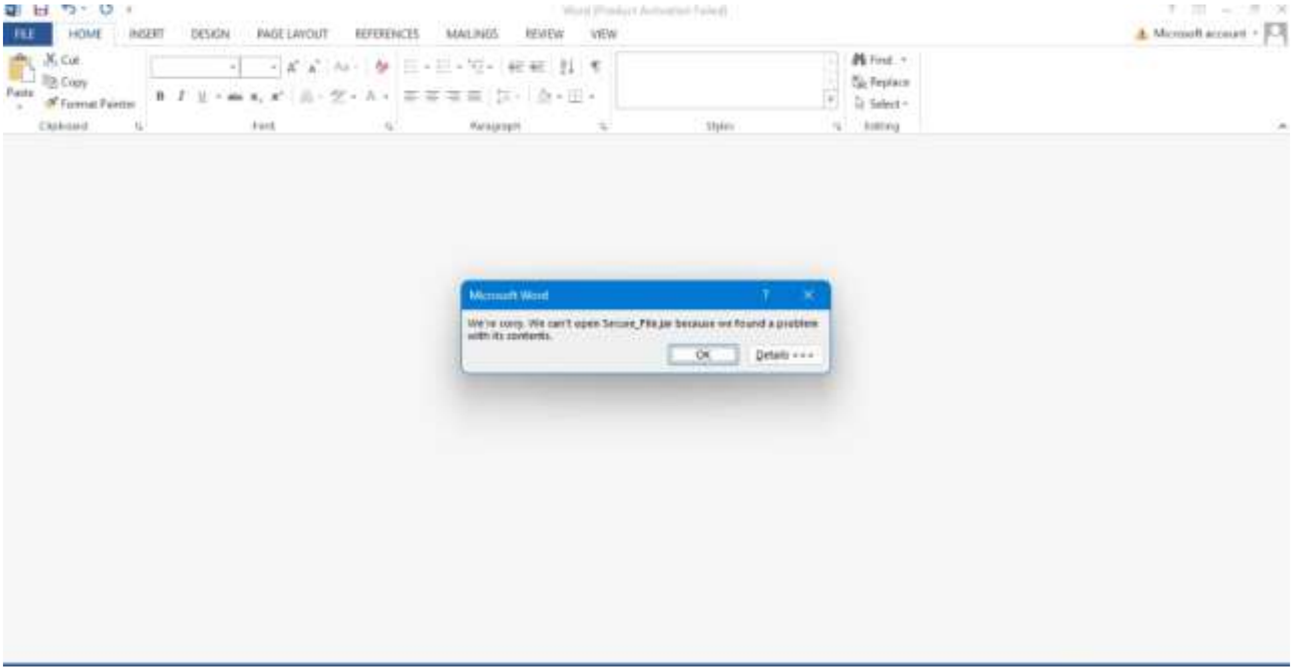
We have Securing Secret Files successfully run and tested



File Text File Contents

IOT is the best thing to happen in cloud.

Figure 2:



Microsoft Word

We're sorry. We can't open Secure_File.jar because we found a problem with its contents.

Figure 3: Final Output

V. CONCLUSION

Numerous strategies exist for safeguarding data and communications from cyber threats. Security systems are employed to protect encryption keys when necessary; however, these measures often fall short. Consequently, systems and services remain under constant scrutiny for potential threats. An intrusion prevention system detects harmful activities on a computing device that are transmitted over the network. Such intrusions can lead to the unauthorized access of sensitive and confidential information. Nevertheless, the majority of security protocols and intrusion detection systems are primarily focused on defending computer systems against external threats. Access control serves as a security mechanism that adapts to evolving technology and contemporary cultural shifts. Various authentication methods arise from different software sectors, including healthcare and collaborative enterprises, necessitating additional administrative tools, considerations of human factors, and governance of violations to ensure effective implementation within specific operational contexts. This article utilizes the device MAC address to illustrate a robust approach to file security and authentication.

VI. REFERENCES

[1] W. Shen, J. Qin, J. Yu, R. Hao, and J. Hu present a study titled "Enabling Identity-Based Integrity Auditing and Data Sharing with Sensitive Information Hiding for Secure Cloud Storage," published in the IEEE Transactions on Information Forensics and Security, volume 14, issue 2, pages 331-346, in February 2019. The DOI for this work is 10.1109/TIFS.2018.2850312.

[2] O. A. Khashan discusses "Secure Outsourcing and Sharing of Cloud Data Using a User-Side Encrypted File System" in IEEE Access, volume 8, pages 210855-210867, published in 2020. The DOI for this article is 10.1109/ACCESS.2020.3039163.

[3] F. Khoda Parast, B. Kelly, S. Hakak, Y. Wang, and K. B. Kent introduce "CephArmor: A Lightweight Cryptographic Interface for Secure High-Performance Ceph Storage Systems," featured in IEEE Access, volume 10, pages 127911-127927, in 2022. The DOI for this publication is 10.1109/ACCESS. 2022.3227384.

[4] S. Mofrad, I. Ahmed, F. Zhang, S. Lu, P. Yang, and H. Cui explore "Securing Big Data Scientific Workflows via Trusted Heterogeneous Environments" in the IEEE Transactions on Dependable and Secure Computing, volume 19, issue 6, pages 4187-4203, published between November and December 2022. The DOI for this research is 10.1109/TDSC.2021.3123640.

[5] Peter Čuřík, Roderik Ploszek, and Pavol Zajac discuss the "Practical Use of Secret Sharing for Enhancing Privacy in Clouds" in Electronics, volume 11, article 2758, published in 2022. The DOI for this article is https://doi.org/10.3390/electronics11172758.

[6] L. Li, D. Jin, T. Zhang, and N. Li propose "A Secure, Reliable and Low-Cost Distributed Storage Scheme Based on Blockchain and IPFS for Firefighting IoT Data," published in IEEE Access, volume 11, pages 97318-97330,