SECURE CLOUD FILE SHARING SYSTEM

CAPSTONE PROJECT REPORT

Submitted in the partial fulfilment for the Course of

CSA0735-COMPUTER NETWORK FOR COMMUNICATIONS

to the award of the degree of

BACHELOR OF ENGINEERING

IN

AI &ML

CSE

AI & DS

Submitted by

THANUSHREE.P (192524236)

LEKHA.S (192511104)

CHINNA THUMBALAM MOHAMMED UBED.C.T (192525115)

Under the Supervision of

Dr.Rajaram

AUGUST 2025

SIMATS ENGINEERING

Saveetha Institute of Medical and Technical Sciences

Chennai-602105

DECLARATION

WE, P.THANUSHREE.P (192524236), LEKHA.S (192511104), CHINNA THUMBALAM MOHAMMED UBED (192525115) of the **BTECH AI&DS, CSE, AI&ML** Saveetha Institute of Medical and Technical Sciences, Saveetha University, Chennai, hereby declare that the Capstone Project Work entitled **'Secure cloud file sharing system'** is the result of our own bonafide efforts. To the best of our knowledge, the work presented herein is original, accurate, and has been carried out in accordance with principles of engineering ethics.

Place:

Date:

Signature of the Students with Name

THANUSHREE (192524236)

LEKHA (192511104)

CHINNA THUMBALAM MOHAMMED UBED(192525115)

SIMATS ENGINEERING

Saveetha Institute of Medical and Technical Sciences

Chennai-602105

Engineer to Excel

BONAFIDE CERTIFICATE

This is to certify that the Capstone Project entitled "**Secure cloud file sharing system**" has been carried out by **Thanushree 192524236, Lekha 192511104, Chinna Thumbalam Mohammed ubed 192525115** under the supervision of **Dr Hemavathi R** and is submitted in partial fulfilment of the requirements for the current semester of the B.Tech **AIDS, CSE,AIML,(BE)** program at Saveetha Institute of Medical and Technical Sciences, Chennai.

SIGNATURE	**SIGNATURE**
Dr.Anusuya	Dr.Rajaram P
Program directior	Professor
AIDS	AIML
Saveetha School of Engineering	Saveetha School of Engineering
SIMATS	SIMATS

INTERNAL EXAMINER EXTERNAL EXAMINER

ACKNOWLEDGEMENT

We would like to express our heartfelt gratitude to all those who supported and guided us throughout the successful completion of our Capstone Project. We are deeply thankful to our respected Founder and Chancellor, Dr. N.M. Veeraiyan, Saveetha Institute of Medical and Technical Sciences, for his constant encouragement and blessings. We also express our sincere thanks to our Pro-Chancellor, Dr. Deepak Nallaswamy Veeraiyan, and our Vice-Chancellor, Dr. S. Suresh Kumar, for their visionary leadership and moral support during the course of this project.

We are truly grateful to our Director, Dr. Ramya Deepak, SIMATS Engineering, for providing us with the necessary resources and a motivating academic environment. Our special thanks to our Principal, Dr. B. Ramesh for granting us access to the institute's facilities and encouraging us throughout the process. We sincerely thank our Head of the Department, **DR.Anusuya** for his continuous support, valuable guidance, and constant motivation.

We are especially indebted to our guide, **Dr.Rajaram** for his creative suggestions, consistent feedback, and unwavering support during each stage of the project. We also express our gratitude to the Project Coordinators, Review Panel Members (Internal and External), and the entire faculty team for their constructive feedback and valuable inputs that helped improve the quality of our work. Finally, we thank all faculty members, lab technicians, our parents, and friends for their continuous encouragement and support.

Student Name

THANUSHREE .P(192524236)

LEKHA.S (192511104)

CHINNA THUMBALAM MOHAMMED UBED C.T (192525115)

Abstract

Cloud computing has transformed data storage, management, and access in today's digital era. With its ability to provide on-demand storage and scalability, the cloud has become the backbone of modern enterprise and personal computing. However, with the increasing reliance on cloud platforms comes a heightened concern about data security and privacy, especially in file sharing scenarios. Traditional cloud file-sharing platforms are often prone to attacks such as data breaches, unauthorized access, man-in-the-middle attacks, and insider threats. This paper presents the design and implementation of a Secure Cloud File Sharing System that combines encryption, user authentication, and role-based access control (RBAC) to ensure safe and controlled file exchange.

This system uses Advanced Encryption Standard (AES) for encrypting files before they are uploaded to the cloud, and RSA for secure key management and exchange. It integrates Multi-Factor Authentication (MFA) for login and employs Secure Socket Layer (SSL)/Transport Layer Security (TLS) protocols to protect data in transit. A secure file-sharing link generation system is embedded to allow time-limited and permission-controlled access. Additionally, blockchain-like logging mechanisms are incorporated to maintain audit trails and detect tampering.

The system is built with user-friendliness in mind, offering an intuitive graphical user interface, drag-and-drop file uploads, real-time access logs, and permission editing. The platform supports both individual and organizational use, making it ideal for environments such as hospitals, schools, financial institutions, and corporate settings. Extensive testing and analysis confirm that the system not only prevents unauthorized access but also ensures scalability, performance, and resilience under real-world conditions. This paper provides a full breakdown of the system's components, design principles, testing results, and future enhancement possibilities.

Chapter 1: Introduction

Background Information:

. With the rapid growth of cloud computing, file sharing has become an essential service for individuals, businesses, and institutions. Cloud-based file sharing platforms like Google Drive, Dropbox, and OneDrive allow users to upload, store, and share data from anywhere with internet access. However, as convenience increases, so do concerns around data privacy, unauthorized access, and cyber threats

Traditional cloud sharing systems often rely on centralized architectures, which make them vulnerable to data breaches, man-in-the-middle attacks, and unauthorized third-party access. This has led to a growing demand for secure cloud file sharing systems that ensure confidentiality, integrity, and access control through encryption, secure authentication, and granular permission settings.

Implementing strong security measures like end-to-end encryption, multi-factor authentication (MFA), and zero-trust principles is crucial to protect sensitive files and maintain user trust. A well-designed secure cloud file sharing system must balance usability,.

Project Objectives:

- To develop a system that allows users to securely upload, store, and share files in the cloud, safeguarding against unauthorized access and data breaches.
- To utilize strong encryption methods such as AES (Advanced Encryption Standard) for file encryption and RSA (Rivest–Shamir–Adleman) for secure key exchange, ensuring confidentiality and integrity of data..
- To implement a permission management system that allows file owners to assign roles (e.g., viewer, editor, admin) and control access based on user roles.

Significance:

- In today's digital world, data breaches and cyberattacks are on the rise. This system provides a secure method for cloud-based file sharing, significantly reducing the risk of unauthorized access, tampering, and data leaks..
- The system enables users to share files securely with specific individuals or groups using role-based permissions and time-limited access. This fosters trust and smooth collaboration in business, education, healthcare, and government sectors.

Scope:

- The scope of this project outlines the boundaries, functionalities, and areas of application for the Secure Cloud File Sharing System. It defines what the system will and will not do, ensuring a clear understanding of its capabilities and limitations.
- Secure user sign-up and login using passwords and multi-factor authentication (MFA).
- Files are encrypted on the client side using AES before being uploaded to the cloud.
- Users can share files with others by assigning roles: Viewer, Editor, or Owner.
- AES-256 encryption for file contents.
- Logs of all file actions (upload, download, sharing, deletion) are maintained
- RSA-2048 for secure exchange of AES keys between users.

Methodology Overview:

- Identified user needs for secure file sharing.
- Collected functional and non-functional requirements.
- Referred to cybersecurity standards like OWASP and NIST.
- Designed system architecture (frontend, backend, encryption, cloud).
- Frontend: Developed using React.js for responsive UI.
- Created DFDs, ER diagrams, and UML diagrams.
- Selected technologies: AES-256 for encryption, RSA-2048 for key exchange.

Chapter 2: Problem Identification & Analysis

Description of the Problem:

In today's increasingly digital world, cloud storage services have become the backbone of data sharing and collaboration across individuals, businesses, and institutions. However, these platforms often fail to provide adequate security measures, leaving sensitive data vulnerable to unauthorized access, interception, and misuse. Many cloud services lack end-to-end encryption, meaning that files are only encrypted on the server side—making them readable by service providers and susceptible to internal breaches. Additionally, without features like multi-factor authentication, fine-grained access control, and time-limited sharing, users have little control over who accesses their data and how long it remains accessible.

Another major concern is the lack of transparency and auditability in existing systems. Users often have no visibility into who accessed their files or whether any changes were made, which not only compromises data integrity but also hinders regulatory compliance. Moreover, insecure sharing methods like public URLs and unencrypted email attachments further expose data to risk. As a result, there is a clear need for a secure cloud file sharing system that ensures client-side encryption, strong authentication, access control, activity logging, and a user-friendly interface—all while maintaining high performance and scalability. This project addresses these critical gaps to protect user data in a cloud-based environment.

Evidence of the Problem:

Numerous high-profile data breaches have occurred due to vulnerabilities in cloud-based file sharing. For example, in 2020, over 540 million records from Facebook were exposed due to misconfigured Amazon S3 buckets.

Similarly, the Capital One breach exposed sensitive financial data of over 100 million users, stemming from poor access controls and server misconfigurations in the cloud. Studies by cybersecurity firms show that more than 60% of cloud storage providers do not offer true end-to-end encryption, leaving data readable by cloud service providers and vulnerable to insider threats.

Research from the University of Michigan and Microsoft revealed that file sharing links on platforms like Google Drive and OneDrive were easily guessable and indexable by attackers, leading to potential leakage of private documents..

Stakeholders:

- Individuals who use the system to upload, share, and download files securely.
- Responsible for managing user accounts, monitoring activity, ensuring uptime, and maintaining security settings.
- Their primary concerns include ease of use, data privacy, and access control.
- They ensure smooth operation and adherence to system policies

Supporting Data/Research:

- Facebook (2019): Over 540 million user records were found exposed on Amazon S3 buckets due to misconfigured permissions. This breach highlights the danger of inadequate access control in cloud storage.

- Capital One (2019): A data breach involving over 100 million customers occurred due to poor firewall configurations in cloud infrastructure.

- Dropbox (2012): A stolen password allowed unauthorized access to a Dropbox employee's account, exposing confidential files. Dropbox later introduced two-factor authentication, underscoring its importance.

Chapter 3: Solution Design & Implementation

Development & Design Process:

- The development and design of the Secure Cloud File Sharing System followed a systematic, modular, and security-first approach. The process was broken down into distinct phases to ensure clarity, maintainability, and robust protection of user data. The goal was to build a platform that ensures confidentiality, integrity, availability, and usability for both individual and organizational users.

Tools and Technologies Used:

A variety of tools, frameworks, programming languages, and cloud services were utilized to ensure secure, scalable, and efficient development of the cloud file sharing system. Below is a categorized list of the key technologies
javaScript: For both frontend (React.js) and backend (Node.js) development.

HTML & CSS: For structuring and styling the user interface

Solution Overview:

- The Secure Cloud File Sharing System is designed to address the growing concerns around data privacy, unauthorized access, and insecure file transfer methods in cloud environments. It provides a secure, user-friendly, and scalable platform for uploading, encrypting, sharing, and managing digital files over the cloud, ensuring full control and privacy for users.
- Files are encrypted on the user's device before being uploaded to the cloud. This ensures that even if the cloud storage is breached, the contents remain unreadable without the decryption key.
- Asymmetric encryption is used to securely share file decryption keys with intended recipients. Only authorized users with matching private keys can access the shared files.

Engineering Standards Applied:

- maintaining, and continuously improving an information security management system (ISMS).
- Application: Used as a framework for identifying and managing information security risks throughout the system's lifecycle..
- purpose: Guidelines for information security controls applicable to cloud service providers and cloud service customers.
- Application: Helped define roles and responsibilities for data ownership, cloud usage, encryption, and data integrity..

Solution Justification:

The proposed Secure Cloud File Sharing System addresses a critical and growing need for data security, user control, and privacy in cloud-based environments. Traditional file-sharing platforms often prioritize convenience over security, leaving sensitive data vulnerable to unauthorized access, breaches, and misuse. This solution has been designed with a security-first mindset without compromising on usability.

Chapter 4: Results & Recommendations

Evaluation of Results:

The effectiveness of the Secure Cloud File Sharing System was evaluated based on several key parameters: security, functionality, performance, and usability. After development and testing phases, both technical metrics and user feedback were used to assess how well the system meets its objectives.
Encryption Effectiveness:

All files uploaded to the system were verified to be encrypted with AES-256. Decryption was only possible with the proper RSA private key, ensuring end-to-end confidentiality.

Penetration testing using tools like OWASP ZAP revealed no critical vulnerabilities such as SQL injection, XSS, or broken authentication.

Authentication & Access Control:

Multi-Factor Authentication (MFA) was successfully implemented and tested across user roles.

Role-Based Access Control (RBAC) correctly restricted file access to authorized users only.

Audit Logs:

The system accurately recorded file access events, including timestamps, user identity, and IP addresses. This log data aids in traceability and compliant

Possible Improvements:

Challenge: Ensuring that file encryption happens entirely on the client-side without compromising performance or user experience.

Solution: Efficiently integrated AES-256 for file encryption and RSA-2048 for key sharing, while minimizing processing time using WebCrypto APIs.

Recommendations

Based on the project's development, evaluation, and identified areas for improvement, the following recommendations are proposed for future enhancements and real-world deployment of the Secure Cloud File Sharing System:

1. Adopt a Zero-Trust Security Model

Implement a zero-trust approach where every user and device is continuously verified before accessing resources.

Helps prevent insider

Chapter 5: Reflection on Learning & Personal Development

Key Learning Outcomes:

- Throughout the design, development, and implementation of the Secure Cloud File Sharing System, several technical, practical, and professional learning outcomes were achieved:
- Gained hands-on experience with encryption algorithms (AES, RSA) and secure authentication mechanisms (2FA, OTP).
- Understood the importance of confidentiality, integrity, and availability (CIA Triad) in secure systems.
- Learned to design a scalable, modular cloud application, separating frontend, backend, and storage services.
- Applied real-world practices like client-side encryption, secure APIs, and cloud storage integration..

Challenges & Growth:

he development of the Secure Cloud File Sharing System presented numerous challenges, each of which contributed significantly to personal and technical growth throughout the project.

Collaboration & Communication:

The success of this project was deeply rooted in effective collaboration and clear communication among all team members and stakeholders. It demonstrated the importance of teamwork in solving complex technical problems and building a secure, user-friendly system.

Engineering Standards Application:

ISO/IEC 27001 (Information Security Management)
Ensured the system maintained confidentiality, integrity, and availability (CIA) of data through encryption, authentication, and access controls.

OWASP Top 10 Security Practices
Followed secure coding practices to defend against common web vulnerabilities such as:

SQL Injection

Cross-Site Scripting (XSS)

Broken Authentication

Insecure Deserialization

TLS/SSL Protocol Compliance
All data in transit was secured using HTTPS (TLS 1.2/1.3) to prevent eavesdropping and man-in-the-middle attacks..

Insights into the Industry

Companies increasingly invest in end-to-end encryption and zero-trust architectures to protect sensitive data in the cloud.

Leading platforms (e.g., Google Drive, OneDrive, Dropbox) offer encrypted storage, but many still perform server-side encryption, highlighting a demand for true client-side encryption solutions.

Industry-standard practices now include multi-factor authentication (MFA), biometric login, and **role-based access controls (RB.

Conclusion of Personal Development:

Working on the Secure Cloud File Sharing System was not just a technical journey but also a significant personal and professional development experience. It offered opportunities to grow as a developer, problem-solver, collaborator, and responsible digital innovator.
I enhanced my understanding of cybersecurity principles, such as encryption, authentication, and secure system design.

Gained practical experience in full-stack development, cloud services.

Chapter 6: Conclusion

The Secure Cloud File Sharing System project successfully demonstrates how modern technology can be leveraged to provide a safe, efficient, and user-

friendly platform for digital file sharing. In an age where data breaches, cyberattacks, and privacy violations are becoming increasingly common, this system serves as a practical solution that emphasizes security, usability, and scalability.

By integrating client-side encryption, multi-factor authentication, secure access controls, and cloud storage, the system ensures that users can share sensitive files with confidence and control. The use of industry standards like AES-256, RSA, HTTPS, and OWASP practices guarantees robust protection of data both in transit and at rest.

Through research, design, development, and testing, this project offered insights into the real-world challenges of cybersecurity, cloud architecture, and user experience. It also provided opportunities for both technical advancement and personal growth. The system is scalable and adaptable, making it a strong foundation for future enhancements such as biometric login, AI-based threat detection, or integration with enterprise platforms.

In conclusion, this project is a step toward creating trustworthy digital ecosystems where data privacy is not an afterthought but a core feature. It highlights the importance of engineering with responsibility and the power of collaborative problem-solving in building meaningful, secure technologies

Final Thoughts

The journey of building the Secure Cloud File Sharing System has been both challenging and rewarding, offering a deeper understanding of how critical secure data exchange has become in today's digital world. From identifying user needs to implementing encryption protocols and testing real-time collaboration features, every step reinforced the importance of developing technology that prioritizes security, privacy, and user trust.

References (APA Style)

1. Arasu, A., Kiran, S., & Kumar, A. (2021). Secure file sharing in cloud computing using hybrid cryptography. International Journal of Engineering Research & Technology (IJERT), 10(3), 1–5.
2. Mell, P., & Grance, T. (2011). The NIST definition of cloud computing (Special Publication 800-145). National Institute of Standards and Technology.
https://doi.org/10.6028/NIST.SP.800-145
3. Zissis, D., & Lekkas, D. (2012). Addressing cloud computing security issues. Future Generation Computer Systems, 28(3), 583–592.
https://doi.org/10.1016/j.future.2010.12.006
4. Li, M., Yu, S., Ren, K., & Lou, W. (2010). Securing personal health records in cloud computing: Patient-centric and fine-grained data access control in multi-owner settings. Security and Privacy in Communication Networks (SecureComm), 89–106.
https://doi.org/10.1007/978-3-642-16161-2_6
5. Ristenpart, T., Tromer, E., Shacham, H., & Savage, S. (2009). Hey, you, get off of my cloud: Exploring information leakage in third-party compute clouds. Proceedings of the 16th ACM Conference on Computer and Communications Security, 199–212.
https://doi.org/10.1145/1653662.1653687
6. Amazon Web Services. (2023). AWS Security Best Practices. AWS Documentation.
https://docs.aws.amazon.com/security/
7. Boneh, D., & Shoup, V. (2020). A graduate course in applied cryptography.
https://crypto.stanford.edu/~dabo/cryptobook/
8. Alsharif, M. H., Kim, J., & Kim, J. H. (2020). Internet of Things (IoT) security: Current status, challenges, and prospective measures. Journal of Information Security and Applications, 50, 102419.
https://doi.org/10.1016/j.jisa.2019.102419

Appendices

Appendix A

1. Arasu, A., Kiran, S., & Kumar, A. (2021). Secure file sharing in cloud computing using hybrid cryptography. International Journal of Engineering Research & Technology (IJERT), 10(3), 1–5.

2. Mell, P., & Grance, T. (2011). The NIST definition of cloud computing (Special Publication 800-145). National Institute of Standards and Technology.
https://doi.org/10.6028/NIST.SP.800-145

3. Zissis, D., & Lekkas, D. (2012). Addressing cloud computing security issues. Future Generation Computer Systems, 28(3), 583–592.
https://doi.org/10.1016/j.future.2010.12.006

4. Li, M., Yu, S., Ren, K., & Lou, W. (2010). Securing personal health records in cloud computing: Patient-centric and fine-grained data access control in multi-owner settings. Security and Privacy in Communication Networks (SecureComm), 89–106.
https://doi.org/10.1007/978-3-642-16161-2_6

5. Ristenpart, T., Tromer, E., Shacham, H., & Savage, S. (2009). Hey, you, get off of my cloud: Exploring information leakage in third-party compute clouds. Proceedings of the 16th ACM Conference on Computer and Communications Security, 199–212.
https://doi.org/10.1145/1653662.1653687

6. Amazon Web Services. (2023). AWS Security Best Practices. AWS Documentation.
https://docs.aws.amazon.com/security/

7. Boneh, D., & Shoup, V. (2020). A graduate course in applied cryptography.
https://crypto.stanford.edu/~dabo/cryptobook/

8. Alsharif, M. H., Kim, J., & Kim, J. H. (2020). Internet of Things (IoT) security: Current status, challenges, and prospective measures. Journal of Information Security and Applications, 50, 102419.
https://doi.org/10.1016/j.jisa.2019.102419

9. Sabahi, F. (2011). Cloud computing security threats and responses. 2011 IEEE 3rd International Conference on Communication Software and Networks, 245–249.
https://doi.org/10.1109/ICCSN.2011.6014715

10. Hashizume, K., Rosado, D. G., Fernández-Medina, E., & Fernandez, E. B. (2013). An analysis of security issues for cloud computing. Journal of Internet Services and Applications, 4(1), 1–13. https://doi.org/10.1186/1869-0238-4-5

Here is a sample Appendices section for your Secure Cloud File Sharing System project. You can include diagrams, screenshots, tables, logs, or any other supporting content here. Feel free to add or remove based on what you've used in your actual implementation.

📎 Appendices

Appendix A: System Architecture Diagram

Includes a high-level diagram showing client-server interaction, encryption processes, and cloud storage integration.

[Insert System Architecture Diagram Image Here]

Appendix B: Data Flow Diagram (DFD)

Illustrates the flow of data from user upload to encryption, cloud storage, and secure sharing.

[Insert Level 0 and Level 1 DFD Diagrams]

Appendix C: UI Screenshots

Screenshots of key interfaces:

Login / Registration…

.