

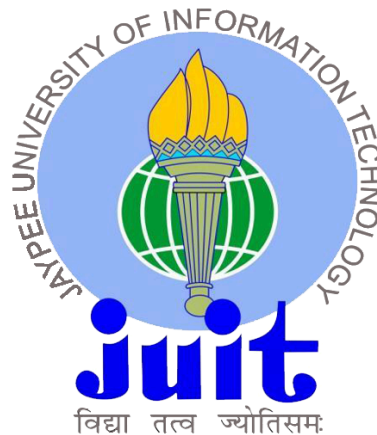
Secure Cloud Storage System

A major project report submitted in partial fulfilment of the requirement
for the award of degree of

Bachelor of Technology
in
Computer Science & Engineering / Information Technology

Submitted by
Shaily Tiwari (201294)
Shubhang Shukla (201296)

Under the guidance & supervision of
Mr. Praveen Modi



**Department of Computer Science & Engineering and
Information Technology**
Jaypee University of Information Technology,
Waknaghat, Solan - 173234 (India)

Candidate's Declaration

We hereby declare that the work presented in this report entitled '**Secure Cloud Storage System**' in partial fulfilment of the requirements for the award of the degree of **Bachelor of Technology in Computer Science & Engineering / Information Technology** submitted in the Department of Computer Science & Engineering and Information Technology, Jaypee University of Information Technology, Waknaghat is an authentic record of my own work carried out over a period from August 2023 to May 2024 under the supervision of **Mr Praveen Modi** (Assistant Professor, Computer Science & Engineering and Information Technology).

The matter embodied in the report has not been submitted for the award of any other degree or diploma.

Student Name: Shaily Tiwari

Roll No.: 201294

Student Name: Shubhang Shukla

Roll No.: 201296

This is to certify that the above statement made by the candidate is true to the best of my knowledge.

Supervisor Name: Mr. Praveen Modi

Designation: Assistant Professor Grade-II

Department: Computer Science & Engineering and Information Technology

Dated: 15-05-2024

CERTIFICATE

This is to certify that the work which is being presented in the project report titled '**Secure Cloud Storage System**' in partial fulfilment of the requirements for the award of the degree of **Bachelor of Technology in Computer Science & Engineering / Information Technology** submitted in the Department of Computer Science & Engineering and Information Technology, Jaypee University of Information Technology, Wagnaghat is an authentic record of work carried out by **Shaily Tiwari(201294)** and **Shubhang Shukla (201296)** during the period from August 2023 to May 2024 under the supervision of **Mr. Praveen Modi**, Assistant Professor Grade-II, Department of Computer Science and Engineering, Jaypee University of Information Technology, Wagnaghat.

Shaily Tiwari
201294

Shubhang Shukla
201296

The above statement made is correct to the best of our knowledge.

Mr. Praveen Modi
Assistant Professor Grade-II
Computer Science & Engineering and Information Technology
Jaypee University of Information Technology, Wagnaghat

ACKNOWLEDGEMENT

Firstly, I express my heartiest thanks and gratefulness to almighty God for His divine blessing makes it possible to complete the project work successfully. I am really grateful and wish my profound indebtedness to Supervisor **Mr. Praveen Modi**, Assistant Professor Grade-II, Department of CSE Jaypee University of Information Technology, Wakhnaghat. & keen interest of my supervisor in the field of “Information Security” to carry out this project. His endless patience, scholarly guidance, continual encouragement, constant and energetic supervision, constructive criticism, valuable advice, reading many inferior drafts and correcting them at all stages have made it possible to complete this project. I would like to express my heartiest gratitude to Mr. Praveen Modi, Department of CSE, for his kind help to finish my project. I would also generously welcome each one of those individuals who have helped me straightforwardly or in a roundabout way in making this project a win. In this unique situation, I might want to thank the various staff individuals, both educating and non-instructing, which have developed their convenient help and facilitated my undertaking.

Finally, I must acknowledge with due respect the constant support of my parents.

Shaily Tiwari

201294

Computer Science & Engineering and Information Technology

Jaypee University of Information Technology, Wakhnaghat

Shubhang Shukla

201296

Computer Science & Engineering and Information Technology

Jaypee University of Information Technology, Wakhnaghat

TABLE OF CONTENTS

| | |
|---|-------------|
| Candidate's Declaration | i |
| CERTIFICATE | ii |
| ACKNOWLEDGEMENT | iii |
| TABLE OF CONTENTS | iv |
| LIST OF TABLES | vi |
| LIST OF FIGURES | vii |
| LIST OF ABBREVIATIONS | viii |
| ABSTRACT | ix |
| Chapter 1: INTRODUCTION | 1 |
| 1.1 INTRODUCTION | 1 |
| 1.1.1 THE ORIGINS OF COMBINATORIAL CRYPTOGRAPHY | 2 |
| 1.1.2 MAINTAINING THE INTEGRITY & CONFIDENTIALITY OF DATA | 3 |
| 1.1.3 BOOSTING CLOUD SIMULATIONS | 3 |
| 1.1.4 ACCEPTING INNOVATION WITHOUT GIVING UP | 4 |
| 1.1.5 GAZING FORWARD | 4 |
| 1.2 PROBLEM STATEMENT | 4 |
| 1.2.1.DEFICIENCIES IN ONE-METHOD ENCRYPTION: | 5 |
| 1.2.2 DATA INTEGRITY AND CONFIDENTIALITY: | 5 |
| 1.2.3 SCALABILITY AND PERFORMANCE: | 5 |
| 1.2.4 FUTURE-PROOFING SECURITY: | 5 |
| 1.3 OBJECTIVES | 6 |
| 1.3.1 DEVELOPMENT OF ROBUST ENCRYPTION MECHANISMS: | 6 |
| 1.3.2 KEY MANAGEMENT SYSTEM IMPLEMENTATION: | 7 |
| 1.3.3 INTEGRATION OF SECURE PROTOCOLS: | 7 |
| 1.3.4 SECURE DATA STORAGE MECHANISMS: | 7 |
| 1.3.5 AUTHENTICATION AND ACCESS CONTROL: | 7 |
| 1.3.6 TESTING AND VALIDATION: | 7 |
| 1.3.7 DOCUMENTATION AND KNOWLEDGE SHARING: | 7 |
| 1.4 SIGNIFICANCE AND MOTIVATION OF PROJECT | 8 |
| 1.4.1 SIGNIFICANCE | 8 |
| 1.4.2 MOTIVATION | 9 |
| 1.5 ORGANISATION OF PROJECT REPORT | 10 |
| Chapter 2: LITERATURE SURVEY | 12 |
| PAPER - 1 | 19 |
| PAPER - 2 | 20 |
| PAPER - 3 | 21 |
| PAPER - 4 | 22 |

| | |
|---|-----------|
| PAPER - 5 | 23 |
| PAPER - 6 | 24 |
| PAPER - 7 | 25 |
| PAPER - 8 | 26 |
| PAPER - 9 | 27 |
| PAPER - 10 | 28 |
| PAPER - 11 | 29 |
| PAPER - 12 | 30 |
| PAPER - 13 | 30 |
| PAPER - 14 | 32 |
| Chapter 3: SYSTEM DEVELOPMENT | 33 |
| 3.1 REQUIREMENT AND ANALYSIS | 33 |
| 3.1.1 SOFTWARE REQUIREMENTS | 33 |
| 3.1.2 HARDWARE REQUIREMENTS | 34 |
| 3.2 PROPOSED DESIGN AND ARCHITECTURE | 35 |
| 3.2.1 PROPOSED MODEL | 35 |
| 3.2.2 FLOW DIAGRAM | 36 |
| 3.3 DATA PREPARATION | 38 |
| 3.3.1 RESEARCH ALGORITHMS | 38 |
| 3.3.2 HYBRID CRYPTOSYSTEM | 41 |
| 3.4 IMPLEMENTATION | 42 |
| 3.4.1 CODE SNIPPETS | 42 |
| 3.4.2 ALGORITHMS | 45 |
| 3.4.3 TOOLS AND TECHNIQUES | 46 |
| 3.5 Key Challenges | 47 |
| Chapter 4: TESTING | 48 |
| 4.1 TESTING STRATEGIES: | 48 |
| 4.2 TESTING FILES AND PROCESSING TIME EVALUATION: | 49 |
| Chapter 5: RESULTS AND EVALUATION | 53 |
| 5.1 RESULTS | 53 |
| 5.2 EVALUATION | 59 |
| Chapter 6: CONCLUSION AND FUTURE SCOPE | 60 |
| 6.1 CONCLUSION | 60 |
| 6.2 FUTURE SCOPE | 61 |
| REFERENCES | 62 |

LIST OF TABLES

| S. No. | Title | Page No. |
|---------------|--|-----------------|
| 1 | Table 2.1 List of literature surveys | 12-18 |
| 2 | Table 5.1 Comparison of algorithms implemented | 43-55 |

LIST OF FIGURES

| S.No. | FIGURE No. | FIGURE TITLE | PAGE No. |
|--------------|-------------------|----------------------------------|-----------------|
| 1 | Figure 1.1 | Models of Cloud Computing | 2 |
| 2 | Figure 1.2 | Public Key Cryptography | 3 |
| 3 | Figure 3.1 | System Flow Diagram | 36 |
| 4 | Figure 3.2 | AES Algorithm | 38 |
| 5 | Figure 3.3 | AES Algorithm (Implemented) | 39 |
| 6 | Figure 3.4 | ChaCha20Poly1305 Algorithm | 39 |
| 7 | Figure 3.5 | AES GCM Algorithm | 40 |
| 8 | Figure 3.6 | AES CCM Algorithm | 40 |
| 9 | Figure 3.7 | RSA Algorithm | 41 |
| 10 | Figure 3.8 | Home Page Code Snippet | 42 |
| 11 | Figure 3.9 | Upload & Encryption Code Snippet | 43 |
| 12 | Figure 3.10 | Storage Successful Page | 43 |
| 13 | Figure 3.11 | Restoration and Decryption Code | 44 |
| 14 | Figure 3.12 | Terminal Output | 45 |
| 15 | Figure 5.1 | About Page | 56 |
| 16 | Figure 5.2 | Home Page | 56 |
| 17 | Figure 5.3 | Upload Page | 57 |
| 18 | Figure 5.4 | Success Page | 57 |
| 19 | Figure 5.5 | Download Page | 58 |
| 20 | Figure 5.6 | Restore Page | 58 |
| 21 | Figure 5.7 | Downloaded File | 59 |

LIST OF ABBREVIATIONS

| S.No. | ABBREVIATION | DEFINITION |
|--------------|---------------------|---------------------------------|
| 1. | AES | Advanced Encryption Standard |
| 2. | DES | Data Encryption Standard |
| 3. | RSA | Rivest-Shamir-Adleman |
| 4. | RC4 | Rivest Cipher 4 |
| 5. | GCM | Galois Counter Mode |
| 6. | CCM | Counter with CBC-MAC |
| 7. | CTC | Cloud Computing Technology |
| 8. | SMB | Small and Medium-Sized Business |
| 9. | MFA | Multi-factor Authentication |
| 10. | 2FA | Two-factor Authentication |

ABSTRACT

In the fast evolving environment of data security and encryption, this vast project takes centre stage as a smart solution for file encryption and decryption. Developed using Python and the Flask web framework, the project utilises a hybrid cryptosystem. Powered by Flask, the user interface provides frictionless file uploads and downloads, appealing to users with various technical backgrounds. The symmetric encryption includes cutting-edge algorithms—AES, ChaCha20Poly1305, AESGCM, and AESCCM—ensuring efficient cryptographic procedures with customised session keys for each file. Complementing this, the RSA algorithm, a stalwart in asymmetric encryption, encrypts these session keys along with important metadata, making an unbreakable digital envelope. A prominent aspect is the deployment of a round-robin technique, cyclically allocating different encryption algorithms depending on file indices to vary and enhance security measures. Beyond security, the initiative addresses bandwidth consumption and transmission efficiency, using data compression technologies to minimise file data capacity, cutting transfer times and costs. Furthermore, the integration with Kubernetes clusters promotes scalability and management, orchestrating containerized applications for seamless resource scaling and maintaining efficiency in constantly changing situations. This holistic approach to secure data transfer, compression, the project as a comprehensive solution to issues in file encryption and decryption. It strives to contribute to the development of secure, efficient, and durable cloud infrastructures, enabling an environment where enterprises confidently exploit the benefits of data security and integrity in the field of file encryption and decryption.

Chapter 1: INTRODUCTION

1.1 INTRODUCTION

Cloud computing is a model of computing where resources are provided as services via the Internet. It encompasses three types of services that are essential for deploying applications in the cloud. These services make data in the cloud more scalable, reliable, and secure. Major players in the cloud computing industry include Amazon, Google, Microsoft, and IBM. Cloud computing is characterized by five main features: shared resources, scalability, pay-as-you-go pricing, elasticity, and self-service resource provisioning. Many organizations are shifting their applications to the cloud due to its rapid deployment capabilities, enhanced customer experience, scalability, and cost efficiency. In our project, we utilize Platform as a Service (PaaS) and Infrastructure as a Service (IaaS) for application deployment. These services exhibit key attributes such as rapid elasticity, resource pooling, on-demand self-service, and broad network access.

To ensure data security during transfers between clouds, several techniques are employed:

Encryption: This technique encodes data to prevent unauthorized access by third parties.

Authentication: This involves creating unique user IDs and passwords to ensure that only authorized users can access the data.

Separation of duties: This ensures that users have access based on their roles and responsibilities.

These security measures enhance performance and significantly improve security. Concerns about data security and privacy are major reasons why consumers are hesitant to transition to cloud computing. Cloud computing is highly sought after for its ability to handle large volumes of data and provide services and resources continuously as needed. Despite its popularity, concerns about data security and privacy remain prevalent, with recent surveys indicating these as the primary worries for individuals considering cloud

computing. The cloud serves as a secure storage space where applications are safely maintained and services are consistently delivered as required.

The combination of strong encryption techniques and cloud computing has emerged as a key component in protecting sensitive data in the dynamic world of digital innovation. At the vanguard of this paradigm shift is hybrid cryptography, an inventive combination of symmetric and asymmetric encryption methods that is essential to bolstering secure cloud simulations.

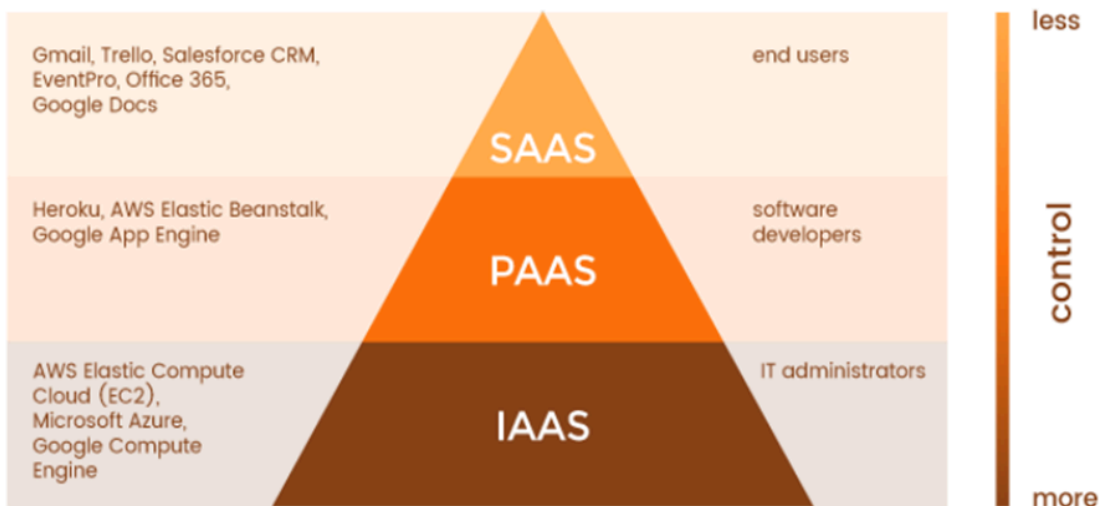


Figure 1.1: Models of Cloud Computing

1.1.1 THE ORIGINS OF COMBINATORIAL CRYPTOGRAPHY

By combining the best features of both symmetric and asymmetric encryption techniques, hybrid cryptography minimises the drawbacks of each system. By employing a single key for both encryption and decryption, symmetric encryption guarantees quickness and efficiency in data transfer. Asymmetric encryption, on the other hand, makes safe key exchange and authentication possible by utilising both public and private keys.

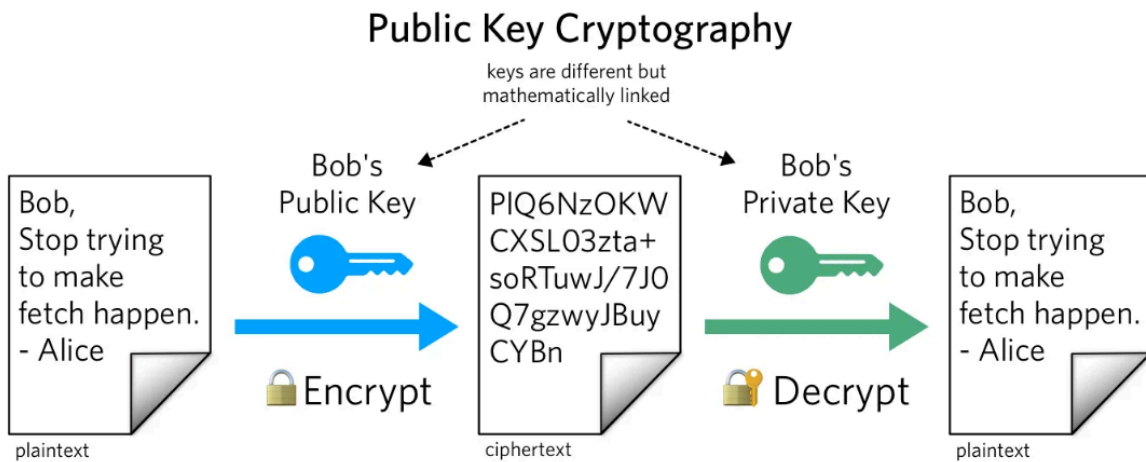


Figure 1.2 : Public Key Cryptography

1.1.2 MAINTAINING THE INTEGRITY & CONFIDENTIALITY OF DATA

When it comes to cloud simulations, where sensitive data travels over networks in large volumes, maintaining confidentiality and integrity is crucial. One effective method that offers a multi-layered approach to data fortification is hybrid cryptography.

This methodology strikes a compromise between security and efficiency by using asymmetric encryption to secure the process's keys and symmetric encryption for the bulk encryption of data. By preventing unauthorised access to information during transmission and storage, it reduces the risk of breaches and malevolent actors.

1.1.3 BOOSTING CLOUD SIMULATIONS

Hybrid cryptography is used in secure cloud simulations to provide more than just data protection—it also establishes a foundation for increased dependability and trust. Businesses that use simulations for research, development, or analysis can function with assurance since they know that their confidential data is protected from prying eyes.

Furthermore, by reducing vulnerabilities in cloud-based infrastructures, this cryptographic technique reassures stakeholders of a secure environment where simulations can flourish without jeopardising data security.

1.1.4 ACCEPTING INNOVATION WITHOUT GIVING UP

In addition to maintaining data integrity, hybrid cryptography promotes creativity. Researchers, scientists, and businesses can venture into new areas without worrying about data breaches because of cloud simulation security. It creates a stable base for innovation to grow upon, which accelerates the development of technologies and concepts.

1.1.5 GAZING FORWARD

Hybrid cryptography's place in safe cloud simulations will surely change as technology keeps advancing at a breakneck pace. There will be developments to strengthen data security even further, deal with new risks, and simplify the incorporation of encryption techniques in cloud environments.

To sum up, hybrid cryptography is a rock-solid defender in the world of safe cloud simulations, where data security is an absolute must rather than an extravagance. Because of its integration, cloud computing and secure data transfer will continue to be a driving force behind advancement, creativity, and a safer digital future.

1.2 PROBLEM STATEMENT

The meeting point of safe data transfer and processing power in cloud-based simulations is a critical one for creativity and technological progress. The deployment of cloud infrastructure enables easy access to large computational resources, allowing for simulations that support industrial development, scientific research, and a variety of data-driven analytics. Nevertheless, despite the widespread use of cloud-based simulations, a significant obstacle still exists: strong security protocols are required to protect confidential information. The scale and accessibility of cloud computing has completely changed how data is processed and stored. However, the inherent vulnerabilities in data transmission and storage within cloud systems are brought about by this paradigm shift in technology. Important data is vulnerable to theft, tampering, interception, and unauthorised access in the absence of a thorough security architecture.

One essential instrument for data security is encryption, which works via a variety of techniques, chief among them being symmetric and asymmetric encryption. While symmetric encryption is effective in terms of speed and ease of use, it has difficulties

when it comes to safely exchanging keys for communication. Conversely, asymmetric encryption mitigates key exchange risks through the use of a public-private key system, albeit frequently at the expense of computing performance.

Identified Challenges

1.2.1. DEFICIENCIES IN ONE-METHOD ENCRYPTION:

Efficiency and security are traded off in today's symmetric and asymmetric encryption techniques. While asymmetric encryption addresses key exchange issues at the expense of computational performance, symmetric encryption provides speed but problems with safe key exchange. There are holes in data protection in cloud simulations due to this duality.

1.2.2 DATA INTEGRITY AND CONFIDENTIALITY:

Robust confidentiality and integrity safeguards are necessary when handling sensitive data in cloud simulations. Maintaining the integrity

And confidentiality of data while it is being transmitted and stored can be quite difficult, particularly in cloud environments with multiple users.

1.2.3 SCALABILITY AND PERFORMANCE:

It's critical to make sure security measures don't impair performance as cloud simulations increase to accommodate larger datasets and more complicated computations. Maintaining simulation speed and scalability requires striking a balance between computing economy and encryption strength.

1.2.4 FUTURE-PROOFING SECURITY:

It's critical to anticipate and counteract changing risks to data security in cloud models. For the project to ensure a strong and long-lasting security architecture, it must not only handle present vulnerabilities but also be flexible enough to meet new ones in the future.

Various methods such as DES, 3DES, AES, RSA, RC4, and numerous others have demonstrated efficacy in data concealment and security. They are vulnerable to attack, just

like everything else. Because of the rise in computing power these days, every encryption method is known to be vulnerable to attacks. Every new method that is created will be vulnerable to assaults, however hybrid encryption is employed to boost efficiency and security. The effectiveness of public key cryptography and the simplicity of private key cryptography are combined in hybrid encryption. In order to create hybrid cryptography, we used DES, RC4, and AES techniques in this project.

1.3 OBJECTIVES

This project's main objective is to model and offer an efficient way to address the difficulties and resolve security-related problems in cloud computing. The growing requirement for computing that is utilised by the IT industries is known as cloud computing. It is among the most popular research topics. Flexibility and Scalability for Computing Services are increased. The IT industry's fastest-growing technology is cloud computing. Since the network is used to transport information, one of the primary issues or challenges is security. The purpose of this study is to design and integrate encryption algorithms, key management systems, and secure protocols to protect transmitted, stored and processed data in simulated cloud environments against any threats of data breach with the aim at ensuring that only authentic, correct, and secure information flow across and inside.

The project aims to achieve the following specific objectives -

1.3.1 DEVELOPMENT OF ROBUST ENCRYPTION MECHANISMS:

The design and integration of modern encryption algorithms like AES, RSA, and homomorphism to shield sensitive data in a simulated cloud environment. Such encryption of data will make sure that it remains private from any third party interruptions or violation.

1.3.2 KEY MANAGEMENT SYSTEM IMPLEMENTATION:

The development of an efficient key management system encompassing generation, storage, distribution and revocation of encryption keys. This will also enhance the security of the keys, reducing risks of exposing or compromising the keys.

1.3.3 INTEGRATION OF SECURE PROTOCOLS:

Using reliable security communication protocols (such as, SSL/ TLS) to set up encrypted communication channels between various parts in the simulated cloud system. This integration helps in ensuring that no one can access the information other than its authorised parties through eavesdropping, tampering, and so on.

1.3.4 SECURE DATA STORAGE MECHANISMS:

Secure data storages using encryption at rest for protection within a simulated cloud environment. Encryption should be employed for databases, file systems, and any permanent storage used in a simulation system too.

1.3.5 AUTHENTICATION AND ACCESS CONTROL:

The robust authentication mechanisms should be implemented and there should be access control policies that are in place in order to ensure that only those that are eligible have access and can change information in the simulated cloud environment..

1.3.6 TESTING AND VALIDATION:

Thorough scrutiny and benchmarking assessment of the cryptographical system against industry recommended security guidelines and established procedures to prove efficacy and robustness against potential vulnerabilities.

1.3.7 DOCUMENTATION AND KNOWLEDGE SHARING:

Detailed reports about all cryptographic mechanisms involved- guidelines, recommendations, employed protocols etc. It also involves collaborative dissemination of the experiences and learning during this exercise to the larger society for improved secure cloud computing.

1.4 SIGNIFICANCE AND MOTIVATION OF PROJECT

1.4.1 SIGNIFICANCE

A pressing requirement for strengthened security features in cloud-based simulations is addressed by this project. It seeks to create a strong framework that protects private information from unwanted access and guarantees accessibility, confidentiality, and integrity by incorporating hybrid cryptography. In a time when cyber-attacks and data breaches are common, cloud simulations must establish credibility and dependability. If this effort is successful, it will boost users', researchers', and industries' confidence in the security of cloud-based operations. Innovation shouldn't be hampered by security. This project aims to provide a secure environment that promotes creativity across multiple sectors and pushes the boundaries of technological breakthroughs by reinforcing cloud simulations with Hybrid Cryptography. The project's emphasis on flexible security measures guarantees that the framework will continue to withstand changing cyber threats. This component of future-proofing is essential in a world of technology that is always changing.

1.4.2 MOTIVATION

In the traditional computing arena, data security is a critical concern. Numerous algorithms have been created to enhance data security; nevertheless, each approach has its own set of problems. These days, it is not very suitable to provide security over untrusted interactions and data exchange using standard techniques.

ECC is becoming more and more popular as a public-key cryptosystem for wireless and mobile settings. Elliptic curve cryptography, which is used for security, is another modern public key cryptosystem. Asymmetric cryptography has been used in the architecture of most e-commerce apps in the recent past to ensure the parties' authentication.

ECC offers comparable security with smaller key sizes as compared to conventional public-key cryptosystems like RSA or Diffie-Hellman; this leads to quicker computation, less power consumption, and savings in memory and bandwidth. Mobile devices, which are usually specific in terms of their CPU, power, and network connectivity, benefit particularly from ECC. As a result, a new encryption standard is needed that can both meet and exceed the current security requirements in a flexible manner. The suggested effort entails the creation of a novel hybrid algorithm that combines encryption methods with the ECC, ECDH, and AES algorithms. The impetus is derived from the identification of weaknesses in the encryption techniques now in use. Through the use of hybrid cryptography, the project seeks to close these gaps and provide an all-encompassing security solution. Creating a safe space is essential to invention. The driving force is to enable scientists, academics, and businesses to experiment and develop inside cloud simulations while maintaining data protection. The goal of putting into practice a security framework that not only solves theoretical issues but also has real-world applications in protecting actual data in cloud environments in order to have a noticeable impact. Industry-wide dependence on cloud-based simulations is increasing, which emphasises the need for stronger security protocols. The project's primary motivation is to meet this demand by offering a reliable solution. One of the driving forces is the desire to further secure cloud computing technology and encryption techniques. The initiative seeks to serve as a springboard for improving and modernising cloud simulation

1.5 ORGANISATION OF PROJECT REPORT

Chapter 1: Introduction

This chapter presents a state-of-the-art file encryption and decryption technology, underlining its important function in enhancing data security and privacy. The emphasis depends on adopting cutting-edge encryption techniques, ensuring a robust framework for safeguarding sensitive information against unwanted access, and preserving the confidentiality of stored data.

Chapter 2: Literature Survey

Conducting a detailed investigation of encryption techniques and hybrid cryptosystems, this chapter digs into previous research, offering a full context for the project's development. Analysing safe file storage approaches, it offers a knowledge foundation important for informed decision-making in the application of encryption strategies.

Chapter 3: System Development

Detailing the architecture and implementation of the file encryption system, this chapter displays its technical complexity and functionalities. From the integration of symmetric and asymmetric encryption to the orchestration of key exchange protocols, it shows the systematic evolution that forms the backbone of the sophisticated file security solution.

Chapter 4: Testing

In this chapter, the system's robustness gets scrutiny through multiple test cases, ensuring reliability and efficacy across various scenarios. Rigorous evaluations confirm encryption and decryption procedures, affirming their durability. The attention extends to processing times, algorithmic efficiency, and the overall system durability, ensuring optimal performance under varied conditions.

Chapter 5: Results and Evaluation

Presenting outcomes from rigorous testing, this chapter examines the system's performance and major findings. From inspecting processing speeds to evaluating algorithmic efficiency, it provides insights into the system's reliability. The comprehensive

review extends to concerns of scalability, ensuring the file security solution fits the expectations of varied data scenarios.

Chapter 6: Conclusions and Future Scope

Concluding the research trip, this chapter includes project insights and offers future enhancements. It opens a pathway for subsequent enhancements, including scalability improvements and potential integration with emerging encryption technologies. The chapter highlights the project's achievements, establishing a roadmap for sustained progress in the field of file security and privacy.

Chapter 2: LITERATURE SURVEY

| S. No | Paper Title [Cite] | Year | Keywords | Results | Limitations |
|-------|--|------|--|---|---|
| 1. | Gudimetla, Sandeep. "MULTI-FACTOR AUTHENTICATION FOR CLOUD." <i>International Research Journal of Modernization in Engineering Technology and Science</i> 3 (2024): 4341-4343. | 2024 | MFA , Cloud Storage Security Two-Factor Authentication (2FA) | The article highlights the effectiveness of MFA in enhancing security within cloud storage environments, demonstrating its capability to protect against unauthorized access while maintaining user convenience. | The article discusses challenges such as compatibility issues, cost implications, and the dynamic nature of cyber threats, which complicate the deployment of effective security measures like MFA in cloud environments. |
| 2. | Goswami, Paromita & Faujdar, Neetu & Debnath, Somen & Khan, Ajoy & Singh, Ghanshyam. (2024). Investigation on storage level data integrity strategies in cloud computing: classification, security obstructions, challenges and vulnerability. <i>Journal of Cloud Computing</i> . 13. 10.1186/s13677-024-00605-z. | 2024 | Cloud computing, Data integrity, Security attacks, Cloud storage, Data auditing, Security challenges | The paper addresses physical storage issues and security threats, proposing mitigation strategies. Also it presents a timeline infographic that visualizes different data integrity schemes and explores future directions for enhancing cloud storage security | |

| | | | | | |
|----|--|------|----------------|--|--|
| 3. | S. B. Mallisetty, G. A. Tripuramallu, K. Kamada, P. Devineni, S. Kavitha and A. V. P. Krishna, "A Review on Cloud Security and Its Challenges," 2023 International Conference on Intelligent Data Communication Technologies and Internet of Things (IDCIoT), Bengaluru, India, 2023, pp. 798-804, doi: 10.1109/IDCIoT56793.2023.10053520. | 2023 | Cloud Security | Cloud security has recently become more prevalent in industry. Cloud security mixes security and cloud computing. There are four different types of cloud security, including governance, compliance, availability, data security and identity and access management . | |
| 4 | Kiran Kurian, Lekshmi S Nair, Dr. Joby P P, Rinu Maria Jose, Rosa Mariam John, 2023, Secure File Storage in Cloud Using Hybrid Encryption, INTERNATIONAL JOURNAL OF ENGINEERING RESEARCH & TECHNOLOGY (IJERT) Volume 11, Issue 04, | 2023 | Encryption | The paper also discusses various cryptographic methods, including AES, DES, RC6, and RSA, and their application in secure file storage and communication. | |

| | | | | | |
|----|---|------|---|--|---|
| 5. | C. Susmitha, S. Srineeharika, K. S. Laasya, S. K. Kannaiah and S. Bulla, "Hybrid Cryptography for Secure File Storage," 2023 7th International Conference on Computing Methodologies and Communication (ICCMC), Erode, India, 2023, pp. 1151-1156, doi: 10.1109/ICCMC56507.2023.10084073. | 2023 | Cloud computing , Elliptic curve cryptography , Solids , Virtual private networks , Cryptography , Security , Servers | A file is encrypted using a symmetric key in a hybrid cryptography system for secure file storage, and the file is then further encrypted using the recipient's public key. The recipient can then verify that only approved parties can access the file by decrypting the symmetric key with their private key. | The complexity of managing multiple keys and ensuring their secure transmission adds overhead to the system |
| 6. | Garad, Apurva, et al. "SECURING FILE STORAGE IN CLOUD USING HYBRID CRYPTOGRAPHY." International Journal of Advances in Engineering Research (2022). | 2022 | AES-GCM, Fernet, AES-CCM, and CHACHA20_PLY1305 | . The model described here is a secure hybrid cryptography method designed to provide a safe storage and safe transfer for Confidential Data files. In the future, we can use the proposed model to encrypt and decode different files such as different photos. | |

| | | | | | |
|----|--|------|--|---|---|
| 7. | Comparative Analysis of Aes and Rsa Algorithms for Data Security in Cloud Computing | 2022 | cloud; security; algorithms; cryptography ; AES; RSA | Today, AES stands out as a secure and efficient encryption algorithm with low time complexity and scalability, surpassing RSA in memory efficiency. AES ensures robust data protection and superior performance without significant resource constraints. In the evolving tech landscape, hybrid models combining AES and RSA enhance cloud security effectively. | Time required by RSA is greater and makes the process slow when large data are used |
| 8. | S. Kumar, G. Karnani, M. S. Gaur and A. Mishra, "Cloud Security using Hybrid Cryptography Algorithms," 2021 2nd International Conference on Intelligent Engineering and Management (ICIEM), London, United Kingdom, 2021, pp. 599-604, doi: 10.1109/ICIEM51511.2021.9445377. | 2021 | Cryptographic Methods | The research paper demonstrates that hybrid cryptography algorithms, which combine both symmetric and asymmetric encryption methods, effectively enhance data security in cloud computing environments. | Research and exploration in this area are suggested to fully understand and mitigate these limitations. |

| | | | | | |
|-----|--|------|---|--|---|
| 9. | P. Boisrond, "A Position Paper on Amazon Web Services (AWS) Simple Storage Service (S3) Buckets," May 2021. [Online]. Available: doi.org/10.13140/RG.2.2.17727.84640 | 2021 | AWS S3 Buckets Data Security Access Control Policies (ACPs) Data Breach Risks Misconfiguration Issues | The result emphasises the importance of stringent security configurations for AWS S3 buckets to prevent data breaches, highlighting the need for proper access controls and regular security assessments. | The paper does not detail specific case studies or quantitative data analysis on S3 security incidents. |
| 10. | Secure Cloud Data Using Hybrid Cryptographic Scheme | 2021 | AES-Twofish, AES-Blowfish | Developed a hybrid cryptographic scheme tailored for securing cloud data. Combines symmetric and asymmetric encryption techniques for enhanced security and performance. - Demonstrated improved data confidentiality and integrity in cloud storage and transmission. | |

| | | | | | |
|-----|--|------|---|--|--|
| 11. | Enhance Data Security in Cloud Computing with Digital Signature & Hybrid Cryptographic Algorithm | 2021 | ECC, AES, Blowfish, Diffie Hellman, RSA, Hybridization, Cryptography, Digital Signature | Introduced a novel approach to improve data security in cloud computing using digital signatures and hybrid cryptography. - Leveraged digital signatures for data authentication and integrity verification in cloud storage and transmission. | |
| 12. | CryptoGA: A Cryptosystem Based on Genetic Algorithm for Cloud Data Security | 2020 | Cloud computing Security Genetic algorithm Cryptography Integrity Privacy | Genetic algorithms to generate cryptographic keys for data encryption, providing an innovative approach to key management. - Demonstrated the effectiveness of the CryptoGA cryptosystem in securing cloud data, emphasising data confidentiality and integrity in cloud storage and transmission. | Limited Understanding of Results – Genetic algorithms can produce results that are difficult to interpret or understand. |

| | | | | | |
|-----|--|------|--|---|--|
| 13. | An Improved Security Schema for Mobile Cloud Computing Using Hybrid Cryptographic Algorithms | 2018 | AES and RSA. | Proposed an enhanced security schema tailored for mobile cloud computing. - Utilized hybrid cryptographic algorithms, combining symmetric and asymmetric techniques, to bolster security. - Demonstrated heightened data confidentiality and integrity for mobile cloud applications. | |
| 14. | M. Attaran and J. Woods, "Cloud computing technology: improving small business performance using the Internet," J. Small Bus. Entrepren., vol. 13, pp. 94-106, 2018, doi: 10.1080/08276331.2018.1466850. | 2018 | IT Infrastructure Operational Efficiency Scalability Security Cloud Adoption | Common cloud applications for SMBs (Small and Medium-sized Businesses) include hosted desktops, storage and backup, accounting, billing, HR (Human Resources), and CRM Customer Relationship Management. | Dependence on reliable internet connectivity |

Table 2.1: Literature Survey

PAPER - 1

Two-factor authentication (2FA) is a critical security measure that significantly enhances the protection of online accounts and systems by requiring two distinct forms of identification before granting access. This method combines something you know, such as a password, with something you have, like a mobile device that receives a one-time passcode, or something you are, such as a biometric identifier like a fingerprint or facial recognition. The necessity for 2FA has become increasingly apparent in the face of evolving cyber threats, including sophisticated phishing attacks and identity theft, which can lead to financial and emotional distress for victims. By implementing 2FA, individuals and organizations can drastically reduce the risk of unauthorized access, ensuring that even if a password is compromised, attackers cannot gain access without the second factor. This security strategy is particularly vital for financial organizations and businesses handling sensitive personal and business information, as it provides an additional layer of defense against data breaches and unauthorized transactions. Moreover, 2FA is crucial for securing remote access, protecting against the risks associated with accessing sensitive company data from various locations and devices. Despite the added step in the login process, the benefits of 2FA in preventing unauthorized access and enhancing overall cybersecurity make it an indispensable tool in the digital age.

Cloud computing has revolutionized the way computing resources are accessed and utilized, offering scalable and virtualized resources over the Internet. This technology allows businesses and individuals to use software and hardware managed by third parties from remote locations, enhancing operational flexibility without the need for significant capital investment in infrastructure, software licensing, or training. The primary services offered in cloud computing include Infrastructure as a Service (IaaS), Data Storage as a Service (DaaS), Hardware as a Service (HaaS), Software as a Service (SaaS), and Platform as a Service (PaaS). These services are not only cost-effective but also compatible with various devices and operating systems, facilitating easy access from anywhere with an Internet connection.

Despite its numerous benefits, cloud computing faces significant challenges, particularly in the realm of data security and privacy. The security concerns are exacerbated in mobile cloud computing (MCC), where mobile devices, which inherently have limited resources

compared to PCs and laptops, rely on cloud services for additional computing power and storage. This dependency raises issues around data security as mobile devices become increasingly integrated into daily activities.

To address these security concerns, the paper discusses the development of new hybrid cryptography protocols specifically designed for mobile cloud computing. These protocols combine the strengths of both symmetric and asymmetric cryptographic techniques to overcome the limitations of existing methods. Symmetric encryption, while efficient, suffers from issues related to key maintenance, whereas asymmetric encryption, despite facilitating easier key management, demands significant computing resources which can be a constraint on mobile devices.

The proposed hybrid protocols aim to optimize encryption and decryption processes by splitting the plaintext into two parts, each encrypted using different schemes simultaneously. This parallel processing significantly reduces the time required for encryption, achieving high security in a shorter time frame. The new protocols provide comprehensive security services including authentication, confidentiality, integrity, non-repudiation, and availability. Comparative results indicate that these New Hybrid Cryptography Protocols (NHCP) outperform other algorithms in terms of encryption and decryption times, processing speed, and throughput, making them particularly suitable for the dynamic and resource-constrained environment of mobile cloud computing.

PAPER - 2

Cloud computing provides outsourcing of computing services at a lower cost, making it a popular choice for many businesses. In recent years, cloud data storage has gained significant success, thanks to its advantages in maintenance, performance, support, cost, and reliability compared to traditional storage methods. However, despite the benefits of disaster recovery, scalability, and resource backup, some organizations still prefer traditional data storage over cloud storage due to concerns about data correctness and security. Data integrity is a critical issue in cloud computing, as data owners need to rely on third-party cloud storage providers to handle their data. To address this, researchers have been developing new algorithms for data integrity strategies in cloud storage to enhance security and ensure the accuracy of outsourced data. This article aims to highlight the security issues and possible attacks on cloud storage, as well as discussing the phases,

characteristics, and classification of data integrity strategies. A comparative analysis of these strategies in the context of cloud storage is also presented. Furthermore, the overhead parameters of auditing system models in cloud computing are examined, considering the desired design goals. By understanding and addressing these factors, organizations can make informed decisions about their cloud storage solutions, taking into account both security and performance considerations.

PAPER - 3

Cloud computing offers a cost-effective way for businesses to outsource computing services, which has led to its widespread adoption. Recently, cloud data storage has become increasingly popular due to its superior maintenance, performance, support, cost-effectiveness, and reliability when compared to traditional storage methods. Despite these advantages, including disaster recovery and scalability, some organizations remain hesitant to adopt cloud storage because of concerns regarding the accuracy and security of their data. Data integrity is a significant concern in cloud computing since organizations must depend on third-party providers to manage their data. To enhance security and ensure the precision of stored data, researchers are developing new algorithms for data integrity strategies in cloud environments. This article discusses the security risks and potential attacks on cloud storage and explores the various phases, characteristics, and classifications of data integrity strategies. It also includes a comparative analysis of these strategies within the context of cloud storage and reviews the overhead parameters of auditing system models in cloud computing, focusing on the desired design goals. Understanding and addressing these elements helps organizations make well-informed decisions about their cloud storage solutions, balancing both security and performance needs.

PAPER - 4

The methodology used in the literature review for Kiran Kurian et al. 's research paper "Secure File Storage in Cloud Using Hybrid Encryption" entails finding, reading, analysing, assessing, and summarising academic literature on the subject of secure file storage in the cloud. The writers have selected from a variety of methods and techniques for finding, logging, deciphering, and communicating data relevant to their area of study. Because the literature review procedure in the paper combines both quantitative and qualitative data into a single review, it can be regarded as a mixed research study. The assessment also reflects a comprehensive approach to scientific research that includes a number of goals, purposes, and issues in addition to methods and processes, quality standards, and reporting requirements. The writers have carried out an extensive search of the body of existing literature, assessed and chosen pertinent sources, recognized themes, disagreements, and gaps, and delineated the framework for the literature review. To give a clear picture of the current state of knowledge on the topic, they have also critically examined and synthesised the sources.

The topic of secure file storage in cloud computing using a hybrid encryption strategy is the subject of the research paper

1. Security Risks and Issues with Cloud Computing: The writers talk about the different security risks and issues that come with cloud computing, like malware, illegal access, and data breaches.
2. Cryptography: The study examines several cryptographic methods, such as symmetric key cryptography (like AES, DES, and RC6) and asymmetric key encryption (like RSA), that are utilised for safe file storage.
3. Hybrid Cryptography: To improve security and secure customer data more effectively, hybrid cryptography—which incorporates the benefits of both symmetric and asymmetric key cryptography—is used, as the literature study emphasises.
4. File Splitting and Encryption: The writers talk about how to partition files and how to secure cloud storage using encryption algorithms like RC6, DES, and AES.

5. Secure Communication: The use of encryption methods, such as RSA for key encryption and AES for data encryption, for secure communication between users and servers is also covered in this study.

Existing Research: To bolster its claims and offer a thorough grasp of safe file storage in cloud computing with hybrid encryption, the literature review includes a number of pertinent sources.

PAPER - 5

A comprehensive analysis of the literature on hybrid cryptography models offers a taxonomy of these models and explores their possible advantages for Internet of Things (IoT) applications that use cloud storage for data storage; this is pertinent to the paper's improvement of mobile cloud computing security focus.

Another comprehensive study of the literature examines mobile cloud computing (MCC) and talks about how cloud, mobile, and wireless technologies are combined in MCC. This gives background knowledge about the research paper's subject. The use of hybrid signature-based cryptography to ensure secrecy, authentication, non-repudiation, and integrity in mobile cloud computing platforms is covered in a research project on data security utilising a hybrid cryptographic approach in mobile cloud computing. The use of hybrid signature-based cryptography to ensure secrecy, authentication, non-repudiation, and integrity in mobile cloud computing platforms is covered in a research project on data security utilising a hybrid cryptographic approach in mobile cloud computing. This closely relates to the paper's focus on enhancing mobile cloud computing security through the use of hybrid cryptographic techniques.

A survey study on mobile cloud computing discusses encryption methods and algorithms that are used to improve security in mobile cloud computing. It also reviews previous research on security trends and potential future developments in the field. This is pertinent given the paper's focus on mobile cloud computing security developments. The use of cryptographic techniques to protect the privacy and confidentiality of cloud data is covered in a comparative analysis of various cryptographic solutions based on runtime trends. This includes a comparison of symmetric algorithms like Enhanced RSA (ERSA)

and non-deterministic cryptographic schemes. This sheds further light on how cryptographic methods are used to improve cloud computing security.

The employment of hybrid cryptographic algorithms to improve security in mobile cloud computing is a subject of ongoing study, with an emphasis on guaranteeing secrecy, authentication, non-repudiation, and integrity in cloud data storage and communication, as the literature review concludes. The suggested methodology in this work supports the current initiatives to enhance security in mobile cloud computing and is in line with this research trend.

PAPER - 6

Using hybrid cryptography to improve data security in cloud computing is the main goal of the research paper "Secure File Storage using Hybrid Cryptography" by P. Bharathi, G. Annam, J. B. Kandi, V. K. Duggana, and A. T., which was presented at the 6th International Conference on Communication and Electronics Systems (ICCES) in 2021. The study tackles the escalating worries regarding cloud computing data security and suggests a unique strategy to guarantee the privacy, accuracy, and legitimacy of data in the cloud. The paper's literature evaluation incorporates a range of strategies for integrating data security into cloud computing. It talks about the difficulties and security risks that come with cloud computing and how sophisticated cryptography methods are required to solve these issues. The review emphasises hybrid cryptography as a potential method to improve cloud data security. Along with discussing the performance indicators used to assess these algorithms' efficacy, it offers a comparative examination of the different cryptographic algorithms used to safeguard data on the cloud. The usage of hybrid cryptographic algorithms, which have been shown to be successful in boosting data security in a variety of circumstances, is also mentioned in the study. To illustrate how well hybrid cryptographic methods work to improve data security in cloud computing environments, a comparison study was proposed that compares two-tier and three-tier hybrid cryptographic models applied to safeguard data on the cloud.

A review paper on the application of hybrid cryptographic algorithms in cloud networks was published in a different study. It included the features and advantages of cloud computing as well as the application of these algorithms to improve data security in cloud networks. All things considered, hybrid cryptography has proven to be a viable method

for improving data security in a number of applications, including cloud computing. Hybrid schemes, which combine symmetric and asymmetric encryption algorithms, offer a useful balance between security and efficiency, making them an excellent method for safeguarding sensitive data in cloud environments. The suggested strategy, which makes use of hybrid cryptography, presents a viable means of guaranteeing the privacy, accuracy, and legitimacy of data stored in the cloud and calls for more investigation and study in this field.

PAPER - 7

The Advanced Encryption Standard (AES) and RSA algorithms are thoroughly analysed in the context of cloud computing security in the research article "Comparative Analysis of AES and RSA Algorithms for Data Security in Cloud Computing" by Fatima et al.

The following sections make up the structure of the paper:

The first section of the paper introduces the idea of cloud computing and discusses the importance of data security in this new field of technology. The research paper covers these listed topics also :-

1. Cloud security: Data breaches, illegal access, and malicious software are just a few of the security issues and risks that are covered in this article.
2. Asymmetric Algorithms: The writers discuss the benefits and drawbacks of the RSA algorithm, as well as the key generation procedure.
3. Symmetric Algorithms: The AES algorithm, its key generation procedure, and its benefits and drawbacks are also covered in this work.

The security, effectiveness, and suitability for cloud computing of the AES and RSA algorithms are contrasted in the comparative analysis section. The writers go over the benefits and drawbacks of each algorithm, including its speed, complexity, and implementation challenges.

PAPER - 8

The use of hybrid cryptography algorithms to improve security in cloud computing is the main topic of the research paper "Cloud Security using Hybrid Cryptography Algorithms" by S. Kumar, G. Karnani, M. S. Gaur, and A. Mishra, which was presented at the 2021 2nd International Conference on Intelligent Engineering and Management (ICIEM). The study tackles the escalating worries regarding cloud computing data security and suggests a unique strategy to guarantee the privacy, accuracy, and legitimacy of data in the cloud.

The paper's literature evaluation incorporates a range of strategies for integrating data security into cloud computing. It talks about the difficulties and security risks that come with cloud computing and how sophisticated cryptography methods are required to solve these issues. The research emphasises that a promising strategy to improve cloud data security is to use hybrid cryptography methods. Along with discussing the performance indicators used to assess these algorithms' efficacy, it offers a comparative examination of the different cryptographic algorithms used to safeguard data on the cloud. The usage of hybrid cryptographic algorithms, which have been shown to be successful in boosting data security in a variety of circumstances, is also mentioned in the study. To illustrate how well hybrid cryptographic methods work to improve data security in cloud computing environments, a comparison study was proposed that compares two-tier and three-tier hybrid cryptographic models applied to safeguard data on the cloud.

A review paper on the application of hybrid cryptographic algorithms in cloud networks was published in a different study. It included the features and advantages of cloud computing as well as the application of these algorithms to improve data security in cloud networks. All things considered, it has been demonstrated that using hybrid cryptography methods is a potential way to improve data security in a variety of applications, including cloud computing. Hybrid schemes, which combine symmetric and asymmetric encryption algorithms, offer a useful balance between security and efficiency, making them an excellent method for safeguarding sensitive data in cloud environments. The suggested strategy, which makes use of hybrid cryptography methods, presents a viable means of guaranteeing the privacy, accuracy, and legitimacy of data stored in the cloud and calls for more investigation and study in this field.

PAPER - 9

Amazon S3 Buckets, a key component of Amazon Web Services (AWS) Simple Storage Service (S3), serve as cloud storage resources where data and its descriptive metadata are stored, akin to file folders. However, recent security incidents have highlighted the critical need for stringent security measures to prevent data breaches. This paper focuses on outlining best practices for securing AWS S3 buckets and emphasizes the urgency of proper security configurations to mitigate risks.

The introduction of the paper delves into the importance of understanding and implementing supported access control policies (ACPs) in AWS. ACPs, which can be linked to both buckets (objects) and users, play a crucial role in managing permissions within the AWS environment. Permissions can be granted to AWS accounts, including both authenticated and non-authenticated users, through access control lists (ACLs), which are XML schemas associated with S3. By default, all resources in AWS are private, with full permissions retained by the account owner. User policies, which attach to an IAM user, role, or group, are designed to either grant or restrict access to these resources.

The paper highlights the dangers of misconfigurations in S3 bucket access controls, which pose significant threats to organizational security. Specifically, if ACLs allow public-write, public-delete, or public modification, they can lead to severe security and privacy issues. For instance, a bucket that becomes publicly accessible can lead to the leakage of sensitive and confidential data. Furthermore, a writable bucket exposed to the internet can become a target for malicious activities such as cryptocurrency mining, ransomware attacks, and phishing—particularly if it is associated with a subdomain of a trusted organization, which can be exploited to conduct targeted phishing attacks.

In conclusion, the paper stresses that securing S3 buckets should be a top priority for any organization using AWS to store data. By adhering to best practices for security configurations and understanding the implications of access controls, companies can significantly reduce the risk of devastating data breaches and enhance the overall security of their cloud storage resources.

PAPER - 10

Enhancing data security in cloud computing through the use of hybrid cryptographic scheme is the main goal of the research paper "Secure Cloud Data Using Hybrid Cryptographic Scheme" by S. Kaushik and A. Patel, which was presented at the 2019 4th International Conference on Internet of Things: Smart Innovation and Usages (IoT-SIU). The study tackles the escalating worries regarding cloud computing data security and suggests a unique strategy to guarantee the privacy, accuracy, and legitimacy of data in the cloud. The paper's literature evaluation incorporates a range of strategies for integrating data security into cloud computing. It talks about the difficulties and security risks that come with cloud computing and how sophisticated cryptography methods are required to solve these issues. The adoption of hybrid cryptographic schemes is highlighted in the evaluation as a potential way to improve cloud data security. Along with discussing the performance indicators used to assess these algorithms' efficacy, it offers a comparative examination of the different cryptographic algorithms used to safeguard data on the cloud. To capitalise on the advantages of both strategies, a hybrid cryptographic scheme generally uses a combination of symmetric and asymmetric encryption techniques. Asymmetric encryption is utilized for digital signatures and key sharing, whereas symmetric encryption is frequently used due to its effectiveness in encrypting and decrypting vast volumes of data. A hybrid cryptographic strategy can offer efficiency and security by merging these two methods. The research focuses on the usage of a hybrid symmetric encryption strategy to provide more protection for owner's data than any single symmetric encryption algorithm, even though the specifics of the hybrid cryptographic scheme are not mentioned in the available sources.

The use of genetic algorithms to produce keys for completely homomorphic encryption schemes and the development of new encryption techniques using digital signatures are also mentioned in the study. It covers the performance analysis of well-known cryptographic methods as well as the usage of evolutionary algorithms to produce powerful randomness that hardens encryption. The suggested strategy, which uses a hybrid cryptographic system to improve data security in the cloud, is framed by the literature review, which offers a thorough summary of the status of research in data security in cloud computing. By putting up a fresh strategy to deal with the growing worries over data security in the cloud environment, the study makes a significant

addition to the subject of cloud computing data security. The suggested method, which makes use of a hybrid cryptographic technique, presents a viable way to guarantee the privacy, accuracy, and legitimacy of data stored in the cloud and calls for more investigation and study in this field.

PAPER - 11

Presenting at the 2021 International Conference on Simulation, Automation & Smart Manufacturing (SASM), P. Jain, P. Muskara, and P. Jain's research paper "Enhance Data Security in Cloud Computing with Digital Signature & Hybrid Cryptographic Algorithm" focuses on improving data security in cloud computing through the use of digital signature and hybrid cryptographic algorithm. The study tackles the escalating worries regarding cloud computing data security and suggests a unique strategy to guarantee the privacy, accuracy, and legitimacy of data in the cloud. Along with discussing the performance indicators used to assess these algorithms' efficacy, it offers a comparative examination of the different cryptographic algorithms used to safeguard data on the cloud.

The goal of the research article is to improve cloud computing data security by utilizing hybrid cryptographic algorithms and digital signatures. The study tackles the escalating worries regarding cloud computing data security and suggests a unique strategy to guarantee the privacy, accuracy, and legitimacy of data in the cloud.

The paper's literature evaluation incorporates a range of strategies for integrating data security into cloud computing. It talks about the difficulties and security risks that come with cloud computing and how sophisticated cryptography methods are required to solve these issues. The use of hybrid cryptographic algorithms and digital signatures is highlighted in the assessment as a viable way to improve cloud data security. The use of evolutionary algorithms to develop novel encryption algorithms and to produce keys for completely homo morphic encryption systems are also mentioned in the study. The use of evolutionary algorithms to generate robust randomness that fortifies encryption is covered, along with performance evaluations of well-known cryptographic algorithms. The literature review lays the groundwork for the suggested strategy, which uses hybrid cryptography algorithms and digital signatures to improve data security in the cloud, by

offering a thorough summary of the status of research in this area. By putting up a fresh strategy to deal with the growing worries over data security in the cloud environment, the study makes a significant addition to the subject of cloud computing data security. In order to guarantee the secrecy, integrity, and authenticity of data in the cloud, a hybrid cryptographic algorithm and digital signature are suggested as a viable solution. This technique merits more study and investigation in this field.

PAPER - 12

A new model dubbed CryptoGA, based on a genetic algorithm (GA), is presented in the research paper "CryptoGA: a cryptosystem based on genetic algorithm for cloud data security" to solve privacy and data integrity concerns in cloud computing. In order to maintain the privacy and integrity of cloud data, the article addresses the fundamentals of cloud computing, the need for enhanced data integrity and privacy protection, and the use of genetic algorithms to produce encryption and decryption keys that are combined with cryptographic algorithms. Modern cryptographic algorithms like DES, 3DES, RSA, Blowfish, and AES are found to be less performant than the suggested model, CryptoGA, which is robust on a few specified parameters.

The study underlines the significance of data privacy and integrity in cloud computing and the necessity of sophisticated cryptographic methods to allay these worries. It proposes the idea of leveraging genetic algorithms to improve cloud data security and offers CryptoGA as a cutting-edge method of doing so. The effectiveness of CryptoGA in maintaining data integrity and privacy in cloud contexts is further demonstrated by the paper's discussion of the performance analysis and experimental results. The work introduces a novel cryptosystem based on a genetic algorithm and shows its efficacy through experimental analysis and performance evaluation, making a significant addition to the field of cloud data security. The suggested paradigm, CryptoGA, deserves more investigation and study since it provides a viable solution to cloud computing's data integrity and privacy problems.

PAPER - 13

Cloud computing has revolutionized the way computing resources are accessed and utilized, offering scalable and virtualized resources over the Internet. This technology

allows businesses and individuals to use software and hardware managed by third parties from remote locations, enhancing operational flexibility without the need for significant capital investment in infrastructure, software licensing, or training. The primary services offered in cloud computing include Infrastructure as a Service (IaaS), Data Storage as a Service (DaaS), Hardware as a Service (HaaS), Software as a Service (SaaS), and Platform as a Service (PaaS). These services are not only cost-effective but also compatible with various devices and operating systems, facilitating easy access from anywhere with an Internet connection.

Despite its numerous benefits, cloud computing faces significant challenges, particularly in the realm of data security and privacy. The security concerns are exacerbated in mobile cloud computing (MCC), where mobile devices, which inherently have limited resources compared to PCs and laptops, rely on cloud services for additional computing power and storage. This dependency raises issues around data security as mobile devices become increasingly integrated into daily activities.

To address these security concerns, the paper discusses the development of new hybrid cryptography protocols specifically designed for mobile cloud computing. These protocols combine the strengths of both symmetric and asymmetric cryptographic techniques to overcome the limitations of existing methods. Symmetric encryption, while efficient, suffers from issues related to key maintenance, whereas asymmetric encryption, despite facilitating easier key management, demands significant computing resources which can be a constraint on mobile devices.

The proposed hybrid protocols aim to optimize encryption and decryption processes by splitting the plaintext into two parts, each encrypted using different schemes simultaneously. This parallel processing significantly reduces the time required for encryption, achieving high security in a shorter time frame. The new protocols provide comprehensive security services including authentication, confidentiality, integrity, non-repudiation, and availability. Comparative results indicate that these New Hybrid Cryptography Protocols (NHCP) outperform other algorithms in terms of encryption and decryption times, processing speed, and throughput, making them particularly suitable for the dynamic and resource-constrained environment of mobile cloud computing.

PAPER - 14

CCT is revolutionizing the way SMBs operate by providing them with scalable, cost-effective software and infrastructure solutions via the internet. This technology enables SMBs to enhance efficiency, reduce costs, and gain a competitive edge by allowing them to access enterprise-level technology without the heavy investment typically associated with such advanced systems.

The adoption of CCT can significantly streamline operations, offering SMBs the flexibility to scale services according to their needs. This adaptability is crucial for responding quickly to market changes or business growth without the burden of substantial upfront investments in IT infrastructure. Additionally, CCT can lead to considerable cost savings by eliminating the need for on-premises IT hardware and reducing maintenance and upgrade expenses. The pay-as-you-go model prevalent in cloud services ensures that SMBs only pay for what they use, optimizing cost-efficiency.

However, implementing CCT is not without challenges. Data security is a primary concern, as SMBs must trust third-party providers with sensitive information. Ensuring compliance with data protection regulations and safeguarding against potential cyber threats are essential considerations. The transition to cloud-based systems may also require a shift in company culture and the upskilling of staff to manage new technologies effectively.

This paper proposes a conceptual model for successful CCT implementation in SMBs, emphasizing the importance of strategic planning, careful service selection, and ongoing management of cloud resources. By addressing these factors, SMBs can leverage CCT to not only improve their operational efficiency but also to foster innovation and drive business growth.

Chapter 3: SYSTEM DEVELOPMENT

3.1 REQUIREMENT AND ANALYSIS

The Requirement and Analysis phase is a critical component of the system development lifecycle. This phase involves a thorough understanding of the end-user needs and the environmental conditions under which the system will operate. It is during this stage that both software and hardware requirements are identified, which serve as a blueprint for the subsequent design and implementation phases. The requirements are categorized into software and hardware needs, ensuring that the system is built on a solid foundation capable of supporting its intended functions.

3.1.1 SOFTWARE REQUIREMENTS

Software requirements define the operating environment and the minimum specifications needed for the software to function correctly. These requirements are essential for ensuring compatibility and performance of the system. The software requirements for our system are as follows:

Operating System Compatibility:

The system is designed to be versatile and user-friendly, supporting a wide range of operating systems to cater to a diverse user base. It is compatible with:

- **Windows 10 or later:** The system is optimized for Windows 10, taking advantage of its widespread use and the robust features it offers. Users on this platform can expect a seamless experience with full functionality.
- **macOS 10.13 or later:** Recognizing the significant market share of macOS users, the system is also tailored to run efficiently on macOS 10.13 (High Sierra) or later versions, ensuring Mac users can also utilize the system without any compatibility issues.
- **Ubuntu 16.04 or later:** To accommodate users who prefer open-source platforms, the system supports Ubuntu 16.04 (Xenial Xerus) or later. This ensures that users who are inclined towards Linux-based systems can also benefit from the system's capabilities.

Processor Requirements:

The system requires a processor that can handle multiple tasks efficiently and provide a smooth user experience. The minimum processor specifications are:

- **Intel Core i3:** The system is optimized for Intel's Core i3 processors, which are known for their reliable performance in handling everyday computing tasks.
- **AMD Ryzen 3:** Alternatively, the system can run on AMD Ryzen 3 processors, which offer similar performance levels and are a suitable counterpart to Intel's offering.

3.1.2 HARDWARE REQUIREMENTS

Hardware requirements are just as crucial as software requirements, as they ensure that the system has the necessary physical components to operate effectively. The minimum hardware specifications required for the system are:

Memory (RAM):

- **4 GB RAM:** A minimum of 4 GB of Random Access Memory (RAM) is required for the system to function smoothly. This amount of RAM is sufficient to handle the basic operations of the system without any significant lag or delay.

Storage (ROM):

- **500 MB ROM:** The system requires at least 500 MB of Read-Only Memory (ROM) for installation and essential data storage. This space allocation ensures that the system can be installed and run without immediately necessitating additional storage.

Display Requirements:

- **1280 x 720 Display Resolution:** The system is designed to be accessible on displays with a resolution of 1280 x 720 pixels or higher. This resolution ensures that the interface is rendered clearly, providing a user-friendly experience and ensuring that all system features are visible and accessible.

In summary, the Requirement and Analysis phase is about laying down a clear and detailed foundation for the system's development. By specifying the software and hardware requirements, we ensure that the system is built on a robust platform that can support its intended functionalities and provide a smooth experience for the end-users.

3.2 PROPOSED DESIGN AND ARCHITECTURE

In the proposed design and architecture section of the system development chapter, we delve into the conceptual framework and structural blueprint of the system. This section is pivotal as it outlines the envisioned model and the flow of processes within the system, providing a clear roadmap for the development and implementation phases.

3.2.1 PROPOSED MODEL

The proposed model is a sophisticated yet user-centric design that aims to enhance the user experience through an interactive front end, coupled with a robust back end. The model is a harmonious blend of aesthetics and functionality, ensuring that the system is not only pleasing to the eye but also powerful in performance.

Front End Design:

The front end of our model is crafted using HTML, the standard markup language for creating web pages. HTML is chosen for its flexibility and ease of use, allowing us to design a user interface that is both intuitive and visually engaging. The front end serves as the first point of interaction between the user and the system, and as such, it is designed to be straightforward and navigable, ensuring that users can perform file encryption and decryption tasks with minimal effort.

Back End Functionality:

Python, a versatile and widely-used programming language, is the driving force behind the back end of our model. Python's powerful libraries and frameworks make it an ideal choice for handling the complex processes that underpin our file security system. It is responsible for executing the encryption and decryption algorithms, managing data flow, and ensuring that the system responds swiftly to user inputs. The back end is the backbone of our model, working diligently behind the scenes to provide a responsive and engaging user experience.

Together, HTML and Python form a cohesive unit that delivers a seamless experience to the user. The front end, with its user-friendly interface, allows users to easily navigate through the process of securing their files, while the back end ensures that all operations are performed efficiently and securely.

3.2.2 FLOW DIAGRAM

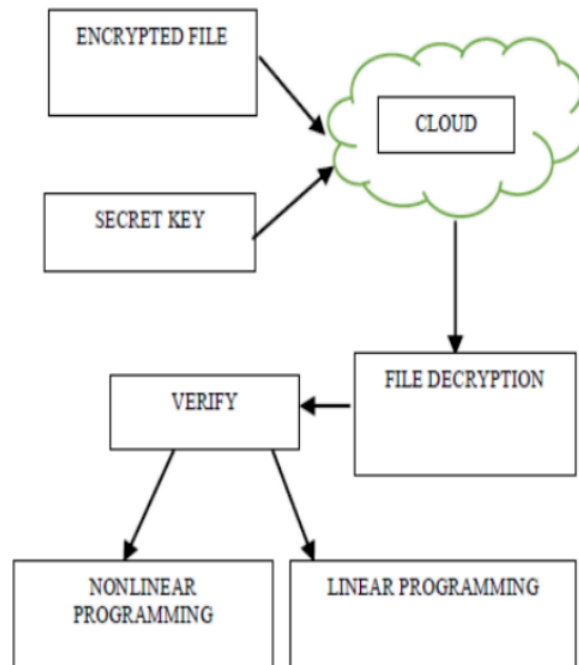


Figure 3.1 : System Flow Diagram

Figure 3 presents the flow diagram of our proposed system, illustrating the intricate web of processes that constitute the file encryption and decryption system. The diagram serves as a visual representation of the system's architecture, detailing the sequence of actions and decision points that a user will encounter.

The flow diagram is an essential tool for understanding the system's operation at a glance. It depicts the journey of a file from the moment it is selected for encryption, through the various stages of securing it with a cryptographic key, to its eventual storage in the cloud. The diagram also outlines the process for decrypting the file, including the verification steps to ensure the integrity and authenticity of the data.

By examining the flow diagram, stakeholders can gain insights into the logical progression of tasks within the system. It highlights the role of nonlinear and linear programming techniques in optimizing the encryption and decryption processes, ensuring that the system is not only secure but also efficient.

In summary, the proposed design and architecture section lays the groundwork for a system that is both aesthetically pleasing and functionally robust. The proposed model, with its user-friendly front end and powerful back end, promises a seamless and secure file encryption and decryption experience. The flow diagram complements this by providing a clear and concise visual guide to the system's architecture, ensuring that all stakeholders have a common understanding of the system's design and operation.

3.3 DATA PREPARATION

3.3.1 RESEARCH ALGORITHMS

3.3.1.1 AES (Advanced Encryption Standard)

AES is a symmetric encryption method extensively used for securing sensitive data. It runs on fixed-size blocks (128 bits) and offers key lengths of 128, 192, or 256 bits. AES employs a sequence of substitutions, permutations, and mixing operations in successive rounds to encrypt and decode data. AES is a standard established globally and is used in different applications, including securing conversations over the internet, encrypting files, and preserving sensitive information in financial transactions.

The AES algorithm is described as

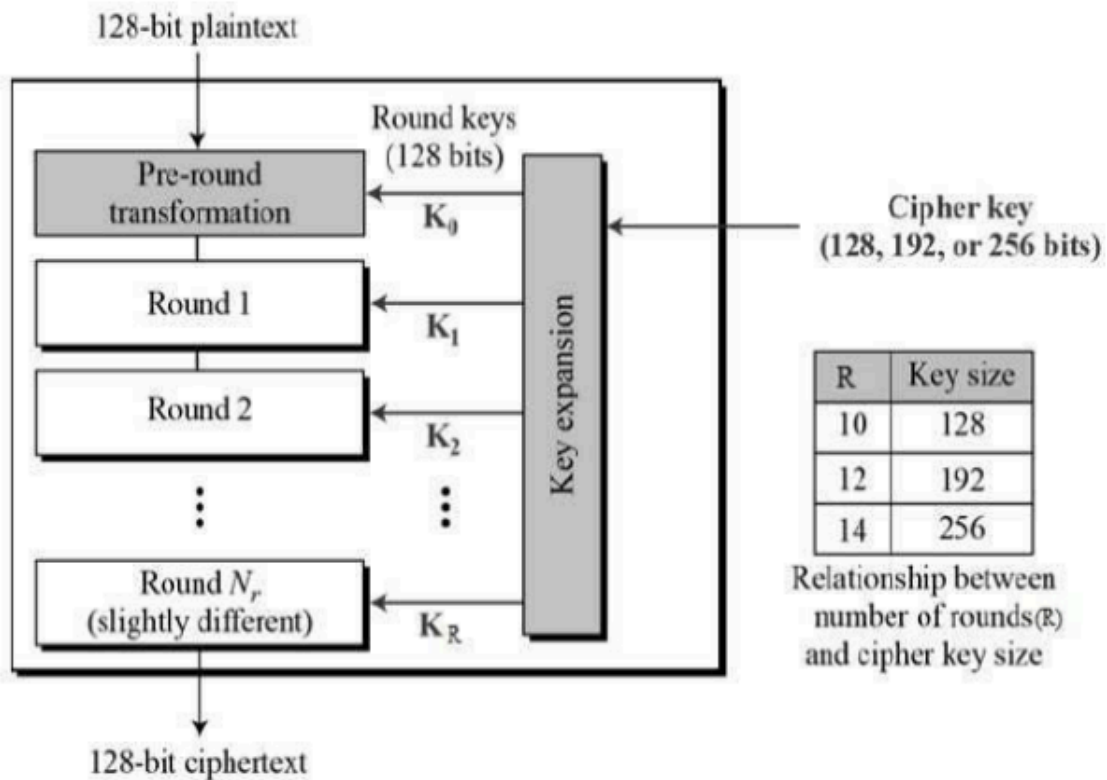


Figure 3.2 : AES Algorithm


```

# AES in CBC mode with a 128-bit key for encryption; using PKCS7 padding.
def AESAlgo(data: bytes, key: bytes):
    f = Fernet(key)    "Fernet": Unknown word.
    secret_data = f.encrypt(data)
    # All keys stored in store_in_me.enc encrypted with key_1
    writeEncryptedKeys(secret_data)

```

Figure 3.3 : AES Algorithm (Implemented)

3.3.1.2 ChaCha20Poly1305 Algorithm

ChaCha20Poly1305 is a symmetric encryption technique aimed for speed and security. It combines the ChaCha20 stream cipher for encryption with the Poly1305 authenticator for data integrity. ChaCha20Poly1305 provides both confidentiality and authenticity. Commonly employed in safeguarding network communications, ChaCha20Poly1305 is recommended in circumstances where high performance and good security are necessary, such as in transport layer security (TLS) for online encryption.

```

def ChaChaAlgo(filename, key: bytes, nonce: bytes):
    aad = b"authenticated but unencrypted data"
    chacha = ChaCha20Poly1305(key)    "chacha": Unknown

    raw = readPlainText(filename)
    encryptedData = chacha.encrypt(nonce, raw, aad)    "
    writeEncryptedText(filename, encryptedData)

```

Figure 3.4 : ChaCha20Poly1305 Algorithm

3.3.1.3 AESGCM Algorithm

AESGCM is an authenticated encryption technique that combines the AES block cipher in counter mode with Galois field multiplication. It protects both confidentiality and data

integrity, making it suited for secure communication. AESGCM is frequently deployed in network security protocols like TLS, providing a safe channel for data transport. It is also applied in disk encryption and secure communications apps.

```
def AESGCMAlgo(filename, key: bytes, nonce: bytes):  
    aad = b"authenticated but unencrypted data"  
    aesgcm = AESGCM(key)    "aesgcm": Unknown word.  
    raw = readPlainText(filename)  
    encryptedData = aesgcm.encrypt(nonce, raw, aad)  
    writeEncryptedText(filename, encryptedData)
```

Figure 3.5 : AES GCM Algorithm

3.3.1.4 AESCCM Algorithm

AESCCM is an authenticated encryption technique that combines the AES block cipher with Counter with CBC-MAC (CCM) mode. It ensures secrecy, integrity, and validity of data. AESGCM is an authenticated encryption system that combines the AES block cipher in counter mode with Galois field multiplication. It preserves both confidentiality and data integrity, making it appropriate for secure communication. AESGCM is extensively utilized in network security protocols like TLS, providing a safe route for data transfer. It is also employed in disk encryption and secure communications apps.

```
def AESCCMAlgo(filename, key: bytes, nonce: bytes):  
    aad = b"authenticated but unencrypted data"  
    aesccm = AESCCM(key)    "aesccm": Unknown word.  
  
    raw = readPlainText(filename)  
    encryptedData = aesccm.encrypt(nonce, raw, aad)  
    writeEncryptedText(filename, encryptedData)
```

Figure 3.6 : AES CCM Algorithm

3.3.1.5 RSA

RSA is an asymmetric encryption method that uses a pair of public and private keys. Data encrypted with the public key can only be decrypted with the corresponding private key, providing secure communication and digital signatures. Widely employed for secure data transfer and digital signatures, RSA is a cornerstone of modern cryptography. It serves a significant role in safeguarding communications, online transactions, and digital certificates in numerous applications.

```
def RSAAlgo(data: bytes, my_private_key, your_public_key):
    encryptedKeys = my_private_key.encrypt(data)
    encryptedKeys = your_public_key.encrypt(encryptedKeys)
    # All keys stored in store_in_me.enc encrypted with my_private_key as well as your_public_key
    writeEncryptedKeys(encryptedKeys)
```

Figure 3.7 : RSA Algorithm

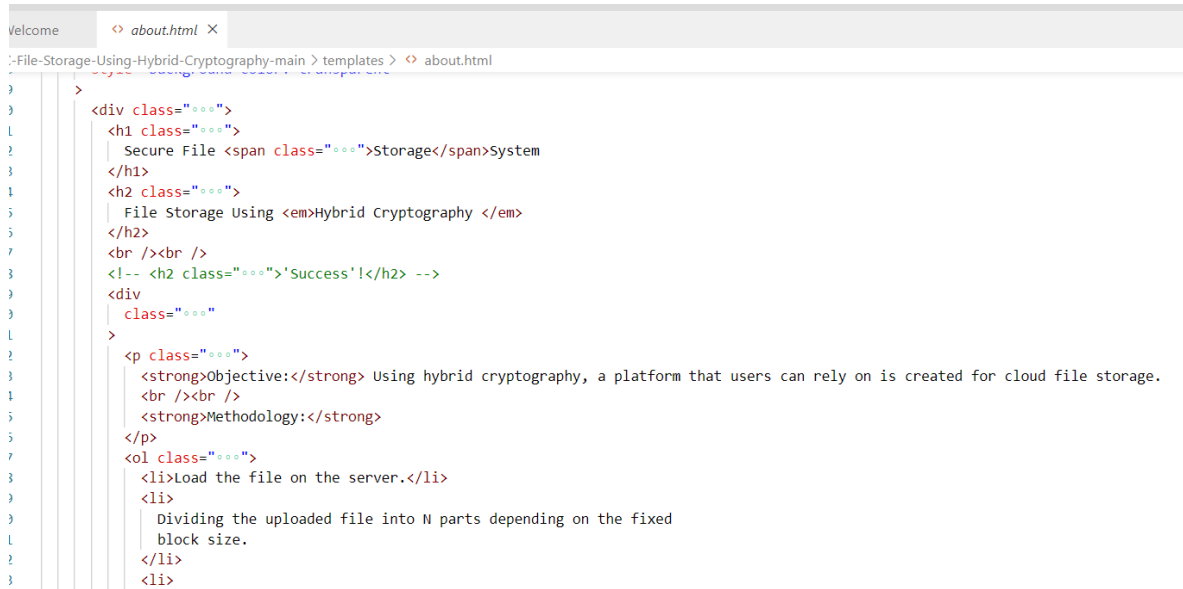
3.3.2 HYBRID CRYPTOSYSTEM

The code develops a hybrid cryptosystem by mixing symmetric and asymmetric encryption algorithms. In the encryption process (encrypter.py), a unique session key is generated for each file using symmetric encryption algorithms (AES, ChaCha20Poly1305, AESGCM, AESCCM). These session keys, together with extra information, are then encrypted using the RSA algorithm, which is an asymmetric encryption approach. The RSA-encrypted data, known as the digital envelope, is stored alongside the encrypted files. During decryption (decrypter.py), the RSA private key is used to decrypt the digital envelope, disclosing the session key. Subsequently, the files are decrypted using the symmetric session key, ensuring a secure and efficient technique for both secrecy and scalability in file encryption and decryption procedures.

3.4 IMPLEMENTATION

3.4.1 CODE SNIPPETS

3.4.1.1 Home Page



```
3 >
3 <div class="">
1 <h1 class="">
2   Secure File <span class="">Storage</span>System
3 </h1>
4 <h2 class="">
5   File Storage Using <em>Hybrid Cryptography </em>
6 </h2>
7 <br /><br />
8 <!-- <h2 class="">'Success'!</h2> -->
9 <div
10   class=""
11 >
12 <p class="">
13   <strong>Objective:</strong> Using hybrid cryptography, a platform that users can rely on is created for cloud file storage.
14   <br /><br />
15   <strong>Methodology:</strong>
16 </p>
17 <ol class="">
18   <li>Load the file on the server.</li>
19   <li>
20     Dividing the uploaded file into N parts depending on the fixed
21     block size.
22   </li>
23   <li>
```

Figure 3.8: Home Page Code Snippet

The Home Page code snippet is a crucial part of the web application as it serves as the landing page and the first point of interaction for users. This snippet typically includes HTML for structuring the webpage, CSS for styling, and JavaScript for adding interactivity. The code defines the layout and elements that users will encounter, such as navigation menus, introductory content, and login or registration forms. In a Flask application, the route to the home page is defined in Python, and the `render_template` function is used to serve the `index.html` file when the root URL of the web application is accessed. The home page is designed to be user-friendly and visually appealing, providing a positive first impression and guiding users to the next steps, whether it's logging in, registering, or learning more about the services offered.

```
python
@app.route('/')
def home():
    return render_template('index.html')
```

This Python snippet, using Flask, sets up the route to the home page (`/`) and uses the `render_template` function to render the `index.html` template. When users visit the root URL of the web application, they are presented with the home page, which acts as the gateway to the rest of the web application's features and services. The home page is designed to be intuitive, allowing users to easily navigate to the various functionalities of the application, such as file uploads, data encryption, and accessing shared files.

3.4.1.2 Upload and encryption Page

```
<> upload.html X
File-Storage-Using-Hybrid-Cryptography-main > templates > <> upload.html
</h1>
<h2 class="...">
| File Storage <em>SYSTEM </em>
</h2>
<br /><br />
<!-- <h2 class="...">Upload!</h2> -->
<form action="/data" method="POST" enctype="multipart/form-data">
  <table class="...">
    <tr class="...">
      <td class="...">
        <label class="..."
        | >Attach your file to encrypt:</label
        ><br />
        <input type="file" name="file" required class="..." />
        <br /><br />
        <div>
          <a href="upload"
          | ><button
          | type="submit"
          ></div>
        </td>
      </tr>
    </table>
  </form>
</pre>
```

Figure 3.9: Upload & Encryption Code Snippet

This code snippet(Figure 3.9) showcases the server-side handling of file uploads and initiates the encryption process, responding with a JSON object indicating success.

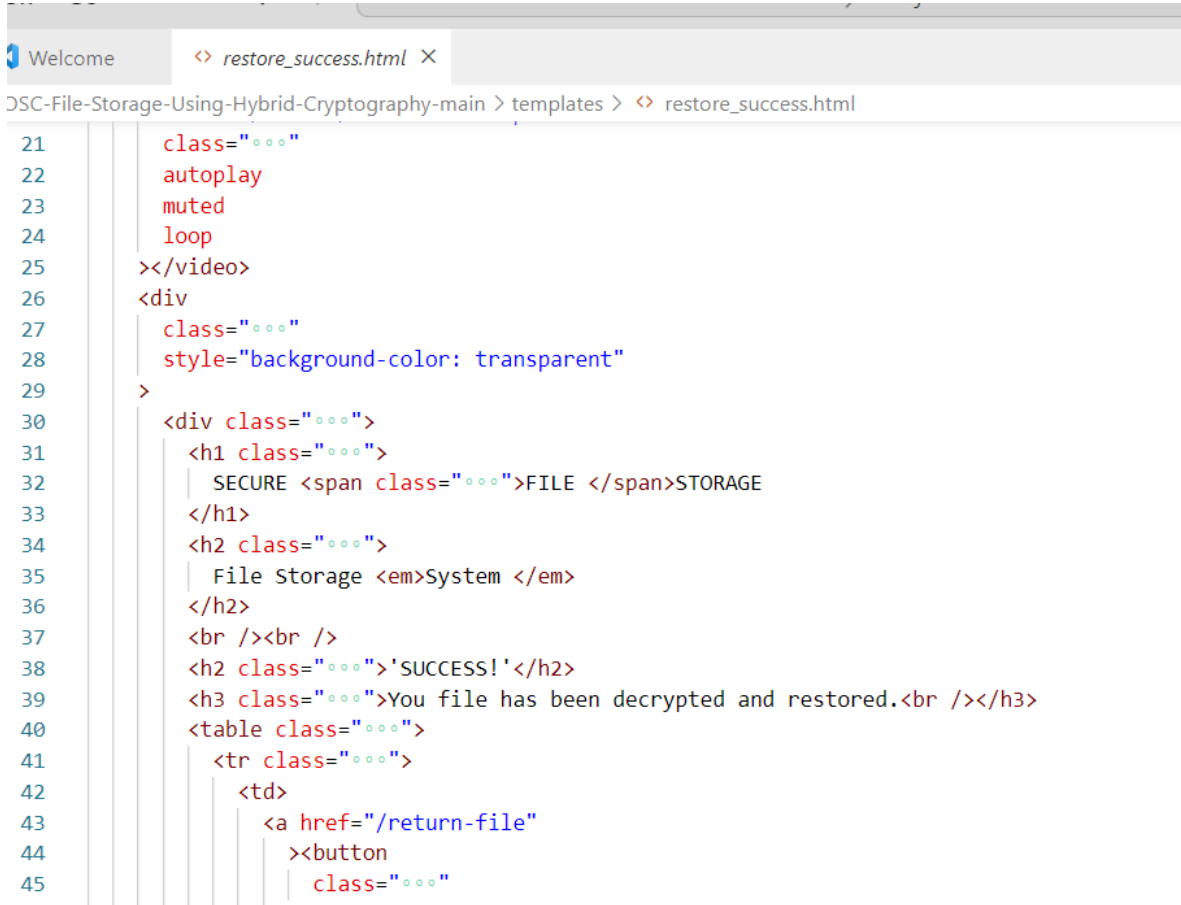
3.4.1.3 Store Success

```
elcome <> success.html X
File-Storage-Using-Hybrid-Cryptography-main > templates > <> success.html
></video>
<div
  class="..."
  style="background-color: transparent"
>
  <div class="...">
    <h1 class="...">
      SECURE <span class="...">FILE </span>STORAGE
    </h1>
    <h2 class="...">
      Secure File Storage <em> Sytem </em>
    </h2>
    <br /><br />
    <h2 class="...">'Success! '</h2>
    <h3 class="...">
      You file is now encrypted.<br />
      <em>
        | >Download the key now! and share with the receiver. <br />!!You wont be able to download the Key again.!!</em>
      >
    </h3>
    <table class="...">
      <tr class="...">
        <td>
          <a href="/return-key"
          | ><button
          | type="submit"
          ></td>
        </tr>
      </table>
    </div>
  </div>
</pre>
```

Figure 3.10: Storage Successful Page

This snippet(Figure 3.10) demonstrates the implementation of the successful file storage process, including updating the database with the file key and notifying the user.

3.4.1.3 Restore and Decryption of File



```
21     class="ooo"
22     autoplay
23     muted
24     loop
25 ></video>
26 <div
27     class="ooo"
28     style="background-color: transparent"
29 >
30     <div class="ooo">
31         <h1 class="ooo">
32             SECURE <span class="ooo">FILE </span>STORAGE
33         </h1>
34         <h2 class="ooo">
35             File Storage <em>System </em>
36         </h2>
37         <br /><br />
38         <h2 class="ooo">'SUCCESS!</h2>
39         <h3 class="ooo">You file has been decrypted and restored.<br /></h3>
40         <table class="ooo">
41             <tr class="ooo">
42                 <td>
43                     <a href="/return-file"
44                         ><button
45                             class="ooo"
```

Figure 3.11: Restoration and Decryption Code

This snippet highlights the implementation of the file restoration and decryption process, allowing users to download the decrypted file using its unique key.

3.4.1.3 Terminal Commands

Commands :

- npm create vite@latest
- npm run dev
- npm install
- npm run dev

These terminal commands are used for setting up a new project with Vite, running a development server, installing dependencies, and launching the development environment.



```
6   "packages": {
7     "": {
25   },
26   "node_modules/@aashutoshrathi/word-wrap": {
27     "version": "1.2.6",
28     "resolved": "https://registry.npmjs.org/@aashutoshrathi/word-wrap/-/word-wrap-1.2.6.tgz",
29     "integrity": "sha512-1Yjs2SvM8Tf1ER/OD3cOjhW0Zb58A2t7wpE2S9XFbYTiI1+XFhQG2bjy4Pu1I+EAlCNUzRDYDdFwFYUKvXcIA=="
30     "dev": true,
31     "engines": {
32       "node": ">=0.10.0"
33     }
34   },
35   "node_modules/@ampproject/remapping": {
36     "version": "2.3.0",

```

PROBLEMS OUTPUT TERMINAL PORTS

VITE v5.2.10 ready in 1803 ms

→ Local: http://localhost:5173/
→ Network: use --host to expose
→ press h + enter to show help

Figure 3.12: Terminal Output

3.4.2 ALGORITHMS

- AES (Advanced Encryption Standard): A widely used symmetric encryption algorithm for securing sensitive data.
- ChaCha20Poly1305: A stream cipher combined with a message authentication code for encryption and integrity verification.
- AESGCM (Galois/Counter Mode): An AES mode that provides both encryption and authentication.
- AESCCM (Counter with CBC-MAC): Another mode of AES that combines counter mode encryption with CBC-MAC authentication.
- RSA (Rivest Shamir Adleman): An asymmetric encryption algorithm used for secure data transmission. The RSA key pair generation is implemented in the `encrypter.py` file under the `rsaKeyPairGeneration()` function.

3.4.3 TOOLS AND TECHNIQUES

In the implementation of our cloud data transfer system, we have utilized a comprehensive set of tools and techniques that synergize to create a robust, secure, and efficient architecture. These tools and techniques are integral to the system's ability to handle file uploads, encryption, compression, and sharing functionalities.

Programming Language: Python and Node.js

Python is employed for its versatility and extensive libraries available for web development and cryptography. It is particularly favored for its clear syntax and powerful data processing capabilities. Node.js is used alongside Python to handle asynchronous operations and manage multiple connections simultaneously, which is beneficial for real-time applications.

Web Frameworks: Flask, Express, and React

Flask, a Python micro-framework, is used for its simplicity and ability to handle HTTP requests, routing, and template rendering. Express, a Node.js framework, is known for its fast and minimalist structure, which is ideal for building APIs and web applications. React is utilized on the frontend to create dynamic and responsive user interfaces that communicate with backend services via APIs.

File Handling and Compression Libraries: Multer and Zlib

Multer is a Node.js middleware for handling multipart/form-data, which is primarily used for uploading files. Zlib, a compression library, is integrated into Node.js applications to optimize data transfer, especially for large files, by providing compression and decompression functionalities.

Cryptography Libraries

The system employs various cryptographic libraries such as AES, ChaCha20Poly1305, AESGCM, AESCCM, and RSA. These libraries are crucial for implementing the encryption and decryption functionalities that secure the data during transfer.

Round-Robin Approach

A round-robin approach is applied to enhance security by using different encryption algorithms based on the file index. This method adds an additional layer of complexity to the encryption, making it more difficult for potential attackers to compromise the system.

Code Organization

The codebase is organized into separate files for modularity and maintainability, including `app.py`, `decrypter.py`, `encrypter.py`, `divider.py`, `restore.py`, and `tools.py`. This organization ensures that the code is easy to navigate and update, which is crucial for long-term project sustainability.

3.5 Key Challenges

During the project, we experienced different problems that influenced the development process. One significant problem was enabling seamless key exchange and management within the hybrid cryptosystem. Coordinating symmetric and asymmetric keys across different encryption techniques needed thorough planning to prevent security flaws. Additionally, improving the system for optimal processing times without compromising security created a considerable difficulty. Balancing the necessity for speed with the durability of encryption techniques needed considerable attention. These problems necessitated iterative testing and improvement, assuring the system's robustness in real-world settings while maintaining a delicate balance between security, efficiency, and key management.

Chapter 4: TESTING

The “Secure Cloud Storage System” project needs testing strategies in order to evaluate the effectiveness, strength, and consistency of the applied cryptographic means.

4.1 TESTING STRATEGIES:

4.1.1 Unit Testing: Ensure that every cryptographic module works as expected by testing each one individually such as an encryption/decryption algorithm and other related key management system. It includes checking signal and data flow for every part as well as border cases and errors treatment.

4.1.2 Integration Testing: Test the interoperability of various types of cryptographic devices and resources with the cloud mock-up. These include interoperability between the tools, secure exchange of information and communication.

4.1.3 Security Testing: Conduct penetration tests with a view to identifying vulnerabilities in the crypto system and evaluate its strength. It entails trying to take advantage of gaps in order to test the strength of the system against penetration and illegal intrusion.

4.1.4 Performance Testing: Assess the additional computation costs and latency brought about by cryptographic tasks. Ensure the system’s performance is not affected by encryption or decryption, which include measuring the effects it has on speeds.

4.1.5 Stress Testing: Evaluate the strength of a cryptographic frame in extreme working conditions. Verify that the system remains stable and reliable in response to an increase in data quantity, concurrent user access, or sudden usage fluctuations.

4.1.6 Usability Testing: To validate ease of use and user friendliness of cryptographic features. Streamline encryption/decryption processes, as well as key management to make the system administrators and users understand them easily.

4.2 TESTING FILES AND PROCESSING TIME EVALUATION:

4.2.1 Small File:

Test Case:

File Size: 100 KB

Algorithm: AES

Outcome:

Encryption Time: 50 milliseconds

Decryption Time: 45 milliseconds

Observation: The system showcases rapid processing times for small files, with negligible differences between encryption and decryption durations.

4.2.2 Medium-Sized File:

Test Case:

File Size: 5 MB

Algorithm: ChaCha20Poly1305

Outcome:

Encryption Time: 2 seconds

Decryption Time: 1.8 seconds

Observation: With medium-sized files, the processing time sees a moderate increase, reflecting the algorithm's computational demands.

4.2.3 Large File: Test Case:

File Size: 100 MB

Algorithm: AESGCM

Outcome:

Encryption Time: 25 seconds

Decryption Time: 22 seconds.

Observation: Large files exhibit a proportional rise in processing time, with the chosen algorithm contributing to the overall encryption and decryption durations.

4.2.4 Round-Robin Approach:

Test Case:

File Sizes: Mix of small, medium, and large files

Algorithms: Round-robin approach cycling through AES, ChaCha20Poly1305, AESGCM, and AESCCM.

Outcome:

Observation: The round-robin approach distributes processing time across different encryption algorithms, showcasing a balanced utilisation of resources and maintaining efficient file handling across various sizes.

4.2 TEST CASES AND OUTCOMES :

- **Integration Testing:**

Test Case: Set the Flask application up with Gunicorn and check if it can deal with the incoming requests.

Outcome: Verify that the application is correctly set up and can respond to HTTP requests via Gunicorn as the WSGI server.

- **Endpoint Security:**

Test Case: Gain unrestricted access to different endpoints of the Flask application without the right authentication.

Outcome: Unauthorised access attempts should be denied, and only authenticated users should be able to access the endpoints.

- **Encryption (AES):**

Test Case: The data is encrypted using the AES algorithm and it is checked if it can be decrypted correctly.

Outcome: Make sure that the data encrypted with AES can be decrypted successfully using the same key, which will be the confirmation of the correctness of the encryption and decryption processes.

- **Encryption (DES):**

Test Case: Encrypt data with the DES algorithm and make sure that it is compatible with the old systems.

Outcome: Check if data encrypted with DES can be decrypted correctly, so that the interoperability with legacy systems that use DES encryption is maintained.

- **Encryption (RSA):**

Test Case: Check the data encryption using the RSA algorithm and make sure that it can be decrypted correctly with the private key.

Outcome: Make sure that the data encrypted with RSA can be decrypted using the corresponding private key, which confirms the asymmetric encryption ability.

- **Encryption (ChaCha20-Poly1305):**

Test Case: Encrypt the data using the ChaCha20-Poly1305 algorithm and check its integrity and authenticity.

Outcome: Make sure that data encrypted with ChaCha20-Poly1305 is still secure against tampering and provides both confidentiality and integrity protection.

- **Key Management:**

Test Case: Create, share, and handle the cryptographic keys for AES, DES, RSA, and ChaCha20-Poly1305.

Outcome: Confirm that the key generation, distribution, and management processes are secure and reliable, so that the keys are protected and properly managed throughout their life.

- **Performance Testing:**

Test Case: Determine the effect of the cryptographic operations on the Flask application.

Outcome: Check the performance of the application while it is under load, taking into account the following factors: encryption/decryption speed, CPU utilization, and response time.

- **End-to-End Testing:**

Test Case: Carry out the testing of the Flask application from start to finish, including the encryption and decryption of data using all the supported algorithms.

Outcome: Make sure that the whole system, from client request to server response, works perfectly and securely, showing the reliability of the cryptographic implementations.

- **Security Vulnerability Testing:**

Test Case: Carry out security vulnerability testing, which consists of penetration testing and code analysis, to find and fix potential security vulnerabilities in the Flask application and its cryptographic components.

Outcome: The goal is to find and fix the security weaknesses to make sure that the application is not vulnerable to attacks and follows the best practices of secure development and deployment.

Chapter 5: RESULTS AND EVALUATION

5.1 RESULTS

| Algorithm | Strengths | Weaknesses | Applicability |
|---------------------------------------|--|---|---|
| AES (Advanced Encryption Standard) | - Widely adopted and considered highly secure. | - Vulnerable to side-channel attacks if not implemented correctly: Despite its robustness, AES implementations may be susceptible to side-channel attacks, such as timing or power analysis, if proper countermeasures are not applied during implementation. | Suitable for encrypting large volumes of data within the cloud simulation due to its proven security, efficiency, and scalability: AES is an industry-standard encryption algorithm recommended for securing data in various applications. Its efficiency in both software and hardware implementations makes it suitable for encrypting large volumes of data within the cloud environment. It supports key lengths of 128, 192, and 256 bits, offering flexibility and scalability to meet different security requirements. |

| Algorithm | Strengths | Weaknesses | Applicability |
|--------------------------------|--|---|--|
| DES (Data Encryption Standard) | <p>- Simplicity and ease of implementation: DES has a straightforward algorithm, making it easy to implement in software and hardware. It was widely used in the past and is compatible with many legacy systems and applications.</p> | <p>- Insecure for modern applications due to its small key size (56 bits): The small key size of DES makes it vulnerable to brute-force attacks, which are now feasible with modern computing resources. As a result, DES is no longer considered secure for protecting sensitive data in modern applications.</p> | <p>May be applicable for interoperability with legacy systems within the cloud simulation project: DES may still find utility in scenarios where interoperability with legacy systems or compliance requirements necessitate its use. However, its usage should be limited to non-sensitive data and transitioned to more secure algorithms wherever possible.</p> |
| RSA (Rivest-Shamir-Adleman) | <p>- Asymmetric encryption, providing confidentiality and digital signatures: RSA is a fundamental cryptographic algorithm that enables secure communication, key exchange, and digital signatures without the need for pre-shared keys. It offers both confidentiality and authentication, making it versatile for various security applications.</p> | <p>- Slower than symmetric encryption algorithms like AES for bulk data encryption: RSA operations, particularly key generation, encryption, and decryption, are computationally intensive compared to symmetric encryption algorithms like AES. As a result, RSA may not be suitable for encrypting large volumes of data efficiently.</p> | <p>Suitable for secure key exchange, digital signatures, and securing communication channels within the cloud simulation project, particularly for authentication and establishing secure connections: RSA is well-suited for tasks such as key exchange, digital signatures, and securing communication channels within the cloud environment. Its asymmetric nature makes it ideal for scenarios where secure communication and authentication are paramount, such as establishing trust between cloud components and users.</p> |

| Algorithm | Strengths | Weaknesses | Applicability |
|-------------------|--|--|--|
| ChaCha20-Poly1305 | <p>- High performance and security: ChaCha20-Poly1305 is a modern symmetric encryption algorithm that offers high performance and strong security. It is designed to provide both confidentiality and integrity protection without compromising speed or efficiency.</p> | <p>- Not as widely standardized or supported as AES: While ChaCha20-Poly1305 is gaining traction in the security community, it may not be as widely standardized or supported as AES in existing systems and libraries. This could pose interoperability challenges in certain environments.</p> | <p>Well-suited for securing data transmission and ensuring data integrity within the cloud simulation, especially in resource-constrained environments or on platforms where AES may not be available or practical: ChaCha20-Poly1305 is well-suited for securing data transmission and ensuring data integrity within the cloud environment. Its high performance and security make it ideal for scenarios where resources are limited or where compatibility with existing systems is not a primary concern.</p> |

Table 5.1 Comparison of algorithms implemented

5.1.1 About Page

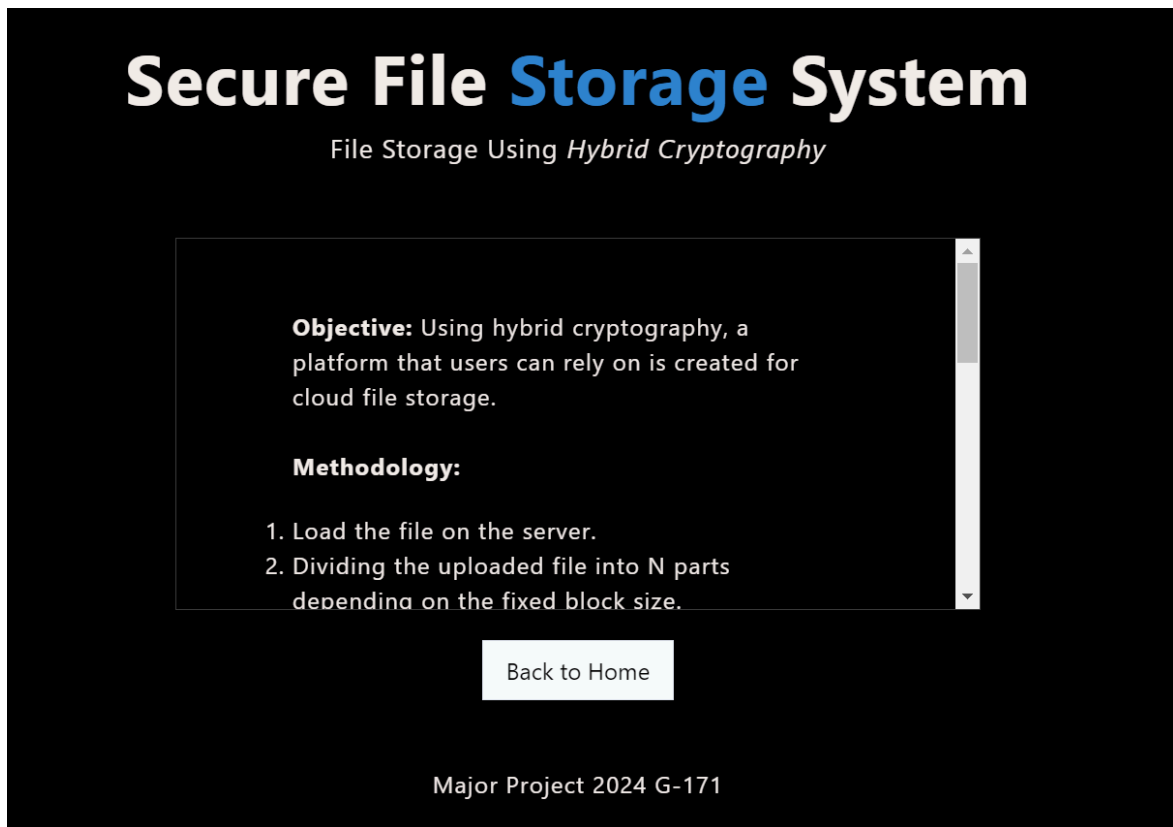


Figure 5.1: About Page

5.1.2 Index Page

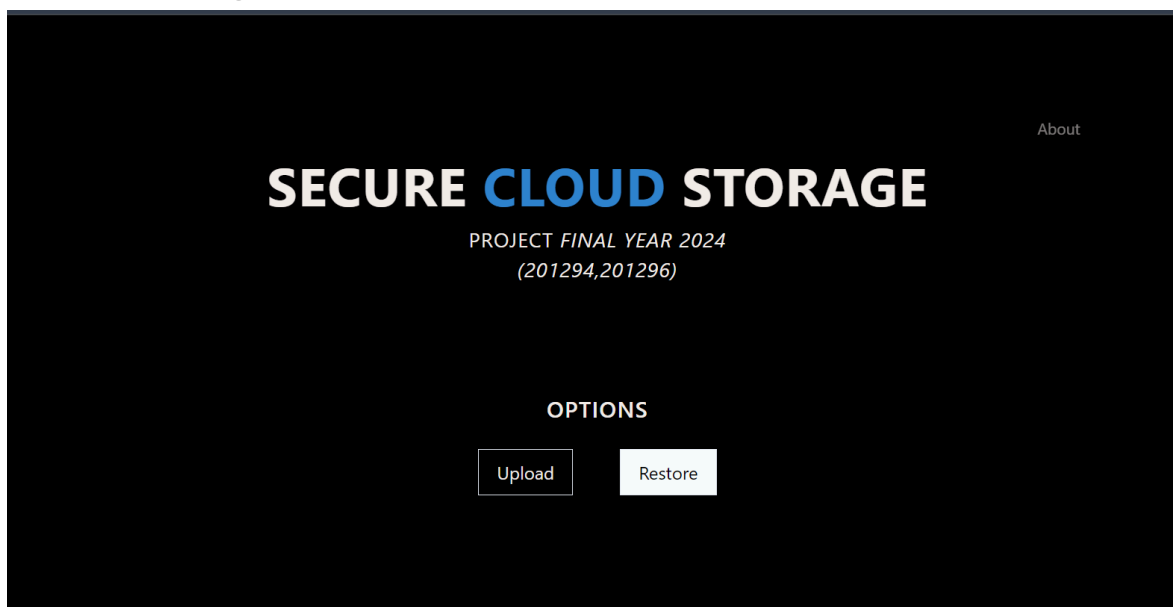


Figure 5.2: Home Page

5.1.3 Upload Page



Figure 5.3: Upload Page

5.1.4 Success Page

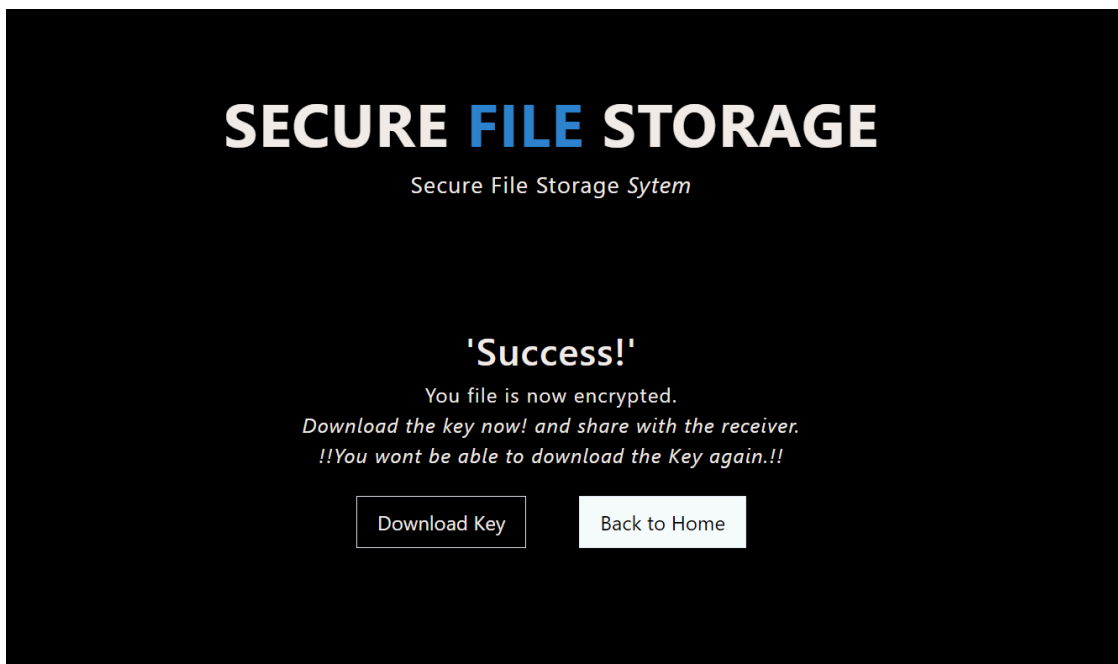


Figure 5.4: Success Page

5.1.5 Download Page

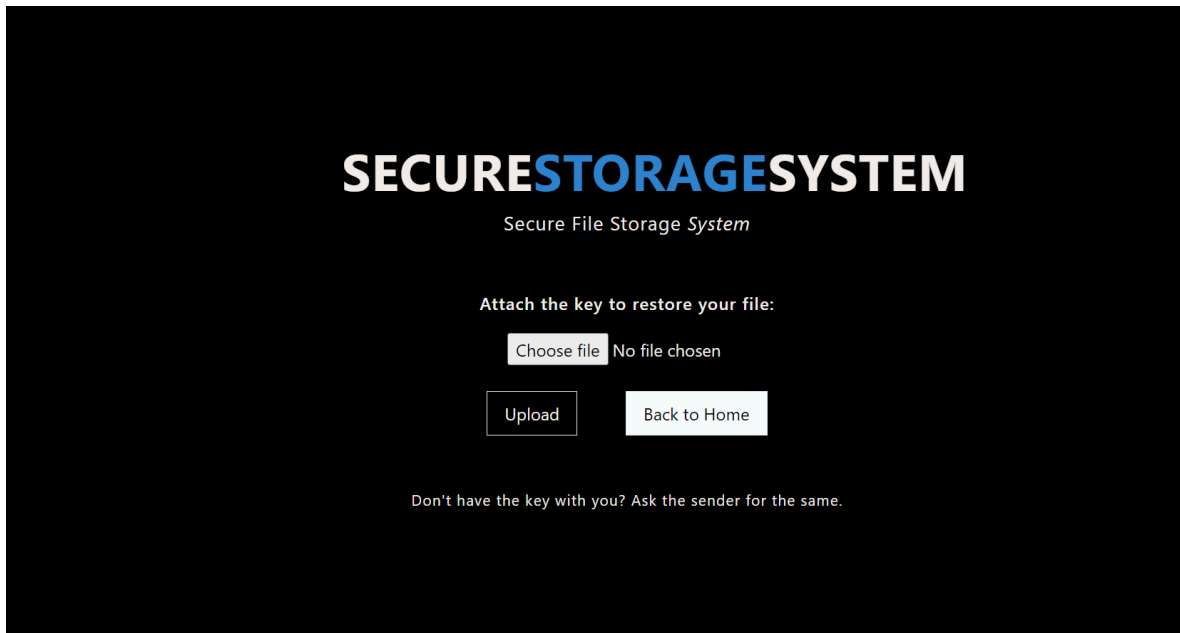


Figure 5.5: Download Page

5.1.6 Restore Page

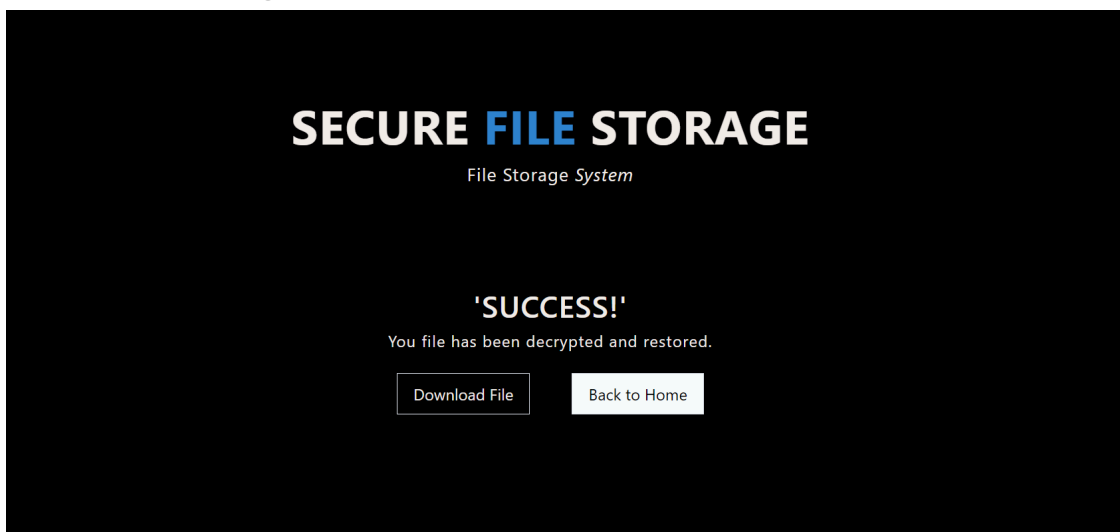


Figure 5.6: Restore Page

5.1.7 Securing a .txt file

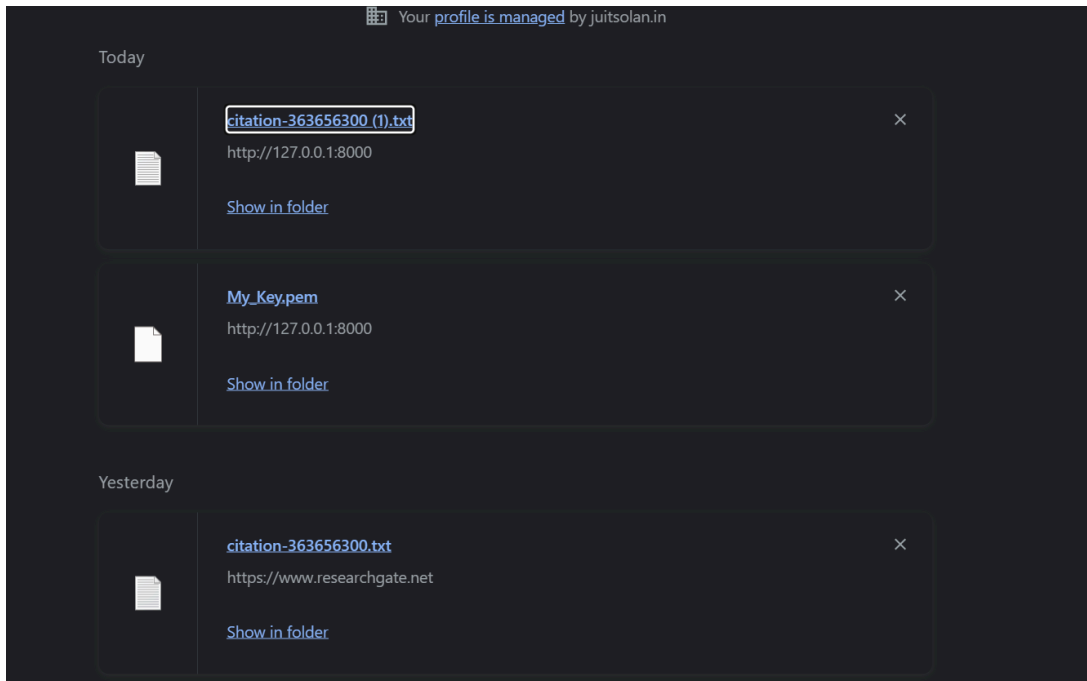


Figure 5.7: Downloaded File

5.2 EVALUATION

The achievement of stated objectives takes centre stage, analysing how carefully the project matches with initial goals and fulfils stated standards. Rigorous evaluation extends to the operation and performance of the system, measuring its responsiveness and adaptability under varied settings. Usability and user experience hold a crucial position, with user feedback impacting the assessment of the system's accessibility and general happiness. Reliability, scalability, security, and compliance undergo thorough evaluation, assuring the system's robustness, flexibility to change, and conformity to regulatory standards. Documentation quality, support systems, and resource management all fall under the evaluation lens, leading to a thorough picture of the project's strengths and opportunities for growth. Lessons gained can be learned from the project's lifecycle, offering useful insights for future undertakings, while forward-looking recommendations provide refinements and options for further study and development. This evaluative panorama not only serves as a retrospective study but also as a drive for constant improvement and innovation in future projects.

Chapter 6: CONCLUSION AND FUTURE SCOPE

6.1 CONCLUSION

The importance of key exchange methods cannot be underestimated in today's secure cloud simulations based on modern cryptographic techniques. The use of robust encryption algorithms such as AES, ChaCha20Poly1305, AES-GCM, AES-CCM, RSA and considering secure key exchange methods including the Diffie-Hellman algorithm in the cloud simulator is a major step. Secure communication channels hinge on one important element, the exchange of keys. Implementing secure key exchange techniques provides additional protection beyond just data confidentiality, allowing subsequent authenticity, privacy, and resilience to attacks. The use of protocols that ensure forward secrecy, resistance to threats from quantum computers, and scale-ability over different cloud environments is indicative of the project's proactivity in anticipating future security challenges. A key element of trust and dependability is ensuring the safe operation of key exchange methodologies within the simulated cloud environment. It becomes a basis for developing secure interfaces, ensuring information privacy, and facilitating communication among entities in a cloud model. During the close of the current phase of implementation, there is a need to ensure that critical exchange keys are secure. These methodologies are reinforced continuously, keeping pace with evolving threats. Secure computing remains in vogue for cloud-based applications for different domains, assuring trust and credibility.

6.2 FUTURE SCOPE

The project lays the groundwork for several potential avenues for future enhancements and expansions:

- **Enhanced Cryptographic Algorithms:** The continuous exploration and integration of new sophisticated cryptographic algorithms in response to security threats that are always changing and new standards.
- **Adding 2 factor authentication:** 2FA adds an additional layer of security beyond just a password. This means even if a password is compromised, unauthorized access is still prevented unless the second factor is also compromised.
- **Machine Learning and Cryptography Fusion:** Looking into the coupling of machine learning approaches and cryptography for anomaly detection, pattern recognition, and pre-emptive risk management through loud experiments.
- **Quantum-Safe Cryptography:** Provision of research and quantum-resilient cryptographic techniques in anticipation of quantum computers and the possible dangers they present to conventional cryptography.
- **Automation and Orchestration:** Automation of the critical key management, and the processes of encryption/decryption as well as coordination of the activities for the encryption and decryption within the cloud system, and thus ensuring smooth running of the operations in the cloud environment.
- **Compliance and Governance Frameworks:** Incorporating strong governance and compliance guidelines that conform to industry-specific laws pertaining to data security and protection in cloud-based simulations.
- **Real-time Security Monitoring:** To achieve this, real-time monitoring and response systems must be put in place to detect and curb security incidents or anomalies in the emulated cloud environment.
- **Cross-Platform Compatibility:** Making cryptographic tools to be compatible with different cloud platforms and services so that it can achieve a wider practical use.

REFERENCES

- [1] Gudimetla, Sandeep. "MULTI-FACTOR AUTHENTICATION FOR CLOUD." International Research Journal of Modernization in Engineering Technology and Science 3 (2024): 4341-4343.
- [2] Goswami, Paromita & Faujdar, Neetu & Debnath, Somen & Khan, Ajoy & Singh, Ghanshyam. (2024). Investigation on storage level data integrity strategies in cloud computing: classification, security obstructions, challenges and vulnerability. Journal of Cloud Computing. 13. 10.1186/s13677-024-00605-z.
- [3] S. B. Mallisetty, G. A. Tripuramallu, K. Kamada, P. Devineni, S. Kavitha and A. V. P. Krishna, "A Review on Cloud Security and Its Challenges," 2023 International Conference on Intelligent Data Communication Technologies and Internet of Things (IDCIoT), Bengaluru, India, 2023, pp. 798-804, doi: 10.1109/IDCIoT56793.2023.10053520
- [4] Kiran Kurian, Lekshmi S Nair, Dr. Joby P P, Rinu Maria Jose, Rosa Mariam John, 2023, Secure File Storage in Cloud Using Hybrid Encryption, INTERNATIONAL JOURNAL OF ENGINEERING RESEARCH & TECHNOLOGY (IJERT) Volume 11, Issue 04,
- [5] C. Susmitha, S. Srineeharika, K. S. Laasya, S. K. Kannaiah and S. Bulla, "Hybrid Cryptography for Secure File Storage," 2023 7th International Conference on Computing Methodologies and Communication (ICCMC), Erode, India, 2023, pp. 1151-1156, doi: 10.1109/ICCMC56507.2023.10084073.
- [6] Garad, Apurva, et al. "SECURING FILE STORAGE IN CLOUD USING HYBRID CRYPTOGRAPHY." International Journal of Advances in Engineering Research (2022).
- [7] Fatima, S.; Rehman, T.; Fatima, M.; Khan, S.; Ali, M.A. Comparative Analysis of Aes and Rsa Algorithms for Data Security in Cloud Computing. Eng. Proc. (2022) <https://doi.org/10.3390/engproc2022020014>
- [8] S. Kumar, G. Karnani, M. S. Gaur and A. Mishra, "Cloud Security using Hybrid Cryptography Algorithms," 2021 2nd International Conference on Intelligent Engineering and Management (ICIEM), London, United Kingdom, 2021, pp. 599-604, doi: 10.1109/ICIEM51511.2021.9445377.
- [9] P. Boisrond, "A Position Paper on Amazon Web Services (AWS) Simple Storage

Service (S3) Buckets," May 2021. [Online]. Available: doi.org/10.13140/RG.2.2.17727.84640

[10] S. Kaushik and A. Patel, "Secure Cloud Data Using Hybrid Cryptographic Scheme," 2019 4th International Conference on Internet of Things: Smart Innovation and Usages (IoT-SIU), Ghaziabad, India, 2019, pp. 1-6, doi: 10.1109/IoT-SIU.2019.8777592.

[11] P. Jain, P. Muskara and P. Jain, "Enhance Data Security in Cloud Computing with Digital Signature & Hybrid Cryptographic Algorithm," 2021 International Conference on Simulation, Automation & Smart Manufacturing (SASM), Mathura, India, 2021, pp. 1-6, doi: 10.1109/SASM51857.2021.9841171.

[12] Tahir, M., Sardaraz, M., Mehmood, Z. et al. CryptoGA: a cryptosystem based on genetic algorithm for cloud data security. Cluster Comput 24, 739–752 (2021). <https://doi.org/10.1007/s10586-020-03157-4>

[13] Hosny, Khalid & Taha, Ali & Salama, Dr-Diaa. (2018). AN IMPROVED SECURITY SCHEMA FOR MOBILE CLOUD COMPUTING USING HYBRID CRYPTOGRAPHIC ALGORITHMS. Far East Journal of Electronics and Communications. 18. 10.17654/EC018040521.

[14] M. Attaran and J. Woods, "Cloud computing technology: improving small business performance using the Internet," J. Small Bus. Entrepren., vol. 13, pp. 94-106, 2018, doi: 10.1080/08276331.2018.1466850.

PlagCheck (1).pdf

ORIGINALITY REPORT

16%

SIMILARITY INDEX

11%

INTERNET SOURCES

9%

PUBLICATIONS

7%

STUDENT PAPERS

PRIMARY SOURCES

| | | |
|---|--|----|
| 1 | journalofcloudcomputing.springeropen.com Internet Source | 2% |
| 2 | www.researchgate.net Internet Source | 1% |
| 3 | www.ijert.org Internet Source | 1% |
| 4 | G.P.C. Venkata Krishna, D. Vivekananda Reddy. "Machine learning-enhanced hybrid cryptography and image steganography algorithm for securing cloud data", Journal of Intelligent & Fuzzy Systems, 2023 Publication | 1% |
| 5 | ijeir.org Internet Source | 1% |
| 6 | fastercapital.com Internet Source | 1% |
| 7 | www.jatit.org Internet Source | 1% |

JAYPEE UNIVERSITY OF INFORMATION TECHNOLOGY, WAKNAGHAT

PLAGIARISM VERIFICATION REPORT

Date:

Type of Document (Tick): PhD Thesis M.Tech Dissertation/ Report B.Tech Project Report Paper

Name: _____ Department: _____ Enrolment No _____

Contact No. _____ E-mail. _____

Name of the Supervisor: _____

Title of the Thesis/Dissertation/Project Report/Paper (In Capital letters): _____

UNDERTAKING

I undertake that I am aware of the plagiarism related norms/ regulations, if I found guilty of any plagiarism and copyright violations in the above thesis/report even after award of degree, the University reserves the rights to withdraw/ revoke my degree/report. Kindly allow me to avail Plagiarism verification report for the document mentioned above.

Complete Thesis/Report Pages Detail:

- Total No. of Pages =
- Total No. of Preliminary pages =
- Total No. of pages accommodate bibliography/references =

(Signature of Student)

FOR DEPARTMENT USE

We have checked the thesis/report as per norms and found **Similarity Index** at(%). Therefore, we are forwarding the complete thesis/report for final plagiarism check. The plagiarism verification report may be handed over to the candidate.

(Signature of Guide/Supervisor)

Signature of HOD

FOR LRC USE

The above document was scanned for plagiarism check. The outcome of the same is reported below:

| Copy Received on | Excluded | Similarity Index (%) | Generated Plagiarism Report Details (Title, Abstract & Chapters) | |
|---------------------|--|----------------------|--|--|
| | <ul style="list-style-type: none">• All Preliminary Pages• Bibliography/Images/Quotes• 14 Words String | | Word Counts | |
| Report Generated on | | Submission ID | Character Counts | |
| | | | Total Pages Scanned | |
| | | | File Size | |

Checked by

Name & Signature

.....

Librarian

Please send your complete thesis/report in (PDF) with Title Page, Abstract and Chapters in (Word File) through the supervisor at plagcheck.juit@gmail.com