IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 2, July 2025
Impact Factor: 7.67

Implementation of Securely Sharing and Data Deduplication on End-To-End Encrypted Documents

Palagati Anusha[1], Kalakonda Giriprasad[2], Kondeti Karthik[3], VangalaThirumal Reddy[4]
[1]Assistant Professor, Department of Computer Science and Engineering
[2,3,4]U.G Scholars, Department of Computer Science and Engineering
Guru Nanak Institute of Technology, Hyderabad, Telangana

Abstract: *This project focuses on securing data storage and transmission using end-to-end encryption while implementing a data deduplication mechanism. The core objective is to ensure that identical files are not duplicated within the system, thus optimizing storage space and minimizing unnecessary redundancy. In the event that a user uploads a file with the same name as an existing file, the system will flag it as a duplicate and prevent its storage. The project uses ABE technique encryption, one of the most secure encryption methods available, to protect user data. This encryption ensures that the file content remains inaccessible to unauthorized users, even if intercepted during transmission. Users will be able to search for their files using encrypted identifiers, preventing exposure of sensitive information during the search process. A proxy server facilitates secure communication between the user and the storage system. The server serves as an intermediary, forwarding data requests securely and ensuring the integrity of the information being exchanged. However, malicious users pose a threat to this system, as they may attempt to exploit vulnerabilities to launch attacks, such as data tampering or unauthorized access. The system has been designed with robust security measures to mitigate these risks and protect the integrity of the stored data. By combining data deduplication with advanced encryption techniques and secure data transmission protocols, this project aims to provide a highly secure, efficient, and scalable solution for managing sensitive files while protecting against potential malicious attacks.*

Keywords: Attribute-Based Encryption (ABE), Searchable Encryption (SE), Keyword Search, Cloud Storage Security, Data Deduplication, Proof of Ownership (PoW), Outsourced Decryption

I. INTRODUCTION

In recent years, the explosive growth of data generated through digital technologies and internet applications has led individuals and organizations to increasingly rely on cloud storage services to manage their data. While cloud computing offers scalable storage and convenient data sharing, it also raises significant concerns regarding data confidentiality, secure access, and efficient resource utilization. To safeguard sensitive information, data owners often encrypt files before uploading them to cloud servers. However, this encryption limits the ability to perform efficient operations such as keyword search and deduplication on the encrypted data.

To address this challenge, Searchable Encryption (SE) has emerged as a promising solution, allowing encrypted data to be searched without compromising confidentiality. Additionally, Attribute-Based Encryption (ABE) enables fine-grained access control, allowing only authorized users to decrypt data based on defined attributes. However, traditional ABE schemes often reveal access policies, which may leak sensitive information about the data or users. Further, the computational burden of decryption in ABE schemes can be substantial, especially in systems with complex attribute structures.

Recent advancements have introduced Attribute-Based Keyword Search (ABKS) schemes that combine secure search functionality with controlled access mechanisms. Yet, these systems still face critical limitations related to high

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 5, Issue 2, July 2025 Impact Factor: 7.67

computational costs, lack of access policy privacy, and the inability to verify the integrity of search results from potentially untrusted cloud service providers (CSPs).

This study aims to develop a secure and efficient ABKS scheme that supports:

- Data integrity verification to ensure that CSPs return correct and complete search results,
- Data deduplication to eliminate redundant data storage, and
- Outsourced decryption to minimize computation for end users without exposing sensitive content to CSPs.

II. REVIEW OF LITERATURE

Numerous research studies have been conducted to address the challenges of secure data storage, deduplication, keyword search, and integrity verification in cloud computing environments. Below is a review of significant contributions in this domain:

H. Yuan, X. Chen, J. Wang, J. Yuan, H. Yan, and W. Susilo (2024) This study proposed a blockchain-integrated system for public data auditing and secure deduplication. By leveraging smart contracts, the system ensures fair arbitration and automatic penalization of malicious Cloud Service Providers (CSPs), while compensating affected users. It introduces message-locked encryption and eliminates random masking, resulting in reduced computational and storage costs.

X. Yang, R. Lu, J. Shao, X. Tang, and A. Ghorbani (2023) The authors developed a deduplication system that supports user-defined access control. The system allows CSPs to manage authorization without compromising data confidentiality, tag consistency, or resistance to brute-force attacks. It effectively balances deduplication efficiency with stringent security requirements.

Y. Li, G. Xu, H. Xian, L. Rao, and J. Shi (2023) Though not directly related to cloud storage, this work presents a malware detection approach using hybrid static and dynamic features for Android apps. It enhances model accuracy by combining permission requests, API calls, and runtime behaviors, which informs broader data security practices in cloud-based systems.

L. Wang, B. Wang, W. Song, and Z. Zhang (2023) This study addresses the security flaws in Convergent Encryption (CE) by proposing a key-sharing mechanism. Only the original data uploader generates and distributes the Convergent Key (CK), significantly reducing redundant key storage and enhancing resistance to brute-force and collusion attacks.

R. B. Sirsat and N. R. Talhar (2023) This research introduces a dual-cloud architecture using both public and private cloud servers. The system enhances deduplication through hash comparison and proof of ownership, ensuring that only unique data is stored while maintaining data confidentiality and system integrity.

H. Hou, J. Yu, and R. Hao (2023) This study proposes a dynamic security-level system where data is encrypted based on its popularity—private data uses semantic encryption, while popular data uses CE. It introduces a mechanism to update authenticators without user intervention when data transitions between popularity states, thus maintaining audit integrity.

D. X. Song, D. Wagner, and A. Perrig (2000) This foundational work introduced Symmetric Searchable Encryption (SSE), where each word in a file is individually encrypted, enabling keyword searches without revealing the actual data. However, the scheme suffers from practical limitations, particularly in managing and distributing symmetric keys securely, making it less scalable for large systems.

D. Boneh, G. Di Crescenzo, R. Ostrovsky, and G. Persiano (2004) The authors proposed PEKS, which allows users to perform keyword searches on encrypted data using public-key cryptography. While it supports single keyword queries and offers improved key management compared to SSE, it lacks support for more expressive queries and efficient performance in large datasets.

J. Li, Q. Wang, C. Wang, N. Cao, and K. Ren (2010) This study addresses the issue of user input errors (e.g., typos) in searchable encryption by proposing a fuzzy keyword search method. The scheme enhances usability without sacrificing data confidentiality but introduces additional computational overhead for fuzzy set generation and matching.

A. Sahai and B. Waters (2005) This paper introduced the concept of ABE, enabling encryption based on user attributes and allowing fine-grained access control. Data is encrypted under a policy, and only users with matching attributes can decrypt. While powerful, the original scheme does not protect access policies, exposing sensitive metadata.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 2, July 2025
Impact Factor: 7.67

J. Li, Z. Liu, and D. Li (2015) The authors proposed an enhanced Ciphertext-Policy ABE (CP-ABE) scheme that supports policy hiding and user revocation. This improves privacy by concealing access rules and maintaining security even when user permissions change, but it involves complex encryption and decryption processes.

Q. Wu, B. Qin, and X. Lin (2016) The authors designed an ABKS scheme that allows secure keyword search while hiding access policies. Although it improves privacy, the scheme supports only single-keyword search and still suffers from high computational overhead due to bilinear pairings.

B. Green, M. Chase, and S. Kamara (2011) This study introduces outsourced decryption, where CSPs help with decryption to reduce user-side workload, without learning anything about the plaintext. This approach significantly reduces computational burden but assumes a semi-trusted CSP.

C. Wang, Q. Wang, K. Ren, and W. Lou (2010) This work focuses on enabling public auditing of cloud-stored encrypted data without compromising data confidentiality. It introduces a third-party auditor to verify data integrity. Although it doesn't handle deduplication directly, it lays a foundation for combining auditing with privacy-preserving features.

S. Halevi, D. Harnik, B. Pinkas, and A. Shulman-Peleg (2011) The study presents a method to ensure proof of ownership in deduplication systems, preventing unauthorized users from claiming existing data. It balances storage efficiency and security but can be vulnerable to dictionary attacks when data is predictable.

Scope of the study

This study focuses on developing a secure and efficient framework for Attribute-Based Keyword Search (ABKS) over encrypted cloud data. The proposed system aims to support data deduplication, integrity verification, and outsourced decryption, ensuring that cloud storage is both secure and resource-efficient. The work addresses the challenges faced in traditional deduplication approaches, especially the risk of data leakage when a global encryption key is used by all users, which compromises confidentiality in a cloud environment where the Cloud Service Provider (CSP) is considered *honest-but-curious*.

By implementing a privacy-preserving ABKS scheme, this research ensures that encrypted data can be efficiently searched and deduplicated without exposing sensitive content or metadata. The scope includes managing secure access for authorized users based on attributes, minimizing computational overhead on end users, and enhancing trust in search results returned by semi-trusted CSPs.

Problem Statement

With the rapid growth of cloud storage systems and data-sharing platforms, securing sensitive user data while optimizing storage remains a critical concern. Although data deduplication is widely used to reduce storage costs by eliminating redundant copies, traditional deduplication methods often compromise user privacy and data confidentiality. Moreover, many existing solutions lack comprehensive protection against advanced threats such as phishing, malware, and inference attacks.

Another limitation is the absence of robust end-to-end encryption, leaving data vulnerable during transmission or storage. Current systems also fail to offer integrity verification, creating risks of incomplete or manipulated search results returned by untrusted or semi-trusted CSPs.

There is a pressing need for a secure, efficient, and privacy-preserving system that:

- Allows encrypted keyword search with fine-grained access control,
- Enables secure deduplication without data exposure,
- Supports proof of ownership and outsourced decryption, and
- Provides verifiable search result integrity in a cloud environment.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 2, July 2025
Impact Factor: 7.67

Objectives of the Study

The primary objective of this study is to design and implement a secure Attribute-Based Keyword Search (ABKS) framework over encrypted cloud data that ensures data confidentiality, integrity, and efficiency. The specific objectives are:

- To develop a secure searchable encryption mechanism that allows authorized users to perform keyword-based searches over encrypted cloud data without exposing sensitive content.
- To implement Attribute-Based Encryption (ABE) for fine-grained access control, enabling only users with appropriate attribute sets to decrypt and access specific data.
- To incorporate data deduplication techniques that eliminate redundant data storage without compromising security or user privacy.
- To enable integrity verification of search results, ensuring that the Cloud Service Provider (CSP) returns accurate, complete, and untampered data to the user.
- To reduce computational overhead on end users by integrating outsourced decryption, allowing semi-trusted CSPs to assist in decryption without accessing the plaintext.
- To address the issue of access policy privacy, preventing unauthorized disclosure of user identity and data access patterns through hidden policy enforcement.
- To develop a system architecture that balances security, performance, and storage efficiency in a semi-trusted cloud environment.

Existing System

Current cloud storage systems often rely on convergent encryption (CE) for deduplication, where data with the same content produces the same ciphertext.

However, CE introduces security risks, such as brute-force attacks and information leakage, especially when files are predictable or of small size.

Many existing systems do not offer:

- Fine-grained access control
- Efficient decryption
- Search result verification
- Traditional searchable encryption systems often expose access policies or do not support hidden policies, putting user privacy at risk.
- Existing ABE-based schemes suffer from high computational costs and limited scalability in practical deployment.

Proposed System

The proposed system introduces a Secure Attribute-Based Keyword Search (ABKS) mechanism that integrates:

- Secure keyword search over encrypted data
- Fine-grained access control using ABE
- Data deduplication to eliminate redundant storage
- Search result integrity verification
- Outsourced decryption to reduce user-side computational burden
- It ensures confidentiality, integrity, and availability of cloud-stored data even in the presence of a semi-trusted Cloud Service Provider (CSP).
- Uses attribute hiding to prevent leakage of access policies or user attributes.
- Supports multiple stakeholders including Data Owners, Data Users, and Proxies.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 2, July 2025
Impact Factor: 7.67

III. METHODOLOGY

Data Encryption & Upload by Data Owner:

- The data owner encrypts files using Attribute-Based Encryption based on access policies.
- Keywords are extracted and encrypted for searchable index creation.

Secure Deduplication:

- The system checks for file duplication without decrypting the data, maintaining privacy.
- Deduplication is performed based on cryptographic proofs such as Proof of Ownership (PoW).

Search by Data User:

- Authorized users generate trapdoors using their attributes and search for specific keywords.
- Only users with matching attribute sets can decrypt the retrieved data.

Search Result Verification:

- CSP returns encrypted search results and corresponding integrity proofs.
- Data users can verify the correctness and completeness of results using cryptographic verification.

Outsourced Decryption:

- Most of the decryption process is delegated to the CSP, reducing the resource load on the user.
- The CSP performs partial decryption without learning plaintext content.

Techniques Used

- **Attribute-Based Encryption (ABE):** Enables fine-grained access control.
- **Searchable Encryption (SE):** Allows searching over encrypted data using keywords.
- **Proof of Ownership (PoW):** Used for secure deduplication without revealing data.
- **Trapdoor Functions:** Secure mechanism for keyword search by authorized users.
- **Hidden Policy Encryption:** Keeps access policies confidential.
- **Outsourced Decryption Techniques:** Shifts computational load from user to CSP.
- **Integrity Verification Mechanisms:** Ensures CSP returns unmodified and complete search results.

System Architecture Description

The system architecture for the proposed Secure Attribute-Based Keyword Search (ABKS) with Deduplication and Integrity Verification consists of the following key entities:

1. Data Owner (DO)

- Encrypts data files using Attribute-Based Encryption (ABE) based on predefined access policies.
- Extracts keywords and creates encrypted indexes (trapdoors).
- Uploads the encrypted data and keyword indexes to the Cloud Service Provider (CSP).
- Performs Proof of Ownership (PoW) to support deduplication.

2. Cloud Service Provider (CSP)

- Stores encrypted data and keyword indexes.
- Checks for data duplication using secure PoW without accessing file contents.
- Executes search queries using the trapdoor generated by users.
- Performs partial decryption (outsourced decryption) and returns search results with integrity proofs.

3. Data User (DU)

- Generates a trapdoor using authorized attribute credentials.
- Sends search requests (trapdoors) to the CSP.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 2, July 2025
Impact Factor: 7.67

- Receives encrypted results and verifies integrity using the CSP's response.
- Performs final decryption on partially decrypted data using their private key.

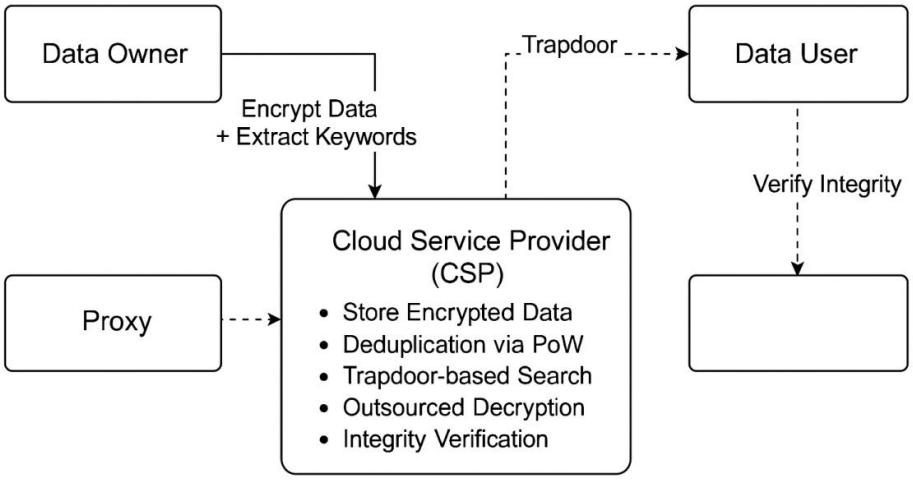
4. Proxy (Optional)

- Assists in managing user attributes and generating trapdoors securely.
- May support partial decryption and delegation when user resources are limited.

Workflow Overview

Data Encryption and Upload:
DO → CSP: Encrypted data + keyword indexes.
Secure Deduplication:
CSP checks for existing ciphertext via secure PoW.
Keyword Search:
DU → CSP: Trapdoor query.
CSP → DU: Matching encrypted files + verification tags.
Outsourced Decryption:
CSP performs computationally heavy part of decryption.
DU completes decryption with minimal processing.
Integrity Verification:
DU checks that search results are correct and complete using integrity tags.

System Architecture



IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 2, July 2025
Impact Factor: 7.67

IV. RESULTS AND DISCUSIONS

4.1 SNAPSHOTS

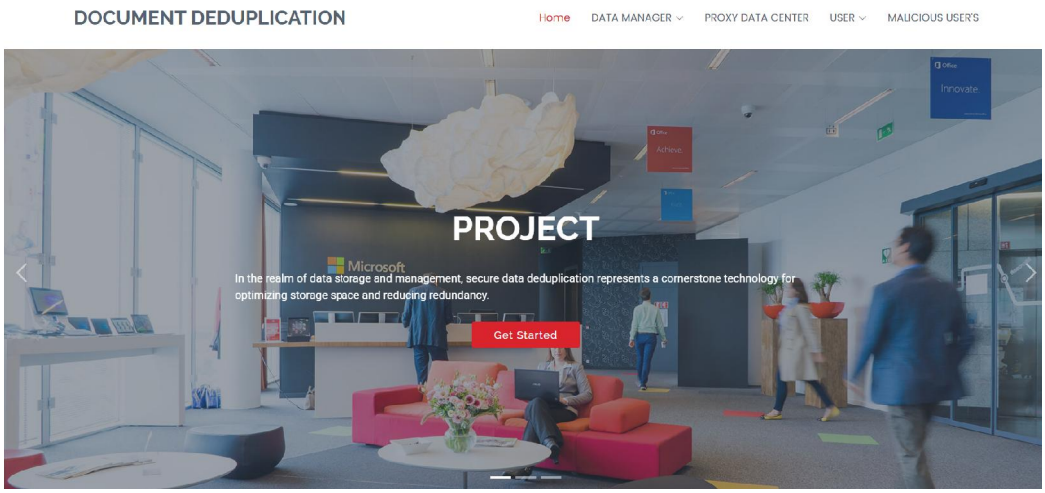


Fig 1 Login page

When we run a code the output will be as shown above figure, this code is being executed using Data Server.

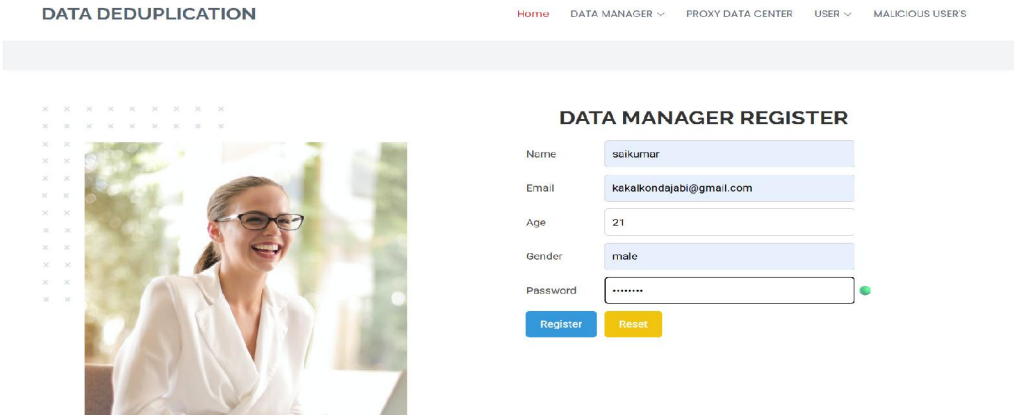


Fig 2 Registration of manager

The data manager will be registered using his credentials. The password must be strong in such a way that no one can hack it as this is the main step in the entier project.

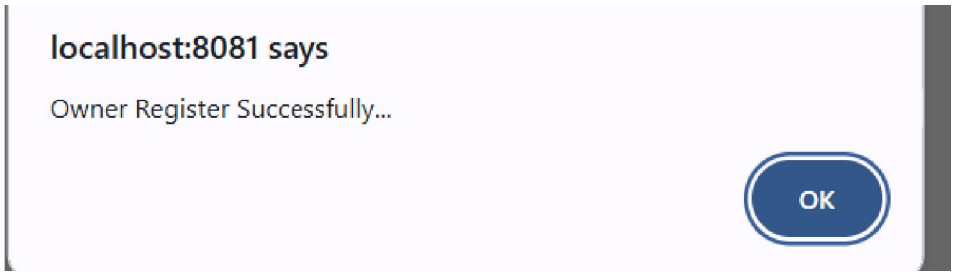


Fig 3 Manager Register successfully

This image shows us that the owner is register successfully and this registeration is possible if only the password provided is strong and correct.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 2, July 2025
Impact Factor: 7.67

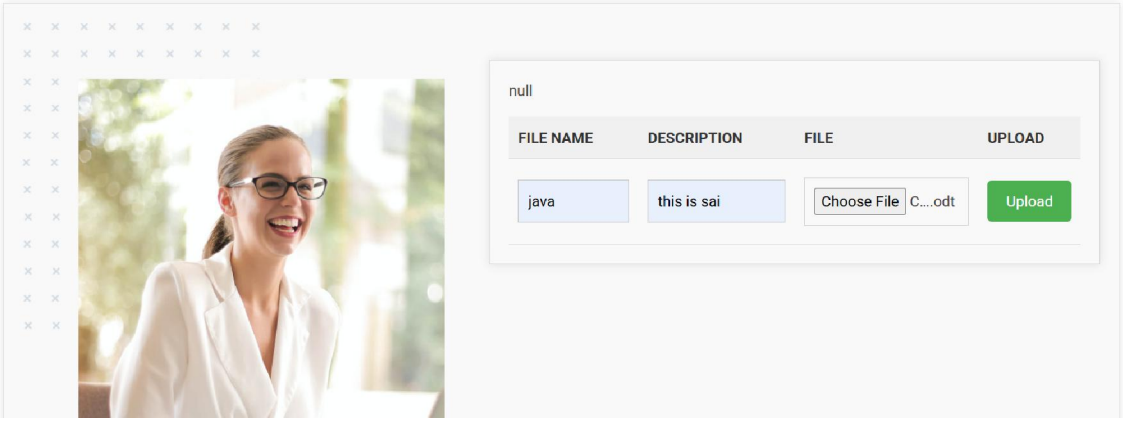


Fig 4 Upload the Data

The above image shows us about uploding of data with the required fields such as file name, description and file that should be uploded

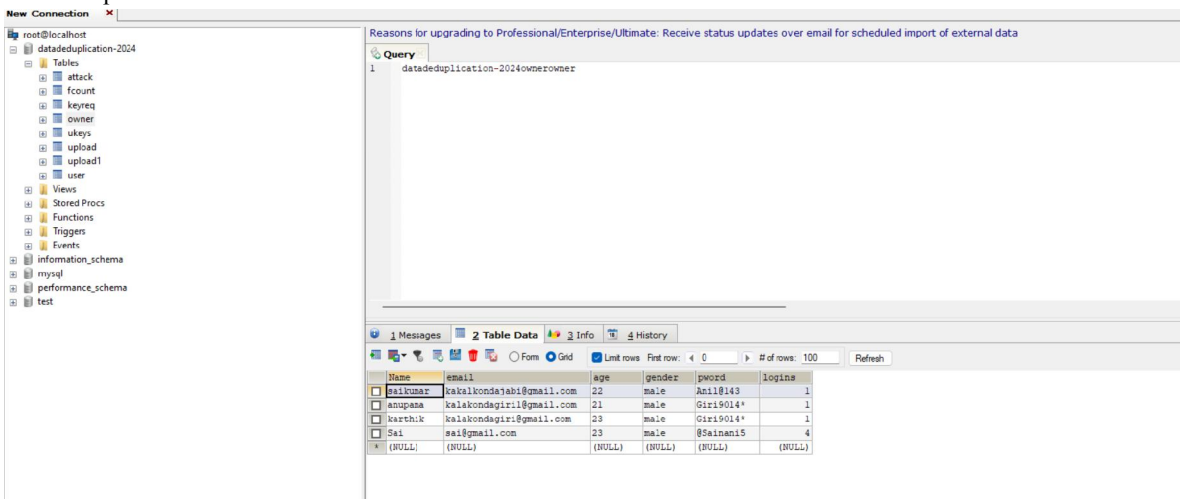


Fig 5 Owner login checking

The above image shows us that the owner is registered and uploded data successfully which is shown is the tabular formate specifying each field in different column.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 2, July 2025
Impact Factor: 7.67

Fig 6 Data Rigister

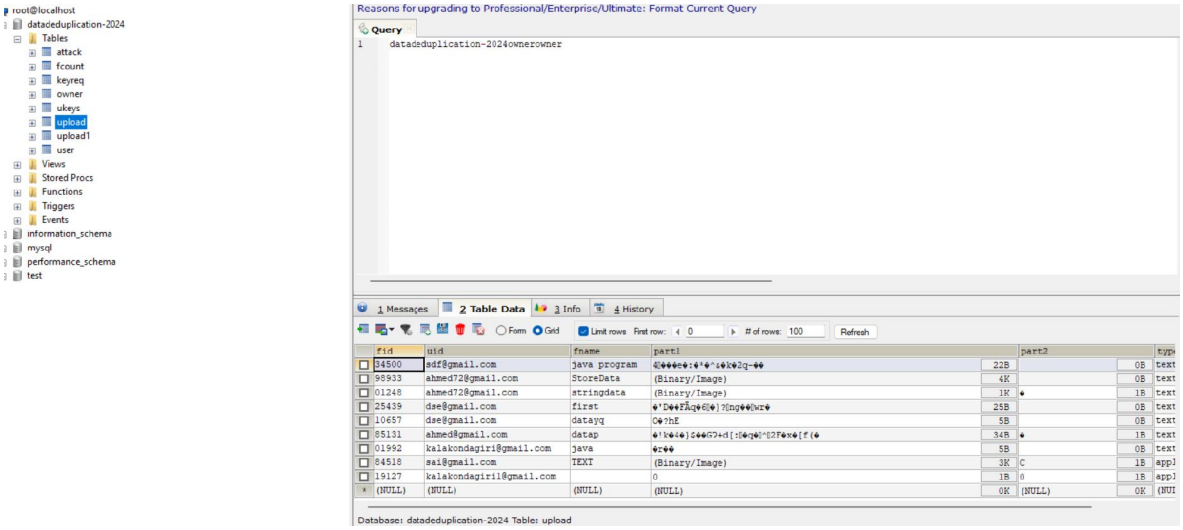


Fig 7 Check Uploaded Data

The above image shows the data uploding successfully in the table formate specifying each field in separate column which shows us the data uploaded.

V. CONCLUSION

In this projects, we propose a deduplication scheme for encrypted data, named Data Dedupulication. It does not relay on any online trusted third party, and it allows dynamic ownership updating. We use cryptographic primitives such as proxy server convergent encryption. Compared with previous schemes, the security of Data Dedupulication is significantly enhanced. In our design, the proxy server can work as an intermediary, which focuses on three core functionalities, in particular, 1) updating the ownership list when a user updates his data, 2) assisting an initial uploader to validate subsequent uploaders, and 3) delivering data encryption with keys. The optimized Proxy server in Data Dedupulication can easily and effectively perform deduplication on encrypted data. Meanwhile, it is enforced that a user is allowed to access the data only if he can provide a valid key. Simulation experiments show that our scheme is applicable and efficient..

VI. FUTURE ENHANCEMENT

The future enhancement data transfer to the cloud is our next improvement. All of the data in the cloud is encrypted. The information provides a private key for data sharing. Only information that has been detected by attackers is stored on the server. Because users need efficiency and CSP is constantly looking for ways to save storage costs, there is some friction between their goals. To further improve system intelligence, the cloud server will be designed to automatically

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 2, July 2025
Impact Factor: 7.67

detect and isolate data that has been targeted or compromised by attackers. This compromised data will be securely stored and monitored for forensic analysis, enhancing the system's incident response capabilities. A key challenge moving forward lies in balancing the user's need for efficiency and privacy with the Cloud Service Provider's (CSP's) objective of reducing storage and operational costs. To address this tension, future versions of the system will focus on implementing adaptive storage strategies, intelligent deduplication, and advanced access control policies.

REFERENCES

[1]. Boneh, D., Di Crescenzo, G., Ostrovsky, R., &Persiano, G. (2004). Public key encryption with keyword search. In Advances in Cryptology – EUROCRYPT 2004 (pp. 506–522). Springer.

[2]. Green, M., Chase, M., &Kamara, S. (2011). Outsourcing the decryption of ABE ciphertexts. In USENIX Security Symposium (pp. 34–34).

[3]. Halevi, S., Harnik, D., Pinkas, B., & Shulman-Peleg, A. (2011). Proofs of ownership in remote storage systems. In Proceedings of the 18th ACM conference on Computer and communications security (pp. 491–500). ACM.

[4]. Hou, H., Yu, J., &Hao, R. (2023). Cloud storage auditing with deduplication supporting different security levels according to data popularity. [Journal Name], [Volume(Issue)], pages.

[5]. Li, J., Liu, Z., & Li, D. (2015). Ciphertext-policy attribute-based encryption with hidden policy and efficient revocation. IEEE Transactions on Dependable and Secure Computing, 12(5), 477–489.

[6]. Li, J., Wang, Q., Wang, C., Cao, N., &Ren, K. (2010). Fuzzy keyword search over encrypted data in cloud computing. In Proceedings of IEEE INFOCOM 2010 (pp. 1–5). IEEE.

[7]. Li, Y., Xu, G., Xian, H., Rao, L., & Shi, J. (2023). Novel Android malware detection method based on multi-dimensional hybrid features extraction and analysis. [Journal Name], [Volume(Issue)], pages.

[8]. Sahai, A., & Waters, B. (2005). Fuzzy identity-based encryption. In Advances in Cryptology – EUROCRYPT 2005 (pp. 457–473). Springer.

[9]. Sirsat, R. B., &Talhar, N. R. (2023). Deduplication in cloud storage on the basis of proof of ownership. [Journal Name], [Volume(Issue)], pages.

[10]. Song, D. X., Wagner, D., &Perrig, A. (2000). Practical techniques for searches on encrypted data. In Proceedings of the 2000 IEEE Symposium on Security and Privacy (pp. 44–55). IEEE.

[11]. Wang, C., Wang, Q., Ren, K., & Lou, W. (2010). Privacy-preserving public auditing for data storage security in cloud computing. In IEEE INFOCOM 2010 (pp. 1–9). IEEE.

[12]. Wang, L., Wang, B., Song, W., & Zhang, Z. (2023). A key-sharing based secured duplication scheme in cloud storage. [Journal Name], [Volume(Issue)], pages.

[13]. Wu, Q., Qin, B., & Lin, X. (2016). Attribute-based keyword search with hidden access policies. IEEE Transactions on Information Forensics and Security, 11(4), 733–745.

[14]. Yang, X., Lu, R., Shao, J., Tang, X., &Ghorbani, A. (2023). Achieving efficient secure deduplication with user-defined access control in cloud. [Journal Name], [Volume(Issue)], pages.

[15]. Yuan, H., Chen, X., Wang, J., Yuan, J., Yan, H., &Susilo, W. (2024). Blockchain-based public auditing and secure deduplication with fair arbitration. [Journal Name], [Volume(Issue)], pages.