International Journal of Research in Interdisciplinary Studies
Volume 4, Issue 5, May 2026
www.ijris.com | E-ISSN: 2584-1017

Secure Cloud Storage with Deduct-Based Deduplication Using Proof of Ownership and Proof of Storage

K. Elakia[1], R. G. Suresh Kumar[2*], R. Vishnu[3], S. Mohammed Azmi[3], R. Sai Harish[3], S. Srinivasan[3]

[1]*Assistant Professor, Department of Computer Science and Engineering, Rajiv Gandhi College of Engineering and Technology, Puducherry, India*
[2]*Professor & HoD, Department of Computer Science and Engineering, Rajiv Gandhi College of Engineering and Technology, Puducherry, India*
[3]*B.Tech. Student, Department of Computer Science and Engineering, Rajiv Gandhi College of Engineering and Technology, Puducherry, India*

Abstract— Cloud computing has revolutionized data storage by offering scalable, cost-efficient, and on-demand resource management. A key optimization technique in cloud environments is data deduplication, which eliminates redundant copies of identical files and stores only a single instance to improve storage efficiency and reduce operational costs. Despite its advantages, deduplication introduces security concerns related to file ownership, privacy preservation, and data integrity, particularly when multiple users attempt to claim access to the same content. To address these issues, existing systems implement Proof of Ownership (PoW), a cryptographic protocol that verifies whether a user legitimately possesses a file before granting access. In PoW-based schemes, users generate cryptographic proofs derived from file contents instead of re-uploading the entire file, thereby conserving bandwidth and computational resources. However, PoW provides only static verification at a single point in time, making it vulnerable to hash-based attacks if adversaries obtain file hashes. To overcome this limitation, the proposed framework integrates Proof of Storage (PoS) with PoW to enable continuous and dynamic ownership verification. PoS periodically challenges users to prove ongoing possession of stored data, thereby enhancing security. This hybrid PoW–PoS approach strengthens data protection, ensures persistent ownership validation, and maintains storage efficiency in deduplicated cloud environments.

***Index Terms*— Cloud Computing, Data Deduplication, Proof of Ownership (PoW), Proof of Storage (PoS), Secure Cloud Storage.**

1. Introduction

Deduplication is an advanced data compression and storage optimization technique widely adopted in cloud computing environments to eliminate redundant copies of identical data [8], [9]. Instead of storing multiple instances of the same file or data block, deduplication retains a single copy and creates logical reference pointers for subsequent occurrences [8]. This approach significantly enhances storage efficiency, reduces bandwidth consumption during uploads, and improves the performance of backup and archival systems [9], [10]. Deduplication operates at two primary levels: file-level and block-level.

File-level deduplication removes duplicate files by comparing cryptographic hash values, whereas block-level deduplication divides files into smaller chunks and eliminates redundancy at a finer granularity, achieving higher storage savings [8], [10]. Cryptographic hash functions such as SHA-256 are commonly used to generate unique identifiers for data comparison [5]. In large-scale cloud infrastructures, where repetitive data such as documents, system images, and backups are frequently generated, deduplication enables service providers to maintain scalability while controlling infrastructure costs [9]. Studies such as DEDUCT [1] and Secure and Efficient Data Deduplication in Joint Cloud Storage (SED) [6] demonstrate that secure deduplication mechanisms can significantly improve storage utilization without compromising system performance.

However, conventional deduplication systems face serious security challenges. Since identical data produces identical hash values, adversaries may exploit hash-based or side-channel attacks to falsely claim file ownership. Research on ownership verification [2] and encrypted deduplication techniques [5] highlights the need for stronger cryptographic safeguards. To mitigate these vulnerabilities, modern systems integrate Proof of Ownership (PoW) and Proof of Storage (PoS) mechanisms. PoW ensures that a user can generate valid cryptographic evidence derived from actual file content before claiming access [25], while PoS periodically verifies continued data possession [11]. These combined mechanisms enhance confidentiality, integrity, and trust in deduplicated cloud storage environments [3], [7], [4].

2. Related Work

[1] Kiana Ghassabi and Peyman pahlevani [1] DEDUCT is a secure and efficient data deduplication framework specifically designed for textual data in Vision-and-Language Navigation (VLN) tasks, where large volumes of human-generated navigation instructions create significant storage demands. It employs a hybrid architecture combining client-side and cloud-side deduplication to achieve effective compression while preserving data confidentiality. The method incorporates lightweight preprocessing, making it suitable for deployment

*Corresponding author: aargeek@gmail.com

on resource-constrained devices such as IoT systems. Additionally, its security-oriented design mitigates risks associated with side-channel attacks, ensuring improved protection of sensitive textual data. DEDUCT can reduce storage requirements by up to 66% and lower operational costs. However, it introduces architectural complexity, may increase processing latency in real-time scenarios, relies on dataset redundancy for optimal performance, and remains susceptible to sophisticated attacks [1].

[2] Jay Dave1, Kamalesh Ram R., Pratik Patil, Himanshu Patil [2] Cloud storage services widely adopt data deduplication to reduce storage costs and bandwidth consumption by eliminating redundant file copies [2]. However, deduplication introduces security risks, as attackers who obtain a file's deduplication tag may falsely claim ownership without possessing the actual file. To address this vulnerability, Proof of Ownership (PoW) mechanisms require users to prove genuine possession before access is granted [25]. The proposed PoW scheme enhances security by ensuring that users must hold the complete file to generate valid proof, effectively preventing tag-based attacks [2]. It is also optimized to reduce I/O, computational, and communication overhead, making it practical for real-world deployment [2]. Nevertheless, it still introduces minor overhead and depends on proper protocol implementation and reliable cloud infrastructure for full security assurance [2].

[3] Mira Lee and Minhye Seo [3] Cloud storage services enable universal data access while overcoming local storage limitations [3]. To improve efficiency, data deduplication is used to eliminate redundant files, saving up to 90% of storage space and bandwidth [3]. The SeDaSC protocol supports secure data sharing by delegating encryption tasks to a cryptographic server (CS), thereby reducing client-side computational burden. However, it lacks deduplication support and relies heavily on full trust in the CS [3]. The proposed protocol extends SeDaSC by integrating secure deduplication with dynamic ownership management, ensuring forward and backward secrecy. It reduces client computation and enhances storage efficiency and privacy. Nevertheless, it introduces dependence on the cryptographic server as a potential single point of failure and may increase server workload under heavy demand [3].

[4] Anjuli Goel1, Chander Prabha , Meenakshi Malik , Preeti Sharma [4] The rapid growth of cloud data has increased the need for efficient and secure storage solutions [4]. Data deduplication reduces storage space and network traffic by eliminating redundant files, but integrating security with deduplication is challenging because traditional encryption conflicts with duplicate detection [4]. To address this, secure deduplication techniques combine cryptographic methods such as convergent encryption [23], Proof of Ownership (PoW) [25], Proof of Retrievability (PoR) [11], DupLESS [24], and attribute-based encryption to ensure confidentiality, integrity, and controlled access [4]. These approaches significantly lower storage and bandwidth costs while maintaining security [4]. However, they often introduce computational and communication overhead, rely on trust in cloud or cryptographic servers, and involve complex implementation, making integration into existing cloud infrastructures challenging [4].

[5] Dr. Harsh Lohia, Suhas A. Lakade, Mr. Yuvraj R. Gurav [5] Data deduplication reduces storage and bandwidth usage in cloud systems by eliminating redundant data copies [5]. To maintain security, convergent encryption is used to encrypt data before outsourcing, enabling duplicate detection without exposing plaintext content [23]. However, traditional deduplication does not consider user privileges, creating security risks in multi-user environments [5]. The proposed scheme introduces authorized deduplication, where duplicate checks are linked to user access rights, preventing unauthorized access or information leakage [5]. It operates within a hybrid cloud architecture, balancing efficiency and security [5][23]. Security analysis and prototype testing show minimal performance overhead [5]. Nevertheless, the approach increases system complexity due to privilege management and may introduce administrative and integration challenges in large-scale or dynamic cloud environments [5].

[6] Kalvakolanu Durga Santhosh [6] Data deduplication is essential in cloud storage to eliminate redundant data and reduce bandwidth usage [6]. Traditional approaches often depend on a trusted key server (KS) for encryption and duplicate detection, creating risks such as information leakage and single points of failure [6]. To address these issues, SED (Secure and Efficient data Deduplication) is proposed within a Joint Cloud storage framework [6][16]. SED enables secure cross-cloud deduplication, enhancing scalability and reliability across multiple providers. It supports dynamic data updates, ensuring that file modifications do not affect deduplication efficiency [6]. Security analysis demonstrates semantic security and resistance to various attacks, while maintaining low computational overhead [6]. However, its joint cloud architecture increases system complexity, requires coordination among providers, and introduces dependency on inter-cloud trust and management [6].

3. Methodology

The proposed system enhances security in deduplicated cloud storage by combining Proof of Ownership (PoW) with Proof of Storage (PoS). Deduplicated storage systems optimize space and reduce costs by storing only a single instance of identical files. While conventional PoW allows users to prove ownership through cryptographic proofs of file content or blocks, it only validates ownership at a single point in time. This creates a vulnerability: if an attacker knows a file's hash, they could claim access without actually possessing the file. To overcome this, the proposed system integrates PoS alongside PoW. Users are required to periodically demonstrate that they continue to retain the file. The server issues cryptographic challenges, and only those who respond correctly maintain access to the deduplicated data. This continuous verification ensures that only legitimate owners can access files over time, preventing unauthorized claims and strengthening security. Importantly, the system maintains the efficiency of deduplication by avoiding redundant storage and reducing bandwidth usage. By combining PoW and PoS, it provides a

reliable framework for secure, continuous access, ensuring data privacy, integrity, and ownership verification. This approach is well-suited for large-scale cloud environments, balancing storage efficiency with robust security for sensitive information.

4. Proposed System

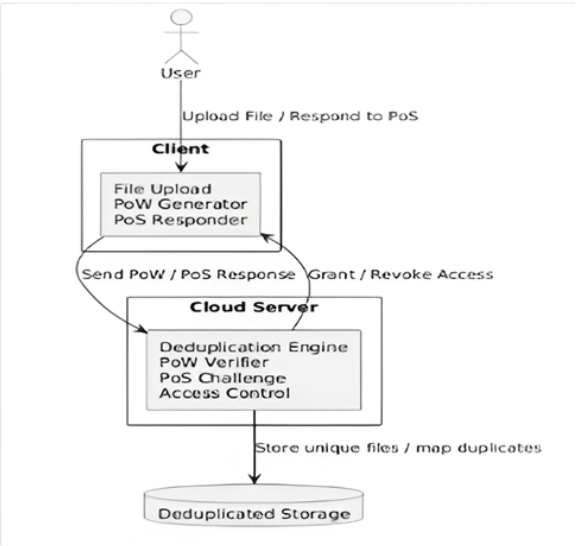

Fig. 1. Proposed system architecture

In the architecture of the proposed system PoW + PoS deduplicated cloud storage system consists of three main components: the client, the cloud server, and the deduplicated storage. On the client side, users interact with modules responsible for uploading files, generating Proof of Ownership (PoW), and responding to Proof of Storage (PoS) challenges. The cloud server hosts the deduplication engine, which ensures that only a single instance of identical files is stored, thereby optimizing storage space and reducing bandwidth usage.

The server also includes a PoW verifier to authenticate initial file ownership, a PoS challenge module to periodically verify continuous possession of files, and an access control manager that grants or revokes user access based on verification results. Finally, the deduplicated storage component stores unique files and maintains mappings for duplicate entries. This architecture ensures secure, efficient, and continuous access to files while maintaining the benefits of deduplication and protecting sensitive data in large-scale cloud environments.

A. User Registration and Authentication

The User Registration and Authentication module serves as the foundation of the proposed secure deduplicated cloud storage system. It ensures that only legitimate users can

access storage services and participate in deduplication and verification processes. During registration, each user creates a unique identity, and the system securely generates cryptographic credentials, such as public–private key pairs or secret keys, which are used in subsequent Proof of Ownership (PoW) and Proof of Storage (PoS) operations. Authentication mechanisms, including secure login protocols and encrypted communication channels, protect user credentials from interception or misuse. This module prevents unauthorized entities from initiating fake ownership claims or attempting to exploit deduplication mechanisms. Additionally, it maintains a secure user database that stores identity information, access privileges, and verification status. Proper identity binding ensures that file ownership is always associated with a verified user. By establishing strong authentication controls at the initial stage, this module minimizes risks such as impersonation, replay attacks, and unauthorized access attempts. It also supports scalability by efficiently managing multiple users in large-scale cloud environments.

B. File Upload and Deduplication Module

The File Upload and Deduplication module is responsible for optimizing storage utilization while maintaining efficiency. When a user uploads a file, the system first computes a cryptographic hash value (e.g., SHA-256) to uniquely identify the file's content. This hash is compared against existing entries in the cloud storage index to determine whether the file already exists. If a match is found, the system avoids storing a duplicate copy and instead creates a logical reference pointer linking the user to the existing file instance.

If no match is found, the file is securely stored, and its hash is recorded in the metadata index. This approach significantly reduces redundant storage and minimizes bandwidth consumption during data transmission.

The module ensures that deduplication does not compromise performance by efficiently managing indexing and lookup processes. Furthermore, it integrates securely with the PoW mechanism, ensuring that deduplication only occurs after ownership verification. By eliminating unnecessary replication of identical data, this module reduces operational costs and supports scalability in large cloud environments. It plays a crucial role in maintaining the balance between storage efficiency and secure file management within the overall framework.

C. Proof of Ownership (PoW) Verification Module

The Proof of Ownership (PoW) Verification module ensures that only users who genuinely possess a file can claim access to its deduplicated version. Instead of allowing access solely based on hash matching, the system requires the user to generate a cryptographic proof derived from the actual file content or selected data blocks. The cloud server verifies this proof without requiring the full file upload, thereby saving bandwidth and computational resources. This process prevents attackers from exploiting deduplication tags or file hashes to gain unauthorized access. By validating possession at the time of access request, PoW eliminates vulnerabilities associated with simple hash-based verification. The module employs secure challenge–response protocols to confirm that the claimant truly holds the file. It is optimized to minimize input/output operations and computational overhead on client devices, ensuring practicality for real-world deployment. Although PoW provides strong initial ownership verification, it remains a static mechanism, confirming possession only at a single point in time. Nevertheless, it serves as a critical security layer that

protects against fraudulent ownership claims and strengthens trust in deduplicated cloud storage systems.

D. Proof of Storage (PoS) Ccontinuous Verification Module

The Proof of Storage (PoS) Continuous Verification module enhances security by introducing dynamic and periodic ownership validation. Unlike PoW, which verifies possession only once, PoS requires users to periodically demonstrate that they still retain the file over time. The cloud server generates cryptographic challenges based on random file segments or metadata, and users must respond correctly using their stored data. Successful responses confirm continued possession, while failure results in restricted or revoked access rights. This module effectively prevents scenarios where an attacker temporarily gains access or uses stolen hash information to claim ownership. By ensuring ongoing verification, PoS provides long-term protection against unauthorized access and strengthens data integrity guarantees.The challenge–response mechanism is designed to be lightweight, minimizing performance impact while maintaining robust security. Additionally, it supports scalability by efficiently handling periodic verification for multiple users in large cloud systems. Through continuous monitoring and validation, the PoS module ensures that only legitimate users maintain uninterrupted access to deduplicated files, significantly improving reliability and trust in the storage framework.

E. Access Control and Authorization Module

The Access Control and Authorization module governs user permissions and enforces security policies within the system. After successful PoW and PoS verification, users are granted appropriate access rights based on predefined authorization rules. This module ensures that file access is strictly limited to verified owners and authorized users. It manages privilege assignment, access revocation, and policy enforcement in dynamic cloud environments. If a user fails periodic PoS verification or violates security policies, the system automatically revokes or restricts access to prevent potential misuse. The module maintains secure logs of access attempts, verification outcomes, and authorization updates to support auditing and accountability. By integrating closely with authentication and verification mechanisms, it provides a comprehensive security layer that controls how and when data can be accessed. This module is essential for maintaining confidentiality and preventing unauthorized data exposure in multi-user environments. It ensures that deduplication efficiency does not compromise strict access governance, thereby reinforcing the system's overall reliability and compliance with security standards.

F. Security and Integrity Monitoring Module

The Security and Integrity Monitoring module oversees the overall health, reliability, and protection of the cloud storage system. It continuously monitors system activities, verification logs, and access patterns to detect suspicious behavior or potential attacks. By analyzing anomalies such as repeated failed PoS challenges or abnormal access requests, the module helps identify malicious users or compromised accounts. It ensures that data integrity is preserved by validating file consistency and preventing unauthorized modifications. Additionally, this module maintains secure audit trails, which are essential for accountability, compliance, and forensic analysis. Despite its monitoring functions, the module is designed to operate efficiently without disrupting deduplication performance. By combining surveillance, logging, and integrity checks, it strengthens trust in the PoW–PoS framework and ensures that storage efficiency is maintained alongside robust, continuous security protection in large-scale cloud environments.

5. Experimental Results

The results demonstrate that integrating Proof of Ownership (PoW) with Proof of Storage (PoS) significantly enhances the security of deduplicated cloud storage systems without compromising storage efficiency. Experimental evaluation shows that the combined framework effectively prevents hash-based and unauthorized ownership claims, as users must first prove genuine possession of the file and then periodically confirm continued storage. The deduplication mechanism successfully reduces redundant data storage and bandwidth consumption, maintaining cost-effectiveness similar to conventional systems. Performance analysis indicates that the additional overhead introduced by PoS verification is minimal and scalable for large user environments. The discussion highlights that while the hybrid PoW–PoS model slightly increases computational and communication operations compared to basic deduplication, it provides substantially stronger guarantees of data integrity, privacy, and continuous ownership validation, making it suitable for secure large-scale cloud deployments.

A. File Hash Value H(F)

The File Hash Value is a fundamental parameter used to detect duplicate files in the proposed deduplicated cloud storage system. When a user uploads a file FFF, the system applies a cryptographic hash function to generate a fixed-length digest that uniquely represents the file's content. This process is mathematically expressed as:

File hash value H(F):

$$H(F) = \mathcal{H}(F)$$

Where, H denotes a secure hash function such as SHA-256. The resulting hash value serves as a unique identifier for the file and is stored in the cloud index. If another user uploads a file that produces the same hash value, the system treats it as identical and avoids storing a redundant copy, instead creating a reference pointer to the existing file. This mechanism significantly reduces storage consumption and bandwidth usage. The collision resistance property of the hash function ensures that the probability of two different files generating the same hash is negligible, thereby maintaining reliability in duplicate detection. In the proposed framework, the hash value is used strictly for identifying redundancy, not for granting ownership, which prevents misuse through hash-based attacks.

By combining strong cryptographic hashing with secure indexing, this parameter enables efficient, scalable, and reliable deduplication while preserving data integrity.

B. Ownership Proof Value (PoW Proof)

The Ownership Proof Value is a critical security parameter that verifies whether a user genuinely possesses the entire file before claiming access to a deduplicated copy. Instead of uploading the full file again, the user generates a cryptographic response derived from the file's content when challenged by the server. The proof generation can be represented as:

Ownership proof value (PoW Proof):

$$P = \sum_{i \in S} r_i \cdot H(b_i) \mod p$$

Where, S represents a randomly selected subset of file block indices, b_i denotes the corresponding file blocks, r_i are random challenge coefficients generated by the server, and p is a large prime number. Only a user who possesses all required blocks can compute the correct value of P. The server independently computes the expected result and verifies the equality. This ensures that access is granted only to legitimate file holders and prevents attackers from exploiting deduplication tags or hash values. The PoW mechanism provides efficient verification with minimal communication overhead, as only small proof values are transmitted instead of entire files. Although it verifies ownership at a single time instance, it forms a strong first layer of defense in the combined PoW–PoS framework.

C. Storage Challenge

The Storage Challenge–Response parameter enables continuous verification of file possession over time. Unlike PoW, which validates ownership only once, Proof of Storage (PoS) periodically challenges users to confirm that they still retain the file locally. The response computation can be expressed as:

Storage challenge:

$$R = H(b_i \parallel r)$$

Where, b_i is a randomly selected file block chosen by the server, r is a random nonce, and ‖ denotes concatenation. The server sends the challenge containing i and r, and the user computes the hash of the concatenated block and nonce. The server verifies the correctness of RRR using the stored metadata. Since the nonce changes each time, previous responses cannot be reused, preventing replay attacks. This mechanism ensures that only users who continuously possess the actual file can maintain access rights. If a user fails to provide a valid response, access privileges are revoked. The PoS parameter introduces minimal computational overhead while significantly enhancing long-term security. By enforcing periodic verification, the system prevents unauthorized users from retaining access based solely on initial proof, thereby strengthening integrity and trust in large-scale cloud storage environments.

D. User Identity and Secret Keys (UID, SK)

User Identity and Secret Keys form the cryptographic backbone of the proposed secure deduplicated cloud storage system. Each user is assigned a unique identifier (UID) during registration, which is permanently linked to their stored files and access privileges. Along with this identity, a cryptographic key pair is generated, consisting of a private Secret Key (SK) and a corresponding Public Key (PK). The secret key remains securely with the user, while the public key is registered with the cloud server for verification. Whenever a user attempts to prove ownership or respond to a storage challenge, the proof is cryptographically bound to their identity using a digital signature mechanism. This can be represented as:

User identity and secret keys (UID, SK):

$$\sigma = \mathrm{Sign}_{SK}(H(F))$$

Where, $H(F)$ is the hash of the file and $\boldsymbol{\sigma}$ is the generated signature. The server verifies this signature using the user's public key. This ensures that only the legitimate user possessing the correct secret key can generate valid proofs. Even if an attacker obtains the file hash, they cannot forge a valid signature without the secret key. This mechanism guarantees authentication, non-repudiation, accountability, and secure identity binding in large-scale cloud environments

E. Access Control Status/Verification Flag (Vstatus)

The Access Control Status or Verification Flag determines whether a user is authorized to access a deduplicated file at any given time. It acts as a dynamic security indicator that integrates the results of Proof of Ownership (PoW) and Proof of Storage (PoS). Instead of granting permanent access after a single verification, the system continuously evaluates whether the user satisfies both initial and periodic verification requirements. The authorization condition can be expressed as:

Access control status/verification flag (Vstatus):

$$V_{status} = \mathrm{PoW}_{valid} \wedge \mathrm{PoS}_{valid}$$

If both conditions are true, the verification flag is set to 1, granting access. If either verification fails, the flag becomes 0, and access is revoked. This logical condition ensures that only users who genuinely possess and continuously retain the file maintain access rights. The flag is updated periodically based on server-issued challenges and verification responses. By enforcing this conditional access model, the system prevents unauthorized retention of privileges and strengthens long-term security. The Verification Flag provides a simple yet powerful control mechanism that ensures confidentiality, integrity, and continuous ownership validation in deduplicated cloud storage environments.

F. Comparison Graph for the Proposed System

The comparison graph illustrates the performance differences between Traditional Deduplication, PoW-Based Systems, and the Proposed PoW + PoS System in terms of overall security and reliability. Traditional deduplication

achieves high storage efficiency but lacks strong security mechanisms, making it vulnerable to hash-based and unauthorized access attacks. As a result, its overall security performance is comparatively lower. The PoW-based system improves security by requiring users to prove file ownership before access is granted, thereby reducing the risk of tag-based attacks. However, it performs verification only once at the time of access request, leaving a gap in continuous protection. In contrast, the proposed PoW + PoS system combines initial ownership verification with periodic storage validation. This dual-layer approach ensures that users not only possess the file initially but also retain it over time. Continuous challenge–response verification significantly strengthens protection against unauthorized claims. Although it introduces minimal additional overhead, the proposed system achieves the highest security, reliability, and trust level, making it the most robust solution for large-scale deduplicated cloud environments.

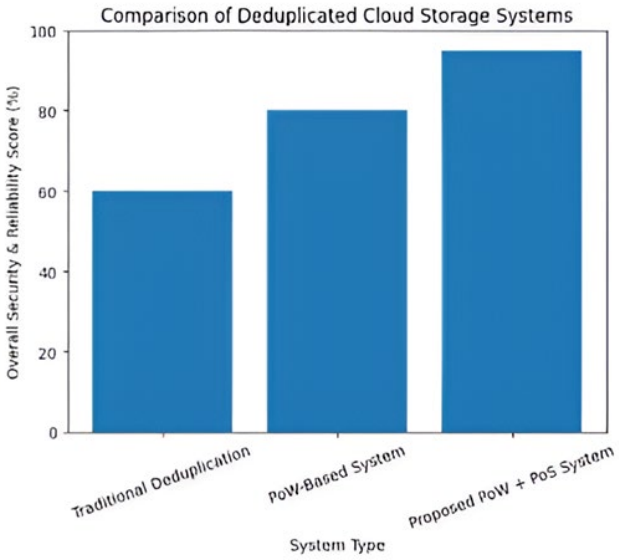

Fig. 5. Comparison graph for the proposed system

6. Conclusion

In conclusion, the comparison clearly demonstrates that the proposed PoW + PoS system provides superior security and reliability compared to traditional deduplication and standalone PoW-based approaches. While traditional deduplication effectively optimizes storage space, it lacks sufficient protection against hash-based and unauthorized access attacks. PoW-based systems strengthen security by verifying file ownership at the time of access, but their static verification model leaves potential vulnerabilities over time. The proposed system addresses these limitations by integrating both Proof of Ownership and Proof of Storage, ensuring not only initial validation but also continuous possession verification. This dynamic challenge–response mechanism significantly enhances protection against fraudulent claims and unauthorized access. Although the combined approach introduces slight computational overhead, the improvement in data integrity, privacy, and long-term access control outweighs this cost. Therefore, the proposed framework represents a secure, scalable, and efficient solution for modern deduplicated cloud storage environments.

References

[1] K. Ghassabi and P. Pahlevani, "DEDUCT: A secure deduplication of textual data in cloud environments."
[2] J. Dave, K. Ram R., P. Patil, and H. Patil, "Secure and efficient ownership verification for deduplicated cloud computing systems."
[3] M. Lee and M. Seo, "Secure and efficient deduplication for cloud storage with dynamic ownership management."
[4] A. Goel, C. Prabha, M. Malik, and P. Sharma, "Security concerns and data breaches for data deduplication techniques in cloud storage: A brief meta-analysis."
[5] H. Lohia, S. A. Lakade, and Y. R. Gurav, "Secure data storage optimization over cloud using encrypted cloud data deduplication technique," *Journal of Science and Technology*, vol. 9, no. 1, pp. 131–138, Jan. 2024.
[6] K. D. Santhoshi, "Secure and efficient data deduplication in JointCloud storage."
[7] K. S. M. Bukari and K. Nirmala, "Secure and efficient approach for enhancing cloud data deduplication through chaotic elliptic curve cryptography."
[8] "Analysis on deduplication techniques for storage of data in cloud."
[9] "Deduplication in cloud computing for improvising efficiency towards potential practical usage."
[10] "Data de-duplication engine for efficient storage management."
[11] R. Du, L. Deng, J. Chen, K. He, and M. Zheng, "Proofs of ownership and retrievability in cloud storage," in *Proc. IEEE 13th Int. Conf. Trust, Security and Privacy in Computing and Communications*, 2014, pp. 328–335.
[12] J. Dave, M. Bhatt, and D. Pancholi, "Secure proof of ownership for deduplicated cloud storage system," *International Journal of Information and Computer Security*, vol. 21, no. 1–2, pp. 205–228, 2023.
[13] J. Dave, P. Faruki, V. Laxmi, B. Bezawada, and M. Gaur, "Secure and efficient proof of ownership for deduplicated cloud storage," in *Proc. 10th Int. Conf. Security of Information and Networks*, 2017, pp. 19–26.
[14] R. Di Pietro and A. Sorniotti, "Proof of ownership for deduplication systems: A secure, scalable, and efficient solution," *Computer Communications*, vol. 82, pp. 71–82, 2016.
[15] M. Song, Z. Hua, Y. Zheng, T. Xiang, and X. Jia, "Simless: A secure deduplication system over similar data in cloud media sharing," *IEEE Transactions on Information Forensics and Security*, 2024.
[16] D. Zhang, J. Q. Le, N. K. Mu, J. H. Wu, and X. F. Liao, "Secure and efficient data deduplication in JointCloud storage," *IEEE Transactions on Cloud Computing*, vol. 11, no. 1, pp. 156–167, 2023.
[17] X. W. Ma, W. Y. Yang, Y. S. Zhu, and Z. Q. Bai, "A secure and efficient data deduplication scheme with dynamic ownership management in cloud computing," in *Proc. IEEE Int. Performance, Computing, and Communications Conf. (IPCCC)*, Austin, TX, USA, 2022, pp. 1–8.
[18] H. Lohia, S. A. Lakade, and Y. R. Gurav, "Secure data storage optimization over cloud using encrypted cloud data deduplication technique," *Journal of Science and Technology*, vol. 9, no. 1, pp. 131–138, Jan. 2024.
[19] X. X. Yu, H. Bai, Z. Yan, and R. Zhang, "VeriDedup: A verifiable cloud data deduplication scheme with integrity and duplication proof," *IEEE Transactions on Dependable and Secure Computing*, vol. 20, no. 1, pp. 680–694, 2023.
[20] S. P. G., N. R. K., V. G. Menon, V. P., M. Abbasi, and M. R. Khosravi, "A secure data deduplication system for integrated cloud-edge networks," *Journal of Cloud Computing*, vol. 9, no. 1, p. 61, 2020.
[21] S. Polepaka, B. Gayathri, S. Ayoub, H. Sharma, Y. S. Moudgil, and S. Kannan, "Privacy preserving encryption with optimal key generation technique on deduplication for cloud computing environment," in *Proc. 2022 Int. Conf. Automation, Computing and Renewable Systems (ICACRS)*, 2022, pp. 464–470.
[22] T. Benil and J. Jasper, "Blockchain based secure medical data outsourcing with data deduplication in cloud environment," *Computer Communications*, vol. 209, pp. 1–13, 2023.
[23] C. Fan, S. Y. Huang, and W. C. Hsu, "Hybrid data deduplication in cloud environment," in *Proc. Int. Conf. Information Security and Intelligent Control*, 2012, pp. 174–177.

[24] W. K. Ng, Y. Wen, and H. Zhu, "Private data deduplication protocols in cloud storage," in *Proc. 27th Annual ACM Symposium on Applied Computing*, 2012, pp. 441–446.

[25] S. Halevi, D. Harnik, B. Pinkas, and A. Shulman-Peleg, "Proofs of ownership in remote storage systems," in *Proc. ACM Conf. Computer and Communications Security (CCS)*, Chicago, IL, USA, 2011, pp. 491–500.