The Scientific Temper (2024) Vol. 15 (4): 3165-3172
Doi: 10.58414/SCIENTIFICTEMPER.2024.15.4.25
E-ISSN: 2231-6396, ISSN: 0976-8653
https://scientifictemper.com/

RESEARCH ARTICLE

ESPoW: Efficient and secured proof of ownership method to enable authentic deduplicated data access in public cloud storage

Sabeerath K., Manikandasaran S. Sundaram

Abstract

The exponential growth of data in cloud environments necessitates efficient storage management solutions. Data deduplication, a technique that eliminates redundant data, has emerged as a key strategy to optimize storage utilization and reduce costs. However, deduplication introduces security challenges, particularly in verifying data ownership and protecting against unauthorized access. This paper presents efficient and secured proof of ownership (ESPoW), a novel proof-verifier technique designed to authenticate data ownership in deduplicated cloud storage environments. ESPoW utilizes a challenge-response mechanism and a unique secret value for each data file to ensure that only legitimate users can access their data, even in the presence of encrypted storage. Through rigorous experimentation and performance analysis, ESPoW demonstrates superior computational efficiency and enhanced security compared to existing methods. This approach provides a robust framework for secure and efficient deduplication in cloud storage, safeguarding sensitive data while optimizing storage resources.

Keywords: Data deduplication, Cloud storage security, proof of ownership, Authentication mechanism, Challenge-response protocol, Secure data access.

Introduction

Cloud computing has become a pervasive term in today's digital landscape, offering vast storage capabilities, immediate accessibility, and user-friendly interfaces from virtually any location and at any time. The demand for cloud services has surged, leading to an exponential increase in both the number of users and the volume of data stored in the cloud. This surge has necessitated higher memory capacities and increased upload bandwidth to accommodate the growing needs (Ding W. *et al*., 2017).

To address the challenges associated with the massive influx of data, cloud storage systems have adopted data deduplication techniques. Data deduplication is a process aimed at eliminating redundant copies of data files, thereby reducing the required storage space and conserving bandwidth. When users upload identical data to cloud storage, it can result in unnecessary duplication, which in turn, raises concerns about data ownership and the protection of sensitive information (Edavalath S. *et al*., 2023).

Services like Dropbox and Google Drive have implemented deduplication procedures to mitigate the inefficiencies associated with redundant data storage (Mgwalima T. *et al*., 2022). These procedures help in minimizing storage costs and optimize the use of network bandwidth. By ensuring that only unique instances of data are stored, these services can effectively reduce overhead and improve system performance (Jiang S. *et al*., 2020).

Data security remains a paramount concern for users who store their private information in the cloud. To protect their data from unauthorized access and potential threats, users often encrypt their data before uploading it to the cloud. However, this encryption can complicate the deduplication process. To address these security challenges, it is crucial to implement mechanisms such as convergent encryption and data ownership verification. These techniques ensure that

PG and Research Department of Computer Science, Adaikalamatha College, Vallam, Thanjavur. Affiliated to Bharathidasan University, Trichy.

***Corresponding Author:** Sabeerath, K., PG and Research Department of Computer Science, Adaikalamatha College, Vallam, Thanjavur. Affiliated to Bharathidasan University, Trichy., E-Mail: sabeerathk@gmail.com

How to cite this article: Sabeerath, K., Sundaram, M. S. (2024). ESPoW: Efficient and secured proof of ownership method to enable authentic deduplicated data access in public cloud storage. The Scientific Temper, **15**(4):3165-3172.

Doi: 10.58414/SCIENTIFICTEMPER.2024.15.4.25

Source of support: Nil

Conflict of interest: None.

Received: 21/10/2024 **Accepted:** 26/11/2024 **Published** : 20/12/2024

deduplicated data remains secure and that only authorized users can access their information (Selvaraj R. wt al., 2023).

The primary benefit of data deduplication lies in its ability to identify data similarities and reduce the unnecessary allocation of storage space. By storing only one copy of redundant data, cloud storage providers can significantly decrease the amount of space needed, thereby lowering storage costs and improving efficiency (Song M. *et al*., 2023).

Previous research has explored various techniques for managing deduplicated data effectively. Building on these studies, the current paper proposes a novel proof-verifier technique designed to authenticate users' data ownership. This technique ensures that users can access their data only after successfully proving their ownership. By implementing such verification methods, cloud storage systems can enhance security and maintain the integrity of deduplicated data. While cloud computing offers immense benefits in terms of storage and accessibility, it also brings challenges related to data security and efficiency. Data deduplication, coupled with robust encryption and ownership verification mechanisms, provides a viable solution to these challenges, ensuring secure and efficient cloud storage.

Related Work

The concept of Proof of Ownership (PoW) involves verifying data ownership through a hashed value created when the data is stored in the cloud. However, a challenge arises when a small hashed value is used as a proxy for the entire file in target-level deduplication. Current deduplication research often uses this small digested value, which is the hashed value for the file, making it highly unreliable. Malicious users, without possessing the file, might obtain the hashed value from another source and then masquerade as legitimate users. This section reviews some of the related articles addressing the Proof of Ownership in Deduplicating the data.

Managing the increasing amount of outsourced data in cloud storage through deduplication, which reduces storage costs by eliminating redundant data, presents a critical challenge. However, as noted by Dave *et al*. (2017), deduplication introduces security vulnerabilities, such as unauthorized access to data. To address this issue, the authors propose a secure and efficient Proof of Work technique, utilizing random matrix-based challenges and convergent encryption to ensure that only legitimate file owners can access deduplicated data. The system architecture, which includes cloud clients, a cloud server, and cloud storage, is designed to mitigate risks posed by semi-trusted servers and malicious clients. The system supports operations such as file upload, download, deletion, and updates, ensuring data confidentiality and availability while preventing unauthorized access and tag inconsistencies. The authors also demonstrate that this approach is efficient, incurring minimal computational overhead compared to traditional methods. Their study reviews pertinent research, highlighting the benefits of the proposed solution and recommending potential improvements.

Efficient management of cloud storage by eliminating duplicate files through data deduplication is a critical focus in addressing both authentication and data security challenges in cloud technology. Sai Kiran, P., Reddy, J. B., and Sainadh, K. G. (2017) propose a system where users generate a hash value of their files using the SHA-256 technique to prevent duplicate uploads. To further enhance security, the system incorporates encryption and one-time passwords (OTP) to verify ownership and protect the data. This method ensures data confidentiality and reliability, maintaining security even if a user account is compromised, as both the OTP and encrypted text are required to access the data. According to the authors, convergent encryption proves effective for deduplication, with the study also recommending the addition of security features such as security questions for future improvements.

The importance of secure data deduplication in cloud storage, particularly when handling dynamic ownership changes, is a critical focus. Mohammed Parveez and Srinidhi H. R. (2018) propose a server-side deduplication method that maintains data security and integrity during ownership transitions. This approach uses randomized convergent encryption and secure group key distribution to prevent data leaks to former users and safeguard against inquisitive cloud servers. By dynamically updating encryption keys and managing ownership, the system ensures that only authorized users can access deduplicated data. The authors demonstrate that this method enhances data privacy, prevents unauthorized access, and maintains deduplication efficiency, offering notable improvements in security and computational performance over existing methods.

A method for ensuring secure data deduplication in cloud storage systems is presented by Jay Dave *et al*. (2020), proposing a Proof of Work (PoW) protocol that utilizes Merkle Trees to efficiently and securely verify data ownership while maintaining user data privacy. This approach tackles the issue of multiple users uploading the same data by ensuring that only authorized users can claim ownership. The Merkle Tree-based PoW scheme enhances security through the use of convergent encryption, which generates unique cryptographic hashes for data chunks, preventing unauthorized access. According to the authors, experimental results show that the proposed scheme effectively balances security and efficiency, making it a practical solution for secure data deduplication in cloud environments.

A new client-side deduplication method aimed at addressing challenges like collusive authentication, brute-force, and duplicate-faking attacks is introduced in the work of Tian, G., Ma, H., Xie, Y., and Liu, Z. (2020). The authors propose a randomized deduplication protocol that leverages a dynamic Key-Encrypting Key (KEK) tree for enhanced ownership management and data sharing.

This KEK tree adapts dynamically to changes in ownership, improving both the flexibility and security of the system. Additionally, the scheme incorporates measures to prevent data loss and ensure data integrity. The paper demonstrates through security and performance analyses that the solution effectively meets security requirements while optimizing system resource management.

A comprehensive overview of current data deduplication techniques, along with the introduction of an innovative Proof of Work (PoW) scheme, is presented by Amer Al-Amer and Osama Ouda (2021). This new scheme focuses on enhancing both security and efficiency in cloud storage. The authors emphasize the importance of preventing unauthorized data access while maintaining storage efficiency. Their proposed PoW scheme utilizes convergent encryption and Merkle trees to securely verify data ownership, enabling clients to prove ownership without revealing sensitive information. The paper demonstrates through experimental results that this scheme significantly reduces computational overhead and enhances data security, offering a practical solution for cloud storage environments.

Addressing the security challenges of data deduplication in cloud storage systems is the focus of Dave, J., Meghna Bhatt, and Deep Pancholi (2023). Deduplication, which reduces storage and communication costs by eliminating redundant data, also poses security risks, such as unauthorized access via file hash knowledge. To mitigate these risks, the authors propose a secure Proof of Work (PoW) scheme where the server randomly selects a set of file blocks as a challenge to verify ownership. Only a legitimate user who possesses the entire file can correctly respond to these challenges, preventing adversaries from exploiting just the hash values of file blocks. The authors establish a lower bound on the number of blocks required for the challenge to ensure security. Through implementation and comparative analysis, they demonstrate that their scheme is effective and outperforms existing state-of-the-art methods.

A protocol aimed at improving data deduplication in cloud storage systems, with a focus on enhancing computational efficiency and ensuring privacy, is presented by Mira Lee and Minhye Seo (2023). Building on the SeDaSC protocol, the proposed method reduces the computational load on the client side by utilizing a cryptographic server (CS) for encryption while maintaining data privacy through message-locked encryption (MLE). This approach also includes dynamic management of client ownership, ensuring both forward and backward secrecy. When ownership of data changes, encryption keys are securely updated, preventing former owners from accessing the data and ensuring new owners cannot access previous versions. The authors provide a detailed outline of the system architecture and security requirements, along with an in-depth analysis of the protocol's computational efficiency and its enhancements to security.

A novel solution to improve both security and efficiency in cloud storage deduplication is proposed by Song M., *et al*. (2024). The authors propose a layered architecture that integrates encryption within the deduplication process, ensuring data confidentiality while minimizing storage space. By dividing the deduplication process into multiple layers, the system offers flexibility and improved performance. The evaluation demonstrates that LSDedup enhances storage efficiency without significant computational overhead, outperforming traditional methods. However, the complexity of managing the layers and potential scalability issues in large-scale environments are areas that could be further explored. Overall, the paper provides a significant contribution to secure cloud storage, balancing the need for security and resource optimization (Table 1).

Diverse experts have suggested several methodologies for establishing data ownership. Nevertheless, the majority of existing systems depend on a singular hashed value derived from the data file to authenticate ownership. This method is vulnerable to hacking attempts by nefarious individuals. Consequently, enhancing the verification process is essential to guarantee secure access to data stored in the cloud. This work presents a new proof-verifier technique named ESPoW (Enhanced Secure Proof of Ownership) to mitigate these issues. ESPoW is engineered to validate and authenticate users with a distinct secret value produced for each data file. In contrast to conventional techniques, ESPoW utilizes a challenge-response mechanism for user verification. This method improves security by guaranteeing that only authorized users, capable of accurately answering the challenges derived from the secret value, can access the data.

ESPoW functions by presenting a challenge to the user, who is required to supply the accurate response derived from the confidential value linked to the data file. This procedure guarantees that, even if an unauthorized individual acquires the hashed value, they cannot access the system without the associated secret value. The implementation of this enhanced verification technique considerably enhances the security of data stored in cloud settings, safeguarding it from illegal access and potential breaches.

ESPoW: Methodology for Proof Verification

The proposed ESPoW is utilized while accessing or downloading the file. Subsequent to data upload, it may be downloaded contingent upon user verification of proof. It is a system for validating user authentication. Upon verifying data ownership, the user gains access to it. The file is accessible according to the Proof of Work. The ESPoW verifier validates the data owner through the challenge-response mechanism.

Each data file saved in the cloud is assigned a confidential value. The secret value SVAL (Secret Value) is generated with

Table 1: significant contribution to secure cloud storage, balancing the need for security and resource optimization

Article	*Focus*	*Security Mechanism*	*Ownership Management*	*Data Confidentiality*	*Computational Efficiency*	*Experimental Results*	*Future Enhancements Suggested*	*Advantages*	*Disadvantages*	*Unique Contribution*
Dave, J., *et al*. (2017)	Proof of Ownership (PoW) using random matrix-based challenges and convergent encryption	Random matrix-based challenges, convergent encryption	Handles threats from semi-trusted servers and malicious clients	Ensures data confidentiality and availability	Minimal computation overhead compared to conventional methods	Demonstrates efficiency with minimal computation overhead	Future enhancements to deduplication system	Efficient, minimal overhead, strong data confidentiality	Potential complexity in implementation	Combines random matrix challenges with convergent encryption for secure PoW
Sai Kiran, P., *et al*. (2017)	Data Deduplication with Proof of Ownership using SHA-256 and OTP	SHA-256, encryption, one-time passwords	Verifies ownership through hash value and OTP	Ensures data confidentiality and reliability	Uses SHA-256 for hash generation, ensuring minimal overhead	Validates the effectiveness of SHA-256 and OTP for deduplication	Further security features like security questions	Efficient deduplication, reliable ownership verification	Dependent on SHA-256 and OTP effectiveness	Integrates SHA-256 with encryption and OTP for secure deduplication
Parveez & Srinidhi (2018)	Secure Data Deduplication with dynamic ownership management	Randomized convergent encryption, secure group key distribution	Updates encryption keys and ownership groups dynamically	Prevents data leakage to revoked users and honest-but-curious cloud servers	Offloads laborious tasks to the cloud server, ensures efficient ownership management	Shows significant improvements in security and computational performance compared to existing methods	Focus on improving security and efficiency in dynamic environments	Enhances data privacy, prevents unauthorized access, efficient ownership management	Requires efficient key management to handle dynamic ownership changes	Introduces dynamic ownership management with randomized convergent encryption and secure group key distribution
Dave, J., *et al*. (2020)	Secure PoW using Merkle Tree for Deduplicated Storage	Merkle Tree-based PoW, convergent encryption	Ensures only authorized users can claim ownership	Ensures user data privacy through convergent encryption	Balances security and efficiency	Balances security and efficiency through experimental results	Further optimization of security and efficiency	Strong security with minimal computational overhead	Complexity in managing Merkle Trees	Utilizes Merkle Trees for efficient and secure PoW
Tian, G., *et al*. (2020)	Randomized deduplication with dynamic Key-Encrypting Key (KEK) tree	Dynamic KEK tree, randomized deduplication	Adjusts dynamically to ownership changes	Prevents data loss, ensures data integrity	Optimizes system resource management	Demonstrates effectiveness in meeting security requirements while optimizing resources	Additional measures to enhance security	Flexible, enhances system security and efficiency	Complexity in managing dynamic KEK trees	Introduces dynamic KEK tree for enhanced ownership management

Al-Amer & Ouda (2021)	Secure PoW scheme using convergent encryption and Merkle trees	Convergent encryption, Merkle trees	Secure verification without exposing sensitive information	Maintains data privacy through secure PoW	Reduces computational overhead, improves efficiency	Shows significant reduction in computational overhead and improvement in data security	Further development to balance security and efficiency trade-offs	Secure and efficient, significant reduction in overhead	Potential complexity in managing Merkle Trees	Combines convergent encryption with Merkle trees for efficient and secure PoW
Dave, J., Bhatt, M., & Pancholi, D. (2023)	Secure Proof of Ownership with random block selection challenge	Random block selection challenge	Verifies ownership by server-selected file blocks	Prevents unauthorized access through secure PoW	Demonstrates effectiveness and performance through implementation	Establishes a lower bound on the number of blocks needed for challenges, demonstrating performance against state-of-the-art	Enhancing the scheme to handle more sophisticated attacks	Effective security against unauthorized access, strong performance	Implementation complexity	Proposes a new PoW scheme with random block selection challenge
Lee & Seo (2023)	Secure Deduplication with Dynamic Ownership Management	Cryptographic server for encryption, message-locked encryption (MLE)	Dynamic management with secure key updates	Maintains data privacy with forward and backward secrecy	Reduces client-side computational load, enhances efficiency	In-depth analysis shows computational efficiency and security enhancements	Enhancing protocols to manage more complex ownership changes and improve computational efficiency	Strong security and efficiency, maintains data privacy	Potential complexity in cryptographic server management	Enhances computational efficiency and privacy through cryptographic server and message-locked encryption (MLE)
Song, M., *et al*. (2024)	Developing a secure deduplication system, LSDedup to enhance storage efficiency.	LSDedup incorporates encryption techniques to enable deduplication.	Ownership of data remains secure through cryptographic techniques.	The encryption of data before deduplication ensures that even if unauthorized access occurs.	block-level deduplication, which reduces the storage footprint while maintaining an acceptable level of processing overhead	Significant improvements in storage savings and computational efficiency.	further improve scalability and reduce the complexity of managing multiple layers	secure, efficient deduplication process that protects data confidentiality, optimizes storage space.	complexity of the layered architecture might pose challenges for implementation and management, and scalability under extremely large-scale deployments	layered, secure deduplication system that integrates encryption into the deduplication process

the Token, file Tag, user ID, and user password. Token and Tag facilitate data identification, while user ID and password are utilized to ascertain the permitted user. This confidential value is saved alongside further data details. When a user attempts to access the data, they provide their credentials and specify the file they wish to access.

SVAL represents the stored value in the database upon data upload, whereas SVAL' is generated when access to a specific file in storage is requested, with the verification of proof occurring when SVAL equals SVAL'. If both values are equivalent, the user may be granted access to the file. The user is not granted authorization to access the file in the cloud storage.

Functionality of ESPoW

ESPoW is an authentication technique that grants access permissions to data users. The user employs their user_id and password to access the systems, which should authenticate them at login. Users must be authenticated to access files in the cloud. The proposed technique verifies the user through the challenge-response method. Each file stored in the cloud is issued a unique secret value. The secret value is derived from the Token, the file's Tag, and the user's ID and password. Tokens and tags facilitate data identification, while user IDs and passwords authenticate authorized users. This confidential value is saved alongside further data details. When a user attempts to access the data, they provide their credentials and specify the file to be accessed.

ESPoW functions as a proof verifier. It derives the secret value solely from the user-submitted credential, along with the token and tag stored in the database. If the secret values from the database match the freshly generated secret values, the user grants permission to access the data. Alternatively, the user denies access to the data. The process steps of the proposed ESPoW proof verifier are as follows:

- Step 1: Users authenticate using their credentials (Username and Password).
- Step 2: Solicit a file for access.
- Step 3: Retrieve the user record from the database and obtain the SVAL.
- Step 4: Produce the SVAL utilizing the information provided by the user and the data contained in the proof-verifier table.
- Step 5: The SVAL from the archived proof-verifier table is juxtaposed with the newly created value. If the values are identical, users are permitted access; otherwise, access is denied.
- Step 6: Upon successful verification, the encrypted data is transmitted to the user.
- Step 7: Obtain the CEK (Convergent Encryption Key) from KYaaS (Key as a Service).
- Step 8: Obtain Convergent Encryption for the purpose of decryption.
- Step 9: Utilize the CEK to decrypt the data and retrieve the original information.

Authentication Procedure of ESPoW

User authentication occurs in two phases: Sign-Up and Sign-In. To utilize cloud storage efficiently every user must register with the proposed system to utilize cloud storage efficiently.

The steps involved in the registration phase are as follows.

Sign-Up Phase

- Users must register with DedupFrame using their credentials.
- The users select a user ID and password.
- Provide more pertinent information.
- If all details are formatted correctly, users are successfully registered with the DedupFrame.

Pseudo Code for Sign-Up

```
sub users_signup()
    User_Input:- UID, PWD, Age, Sex, DOB, E_Mail, Mobile_No, other_details.
    Start
    USR_CRE⇓ User_Input
    // Validate the USR_CRE for Format correction
    If no format correction in USR_CRE
    Registration Successful
    else
    msgbox("users are asked to check the format of data")
    end
```

Upon successful registration, the user may log into the system at any time. The steps in the login process are as follows.

Sign-In Phase

- Users submit their user Id (UID) and password (PWD).
- UID and PWD are validated with the KTaaS.
- If the validation is successful, then users are allowed to use the system; otherwise, they are asked to enter the correct UID and PWD.

Pseudo Code for Login

```
sub user_signin()
    User_Input:- UID, PWD
    Declaration
    UID⇓User Id submitted by the user during login
    PWD⇓Password submitted by the user during login
    SUID⇓User Id from KTaaS for the corresponding user
    SPWD⇓Password from KTaaS for the corresponding user
    start
    // valid the login
    if (UID==SUID) && (PWD==SPWD)
    users are switched to next level authentication
    else
    msgbox("enter correct UID and PWD") end
```

Users employ the recommended ways to upload and download their files at any time. Upon logging into the system, users upload files; but, to download them, they must re-submit their credentials to authenticate ownership of the contents. The suggested system checks the file in cloud storage and determines whether the requested file is related with the current user's credentials. The verification is conducted utilizing the confidential value in the proof-verifier table.

The secret value is produced with any cryptographic hashing algorithm. Within the proposed framework, it is created twice. The initial instance occurs when a file is uploaded to cloud storage, and the second instance occurs when a file is requested for download. The identical crypt is utilized for the generation of the secret value on both occasions. Unauthorized users appear to be unable to access the file.

Results and Discussion

The ESPoW methodology was evaluated in a cloud simulation environment with Microsoft Azure, specifically on a Windows Server 2008 micro instance with 30GB of storage and 1GB of RAM. The cloud-based application, created in C#.NET using Visual Studio 2012, was deployed as a service on Azure. This configuration facilitated the incorporation of critical capabilities, like convergent encryption, token management, and proof of ownership, which are vital for the successful execution of the ESPoW framework. To assess the efficacy of ESPoW, we quantified the duration needed to download data of differing quantities from the cloud, juxtaposing it with established deduplication techniques such as (Dave, J. *et al*. 2023) and (Song, M. *et al*. 2024). Table 2 demonstrates that ESPoW consistently attained reduced processing times for all evaluated file sizes.

For example, ESPoW required 112 milliseconds to process a 5MB file, but (Dave, J., *et al*. 2023) took 154 milliseconds, and (Song, M., *et al*. 2024) took 129 milliseconds. This trend persisted as the file size augmented, illustrating that ESPoW scales effectively with data volume.

The findings indicate that ESPoW far surpasses conventional deduplication techniques regarding computational efficiency. The decreased processing speeds are mostly attributable to ESPoW's optimized challenge-response mechanism, which reduces duplicate data management and expedites verification procedures. This enhancement is especially beneficial in cloud systems where rapid data retrieval and minimal latency are essential. Moreover, ESPoW's capacity to manage various file sizes efficiently indicates its appropriateness for diverse and resource-constrained cloud environments, hence augmenting its value as a safe and effective storage solution.

Table 2: Computational time caused by the proposed and existing techniques

Size	*(Dave, J., et al. 2023)*	*(Song, M., et al. 2024)*	*ESPoW*
5MB	154	129	113
10MB	299	245	229
15MB	402	357	343
20MB	511	460	442
25MB	633	574	559

In addition to computational performance, ESPoW enhances data security using a specialized proof-verification mechanism that utilizes a unique secret value (SVAL) provided to each file. This approach guarantees that only authorized users can access the data, even if the hashed value is breached. Our simulation validated that ESPoW efficiently protects against unauthorized access and potential data breaches, mitigating prevalent security issues related to deduplication in cloud storage settings. ESPoW enhances storage efficiency by integrating secure proof-of-ownership with data deduplication, while maintaining stringent requirements of data security and integrity.

Conclusion

The ESPoW method offers a thorough resolution to the issues of storage efficiency and data security in cloud environments. ESPoW utilizes a distinctive challenge-response technique and produces a unique secret value for each data file, so guaranteeing that only authorized users may authenticate their ownership and access their data. This method markedly improves security and diminishes computational requirements relative to current deduplication strategies. The findings of our research demonstrate that ESPoW provides enhanced performance regarding processing speed and data security, rendering it a feasible choice for secure cloud storage. Future studies may focus on enhancing the ESPoW framework to accommodate intricate data access patterns and augment its flexibility for various cloud infrastructures, hence promoting more safe and effective data management practices in cloud computing.

Acknowledgment

We sincerely acknowledge the department head, Dr L. Nagarajan, and Dr N. Sumathi, the institution's principal, for providing the facility to complete this paper Successfully.

References

Al-Amer, A., & Ouda, O. (2021). Secure and efficient proof of ownership scheme for client-side deduplication in cloud environments. International Journal of Advanced Computer Science and Applications, 12(12), 916-923. https://doi.org/10.14569/IJACSA.2021.0121290

Dave, J., Bhatt, M., & Pancholi, D. (2023). Secure proof of ownership for a deduplicated cloud storage system. International Journal of Information and Computer Security, 21(1/2), 205-228. https://doi.org/10.1504/IJICS.2023.131097

Dave, J., Dutta, A., Faruki, P., Laxmi, V., & Gaur, M. S. (2020). Secure proof of ownership using Merkle tree for deduplicated storage. Automation and Remote Control, 54(4), 358–370. https://doi.org/10.3103/S0146411620040033

Dave, J., Faruki, P., Laxmi, V., Bezawada, B., & Gaur, M. (2017). Secure and efficient proof of ownership for deduplicated cloud storage. In *Proceedings of the 10th International Conference on Security of Information and Networks* (pp. 63-69). Association for Computing Machinery. https://doi.org/10.1145/3136825.3136889

Ding, W., Wang, P., & Choo, K.-K. R. (2017). Secure data deduplication with dynamic ownership management in cloud storage. IEEE Transactions on Information Forensics and Security, 13(6), 1602-1612. https://doi.org/10.1109/TIFS.2017.2786724

Edavalath, S., & Sundaram, M. S. (2023). Cost-based resource allocation method for efficient allocation of resources in a heterogeneous cloud environment. The Scientific Temper, 14(04), 1339–1344. https://doi.org/10.58414/SCIENTIFICTEMPER.2023.14.4.41

Jiang, S., Jiang, T., & Wang, L. (2020). Secure and efficient cloud data deduplication with ownership management. IEEE Transactions on Services Computing, 13(6), 1152-1165. https://doi.org/10.1109/TSC.2017.2771280

Lee, M., & Seo, M. (2023). Secure and efficient deduplication for cloud storage with dynamic ownership management. Applied Sciences, 13(24), 13270. https://doi.org/10.3390/app132413270

Mgwalima, T., & Ndlovu, A. (2022). A review of the past, present, and future of secure data deduplication. International Journal of Engineering Research & Technology (IJERT), 11(12), 67-71. https://doi.org/10.17577/IJERTV11IS120020

Parveez, M., & Srinidhi, H. R. (2018). Secure data deduplication with dynamic ownership management in cloud storage. International Journal of Advanced Research in Innovative Ideas in Education (IJARIIE), 4(3), 1078-1080.

Sai Kiran, P., Reddy, J. B., & Sainadh, K. G. (2017). Data deduplication in cloud storage with proof of ownership. *International Journal of Pure and Applied Mathematics, 116*(5), 141-145.

Selvaraj, R., & Sundaram, M. S. (2023). ECM: Enhanced confidentiality method to ensure the secure migration of data in VM to cloud environment. The Scientific Temper, 14(03), 902–908. https://doi.org/10.58414/SCIENTIFICTEMPER.2023.14.3.53

Song, M., Hua, Z., Zheng, Y., Huang, H., & Jia, X. (2024). LSDedup: Layered Secure Deduplication for Cloud Storage. IEEE Transactions on Computers, 73(2), 422-435. https://doi.org/10.1109/TC.2023.3331953

Song, M., Hua, Z., Zheng, Y., Xiang, T., & Jia, X. (2023). Enabling Transparent Deduplication and Auditing for Encrypted Data in Cloud. IEEE Transactions on Dependable and Secure Computing, 1-18. https://doi.org/10.1109/TDSC.2023.3334475

Tian, G., Ma, H., Xie, Y., & Liu, Z. (2020). Randomized deduplication with ownership management and data sharing in cloud storage. Journal of Information Security and Applications, 51, 102432. https://doi.org/10.1016/j.jisa.2019.102432