**International Journal of Computer Science & Communications**
Vol. 11, No. 1, 2026, pp. 1-14
http://ijcsc.ielas.org/index.php/ijcsc
ISSN: 2545-4153

# BLOCKCHAIN-BASED PRIVACY-PRESERVING IMAGE DEDUPLICATION AND DECENTRALIZED INTEGRITY AUDITING IN CLOUD STORAGE

**Tanneeru Akhiladatta[1], KP Supreethi[2]**

Department of Computer Science and Engineering
University College of Engineering, Science and Technology
Jawaharlal Nehru Technological University Hyderabad
Kukatpally, Hyderabad, Telangana – 500085
Email: akhiladatta1411@gmail.com[1], supreethi.pujari@jntuh.ac.in[2]

Received: 2026-08-26 Accepted: 2026-09-12 Published online: 2026-09-26

**Abstract**

The rapid growth of digital image collections has increased the storage, communication, and management requirements of cloud storage systems. Conventional data deduplication techniques generally depend on cryptographic hashes, which are effective for detecting identical files but are unable to identify visually similar images having different binary representations. This paper presents a blockchain-based privacy-preserving image deduplication and integrity auditing framework that combines perceptual hashing, secure encryption, ownership verification, cloud storage, and blockchain-based integrity management. The proposed framework preprocesses uploaded images to obtain a standardized representation and generates a perceptual hash (pHash) to capture their visual characteristics. The generated pHash is compared with previously stored hashes using Hamming distance to classify an uploaded image as a new image, exact duplicate, or near-duplicate. For duplicate images, ownership verification is performed before updating the ownership record, thereby avoiding unnecessary physical storage of another copy. New images are encrypted before being stored in the cloud-side database, while associated metadata and ownership information are maintained securely. A SHA-256-based file identifier is generated for reliable file identification, and blockchain smart contracts are used to maintain integrity-related records. The implemented prototype uses Python, SQLite, an Ethereum-compatible blockchain environment, Ganache, Solidity smart contracts, and web3.py. Functional evaluation demonstrates that the framework successfully performs image-aware duplicate detection, encrypted storage, ownership verification, and blockchain-based integrity recording while reducing redundant image storage.

**Keywords:** Cloud Storage, Image Deduplication, Perceptual Hashing, pHash, Blockchain, Integrity Auditing, Ownership Verification, Data Security, Hamming Distance, Smart Contract.

## 1. INTRODUCTION

Cloud storage has become an important infrastructure for storing and accessing large volumes of digital information. The widespread use of smartphones, social media platforms, surveillance systems, medical imaging applications, and digital content-

sharing services has resulted in a continuous increase in image data. Although cloud storage provides scalable and convenient data management, storing multiple copies of identical or visually similar images increases storage consumption and communication overhead.

Data deduplication is commonly used to eliminate redundant data by storing a single copy of repeated content. Existing secure cloud deduplication schemes have combined deduplication with integrity auditing and blockchain mechanisms to improve storage efficiency and trust in outsourced storage environments. Zhang et al. (2025) proposed a blockchain-based privacy-preserving deduplication and integrity auditing scheme that addresses ownership privacy, key management, and decentralized auditing. Miao et al. (2024) similarly integrated blockchain with data deduplication and integrity auditing to reduce repeated authenticator and key storage. These approaches demonstrate the effectiveness of combining deduplication and blockchain-based auditing for cloud storage security.

However, conventional cryptographic hashing treats a file as a binary sequence. A small modification to an image, such as resizing, compression, brightness adjustment, or format conversion, can result in a completely different cryptographic hash even though the images remain visually similar. Consequently, exact file-level hashing is insufficient for image-specific deduplication where visually similar content should also be considered. Perceptual hashing provides an alternative approach by generating a compact representation based on visual characteristics. Perceptual image hashing has been widely investigated for image authentication, content identification, and similarity analysis. Qin et al. (2021) demonstrated the use of perceptual hashing for image content authentication, while Xiong et al. (2021) discussed perceptual robustness and discrimination properties of image hashing techniques.

This paper proposes an image-specific cloud deduplication framework that combines perceptual hashing with privacy-preserving storage and blockchain-based integrity auditing. The system generates a pHash for every uploaded image and compares it with stored perceptual hashes using Hamming distance. The framework distinguishes new, exact duplicate, and near-duplicate images. For duplicate content, ownership verification is performed before associating the requesting user with the existing stored image. For new images, encryption and secure storage are performed, followed by blockchain-based integrity recording.

The main contributions of this work are:

1. An image-specific deduplication mechanism based on perceptual hashing rather than only cryptographic file matching.
2. Hamming-distance-based classification of images into new, exact duplicate, and near-duplicate categories.
3. Ownership verification for duplicate images to prevent unauthorized claims over existing cloud content.
4. Encrypted cloud-side storage that avoids retaining multiple physical copies of duplicate images.
5. Blockchain-based recording and verification of integrity-related metadata through smart contracts.
6. An implemented prototype integrating image preprocessing, pHash generation, SHA-256 file identification, encryption, SQLite storage, ownership management, and Ethereum-compatible blockchain functionality.

## 2. RELATED WORK

Secure cloud deduplication has received significant attention because redundant storage increases both infrastructure requirements and operational cost. Xu et al. (2021) proposed a blockchain-enabled deduplicatable data auditing mechanism for network storage services. Their approach combines client-side deduplication with blockchain-assisted auditing and uses immutable blockchain records to improve the credibility of audit results.

Tian et al. (2022) proposed a blockchain-based secure deduplication and shared auditing scheme for decentralized storage. The framework combines secure deduplication with blockchain-based auditing to address storage redundancy and centralized auditing concerns. The work demonstrates that blockchain can provide a trustworthy mechanism for maintaining auditing information.

Zhang et al. (2023) presented an efficient integrity auditing mechanism with secure deduplication for blockchain storage. The work addresses the challenge of combining blockchain-based storage with efficient integrity verification and deduplication while reducing the storage burden associated with blockchain systems.

Li et al. (2023) proposed a blockchain-based transparent integrity auditing and encrypted deduplication framework for cloud storage. The scheme integrates encrypted deduplication with blockchain-based auditing and reduces dependence on conventional trusted third-party auditing mechanisms.

Song et al. (2023) further investigated blockchain-based deduplication and integrity auditing over encrypted cloud storage. Their approach considers ownership privacy and integrity verification together and demonstrates the feasibility of combining encrypted deduplication with blockchain mechanisms.

Miao et al. (2024) proposed blockchain-based shared data integrity auditing and deduplication using key and authenticator deduplication mechanisms. Their work demonstrates the reduction of repeated key and authenticator storage while maintaining decentralized auditing.

The recent work of Zhang et al. (2025) integrates privacy-preserving deduplication and blockchain-based integrity auditing and specifically addresses ownership privacy during deduplication and auditing. These studies establish a strong foundation for secure cloud deduplication and blockchain-based integrity verification.

Although these schemes provide effective solutions for general outsourced data, their primary deduplication decisions are generally based on cryptographic or file-level representations. Image data introduces an additional requirement because visually identical or near-identical images may possess different binary representations. Therefore, the proposed work focuses specifically on integrating perceptual image hashing with secure deduplication and blockchain-based integrity management.

## 3. PROPOSED SYSTEM

The proposed system provides a secure framework for image-specific deduplication in cloud storage. The system is designed around five major objectives: reducing redundant image storage, preserving confidentiality, verifying ownership, maintaining integrity, and supporting visually similar image detection.

The system begins when an authenticated user uploads an image. The image is validated and passed to the preprocessing module. The preprocessing stage standardizes the image representation for perceptual hashing while retaining the original image for secure storage. The system then generates a perceptual hash representing the visual characteristics of the image.

The generated pHash is compared against the perceptual hashes maintained in the cloud-side database. Hamming distance is used as the similarity measure. Based on the configured comparison threshold, the system identifies whether the uploaded image is a new image, an exact duplicate, or a near-duplicate.

If the image is identified as a duplicate, the system does not create another physical image copy. Instead, ownership verification is performed. After successful

verification, the ownership information associated with the existing image is updated. This approach allows multiple legitimate users to reference the same stored content while reducing storage redundancy.

If the uploaded image is classified as new, the system generates a unique file identifier using SHA-256 and encrypts the original image before storing it in the cloud-side database. The associated perceptual hash, file identifier, ownership information, and other required metadata are maintained for future operations.

After successful storage, integrity-related information is recorded through an Ethereum-compatible blockchain environment using a Solidity smart contract. During retrieval, the system verifies ownership and validates the corresponding blockchain record before decrypting and returning the requested image.

## 4. SYSTEM ARCHITECTURE

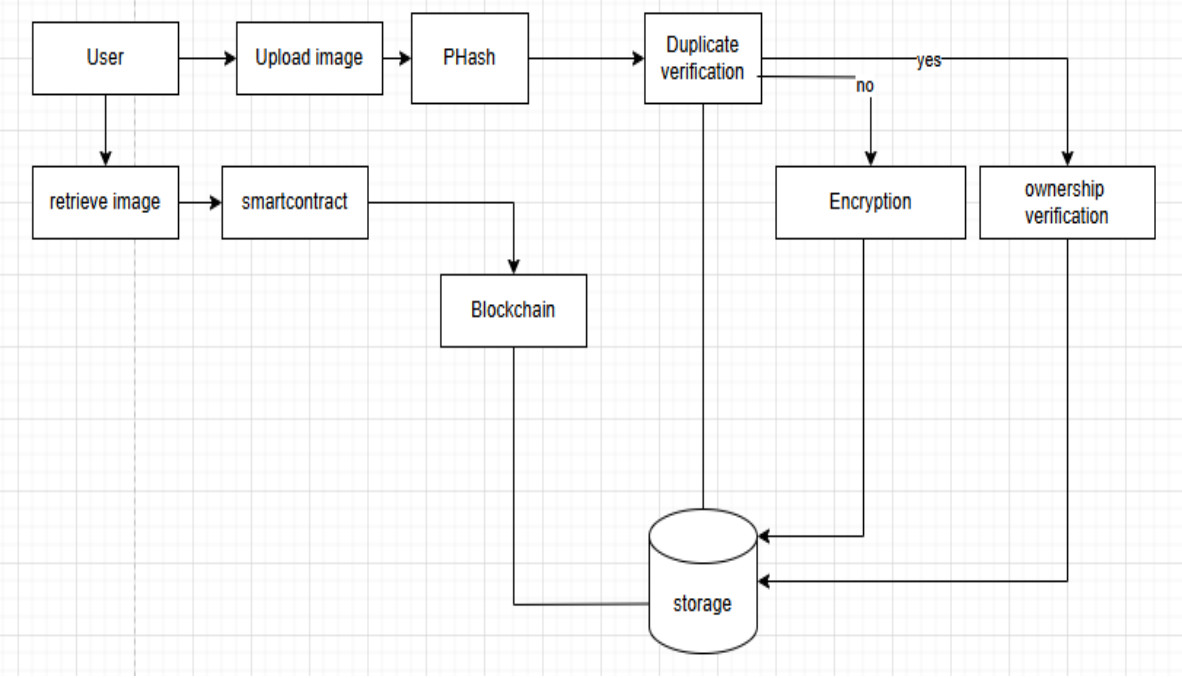


***Figure 1.*** *Proposed System Architecture for Blockchain-Based Privacy-Preserving Image Deduplication and Integrity Auditing*

The proposed architecture consists of the user/client interface, image-processing and deduplication components, cloud service provider, storage database, blockchain network, and smart contract. These components cooperate to provide secure image

upload, duplicate detection, encrypted storage, ownership management, retrieval, and integrity verification.

The User is the primary entity interacting with the system. The user can upload an image for storage or request retrieval of an already stored image. Authentication and ownership information are considered during operations that require access to stored content.

The Image Upload component receives the image from the user and validates the input before processing. The uploaded image is not immediately stored in plaintext. Instead, it is passed to the preprocessing stage so that the system can determine whether equivalent or visually similar content already exists.

The Image Preprocessing component standardizes the representation of the image before perceptual hashing. In the implemented prototype, the image is resized to a fixed representation and converted into a suitable color representation before pHash generation. The preprocessing operation is used for duplicate detection and does not replace the original image that is eventually stored.

The pHash Generation component generates a perceptual hash from the preprocessed image. Unlike a cryptographic file hash, a perceptual hash is designed to preserve similarity for images that remain visually similar after certain content-preserving modifications. This makes perceptual hashing suitable for image-specific duplicate detection.

The Duplicate Detection component compares the generated pHash with previously stored pHash values. Hamming distance is used to measure the difference between hash values. A distance of zero indicates an exact perceptual match, while a value within the configured threshold indicates a near duplicate. If no stored hash satisfies the similarity condition, the image is treated as a new image.

For a duplicate image, the request is transferred to the Ownership Verification component. The system verifies that the requesting user is authorized to associate the duplicate image with the existing stored content. When verification succeeds, the ownership record is updated rather than storing another physical copy of the image.

For a new image, the system invokes the Encryption component. The original image is encrypted before being stored in the cloud-side storage database. This prevents the cloud storage layer from directly exposing the original image content.

The Storage Database maintains the encrypted image and the metadata required by the application. The database stores information such as the file identifier, perceptual hash, encrypted image data, and ownership-related information. Storing only one physical copy for duplicate content provides the main storage-saving benefit of the proposed deduplication mechanism.

The Blockchain Network provides an additional integrity layer. Instead of storing the complete image on the blockchain, the system records integrity-related metadata through a smart contract. This avoids the high cost and impracticality of storing large image files directly on-chain.

The Smart Contract provides blockchain-side functions for recording and verifying integrity information. Once a new image has been successfully stored, its corresponding blockchain record is created. During retrieval, the blockchain record can be checked to determine whether the expected integrity information remains consistent.

The retrieval path operates in the reverse direction. When an authorized user requests an image, the system verifies ownership and checks the corresponding blockchain information. If verification succeeds, the encrypted image is retrieved from storage and decrypted before being returned to the authorized user.

This architecture therefore separates the responsibilities of image processing, duplicate detection, secure storage, ownership management, and blockchain-based integrity verification. The separation also allows the cloud database to handle large encrypted objects while the blockchain maintains only the information necessary for trusted integrity verification.

## 5. METHODOLOGY AND IMPLEMENTATION

### *5.1 Image Preprocessing*

The preprocessing stage prepares an uploaded image for perceptual hash generation. The prototype standardizes the image representation before calculating the pHash. The preprocessing operation improves the consistency of similarity comparison by reducing variations caused by different image dimensions and representations.

The original image is preserved separately because the preprocessing representation is intended for similarity analysis rather than permanent storage.

*5.2 Perceptual Hash Generation*

After preprocessing, a perceptual hash is generated for the image. Perceptual hashing represents visual characteristics in a compact hash form. Research on image hashing has demonstrated its usefulness for content authentication and similarity analysis (Qin et al., 2021; Xiong et al., 2021).

The generated pHash is stored as image metadata. When another image is uploaded, its pHash is compared with the existing values rather than comparing the complete image files.

*5.3 Hamming-Distance-Based Duplicate Detection*

Hamming distance measures the number of differing bit positions between two hash values. For two pHash values (H_1) and (H_2), the distance can be represented as: [D_H(H_1,H_2)=\sum_i(H_{1i}\oplus H_{2i})]

where (H_{1i}) and (H_{2i}) represent corresponding hash bits and (\oplus) denotes XOR.

The implementation uses the calculated distance to classify an uploaded image into three categories:

- **Exact_duplicate:** the uploaded image matches an existing perceptual hash exactly.
- **Near_duplicate:** the perceptual hash is within the configured Hamming-distance threshold.
- **New_image:** no existing perceptual hash satisfies the similarity condition.

This classification makes the deduplication mechanism image-aware rather than dependent only on exact binary equality.

*5.4 SHA-256 File Identification*

In addition to perceptual hashing, the system generates a SHA-256-based file identifier. The cryptographic identifier provides a deterministic representation of the uploaded file and is useful for unique file identification and integrity-related operations. The perceptual hash and cryptographic file identifier serve different purposes. The pHash is used for visual similarity and deduplication, whereas the SHA-256 identifier provides a cryptographic identity for the stored file.

*5.5 Ownership Verification*

Duplicate detection alone is insufficient because a user should not automatically gain access to an existing stored image simply because a similar image has been detected. Therefore, the system performs ownership verification for duplicate content.

The implemented ownership module generates an ownership challenge and proof and verifies the submitted proof. Once ownership verification succeeds, the user's ownership information can be associated with the existing stored image.

This prevents unnecessary duplication while preserving controlled access to previously stored content.

*5.6 Image Encryption and Storage*

New images are encrypted before being stored in the cloud-side storage layer. The implementation uses a symmetric encryption mechanism through the Python cryptographic library. The encrypted image is stored as binary data in the SQLite database.

The database also stores the metadata required for future duplicate detection, ownership verification, and retrieval. The separation of encrypted image content from blockchain metadata prevents large image files from being placed directly on the blockchain.

*5.7 Blockchain Integrity Recording*

Blockchain is used as an integrity and audit-support layer rather than as bulk image storage. The implementation uses an Ethereum-compatible blockchain environment with Ganache for development and testing and a Solidity smart contract for recording and verifying file-related metadata.

After a new image has been successfully stored, the relevant integrity information is submitted to the smart contract. Blockchain records provide an immutable reference that can be checked during future retrieval.

Previous blockchain-based auditing studies have demonstrated the usefulness of immutable ledgers for improving trust and transparency in cloud storage (Xu et al., 2021; Li et al., 2023; Zhang et al., 2025).

*5.8 Image Retrieval*

During retrieval, the system identifies the requested stored image and verifies the corresponding ownership information. The blockchain record is then checked before the encrypted image is returned.

If the integrity verification is successful, the encrypted image is decrypted and delivered to the authorized user. This retrieval process combines access verification, integrity checking, and confidential image recovery.

## 6. RESULTS AND DISCUSSION

The functional evaluation demonstrates that the proposed system can distinguish new images from existing content using perceptual similarity. When an image is uploaded for the first time, the system generates its perceptual hash, creates a file identifier, encrypts the image, stores the encrypted content, and records the corresponding integrity information.

When the same image is uploaded again, the system identifies an exact duplicate through pHash comparison. Instead of creating another encrypted image copy, the system proceeds to ownership verification. After successful verification, the ownership record is updated. This demonstrates the primary storage-saving mechanism of the proposed framework.

The near-duplicate test demonstrates the advantage of perceptual hashing over conventional cryptographic file matching. A visually similar image can produce a different SHA-256 value because its binary representation has changed, while its perceptual hash can remain sufficiently close for similarity-based detection. Therefore, the proposed approach is capable of identifying image relationships that cannot be identified using exact cryptographic equality alone.

The encryption test confirms that the cloud-side database stores encrypted image content rather than the original plaintext image. This provides confidentiality at the storage layer.

The blockchain test confirms that the implemented smart contract can record file-related integrity information on the Ethereum-compatible development blockchain. The recorded blockchain transaction can subsequently be checked during retrieval.

The combined results demonstrate that the proposed framework integrates four important security and efficiency properties: image-aware deduplication, confidentiality,

ownership verification, and integrity auditing. Existing blockchain-based deduplication studies have primarily concentrated on general data objects and cryptographic deduplication. The proposed system extends this direction toward image-specific deduplication by introducing perceptual similarity into the storage workflow.

## 7. SECURITY AND STORAGE-EFFICIENCY ANALYSIS

### *7.1 Storage Efficiency*

The major storage benefit is obtained when duplicate or near-duplicate images are detected. Instead of storing another physical copy for every repeated upload, the system maintains one encrypted image and updates the corresponding ownership information.

Therefore, if (N) upload requests correspond to the same stored image and only one physical copy is retained, the number of physical image copies is reduced from (N) to one. The actual storage saving depends on the number and size of duplicate images present in the workload.

### *7.2 Confidentiality*

The image is encrypted before being written to the cloud-side database. Consequently, the database does not need to store the original image in plaintext. The encryption layer therefore provides confidentiality for stored image content.

### *7.3 Ownership Protection*

Ownership verification prevents a duplicate-detection result from automatically granting access to an existing image. A user must successfully complete the ownership-verification process before the ownership information associated with the existing content is updated.

### *7.4 Integrity Protection*

Blockchain provides an immutable record for the integrity-related information associated with stored images. The image itself remains in the cloud database, while blockchain stores the information necessary for subsequent verification. This design avoids placing large binary objects directly on-chain.

### *7.5 Privacy Considerations*

The proposed framework separates image content from blockchain records. Only integrity-related metadata is recorded through the smart contract, while the actual image

remains encrypted in cloud-side storage. This reduces direct exposure of image content through the blockchain layer.

## 8. LIMITATIONS

The current implementation is a functional prototype and has several limitations. First, the effectiveness of near-duplicate detection depends on the selected Hamming-distance threshold. A threshold that is too small may fail to identify visually similar images, while a threshold that is too large may increase false similarity decisions.

Second, perceptual hashing is primarily effective for content-preserving modifications. Significant cropping, rotation, object replacement, or extensive editing may produce perceptual hashes that differ beyond the selected threshold. Therefore, the proposed mechanism should not be interpreted as a general semantic image-recognition system.

Third, the current prototype uses SQLite as the cloud-side storage simulation and Ganache as the blockchain development environment. A production deployment would require scalable cloud storage, distributed database infrastructure, secure key-management services, and a production blockchain network.

Finally, blockchain transactions introduce additional latency and transaction costs in real deployments. The current prototype demonstrates functional blockchain integration rather than a large-scale production performance benchmark.

## 9. FUTURE WORK

Future work can focus on improving the robustness of image similarity detection for transformations such as cropping, rotation, geometric distortion, and object-level modifications. Hybrid perceptual hashing techniques can be investigated to improve robustness while maintaining the computational efficiency required for cloud deduplication.

The storage layer can also be extended from SQLite to distributed cloud databases and object-storage systems. Key management can be strengthened through dedicated key-management services and stronger access-control mechanisms.

The blockchain component can be extended to support multiple cloud providers and cross-cloud integrity auditing. Smart contracts can also be enhanced to support controlled ownership revocation and more detailed audit trails.

Large-scale benchmarking using real-world image datasets would provide quantitative measurements for storage reduction, duplicate-detection accuracy, processing time, blockchain transaction overhead, and retrieval performance. Such evaluation would provide stronger evidence of scalability and practical deployment suitability.

## 10. CONCLUSION

This paper presented a blockchain-based privacy-preserving image deduplication and integrity auditing framework for cloud storage. The proposed system combines perceptual hashing, Hamming-distance-based similarity detection, SHA-256 file identification, encryption, ownership verification, cloud-side storage, and blockchain smart contracts within a unified workflow.

Unlike conventional cryptographic deduplication, the proposed approach considers the visual characteristics of images and can distinguish exact duplicates, near duplicates, and new images. Duplicate images do not require another physical copy, and successful ownership verification allows the existing ownership information to be updated. New images are encrypted before storage, while blockchain records provide an additional mechanism for integrity verification.

The implemented prototype demonstrates the feasibility of integrating image-specific deduplication with privacy and blockchain-based integrity management. The results show that perceptual hashing can complement cryptographic file identification by addressing the specific requirements of image data. The framework therefore provides a practical foundation for reducing redundant cloud image storage while maintaining confidentiality, ownership control, and integrity verification.

## REFERENCES

1. Q Zhang et al., “Blockchain-Based Privacy-Preserving Deduplication and Integrity Auditing in Cloud Storage”, IEEE TRANSACTIONS ON COMPUTERS, VOL. 74, NO. 5, pp. 1717–1729, May 2025.
2. Z Hua et al., “Blockchain-Based Shared Data Integrity Auditing and Deduplication”, IEEE TRANSACTIONS ON DEPENDABLE AND SECURE COMPUTING, VOL.21, NO.4, pp.3688-3703, JULY/AUGUST2024
3. Y XU et al., “A Blockchain-Enabled Deduplicatable Data Auditing Mechanism for Network Storage Services”, IEEE TRANSACTIONS EMERGING TOPICS IN COMPUTING, pp. 1–1432, SEPTEMBER 2021.
4. Q. Zhang et al., “Efficient integrity auditing mechanism with secure deduplication for blockchain storage,” IEEE Trans. Comput., vol. 72, no. 8, pp. 2365–2376, Aug. 2023.

5. J. Chang et al.,"Identity-based remote data checking with a designated verifier**,"** Journal of Cloud Computing, vol. 11, no. 1, pp. 1–14, 2022**.**
6. X. Zhang et al., "Conditional anonymous certificateless public auditing scheme supporting data dynamics for cloud storage systems**,"** IEEE Transactions on Network and Service Management*,* vol. 19, no. 4, pp. 5333–5347, 2022.
7. S. Li et al., "Blockchain-based transparent integrity auditing and encrypted deduplication for cloud storage," IEEE Transactions on Services Computing*, vol.* 16, no. 1, pp. 134–146, 2023.
8. Wang et al., "VeriDedup: A Verifiable Cloud Data Deduplication Scheme with Integrity and Duplication Proof," IEEE Transactions on Dependable and Secure Computing, vol. 20, no. 6, pp. 4695–4709, Nov./Dec. 2023.
9. Zhang, Y. et al., "Secure Password-Protected Encryption Key for Deduplicated Cloud Storage Systems," IEEE Transactions on Dependable and Secure Computing, vol. 19, no. 6, pp. 4012–4025, Nov./Dec. 2022.
10. Tian, Y. et al., "Blockchain-Based Secure Deduplication and Shared Auditing in Decentralized Storage," IEEE Transactions on Dependable and Secure Computing, vol. 19, no. 5, pp. 3128–3142, Sept./Oct. 2022.