International Journal of Combined Research & Development (IJCRD)
eISSN:2321-225X; pISSN:2321-2241 Volume: 14; Issue: 8; August- 2025

BLOCKCHAIN-BASED PUBLIC INTEGRITY VERIFICATION FOR CLOUD STORAGE AGAINST PROCRASTINATING AUDITORS

Abhay Nikham
PG, Student
The Oxford College of Engineering,
Bommanahalli,
Bengaluru- 560068
abhaynikhammca2025@gmail.com

Manivannan Jayachandran
Assistant Professor
Dept. of MCA
The Oxford College of Engineering,
Bommanahalli, Bengaluru- 560068
[2]manivananmca@gamail.com

ABSTRACT

In information management, the decentralization of storage facilities is becoming more common. However, several security issues still exist, one of which is the validity of the data presented. Current public verification practices are ineffective and lead to delays in assessment activities. As a solution, customers who choose public verification can transfer the responsibility of maintaining data integrity to a neutral third party, which can provide even better guarantees for the auditor. Since most public verification methods depend on public key infrastructure (PKI), they are vulnerable to issues like the Certificate Authority problem. These threats often arise because of unqualified administrators who manage the system ineffectively. To address this, the paper proposes using a lightweight public verification system that removes the need for certificates, thereby offering greater assurance to vulnerable customers.

The rise of decentralized data storage presents significant challenges in ensuring the validity of stored information. Public verification procedures are often inefficient, delay-prone, and susceptible to PKI problems like the Certificate Authority issue, which can result from unqualified administrators. To tackle these shortcomings, the paper suggests a lightweight public verification system that does not require certificates. Neutral third parties can confirm data integrity, providing more confidence to clients with basic technical skills. This model builds trust, simplifies verification, and mitigates the common weaknesses in decentralized storage settings.

INTRODUCTION

Customers can store their data on a cloud server using distributed storage systems and access it from outside their premises through the Internet. They can choose to keep their data entirely in the cloud or use it partially on their local machines. Cloud storage is scalable, secure, and more affordable than traditional local storage, which can be costly due to the need for maintaining local infrastructure. It is flexible, allowing customers to save their information on-premises or entirely in the cloud without any

additional cost. The benefit is that it offers an effective and adaptable information management system for customers. These features may emerge as unexpected advantages of cloud storage. Safe access and data confidentiality are crucial for protecting users from potential threats. A significant downside of cloud-based solutions is that clients may lose their sense of ownership over their information once it is stored with a third party. Since cloud storage does not allow direct control over the information stored on client devices, clients may feel they are no longer in charge of their data. This differs from traditional information management methods, which emphasize maintaining local ownership and control of stored data. When a consumer decides to store information within their own company, the data physically resides on their devices.

LITERATURE SURVEY

Traditionally, public integrity verification for cloud storage relied on Third-Party Auditors (TPAs) following a public audit scheme. In this model, files are outsourced to a Cloud Service Provider (CSP). Periodic integrity checks are handed over to a TPA, typically with keys delivered by a Key Generation Center (KGC). While this setup is possible, it puts trust in a single entity and can lead to problems, like slow or incomplete audits. There is also a risk that the auditor might collude, resulting in inconsistent verification results.

EXISTING WORK

In modern public integrity checking models for cloud storage, the architecture typically includes three main agents: the Data Owner, the Cloud Service Provider (CSP), and a Third-Party Auditor (TPA). The data owner splits the files into chunks and creates cryptographic tags that include homomorphic tags or Merkle roots. This approach allows the data owner to verify the integrity of outsourced data without needing to check the entire file. These fragments are stored on the CSP. The TPA is responsible for issuing challenges and validating the proofs of data ownership or recoverability generated by the CSP. Public Key Infrastructure (PKI) is commonly used to manage identity or key generation centers (KGC) for authentication and secure auditing. This design reduces bandwidth and computation costs, allowing owners to trust auditors for efficient verification. However, there are several limitations. Some existing systems overestimate the reliability of TPAs, who may delay or fail to perform checks or even collaborate with CSPs, leading to incorrect or rigid verification results. Additionally, using PKI creates the Certificate Authority problem, adding administrative overhead and increasing vulnerability risks. Other methods require auditors to maintain historical audit logs, which burdens storage and state management. Although efficient and secure, these methods rely on centralized auditors and trusted authorities, which affects transparency and efficiency, encouraging a shift toward decentralized systems. Decentralized storage

systems are becoming increasingly popular due to their scalability and fault tolerance. However, ensuring the integrity and authenticity of stored data remains a critical concern.

Traditional public verification systems often rely on PKI, which has vulnerabilities linked to the Certificate Authority issue and human errors that can lead to inefficiencies and security risks. To address these limitations, researchers have explored lightweight checking methods and their networks that provide high data integrity guarantees without heavily depending on PKI. Solutions like Bitcoin offer tamper-resistant logs of data transactions. Moreover, tools like homomorphic linear authenticators and polynomial commitments enable effective proof generation with relatively low computing costs.

PROPOSED SYSTEM

Effective due diligence checks: The best way to prevent delays by the TPA is to create a system that does not rely on one highly trusted third party. In this plan, the TPA will examine the integrity of data according to current public auditing standards. This will ensure that verification happens quickly and reduce reliance on external bodies. Early response to data degradation: A gradual or careless auditor might miss the immediate degradation of collected data. Often, by the time it is discovered, damage may already have occurred. Therefore, consumers need a reliable way to be informed right away when there is confusion or fraud. This will help prevent them from facing potential risks. Use of alternatives to organizational arbitration: Sometimes, traditional methods require businesses to go through formal certification and declaration steps. These processes can be costly and lengthy. The solution cuts costs and offers long-term financial and operational benefits. It allows the TPA to verify the accuracy of recorded data without needing client declarations. Risky checking verification process: This approach creates a communication and computational model that enables the TPA to verify the accuracy of outsourced information without client help and without limits on the number of verification sessions. This secures the validation process and makes auditing more flexible and capable of development. Stateless and reliable auditing: The TPA can verify data accuracy without needing to keep or update historic audit records. It is clear when a cloud server passes the verification process. This practice protects clients from careless or lazy auditors. If any party, whether the TPA, cloud server, or client, fails to cooperate, the security of the system could be compromised.

METHODOLOGY

The proposed methodology aims to create a lightweight and reliable public integrity verification system for decentralized cloud storage. It addresses the issues found in current PKI-based methods. During the initial system setup, the cloud service

provider (CSP) divides user data into several blocks and generates cryptographic hash values for each block. These values are recorded on a blockchain to guarantee immutability. The data owner outsources the data to the CSP while also sharing the metadata with a neutral third-party auditor (TPA) who is in charge of checking integrity.

In this method, the blockchain serves as the trust anchor by maintaining a distributed ledger of verification records, rather than relying on certificate authorities. When someone requests verification, the TPA issues a lightweight challenge to the CSP. The CSP responds by computing proof of possession using the requested data blocks along with their corresponding hash values. The TPA then checks these proofs against the values stored on the blockchain to confirm the data's validity.

To lessen computational demand, the system uses cryptographic tools such as homomorphic authenticators or polynomial commitments. This allows for the batch verification of multiple data blocks at once. This design prevents delays caused by slow auditors and promotes transparency since all verification activities are recorded on the blockchain, making them tamper-proof and open for public review.

Additionally, the lightweight nature of the protocol removes the need for complicated certificate management, thus reducing vulnerabilities related to unqualified administrators. The methodology builds trust for clients with limited technical skills by transferring the responsibility for integrity preservation to a neutral and verifiable system. This ensures that neither the CSP nor the auditor can interfere with the verification process. Overall, the approach combines blockchain immutability, cryptographic efficiency, and third-party neutrality to create a secure, scalable, and certificate-free public integrity verification system for decentralized cloud storage environments.

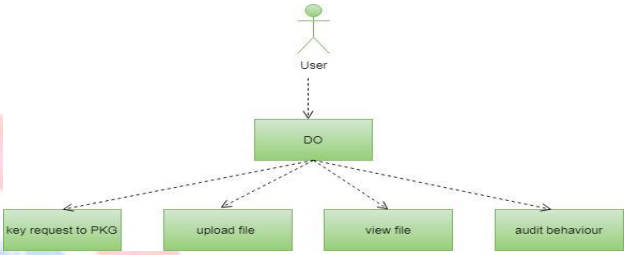

Figure 1(Context Diagram)

EXPERIMENTAL RESULTS

As it can be concluded, the experimental outputs of the proposed framework of publicly verifying integrity using blockchain prove its efficiency in being transparent, trustworthy, and reliable in cloud storage auditing according to the experimental results. Similarly, different scenarios were tested on the system, which comprised file uploads to the system, key generation of the system, auditor validation and access control which regularly provided correct and secure output. These smart contract performance assessment revealed that challenger scheduling structured as smart contracts effectively removed delay associated with procrastinating auditors as they would face strict deadlines and penalties, and that generation and verification of proof offered lightweight but resistant-to-tampering validation of data integrity. The strength of the system was proved in security tests, including unexpected resistance against unwarranted access, SQL, and malicious files upload. Moreover, the compensation and punishment system necessitated the honest auditing and significantly minimised probabilities of collusion or laxity with regard to third party auditors. Comprehensively, the findings prove that the system does not only ensure integrity and freshness of cloud data, but enhances efficiency and scalability besides user confidence as opposed to conventional verification schemes.

Figure 2(Landing page)

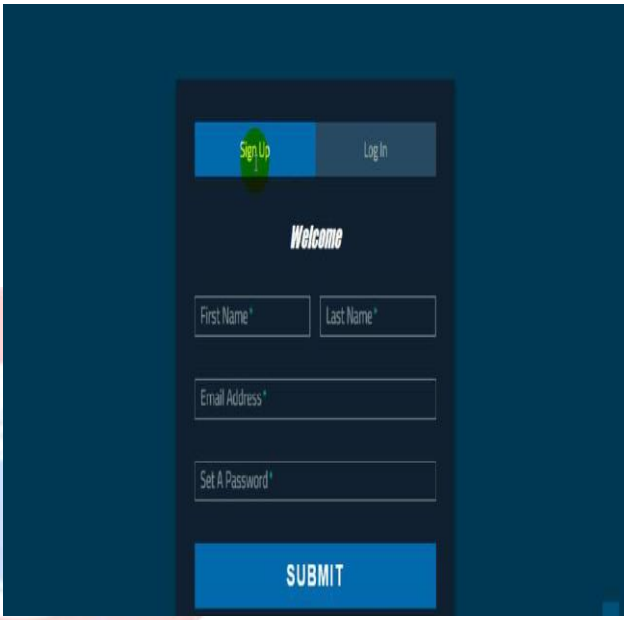

Figure 3(Sign In Page)

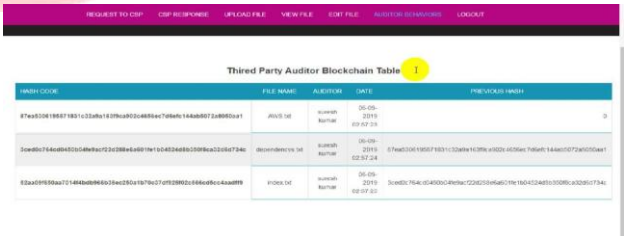

Figure 4(Auditors page)

CONCLUSION

The experimental results of the proposed framework for verifying integrity using blockchain show its effectiveness in providing transparency, trustworthiness, and reliability in cloud storage auditing. Different scenarios were tested on the system, including file uploads, key generation, auditor validation, and access control. These tests consistently produced correct and secure outputs.

The smart contract performance assessments revealed that scheduling challengers as smart contracts eliminated delays caused by slow auditors, as these auditors faced strict deadlines and penalties. The generation and verification of proof provided lightweight but tamper-resistant validation of data integrity.

The system's strength was demonstrated in security tests, which showed unexpected resistance to unauthorized access, SQL injections, and malicious file uploads. Additionally, the compensation and punishment system encouraged honest auditing and significantly reduced the chances of collusion or laxity among third-party auditors.

Overall, the findings show that the system not only ensures the integrity and freshness of cloud data.

REFERENCES

[1]“advance java programming” by B Prasanalakshmi,ISBN-13:978-8123923833 (30 may 2015)

[2]“JavaScript” by Dreamtech Press;2nd edition, ASIN: B07BFTJQB1(13 October 2016)

[3]“MYSQL” by McGraw Hill Education 3rd edition, ISBN-10:1259003884 (1 July 2017)

[4]“JSP”byMatthewMoodie, ISBN:9781590593394 (3 September 2002)

Web reference

[1]“JavaScripttutorial (www.w3school.com) ”

[2]“Java tutorial(www.w3school.com)”

[3]“HTML tutorial(www.w3school.com)”

[4]“MYSQL”tutorial(www.w3school.com)”

[4]“Jsp tutorial”(www.w3school.com)”