LIGHTWEIGHT AND SECURE AUDITING FOR SHARED CLOUD DATA

#1 SK. HIMAM BASHA, #2 P. PAVAN KUMAR
#1 ASSISTANT PROFESSOR
#2 MCA SCHOLAR
DEPARTMENT OF MASTER OF COMPUTER APPLICATIONS,
QIS COLLEGE OF ENGINEERING & TECHNOLOGY
VENGAMUKKAPALEM(V), ONGOLE, PRAKASAM DIST., ANDHRA PRADESH- 523272

ABSTRACT

Data integrity, a core security issue in reliable cloud storage, has received much attention. Data auditing protocols enable a verifier to efficiently check the integrity of the outsourced data without downloading the data. A key research challenge associated with existing designs of data auditing protocols is the complexity in key management. In this paper, we seek to address the complex key management challenge in cloud data integrity checking by introducing fuzzy identity-based auditing, the first in such an approach, to the best of our knowledge. More specifically, we present the primitive of fuzzy identity-based data auditing, where a user's identity can be viewed as a set of descriptive attributes. We formalize the system model and the security model for this new primitive. We then present a concrete construction of fuzzy identity-based auditing protocol by utilizing biometrics as the fuzzy identity. The new protocol offers the property of error-tolerance, namely, it binds with private key to one identity which can be used to verify the correctness of a response generated with another identity, if and only if both identities are sufficiently close. We prove the security of our protocol based on the computational Diffie-Hellman assumption and the discrete logarithm assumption in the selective-ID security model. Finally, we develop a prototype implementation of the protocol which demonstrates the practicality of the proposal.

I. INTRODUCTION

BIG data is eliciting attention from the academia as well as the industry. Over 2:5 quintillion bytes of data are reportedly created every day in the world, so much that 90% of the data has been created in the last two years alone. The explosive growth in the volume of data captured by the machines, sensors, IoT and other means, has changed our lifestyle gradually. According to a prediction by IDC (International Data Corporation), data set will grow 10-fold by the year of 2020 and there will be 5,200 GB of data for every person on earth Traditional storage model cannot meet the people's requirements due to the increasing large amount of data, which leads to the emergence of cloud storage. As a basic service of IaaS (Infrastructure as a service) model in cloud computing, cloud storage enables data owners to store their files to the cloud and deletes the local copy of the data, which dramatically reduces the burden of maintenance and management of the data. Cloud storage has a number of eye-catching

features, say global data access, independent geographical locations, on demand self service, resource elasticity and so on. Currently, both the individuals and big companies are enjoying the benefits due to cloud storage services. Despite the benefits offered by cloud storage, there are many inherent security risks. For example, when data owners outsource their data to the cloud, they generally lose physical possession of their data and may have no idea where their data are actually stored or who has the permission to getting access to their data. That is to say, it is the cloud servers who control the fate of the data after the data owners uploading their files to the cloud. While most cloud service providers are honest (e.g. due to their vested interest in ensuring a good reputation and avoiding civil litigations), data loss incidents are inevitable. This is not surprising. For example, a short-time crash of the cloud server or the breakdown of the storage medium (e.g RAM) will corrupt the data easily. Moreover, users' data may be lost due to deliberate deletion by cloud servers in order to make the available storage space for other files to get more profit. A survey reported that 43% of the respondents had lost their outsourced data and had to resort to recovering the data from backups. Data loss incident happens frequently in reality and has been regarded as one of the key security concerns in cloud storage. For example, Amazon's cloud crash disaster permanently destroyed many customers' data. It was reported that "the data loss was apparently small relative to the total data stored, but anyone who runs a website can immediately understand how terrifying a prospect any data loss is". As a consequence, data owners require a strong integrity guarantee of their outsourced data and they want to make sure that the cloud servers store their data correctly. Therefore, cloud data integrity is of particular importance in secure and reliable cloud storage.

II. RELATED WORKS

1. Lightweight Secure Cloud Auditing Scheme for Shared Data Supporting Identity Privacy and Traceability

Authors: Jun-Feng Tian, Xuan Jing, Rui-Fang Guo (SPNCE 2019)

Description: Proposes LSSA, an auditing scheme built on a virtual TPM pool, enabling lightweight computation for group members and supporting dynamic group operations like joins and leaves.

Merits:

- Efficient computation for low-capability users
- Ensures identity privacy and traceability

Demerits:

- Relies on trusted TPM agents.

2. LDAP: Lightweight Deduplication and Auditing Protocol for Secure Data Storage

Authors: Author(s) of LDAP (2019)

Description: Combines symmetric encryption, deduplication, and integrity auditing using DHT structures; offers dynamic data operations with minimal overhead.

Merits:

- Reduces storage costs via deduplication

- Efficient auditing suited for dynamic dataset

Demerits:

- Mentioned in 2017–2019 context; may have limited evaluation on large-scale shared data scenarios.

3. Audita: A Blockchain-based Auditing Framework for Off-chain Storage

Authors: Francatiet al. (2019)

Description: Integrates blockchain to enforce transparent, tamper-evident auditing of off-chain storage. Employs fair challenge schemes and works across Quorum-based setups.

Merits:

- Provides immutable audit logs
- Incentivizes honest storage through challenge the mechanisms.

Demerits:

- Blockchain overhead (e.g., transaction latency)
- Scalability concerns for high-frequency audits.

4.Towards Privacy-assured and Lightweight On-chain Auditing of Decentralized Storage

Authors: Du et al. (2020)

Description: Combines homomorphic authenticators with polynomial commitments to create compact (288 B), constant-time proofs suitable for blockchain-based storage auditing on Ethereum .

Merits:

- Extremely lightweight on-chain footprint
- Maintains proof privacy and scalability

Demerits:

- Tailored to decentralized storage; may require adaptation for centralized cloud systems.

5. Secure Consistency Verification for Untrusted Cloud Storage by Public Blockchains

Authors: Li et al. (2019)

Description: Introduces ContractChecker, a blockchain-aided protocol that cross-verifies logs from cloud and clients to detect inconsistencies and tampering .

Merits:

- Detects misbehavior from both sides
- Reduces client overhead for log auditing

Demerits:

- Dependent on blockchain transaction reliability
- Vulnerable to network-level issues like forks on public ledgers.

III. SYSTEM ANALYSIS

EXISTING SYSTEM

A key research challenge associated with existing designs of data auditing protocols is the complexity in key management. In this paper, we seek to address the complex key management challenge in cloud data integrity checking by introducing fuzzy identity-based auditing, the first in such an approach, to the best of our knowledge.

PROPOSED SYSTEM

The proposed protocol revolutionizes key management in traditional remote data integrity checking protocols. We also presented the the system and security models for this primitive, and a concrete fuzzy identity based data integrity auditing

protocol using the biometric based identity as an input. We then demonstrated the security of the protocol in the selective-ID model. The prototype implementation of the protocol demonstrates the practicality of the proposal. Future work includes implementing and evaluating the proposed protocol in a real-world environment. Proposed the concept of remote data integrity checking (RDIC, is also known as data integrity auditing), which comprises three parties, namely: cloud server, data owner and third party auditor (TPA). A publicly verifiable RDIC protocol allows the TPA or anyone to check the integrity of the stored data on the cloud without the need to retrieve the entire dataset. The concept of proof of irretrievability (POR), as well as providing a construction based on short signature algorithm and proving its security in the random oracle model. A number of remote data integrity checking protocols have been proposed catering to different real world requirements, such as dynamic operation privacy-preserving and publicly auditing. The secret key to the user's identity, without the need for a digital certificate. Since then, a number of ID-based schemes (including remote data auditing protocols) have been proposed. Example, several ID-based remote data auditing protocols were proposed and in these protocols, identity information is an arbitrary text string. The latter comprises user's name, IP address and E-mail address, which allows a user to register for a private key corresponding to his identity from the private key generation center.

IV. IMPLEMENTATION

Modules:

- **Data Owner**
- **TPA Auditing.**
- **Server.**
- **User.**

MODULE DESCRIPTION

Data Owner:

The client wants to upload new files to the cloud, it needs to verify the validity of the encrypted secret key from the cloud and recover the real secret key. We show the time for these two processes happened in different time periods. They only happen in the time periods when the client needs to upload new files to the cloud. Furthermore, the work for verifying the correctness of the encrypted secret key can fully be done by the cloud

TPA Auditing:

TPA to check the integrity of the stored data on the cloud without the need to retrieve the entire dataset. A HVT aggregates response of the challenged blocks into a single value, which significantly reduces the communication costs between the server.TPA is the trusted entity designated to verify the cloud data's integrity on behalf of the cloud user upon request.TPA and cloud server run a challenge response protocol for data integrity auditing to determine if the stored data are intact. Homomorphism and allows the TPA to detect the corruption of the file F in cloud without heavy communication overhead.TPA samples on the blocks of the file M to generate a challenge chal and sends chal to the cloud

server. According to the challenge, the server generates proof resp by aggregating the challenged blocks and the corresponding authenticators in the Response algorithm. Finally, the TPA verifies the response resp to determine whether the file F is intact on the cloud.

Server:
That is to say, it is the cloud servers who control the fate of the data after the data owners uploading their files to the cloud. While most cloud service providers are honest (e.g. due to their vested interest in ensuring a good reputation and avoiding civil litigations), data loss incidents are inevitable. As a consequence, data owners require a strong integrity guarantee of their outsourced data and they want to make sure that the cloud servers store their data correctly. Therefore, cloud data integrity is of particular importance in secure and reliable cloud storage.

A HVT aggregates response of the challenged blocks into a single value, which significantly reduces the communication costs between the server and the TPA.

In the schemes discussed above, the data owner has a pair of public/private keys (pk and sk respectively), where sk is used to generate authenticators of blocks and pk is used to verify a proof generated by the cloud server.

Finally, both TPA and cloud server run a challenge response protocol for data integrity auditing to determine if the stored data are intact

Data Sharing:
The shared data are signed by a group of users. Therefore, disputes between the two parties are unavoidable to a certain degree. So an arbitrator for dispute settlement is indispensable for a fair auditing scheme. We extend the threat model in existing public schemes by differentiating between the auditor (TPAU) and the arbitrator (TPAR) and putting different trust assumptions on them. Because the TPAU is mainly a delegated party to check client's data integrity and the potential dispute may occur between the TPAU and the CSP, so the arbitrator should be an unbiased third party who is different to the TPAU.

As for the TPAR, we consider it honest-but-curious. It will behave honestly most of the time but it is also curious about the content of the auditing data, thus the privacy protection of the auditing data should be considered. Note that, while privacy protection is beyond the scope of this paper, our scheme can adopt the random mask technique proposed for privacy preservation of auditing data, or the ring signatures in to protect the identityprivacy of signers for data shared among a group of users.

Auditing:
Public auditing schemes mainly focus on the delegation of auditing tasks to a third party auditor (TPA) so that the overhead on clients can be offloaded as much as possible. However, such models have not seriously considered the fairness problem as they usually assume an honest owner against an untrusted CSP. Since the TPA acts on behalf of the owner, then to what extent could the

CSP trust the auditing result? What if the owner and TPA collude together against an honest CSP for a financial In this sense, such models reduce the practicality and applicability of auditing schemes.

Secret Key Update:

The key update workload is outsourced to the TPA. In contrast, the client has to update the secret key by itself in each time period in scheme. We compare the key update time on client side between the both schemes the key update time on the client is related to the depth of the node corresponding to the current time period.Outsource key updates for cloud storage auditing with key-exposure resilience.

Cloud storage auditing protocol with verifiable outsourcing of key updates. In this protocol, key updates are outsourced to the TPA and are transparent for the client. In addition, the TPA only sees the encrypted version of the client's secret key, while the client can further verify the validity of the encrypted secret keys when downloading them from the TPA. We give the formal security proof and the performance simulation of the proposed scheme.

User:

Identity can be viewed as a set of descriptive attributes. We formalize the system model and the security model for this new primitive. We then present a concrete construction of fuzzy identity-based auditing protocol by utilizing biometrics as the fuzzy identity. The new protocol offers the property of error-tolerance, namely, it binds with private key to one identity which can be used to verify the correctness of a response generated with another identity, if and only if both identities are sufficiently close.

User data may be lost due to deliberate deletion by cloud servers in order to make the available storage space for other files to get more profit. A survey reported that 43% of the respondents had lost their outsourced data and had to resort to recovering the data from backups. Data loss incident happens frequently in reality and has been regarded as one of the key security concerns in cloud storage.

A registration authority that validates the identity of users requesting information from the CA, a central directory, and a certificate management system.

The secret key to the user's identity, without the need for a digital certificate. Since then, a number of ID-based schemes (including remote data auditing protocols) have been proposed. The user's identity may not be truly unique if the identity information is not chosen properly (e.g. using a common name such as "John Smith"). Secondly, a user needs to "prove" to the private key generator centre that he is indeed entitled to a claimed identity, such as presenting a legal document supporting the claim.

In a cloud environment where multiple users collaborate, frequent key updates are essential to maintain confidentiality and prevent unauthorized access, especially when users are revoked or new members join the group. A well-designed key update process should minimize communication

costs while enabling efficient re-distribution of new keys to authorized users. By integrating cryptographic techniques, such as key rotation and re-keying schemes, secret key updates enhance forward and backward security, ensuring that past data cannot be accessed by revoked users and future data remains protected. This approach not only strengthens data privacy and integrity but also maintains the lightweight nature of auditing, making it suitable for practical deployment in large-scale cloud environments.

V. RESULTS AND DISCUSSION

Fig 1 Home Page

The image highlights the concept of **Fuzzy Identity-Based Data Integrity Auditing for Reliable Cloud Storage Systems,**emphasizing secure and efficient mechanisms for ensuring data trustworthiness in the cloud. It suggests that cloud storage services can be enhanced with features such as data integrity verification, fuzzy identity-based authentication, and threshold secret sharing. These techniques collectively provide lightweight auditing, improve reliability, and safeguard sensitive data against tampering or unauthorized access. The visual also underlines the scalability and ease of provisioning servers, making cloud solutions more adaptable and secure for users and organizations handling large-scale storage needs.

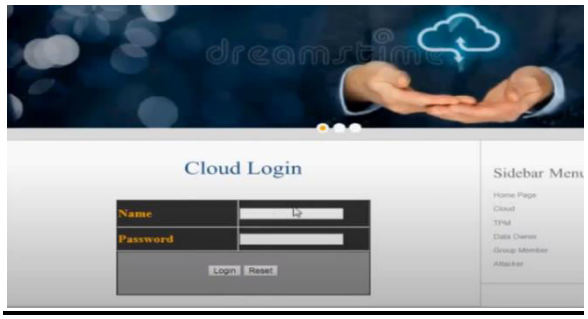


Fig 2 Login Page

The image above depicts a user interface for a **Cloud Login** portal, likely part of a web-based application for managing cloud storage or services. The login form requires a username and password, with the username "cloud" already filled in and a masked password being entered. The page features a clean and minimalistic design, with a simple menu on the right-hand side listing sections such as User, Cloud, Owner, and TPA (Third Party Auditor). At the top, there's a banner that highlights key concepts like "Cloud Storage," "Data Integrity," "Fuzzy Identity," and "Threshold Secret Sharing," suggesting that the platform is focused on secure and intelligent data management in the cloud. This kind of portal is typically used by users to authenticate themselves before accessing sensitive or private data stored on cloud servers. The interface supports basic functionalities such as logging in and resetting the form, indicating it's in the early or functional stage of a secure login system.

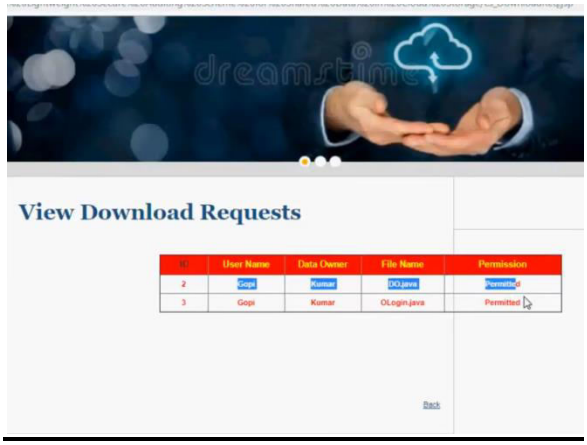


Fig 3
Download requests Page

The image presents a **File Details page** from a cloud storage auditing system. It displays key attributes of a stored file, including the file ID, file name, and multiple block or digital signatures that verify the integrity of the data. Each block is associated with a unique cryptographic signature, ensuring that any modification or tampering can be detected. The system also records the upload date and time, offering traceability for data management.

Additionally, there is an option to view the file details directly, along with a navigation link to return to the previous page.

This structured representation strengthens transparency and trust in cloud environments by providing secure and verifiable file auditing.

Fig 4
Data Integrity Page

The image illustrates a**Data Integrity Proof** interfacedesigned for verifying the authenticity of files stored in the cloud. Users are required to enter the file name and select the corresponding block to check, after which the system provides verification. This process ensures that the stored data has not been altered or tampered with, maintaining trust between the user and the cloud service provider. The verification mechanism strengthens data security by validating each block through cryptographic proofs, offering reliability and transparency. Additionally, a simple "Verify" button makes the system user-friendly while ensuring robust auditing of cloud data integrity.

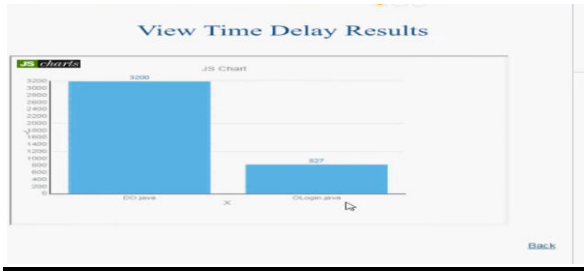


Fig 5
Time delay Page

The image shows the **View Time Delay Results** chart, representing performance analysis of two Java files. The bar graph compares execution delays, where DO.java shows a significantly higher delay of 3200 units, while OLogin.java demonstrates a much lower delay of 827 units. This visualization highlights the efficiency difference between the two processes, suggesting that OLogin.java performs faster and with less computational overhead. Such results are valuable in cloud auditing and integrity verification systems, as they help in evaluating the time efficiency of different modules, ensuring that lightweight operations are prioritized for better system performance.

Fig 6
Meta data Page

The image presents a **View File's Meta Data** section, which provides important details about stored files in the cloud system. The table lists file identifiers, file names, data owners, and corresponding private keys associated with each file. For example, files like DO.java and OLogin.java are owned by Kumar, while SDetails.java and FDeliver.java belong to Manjunath. Each file has a unique private key to ensure secure access and prevent unauthorized usage. This metadata management plays a vital role in maintaining data confidentiality, ownership verification, and secure file auditing within a cloud storage environment.

VI. FUTURE SCOPE AND CONCLUSION

Cloud storage services have become an increasingly important part of the information technology industry in recent years. Thus, ensuring the integrity of data outsourced to the cloud is of paramount importance. In this paper, we presented the first fuzzy identity-based data integrity auditing protocol. The proposed protocol revolutionizes key management in traditional remote data integrity checking protocols. We also presented the the system and security models for this primitive, and a concrete fuzzy identity based data integrity auditing protocol using the biometric based identity as an input. We then demonstrated the security of the protocol in the selective-ID model. The prototype implementation of the protocol demonstrates the practicality of the proposal. Future work includes implementing and evaluating the proposed protocol in a real-world environment.

REFERENCE

[1] M. Hogan, F. Liu, A. Sokol and J. Tong, "NIST, Cloud Computing Standards Roadmap," NIST Cloud Computing Standards Roadmap Working Group, SP 500-291-v1.0, NIST, Jul, 2011.

[2] M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. H. Katz, A. Konwinski, G. Lee,

D. A. Patterson, A. Rabkin, I. Stoica, and M. Zaharia, “Above the clouds: A berkeley view of cloud computing, University of California, Berkeley, Tech. Rep.

[3] Y. Deswarte, J. J. Quisquater and A. Saidane. “Remote integrity checking”. Integrity and Internal Control in Information Systems VI. Springer US, pp.1-11, 2004.

[4] G. Ateniese, R. C. Burns, R. Curtmola, J. Herring, L. Kissner, Z. N. J. Peterson and D. X. Song, “Provable data possession at untrusted stores,” in Proc. of ACM Conference on Computer and Communications Security, pp.598-609, 2007.

[5] G. Ateniese, S. Kamara and J. Katz. “Proofs of storage from homomorphic identification protocols”. Proc. of ASIACRYPT, pp.319-333, 2009.

[6] R. L. Rivest, A. Shamir and L. Adleman. “A method for obtaining digital signatures and public-key cryptosystems”. Communications of the ACM, 21(2), pp.120-126, 1978.

[7] H. Shacham and B.Waters, “Compact proofs of retrievability,” Proc. of Cryptology-ASIACRYPT, 5350, pp.90-107, 2008.

[8] D. Boneh , B. Lynn, and H. Shacham, “Short signatures from the weil pairing”, In Proc. of Asiacrypt 2001, pp.514-532, 2001.

[9] C. C. Erway, A. Kupcu and C. Papamanthou. “Dynamic provable data possession”. ACM Transactions on Information and System Security (TISSEC), 17(4), 15, 2015.

[10] Q. Wang, C. Wang, J. Li, K. Ren and W. Lou, “Enabling public verifiability and data dynamics for storage security in cloud computing”. Proc. of ESORICS2009, LNCS 5789, pp.355-370, 2009.

[11] Y. Yu, J.B. Ni, M. H. Au, H.Y. Liu, H.Wang and C.X. Xu. “Improved security of a dynamic remote data possession checking protocol for cloud storage”. Expert Syst. Appl. 41(17), pp.7789-7796, 2014.

[12] Y. Zhu, G. J. Ahn, H. Hu, S. S. Yau, H. G. An and C. J. Hu, “Dynamic audit services for outsourced storages in Clouds”. IEEE Trans. Services Computing, 6(2), pp. 227-238, 2013.

[13] Y. Yu, M.H. Au, Y. Mu, S.H. Tang, J. Ren, W. Susilo and L.J. Dong, “Enhanced privacy of a remote data integrity-checking protocol for secure cloud storage ”International Journal of Information Sececurity.14(4), pp.307-318, 2015.

[14] C.Wang, Q.Wang, K. Ren, andW. Lou, “Privacy-preserving public auditing for data storage security in cloud computing”. Proc. of IEEE INFOCOM, pp.525-533, 2010.

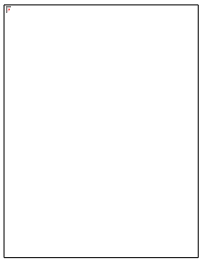
[15] C. Wang, S. S.Chow, Q. Wang, K. Ren, and W. Lou, “Privacypreserving public auditing for secure cloud storage”. IEEE Transactions on Computers, 62, pp.362-375, 2013.

[16] C. Wang, K. Ren, W. Lou, and J. Li, "Toward publicly auditable secure cloud data storage services", IEEE Network, 24, pp.19-24,2010.

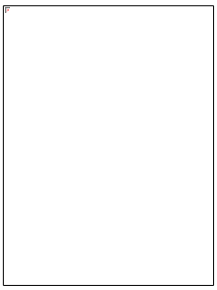
[17] Y. Yu, J.B. Ni, M. H. Au, Y. Mu, B.Y. Wang and H. Li. "Comments on a Public Auditing Mechanism for Shared Cloud Data Service". IEEE Transantions on Services Computing, 8(6),pp.998-999, 2015.

[18]A. Shamir. "Identity-based cryptosystems and signature schemes". Advances in cryptology. pp.47-53, 1985.

AUTHORS PROFILE



Mr. SK. HIMAM BASHA is an Assistant Professor in the Department of Master of Computer Applications at QIS College of Engineering &Technology, Ongole, Andhra Pradesh. He earned his Master of Computer Applications MCA from Anna University, Chennai. With a strong research background, He has authored and co- authored research papers published in reputed peer-reviewed journals. His research interests include ML, AI, Cloud Computing, and Programming Language He is committed to advancing research and fostering innovation while mentoring students to excel in both academic and professional pursuits.



Mr. P. PAVAN KUMAR is a postgraduate student pursuing a MCA in the Department of Computer Applications at QIS College of Engineering & Technology, ongole an Autonomous college in prakasam dist. He completed his undergraduate degree in BCA(Computers) from (Acharya Nagarjuna University).

His academic interests include Cloud Computing, Artificial Intelligence, Cyber Security, and Data Structures.