e-ISSN:2582-7219

INTERNATIONAL JOURNAL OF MULTIDISCIPLINARY RESEARCH IN SCIENCE, ENGINEERING AND TECHNOLOGY

Volume 7, Issue 7, July 2024

INTERNATIONAL
STANDARD
SERIAL
NUMBER
INDIA

Impact Factor: 7.521

| ISSN: 2582-7219 | www.ijmrset.com | Impact Factor: 7.521| Monthly, Peer Reviewed & Referred Journal|

| Volume 7, Issue 7, July 2024 |

| DOI:10.15680/IJMRSET.2024.0707041 |

Public Auditing Mechanism Protecting Privacy for Cloud Computing Data Storage Security

Dr.P.Jayasree, Mr.Dhusyanth

Associate Professor, Department of Computer Applications (UG), Hindusthan College of Arts &Science, Coimbatore, Tamil Nadu, India

Student, Department of Computer Applications (UG), Hindusthan College of Arts & Science, Coimbatore, Tamil Nadu, India

ABSTRACT: It is typical with cloud data services for data to be shared among several users in addition to being stored in the cloud. Unfortunately, due to human mistake and hardware/software faults, cloud data integrity is susceptible to doubt. In order to effectively audit cloud data integrity without having to get all of the data from the cloud server, a number of procedures have been developed for both data owners and public verifiers. Public verifiers will unavoidably learn sensitive information—identity privacy—through open auditing on the accuracy of shared data using these current methods. In this article, we put forth a brand-new privacy-preserving technique that facilitates open auditing of shared cloud data.

Specifically, we utilize ring signatures to calculate verification metadata required for verifying the accuracy of shared data. Our approach allows public verifiers to efficiently verify the integrity of shared data without having to retrieve the entire file, while maintaining the privacy of the signer's identity on each block. Furthermore, rather than confirming each auditing work one at a time, our technique may handle several auditing tasks at once. The outcomes of our experiments show how successful and efficient our approach is at ensuring the integrity of shared data.

KEYWORDS: Cloud Computing, Public Key, Servers, Privacy, Data Privacy, Information Sharing, Cloud Computing, Public Auditing, Privacy Preserving, Shared Data.

I. INTRODUCTION

Allotted computing is internet-primarily based completely registering, wherein shared assets, programming, and facts are given to computers and precise devices on hobby. It depicts each different complement, utilization, and conveyance version for IT administrations in moderate of the net. it's miles been imagined because of the reality the reducing aspect statistics innovation (IT) advent modeling for ventures, due to its big style of exquisite elements of hobby within the IT records: on-hobby self-control, pervasive tool get right of entry to, region free asset pooling, rapid asset flexibility, utilization primarily based actually valuing and transference of threat. As a elaborate innovation with enormous ramifications, Cloud Computing is changing the very way of the way corporations use data innovation. One essential part of this perfect version shifting is that statistics is being unified or outsourced to the Cloud. From customers' issue of view, along with each people and IT undertakings, setting away facts remotely to the cloud in an adaptable on-interest way brings attractive blessings: assist of the load for functionality management, widespread facts get proper of get right of entry to the with location autonomy, and evasion of capital consumption on gadget, programming, and college systems of assist and so forth.

On the identical time as Cloud Computing makes the ones opportunities more attractive than all over again in contemporary reminiscence, it moreover brings new and attempting out safety dangers closer to customers' outsourced records. The records trustworthiness of shared data in the cloud also can at gift be bargained. Outsider Auditor is slightly screen.

Which evaluations the data honesty for the sake of cloud management issuer without convalescing aggregate data? It stressful conditions the cloud server for the accuracy of data stockpiling at the same time as retaining no non-public data. To permit off the burden of manipulate of records of the information owner, TPA will compare the data of customer. It quench the contribution of the customer thru the use of reading that whether or not or not her information positioned away within the cloud are to ensure in region, which can be vital in venture economies of scale for Cloud Computing. At that factor it offers up the assessment record which might help owners to assess the risk of their subscribed cloud records administrations, and it will likewise be gainful to the cloud control provider to beautify their cloud primarily based actually administration degree. Along the ones traces, TPA will assist statistics owner and furthermore customers to verify that his data are sheltered inside the cloud and administration of records may be less troubling to data proprietor. Thusly, to empowering a safety safeguarding outsider Auditing convention, self-keeping to consumer renouncement, is the hassle we are going to deal with in this paper. Our survey is among unusual ones to enhance protection saving open reviewing in allocated computing, with an interest on consumer renouncement.

OBJECTIVE:
The propose a novel privacy-preserving mechanism that supports public auditing on shared data stored in the cloud.

II. RELATED WORK

- Ateniese et al. are the first to consider public auditability in their defined “provable data possession” (PDP) model for ensuring possession of data files on untrusted storages. Their scheme utilizes the RSA based homomorphic linear authenticators for auditing outsourced data and suggests randomly sampling a few blocks of the file. However, the public auditability in their scheme demands the linear combination of sampled blocks exposed to external auditor. When used directly, their protocol is not provably privacy preserving, and thus may leak user data information to the auditor.
- Juels et al. describe a “proof of retrievability” (PoR) model, where spot-checking and error-correcting codes are used to ensure both “possession” and “retrievability” of data files on remote archive service systems. However, the number of audit challenges a user can perform is fixed a priori, and public auditability is not supported in their main scheme. Although they describe a straightforward Merkle-tree construction for public PoRs, this approach only works with encrypted data.
- Dodis et al. give a study on different variants of PoR with private auditability. Shacham et al. design an improved PoR scheme built from BLS signatures with full proofs of security in the security model defined in.
- Similar to the construction, they use publicly verifiable homomorphic linear authenticators that are built from provably secure BLS signatures. Based on the elegant BLS construction, a compact and public verifiable scheme is obtained. Again, their approach does not support privacy-preserving auditing for the same reason. Shah et al. propose allowing a TPA to keep online storage honest by first encrypting the data then sending a number of pre-computed symmetric-keyed hashes over the encrypted data to the auditor.
- The auditor verifies both the integrity of the data file and the server’s possession of a previously committed decryption key. This scheme only works for encrypted files, and it suffers from the auditor statefulness and bounded usage, which may potentially bring in online burden to users when the keyed hashes are used up.
- In other related work, Ateniese et al. propose a partially dynamic version of the prior PDP scheme, using only symmetric key cryptography but with a bounded number of audits. In, Wang et al. consider a similar support for partial dynamic data storage in a distributed scenario with additional feature of data error localization.
- In a subsequent work, Wang et al. propose to combine BLS-based HLA with MHT to support both public auditability and full data dynamics.
- Almost simultaneously, Erway et al. developed a skip lists based scheme to enable provable data possession with full dynamics support. However, the verification in these two protocols requires the linear combination of sampled blocks just as, and thus does not support privacy-preserving auditing. While all the above schemes provide methods for efficient auditing and provable assurance on the correctness of remotely stored data, none of them meet all the requirements for privacy preserving public auditing in cloud computing.
- More importantly, none of these schemes consider batch auditing, which can greatly reduce the computation cost on the TPA when coping with a large number of audit delegations.

III. SYSTEM ANALYSIS

3.1 EXISTING SYSTEM

❖ Many mechanisms have been proposed to allow not only a data owner itself but also a public verifier to efficiently perform integrity checking without downloading the entire data from the cloud, which is referred to as public auditing . In these mechanisms, data is divided into many small blocks, where the owner independently signs each block; and a random combination of all the blocks instead of the whole data is to be retrieved during integrity checking. A public verifier could be a data user (e.g., researcher) who would like to utilize the owner’s data via the cloud or a third-party auditor (TPA) who can provide expert integrity checking services.

❖ Moving a step forward, Wang et al. designed an advanced auditing mechanism .so that during public auditing on cloud data, the content of private data belonging to a personal user is not disclosed to any public verifiers. Unfortunately, current public auditing solutions mentioned above only focus on personal data in the cloud .We believe that sharing data among multiple users is perhaps one of the most engaging features that motivates cloud storage. Therefore, it is also necessary to ensure the integrity of shared data in the cloud is correct.

❖ Existing public auditing mechanisms can actually be extended to verify shared data integrity. However, a new significant privacy issue introduced in the case of shared data with the use of existing mechanisms is the leakage of identity privacy to public verifiers.

3.1.1 DISADVANTAGES

✓ Failing to preserve identity privacy on shared data during public auditing will reveal significant confidential information to public verifiers.

✓ Protect this confidential information is essential and critical to preserve identity privacy from public verifiers during public auditing.

3.2 PROPOSED SYSTEM

❖ In this research, to solve the above privacy issue on shared data, we propose Oruta, a novel privacy-preserving public auditing mechanism.

❖ More specifically, we utilize ring signatures to construct homomorphic authenticators in Oruta, so that a public verifier is able to verify the integrity of shared data without retrieving the entire data while the identity of the signer on each block in shared data is kept private from the public verifier.

❖ In addition, we further extend our mechanism to support batch auditing, which can perform multiple auditing tasks simultaneously and improve the efficiency of verification for multiple auditing tasks.

❖ Meanwhile, Oruta is compatible with random masking, which has been utilized in WWRL and can preserve data privacy from public verifiers. Moreover, we also leverage index hash tables from a previous public auditing solution to support dynamic data. A high-level comparison among Oruta and existing mechanisms is presented.

❖ The propose system, a privacy-preserving public auditing mechanism for shared data in the cloud. We utilize ring signatures to construct homomorphism authenticators, so that a public verifier is able to audit shared data integrity without retrieving the entire data, yet it cannot distinguish who is the signer on each block.

❖ To improve the efficiency of verifying multiple auditing tasks, we further extend our mechanism to support batch auditing. There are two interesting problems we will continue to study for our future work. One of them is traceability, which means the ability for the group manager to reveal the identity of the signer based on verification metadata in some special situations.

3.2.1 ADVANTAGES

✓ A public verifier is able to correctly verify shared data integrity.
✓ A public verifier cannot distinguish the identity of the signer on each block in shared data during the process of auditing.
✓ The ring signatures generated for not only able to preserve identity privacy but also able to support blockless verifiability.
✓ The proposed system can perform multiple auditing tasks simultaneously
✓ They improve the efficiency of verification for multiple auditing tasks.
✓ High security provide for file sharing.

IV. ADVANCED ENCRYPTION STANDARD (AES):

The Advanced Encryption Standard (AES) is an encryption algorithm for securing sensitive but unclassified material by U.S. Government agencies and, as a likely consequence, may eventually become the defacto encryption standard for commercial transactions in the private sector. (Encryption for the US military and other classified communications is handled by separate, secret algorithms.)In January of 1997, a process was initiated by the National Institute of Standards and Technology (NIST), a unit of the U.S. Commerce Department, to find a more robust replacement for the Data Encryption Standard (DES) and to a lesser degree Triple DES. The specification called for a symmetric algorithm (same key for encryption and decryption) using block encryption (see block cipher) of 128 bits in size, supporting key sizes of 128, 192 and 256 bits, as a minimum. The algorithm was required to be royalty-free for use worldwide and offer security of a sufficient level to protect data for the next 20 to 30 years. It was to be easy to implement in hardware and software, as well as in restricted environments (for example, in a smart card) and offer good defended against various attack techniques. The entire selection process was fully open to public scrutiny and comment, it being decided that full visibility would ensure the best possible analysis of the designs. In 1998, the NIST selected 15 candidates for the AES, which were then subject to preliminary analysis by the world cryptographic community, including the National Security Agency. On the basis of this, in August 1999, NIST selected five algorithms for more extensive analysis. These were:

- ✓ MARS, submitted by a large team from IBM Research
- ✓ RC6, submitted by RSA Security
- ✓ Rijndael, submitted by two Belgian cryptographers, Joan Diemen and Vincent Rijmen
- ✓ Serpent, submitted by Ross Andersen, Eli Biham and Lars Knudsen
- ✓ Twofish, submitted by a large team of researchers including Counterpane's respected cryptographer, Bruce Schneier

Implementations of all of the above were tested extensively in ANSI C and Java languages for speed and reliability in such measures as encryption and decryption speeds, key and algorithm set-up time and resistance to various attacks, both in hardware- and software-centric systems. Once again, detailed analysis was provided by the global cryptographic community (including some teams trying to break their own submissions). The end result was that on October 2, 2000, NIST announced that Rijndael had been selected as the proposed standard. On December 6, 2001, the Secretary of Commerce officially approved Federal Information Processing Standard (FIPS) 197, which specifies that all sensitive, unclassified documents will use Rijndael as the Advanced Encryption Standard.Also see cryptography, data recovery agent (DRA) RELATED GLOSSARY TERMS: RSA algorithm (Rivest-Shamir-Adleman), data key, greynet (or graynet), spam cocktail (or anti-spam cocktail), fingerscanning (fingerprint scanning),munging, insider threat, authentication server, defense in depth, nonrepudiation

Explanations

AES is based on a design principle known as a Substitution permutation network. It is fast in both software and hardware. Unlike its predecessor, DES, AES does not use a Feistel network.AES has a fixed block size of 128 bits and a key size of 128, 192, or 256 bits, whereas Rijndael can be specified with block and key sizes in any multiple of 32 bits, with a minimum of 128 bits. The blocksize has a maximum of 256 bits, but the key size has no theoretical maximum.AES operates on a 4×4 column-major order matrix of bytes, termed the state (versions of Rijndael with a larger block size have additional columns in the state). Most AES calculations are done in a special field. The AES cipher is specified as a number of repetitions of transformation rounds that convert the input plaintext into the final output of ciphertext. Each round consists of several processing steps, including one that depends on the encryption key. A set of reverse rounds are applied to transform ciphertext back into the original plaintext using the same encryption key.

High-level description of the algorithm

- KeyExpansion—round keys are derived from the cipher key using Rijndael's key schedule
- Initial Round

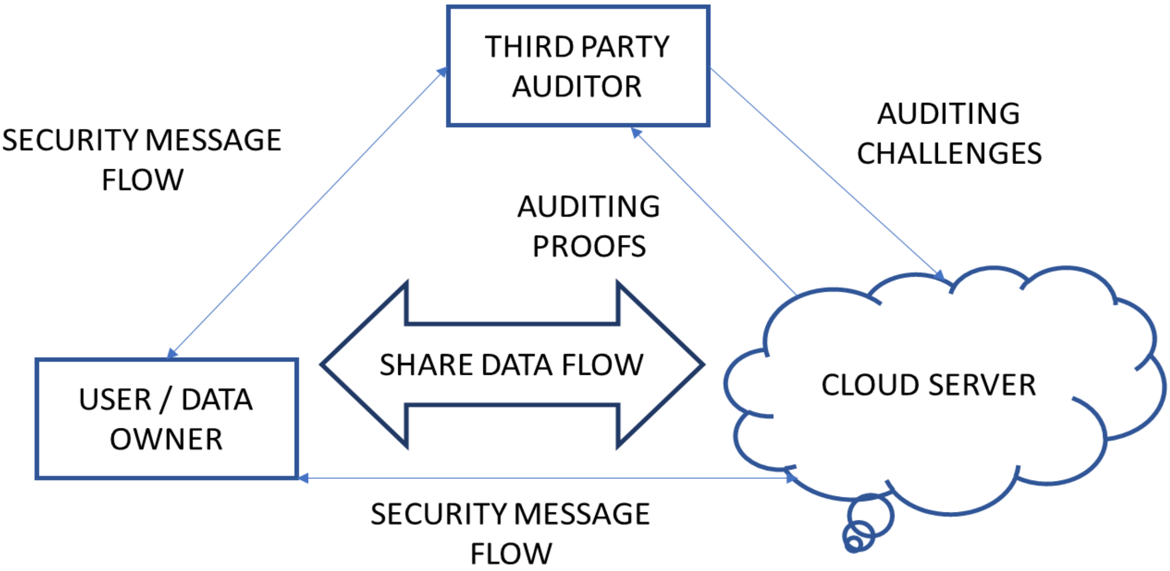
AddRoundKey—each byte of the state is combined with the round key using bitwise xor

- Rounds

a) SubBytes—a non-linear substitution step where each byte is replaced with another according to alookup table.
b) ShiftRows—a transposition step where each row of the state is shifted cyclically a certain number of steps.
c) MixColumns—a mixing operation which operates on the columns of the state, combining the four bytes in each column.
d) AddRoundKey

- Final Round (no MixColumns)

1. SubBytes
2. ShiftRows
3. AddRoundKey



V. CONCLUSION

In this research, a public auditing system for shared cloud data that protects privacy is proposed. We use ring signatures to build homomorphic authenticators, which allow a public verifier to audit shared data integrity without having to get all of the data, but they are unable to identify the signer on each block. We further enhance our approach to accommodate batch auditing, which will increase the efficiency of checking multiple auditing tasks.

They provide a public auditing method that protects privacy for cloud computing data storage security. We ensure that the cloud user is not burdened with time-consuming and potentially costly auditing tasks; instead, we reduce their concerns about their outsourced data leakage by using random masking and the homomorphic linear authenticator to ensure that the TPA does not obtain any knowledge about the data content stored on the cloud server during the efficient auditing process. Our privacy-preserving public auditing protocol is further extended into a multi-user setting, where the TPA can execute multiple auditing tasks in a batch manner for optimal efficiency, given that TPA may concurrently handle multiple audit sessions from different users for their outsourced data files. Detailed investigation demonstrates that our plans are provably secure and highly efficient.

VI. FUTURE WORK

There are two interesting problems we will continue to study for our future work. One of them is traceability, which means the ability for the group manager (i.e., the original user) to reveal the identity of the signer based on verification metadata in some special situations. Since Oruta is based on ring signatures, where the identity of the signer is unconditionally protected [21], the current design of ours does not support traceability. To the best of our knowledge, designing an efficient public auditing mechanism with the capabilities of preserving identity privacy and supporting traceability is still open. Another problem for our future work is how to prove data freshness (prove the cloud possesses the latest version of shared data) while still preserving identity privacy.

REFERENCES

1. Wang, B. Li, and H. Li, “Oruta: Privacy-Preserving Public Auditing for Shared Data in the Cloud,” Proc. IEEE Fifth Int’l Conf. Cloud Computing, pp. 295-302, 2012.
2. 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, “A View of Cloud Computing,” Comm. ACM, vol. 53, no. 4, pp. 50-58, Apr. 2010.
3. Ren, C. Wang, and Q. Wang, “Security Challenges for the Public Cloud,” IEEE Internet Computing, vol. 16, no. 1, pp. 69-73, 2012.
4. Song, E. Shi, I. Fischer, and U. Shankar, “Cloud Data Protection for the Masses,” Computer, vol. 45, no. 1, pp. 39-45, 2012.

5. Wang, Q. Wang, K. Ren, and W. Lou, “Privacy-Preserving Public Auditing for Data Storage Security in Cloud Computing,” Proc. IEEE INFOCOM, pp. 525-533, 2010.
6. Wang, M. Li, S.S. Chow, and H. Li, “Computing Encrypted Cloud Data Efficiently under Multiple Keys,” Proc. IEEE Conf. Comm. and Network Security (CNS ’13), pp. 90-99, 2013.
7. Rivest, A. Shamir, and L. Adleman, “A Method for Obtaining Digital Signatures and Public Key Cryptosystems,” Comm. ACM, vol. 21, no. 2, pp. 120-126, 1978.
8. The MD5 Message-Digest Algorithm (RFC1321). https://tools. ietf.org/html/rfc1321, 2014.
9. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z. Peterson, and D. Song, “Provable Data Possession at Untrusted Stores,” Proc. 14th ACM Conf. Computer and Comm. Security (CCS ’07), pp. 598-610, 2007.
10. Shacham and B. Waters, “Compact Proofs of Retrievability,” Proc. 14th Int’l Conf. Theory and Application of Cryptology and Information Security: Advances in Cryptology (ASIACRYPT ’08), pp. 90- 107, 2008.
11. Erway, A. Kupcu, C. Papamanthou, and R. Tamassia, “Dynamic Provable Data Possession,” Proc. 16th ACM Conf. Computer and Comm. Security (CCS’09), pp. 213-222, 2009.
12. Wang, C. Wang, J. Li, K. Ren, and W. Lou, “Enabling Public Verifiability and Data Dynamic for Storage Security in Cloud Computing,” Proc. 14th European Conf. Research in Computer Security (ESORICS’09), pp. 355-370, 2009.
13. Wang, Q. Wang, K. Ren, and W. Lou, “Ensuring Data Storage Security in Cloud Computing,” Proc. 17th Int’l Workshop Quality of Service (IWQoS’09), pp. 1-9, 2009.
14. Chen, R. Curtmola, G. Ateniese, and R. Burns, “Remote Data Checking for Network Coding-Based Distributed Storage Systems,” Proc. ACM Workshop Cloud Computing Security Workshop (CCSW’10), pp. 31-42, 2010.
15. Zhu, H. Wang, Z. Hu, G.-J. Ahn, H. Hu, and S.S Yau, “Dynamic Audit Services for Integrity Verification of Outsourced Storages in Clouds,” Proc. ACM Symp. Applied Computing (SAC’11), pp. 1550-1557, 2011.
16. Cao, S. Yu, Z. Yang, W. Lou, and Y.T. Hou, “LT Codes-Based Secure and Reliable Cloud Storage Service,” Proc. IEEE INFOCOM, 2012.
17. Wang, B. Li, and H. Li, “Certificateless Public Auditing for Data Integrity in the Cloud,” Proc. IEEE Conf. Comm. and Network Security (CNS’13), pp. 276-284, 2013.
18. Wang, S.S. Chow, Q. Wang, K. Ren, and W. Lou, “Privacy-Preserving Public Auditing for Secure Cloud Storage,” IEEE Trans. Computers, vol. 62, no. 2, pp. 362-375, Feb. 2013.
19. Wang, B. Li, and H. Li, “Public Auditing for Shared Data with Efficient User Revocation in the Cloud,” Proc. IEEE INFOCOM, pp. 2904-2912, 2013.
20. Wang, B. Li, and H. Li, “Panda: Public Auditing for Shared Data with Efficient User Revocation in the Cloud,” IEEE Trans. Services Computing, 20 Dec. 2013, DOI: 10.1109/TSC.2013.2295611.
21. Boneh, C. Gentry, B. Lynn, and H. Shacham, “Aggregate and Verifiably Encrypted Signatures from Bilinear Maps,” Proc. 22nd Int’l Conf. Theory and Applications of Cryptographic Techniques: Advances in Cryptology (EUROCRYPT’03), pp. 416-432, 2003.
22. Wang, H. Li, and M. Li, “Privacy-Preserving Public Auditing for Shared Cloud Data Supporting Group Dynamics,” Proc. IEEE Int’l Conf. Comm. (ICC’13), pp. 539-543, 2013.
23. Wang, S.S. Chow, M. Li, and H. Li, “Storing Shared Data on the Cloud via Security-Mediator,” Proc. IEEE 33rd Int’l Conf. Distributed Computing Systems (ICDCS’13), pp. 124-133, 2013.
24. Boneh, X. Boyen, and H. Shacham, “Short Group Signatures,” Proc. 24th Ann. Int’l Cryptology Conf. (CRYPTO’04), pp. 41-55, 2004.
25. Wang, B. Li, and H. Li, “Knox: Privacy-Preserving Auditing for Shared Data with Large Groups in the Cloud,” Proc. 10th Int’l Conf. Applied Cryptography and Network Security (ACNS’12), pp. 507-525, June 2012.
26. Brickell, J. Camenisch, and L. Chen, “Direct Anonymous Attestation,” Proc. 11th ACM Conf. Computer and Comm. Security (CCS’04), pp. 132-145, 2004.

INNO SPACE
SJIF Scientific Journal Impact Factor

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

निस्केयर
NISCAIR

INTERNATIONAL JOURNAL OF MULTIDISCIPLINARY RESEARCH IN SCIENCE, ENGINEERING AND TECHNOLOGY

| Mobile No: +91-6381907438 | Whatsapp: +91-6381907438 | ijmrset@gmail.com |

www.ijmrset.com