INTERNATIONAL JOURNAL OF ADVANCE RESEARCH, IDEAS AND INNOVATIONS IN TECHNOLOGY

ISSN: 2454-132X
Impact Factor: 6.078
(Volume 9, Issue 2 - V9I2-1168)
Available online at: https://www.ijariit.com

Remote data integrity checking with a designated verifier while preserving identity-based privacy

Kethireddy Anusha
anushaajay11153@gmail.com
Annamacharya Institute of Technology and Sciences, Rajampet, Andhra Pradesh

Hasthavaram Sai Leela
saileelareddy8@gmail.com
Annamacharya Institute of Technology and Sciences, Rajampet, Andhra Pradesh

Koona Naga Navya Sree
navyasree6814@gmail.com
Annamacharya Institute of Technology and Sciences, Rajampet, Andhra Pradesh

Mohammed Arshad Shaik
arshadshanu02@gmail.com
Annamacharya Institute of Technology and Sciences, Rajampet, Andhra Pradesh

Sanjapuri Naga Arudra Kumar
akasharudra@gmail.com
Annamacharya Institute of Technology and Sciences, Rajampet, Andhra Pradesh

Mukka Pramukha
mpramukhareddy@gmail.com
Annamacharya Institute of Technology and Sciences, Rajampet, Andhra Pradesh

ABSTRACT

The authenticity of outsourced data may be effectively checked using the remote data possession checking method, which can be classified into both private and public verification. Public verification can be verified by any cloud user, however, private verification can only be verified by the data owner. However, In the majority of real-world scenarios, the data owner anticipates that just a specific verifier will be able to carry out integrity-checking duties and that the verifier won't be able to learn anything about the data. Yan et al. suggest a remote data possession verification scheme with a designated validator to make sure that only the designated verifier can guarantee the data's integrity and that others cannot. However, this strategy makes use of public-key technology and disregards privacy-related concerns. To overcome these shortcomings, we propose an identity-based remote data possession checking method that fulfills the data owner's demand to select a unique verifier. This method also employs a Merkle hash tree for dynamic data updating and a randomized integer for blind data integrity verification that protects user privacy. Our method can be used to get around the public key infrastructure's burdensome certificate administration. We showed that our system is secure using the computing Diffie-Hellman assumptions and the discrete optimization assumption. Our plan is practical and effective in practical contexts, as shown by conceptual analysis and experimental results.

Keywords: *Identity-Based Cryptography, Designated Verifier, Privacy Protection, Data Dynamics*

1. INTRODUCTION

Wide-ranging uses of cloud computing have been found in the IT sector, social groups, and financial transactions. A Cloud often retains a lot of data over its lifetime in order to complete the targeted application and fulfil its features. Specifically, distributed and centralised techniques are the two main methodologies used for information storage and access in cloud networks. As opposed to centralised data storage, distributed data storage reduces single points of failure, uses less bandwidth, etc.

A concept called cloud computing offers enormous memory space and processing power at a reasonable price. It provides customers with the ability to access desired services regardless of the time or location across many platforms (such as mobile devices and personal computers), which is quite convenient for cloud users. Cloud storage services, like those offered by Amazon S3, Microsoft Azure, and Apple's iCloud, are only one of the many services that cloud computing offers. But it also faces a number of security risks, which are cloud users' main worries. To ensure that only users who are currently permitted can share the outsourced data, users wish to limit access to the data when outsourcing it to a cloud server. A data sharing system can offer backward secrecy and confidentiality. Forward secrecy can also be guaranteed by the procedure of re-encrypting all shared storage after decrypting it.

However, this poses fresh difficulties. In general, it is not a good idea to utilise the secret key to regularly update the ciphertext and should only be used for standard decryption. The process of download-decrypt-reencrypt-upload must be periodically performed by the data provider in in order to modify the shared data's ciphertext. For cloud users with limited computing and storage capacity, this procedure results in high communication and processing costs.

2. LITERATURE SURVEY

We call attention to the fact that a hostile cloud server can make the privacy-preserving adaptable trapdoor hash authentication tree technique useless in the work Cryptoanalysis of a Reliable Data Structure System With Public Privacy-Preserving Auditing. The cloud server can indeed pass third-party audits regardless of whether the outsourced data is modified randomly.
An adaptive, secure file format with confidentiality for big data is described in the study. Due to the rapid development of technologies such as 5 Gigabit Ethernet, Big Data, and the Internet of Things, Similar to a stream, data is often routinely and dynamically produced in a variety of settings. As a result, we described it as a "massive data stream," which is significant in many facets of daily life. But when this enormous data stream is on an insecure server, like a cloud service, and when it encounters difficulties raised by external audits, including latency sensitivity, unclear data size, and privacy leaks, how and where to verify it will become challenging. By incorporating trap hashes and BLS signatures into the Merkle hash tree, we propose the confidentiality adaptable trapdoor hash authenticating tree (P-ATHAT) as a special authenticate data structure to address these issues. The P-ATHAT approach can do live streaming data verification and can continuously extend its design as additional data is given. By addressing the particular point of failure issue with standard authentication trees, these qualities not only reduce the authentication path but also make the system more resilient. To address the privacy leak problem in third-party audits, we additionally provide a hash-based validation technique across tree topologies. After a comprehensive experimental evaluation and security analysis, the proposed approach i
s shown to be desirable for massive data streaming verification and confidentiality preservation in real-world applications.
The security of the protocol is discussed in the paper On the Privacy of an Identity-Based Multiple Data Auditing Standard for Big Data Storage. In this research, we draw attention to a security issue in the volatile memory internal audit method for huge data proposed by Shang et al. that is identity-based.

We keep emphasising that their policies are susceptible to a danger called private key exposure, in which the service (SP) might really retrieve the data owner's (DO) secret key from the stored data. Additionally, SP can provide evidence that can withstand an audit challenge even after all container and tag pairings have been eliminated. We intend to prevent such problems in future designs by recognising these design defects.

Integrity of data might be checked using an identity-based client-server research technique, which also offers a simple authentication scheme and verification to many users, as stated in the article Identity-Based Different Data Cleansing for Big Data Storage. Dynamic operations are not supported by earlier work on identity-based computer resource auditing, though. In these systems, the generation of tags is related to the indication of the information block, which itself is connected to update actions like change, insertion, and deletion of data. Users must change the names of all succeeding blocks if they execute the next data block. Users must download the full dataset, continually update it, and then upload the file type to a big data platform in order to upgrade in the future. We present a dynamic data internal audit scheme that uses identification that enables operations on data streams such as upgrade, insertion, and deletion in this study. There isn't any other identity-based data internal audit method that works with energetic operations that we are aware of. We are employing the cryptographic tree database structure for node tag authentication, which helps periodically update with integrity guarantee, in order to execute effective and scalable operations. The suggested technique is effective and secure, according to both performance and safety assessments.

The study describes Identity-Based Security Model Remote Integrity Of data Checking for Cloud Storage. Although storage services on the cloud are more affordable for consumers to maintain and manage vast volumes of data, they cannot actually guarantee the security of their personal information. It has been recommended to use a variety of remote data integrity verification (RDIC) techniques to verify data without download it. The great majority of contemporary schemes struggle with the challenging certificate management that comes with a public key system while ignoring the crucial matter of privacy protection. In order to solve these problems, a novel Identity-based RDIC technique is proposed in this study that makes use of homomorphic verifiable labels to minimise system complexity.

The original data is hidden in the proof via random integer addition, which prevents the verifier from knowing anything concerning the data all throughout integrity verification process. Our method is proven to be secure when a computational Diffie-Hellman issue is assumed. The experiment's results unmistakably demonstrate how useful and beneficial our strategy is in real-world applications. Identity-based remote data integrity checking with cloud storage with ideal data privacy preservation is described in the article. With remote data integrity checking, a computer storage device, such as a distant server, might demonstrate to a validator that it is really accurately preserving the data of a data owner (RDIC).

Several RDIC procedures have been documented in the literature to this point. The bulk of these models, however, have a problem with advanced key management, which suggests they depend on expensive public key infrastructure (PKI), making it challenging to deploy RDIC in real-world applications.

The proposed ID-based RDIC technique does not reveal any information about the data saved to the verification during the RDIC procedure in order to streamline the system and reduce the expense of developing and maintaining public connected to implementation support in PKI-based RDIC strategies. The novel model is demonstrated to be consistent against the malicious server and achieves zero knowledge confidentially against a verifier in the generic group model. The recommended solution has been subjected to rigorous security analysis, and implementation results show that it is both workable & safe in real-world applications.

The article Privacy-Preserving Cloud Auditing to Multi Users with Authentication and Traceability describes a scheme for preserving privacy in the cloud. With the wide - spread acceptance of online storage, users may enjoy an assortment of conveniences, including low-cost remote information storage and able to adapt data sharing. Even though cloud service providers (CSPs) really aren't completely trusted, numerous cloud internal audit systems have been developed to ensure the security and authenticity of shared data. However, current cloud auditing methods have some security flaws, such as user identity leaks, attacks involving denial of service, and single-manager abusing power. To address the aforementioned issues,We create a certificateless signature technology-based multi-user permission and auditability cloud internal audit system that protects user privacy.Because, unlike traditional methods, ours accomplishes user identity confidentiality without the use of signatures and ring signature procedures, the tag is small. Our system, on the other hand, allows at least d supervisors to collaborate together to determine the identity of an agressive user, preventing the abuse of a single owner's authority and making sure non-frame ability. We also incorporate a verification procedure here between CSP and the 3rd auditor to halt the denial-of-service attack (TPA). In other phrases, our strategy could solve the problem of anyone asking the CSP for proofs, preventing congestion issues and cloud resource waste. In terms of functionality, the proposed system enables efficient user ejection from a group. We can avoid the hassle of managing credentials and the issue of key escrow by using certificateless cryptography. In a certificate-free cryptographic algorithms setting, Our system is proven to be secure against two different environmental adversaries, according to security analysis.

Identity-Based Signature Scheme with RKA Security is described in the paper The use of cryptographic techniques such as encrypting public keys, digital certificates, and pseudorandom functions is an instance of a related-key attack (RKA). However, we notice that now the identity-based signature (IBS), a necessary building block for identity-based cryptology and one proposed by Shamir in 1984, doesn't really seem to take RKA security into account. In this study, we try to create a security model for RKA security and reveal it to IBS schemes for the first time. More specifically, we believe the RKA occurs in the users' ability to sign keys or even the KGC master key, resulting in two distinct RKA equities for IBS. We show the fact that the most impactful Schnorr-like IBS method developed by Galindo and Garcia is Identifying appropriate by launching a simple RKA.

A simple change, on the other hand, results in an RKA-secure IBS scheme, by which we include a comprehensive security reasoning using a spontaneous oracle. At last, the performance analysis shows that the revamped system still provides greater security while remaining highly efficient.

The article describes Remote Data Checking with a Designated Verifier for Cloud Storage. The remote data possession checking (RDPC) process quickly and easily verifies the consistency of files saved in cloud storage. Public verification makes it possible for anybody to assess the reliability of remote data, expanding the public's access to its potential uses in cloud storage. Private verification, which is often used to authenticate secret data, only permits the data owner to confirm the information.Although other persons are unable to visit the contents in data storage, the data owner frequently expect a specific user to do so in real-world applications. Which neither public nor private validation will satisfy this condition, it should be evident. However, the malicious cloud server has the ability to attack the DV-PDP with replay assaults. We suggest a new RDPC technique with the chosen verifier to overcome this problem, in which the data owner specifies a specified verification to confirm the accuracy of the data. Using the computationally Diffie-Hellman assertion, we demonstrate the security of our RDPC method in a random oracle scenario.Our technique delivers a high error detection probability while needing less communication, storage, or computing, as shown by theoretical research and lab findings.

According to the paper Ownership of Identity-Based Distributed Verifiable Data in Multicloud Disk space, remote data integrity testing is critical in cloud storage. It can force clients to verify that one‘s outsourced data has been retained without having to download the entire data set. In certain application scenarios, customers may be obligated to keep their information on multicloud servers.

To minimize the expense to the verifier, the way of maintaining methodology must be effective. From these two perspectives, we propose a distinct identity-based dispersed attribute based possession (ID-DPDP) paradigm in multicloud storage for remote server integrity verification. The formalised model and security mechanism are provided.We build a special ID-DPDP protocol on the basis of interpolation pairings. Under the computational Diffie-Hellman problem's hardness assumption, the proposed ID-DPDP protocol is demonstrably secure. Our ID-DPDP technique is effective and flexible in addition to being institutionally beneficial because this eliminates the requirement for certificate management. The proposed ID-DPDP protocol may conduct confidential, delegated, or public verification, I believe it relies on the client's authorization.

3. EXISTING SYSTEM

Depending on the kind of verification utilised, the majority of schemes can be divided into two categories: private verification & public verification. They fail when the owner of the data only allows a specific verifier to access the data. However, to address this issue, this approach must verify every data block for every challenge, which raises the computing cost.

It should be emphasised that the aforementioned plan usesa third-party validator to check the accuracy of the data. Numerous integrity checking approaches are used because of the issue that verifiers could reveal the data of the data owner. The aforementioned systems rely on PKI technology and call for labor-intensive certificate management procedures. presented a multi-cloud storage identity-based RDPC method. The user's public key serves as an individual identification in this system. first developed a security model for it and an identity-based signature scheme's related-key attack vulnerability.

Drawbacks:

- In cloud networks, achieving a privacy-preserving auditing is always difficult due to problems with authentication, hijacking, and user collusion.
- In order to target data, the conspiring users want appropriate information, such as a secret key.

•Users and the key authority must use a secure path who have not had their keys revoked to communicate fresh keys, however the current approach only provides selective security.

Proposed Work:

In this research, we have proposed an remote data possession verification method that can guarantee that only the designated verifier may examine the data integrity while others cannot. Yet, neither the privacy of personal data nor the public-key technology employed in this system are taken into consideration. We solve these issues by proposing the identity-based remote data possession verification method that satisfies the data owner's demand to select a unique verifier.In addition, our method uses a random number to carry out a blind data security proof that protects data privacy while allowing for dynamic data updating. Our method also allows us to circumvent the challenging certificate administration with in public key infrastructure.

Advantages:

- Computationally efficient.
- Cost-effective
- Enables the cloud storage preserves data integrity and confidentiality

Algorithm:

AES Algorithm

ALGORITHM AES

The most popular and widely utilised symmetrical encryption algorithm which is most probably to be used nowadays is the Advanced Encryption Method (AES). It can be detected at a minimum six times faster than triple DES.

A successor was needed since DES's key length was insufficient. It was believed that as processing power rose, it would be susceptible to a thorough key search attack. Triple DES was developed to solve this problem, however it was found to be slow.

The following are AES's features:

- Symmetric block cypher with symmetric keys • Data in 128 bits and keys in 128/192/256 bits
- Triple-DES is stronger and faster than this.

Full specification and design details should be provided. Software should be implementable in Java and C.

Operation:

AES employs iterative cyphers as opposed to a Feistel cypher. Its structural foundation is a "substitution-permutation network". It consists of a number of interconnected operations, some of which require bit relocation and some of which switch some output for inputs

(permutations).

It's interesting that AES uses bytes rather than bits for all of its calculations. AES consequently considers a plaintext block's 128 bits to be 16 bytes.

Those 16 bytes are set up into a matrix of four columns and four rows in order to be handled. A number of rounds of AES varies according to the key size, unlike DES, which stays the same. For 128-bit keys, AES employs ten rounds, for 192-bit keys, twelve rounds, and for 256-bit keys, twenty-four rounds.

Result and Analysis

4.CONCLUSION

This study suggests a verifier-designated identity-based remote data integrity checking scheme. This plan can also achieve data privacy protection and resolve the semi-trusted verifier problem. Data owner is unable to access data that has already been shared or data that is shared in the future. Additionally, a concrete IBE construction is shown. The proposed IBE system is demonstrated to be adaptive-secure in the decisional standard model.

5. REFERENCES

[1] M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica, and M. Zaharia, ‘‘A view of cloud computing,’’ Commun. ACM, vol. 53, no. 4, pp. 50–58, Apr. 2010.

[2] D. Zissis and D. Lekkas, ‘‘Addressing cloud computing security issues,’’ Future Gener. Comput. Syst., vol. 28, no. 3, pp. 583–592, Mar. 2012.

[3] J. Lu, F. Nan, Y. Huang, C.-C. Chang, Y. Du, and H. Tian, ‘‘Privacypreserving public auditing for secure data storage in fog-to-cloud computing,’’ J. Netw. Comput. Appl., vol. 127, pp. 59–69, Dec. 2018.

[4] Y. Deswarte, J.-J. Quisquater, and A. Saidane, ‘‘Remote integrity checking,’’ in Proc. Work. Conf. Integrity Internal Control Inf. Syst., Cham, Switzerland: Springer, 2003, pp. 1–11.

[5] G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z. Peterson, and D. Song, ‘‘Provable data possession at untrusted stores,’’ in Proc. 14th ACM Conf. Comput. Commun. Secur. (CCS), 2007, pp. 598–609.
[6] B. Wang, B. Li, and H. Li, ‘‘Panda: Public auditing for shared data with efficient user revocation in the cloud,’’ IEEE Trans. Serv. Comput., vol. 8, no. 1, pp. 92–106, Jan./Feb. 2015.
[7] Y. Feng, Y. Mu, G. Yang, and J. K. Liu, ‘‘A new public remote integrity checking scheme with user privacy,’’ in Proc. Australas. Conf. Inf. Secur. Privacy. Berlin, Germany, Springer, 2015, pp. 377–394.
[8] H. Yan, J. Li, and Y. Zhang, ‘‘Remote data checking with a designated verifier in cloud storage,’’ IEEE Syst. J., vol. 14, no. 2, pp. 1788–1797, Jun. 2020.
[9] A. Juels and B. S. Kaliski, ‘‘Pors: Proofs of retrievability for large files,’’ in Proc. 14th ACM Conf. Comput. Commun. Secur. (CCS), 2007, pp. 584–597.
[10] H. Shacham and B. Waters, ‘‘Compact proofs of retrievability,’’ in Proc. Int. Conf. Theory Appl. Cryptol. Inf. Secur., Berlin, Germany, Springer, 2008, pp. 90–107.

