International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 09 Issue: 04 | April - 2025 SJIF Rating: 8.586 ISSN: 2582-3930

Enhanced Data Integrity Auditing and Sharing Scheme with highly confidential data for Secure Cloud Storage.

Prof. Shinde Sadashiv P.[1]

[1]*Amrutvahini college of engineering Sangamner, Ahmednagar*

Abstract - *Cloud Computing provides a user -based data storage and calculation method. We can use cloud computing to maintain privacy and confidentiality of cloud data. We have to pay for use and requires an Internet connection. Due to the lack of data security cloud, it provides an effective way to save data in encrypted form in the cloud. The aim is to prevent the patient's abuse and strive to require data as the patient requires. Serious safe and protected concerns are too much of the problems that stand in the way of a wide acceptance of the framework. IT application plays an important role in health and care for patients. Cloud users upload personal or confidential data to the cloud data center. In previous electronic health record systems, they cannot handle dynamic changes related to a number of users. Our main motive is to protect data from unauthorized access. In the previous system file, file recording operations are not carried out safely and abused for data due to lack of security. The cloud file may contain some sensitive information, and sensitive information should not be exposed to others when the cloud file is shared and encryption of the entire shared file can make sensitive information.*

Keywords — Encryption, Cloud computing; Data integrity auditing; Data sharing; Sensitive information hiding, Confidentiality, Confidentiality.

1.INTRODUCTION

Sensitive information should not be exposed to others when sharing a cloud file. Encryption of the entire shared file can Be aware of sensitive information hiding, but will make this shared file able to be used by others. How to realize data sharing with Sensitive information hiding at the integrity audit of remote data has not yet been explored. To solve it Problem We suggest a scheme of the Audit of Integrity of Remote Data that realizes data sharing with sensitive information that hidden this paper. In this scheme, disinfectants are used to disinfect data blocks corresponding to sensitive information File and transformation of signatures of this data into valid for disinfected file. These signatures are used to verify Integrity of the sanitized ensemble in the Audit Integrity Audit. As a result, our scheme is stored in the cloud capable of to be shared and use others on condition that sensitive information is hidden while auditing integrity of remote data. He is still able to perform effectively. Meanwhile, the proposed scheme is based on a cryptography based on identity that simplifies Complicated certificate management. Security and performance assessment analysis show that the proposed scheme is safe and efficient.

Existence System-

In remote honesty information that checks the plans, the information owner must create brands from the bat to prevent information from information from information before transferring it to the cloud. These signs are used to prove that the cloud actually prevents this information from exploring credibility, and then the information owner transmits this information from their relationship marks to the cloud. The information deferred in the cloud is regularly shared above various clients in many distributed storage applications such as Google, Cloud. Sharing information as Standout among the most basic peaks in the distributed storage allows you to pass on to different clients to others. In any case, this mutual information may contain some delicate data. For example, electronic health records (EHRS) postpone what more, participate in the cloud largely contain audited data of the company (company name, telephone number and identification number, etc.)

Literature survey-

Demonstrable holding of data (PDP), established Atenises et al. [1] is pioneering in these studies. In their scheme, the file owner consumes only low storage space to keep the metadata for the file. The file is recorded by the owner to the cloud server together with the corresponding metadata and removed from the locals. Then the file owner or public auditor can accidentally question the data blocks of this file in the cloud server. Upon receipt of the call, the cloud server must generate evidence based on the file and the metadata it holds. By verifying this evidence, the auditor can check the integrity of the file. It can be seen that the PDP model allows users

or public auditors to check whether the cloud server holds an intact file without having to load the entire file. However, the PDP model was unable to solve the dynamic data audit. Atenises et al. [2] Furthermore, it designed a dynamic PDP scheme that implements a reliable method of editing and deleting data on the cloud server, but does not support data insertion. Juels and Kaliski [3] defined a model called as evidence of loading capacity (POR) and suggested a practical scheme. In this scheme, the data stored in the cloud can be obtained and ensured the integrity of this data. Based on the function of pseudorandom and signature BLS, Shacham and Waters have designed a private data integrity audit scheme and a public remote data integrity scheme. In order to protect the privacy of Wang et al. [5] He proposed a scheme of auditing integrity for integrity for privacy protection using random camouflage technique. Wang et al. [6] They proposed another scheme of the audit of data integrity to support full data dynamics using the Merle Hash tree. To reduce the damage to the exposure of users, Yu et al. [7] and [8] and YU and Wang [9] Proposed schemes of auditing resistant to key exposure resistant to resistance to integrations based on key update techniques [10]. Data sharing is an important application in cloud storage scenarios. To protect the privacy of the user's identity, Wang et al. [11] He suggested the audit of the shared privacy data to adjust the ring signature for secure cloud storage.It can be concluded that the solution of security concerns about data sharing, as mentioned above, should meet the following properties:

Hiding sensitive information. Cloud servers and researcher cannot stop sensitive information about general data. The traditional method is the encryption of the entire file, but encrypted data are not effective for researchers and therefore do not meet the general demand of data. Distribution of decryption keys to scientists seems to be useful for them to make the original normal use. In real scenarios, however, the database has no way to know scientists who use common files. As a result, it is unrealistic to hide sensitive information by encrypting the entire shared file. Therefore, the solution of confidential information that prevents parts with sensitive information should therefore be hidden, while other parts are publicly available.

A harmful prevention manager.

The solution can anticipate a harmful manipulation of shared data. More precisely, it is unable to pass the signatures created by the administrator to verify the cloud server and the auditor.

Auditing data integrity.

In order to guarantee the integrity of shared data and prevent dishonest operation of the cloud server, the solution should provide an effective mechanism of remote audit that meets the condition that sensitive parts of data are hidden.

To achieve data sharing with sensitive hidden information, we are considering using the idea in a disinfectant signature.

However, this is Ground if this disinfectable signature is used directly in the integrity of remote data. First, this signature in is designed on the basis of chameleon hash. However, many chameleons hash shows a problem with key exposure. To avoid this security problem, it requires a signature used in [40] Significantly unforgivable chameleon hash, which inevitable will cause huge computing overhead. Second, the signature used in does not support a block without a block. This means that the verifier must download all the data from the cloud to verify the integrity of the data, causing huge communication and excessive time verification time to verify. Third, the signature used in is based on PKI, which suffers from complicated certificates.

Algorithm Proof Verify(chal,pp,P) The TPA verifies the correctness of auditing proof as follows:

1. Naive Bayes Steps:

• Given training dataset D which consists of documents belonging to different class say Class A
and Class B
• Calculate the prior probability of class A=number of objects of class A/total number of objects
Calculate the prior probability of class B=number of objects of class B/total number of objects
• Find NI, the total no of frequency of each class
Na=the total no of frequency of class A
Nb=the total no of frequency of class B
Find conditional probability of keyword occurrence given a class:
P (value 1/Class A) =count/ni (A)

P (value 1/Class B) =count/ni (B)
P (value 2/Class A) =count/ni (A)
P (value 2/Class B) =count/ni (B)
P (value n/Class B) =count/ni (B)

IV.ARCHITECTURE- DIAGRAM

In the system, we design and develop a system for protecting information and confidential data. Our Main Purpose of the System is to protect the data from unauthorized access. Encrypting the whole shared file can realize the secret information hiding, but will make this shared file unable to be used by others. Signatures are used to verify the integrity of the sanitized file in the phase of integrity auditing. The user first blinds data blocks corresponding to the personal sensitive file information and generates the corresponding signatures. These signatures are used to guarantee the authenticity of the file and verify the integrity of the file. Then the user sends this blind file and its corresponding signatures of disinfection. After receiving a message from the user, the disinfector disinfects these blinds

Data blocks and data blocks corresponding to the sensitive information of the organization and then transform the signatures of disinfected data blocks into a valid for disinfected file. Finally, the disinfectant sends this sanitized file and its corresponding signatures to the cloud. These signatures are used to verify the integrity of the sanitized file in the Integrity audit phase. When TPA wants Verify the integrity of the sanitized file stored in the cloud and sends an audit challenge to the cloud.

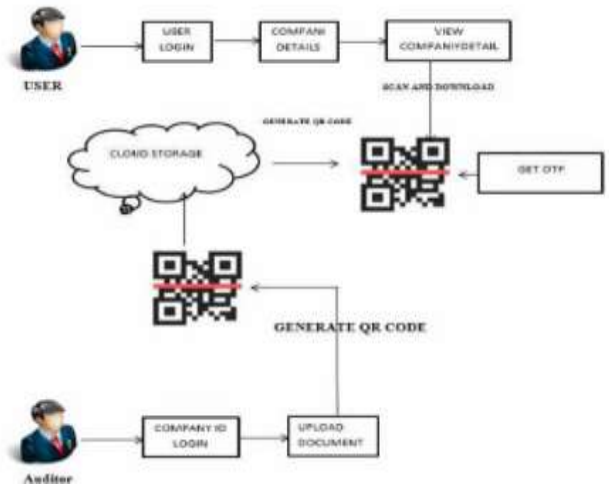

Fig.2 Architecture Diagram

CONCLUSION AND FUTURE SCOPE

In this article we have proposed to integrate data -based data Audit scheme for secure cloud storage that supports Sharing data with sensitive information. In our Scheme, file saved in the cloud can be shared and applied Others on condition that sensitive information file is protected. In addition, Auditing the Integrity of Remote Data It is still possible to effectively perform with time management. Demonstrates safety evidence and experimental analysis that the proposed scheme reaches the desired security and efficiency. Also the process of disinfection further increases the performance of an audit scheme where an approach. The disinfected file is only more difficult to use user assuming access.

REFERENCES

[1] G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z. Peterson, and D. Song, "Provable data possession at untrusted stores," in Proceed ings of the 14th ACM conference on Computer and co1munications security. Acm, 2007, pp. 598–609.

[2] G. Ateniese, R. Di Pietro, L. V. Mancini, and G. Tsudik, "Scalable and efficient provable data possession," in Proceedings of the 4th international conference on Security and privacy in communication netowrks. ACM, 2008, p. 9.

[3] A. Juels and B. S. Kaliski, Jr., "Pors: Proofs of retrievability for large files," in Proc. 14th ACM Conf. Comput. Commun. Secur., 2007, pp. 584–597.

[4] H. Shacham and B. Waters, "Compact proofs of retriev ability," J. Cryptol., vol. 26, no. 3, pp. 442–483, Jul. 2013.

[5] C. Wang, S. S. M. Chow, Q. Wang, K. Ren, and W. Lou, "Privacy preserving public auditing for secure cloud storage," IEEE Trans. Comput., vol. 62, no. 2, pp. 362–375, Feb. 2013.

[6] D. A. B. Fernandes, L. F. B. Soares, J. V. Gomes, M. M. Freire, and P. R. M. Inácio, ―Security issues in cloud environments: A survey,‖ Int. J. Inf. Secur., vol. 13, no. 2, pp. 113–170, Apr. 2014.

[7] L. F. B. Soares, D. A. B. Fernandes, J. V. Gomes, M. M. Freire, and P. R. M. Inácio, Cloud Security: State of the Art. Berlin, Germany: Springer, 2014.

[8] H. Shacham and B. Waters, ―Compact proofs of retrievability,‖ J. Cryptol., vol. 26, no. 3, pp. 442–483, Jul. 2013.

[9] C. Wang, S. S. M. Chow, Q. Wang, K. Ren, and W. Lou, ―Privacypreserving public auditing for secure cloud storage,‖ IEEE Trans. Comput., vol. 62, no. 2, pp. 362–375, Feb. 2013

[10] M. Green, ―The threat in the cloud,‖ IEEE Security Privacy, vol. 11, no. 1, pp. 86–89, Jan./Feb. 2013.

[11] K. Yang and X. Jia, ―Data storage auditing service in cloud computing: Challenges, methods and opportunities,‖ World Wide Web, vol. 15, no. 4, pp. 409–428, 2012.

[12] Q. Wang, C. Wang, K. Ren, W. Lou, and J. Li, "Enabling public auditability and data dynamics for storage security in cloud computing," IEEE Trans. Parallel Distrib. Syst., vol. 22, no. 5, pp. 847–859, May 2011.

[13] J. Yu, K. Ren, C. Wang, and V. Varadharajan, "Enabling cloud storage auditing with key-exposure resistance," IEEE Trans. Inf. Forensics Security, vol. 10, no. 6, pp. 1167–1179, Jun. 2015. [14] J. Yu, K. Ren, and C. Wang, "Enabling cloud storage auditing with verifiable outsourcing of key updates," IEEE Trans. Inf. Forensics Security, vol. 11, no. 6, pp. 1362–1375, Jun. 2016.

[15] J. Yu and H. Wang, "Strong key-exposure resilient auditing for secure cloud storage," IEEE Trans. Inf. Forensics Security, vol. 12, no. 8, pp. 1931–1940, Aug. 2017.

[16] J. Yu, R. Hao, H. Xia, H. Zhang, X. Cheng, and F. Kong, "Intrusionresilient identity-based signatures: Concrete scheme in the standard model and generic construction," Inf. Sci., vols. 442–443, pp. 158–172, May 2018.

[17] B. Wang, B. Li, and H. Li, "Oruta: Privacy-preserving public auditing for shared data in the cloud," in Proc. IEEE 5th Int. Conf. Cloud Comput. (CLOUD), Jun. 2012, pp. 295–302.