International Journal of Scientific Research in Science, Engineering and Technology
Print ISSN: 2395-1990 | Online ISSN : 2394-4099 (www.ijsrset.com)
doi : https://doi.org/10.32628/IJSRSET

Data Sharing and Sensitive Information Hiding with Revocation System in Cloud

Premnath G. Kharat[1], Prof. Ashok Kamthane[2]

[1]PG Scholar, Department of Computer Science and Engineering, MPGI School of Engineering, Nanded, Maharashtra, India

[2]Assistant Professor, Department of Computer Science and Engineering, MPGI School of Engineering, Nanded, Maharashtra, India

Article Info
Volume 7 Issue 4
Page Number: 202-209
Publication Issue :
July-August-2020

Article History
Accepted : 01 Aug 2020
Published : 05 Aug 2020

ABSTRACT

With the explosive growth of data, it is a heavy burden for users to store a large amount of data locally. Therefore, more and more organizations and individuals would like to store their data in the cloud. It provides a storage platform for enterprises and individuals. Cloud storage provides users with immediate access to a broad range of resources and applications hosted in the infrastructure of another organization via a web service interface. Storage maintenance tasks, such as purchasing additional storage capacity, are offloaded to the responsibility of a service provider. The security issues are one of the main exposures in cloud storage auditing. The Electronic Health Record (EHR), which is commonly used as cloud storage, contains some sensitive information and this sensitive information can be uncovered when cloud files are shared. Encrypting the entire shared file can realize the sensitive information hiding; however will make this shared file unable to be utilized by others. The most effective method to acknowledge data sharing to sensitive information hiding in remote data integrity auditing still has not been investigated up to now. We addressed such types of problems by proposing a Revocation algorithm for reliable data sharing with sensitive information hiding. The proposed system sanitized data block with respect to sensitive information of files and transforms these data block's signatures into valid ones for the sanitized file. The signature is used to verify the integrity of a sanitized file during an integrity audit. This technology can ensure the storage and sharing of files in the cloud and hide sensitive information.

Keywords: Cloud Storage, Data Sharing, Electronic Health Record, Sanitizer, Sensitive Information.

I. INTRODUCTION

As cloud computing, the new technology paradigm and expectations are growing more and more popular. To provides the user with the attractiveness of the calculated resources, Companies and those who outsource can take time to calculate the workload in the cloud and add funds to the Implementation and maintenance of the hardware and software.

Recently, the outsourcing calculation is noticed, and it is widely studied. It has been considered in numerous applications including scientific computing [1], linear algebra computing [2], linear programming computing [3], and modular power computing [4], and so on. In addition to cloud computing, it can also provide users with the charm of storage resources. Cloud storage is the most important service cloud computing to look at Universal. But the biggest benefit of cloud storage is that it allows users to take on new security challenges. While there is one important security issue, how efficiently integrity checks and data are stored.

To address this issue, many audit protocols for cloud storage have been proposed lately. Cloud storage is the trigger for new security threats to data owners. Multiple cloud users who use cloud storage to do some serious security work. The primary concern of cloud users is the integrity of externally-commissioned files. Several factors can lead to data corruption. First, we have earned the trust of cloud service providers. As a result, for financial reasons, the cloud service provider removed data that had been accessed only rarely, and spent additional money using other files, and secondly, the accumulated data was a hostile attack, or if there was an error in the cloud server failure management due to corruption. However, there is also a reputation for maintaining and loss of cloud service providers intentionally hides data. In cloud storage, data integrity and leakage have become the main concern of cloud users[5].

In recent years, the series of cloud storage security incidents and accidents have worsened for cloud users. Take Amazon's cloud crash disaster as an example. Amazon's huge EC2 cloud service broke the data of some of the cloud users permanently in 2011. Data loss was small compared to the total data saved, but anyone running a website immediately sees how the loss of any data corrupts what is the possibility of access data in case of insufficient detection data corruption. Therefore, it is becoming necessary for cloud users to frequently check that the outsourced data remains intact [6].

A key impact issue, as another important issue in cloud storage audit, has been addressed recently [7]. By nature, this problem itself is not trivial. Once the client's private key for auditing storage is opened to the cloud, the cloud can easily hide data loss incidents to maintain its reputation, even discard client data that is rarely accessed to save storage space. To solve this problem, propose a cancellation algorithm to prevent a sensitive file using the unauthorized people. The system of the first sanitizer [1], sanitizes blind blocks data and data blocks that match the organization's sensitive information and then convert the signature of the sanitized data blocks into the current sanitation file. Lastly, the sanitizer sends this sanitized file and its equivalent cloud signature which are used for the integrity of the hygiene file integrity test audit phase. After the sanitization process, TPA checks the integrity of sanitized files stored in the cloud and sends an audit challenge to the cloud. The cloud responds to TPA with audit evidence of data possession and finally, the TPA validates the integrity of the sanitized file to check whether this audit evidence is correct or not. In this way, our proposed system provides security for the sharing of data.

Objectives:

- To reduce the cost of data sharing services provided by cloud computing.

- The user stores personal data on the cloud server and it is easy to share only with authorized customers who have access to the data.
- Multiple user access the cloud service.
- To proposes, a Revocation Algorithm avoids the use of sensitive files from unauthorized users.

II. LITERATURE SURVEY

In paper [1], the author proposes a scheme to audit the integrity of remote data, which allows the sharing of data with hidden confidential information. In this method, the sanitizer is used to block data that corresponds to sensitive information in a file, and valid signatures for sanitized files this signature is used to verify the integrity of the sanitized files during the integrity audit phase. These technologies can ensure the storage and exchange of files in the cloud and hide sensitive information. This approach is based on an encryption-based identity. In paper [2], authors suggest an audit approach to cloud storage. Used to reduce the calculation load on the user side. This method introduces third-party media (TPM). Perform long tasks on your behalf. Where TPA generates authentication on behalf of the user and the user to verify the integrity of the data. In this way, data is protected from TPA. This article focuses on cloud storage auditing. The authors of the study called the audit process of cloud storage a major reduction in exposure to the damage of the client." This protocol may be used in a format that uses the key exposure power of the audit protocol and the security model definition. In this design, the binary tree structure and key navigation techniques are applied to update the private key of the client [3].

The author in [4] describes how to update keys, and the client suggested by the new key of the update method performs a cloud storage audit outsourcing key validation. This paradigm allows the keys to be updated safely and kept uniformly, so that the client's key update burden is particularly severe. It utilizes many existing audit designs and acts as the authorized party of this design to ensure the audit of the storage. In this design, TPA has kept all of these tedious tasks on behalf of the client, the encrypted version of the client's personal key, and formulated the definition of this paradigm with these surprising security models. Detailed design instantiations allow minimal safety certification-performance simulation is safe and efficient. The paper [5], propose an efficient public audit in maintaining the privacy and identity tracking of group members. Blind the system to obtain the privacy certification of the real data of the signature technology. Use the proposed method to further design the audit system for the actual scenario.

In paper [6] author suggests a new public audit mechanism for the consistency of shared data. Using the idea to close the proxy you can use the re-login block on behalf of existing users, you can cancel and re-sign existing users, and in any case share data back to the cloud without having to protect your data from being retrieved from the entire data cloud. At the same time the organization provides a comprehensive audit of audit work. In paper [7], uses the Shamir secret sharing concept to propose a new Public Audit Approach for the consistency of shared data. It also supports safe and efficient auditing by improving authentication tags based on polynomial. In this paper [8], an ID-based proxy-oriented data upload and remote data integrity checking model are proposed. The formal definition of a system model is a security model. Then design a specific ID-PUIC protocol using a bilinear coupling. Patent application security system [9], namely SecCloud. SecCloud helps clients generate data, as well as cloud Day SecCloud+stored in the integrity of the audit, even compared to the previous operation of cloud Map Reduce, is motivated by the fact that customers always want to encrypt data before uploading to the audit of the integrity of the encrypted data and the duplication of safety.

The authors [10] propose a new server-side de-duplication scheme for encrypted data. The change of ownership of the data completely encrypts the

random Fusion, utilizing the group's core distribution, and presents an outsider access control for the cloud where it prevents data leakage, but also cancels the user's data for the sake of honesty or more than the cloud storage server has proposed a method that also randomizes the attack against the inconsistency of we guarantee the integrity of your data. That's why security has proposed an enhanced plan.

III. PROPOSED SYSTEM

Proposed System Architecture

The system architecture contains five type entities namely: The User, The Sanitizer, The Private Key Generator (PKG), The Cloud, and last one the Third Party Auditor (TPA), this is shown in Fig. 1:

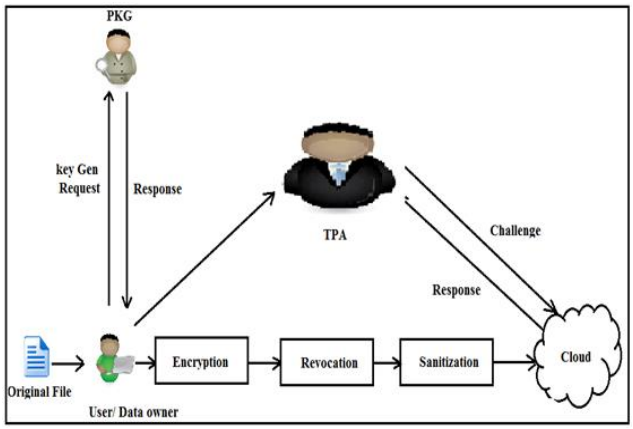

Fig 1: Proposed System Architecture

- **The cloud:** On the cloud, users save a lot of data. The cloud storage service allows users to upload data to the cloud, and automatically share data with other users.
- **The Sanitizer:** It is responsible for sanitizing data blocks that match to sensitive information in files (such as personal and organizational information), is responsible for sanitizing data blocks.
- **The Public Key Generation (PKG):** Other entities rely on the generation of Public Key. The functionality of this PKG is the generation of public parameters and private key for the user as per their identity ID.
- **Revocation:** Whenever data owner wants to share reliable data with some selected authorized user from the set of all authorized user of organization, the system use revocation algorithm to revoke the particular set of the user as input then select the encrypted reliable data and time stamp used for at what time revoked user is removed from the organization, then system update the list of authorized and sent it to the server with encrypted reliable data.
- **The Third-Party Auditor (TPA):** It is a public verifier. It checks the integrity of data stored in the cloud on behalf of users.

On the cloud, users save a lot of data. The cloud storage service allows users to upload data to the cloud, and automatically share data with other users.

The sanitizer, which is responsible for sanitizing data blocks that correspond to sensitive information in files (such as personal and organizational information), is responsible for sanitizing data blocks. Other entities rely on the generation of Public Key. The functionality of this PKG is the generation of public parameters and private key for the user according to their identity ID. Whenever data owner wants to share reliable data with some selected authorized user from the set of all authorized user of organization, use revocation algorithm to revoke the particular set of the user as input then select the encrypted reliable data and time stamp used for at what time revoked user is removed from the organization, then we update the list of authorized and sent it to the server with encrypted reliable data. A Third-party Auditor is a public verifier. It checks the integrity of data stored in the cloud on behalf of users.

In this system first blinds sensitive information data from files and then generate the corresponding signature for those blinding files. The signature is used to ensure the authenticity of the file and to verify the integrity of the file. Then the user sends the sanitizer a blind file and its corresponding signature. After receiving the user's message, the

sanitizer sanitizes these blinded blocks of data and data blocks that correspond to the sensitive information of the organization and then transforms the signatures of the sanitized data blocks into valid for the sanitized file. Finally, the sanitizer sends this sanitized file and its corresponding signature of the cloud. These signatures are used for the integrity of the audit phase of the sanitary file integrity test. After the sanitizing process, the TPA wants to check the integrity of the sanitized files stored in the cloud; it sends an auditing challenge to the cloud. And then, the cloud responds to the TPA with the audit-proof of data possession. And finally, the TPA verifies the integrity of the sanitized file to check whether this audit evidence is correct or not.

In the revocation system, add some user those who cannot able to download that sensitive file called revoked user. When a user requests to download the file from the cloud: cloud firstly checks the requested user if the revoked user if it is revoked user then the cloud will not allow downloading that file. If it is not a revoked user then he/she can download the file using the private key. Also, added the validation period to the file.

Algorithm

Algorithm 1: Identity-Based Integrity

1) Algorithm Setup
   i. The PKG chooses two multiplicative cyclic groups of prime order, a generator, a bilinear map, and a pseudorandom function.
   ii. The PKG randomly chooses an element and a cryptographic hash function.
   iii. The PKG computes the public value and the master secret key.
   iv. The PKG publishes system parameters and holds the master secret key msk.

2) Algorithm Extract
   i. After receiving the user's identity, the PKG randomly picks a value and computes it as the private key of the user ID. The PKG sends it to the user ID.
   ii. The user ID verifies the correctness of the received private key
   iii. If parameters do not hold, the user ID refuses the private key; otherwise, accepts it.

3) Algorithm SigGen
   i. The user ID randomly chooses a value and calculates a verification value.
   ii. The user ID employs the secret seed to calculate the blinding factor which is used to blind the data blocks corresponding to the personal sensitive information.
   iii. The user ID calculates a transformation value which is used to transform the signature in the Sanitization algorithm.

4) Algorithm Sanitization
   i. The sanitizer checks the validity of the file by verifying a valid signature.
   ii. The sanitizer respectively verifies the correctness of the signature

5) Algorithm ProofGen
   i. The TPA verifies the validity of the file
   ii. After receiving an auditing challenge from the TPA, the cloud generates a proof of data possession

6) Algorithm ProofVerify
   i. The TPA verifies the correctness of auditing proof

Algorithm 2: Revocation [Proposed Algorithm]

Input: List of the user for revocation

Output: Revoke user from an organization

Process:

1. Start
2. Take encrypted File and TimeStamp
3. Remove selected user from the organization for the selected Timestamp
4. Update list of user

5. Upload updated list of the user to a server with encrypted
6. End

IV. MATHEMATICAL MODEL

System S can be defined as:

1. Private Key Generation

Choose g, a number whose multiplicative order modulo p is q
q is the smallest positive integer such that
Prime p such that p – 1 is a multiple of q
Choose an element μ', μ1, μ2…… μn;
Let H be hashing function
U={UI, UF, Uu} is a user
UI={UI1, UI2} set on I/P
UI1= User authentication detail contain UID= (UID1, UID2, UIDn)
UI2= File data
Compute private key of user [1]

$$sk_{ID} = (g_2^x \left(\mu' \prod_{j=1}^{l} \mu_j^{UID}, g^{rID} \right)$$

rID= Randomly picks values from the user ID.

2. Generate Signature:

Compute Signature Sig [1]

$$sig = (g_2^x(\mu' \prod_{j=1}^{l} \mu_j^{UID})^{rID}, (H(name), \mu^{mi})^r))$$

Where,
m_i^* is a blinded file

3. Data Auditor:

DAI= {DAI1, DAI2}\\ A set of input
DAI1-User Registration Details
DAI2-Block Details
τ= tag or file [1]

$$\tau = \tau_0 || DAI$$

$$\tau_0 = name || g^{rID} || g^r$$

4. Sanitization

The Sanitizer checks the signature is valid or not.

The sanitizer respectively verifies the correctness of signature sig as follows:

$$e(sig, g) = e(g1, g2). e\left(\mu' \prod_{j-1}^{l} \mu_j^{ID_j}, g^{rID}\right). e(H(name||i). u^{m_i^*}, g^r)$$

5. Cloud:

CI= {CI1, CI2}, A set of input
CI1-Block Data
CI2-Challenge Message
CF= {CF1 CF2} \\ A set of functions
CF1=Save Blocks and it's hash
CF2=Generate Responce
CO=Output of cloud
CO= {CO1, CO2}
CO1=Set of Blocks = {b1, b2.., bn}
CO2=Generated Response of File block

6. Revocation:

P = {L,R,O}
L= {L1, L2}\\ A set of Input
L1= List of user
L2= Revoke list of user

R= {R1, R2} \\ A set Of Function
R1 = Revoke User
R2 = Secret key updating

O = {OP} \\ A set Of Output
OP= List of the revoked user

V. RESULT AND DISCUSSION

Experimental Setup

The system utilizes the Java framework of the Windows platform. The Net bean IDE is used as a development tool. The system does not require you to run specific hardware; any standard machine can run the application. As you acknowledge that you may

experience time in the actual cloud. For data sets, the system uses a large number of files of different sizes.

Result

This section discussed the experimental result of the proposed system. Table I shows the time comparison between existing and proposed system algorithms. Figure 2 shows the time require to access data of the proposed system with the existing system. We show that the data access time of the proposed revocation algorithm is small for data sharing with the selected authorized users and that there is no traffic in the data access process. When a hundred users send an administrator a request to access the file, the administrator will only allow 60 users. So the less time requires accessing the file and there is no traffic in the data accessing process.

Table I: Time Comparison

Algorithm	Time in ms
Data access time without Revocation	90
Data access time with Revocation	40

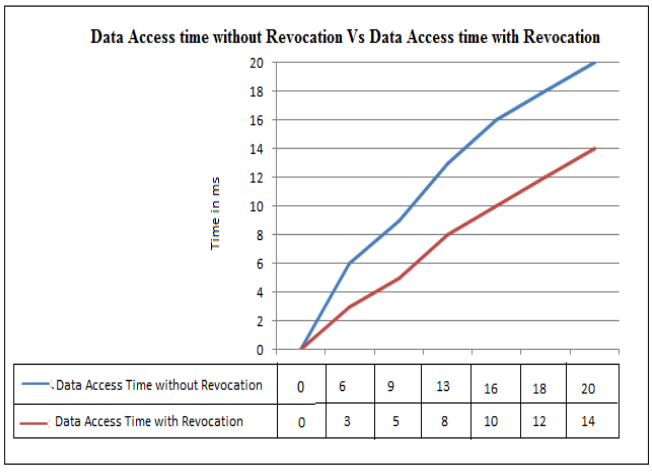

Fig 2: Data access time with and without Revocation

Table II shows a performance comparison. Figure 3 shows when many users improve the performance of their system with revocation and without revoked. The graph shows that the performance of the system with revocation is better than the performance of the system without revocation.

TABLE II: Performance Comparison

System	No of users
Performance of system without Revocation	20
Performance of system with Revocation	16

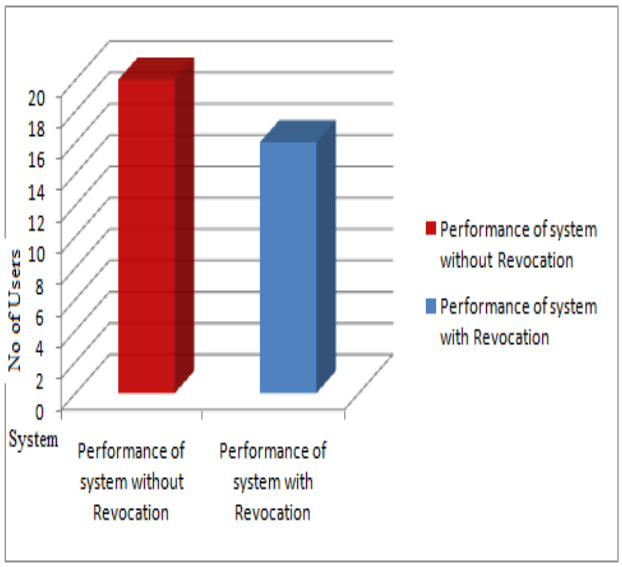

Fig 3: Performance Graph

VI. CONCLUSION

The paper proposes a Revocation algorithm for reliable data sharing that supports data sharing with sensitive information hiding. This supports data sharing with selected authorized users within an organization. Data accessing time of the proposed Revocation algorithm is less because of data sharing with selected authorized users and there is no traffic in the data accessing process. In this technique files stored in the cloud can be shared and used by others on the condition that the sensitive information of the file is protected. The proposed mechanism can achieve desirable security and efficiency.

VII.ACKNOWLEDGMENT

I would like to give my sincere gratitude to our guide I/C Dean Associate Prof. Ashok Kamthane who

encouraged and guided me throughout this paper. I am especially grateful for H.O.D and PG Co-Ordinator Prof. Shital Gaikwad for her valuable guidance and encouragement and for allowing me to use the college resources and facilities provided by them. Thanks and deep regards to I/C Dean Associate Prof. Ashok Kamthane for his support and encouragement.

VIII. REFERENCES

[1]. k. ren, c. wang, j. yu and v. varadharajan, "enabling cloud storage auditing with key-exposure resistance," IEEE [2015]

[2]. b. li, b. wang and h. li, "panda: public auditing for shared data with efficient user revocation in the cloud," [2015]

[3]. d. wang, m. xu, s. fu y. luo and j. deng, "efficient integrity auditing for shared data in the cloud with secure user revocation," [2015]

[4]. k. ren, j. yu and c. wang, "enabling cloud storage auditing with verifiable outsourcing of key updates," [2016]

[5]. w. shen, g. yang, q. su, z. fu, j. yu, and r. hao, "enabling public auditing for shared data in cloud storage supporting identity privacy and traceability," [2016]

[6]. d. he, h. wang, and s. tang, "identity-based proxy oriented data uploading and remote data integrity checking in public cloud," [2016]

[7]. d. xie, j. li, j. li and z. cai, "secure auditing and deduplicating data in cloud," [2016]

[8]. d. koo, j. hur, y. shin and k. kang, "secure data deduplication with dynamic ownership management in cloud storage," [2016]

[9]. h. zhang, x. lu, w. shen, h. xia, j. yu, and r. hao, "light-weight and privacy-preserving secure cloud auditing scheme for group users via the third party medium,", [2017]

[10]. jiayu, wentingshen, jingqin, ronghao, and jiankunhu, "enabling identity-based integrity auditing and data sharing with sensitive information hiding for secure cloud storage", [2018]

Author Profile

Mr. Premnath Gangaram Kharat received BE CSE Degree in 2008 from Solapur University and he is student of MPGI School of Engg. Doing ME in Computer Science and Engineering. An academic with 10 years of teaching experience.

Prof. Ashok Namdev Kamthane is a retired Associate Professor of the Department of Electronic and Telecommunication Engineering, S.G.G.S. Institute of Engineering and Technology, Nanded, Maharashtra, India. An academic with 37 years of teaching experience, he has authored more than a dozen books and presented several technical papers at national and international conferences. He has earned a first class in ME (Electronics) from S.G.G.S. College of Engineering and Technology. His ME dissertation work from Bhabha Atomic Research Center, Trombay, Mumbai, was on development of the hardware and software using 8051 (8-bit microcontroller) Acoustic Transceiver System required in submarines. He is currently working as Associate. Professor in Department of Computer Science and Engg. MPGI School of Engineering and Also he is Working as I/C Dean MPGI School of Engineering Nanded.

Cite this article as :

Premnath G. Kharat, Prof. Ashok Kamthane , "Data Sharing and Sensitive Information Hiding with Revocation System in Cloud", International Journal of Scientific Research in Science, Engineering and Technology (IJSRSET), Online ISSN : 2394-4099, Print ISSN : 2395-1990, Volume 7 Issue 4, pp. 202-209, July-August 2020.

Journal URL : http://ijsrset.com/IJSRSET1207458