Privacy-Preserving Data Integrity checking by using Secured Public Auditing on Cloud Storage

Sivakumar P and Ganesh kumar P

Article history:
Received 16 April 2015
Accepted 12 June 2015
Available online 26 June 2015

Keywords:
Public auditing, privacy-preserving, data sharing, cloud computing

ABSTRACT

Background: Cloud computing becomes the most popular environment due to its nature of sharing the resources to the cloud users as per their requirement. Security and data integrity becomes the biggest issue in cloud environment where the cloud users are storing their sensitive information and enabling access to other users. The cloud providers need to provide the promising environment to the cloud user to increase their reputation level. Objective: The main goal of this work is to implement the privacy preserved secured public auditing in the cloud data storage environment where the sensitive information are shared. To do so, data's are encrypted in block level before storing it in the cloud server and make ease of data integrity checking. The Third Party Auditor (TPA) is introduced for public auditing in order to make sure the data integrity of cloud data which can also reduce the burden of cloud server. Results: The experimental tests were conducted between the existing approach and the proposed methodology amongst the varying set of workload sizes. The performance evaluation was conducted between the terms called the query time, computation overhead and storage overhead. The results conducted were proves that the proposed methodology leads to better result than the existing approach in term of all performance metrics. Conclusion: Our experimental results exhibit the effectiveness and efficiency of our mechanism when auditing the integrity of shared data.

INTRODUCTION

Grid At lower trivial cost the users able to store the data and can access the shared data in the cloud. These services are provided by the cloud service provider such as Amazon where they are having more powerful data centers. In mutual manner the data stored in the cloud is shared among a group of members, most cloud storage present Google drive, icloud.

As the cloud environment is untrusted, the cloud users are would be conjecture to know as their data stored in the cloud remain unharmed for longer period of time, this is because of sudden hardware failures and human errors. To make this issue even better the service providers of cloud inform to users about the data errors, in order to maintain position of their services and to avoid profits losing. Thus before the data utilization such as working out over cloud data or search the integrity of cloud data is verified.

The traditional methods for checking data rightness is to take back the whole data from the cloud then verifying the data integrity by read-through the correctness of signatures(eg.,) RSA algorithm or hash values. Indeed this standard is able to effectively check the correctness of data in cloud. Even though the efficiency of using this traditional approach for checking the data correctness in cloud is in doubt. The central cause is that in general the size of cloud data is huge. However most of prior work focus only on checking the integrity of personal data in the cloud.

Downloading the whole data from cloud for verifying data integrity will cost more or waste user’s amounts of resources and working out especially when data have been corrupted in cloud. The cloud users need not to download the whole data to local machine during the process of data mining. Because the service providers offers users to perform computation directly on data that is stored in large servers of cloud.

In recent times many mechanisms have been projected to allow not only data owners but also public verifiers for professionally checking the integrity without downloading the whole data from cloud, which is referred as public auditing. To guard data in cloud it is best to introduce third party auditor.
(TPA) to perform auditing tasks on behalf of users which provides the peace of mind to users of cloud.

Recently wand et al, has proposed privacy preserving public auditing mechanism for data that is shared among the group members of cloud based on the ring signatures, in which the identity of the signer on each block in shared data is closed to the third party auditor during performing of auditing task. By ring signatures the information used for verification is large as well as for auditing it takes time. It increases the linearly with number of users in group. Thus ring signature introduces a major computation burden to all users. Additionally ring signature prevents group manager to sketch identity of user when someone in the group is misbehaved.

The rest of paper is organized as follows: Section 2 overviews the related work. In Section 3 the proposed system is presented. Section 4 shows performance analysis and Section 5 concludes the paper.

**Related works:**

Ateniese et al, proposed the provable data possession in which the public auditing is performed by a verifier on integrity of data without retrieving the entire data that is stored in untrusted cloud server. However this method is only fitting for inert data.

Jules and kaliski (2007) defined a further related model called proofs of retrievability (POR), which is also able to check the data correctness on an untrusted server of cloud. The original file is added with sentinels. The sentinels are the randomly valued check blocks. The verifier challenges the untrusted server of the cloud where the data is stored, by specifying the position of a sentinels group and requesting the server to return the coupled sentinels value. Ateniese et al has projected a scalable and efficient provable data possession using symmetric keys to develop the competence of verification. Regrettably this approach cannot able to hold up for public verification and also this offers only a few authentication requests to each users.

Wang and Li (2014) designed two enhanced proofs of retrievability mechanisms which are based on signatures of BLS and pseudo random functions. To support for fully dynamic data the Zhu et al has subjugated index hash tables. Based on the rank based authenticated dictionary a dynamic PDP has been proposed by the author Erway. Hao also designed a approach in which the RSA method is used for active public auditing system.

Zhu et al has designed a approach to preserve privacy data in cloud server from the third party auditor. In (2012) the author has proposed a cryptographic storage scheme that enables safe file sharing on cloud servers. In this approach, the files are divided into filegroups and these filegroups is encrypted with unique file block key, thus the owner of the data share the filegroups with others through delivering the interconnected lockbox key. The lockbox key is used to encrypt the block keys of files. On the other hand this approach brings overhead for large scale file sharing because of sharing of heavy key between users.

Ateniese et al (2009) projected a proxy reencryptions system to protect the circulated storage exclusively the data owner encrypts blocks of content keys. These keys are additional encrypted by using master public key. For access control the cloud server uses proxy cryptography to reencrypt directly the corresponding content keys by using public key of granted users. Regrettably between the server the collision attack occurs and revoked malicious user knows the decryption keys of all the blocks that are encrypted.

The author Yu presented a scalable and grained data access control scheme in cloud based on KP-ABE system (2009). In this system the owner of the data will encrypt file by using a random key, where this key is again encrypted with attributes of a set by using KP-ABE. Even though single owner method make difficult for the implementation of applications with this scenario, where any user in a group should be allowed to share or to store the data files in cloud.

The author Lu (2014) based on group signatures and cipher text policy the secure provenance method has been proposed. In this scheme every user has two keys and attribute key and also the system is set with single attribute. Thus the user able to encrypt data file using attribute based encryption and the other users in group decrypt the encrypted data. For the time being the user signs encrypted data with group signature key for privacy preserving and traceability. However this scheme does not supports to the user revocation.

In this paper we take advantages of group signatures and active transmit encryption to construct fully homomorphic authenticators where the privacy is preserved and thus traceability is achieved.

**Privacy preserved secured public auditing:**

The company uses a cloud to store their data; it allows its staff to share their data files among their group. As there are three diverse entities, they are the cloud, secondly the group manager and the group members. In cloud the cloud service providers like Amazon provides the data storage services to companies. The server in the cloud is untrusted because the server will not delete the user data but will try to know the content of user stored data and identities of users.

The administrator of company is the group manager; the manager charge is generation of system parameters, registration of users, user revocation and revealing the identity of clash data owner. Group members are the trusted members as they store their private data and also share their data among there group members in the cloud server. The number of users in group changes dynamically.
Fig. 1: Model of proposed system.

The Broadcast encryption enables the sender to pass on their encrypted data to a set of users thus only honored subset of users can decrypt the data. Besides the active transmit encryption allows the manager of group to actively take in new members while preserving the information, i.e., the size of cipher text and group encryption key are not required any modification and also the decryption key need not to be recomputed.

The author Chaum and van Heyst (1990) has first introduced the concept of group signatures. The group signature scheme allows the users in a group to sign messages while maintaining the identity from verifiers. To register in group the group manager for each user \(i\) with identity \(ID_i\), the group manager randomly selects a number \(Z_i\) and computes \(A_i\) and \(B_i\) equation as

\[ A_i = R \in P_1 \]
\[ B_i = G \in P_1 \]

The group manager adds \((A_i, Z_i, ID_i)\) into group list of users, this information is used in traceability phase. After the registration through fully homomorphic encryption the user gets a private decryption key, public key and evaluation key. The ID represents the group identification.

**FHE KeyGen(a) = (P_{uk}, P_{rk}, P_{ek})**: The \(P_{uk}\) represents the public key and \(P_{rk}\) represents the private key.

**FHE Encry_{pk}(b) = c**: Encrypting the data of user using public key which gives the output cipher text.

**FHE Decry_{ek}(c) = b**: The cipher text is converted to plain text of user data using the private key.

Through publicly available revocation list of users the group manager performs user revocation, based on which the users encrypt the data files and thus guarantee the secrecy against revoked members. The group manager daily updates the list even when no user revoked. To declare the validity the list is encircled by signature sig.

**Revocation Verification**:

Input: System Parameters \((H_0, H1, H2)\), group signature \(\Omega\), revocation keys \(A_1...A_n\)

Output: Valid or Invalid

Begin

\[ Set\ temp = e(D_1,H1)e(D_2,H_2) \]
For \(i = 0\) to \(n-1\)
If \(e(D_1-A_i, H_0) = temp\)
Return valid
Else
Return Invalid
End

**Traceability**:

When clash occurs in data the tracing operation is performed by manager of the group to discover the real owner of data identity. The signature is \(\Omega=(D_1,D_2,D_3,c,s_0,s_1,s_2,s_3)\) and then manager employs his private key as \((x_1,x_2)\) to work out \(A_i=D_3-(x_1.D_1+x_2.D_2)\). To find out the list of user identity the manager can find by parameter \(A_i\).

Input: The user revocation parameters are \((R_1, x_1),...,(R_r, x_r)\), and the private key is \((A, x)\).

Output: \(A_r\) or Null

Begin

\[ Set\ temp = A \]
For \(\delta = 1\) to \(r\)
If \(x=x_\delta\)
Return null
Else set temp = \((P_{uk}- temp) \)
Return temp
End

The size of the group signature is not dependents on the number of users in the group.

**Experimental result**:

The performance evaluation of our proposed approach is done compare the effectiveness of our algorithm. The performance evaluation is done based many performance metrics. Those are Query time, computation overhead, storage overhead, and Query response time.

**Query time**:

The time taken to submit a query with difference workload sizes are evaluated and compared with the existing algorithms. The below graph shows the comparison of proposed and existing algorithm with different file sizes.
In Fig 2 represents the Query time of existing and proposed schemes. In that graph X-axis represents the file size in the range of 5 to 35 and Y-axis represents the query time various from 0 to 3000 ms. From that results shown in Fig 2, our proposed scheme has efficient than the existing system.

**Computation overhead:**

The computation overhead defines the overall processing capacity utilized for process the user submitted query. The computation overhead should be minimized in order to improve the overall effectiveness of the proposed method.

![Fig. 3: Computation Overhead comparison.](image)

In Fig 3 represents the computation overhead of existing and proposed schemes. In that graph X-axis represents the number of nodes in the range of 5 to 45 and Y-axis represents the computation overhead in the range of 0 to 60. From that results shown in Fig 3, our proposed scheme has efficient than the existing system.

![Fig. 4: Storage Overhead.](image)

**Storage overhead:**

The storage overhead defines the amount of space consumed for storing the data owner’s encrypted data into the cloud servers. It is ratio between the time complexity taken to store the data and retrieve the stored data.

In Fig 4 represents the storage overhead of existing and proposed schemes. In that graph X-axis represents the number of nodes in the range of 5 to 45 and Y-axis represents the storage space various from 0 to 250 MB. From that results shown in Fig 4, our proposed scheme has efficient than the existing system.

**Conclusion:**

Cloud computing becomes the most popular technology due to its functionality of outsourcing its resources to the cloud providers. The proposed methodology in this work aims to preserve the privacy of the cloud users by encrypting the sensitive information of the cloud users before storing it in the cloud storage. This methodology also enables the anonymous data authentication over the cloud data by the third party users. The burden of cloud service providers are reduced considerably by introducing the third party public auditor who is responsible for checking the integrity of the cloud data. The experimental tests conducted were proves that the proposed methodology provides the better result than the existing approaches.

**REFERENCES**


