Authorized Dynamic Fair Public Auditing of Data on Cloud

R. Santha Maria Rani[1], Dr. Lata Ragha[2]
[1, 2] Department of Computer Engineering
[1] Terna Engineering College
[2] Fr. C. Rodrigues Institute of Technology, Maharashtra, India

Abstract- *Cloud storage is the service available to users over a network where the data is maintained, managed, backed up remotely. The data auditing with Third Party Auditor (TPA) is for the integrity of data stored in the cloud. The auditing scheme consists of data owner, TPA and cloud server. The data owner performs operations such as splitting the file to blocks, encrypting them, generating a hash value for each, concatenating it. The TPA is checking the data integrity. In cloud, both owner and the CSP can spoof or masquerade. The CSP has the motive to reclaim sold storage by deleting never accessed data. So to maintain a reputation, hides data loss accidents. Also the owner intentionally claims data corruption as he can get compensation from the CSP. So the TPA should be able to monitor the integrity and authenticity of the entities. This paper proposes an efficient and secure auditing scheme where Third Party Arbitrator helps TPA to find the honesty of both. Also it provides privacy preserving, public auditing, data integrity and confidentiality.*

Keywords- Third Party Auditor (TPA), Third Party Arbitrator(TPAR),Cloud Service Providers (CSPs), Merkle-Hash Tree(MHT), Provable data Possession (PDP),Dynamic Hash Table(DHT).

I. INTRODUCTION

Cloud computing is useful for data storage and processing. Many data applications are using cloud system as Infrastructure, Platform, and Software that users can use IT services [1][2][3][4].

Compared to traditional systems Cloud computing has some advantages.

- The investment amount to purchase and maintain IT facilities is reduced.
- It provides elasticity, scalability and efficiency for task executions.

Cloud stores the data to data centers which are large and remotely located. A Trusted Third Party Auditor (TPA) secures interactions between Cloud user and cloud service provider. TPA is to provide end-to-end security. The TPA demands retrieval of user data which does not remain secret. Also TPA has to remember the keys for transactions. It is represented in Fig 1.

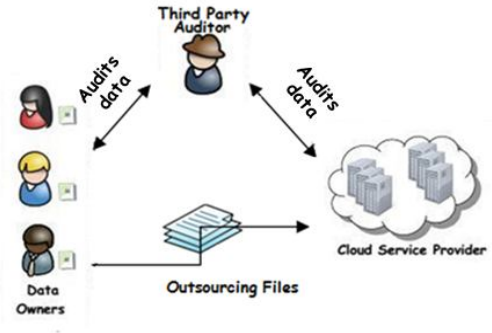

Figure 1. Relationships between Owner, CSP and TPA [12]

Any small change in network data needs assessing full data block. Storing data to the cloud has security threats:

- Network attacks, hardware failures and administrative errors.
- CSP may reclaim storage of never accessed data, or hide data loss accidents for its reputation.
- Downloading all the data for checking integrity is not viable because of expensive communication overhead.

As the data audit is conducted by a trusted third party, it is named as auditing-as-a-service [AaaS] [5] by cloud user's perspective. A security problem in supporting public verifiability is achieved by adding an additional authorization with client, CSS and a third-party auditor (TPA).

II. RELATED WORK

In Dynamic Audit Service Outsourcing for Data Integrity [6] scheme, the problem of providing simultaneous public auditability and data dynamics for remote data integrity check is explored. Also it improved the existing proofs by manipulating the classic Merkle Hash Tree construction for

block tag authentication. It explores the technique of bilinear aggregate signature to extend into a multiuser setting to support efficient handling of multiple auditing tasks, where TPA can perform multiple auditing tasks simultaneously. In MuR-DPA(Multi Replica – Dynamic Public auditing): Top-down Leveled Multi-replica based Secure Public Auditing [7] Scheme, all replica blocks for each data block are organized into a same replica sub-tree. This can support fully dynamic data updates, authentication of block indices and efficient verification of updates. It experienced much less communication overhead for both update verification and integrity verification. Whereas the difficulty was in supporting secure public auditing of dynamic data and streaming data with constant-sized integrity proofs. Scalable Two-Phase Top-Down Specialization Approach [8] Scheme proposed a scalable two-phase top-down specialization (TDS) approach to anonymize large-scale data sets using the Map Reduce framework on cloud. Identity-Based Distributed Provable Data Possession [9] scheme proposed the first ID-DPDP protocol which is provably secure under the assumption that the CDH problem is hard. ID-DPDP protocol can realize private verification, delegated verification and public verification based on the client's authorization. In Dynamic-Hash-Table Based Public Auditing [10], a public auditing scheme for secure cloud storage using dynamic hash table (DHT), it migrated the auditing metadata from the CSP to the TPA, and thereby significantly reduces the computational cost and communication overhead. It exploited the aggregate BLS (Boneh–Lynn–Shacham) technique from bilinear maps to perform multiple auditing tasks simultaneously. All the signatures by different users on various data blocks into a single and verify it for only one time to reduce the communication cost in the verification process. The existing data auditing schemes have various potential risks and inefficiency such as security risks in unauthorized auditing requests and inefficiency in processing small updates still exist. And also they have not seriously considered the fairness problem as they usually assume an authentic owner against an untrusted Cloud Service provider (CSP).

Table 1. Comparison of Existing Schemes

Year	Authors	Technique	Method used	Short comes
2011	Q. Wang, C.Wang, K.Ren, W.Lou J. Li,	DPDP-MHT	Merkle hash tree (MHT)	Heavy computation cost of the TPA Large communication overhead
2014	C. Liu, C. Yang, X. Zhang L. Wang I.J. Chen	MUR-DPA	Used Authenticated Data Structure (ADS) based on the MHT	Works only with constant-sized integrity proofs
2014	X. Zhang, L.T. Yang, C. Liu, J. Chen	TDS	Map Reduce framework	No scalable privacy preservation aware analysis
2015	Wang, Huaqun	ID-DPDP	Distributed Provable Data Possession (DPDP)	Verification delay occurs
2016	H. Tian, Y. Chen, C. Chang	DHT-PA (Dynamic hash table-public audit)	Dynamic Hash table	Communication cost is greater

III. PROPOSED METHODOLOGY

A. Architecture of Proposed System:

In the cloud environment, both clients and CSPs have the motive to create a false entity. The scheme which supports the variable-sized data blocks, authorized third party auditing and fine-grained dynamic data updates is described in four parts which are shown in Fig. 2 is as follows:

- User: Users have data to be stored in the cloud.
- Cloud Service Provider (CSP): A CSP has significant resources and expertise in building and managing distributed cloud storage servers.
- Third Party Auditor (TPA): A TPA is trusted to assess the cloud storage services.
- Third Party Arbitrator (TPAR): A TPAR is trusted to fairly settle any dispute about proof verification and dynamic update, and find out the false entity party.

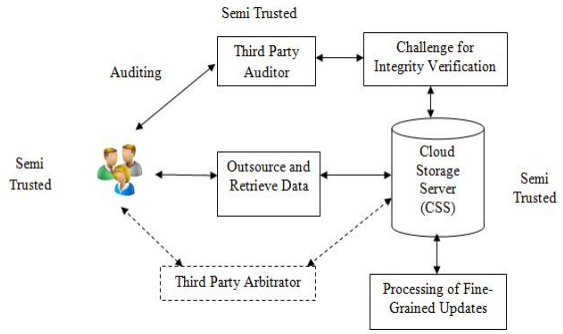

Figure 2. Dynamic Data Auditing Scheme

The proposed mechanism works as follows:

1. The owner of the file will upload the data to the cloud [The file is fragmented into blocks, encrypted and stored in cloud].Each file is recognized by a unique ID and the details about file ID and its associated block data are maintained in a log file in cloud by CSS[for each file the log detail is updated].
2. After uploading, the owner of the file can update /modify the existing file in cloud.
3. When the file gets uploaded to the cloud the third party auditor (TPA) verifies the file and its blocks. If the file is consistent, then verified flag is set with unique key value.
4. Similarly the CSS also verifies the file like TPA. The generated key will be similar, if the file is consistent.
5. The client after logging in to the cloud can get to view the files available, but the client cannot access it.
6. Before giving access to the client the third party arbitrator authenticates the file by matching the key provided by TPA and CSS to TPAR; if it matches, then the client is given access to the file.
7. After the updation of the file, step 3 to step 7 are followed the same.
8. For each client's request, before giving access to the client, the TPAR authenticates and then proceeds accordingly.

B. Algorithm used in Proposed System:

1) Integrity Verification

1. TPA must show CSS that it is authorized by the file owner before it can challenge a certain file. TPA will decide to verify some blocks from the total blocks. Then, a challenge message is generated. TPA then sends challenge to CSS.
2. After receiving challenge, CSS will first verify signature and the client's public key and output REJECT if it fails. Otherwise, CSS will compose the proof P as then output P. CSS will send P to TPA.
3. After receiving P, TPA verify signature by using public keys. If they are equal, then returns TRUE, otherwise it returns FALSE.

2) Arbitration

a) **Arbitration on Integrity Proof**

1) The Third party Arbitrator (TPAR) requests (C_Seq, S_Sig) from the client. Then he checks the signature S_Sig of the CSP. If it is invalid, the TPAR may unauthorized the client for their ineffiency ; otherwise the TPAR proceeds.
2) The TPAR requests {S_Seq,C_Sig} from the CSP. Then he checks the signature C_Sig of the client. If the signature does not verify correctly, the TPAR may unauthorized the CSP for their ineffiency; otherwise the TPAR proceeds.
3) If C_Seq = S_Seq, then the TPAR requests from the client the challenged set that causes dispute on proof verification and retransmit it to the CSP to run the auditing scheme. The CSP computes the proof returns it to the TPAR for verification.
4) If there is a mismatch in C_Seq and S_Seq. The TPAR can be sure that the party who gives a smaller sequence number is performing a replay attack; he may unauthorize the cheating party. Specifically, if C_Seq > S_Seq, the client is cheating by replaying an old signature from the CSP; if S_Seq > C_Seq, the CSP is cheating by replaying an old signature from the client.

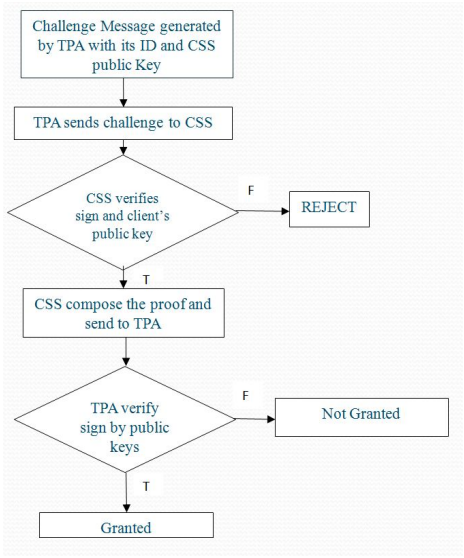

Figure 3. Flowchart for Arbitration on Integrity Proof

b) Arbitration on Dynamic Update

The first two steps are the same as that of the arbitration protocol on integrity proof. According to the result of sequence number comparison (C_Seq and S_Seq), we divide the protocol into two situations.

*The sequence numbers match (C_Seq = S_Seq)

1) The TPAR requests the update record from the client.
2) For block modification and insertion, the TPAR verifies the correctness. If fails, the TPAR may unauthorized the client for cheating; otherwise, the TPAR is convinced that the updated block.

3) The TPAR transmits to the CSP, and requests on the small challenge set from the CSP. If fails, the TPAR may unauthorized the CSP for denying the update; otherwise, the TPAR proceeds.
4) The TPAR requests and verifies new signatures from both parties. The TPAR may unauthorize the party who sends an invalid signature. If both signatures verify, the TPAR forwards to the CSP, and to the client.

*The sequence numbers mismatch (C_Seg ≠ S_Seq)

1. C_Seq < S_Seq. The server is cheating by replaying an old signature from the client.
2. C_Seq > S_Seq + 1. The client is cheating by replaying an old signature from the CSP.
3. C_Seq = S_Seq + 1. This occurs when the CSP receives the client's update request and refuses to update and send his signature to the client. There are three possibilities here.
4. The update record from the client is invalid, so the CSP refuses to update and contacts the TPAR for arbitration.
5. The update record from the client is valid, but the CSP responds with invalid signature, so the client contacts the TPAR for arbitration.
6. The update record from the client is valid, but the CSP maliciously denies the update, so the client contacts the TPAR for arbitration.

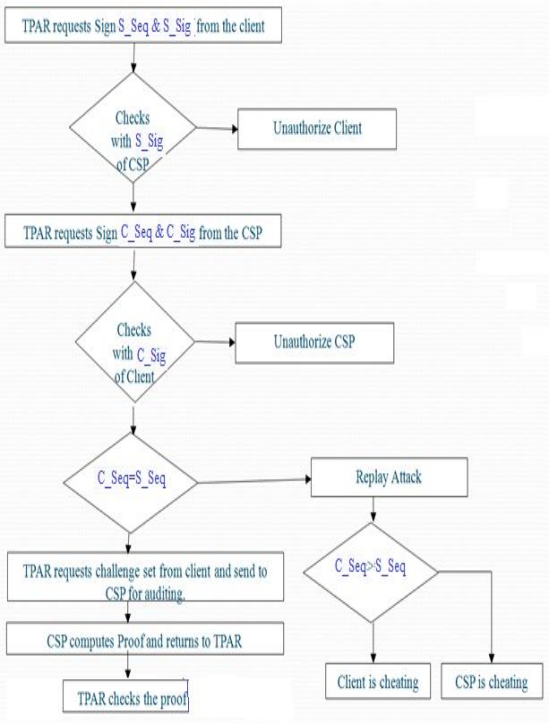

Figure 4. Flowchart for Arbitration on Dynamic Update

IV. IMPLEMENTATION

A. Implementation details and modules:

1) Registration Phase:

The owner and user can login only after CSP acceptance.TPA is helping user to authenticate the owner/user.

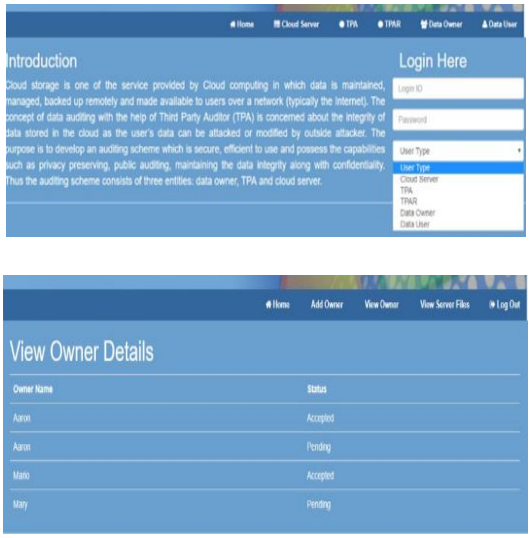

Figure 5. Registration Phase

3) Verification Phase:

When the file gets uploaded to the cloud the third party auditor (TPA) verifies the file and its blocks. If the file is consistent, then verified flag is set with unique key value. Similarly the CSS also verifies the file, the same way as (TPA) and the generated key will also be similar, if the file is consistent. The client after logging in to the cloud can get to view the files available, but the client cannot access it. Before giving access to the client the third party arbitrator authenticates the file by matching the key provided by TPA and CSS to TPAR; if it matches, then the client is given access to the file. The client can view only specific owner's files if that owner has granted the permission to access. To access the file, the client has to explicitly send a request for the specific file.

Figure 6. Verification Phase

4) Security Phase:

If an unauthorized user is trying to access the file, then CSP will block the user.

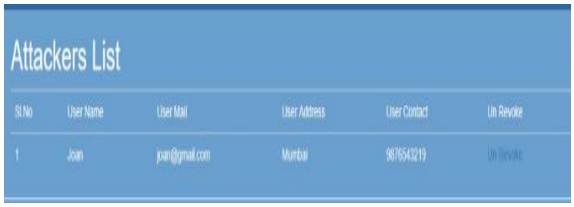

Figure 7. Security Phase

B. Result analysis

According to our auditing system proof is generated for every block of maximum 2000 characters. So the proof size is increased when the file size increased. But the number of proofs are used here is limited which means the proofs are repeated randomly. So storage cost of proofs is reduced. At the same time security level increased because of multiple proofs for single size.

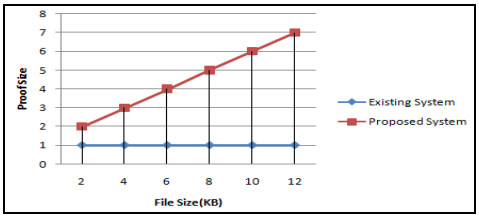

Figure 8. Proof Size vs. File Size

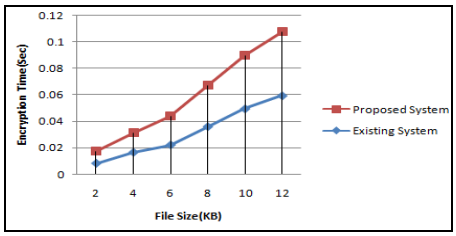

Figure 9 (a). Encryption Time vs. File Size

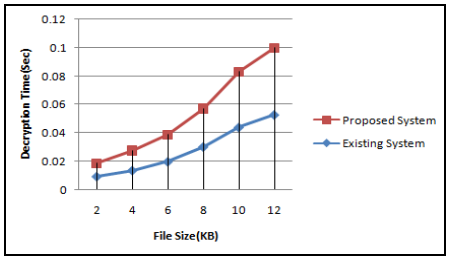

Figure 9(b). Decryption Time vs. File Size

The proposed system results to support larger file blocks with multiple (but only a predefined size of) sectors each. The updates chosen for experiments are filled with random data. Results are shown in Fig. 8 and 9. For updates in files of equal size, the increased storage on CSS in proposed system stays constant, while in the old scheme, the storage increases linearly with the increase in size of the affected block.

V. CONCLUSION

This scheme has provided a formal analysis on possible types of fine-grained data updates and proposed a scheme that can fully support authorized auditing and fine-grained update requests. Based on this scheme, we have also proposed a modification that can dramatically reduce communication overheads for verifications of small update. Theoretical analysis and experimental results have demonstrated that this scheme can offer enhanced security and flexibility. Also it significantly minimizes overheads for data applications with a large number of frequent minor updates in social media and business transactions.

REFERENCES

[1] G. Ateniese, R.B. Johns, R. Curtmola, J. Herring, L. Kissner, Z. Peterson, and D. Song, ‘‘Provable Data Possession at Untrusted Stores,’’ in Proc. 14th ACM Conf. on Comput. and Commun. Security (CCS), 2007, pp. 598-609.

[2] H. Shacham and B. Waters, ‘‘Compact Proofs of Retrievability,’’ in Proc. 14th Int’l Conf. on Theory and Appl. of Cryptol. and Inf. Security (ASIACRYPT), 2008, pp. 90-107.

[3] R. Curtmola, O. Khan, R.C. Burns, and G. Ateniese, ‘‘MR-PDP: Multiple-Replica Provable Data Possession,’’ in Proc. 28th IEEE Conf. on Distrib. Comput. Syst. (ICDCS), 2008, pp. 411-420.

[4] C. Erway, A. Kupcu, C. Papamanthou, and R. Tamassia,

‘‘Dynamic Provable Data Possession,’’ in Proc. 16th ACM Conf. on Comput. and Commun. Security (CCS), 2009, pp. 213-222.

[5] Chang Liu, Jinjun Chen, Laurence T. Yang, Xuyun Zhang,Chi Yang, Rajiv Ranjan, And Ramamohanarao Kotagiri” Authorized Public Auditing Of Dynamic Big Data Storage On Cloud With Efficient Verifiable Fine-Grained Updates” IEEE Transactions On Parallel And Distributed Systems, Vol. 25, No. 9, September 2014.

[6] Q. Wang, C. Wang, K. Ren, W. Lou and J. Li, ‘‘Enabling Public Auditability and Data Dynamics for Storage Security in Cloud Computing’’, Vol. 22, no. 5, IEEE Trans. on Parallel and Distributed Systems, pp. 847-859, 2011.

[7] Chang Liu ; Rajiv Ranjan ; Chi Yang ; Xuyun Zhang ; Lizhe Wang ; Jinjun Chen “MuR-DPA: Top-Down Levelled Multi-Replica Merkle Hash Tree Based Secure Public Auditing for Dynamic Big Data Storage on Cloud”,Vol. 64,no. 9, IEEE Trans. on Computers,pp. 2609 - 2622 , 2015.

[8] C. Wang, Q. Wang, K. Ren, and W. Lou, ‘‘Privacy-Preserving Public Auditing for Data Storage Security in Cloud Computing,’’ in Proc. 30st IEEE Conf. on Comput. and Commun. (INFOCOM), 2010, pp. 1-9.

[9] Wang, Huaqun. "Identity-Based Distributed Provable Data Possession in Multicloud Storage", Services Computing, IEEE Transactions on 8.2 (2015): 328-340.

[10] H. Tian, Y. Chen, C. Chang, “Dynamic-Hash-Table Based Public Auditing for Secure Cloud Storage”, Vol. PP, Issue: 99, IEEE Transactions on Service Computing, Manuscript ID, DEC 2016

[11] Chang Liu, Jinjun Chen,”Authorized Public Auditing of Dynamic Big Data Storage on Cloud with Efficient Verifiable Fine-Grained Updates”, IEEE Transaction on Parallel and Distributed System, vol. 25, no 9,September 2014

[12] R.S.M Rani, Dr. Lata Ragha “Dynamic Public Data Auditing Schemes On Cloud: A Survey” International Journal of Advanced Research in Computer Science and Software Engineering, Vol 8, January 2018: pp 76-78