International Journal of Scientific Engineering and Technology Research

ISSN 2319-8885
Vol.07,Issue.04,
April-2018,
Pages:0918-0921

www.ijsetr.com

Group Users Data Dynamics with Fair Arbitration and Auditing in Cloud

K. V. CHAITHANYA KRISHNA KANTH[1], DR. K. SREENIVASULU[2]

[1]PG Scholar, Dept of CSE, G.Pullaiah College of Engineering And Technology, AP, India,E-mail: kvchaitanya.cse@gmail.com.

[2]Professor, Dept of CSE, G.Pullaiah College of Engineering And Technology, AP, India, E-mail: sreenu.kutala@gmail.com.

Abstract: Many new security threats are found in the outsourcing of data to the cloud. The powerful machines and strong security mechanisms were provided by the Cloud Service Provider (CSP) in early stage. Data auditing scheme enables the cloud users to check the integrity of the stored data without downloading them, they are known as block-less verification. Auditing schemes help the user to interact with the CSP to check the correctness of their outsourced data. The current scheme, achieves a efficient handling of dynamic data. So, we extend the existing threat model by adopting signature exchange idea, so that possible dispute will settle fairly .We enhanced our schema is secure and performance evaluation. Perform evaluation will say about the evaluation of the overhead of data dynamics. We differentiate block indices, tag indices and devise an index switcher to avoid tag re-computation caused by block up data operation. In existing, the client and the CSP potentially misbehave while auditing and data update. So we extended to provide fair arbitration for solving disputes between group of owners and clients and the CSP and promotion of auditing schemes in the cloud environment.

Keywords: Cloud Service Provider (CSP), Provable Data Possession (PDP), Proofs of Retrievability (PoR).

I. INTRODUCTION

Data auditing schemes can enable cloud users to check the integrity of their remotely stored data without downloading them locally, which is termed as block less verification. With auditing schemes, users can periodically interact with the CSP through auditing protocols to check the correctness of their outsourced data by verifying the integrity proof computed by the CSP, which offers stronger confidence in data security because user's own conclusion that data is intact is much more convincing than that from service providers. Generally speaking, there are several trends in the development of auditing schemes. First of all, earlier auditing schemes usually require the CSP to generate a deterministic proof by accessing the whole data file to perform integrity check, e.g., schemes in [1], [2] use the entire file to perform modular exponentiations. Such plain solutions incur expensive computation overhead at the server side; hence they lack efficiency and practicality when dealing with large-size data. Represented by the "sampling" method in "Proofs of Retrievability" (PoR) [3] model and "Provable Data Possession" (PDP) [4] model, later schemes [5], [6] tend to provide a probabilistic proof by accessing part of the file, which obviously enhances the auditing efficiency over earlier schemes. Secondly, some auditing schemes [3], [7] provide private verifiability that require only the data owner who has the private key to perform the auditing task, which may potentially overburden the owner due to its limited computation capability. Ateniese el al. [4] was the first to propose to enable public verifiability in auditing schemes.

In contrast, public auditing schemes [5], [6] allow anyone who has the public key to perform the auditing, which makes it possible for the auditing task to be delegated to an external third party auditor (TPA). A TPA can perform the integrity check on behalf of the data owner and honestly report the auditing result to him [8]. Thirdly, PDP [4] and PoR [3] intend to audit static data that are seldom updated, so these schemes do not provide data dynamics support. But from a general perspective, data update is a very common requirement for cloud applications. If auditing schemes could only deal with static data, their practicability and scalability will be limited. On the other hand, direct extensions of these static data oriented schemes to support dynamic update may cause other security threats, as explained in [6]. To our knowledge, only schemes in [6], [9], [10] provide built-in support for fully data dynamic operations (i.e., modification, insertion and deletion), but they are insufficient in providing data dynamics support, public verifiability and auditing efficiency simultaneously, as will be analyzed in the section of related work. From these trends, it can be seen that providing probabilistic proof, public verifiability and data dynamics support are three most crucial characteristics in auditing schemes. Among them, providing data dynamics support is the most challenging. This is because most existing auditing schemes intend to embed a block's index i into its tag computation, e.g., H(i||v) in [4] or H(name||i) in [5], which serves to authenticate challenged blocks. However, if we insert or delete a block, block indices of all subsequent blocks will change, then tags of these blocks have to be re-computed.

This is unacceptable because of its high computation overhead. We address this problem by differentiating between tag index (used for tag computation) and block index (indicate block position), and rely an index switcher to keep a mapping between them. Upon each update operation, we allocate a new tag index for the operating block and update the mapping between tag indices and block indices. Such a layer of indirection between block indices Band tag indices enforces block authentication and avoids tag re-computation of blocks after the operation position simultaneously. As a result, the efficiency of handling data dynamics is greatly enhanced. Furthermore and important, in a public auditing scenario, a data owner always delegates his auditing tasks to a TPA who is trusted by the owner but not necessarily by the cloud. Current research usually assumes an honest data owner in their security models, which has an inborn inclination toward cloud users. However, the fact is, not only the cloud, but also cloud users, have the motive to engage in deceitful behaviors. For example, a malicious data owner may intentionally claim data corruption against an honest cloud for a money compensation, and a dishonest CSP may delete rarely accessed data to save storage. Therefore, it is of critical importance for an auditing scheme to provide fairness guarantee to settle potential disputes between the two parties.

Zheng et al. [11] proposed a fair PoR scheme to prevent a dishonest client from accusing an honest CSP, but their scheme only realizes private auditing. Kupccu [12] proposed general arbitration protocols with automated payments using fair signature exchange protocols [13]. Our work also adopts the idea of signature exchange to ensure the metadata correctness and protocol fairness, and we concentrate on combining efficient data dynamics support and fair dispute arbitration into a single auditing scheme. To address the fairness problem in auditing, we introduce a third-party arbitrator (TPAR) into our threat model, which is a professional institute for conflicts arbitration and is trusted and payed by both data owners and the CSP. Since a TPA can be viewed as a delegator of the data owner and is not necessarily trusted by the CSP, we differentiate between the roles of auditor and arbitrator. Moreover, we adopt the idea of signature exchange to ensure metadata correctness and provide dispute arbitration, where any conflict about auditing or data update can be fairly arbitrated in group.

II. PROBLEM STATEMENT

Auditing schemes mainly focus on the delegation of auditing tasks to a third party auditor (TPA) so that the overhead on clients can be offloaded as much as possible. However, such models have not seriously considered the fairness problem as they usually assume an honest owner against an untrusted CSP. Since the TPA acts on behalf of the owner, then to what extent could the CSP trust the auditing result? What if the owner and TPA collude together against an honest CSP for a financial compensation? In this sense, such models reduce the practicality and applicability of auditing schemes. In a cloud scenario, both owners and CSP have the motive to cheat. The CSP makes profit by selling its storage capacity to cloud users, so he has the motive to reclaim sold storage by deleting rarely or never accessed data, and even hides data loss accidents to maintain a reputation. Here, we assume the CSP is semi-trusted, namely, the CSP behaves properly as prescribed contract most of the time, but he may try to pass the integrity check without possessing correct data. On the other hand, the owner also has the motive to falsely accuse an honest CSP, e.g., a malicious owner intentionally claims data corruption despite the fact to the contrary so that he can get a compensation from the CSP. Therefore, disputes between the two parties are unavoidable to a certain degree. So an arbitrator for dispute settlement is indispensable for a fair auditing scheme. We extend the threat model in existing public schemes by differentiating between the auditor (TPAU) and the arbitrator (TPAR) and putting different trust assumptions on them. Because the TPAU is mainly a delegated party to check client's data integrity, and the potential dispute may occur between the TPAU and the CSP, so the arbitrator should be an unbiased third party who is different to the TPAU. As for the TPAR, we consider it honest-but-curious. It will behave honestly most of the time but it is also curious about the content of the auditing data, thus the privacy protection of the auditing data should be considered.

III. IMPLEMENTATION

In the cloud environment, both clients and CSPs have the motive to cheat. In our group user's scheme, the index switcher is used by the auditor to obtain tag indices for requested blocks at proof verification phase, thus the verification result relies on the correctness of the index switcher. However, the generation and update of index switcher are performed by the data owner only, it will potentially give a dishonest owner the opportunity of falsely accusing an honest CSP. In this sense, we must provide some mechanism to ensure the correctness of the index switcher and further the fairness of possible arbitration, so that no group can frame the other group without being detected. Straightforward way is to let the arbitrator (TPAR) keep a copy of the index switcher. Since the change of the index switcher is caused by dynamic operations, the client can send necessary update information (i.e., operation type, operation position, new tag index) to the TPAR for each update operation. With this information, the arbitrator could re-construct the latest version of the index switcher, whose correctness decides the validity of later arbitration. However, such a solution costs O(n) storage at the arbitrator side and needs the arbitrator to be involved in each update operation. Ideally, we want the TPAR only undertake the role of an arbitrator who involves only at dispute settlement, and maintains a constant storage for state information, i.e., public keys of the client and the CSP. As an alternative, we employ the signature exchange idea in to ensure the correctness of the index switcher. Specifically, we rely on both parties exchanging their signatures on the latest index switcher at each dynamic operation. We further extend the project by implementing the data dynamicity, fair arbitration on Group of user and owner.

IV. SYSTEM MODEL

The system model involves four different entities: the data owner/cloud user, who has a large amount of data to be stored in the cloud, and will dynamically update his data (e.g., insert, delete or modify a data block) in the future; the cloud service provider (CSP), who has massive storage space and computing power that users do not possess, stores and manages user's data and related metadata (i.e., the tag set and the index switcher); the third party auditor (TPAU) is similar to the role of TPA in existing schemes, who is a public verifier with expertise and capabilities for auditing, and is trusted and payed by the data owner (but not necessarily trusted by the cloud) to assess the integrity of the owner's remotely stored data; the third party arbitrator (TPAR), who is a professional institute for conflict arbitration and trusted by both the owner and the CSP, which is different to the role of TPAU. Cloud users rely on the CSP for data storage and maintenance, and they may access and update their data. To alleviate their burden, cloud users can delegate auditing tasks to the TPAU, who periodically performs the auditing and honestly reports the result to users. Additionally, cloud users may perform auditing tasks themselves if necessary as shown in Fig.1. For potential disputes between the auditor and the CSP, the TPAR can fairly settle the disputes on proof verification or data update. Note in following sections, we may use the terms "TPAU" and "auditor" interchangeably, so are the terms "TPAR" and "arbitrator". We further extend the project by implementing the data dynamicity, fair arbitration on Group of user and owner.

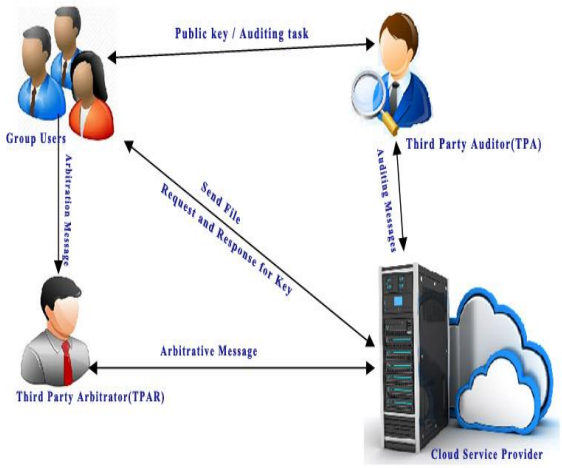

Fig.1. System Architecture.

V. MODULES DESCRIPTION

A. Group users

This module includes the group User registration and group user login details. Every Group User need to register while accessing to the cloud. Every Group User will be activated by the Cloud. After Cloud activated, every Client need to provide public key to login the user home. Public key will be provided by third party auditor. Client can view file details and can insert, modify and delete the file with help of TPAR. Client will have the TPAR message whenever the user update the file.

B. Third Party Auditor (TPA) Module

It acts as semi-cloud. PA Provide public key for every user to access the user home page. After cloud given auditing proof then only TPA can audit all files.

C. Third Party Arbitrator (TPA) Module

It acts as fair dispute for users and cloud. Intimate the files message, each time user insert, modify, delete files to cloud. Send TPAR message to user and cloud.

D. Cloud Module

Activate data client. Cloud sends storage auditing proof for all files to TPA. Cloud can view the client downloaded files from cloud. Cloud will have the TPAR message whenever the user updates the file.

VI. CONCLUSION

In this paper we define the efficient fair dispute arbitration, integrity auditing, and an effective data dynamics for group of owners and users. We introduce a third-party arbitrator which is a professional institute for conflicts arbitration and is trusted and played by both data owners and the CSP. Since a TPA can be viewed as a delegator of the data owner and is not necessarily trusted by the CSP, we differentiate between the roles of auditor and arbitrator. Moreover, we adopt the idea of signature exchange to ensure metadata correctness and provide dispute arbitration, where any conflict about auditing or data update can be fairly arbitrated in a group. We address the data dynamics problem by differentiating between tag index (used for tag computation) and block index (indicate block position), and rely an index switcher to keep a mapping between them. Upon each update operation, we allocate a new tag index for the operating block and update the mapping between tag indices and block indices. Such a layer of indirection between block indices and tag indices enforces block authentication and avoids tag re-computation of blocks after the operation position simultaneously. As a result, the efficiency of handling data dynamics is greatly enhanced. Generally, this paper proposes a new auditing scheme to address the problems of data dynamics support, public verifiability and dispute arbitration simultaneously in a group.

VII. REFERENCES

[1] Y. Deswarte, J.-J. Quisquater, and A. Sa¨ıdane, "Remote integrity checking," in Proc. 5th Working Conf. Integrity and Intl Control in Information Systems, 2004, pp. 1–11.

[2]D.L.GazzoniFilho and P.S.L. M. Barreto, "Demonstrating data possession and uncheatable data transfer." IACR Cryptology ePrint Archive, Report 2006/150, 2006.

[3] A. Juels and B. S. KaliskiJr, "Pors: Proofs of retrievability for large files," in Proc. 14th ACM Conf. Computer and Comm. Security (CCS07), 2007, pp. 584–597.

[4] 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. Computer and Comm. Security (CCS07), 2007, pp. 598–609.

[5] H. Shacham and B. Waters, "Compact proofs of retrievability," in Proc. 14th Intl Conf. Theory and Application of Cryptology and Information Security: Advances in Cryptology (ASIACRYPT 08), 2008, pp. 90–107.
[6] Q. Wang, C. Wang, J. Li, K. Ren, and W. Lou, "Enabling public verifiability and data dynamics for storage security in cloud computing," in Proc. 14th European Conf. Research in Computer Security (ESORICS 08), 2009, pp. 355–370.
[7] M. A. Shah, R. Swaminathan, and M. Baker, "Privacy-preserving audit and extraction of digital contents." IACR Cryptology ePrint Archive, Report 2008/186, 2008.
[8] C. Wang, K. Ren, W. Lou, and J. Li, "Toward publicly auditable secure cloud data storage services," Network, IEEE, vol. 24, no. 4, pp. 19–24, 2010.
[9] C. Erway, A. K¨upc¸ ¨ u, C. Papamanthou, and R. Tamassia, "Dynamic provable data possession," in Proc. 16th ACM Conf. Computer and Comm. Security (CCS 09), 2009, pp. 213–222.
[10] Y. Zhu, H.Wang, Z. Hu, G.-J.Ahn, H. Hu, and S. S. Yau, "Dynamic audit services for integrity verification of outsourced storages in clouds," in Proc. ACM Symp. Applied Computing (SAC 11), 2011, pp. 1550–1557.
[11] Q. Zheng and S. Xu, "Fair and dynamic proofs of retrievability," in Proc. 1st ACM Conf. Data and Application Security and Privacy (CODASPY 11), 2011, pp. 237–248.
[12] A. K¨upc¸ ¨ u, "Official arbitration with secure cloud storage application," The Computer Journal, pp. 138–169, 2013.
[13] N. Asokan, V. Shoup, and M. Waidner, "Optimistic fair exchange of digital signatures," in Proc. 17th Intl Conf. Theory and Applications of Cryptographic Techniques: Advances in Cryptology (EUROCRYPT98), 1998, pp. 591–606.
[14] E.-C. Chang and J. Xu, "Remote integrity check with dishonest storageserver," in Proc. 13th European Conf. Research in Computer Security (ESORICS 08), 2008, pp. 223–237.
[15] F. Seb´e, J. Domingo-Ferrer, A. Martinez-Balleste, Y. Deswarte, and J.-J.Quisquater, "Efficient remote data possession checking in critical information infrastructures," IEEE Trans. Knowledge and Data Eng., vol. 20, no. 8, pp. 1034–1038, 2008.