ZKG INTERNATIONAL ISSN: 2366-1313

On Security of an Identity- Based Dynamic Data Auditing Protocol for big Data Storage

[1]CH SUKANYA, [2]BANDARU AKHIL, [3]KORAMONI NAVEEN KUMAR, [4]AVULA HARISH

[1]Assistant Professor, Dept.of CSE, Teegala Krishna Reddy Engineering College, Meerpet, Hyderabad,

[2]BTech student, Dept.of CSE, Teegala Krishna Reddy Engineering College, Meerpet, Hyderabad, akhilbandaru05@gmail.com

[3]BTech student, Dept.of CSE, Teegala Krishna Reddy Engineering College, Meerpet, Hyderabad, Koramoninaveenkumar@gmail.com

[4]BTech student, Dept.of CSE, Teegala Krishna Reddy Engineering College, Meerpet, Hyderabad, avulaharishy@gmail.com

Abstract: *In this paper, we point out the security weakness of Shang et al.'s identity-based dynamic data auditing protocol for big data storage. Specifically, we identify that their protocol is vulnerable to a secret key reveal attack, i.e., the service provider (SP) can reveal the secret key of the data owner (DO) from the stored data. Further, SP can also generate proof to pass the challenge of TPA (Third-party auditor) even if all block and tag pairs have been deleted. We hope that by identifying these design flaws, similar weaknesses can be avoided in future designs.*

Keywords: *Cloud computing, Third party auditor, Data owner, Service provider, Cloud security.*

I. INTRODUCTION

With the explosive growth of data, it is a heavy burden for users to store the sheer amount of data locally. Therefore, more and more organizations and individuals would like to store their data in the cloud. However, the data stored in the cloud might be corrupted or lost due to the inevitable software bugs, hardware faults, and human errors in the cloud. In order to verify whether the data is stored correctly in the cloud, many remote data integrity auditing schemes have been proposed. In remote data integrity auditing schemes, the data owner first needs to generate signatures for data blocks before uploading them to the cloud. These signatures are used to prove the cloud truly possesses these data blocks in the phase of integrity auditing. And then the data owner uploads these data blocks along with their corresponding signatures to the cloud. The data stored in the cloud is often shared across multiple users in many cloud

ISSN: 2366-1313

storage applications, such as Google Drive, Dropbox, and iCloud. Data sharing as one of the most common features in cloud storage, allows a number of users to share their data with others. However, these shared data stored in the cloud might contain some sensitive information. For instance, the Electronic Health Records (EHRs) stored and shared in the cloud usually contain patients' sensitive information (patient's name, telephone number, ID number, etc.) and the hospital's sensitive information (hospital's name, etc.). If these EHRs are directly uploaded to the cloud to be shared for research purposes, the sensitive information of patients and hospital will be inevitably exposed to the cloud and the researchers. Besides, the integrity of the EHRs needs to be guaranteed due to the existence of human errors and software/hardware failures in the cloud. Therefore, it is important to accomplish remote data integrity auditing on the condition that the sensitive information of shared data is protected [1].

In today's world of digitization, cloud computing has emerged as a concept for handling big data. This paper focuses on the nature of Big Data, the origin of big data, and security-related issues with big data. Data originated from various domains like science, education, industry, healthcare, and many more. The features of data generated from different sources are different. The definition of Big data includes 5 V: Velocity, Volume, Variety, Value, and Veracity. Big data is supported by new infrastructure and tools. Cloud-based infrastructure, storage, network, and high computing performance help to manage the feature of big data. New data-centric security models for trusted infrastructure and data processing and storage are also proposed for the above purpose. Big Data is not a simple Database rather it contains large-scale data processing and data analytics. The most important part of big data is its support for Dynamicity. Big data require different data-centric operational models and protocols. Sometimes an object or event-related data go through a number of transformations and became more distributed between traditional security domains [2].

Cloud storage, as it can provide users with efficient, secure, and low-cost storage services without having to build a storage platform by themselves, has become a popular application along with the wide spread of cloud computing. However, as data users lose physical control over their data, the security of the outsourced data has attracted researchers' considerable attention. Among all security issues of

ISSN: 2366-1313

cloud storage, data integrity is the most basic one, as it convinces data users that the data, they store on the service provider is complete. To deal with the data integrity issue, the concept of data integrity auditing has been proposed, which usually adopts a third-party auditor to audit whether the service provider has stored the users' data intact. Up to now, many public data integrity auditing protocols have been proposed by researchers to achieve different security and functional features, such as privacy preservation, storage correctness, batch verification, and dynamic support. However, most of the solutions are designed based on the public key infrastructure (PKI), which however brings a heavy key management burden to the data users. As a result, identity-based cryptography has been exploited by researchers in public auditing protocols to avoid the potential key management problem [3].

MOTIVATION

We investigate how to achieve data sharing with sensitive information hiding in remote data integrity auditing, and propose a new concept called identity-based shared data integrity auditing with sensitive information hiding for secure cloud storage. In such a scheme, sensitive information can be protected and other information can be published. It makes the file stored in the cloud able to be shared and used by others on the condition that the sensitive information is protected, while the remote data integrity auditing is still able to be efficiently executed.

We design a practical identity-based shared data integrity auditing scheme with sensitive information hiding for secure cloud storage. A sanitizer is used to sanitize the data blocks corresponding to the sensitive information of the file. In our detailed scheme, firstly, the user blinds the data blocks corresponding to the personal sensitive information of the original file and generates the corresponding signatures, and then sends them to a sanitizer. The sanitizer sanitizes these blinded data blocks into a uniform format and also sanitizes the data blocks corresponding to the organization's sensitive information. It also transforms the corresponding signatures into valid ones for the sanitized file. This method not only realizes remote data integrity auditing but also supports data sharing on the condition that sensitive information is protected in cloud storage. To the best of our knowledge, this is the first scheme with the above functions. Besides, our scheme is based on identity-based cryptography, which simplifies complex certificate management.

II. LITERATURE SURVEY

ISSN: 2366-1313

Cloud computing is in high demand today because of its features like scalability, elasticity, and efficiency in supporting dynamic data. Cloud users are able to conveniently scale up/ down their virtual allocated resources according to their current needs with minimal management effort and service interruption. The most existing problem in the cloud is data security and privacy. Integrity verification for outsourced data storage is the main area of today's research. Jules et al. [4] proposed a model based on POR which is only applicable to static data storage.

In order to verify the integrity of the data stored in the cloud, many remote data integrity auditing schemes have been proposed. To reduce the computation burden on the user side, a Third-Party Auditor (TPA) is introduced to periodically verify the integrity of the cloud data on behalf of the user. Ateniese et al. first proposed a notion of Provable Data Possession (PDP) to ensure data possession on the un-trusted cloud. In their proposed scheme, homomorphic authenticators and random sampling strategies are used to achieve block less verification and reduce I/O costs. Juels and Kaliski defined a model named Proof of Retrievability (PoR) and proposed a practical scheme. In this scheme, the data stored in the cloud can be retrieved and the integrity of these data can be ensured. Based on pseudo-random function and BLS signature

Ateniese et al. [5] proposed the first remote data auditing scheme which can check the integrity of static data, then they improved this scheme to support dynamic auditing of appending operations.

Juels et al. [6] proposed a recoverable scheme for large files, namely proofs of retrievability (POR). This scheme can restore data if data is damaged. But the number of times the data owner can verify data is limited. Based on their work, Shacham and Waters proposed an improved solution, namely compact POR, which can verify data for unlimited times. Both schemes do not support dynamic operations.

Erway et al. [7] proposed the first dynamic data auditing scheme. They presented a rank-based authenticated dictionary built over a skip list to support fully dynamic operations, including insertion, deletion, modification, and appending.

Wang et al. [8] proposed a third-party auditor (TPA) cloud storage security model where TPA audits data on behalf of data owners. One of the drawbacks of this scheme is not support privacy protection.

Wang et al. [9] proposed a public auditing scheme that can realize privacy protection

ISSN: 2366-1313

in cloud storage services. This scheme uses random mask technology to ensure that TPA cannot obtain any useful information from the auditing process.

Sookhak et al. [12] proposed a remote data auditing scheme using the algebraic signature of which the computation cost is very low. This scheme has a security issue that the server can use a valid response to answer all challenges

III. PROPOSED WORK

In this section, we briefly review Shang et al.'s identity-based dynamic data auditing protocol for big data storage, which mainly includes four entities, i.e., KGC (Key Generate Center), DO (data owner), SP (service provider), and TPA (third-party auditor), where KGC is responsible for the parameter generation and SP is an un-trusted party. Specifically, their protocol comprises eight algorithms, i.e., Setup, Extract, TagGen, Challenge, GenProof, CheckProof, ExcuteUpdate and VerifyUpdate. For the eight algorithms, we will omit reviewing the last two dynamic operation algorithms in their protocol, as they are not directly related to our attack.

IMPLEMENTATION

In this example, the sensitive information of EHRs contains two parts. One is the personal sensitive information (patient's sensitive information), such as patient's name and patient's ID number. The other is the organization's sensitive information (hospital's sensitive information), such as the hospital's name∗ . Generally speaking, the above sensitive information should be replaced with wildcards when the EHRs are uploaded to cloud for research purpose. The sanitizer can be viewed as the administrator of the EHR information system in a hospital. The personal sensitive information should not be exposed to the sanitizer. And all of the sensitive information should not be exposed to the cloud and the shared users. A medical doctor needs to generate and send the EHRs of patients to the sanitizer for storing them in the EHR information system. However, these EHRs usually contain the sensitive information of patient and hospital, such as patient's name, patient's ID number and hospital's name. To preserve the privacy of patient from the sanitizer, the medical doctor will blind the patient's sensitive information of each EHR before sending this EHR to the sanitizer.

The medical doctor then generates signatures for this blinded EHR and sends them to the sanitizer. The sanitizer stores these messages into EHR information system. When the medical doctor needs the EHR, he sends a request to the sanitizer. And then the sanitizer downloads

ISSN: 2366-1313

the blinded EHR from the EHR information system and sends it to the medical doctor. Finally, the medical doctor recovers the original EHR from this blinded EHR. When this EHR needs to be uploaded and shared in the cloud for research purpose, in order to unify the format, the sanitizer needs to sanitize the data blocks corresponding to the patient's sensitive information of the EHR. In addition, to protect the privacy of hospital, the sanitizer needs to sanitize the data blocks corresponding to the hospital's sensitive information. Generally, these data blocks are replaced with wildcards. Furthermore, the sanitizer can transform these data blocks' signatures into valid ones for the sanitized EHR. It makes the remote data integrity auditing still able to be effectively performed. During the process of sanitization, the sanitizer does not need to interact with medical doctors. Finally, the sanitizer uploads these sanitized EHRs and their corresponding signatures to the cloud. In this way, the EHRs can be shared and used by researchers, while the sensitive information of EHRs can be hidden. Meanwhile, the integrity of these EHRs stored in the cloud can be ensured.

The sanitizer is necessary because of the following reasons. Firstly, after the data blocks corresponding to the patient's sensitive information are blinded, the contents of these data blocks might become messy code. The sanitizer can unify the format by using wildcards to replace the contents of these data blocks. In addition, the sanitizer also can sanitize the data blocks corresponding to the hospital's sensitive information such as hospital's name by using wildcards, which protects the privacy of the hospital. Secondly, the sanitizer can facilitate the information management. It can sanitize the EHRs in bulk, and uploads these sanitized EHRs to the cloud at a fixed time. Thirdly, when the medical doctor needs the EHR, the sanitizer as the administrator of EHR information system can download

IV. METHODOLOGY

The user firstly blinds the data blocks corresponding to the personal sensitive information of the file, 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 blinded file and its corresponding signatures to the sanitizer. After receiving the message from the user, the sanitizer sanitizes these blinded data blocks and the data blocks corresponding to the organization's sensitive information, and then transforms the signatures of sanitized data blocks into valid ones for the sanitized file. Finally,

ISSN: 2366-1313

the sanitizer 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 phase of integrity auditing. When the TPA wants to verify the integrity of the sanitized file stored in the cloud, he sends an auditing challenge to the cloud. And then, the cloud responds to the TPA with an auditing proof of data possession. Finally, the TPA verifies the integrity of the sanitized file by checking whether this auditing proof is correct or not.

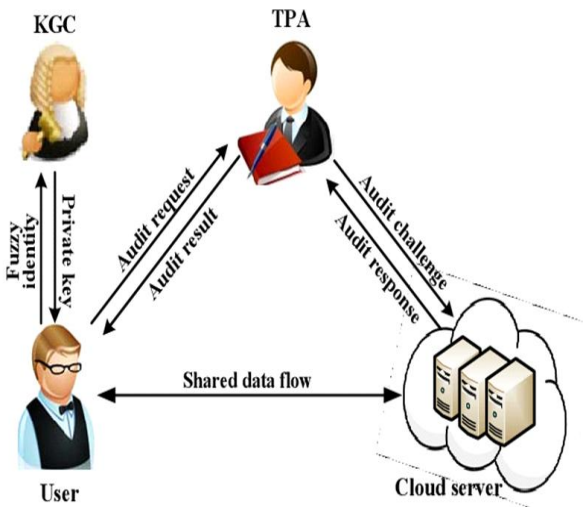

Fig.1 System architecture

V. RESULTS

We first give the functionality comparison among our scheme and several related schemes, and the computation overhead comparison between our scheme and Shacham et al. scheme. And then discuss the communication overhead and the computation complexity of our scheme. At last, we evaluate the performance of our scheme in experiment.

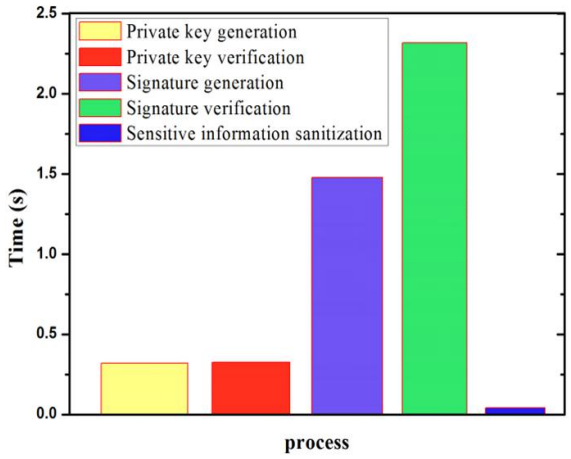

Fig.2 Performance of difference processes

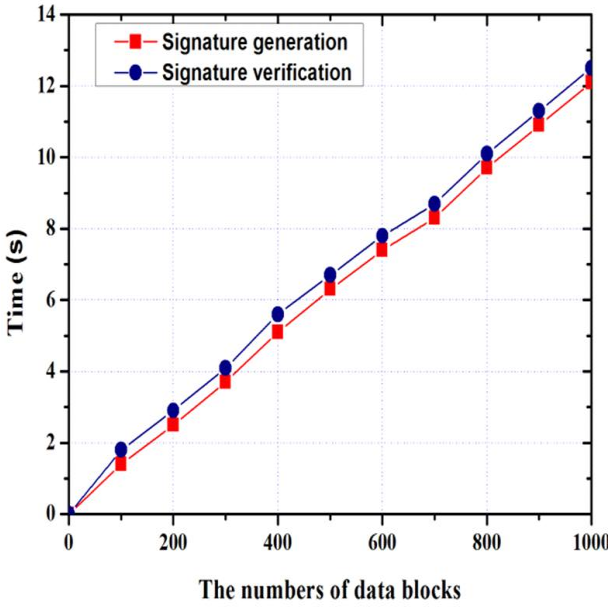

Fig.3 The computation overhead in the process of signature generation and signature verification

ISSN: 2366-1313

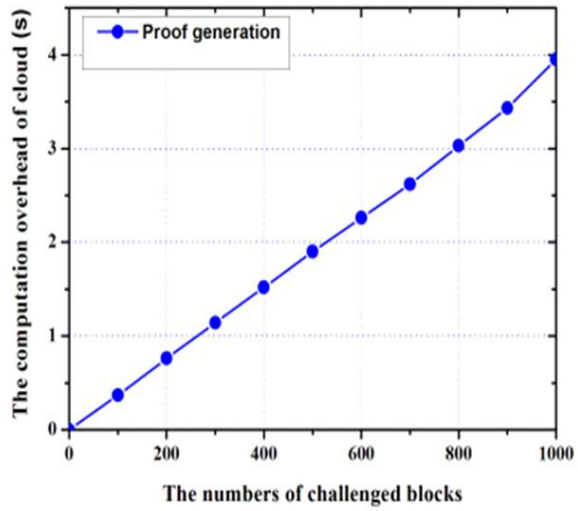

Fig.4 The computation overhead of the cloud in the phase of integrity auditing

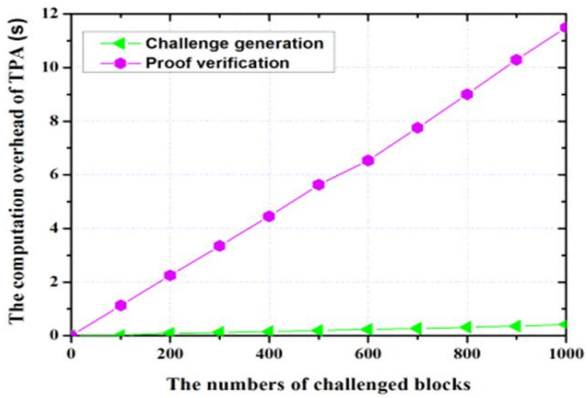

Fig.5 The computation overhead of the TPA in the phase of integrity auditing

VI. CONCLUSION

In this paper, we proposed an identity-based data integrity auditing scheme for secure cloud storage, which supports data sharing with sensitive information hiding. In our scheme, the file stored in the cloud can be shared and used by others on the condition that the sensitive information of the file is protected. Besides, the remote data integrity auditing is still able to be efficiently executed. The security proof and the experimental analysis demonstrate that the proposed scheme achieves desirable security and efficiency.

REFERENCES

[1] K. Ren, C. Wang, and Q. Wang, "Security challenges for the public cloud," IEEE Internet Computing, vol. 16, no. 1, pp. 69–73, Jan 2012.

[2] G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z. Peterson, and D. Song, "Provable data possession at untrusted stores," in Proceedings of the 14th ACM Conference on Computer and Communications Security, ser. CCS '07, 2007, pp. 598–609.

[3] A. Juels and B. S. Kaliski, "Pors: Proofs of retrievability for large files," in Proceedings of the 14th ACM Conference on Computer and Communications Security, ser. CCS '07, 2007, pp. 584–597.

[4] H. Shacham and B. Waters, "Compact proofs of retrievability," J. Cryptology, 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 Transactions on Computers, vol. 62, no. 2, pp. 362–375, 2013.

[6] S. G. Worku, C. Xu, J. Zhao, and X. He, "Secure and efficient privacy-preserving public auditing scheme for cloud storage," Comput. Electr. Eng., vol. 40, no. 5, pp. 1703–1713, Jul. 2014.

[7] C. Guan, K. Ren, F. Zhang, F. Kerschbaum, and J. Yu, "Symmetric-key based proofs of retrievability supporting public verification," in Computer Security – ESORICS 2015. Cham: Springer International Publishing, 2015, pp. 203–223.

[8] W. Shen, J. Yu, H. Xia, H. Zhang, X. Lu, and R. Hao, "Light-weight and privacy-preserving secure cloud auditing scheme for group users via the third party medium," Journal of Network and Computer Applications, vol. 82, pp. 56–64, 2017.

[9] J. Sun and Y. Fang, "Cross-domain data sharing in distributed electronic health record systems," IEEE Transactions on Parallel and Distributed Systems, vol. 21, no. 6, pp. 754–764, June 2010.

[10] G. Ateniese, R. D. 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, 2008, pp. 1–10.

[11] C. Erway, C. Papamanthou, and R. Tamassia, "Dynamic provable data possession," in ACM Conference on Computer and Communications Security, 2009, pp. 213–222.

[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 Transactions on Parallel and Distributed Systems, vol. 22, no. 5, pp. 847–859, May 2011.