International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Index Copernicus Value (2015): 78.96 | Impact Factor (2015): 6.391

Survey On: Auditing and Resisting Key Exposure on Cloud Storage

Akshata Bhand[1], D. A. Meshram[2]

[1, 2]ME (Information Technology), RMD Sinhgad School of Engineering, Pune

Abstract: ***A Cloud capacity reviewing is seen as a basic administration to verify the veracity of the information out in the open cloud. Existing examining conventions are altogether in light of the supposition that the Client's mystery key for examining is totally secured. Such supposition may not generally be held, due to the likely frail suspicion that all is well and good and additionally low security settings at the customer. In a large portion of the current evaluating conventions would unavoidably get to be distinctly not able to work when a mystery key for evaluating is uncovered. It is explored on the best way to decrease the harm of the customer's key disclosure in distributed storage evaluating, and give the main helpful illustration to this new issue setting. Formalized the definition and the security model of inspecting convention with key-presentation strength and propose such a convention. Used and built up a novel authenticator development to bolster the forward security and the property of piece less undeniable nature utilizing the current plan. The security verification and the execution investigation appear that the anticipated convention is secured and efficient.***

Keywords: Cloud storage auditing, homomorphism linear authenticator, cloud computation, key exposure resistance

1. Introduction

Cloud storage auditing is used to verify the integrity of the data stored in public cloud, which is one of the important security techniques in cloud storage. In recent years, auditing protocols for cloud storage have attracted much attention and have been researched intensively [1]. These protocols focus on several different aspects of auditing, and how to achieve high bandwidth and computation efficiency is one of the essential concerns [3]. For that purpose, the Homomorphism Linear Authenticator (HLA) technique that supports block less verification is explored to reduce the overheads of computation and communication in auditing protocols, which allows the auditor to verify the integrity of the data in cloud without retrieving the whole data.

Many cloud storage auditing protocols like have been proposed based on this technique [1]-[8]. The privacy protection of data is also an important aspect of cloud storage auditing. In order to reduce the computational burden of the client, a third-party auditor (TPA) is introduced to help the client to periodically check the integrity of the data in cloud. However, it is possible for the TPA to get the client"s data after it executes the auditing protocol multiple times [3]. Auditing protocols are designed to ensure the privacy of the client"s data in cloud. Another aspect having been addressed in cloud storage auditing is how to support data dynamic operations [9].

Key exposure could happen due to several reasons:

1) Key management- Key management is a process which is done by the client. In case any fault occurs and if the client is using a cheap software-based key management, then key exposure is possible.
2) Internet based security attacks- Suppose if a client downloads any data or file and if that it contains malicious program, then it may infect the system. This allows the hackers to easily access any confidential data [4].
3) Trading with hackers- It can happen that cloud also earns incentives by trading with the concerned hackers. In this process, the cloud can get the client's data and forge the authenticator by regenerating false data or by hiding data loss. Thus, dealing with key exposure is a vital issue in cloud storage and various methodologies were adopted.

2. Literature Survey

1. Enabling Cloud Storage Auditing With Key-Exposure Resistance

Authors- Jia Yu, Kui Ren, Cong Wang and Vijay Varadharajan

In this paper deal with the client's key exposure in cloud storage auditing. Author propose a new paradigm called auditing protocol with key-exposure resilience. In such a protocol, the integrity of the data previously stored in cloud can still be verified even if the client's current secret key for cloud storage auditing is exposed. Formalize the definition and the security model of auditing protocol with key-exposure resilience, and then propose the first practical solution. The security proof and the asymptotic performance evaluation show that the proposed protocol is secure and efficient [1].

2."Enhancing Data Security In Cloud Storage Auditing With Key Abstraction"

Authors- Priyadharshni, and Geo Jenefer. G

In this paper two basic solutions for the key-exposure problem of cloud storage auditing is discussed and implemented. The first is a naive solution, which in fact cannot fundamentally solve this problem. The second is a slightly better solution, which can solve this problem but has a large overhead. They are both impractical when applied in realistic settings. And then core protocol that is much more efficient than both of the basic solutions [2] .

3."An Efficient Cloud Storage Batch Auditing Without Key Exposure Resistance Using Public Verifier"

Authors- T Yawaikha, R Meyanand

Paper presents study on how to deal with the client's key without exposing into the cloud. The auditing performed by public verifier not only audits the data but also verifies the integrity of the data in cloud. The concept of user revocation allows to revoke the invalid key registered. Formalize the definition and the security model of auditing protocol without key-exposure resilience, and then propose and verify the first practical solution [3]

4."Survey Paper On Cloud Storage Auditing With Exposure Resistance"
Authors- Sneha Singha, S. D. Satav
As this complete paper narrates the different methodologies on enabling cloud storage auditing with key exposure resilience, but none of the methodologies seems to be perfect. So, this survey paper as a bit proposes a method of an effective key exposure resistance where we adopt the deduplication strategy of data. Moreover, it will check the duplicacy of data and eliminate the redundant one using MD5 hashing algorithm. After the public and private keys are generated, it uses tile bitmap method wherein it will check the previous and the current versions of the data to ease the auditor's workload and to make the system more efficient [4]

5. "Efficient provable data possession for hybrid clouds"
Authors-Y. Zhu, H. Wang, Z. Hu, G.-J. Ahn, H. Hu, and S. S. Yau
This paper addressed the construction of PDP scheme for hybrid clouds. Based on homomorphic verifiable responses and hash index hierarchy, Author proposed a cooperative PDP scheme to support dynamic scalability on multiple storage servers. Experiments showed that our schemes require a small, constant amount of overhead [5] .

3. Proposed System

At a high level, our setting of interest is an enterprise network, consisting of a group of affiliated clients (for example, employees of a company) who will use the SCSP and store data with deduplication technique. In deduplication can be frequently used in these settings for data backup and disaster recovery applications while greatly reducing storage space. Such systems are widespread and are often more suitable, in to user file backup and synchronization applications than richer storage abstractions [14]. There are three entities defined in our system, that is, users, private cloud and S-CSP in public cloud. The S-CSP performs deduplication by checking if the contents of two files are the same and stores only one of them. The access right to a file is defined based on a set of privileges.

4. System Architecture

Cloud data storage service includes the user(U), who has the large data to be stored in cloud; the cloud server(CS), managed by cloud service provider(CSP) with significant storage; the third party auditor(TPA), trusted to access the CSP according to users request. When user store the data, the copy is sent to both the CSP and TPA. To verify the correctness of data stored in cloud, auditing process is done.

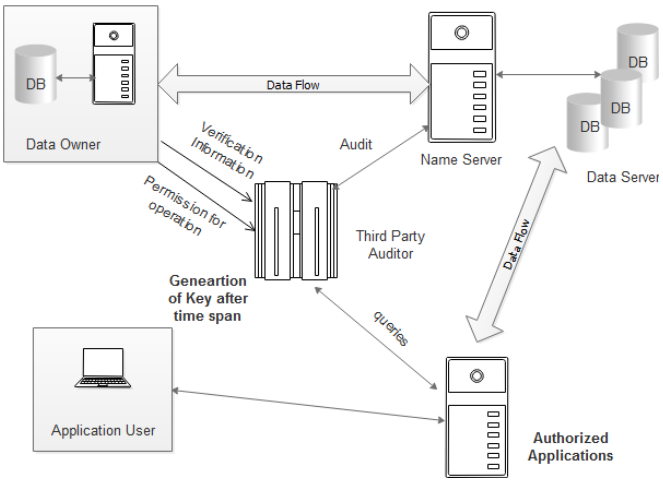

Figure 1: System Architecture

Here the auditing process is carried out TPA, it must efficiently audit without bringing any changes to the original data. For auditing, the data which is in TPA is used. Public auditability: Allow the TPA to verify the correctness of data without demanding the copy of data. Privacy preserving: To ensure that TPA cannot retrieve the data content during the auditing process. Lightweight: To allow TPA to perform auditing with minimum communication and computation overhead.

5. Algorithm

An auditing protocol with key-exposure resilience is composed by five algorithms (SysSetup,

KeyUpdate, AuthGen, ProofGen, ProofVerify), shown below:

5.1 SysSetup(1k, T) →(PK, SK0):

The system setup algorithm is a probabilistic algorithm which takes as input a security parameter *k* and the total number of time periods *T*, and generates a public key *PK* and the initial client"s secret key *SK*0. This algorithm is run by the client.

5.2 Key Update (PK, j, SK j) → (SK j+1):

The key update algorithm is a probabilistic algorithm which takes as input the public key *PK*, the current period *j* and a client"s secret key *SK j* , and generates a new secret key *SK j*+1 for the next period *j* + 1. This algorithm is run by the client.

5.3 Auth Gen (PK, j, SK j , F) → (_):

The authenticator generation algorithm is a probabilistic algorithm which takes as input the public key *PK*, the current period *j* , a client"s secret key *SK j* and a file *F*, and generates the set of authenticators _ for *F* in time period *j*. This algorithm is also run by the client.

5.4 Proof Gen (PK, j,Chal, F,_) → (P):

The proof generation algorithm is a probabilistic algorithm which takes as input the public key *PK*, a time period *j* , a challenge *Chal*, a file *F* and the set of authenticators _, and generates a proof *P* which means the cloud possesses *F*. Here, *(j,Chal)* pair is issued by the auditor, and then used by the cloud. This algorithm is run by the cloud.

5.5 Proof Veri f y(PK, j,Chal, P) → ("True" or "False")

The proof verifying algorithm is a deterministic algorithm which takes as input the public key *PK*, a time period *j* , a challenge *Chal* and a proof *P*, and returns "*True*" or "*False*". This algorithm is run by the client**.**

Project Flow

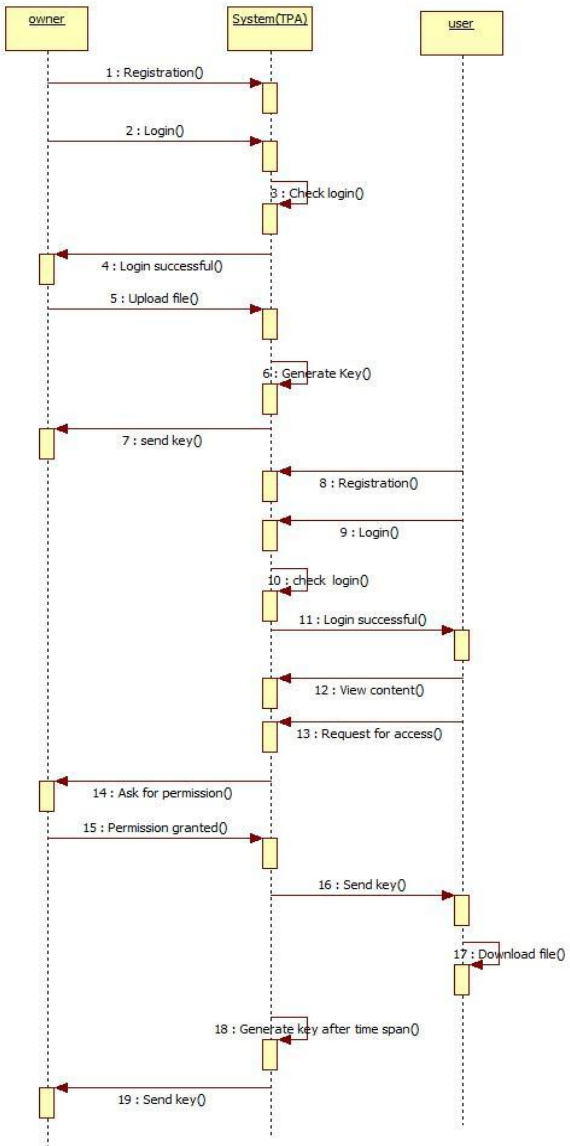

Figure 2: Sequence Diagram

6. Conclusion

We examine on the best way to manage the client"s enter introduction in distributed storage evaluating. We propose another worldview called reviewing convention with key-presentation flexibility. In such a convention, the uprightness of the information beforehand put away in cloud can at present be confirmed regardless of the possibility that the client"s current mystery key for distributed storage inspecting is uncovered. We formalize the definition and the security model of reviewing convention with key-presentation versatility, and after that propose the principal down to earth arrangement. The security confirmation and the asymptotic execution assessment demonstrate that the proposed convention is secure and proficient.

References

[1] Vijay varadhanajan , Jia Yu ,Kai Ren "*Enabling Cloud Storage Auditing With Key Exposure Resistance*" IEEE Transcation on information forensics and security,Vol 10.

[2] Priyadharshni, Geo Jenefer. G " *Enhancing Data Security In Cloud Storage Auditing With Key Abstraction*" Vo.2,Issue 2,Oct 2015.

[3] T Yawaikha,R Meyanand, " *An Efficient Cloud Storage Batch Auditing Without Key Exposure Resistance Using Public Verifier*" International conference on system 2016

[4] Sneha Singha"*Survey Paper On Cloud Storage Auditing With Exposure Resistance*" IJSR

[5] Y. Zhu, H. Wang, Z. Hu, G.-J. Ahn, H. Hu, and S. S. Yau, "Efficient provable data possession for hybrid clouds," in *Proc. 17th ACM Conf. Comput. Commun. Secur.*, 2010, pp. 756–758.

[6] K. Yang and X. Jia, "Data storage auditing service in cloud computing: Challenges, methods and opportunities," *World Wide Web*, vol. 15, no. 4, pp. 409–428, 2012.

[7] H. Shacham and B. Waters, "Compact proofs of retrievability," in *Advances in Cryptology—ASIACRYPT. Berlin, Germany:* Springer-Verlag, 2008, pp. 90–107.

[8] C. Wang, K. Ren, W. Lou, and J. Li, "Toward publicly auditable secure cloud data storage services," *IEEE Netw., vol. 24, no. 4, pp. 19–24,* Jul./Aug. 2010.

[9] K. Yang and X. Jia, "An efficient and secure dynamic auditing protocol for data storage in cloud computing," *IEEE Trans. Parallel Distrib. Syst.*, mvol. 24, no. 9, pp. 1717–1726, Sep. 2013.

[10] C. Wang, S. S. M. Chow, Q. Wang, K. Ren, and W. Lou, "Privacypreserving public auditing for secure cloud storage," *IEEE Trans. Comput.*, vol. 62, no. 2, pp. 362–375, Feb. 2013.