RESEARCH ARTICLE OPEN ACCESS

Auditing and De-Duplication in Cloud Computing

E Lokesh Kumar [1], Krishna Narayan P [2], P.Renuka Devi [3]
Student [1] & [2], Assistant Professor [3]
Department of Computer Science and Engineering
SRM University, Kattankulathur
Tamil Nadu - India

ABSTRACT
As the cloud computing technology develops during the last decade, outsourcing data to cloud service for storage becomes an attractive trend, which benefits in sparing efforts on heavy data maintenance and management. Nevertheless, since the outsourced cloud storage is not fully trustworthy, it raises security concerns on how to realize data de-duplication in cloud while achieving integrity auditing. In this work, we study the problem of integrity auditing and secure de-duplication on cloud data. Specifically, aiming at achieving both data integrity and de-duplication in cloud, we propose two secure systems, namely XCloud and XCloud+. XCloud introduces an auditing entity with maintenance of a cloud, which helps clients generate data tags before uploading as well as audit the integrity of data having been stored in cloud. Compared with previous work, the computation by user in XCloud is greatly reduced during the file uploading and auditing phases. X Cloud+ is designed motivated by the fact that customers always want to encrypt their data before uploading, and enables integrity auditing and secure de-duplication on encrypted data.

Keywords:- Cloud Computing, De-Duplication, Integrity, Auditing

I. INTRODUCTION

Cloud storage is a model of networked enterprise storage where data is stored in virtualized pools of storage which are generally hosted by third parties. Cloud storage provides customers with benefits, Ranging from cost saving and simplified convenience, to mobility opportunities and scalable service. These great features attract more and more customers to utilize and storage their personal data to the cloud storage: according to the analysis report, the volume of data in cloud is expected to achieve 40 trillion gigabytes in 2020. Even though cloud storage system has been widely adopted, it fails to accommodate some important emerging needs such as the abilities of auditing integrity of cloud files by cloud clients and detecting duplicated files by cloud servers. We illustrate both problems below. The first problem is integrity auditing. The cloud server is able to relieve clients from the heavy burden of storage management and maintenance. The most difference of cloud storage from traditional in-house storage is that the data is transferred via Internet and stored in an uncertain domain, not under control of the clients at all, which inevitably raises clients great concerns on the integrity of their data. These concerns originate from the fact that the cloud storage is susceptible to security threats from both outside and inside of the cloud , and the uncontrolled cloud servers may passively hide some data loss incidents from the clients to maintain their reputation. What is more serious is that for saving money and space, the cloud servers might even actively and deliberately discard rarely accessed data files belonging to an ordinary client. Considering the large size of the outsourced data files and the clients' constrained resource capabilities, the first problem is generalized as how can the client efficiently perform periodical integrity verifications even without the local copy of data files.

The second problem is secure de-duplication. The rapid adoption of cloud services is accompanied by increasing volumes of data stored at remote cloud servers. Among these remote stored files, most of them are duplicated: according to a recent survey by EMC, 75% of recent digital data is duplicated copies. This fact raises a technology namely de-duplication, in which the cloud servers would like to de-duplicate by keeping only a single copy for each file (or block) and make a link to the file (or block) for every client who owns or asks to store the same file (or block). Unfortunately, this action of de-duplication would lead to a number of threats potentially affecting the storage system, for

example, a server telling a client that it (i.e., the client) does not need to send the file reveals that some other client has the exact same file, which could be sensitive sometimes. These attacks originate from the reason that the proof that the client owns a given file (or block of data) is solely based on a static, short value (in most cases the hash of the file). Thus, the second problem is generalized as how can the cloud servers efficiently confirm that the client (with a certain degree assurance) owns the uploaded file (or block) before creating a link to this file (or block) for him/her.

II. PROBLEM STATEMENT

The problem is to solve the issue of duplication of data in the cloud as well as to maintain the integrity of data stored in the cloud thus simplifying the heavy burden of storage management and maintenance of data.

III. LITERATURE SURVEY

Several papers have been published in rated to providing security as well as for proper utilisation of storage space available in cloud .

One of such paper published by Zheng Yan and her colleagues deal with similar problem of proper utilisation of storage space available in the cloud. In system proposed by them makes use of an active cloud storage system where multiple users can register by creating their own accounts. The cloud storage system provide them space where the can store their data files. Any number of users can register in that system. User can only login using his or her login credentials. The whole system is monitored by an admin. Any other user who wants to access a particular file was first required to take permission from the data owner. The data owner would then issue him the permission keys that were required to access the files. The main disadvantage in this system was that data owner was needed to provide permission key every time to the requesting owner thus compromising the security of the data files stored. Furthermore there was no proper mechanism to protect integrity of the data file stored. Another mechanism that was proposed in relation to this topic was Message-Locked Encryption. In this mechanism the key under which encryption and decryption are performed is itself derived from the message. MLE provides a way to achieve secure de-duplication (space-efficient secure outsourced storage), a goal currently targeted by numerous cloud-storage providers. This provides definitions both for privacy and for a form of integrity that we call tag consistency. Based on this foundation, we make both practical and theoretical contributions. On the practical side, we provide ROM security analyses of a natural family of MLE schemes that includes deployed schemes. On the theoretical side the challenge is standard model solutions, and we make connections with deterministic encryption, hash functions secure on correlated inputs and the sample-then-extract paradigm to deliver schemes under different assumptions and for different classes of message sources.

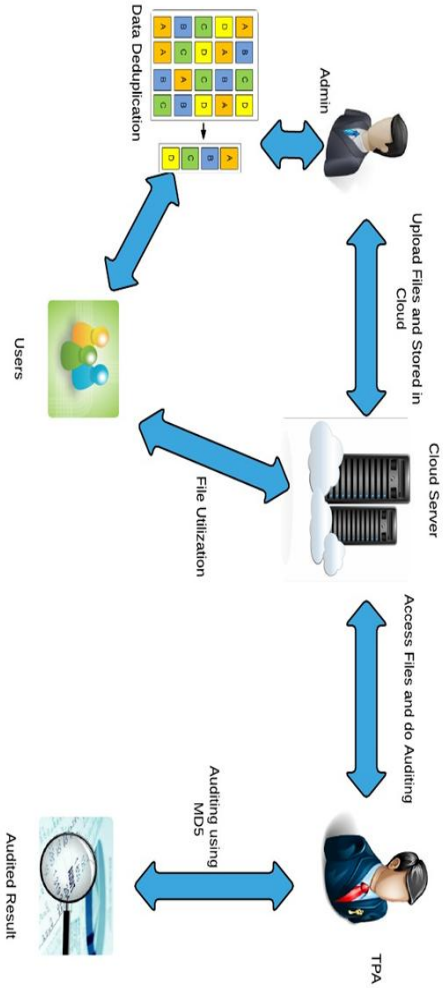
Similarly other papers have been published related to providing security in the cloud. One such paper published in 2014 deals with the topic of preserving privacy as well as enabling public auditing in the clouds. With cloud data services, it is commonplace for data to be not only stored in the cloud, but also shared across multiple users. Unfortunately, the integrity of cloud data is subject to scepticism due to the existence of hardware/software failures and human errors. Several mechanisms have been designed to allow both data owners and public verifiers to efficiently audit cloud data integrity without retrieving the entire data from the cloud server. However, public auditing on the integrity of shared data with these existing mechanisms will inevitably reveal confidential information—identity privacy—to public verifiers. They proposed a novel privacy-preserving mechanism that supports public auditing on shared data stored in the cloud. In particular, we exploit ring signatures to compute verification metadata needed to audit the correctness of shared data. With our mechanism, the identity of the signer on each block in shared data is kept private from public verifiers, who are able to efficiently verify shared data integrity without retrieving the entire file. In addition, our mechanism is able to perform multiple auditing tasks simultaneously instead of verifying them one by one.

A similar paper was published in 2013 which also dealt with the topic of preserving data integrity in the cloud. Using Cloud Storage, users can remotely store their data and enjoy the on-demand high quality applications and services from a shared pool of configurable computing resources, without the burden of local data storage and maintenance. However, the fact that users no longer have physical possession of the outsourced data makes the data

integrity protection in Cloud Computing a formidable task, especially for users with constrained computing resources. Moreover, users should be able to just use the cloud storage as if it is local, without worrying about the need to verify its integrity. Thus, enabling public audit ability for cloud storage is of critical importance so that users can resort to a third party auditor (TPA) to check the integrity of outsourced data and be worry-free. To securely introduce an effective TPA, the auditing process should bring in no new vulnerabilities towards user data privacy, and introduce no additional online burden to user. In this paper, we propose a secure cloud storage system supporting privacy-preserving public auditing. We further extend our result to enable the TPA to perform audits for multiple users simultaneously and efficiently. Extensive security and performance analysis show the proposed schemes are provably secure and highly efficient.

Another paper published in relation to this topic introduced twin cloud architecture. Cloud computing promises a more cost effective enabling technology to outsource storage and computations. Existing approaches for secure outsourcing of data and arbitrary computations are either based on a single tamper-proof hardware, or based on recently proposed fully holomorphic encryption. The hardware based solutions are not scalable, and fully hemimorphic encryption is currently only of theoretical interest and very inefficient. this paper we propose architecture for secure outsourcing of data and arbitrary computations to an untrusted commodity cloud. In our approach, the user communicates with a trusted cloud (either a private cloud or built from multiple secure hardware modules) which encrypts and verifies the data stored and operations performed in the cloud.

IV. SYSTEM ARCHITECTURE



PROPOSED ARCHITECTURE

V. OTHER PAPER COMPARISION

Sr. No.	Paper Name	Technique	Advantages	Disadvantages	Results
1	Encrypted Data Management with De-duplication in Cloud Computing	Encrypted data management in cloud	Deduplication can be prevented	Sensitive information can be revealed, key has to be issued every time for data usage.	The results showed whetheher the data is already in the cloud or not and whether that user has permission to access the file.
2	Privacy Preserving Public Auditing of Cloud Data	Public Auditing	Maintaing Data integrity	Tough to implement	The result displays that Data stored in cloud is fully secured and integrity is restored.
3	Message Locked Encryption	Message Locked Encryption System	Key is derived from message itself.	No method to maintain data integrity	The results showed that data is encrypted and key can be derived from message itself

VI. PROPOSED SYSTEM

Our system introduces advanced de-duplication Techniques. Every new user has to sign up first.User will then proceed to upload a file. Before uploading a file in the cloud user will first check Whether the file is already been uploaded or not. If not user can upload the file. Each time a user uploads a new file it is first encrypted and it's encrypted copy is stored at the server. This encrypted copy is then used to check whether a file is already in the cloud or not. Each time a user tries to upload a file first it is checked whether the file is already uploaded in cloud or not. The files are encrupted usind convergent algorithm. We use MD5 algoritm to check the integrity of data. MD5 is an algorithm based on no. of bits of data a File should contain.

Public verifier is able to audit data without Retrieving the full file.

VII. ADVANTAGE

There are many advantages of developing such system. Some of these advantages are listed below. They are:

- security issue will not be there.
- More security is provided for files stored In the cloud and better auditing mechanism is used for checking integrity of data.
- Faster recoveries

VIII. SYSTEM MODULES

There are mainly two different modules that are being used in this system :

1. Detecting Duplication :
Users will have to first check for duplicate files before uploading files in the cloud. Only after verifying that a given file is not in the cloud user can upload that file.

2. Auditing:
Integrity auditing of the data stored in the cloud storage will be done by public auditors with the help of MD5 algorithm.

IX. CONCLUSION

Hence we propose a system that not only addresses the current problem of de-duplication but also provides a secure method of auditing the data stored in the cloud.

REFERNCES

[1] M. Bellare, S. Keelveedhi, and T. Ristenpart, "Message-Locked Encryption and Secure Deduplication," Advances in Cryptology (EUROCRYPT 13), LNCS 7881, 2013, pp. 296–312.

[2] Zheng Yan, Mingjun Wang, and Yuxiang Li, Athanasios V. Vasilakos,"Encrypted Data Management with De-Duplication in Cloud Computing" 2325-6095/16 IEEE CLOUD COMPUTING , IEEE 2016.

[3] M. Bellare, S. Keelveedhi, and T. Ristenpart, "DupLESS: Server-Aided Encryption for Deduplicated Storage,"
Proc. 22nd Usenix Conf. Security, 2013, pp. 179–194.

[4] Z.C. Wen et al., "A Verifiable Data Deduplication Scheme in Cloud Computing," Proc. Int'l Conf. Intelligent Networking and Collaborative Systems, 2014, pp. 85–90.

[5] J. Li et al., "A Hybrid Cloud Approach for Secure Authorized Deduplication," IEEE Trans. Parallel Distributed
Systems, vol. 26, no. 5, 2015, pp. 1206–1216.

[6] D.T. Meyer and W.J. Bolosky, "A Study of Practical Deduplication," ACM Trans. Storage, vol. 7, no. 4, 2012, pp. 1–20.

[7] Z. Yan, W. Ding, and H. Zhu, "Manage Encrypted Data Storage with Deduplication in Cloud," Proc. Int'l Conf. Algorithms and Architectures for Parallel Processing (ICA3PP), 2015, pp. 547–561.

[8] P. Puzio et al., "ClouDedup: Secure Deduplication with Encrypted Data for Cloud Storage," Proc. IEEE 5th Int'l Conf. Cloud Computing Technology and Science, 2013, pp. 363–370.

[9] P. Meye et al., "A Secure Two-Phase Data Deduplication Scheme," Proc. IEEE 6th Int'l Symp. Cyberspace Safety and Safety and Security, IEEE 11th Int'l Conference,2014.