IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 5, Issue 3, March 2016

Secure Data Deduplication in Hybrid Cloud

Krishnapraseeda V[1], Karthikeyan C[2]

PG Scholar M.E, Dept of Computer Science and Engineering, United Institute of Technology, Coimbatore, India[1]

Assistant Professor, Dept of Computer Science and Engineering, United Institute of Technology, Coimbatore, India[2]

Abstract: The cloud computing technology has developed during the past decade; in which data outsourced to cloud service for storage has become an attractive trend, which benefits in sparing efforts on heavy data maintenance and management. Also the security for the cloud is still a big task. However, the cloud storage of outsourced data cannot be trusted fully and security concerns are raised on how the data deduplication in cloud achieves integrity audit. In our work, we deal with the issue of secure integrity audit and data deduplication on cloud storage. Specifically, we aim at achieving both cloud data integrity and deduplication, while we propose two secure systems namely SecCloud and SecCloud+, also the process of sending and receiving in case of secure by means of same file content with same file name. SecCloud enables an auditing scheme with the application of a MapReduce cloud, where the clients are responsible to generate data tags before uploading data and audit the integrity which has been stored in cloud as well. Compared to the related works, the computation of integrity auditing of user's file in SecCloud is highly reduced. SecCloud+ ensures that the files are encrypted before uploading, which provides secure data integrity and encrypted data deduplication.

Index terms: Cloud Computing, Deduplication, Integrity, SecCloud, MapReduce, Auditing.

I. INTRODUCTION

Cloud computing is the term used to share the resources globally with less cost. We can also call as "IT ON DEMAND". It provides three types of services i.e., Infrastructure as a service (IaaS), Platform as a service (PaaS) and Software as a service (SaaS). The processors, together with the Software as a service (SaaS) architecture, transforms data centres into pools of computing service on a huge scale and are cheaper and more powerful. The users access the cloud applications through the web browsers having internet connection. Moving files to cloud brings convenience and reduces manage hardware complexities. The Cloud data are maintained by Cloud service providers (CSP) with various incentives different levels of services.

A. Architecture of Cloud Computing

End users access the cloud based applications through the web browsers with internet connection. Moving- files to cloud brings convenience and reduces to manage hardware complexities.

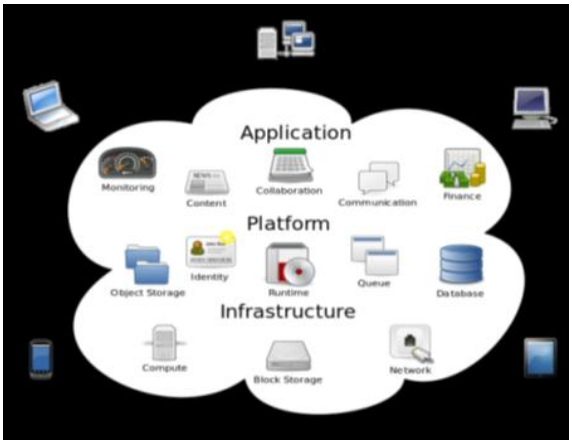

Fig1. Architecture of Cloud Computing

The Cloud data are maintained by Cloud service providers (CSP) with various incentives for different levels of services. However it eliminates the responsibility of local machines to maintain data, there is a chance to lose data or it effects from external or internal attacks.

To maintain the integrity and availability of data, many authors proposed several algorithmic rules and methods that enable the data correctness and verification. Cloud servers not only stores data like a data ware house , it also provides persistent data updates with the help of users using various operations like insert, delete , update and append.

B. Data Deduplication

In Cloud Computing, data deduplication is a differentiated data compression technique for simplifying duplicate copies of persistent data.. This method is applied to put right the storage exploitation and can be practiced to network transition of data for reduction of number of bytes that must be transferred. In this deduplication process, incomparable data chunks, or patterns, are detected and stored during technical analysis process. While the case study continues, the extra chunks are examined to already stored copy and when a match is found, the unnecessary chunk is substituted with a reference which directs to the stored chunk. Acknowledged that the same pattern of bytes might proceed dozens, hundreds, or thousands of times (the frequency of match depends on chunk size), the amount of data that must be stored or sent preserves to be greatly reduced.

In a simple form, deduplication encloses on the file level; ie., it knocks out duplicate copies of same or similar file. This kind of method is sometimes called File-level deduplication or Single Instance Storage (SIS).

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 5, Issue 3, March 2016

Deduplication also takes place at block level, eliminating a duplicated data block that occurs in non-identical files. Block-level deduplication clears up more space when compared to SIS, and a particular form known as variable block or variable length deduplication has become very well-known. Frequently the phrase "data deduplication" is utilized as a synonym for block-level deduplication.

C. Deduplication Vs Compression

Deduplication is sometimes confused with compression, another technique for reducing storage requirements. While deduplication eliminates redundant data, compression uses algorithms to save data more concisely. Some compression is lossless, meaning that no data is lost in the process, but "lossy" compression, which is frequently used with audio and video files, actually deletes some of the less-important data included in a file in order to save space. By contrast, deduplication only eliminates extra copies of data; none of the original data is lost. Also, compression doesn't get rid of duplicated data.

D. In-line deduplication

This is the process where the deduplication hash calculations are created on the target device as the data enters the device in real time. If the device spots a block that it already stored on the system it does not store the new block, just references to the existing block. The benefit of in-line deduplication over post-process deduplication is that it requires less storage as data is not duplicated. On the negative side, it is frequently argued that because hash calculations and lookups takes so long, it can mean that the data ingestion can be slower thereby reducing the backup throughput of the device. However, certain vendors with in-line deduplication have demonstrated equipment with similar performance to their post-process deduplication counterparts. Post-process and in-line deduplication methods are often heavily debated.

II. RELATED WORK

SecCloud introduces an auditing entity with maintenance of a Map Reduce cloud, which helps clients generate data tags before uploading as well as audit the integrity of data having been stored in cloud. The proposed SecCloud system has achieved both integrity auditing and files deduplication. Specially, there exists many cloud techniques that were designed for cloud computing environment. For example, Giuseppe Ateniese constructed a highly efficient and provably secure PDP technique based entirely on symmetric key cryptography, while not requiring any bulk encryption. Also, in contrast with its predecessors, our PDP technique allows outsourcing of dynamic data, i.e, it efficiently supports operations, such as block modification, deletion and append [4]. The central goal in PDP is to allow a client to efficiently, frequently and securely verify that a server – who purportedly stores client's potentially very large amount of data – is not cheating the client [4]. Qian Wang describes the problem of ensuring the integrity of data storage in Cloud Computing. In particular, he considered the task of allowing a third party auditor (TPA), on behalf of the cloud client, to verify the integrity of the dynamic data stored in the cloud [8]. The introduction of TPA eliminates the involvement of client through the auditing of whether his data stored in the cloud is indeed intact, which can be important in achieving economies of scale for Cloud Computing [5]. Roberto Di Pietro introduced a novel Proof of Ownership (POW) scheme that has all features of the state-of-the-art solution while incurring only a fraction of the overhead experienced by the competitor; second, the security of the proposed mechanisms relies on information theoretical (combinatoric) rather than computational assumptions [11]. Yan Zhu presented a cooperative PDP (CPDP) scheme based on homomorphic verifiable response and hash index hierarchy [2]. They also prove the security of our scheme based on multi-prover zero-knowledge proof system, which can satisfy completeness, knowledge soundness, and zero-knowledge properties [3].

In addition, they articulated performance optimization mechanisms for our scheme, and in particular present an efficient method for selecting optimal parameter values to minimize the computation costs of clients and storage service providers [5]. Michael Armbrust's main idea was to transform a large part of the IT industry, making software even more attractive as a service and shaping the way IT hardware is designed and purchased. When a Cloud is made available in a pay-as-you-go manner to the general public, we call it a Public Cloud [13]. Giuseppe Ateniese introduces a model for provable data possession (PDP) that can be used for remote data checking: A client that has stored data at an untrusted server can verify that the server possesses the original data without retrieving it [3]. In this paper, a new notion called private data deduplication protocol, a deduplication technique for private data storage is introduced and formalized by Wee Keong Ng [6]. Their view is a complement of the state-of-the-art public data deduplication protocols [7]. The main goal of Yan Kit Li in this paper is to protect the confidentiality of sensitive data while supporting deduplication, the convergent encryption technique has been proposed to encrypt the data before outsourcing [1]. To better protect data security, this paper makes the first attempt to formally address the problem of authorized data deduplication [5]. Different from traditional deduplication systems, the differential privileges of users are further considered in duplicate check besides the data itself [14].

III. SYSTEM MODEL

SecCloud introduces an auditing entity with maintenance of a Map Reduce cloud, which helps clients generate data tags before uploading as well as audit the integrity of data having been stored in cloud. This design fixes the issue of previous work that the computational load at user or auditor is too huge for tag generation. For completeness of fine-grained, the functionality of auditing designed in SecCloud is supported on both block level and sector level. In addition, SecCloud also enables secure deduplication. Notice that the "security" considered in SecCloud is the prevention of leakage of side channel information. In order to prevent the leakage of such side

channel information, we follow the tradition of and design a proof of ownership protocol between clients and cloud servers, which allows clients to prove to cloud servers that they exactly own the target data.

Disadvantages:-
One critical challenge of cloud storage services is the management of the ever-increasing volume of data. Whenever data is transformed, concerns arise about potential loss of data. By definition, data deduplication systems store data differently from how it was written.

IV. OUR CONSTRUCTION

In this paper, we solve this open problem and propose SecCloud+, which allows for integrity auditing and deduplication on encrypted files. Cloud Clients have large data files to be stored and rely on the cloud for data maintenance and computation. They can be either individual consumers or commercial organizations. Cloud Servers virtualize the resources according to the requirements of clients and expose them as storage pools. Typically, the cloud clients may buy or lease storage capacity from cloud servers, and store their individual data in these bought or rented spaces for future utilization. Auditor which helps clients upload and audit their outsourced data maintains a MapReduce cloud and acts like a certificate authority. This assumption presumes that the auditor is associated with a pair of public and private keys. Its public key is made available to the other entities in the system.

Advantages:-
- It provides the Integrity auditing by clustering the files with removing the duplicate files.
- The duplicate files are mapped with a single copy of the file by mapping with the existing file in the cloud.
- Integrity Auditing: The first design goal of this work is to provide the capability of verifying correctness of the remotely stored data.

Convergent encryption:
Convergent encryption [4], [8] provides data confidentiality in deduplication. A user (or data owner) derives a convergent key from each original data copy and encrypts the data copy with the convergent key.

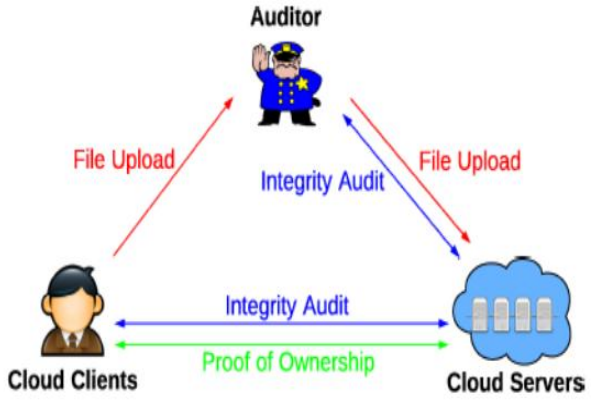

Fig2. Architecture of Convergent Encryption

In addition, the user also derives a tag for the data copy, such that the tag will be used to detect duplicates. Here, we assume that the tag correctness property [4] holds, i.e., if two data copies are the same, then their tags are the same. To detect duplicates, the user first sends the tag to the server side to check if the identical copy has been already stored. Note that both the convergent key and the tag are independently derived and the tag cannot be used to deduce the convergent key and compromise data confidentiality. Both the encrypted data copy and its corresponding tag will be stored on the server side.

Proof of ownership:
The notion of proof of ownership [8] enables users to prove their ownership of data copies to the storage server. Specifically, PoW is implemented as an interactive algorithm (denoted by PoW) run by a prover (i.e., user) and a verifier (i.e., storage server). The verifier derives a short value fðMÞ from a data copy M.

V. EXPERIMENTAL RESULT

The result focuses on the file size with content of the file so that the cloud have unique storage system and through this the efficiency of the system get enhanced and processed well in the system of cloud storage and auditing process this leads to the efficiency of the file system. To achieve both data integrity and deduplication in cloud, the proposed concept has done the file based system model. Also an auditing entity with maintenance of a MapReduce cloud, in which it helps the clients to generate data tags before uploading as well as audit the integrity of data having been stored in cloud.

VI. CONCLUSION

Thus the system has processed a Proof of Ownership protocol for preventing the leakage of side channel information in data deduplication. Compared with previous work, the computation by user is greatly reduced during the file uploading and auditing phases and also an advanced construction motivated by the fact that customers always want to encrypt their data before uploading, and allows for integrity auditing and secure deduplication directly on encrypted data.

REFERENCES

[1] Jin Li, Yan Kit Li, Xiaofeng Chen, Patrick P.C. Lee, and Wenjing Lou, "A Hybrid Cloud Approach for Secure Authorized Deduplication", IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, VOL. 26, NO. 5, MAY 2015.

[2] Yan Zhu, Hongxin Hu, Gail-Joon Ahn, Senior Member, IEEE, Mengyang Yu, "Cooperative Provable Data Possession for Integrity Verification in Multi-Cloud Storage", IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, 2012.

[3] Giuseppe Ateniese_, Randal Burns_, Reza Curtmolay, Joseph Herring_, Osama Khan_, Lea Kissnerz, "Remote Data Checking Using Provable Data Possession", Comp. Sci. Dept., Naval Postgraduate School - znpeters@nps.edu, 2008.

[4] Giuseppe Ateniese, Roberto Di Pietro, Luigi V. Mancini, Gene Tsudik, "Scalable and Efficient Provable Data Possession", IACR Cryptology Archive 01/2008; 2008:114.

[5] P. Anderson and L. Zhang, "Fast and secure laptop backups with encrypted de-duplication," in Proc. 24th Int. Conf. Large Installation Syst. Admin., 2010, pp. 29–40.

DOI 10.17148/IJARCCE.2016.53218

[6] Wee Keong Ng,Yonggang Wen, Huafei Zhu, "Private Data Deduplication Protocols in Cloud Storage", SAC'12 March 2529, 2012, Riva del Garda, Italy,Copyright 2011.
[7] M. Bellare, S. Keelveedhi, and T. Ristenpart, "Message-locked encryption and secure deduplication," in Proc. 32nd Annu. Int. Conf. Theory Appl. Cryptographic Techn., 2013, pp. 296–312.
[8] Qian Wang, Cong Wang, Jin Li, Kui Ren and Wenjing Lou, "Enabling Public Verifiability and Data Dynamics for Storage Security in Cloud Computing", Springer-Verlag, 2008, pp. 90–107.
[9] M. Bellare, C. Namprempre, and G. Neven, "Security proofs for identity-based identification and signature schemes," J. Cryptol., vol. 22, no. 1, pp. 1–61, 2009.
[10] S. Bugiel, S. Nurnberger, A. Sadeghi, and T. Schneider, "Twin clouds: An architecture for secure cloud computing," in Proc. Workshop Cryptography Security Clouds, 2011, pp. 32–44.
[11] Roberto Di Pietro, Alessandro Sorniotti, "Boosting Efficiency and Security in Proof of Ownership for Deduplication", http://www.researchgate.net/publication/268016221, MAY 2012.
[12] J. R. Douceur, A. Adya, W. J. Bolosky, D. Simon, and M. Theimer, "Reclaiming space from duplicate files in a serverless distributed file system," in Proc. Int. Conf. Distrib. Comput. Syst., 2002, pp. 617–624.
[13] Michael Armbrust, Armando Fox, Rean Griffith, Anthony D. Joseph, Randy Katz, "Above the Clouds: A Berkeley View of Cloud Computing", UC Berkeley Reliable Adaptive Distributed Systems Laboratory _ http://radlab.cs.berkeley.edu/ February 10, 2009.
[14] S. Halevi, D. Harnik, B. Pinkas, and A. Shulman-Peleg, "Proofs of ownership in remote storage systems," in Proc. ACM Conf. Comput. Commun. Security, 2011, pp. 491–500.
[15] J. Li, X. Chen, M. Li, J. Li, P. Lee, and W. Lou, "Secure deduplication with efficient and reliable convergent key management," in Proc. IEEE Trans. Parallel Distrib. Syst., http:// doi.ieeecomputersociety.org/10.1109/TPDS.2013.284, 2013.
[16] M. W. Storer, K. Greenan, D. D. E. Long, and E. L. Miller, "Secure data deduplication," in Proc. 4th ACM Int. Workshop Storage Security Survivability, 2008, pp. 1–10.

DOI 10.17148/IJARCCE.2016.53218