ISSN 2395-1621

A Review on a Novel Approach for Secure Authorized Deduplication Using Hybrid Cloud

[#1]Akash Dodke
[1]akashdodke@gmail.com
[#1]Computer Engineering, SP Pune University India

ABSTRACT

A hybrid cloud architecture is combination of a public cloud and a private cloud bound together by either standardized or proprietary technology that enables data and application portability. The data owners only outsource their data storage by utilizing public cloud while the data operation is managed in private cloud. To make data management scalable in cloud computing, deduplication has been a very well-known technique recently is use. Deduplication reduces your bandwidth requirements, speeds up the data transfers, and it keeps your cloud storage needs to a minimum. Data deduplication is one of important data compression techniques for eliminating duplicate copies of repeating data, and has been widely used in cloud storage to reduce the amount of storage space and save bandwidth. Proposed system presents an advanced scheme to support stronger security by encrypting the file with differential privilege keys. In this way, the users without corresponding privileges cannot perform the duplicate check. To maintain the confidentiality of data the convergent encryption technique has been used to encrypt the data before outsourcing. Authorized deduplication system support differential authorization duplicate check. Different from traditional deduplication systems, the differential privileges of users are further considered in duplicate check besides the data itself and Backup server.

Keywords—Deduplication, authorized duplicate check, confidentiality, hybrid cloud

ARTICLE INFO

Article History

Received :28th December 2015

Received in revised form :

29th December 2015

Accepted :1st January , 2016

Published online :

4th January 2015

I. INTRODUCTION

De-duplication brings a lot of advantages, security and privacy concerns arise as users sensitive data are susceptible to both the insider and outsider attacks. When compares the traditional encryption with data duplication. It will provide data confidentiality. In the traditional encryption requires different users to encrypt data with their own keys. Thus identical copies of different users will lead to different cipher texts, making de-duplication impossible.

One of the new technique has been proposed to encrypt data confidentiality while making de-duplication feasible, i,e convergent encryption. This convergent encrypt provides one convergent key to encrypt/decrypt the data, which is obtained by computing the cryptographic hash value of the content of the data copy. After completion of key generation and data encryption, users retain the keys and send the cipher text to the cloud. Since the encryption operation is deterministic and is derived from the data content, identical data copies generate the same convergent key and hence the same cipher text.

A secure proof of ownership protocol is also required to provide the proof that the user indeed owns. This is all for prevent unauthorized access, the same file duplicate will found, this process will occur. A pointer from the server will provide to user, after the proof submission, who are having the subsequent file without needing upload the same file. The encrypted file can be downloaded by the user and also decrypted by the corresponding data users with their convergent keys. Thus, convergent encryption allows the cloud to perform de-duplication on the cipher texts and the proof of ownership prevents the unauthorized user to access the file. Suspended." Such type of e-mail frauds have been happened already. Still, reliable online banking and ecommerce are very safe, every time be very careful while providing your personal financial information through internet.

Problem Statement :

1) Traditional encryption, while providing data confidentiality, is incompatible with data deduplication.
2) Identical data copies of different users will lead to different cipher texts, making deduplication impossible.

II. LITERATURE SURVEY

In this section, we have described earlier work done related to the deduplication mechanisms. In previous deduplication systems cannot support differential authorization duplicate check, which is important in many applications. In such an authorized deduplication system, each user is issued a set of privileges during system initialization. Following are some approaches for deduplication.

i) Post-process deduplication
ii) In-line duplication
iii) Source versus target dedupication

Now, let us see these approaches in brief.

A) **Post-process Deduplication:**

With post-process deduplication, new data is first stored on the storage device and then a process at a later time will analyze the data looking for duplication. The benefit is that there is no need to wait for the hash calculations and lookup to be completed before storing the data thereby ensuring that store performance is not degraded. Implementations offering policy-based operation can give users the ability to defer optimization on "active" files, or to process files based on type and location. One potential drawback is that you may unnecessarily store duplicate data for a short time which is an issue if the storage system is near full capacity.

B) **In-line duplication**

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.

C) **Source versus target dedupication**

Another way to think about data deduplication is by where it occurs. When the deduplication occurs close to where data is created, it is often referred to as "source deduplication." When it occurs near where the data is stored, it is commonly called "target deduplication." Source deduplication ensures that data on the data source is deduplicated. This generally takes place directly within a file system. The file system will periodically scan new files creating hashes and compare them to hashes of existing files.

When files with same hashes are found then the file copy is removed and the new file points to the old file. Unlike hard links however, duplicated files are considered to be separate entities and if one of the duplicated files is later modified, then using a system called Copy-on-write a copy of that file or changed block is created. The deduplication process is transparent to the users and backup applications. Backing up a deduplicated file system will often cause duplication to occur resulting in the backups being bigger than the source data. Target deduplication is the process of removing duplicates of data in the secondary store. Generally this will be a backup store such as a data repository or a virtual tape library.

One of the most common forms of data deduplication implementations works by comparing chunks of data to detect duplicates. For that to happen, each chunk of data is assigned identification, calculated by the software, typically using cryptographic hash functions. In many implementations, the assumption is made that if the identification is identical, the data is identical, even though this cannot be true in all cases due to the pigeonhole principle; other implementations do not assume that two blocks of data with the same identifier are identical, but actually verify that data with the same identification is identical. If the software either assumes that a given identification already exists in the deduplication namespace or actually verifies the identity of the two blocks of data, depending on the implementation, then it will replace that duplicate chunk with a link. Once the data has been deduplicated, upon read back of the file, wherever a link is found, the system simply replaces that link with the referenced data chunk. The deduplication process is intended to be transparent to end users and applications.

III. PROPOSED SYSTEM

Convergent encryption has been proposed to enforce data confidentiality while making deduplication feasible. It encrypts/decrypts a data copy with a convergent key, which is obtained by computing the cryptographic hash value of the content of the data copy. After key generation and data encryption, users retain the keys and send the cipher text to the cloud. Since the encryption operation is deterministic and is derived from the data content, identical data copies will generate same convergent key and hence the same cipher text. To prevent unauthorized access, a secure proof of ownership protocol is also needed to provide the proof that the user indeed owns the same file when a duplicate is found. After the proof, subsequent users with the same file will be provided a pointer from the server without needing to upload the same file. A user can download the encrypted file with the pointer from the server, which can only be decrypted by the corresponding data owners with their convergent keys.

Thus, convergent encryption allows the cloud to perform deduplication on the cipher texts and the proof of ownership prevents the unauthorized user to access the file. However, previous deduplication systems cannot support differential authorization duplicate check, which is important in many applications. In such an authorized deduplication system, each user is issued a set of privileges during system initialization each file uploaded to cloud is also bounded by a set of privileges to specify which kind of users is allowed to perform the duplicate check and access the files.

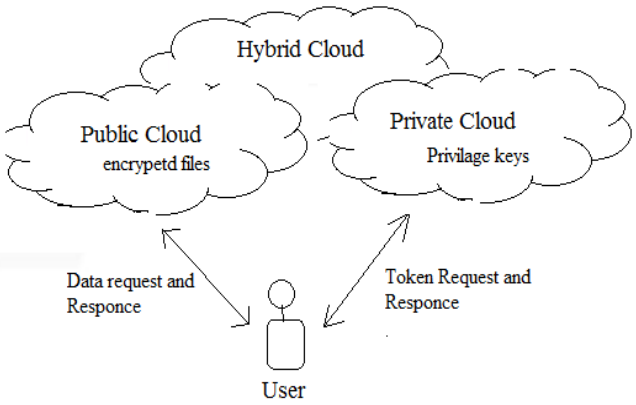

Fig 2:
Proposed System Architecture

Even collude with the S-CSP. Security analysis demonstrates that our system is secure in terms of the definitions specified in the proposed security model specifically; we present an advanced scheme to support stronger security by encrypting the file with differential privilege keys. In this way, the users without corresponding privileges cannot perform the duplicate check. Furthermore, such unauthorized users cannot decrypt the cipher text
The users have to prove that the data which he want to upload or download is its own data. That means he has to provide the convergent key and verifying data to prove his ownership at server.

we enhance our system in security. Specifically, we present an advanced scheme to support stronger security by encrypting the file with differential privilege keys. In this way, the users without corresponding privileges cannot perform the duplicate check. Furthermore, such unauthorized users cannot decrypt the cipher text even collude with the S-CSP. Security analysis demonstrates that our system is secure in terms of the definitions specified in the proposed security model.

Proposed System Algorithm

In this section, we use two types of algorithms,
1). For file uploading.
2). For file downloading.

FOR UPLOADING A FILE
BEGIN
Step –1 Read file
Step –2 Cloud server checks for duplication
Step –3 Sends duplication response whether the file already exists or not
Step – 4 If the file does not exist
4.1 Display “file does not exist”
Step – 5 Then it uploads the file
Step – 6 If the file already exist
6.1 Display “file already exist”
END

FOR DOWNLOADING A FILE
BEGIN
Step –1 Read file
Step –2 Cloud server checks for duplication
Step –3 Sends duplication response whether the file already exists or not
Step –4 If the file exist
-4.1 Display “file exist”
Step –5 then it downloads the file
Step –6 If the file does not exist
-6.1 Display “file does not exist”
END

IV. FEATURES OF PROPOSED SYSTEM

- It makes overhead to minimal compared to the normal convergent encryption and file upload operations.
- Data confidentiality is maintained.
- Secure compared to existing techniques.
- Traditional encryption, while providing data confidentiality, is incompatible with data deduplication.
- Backup approach more security to data.

V. CONCLUSION

In this paper, the notion of authorized data deduplication was proposed to protect the data security by including differential privileges of users in the duplicate check. We also presented several new deduplication constructions supporting authorized duplicate check in hybrid cloud architecture, in which the duplicate-check tokens of files are generated by the private cloud server with private keys... We propose the idea of authorized data deduplication to provide the data security by including differential authority of users in the duplicate check.

In public cloud our data are securely store in encrypted format, and also in private cloud our key is store with respective file. There is no need to user remember the key. So without key anyone can’t access our file or data from public cloud.The advantage of proposed approach is, it also used for backup Purpose. In this way the system will provide an efficient approach for authorised and secure deduplication.

VI. FUTURE WORK

It excludes the security problems that may arise in the practical deployment of the present model. Also, it increases the national security. It saves the memory by deduplicating the data and thus provides us with sufficient memory. It provides authorization to the private firms and protects the confidentiality of the important data

ACKNOWLEDGEMENT

I am glad to express my sentiments of gratitude to all who rendered their valuable guidance to us. I would like to express our appreciation and thanks to the Principal of our college. I am thankful to the Head of Department and my guide **Prof.P.M.Mane**. I also thank to the anonymous reviewers for their valuable comments.

REFRENCES

[1] Jin Li, Yan Kit Li, Xiaofeng Chen, Patrick P. C. Lee, Wenjing Lou, “A Hybrid Cloud Approach for Secure Authorized Deduplication,” in 2015 IEEE Int.Conf. on Parallel and distributed systems, Vol No.99.

[2] P. Anderson and L. Zhang. Fast and secure laptop backups withencrypted de-duplication. In Proc. of USENIX LISA, 2010.

[3] M. Bellare, S. Keelveedhi, and T. Ristenpart. Dupless: Server-aided encryption for deduplicated storage. In USENIX Security Symposium, 2013.

[4] M. Bellare, S. Keelveedhi, and T. Ristenpart. Message-locked encryption and secure deduplication. In EUROCRYPT, pages 296–312, 2013.

[5] M. Bellare, C. Namprempre, and G. Neven. Security proofs for identity-based identification and signature schemes. J. Cryptology, 22(1):1–61, 2009.

[6] M. Bellare and A. Palacio. Gq and schnorr identification schemes: Proofs of security against impersonation under active and concur-rent attacks. In CRYPTO, pages 162–177, 2002.

[7] S. Bugiel, S. Nurnberger, A. Sadeghi, and T. Schneider. Twin clouds: An architecture for secure cloud computing. In Workshop on Cryptography and Security in Clouds (WCSC 2011), 2011.

[8] 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 ICDCS, pages 617–624, 2002.

[9] D. Ferraiolo and R. Kuhn. Role-based access controls. In 15thNIST-NCSC National Computer Security Conf., 1992.

[10]GNU/Libmicrohttpd. http://www.gnu.org/software/libmicrohttpd/.

[11] S. Halevi, D. Harnik, B. Pinkas, and A. Shulman-Peleg. Proofs of ownership in remote storage systems. In Y. Chen, G. Danezis, and V. Shmatikov, editors, ACM Conference on Computer and Communications Security, pages 491–500. ACM, 2011.

