ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 5, May 2016

A Hybrid Cloud Approach for Secure Authorized De-Duplication

Ch.Venkateswarlu, B.L.Sravanthi, R.Rajya Lakshmi

Assistant Professor, Dept. of MCA, Narayana Engineering College, Nellore, AP, India

Student, Dept. of MCA, Narayana Engineering College, Nellore, AP, India

Student, Dept. of MCA, Narayana Engineering College, Nellore, AP, India

ABSTRACT: Data de-duplication 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 band width. To protect the confidentiality of sensitive data while supporting de-duplication, the convergent encryption technique has been proposed to encrypt the data before outsourcing. To better protect data security, this paper makes the first attempt to formally address the problem of authorized data de-duplication. Different from traditional de-duplication systems, the differential privileges of users are further considered in duplicate check besides the data itself .We also present several new de-duplication constructions supporting authorized duplicate check in a hybrid cloud architecture. Security analysis demonstrates that our scheme is secure in terms of the definitions specified in the proposed security model. As a proof of concept, we implement a prototype of our proposed authorized duplicate check scheme and conduct test bed experiments using our prototype. We show that our proposed authorized duplicate check scheme incurs minimal overhead compared to normal operations.

KEYWORDS: De-duplication, authorized duplicate check, confidentiality, hybrid cloud**1**

I. INTRODUCTION

Cloud computing provides seemingly unlimited “virtualized” resources to users as services across the whole Internet, while hiding platform and implementation details. Today’s cloud service providers offer both highly available storage and massively parallel computing resources at relatively low costs. As cloud computing becomes prevalent, an increasing amount of data is being stored in the cloud and shared by users with specified *privileges*, which define the access rights of the stored data. One critical challenge of cloud storage services is the management of the ever-increasing volume of data. To make data management scalable in cloud computing, de-duplication [17] has been a well-known technique and has attracted more and more attention recently. Data de-duplication is a specialized data compression technique for eliminating duplicate copies of repeating data in storage. The technique is used to improve storage utilization and can also be applied to network data transfers to reduce the number of bytes that must be sent. Instead of keeping multiple data copies with the same content, de-duplication eliminates redundant data by keeping only one physical copy and referring other redundant data to that copy

De-duplication can take place at either the file level or the block level. For file level De-duplication, it eliminates duplicate copies of the same file. De-duplication can also take place at the block level, which eliminates duplicate blocks of data that occur in non-identical files .Although data de-duplication brings a lot of benefits, Security and privacy concerns arise as users’ sensitive data are susceptible to both inside and outside attacks Traditional encryption, while providing data confidentiality, is incompatible with data de-duplication. Specifically, Traditional encryption requires different users to encrypt their data with their own keys. Thus, identical data copies of different users will lead to different cipher texts, making de-duplication impossible. Convergent encryption [8] has been proposed to enforce data confidentiality while making de-duplication 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 the same convergent key and hence the same cipher text. To prevent unauthorized access, a secure proof of ownership protocol [11] is also needed to provide the proof that the user indeed owns the same file when a duplicate is found. After the proof,

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 5, May 2016

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 de-duplication on the cipher texts and the proof of ownership prevents the unauthorized user to access the file. However, previous de-duplication systems cannot support *differential authorization duplicate check*, which is important in many applications. In such an authorized de-duplication system, each user is issued a set of privileges during system initialization (in Section 3, we elaborate the definition of a privilege with examples).Each file uploaded to the 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. Before submitting his duplicate check request for some file, the user needs to take this file and his own privileges as inputs. The user is able to find a duplicate for this file if and only if there is a copy of this file and a matched privilege stored in cloud. For example, in a company, many different privileges will be assigned to employees. In order to save cost and efficiently management, the data will be moved to the storage server provider (SCSP) in the public cloud with specified privileges and the de-duplication technique will be applied to store only one copy of the same file. Because of privacy consideration, some files will be encrypted and allowed the duplicate check by employees with specified privileges to realize the access control. Traditional de-duplication systems based on convergent encryption, although providing confidentiality to some extent, do not support the duplicate check with differential privileges. In other words, no differential privileges have been considered in the de-duplication based on convergent encryption technique. It seems to be contradicted if we want to realize both de-duplication and differential authorization duplicate check at the same time.

II. BACKGROUND OF RELATED WORK

Secure De-duplication. With the advent of cloud computing, secure data de-duplication has attracted much attention recently from research community. Yuan et al. [24] proposed a de-duplication system in the cloud storage to reduce the storage size of the tags for integrity check. To enhance the security of de-duplication and protect the data confidentiality, Bellare et al. [3] showed how to protect the data confidentiality by transforming the predictable message into unpredictable message. In their system, another third party called key server is introduced to generate the file tag for duplicate check. Stane et al. [20] presented a novel encryption scheme that provides differential security for popular data and unpopular data. For popular data that are not particularly sensitive, the traditional conventional encryption is performed. Another two-layered encryption scheme with stronger security while supporting de-duplication is proposed for unpopular data. In this way, they achieved better tradeoff between the efficiency and security of the outsourced data. Li et al. [12] addressed the key management issue in block-level de-duplication by distributing these keys across multiple servers after encrypting the files.

Convergent Encryption. Convergent encryption [8] ensures data privacy in de-duplication. Bellare et al. [4] formalized this primitive as message-locked encryption, and explored its application in space-efficient secure outsourced storage. Xu et al. [23] also addressed the problem and showed a secure convergent encryption for efficient encryption, without considering issues of the key-management and block-level de-duplication. There are also several implementations of convergent implementations of different convergent encryption variants for secure de-duplication (e.g., [2], [18], [21], [22]). It is known that some commercial cloud storage
Providers, such as Bitcasa, also deploy convergent encryption.

Proof of ownership. Halevi et al. [11] proposed the notion of "proofs of ownership" (POW) for de-duplication systems, such that a client can efficiently prove to the cloud storage server that he/she owns a file without uploading the file itself. Several POW constructions based on the Merkle-Hash Tree are proposed [11] to enable client-side de-duplication, which include the bounded leakage setting. Pietro and Sorniotti [16] proposed another efficient POW scheme by choosing the projection of a file onto some randomly selected bit-positions as the file proof. Note that all the above schemes do not consider data privacy. Recently, Ng al. [15] extended POW for encrypted files, but they do not address how to minimize the key management overhead.

Twin Clouds Architecture. Recently, Bugiel et al. [7] provided an architecture consisting of twin clouds for secure outsourcing of data and arbitrary computations to an untrusted commodity cloud. Zhang et al. [25] also presented the hybrid cloud techniques to support privacy-aware data-intensive computing. In our work, we consider to address the authorized de-duplication problem over data in public cloud. The security model of our systems is similar to those related work, where the private cloud is assume to be honest but curious.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 5, May 2016

III. CONTRIBUTION

In this paper, aiming at efficiently solving the problem of de-duplication with differential privileges in cloud computing, we consider a hybrid cloud architecture consisting of a public cloud and a private cloud. Unlike existing data de-duplication systems, the private cloud is involved as a proxy to allow data owner/users to securely perform duplicate check with differential privileges. Such architecture is practical and has attracted much attention from researchers. The data owners only outsource their data storage by utilizing public cloud while the data operation is managed in private cloud. A new de-duplication system supporting differential duplicate check is proposed under this hybrid cloud architecture where the S-CSP resides in the public cloud. The user is only allowed to perform the duplicate check for files marked with the corresponding privileges. Furthermore, 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 colludes with the S-CSP. Security analysis Notations Used in This Paper demonstrates that our system is secure in terms of the definitions specified in the proposed security model. Finally, we implement a prototype of the proposed authorized duplicate check and conduct test bed experiments to evaluate the overhead of the prototype. We show that the overhead is minimal compared to the normal convergent encryption and file upload operations.

IV. PROPOSED METHODOLOGY

4.1 Our Proposed System Description

To solve the problems of the construction in Section 4.1, we propose another advanced de-duplication system supporting authorized duplicate check. In this new de-duplication system, hybrid cloud architecture is introduced to solve the problem. The private keys for privileges will not be issued to users directly, which will be kept and managed by the private cloud server instead. In this way, the users cannot share these private keys of privileges in this proposed construction, which means that it can prevent the privilege key sharing among users in the above straight forward construction. To get a file token, the user needs to send a request to the private cloud server. The intuition of this construction can be described as follows. To perform the duplicate check for some file, the user needs to get the file token from the private cloud server. The private cloud server will also check the user's identity before issuing the corresponding file token to the user. The authorized duplicate check for this file can be performed by the user with the public cloud before uploading this file. Based on the results of duplicate check, the user either uploads this file or runs POW. Before giving our construction of the de-duplication system, we define a binary relation R = *f* ((*p, p′*)*g* as follows. Given two privileges *p* and *p′*, we say that *p* matches *p′* if and only if R (*p, p′*) = 1. This kind of a generic binary relation definition could be instantiated based on the background of applications, such as the common hierarchical relation. More precisely, in a hierarchical relation, *p* matches *p′* if *p* is a higher-level privilege. For example, in an enterprise management system, three hierarchical privilege levels are defined as Director, Project lead, and Engineer, where Director is at the top level and Engineer is at the bottom level. Obviously, in this simple example, the privilege of Director matches the privileges of Project lead and Engineer. We provide the proposed de-duplication system as follows.

System Setup. The privilege universe *P* is defined as in Section 4.1. A symmetric key *kpi* for each *pi 2 P* will be selected and the set of keys *fkpi gpi* ∈*P* will be sent to the private cloud. An identification protocol = (Proof, Verify) is also defined, where Proof and Verify are the proof and verification algorithm respectively. Furthermore, each user *U* is assumed to have a secret key *skU* to perform the identification with server's .Assume that user *U* has the privilege set *PU*. It also initializes a POW protocol POW for the file ownership proof. The private cloud server will maintain a table which stores each user's public information *PKU* and its corresponding privilege set *PU*. The file storage system for the storage server is set to be?

File Uploading. Suppose that a data owner wants to upload and share a file *F* with users whose privilege belongs to the set *PF* = *fpjg*. The data owner needs interact with the private cloud before performing duplicate check with the S-CSP. More precisely, the data owner performs an identification to prove its identity with private key *skU*. If it is passed, the private cloud server will find the corresponding privileges *PU* of the user from its stored table list. The user computes and sends the file tag ϕF = TagGen (*F*) to the private cloud server, who will return $f\phi' F;p$= TagGen(ϕF , *kp_*)*g* back to the user for all *p_* satisfying R(*p, p_*) = 1 and *p 2 PU*. Then, the user will interact and send the file token $f\phi' F;p\_g$ to the S-CSP.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 5, May 2016

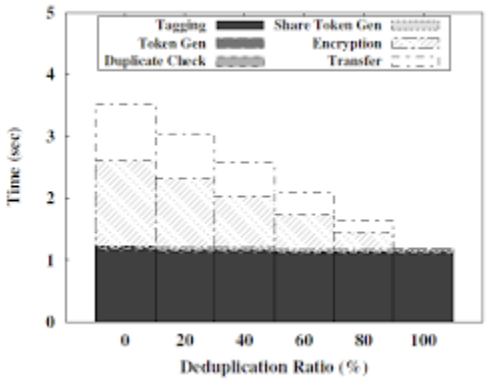
• If a file duplicate is found, the user needs to run the POW protocol POW with the S-CSP to prove the file ownership. If the proof is passed, the user will be provided a pointer for the file. Furthermore, a proof from the S-CSP will be returned, which could be a signature on *fϕ′ F;p g*, *pkU* and a time stamp. The user sends the privilege set *PF* = *fpjg* for the file *F* as well as the proof to the private cloud server. Upon receiving the request, the private cloud server first verifies the proof from the S-CSP. If it is passed, the private cloud server computes *fϕ′ F;p_*= TagGen(*ϕF , kp_)g* for all *p_* satisfying R(*p, p_*) = 1 for each *p 2 PF -PU*, which will be returned to the user. The user also uploads these tokens of the file *F* to the private cloud server. Then, the privilege set of the file is set to be the union of *PF* and the privilege sets defined by the other data owners.

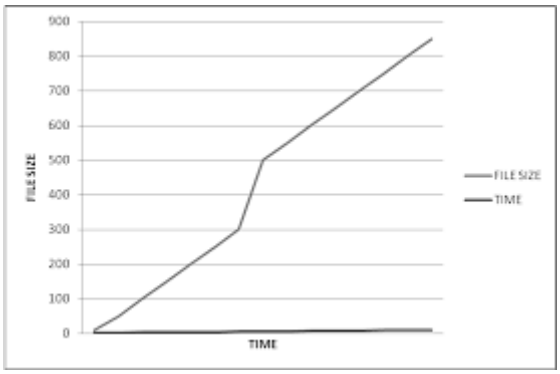
• Otherwise, if no duplicate is found, a proof from the S-CSP will be returned, which is also a signature on *fϕ′ F;p_g*, *pkU* and a time stamp. The user sends the privilege set *PF* = *fpjg* for the file *F* as well as the proof to the private cloud server. Upon receiving the request, the private cloud server first verifies the proof from the S-CSP. If it is passed, the private cloud server computes *fϕ′ F;p_*= TagGen(*ϕF , kp_)g* for all *p_* satisfying R(*p, p_*) = 1 and *p 2 PF* . Finally, the user computes the encrypted file *CF* = EncCE (*kF, F*) with the convergent key *kF* = KeyGenCE(*F*) and uploads *FCF, fϕ′ F;p _gg* with privilege *PF* .

File Retrieving. The user downloads his files in the same way as the de-duplication system in Section 4.1. That is, the user can recover the original file with the convergent key *kF* after receiving the encrypted data from the S-CSP.

V. EXPERIMENTAL RESULTS

To evaluate the effect of number of stored files in the system, we upload 10000 10MB unique files to the system and record the breakdown for every file upload. From Figure 3, every step remains constant along the time. Token checking is done with a hash table and a linear search would be carried out in case of collision. Despite of the possibility of a linear search, the time taken in duplicate check remains stable due to the low collision probability.





Acronym	Description
S-CSP	Storage-cloud service provider
POW	Proof of Ownership
(*PKU,skU*)	User's public and secret key pair
KF	Convergent encryption key for file *F*
PU	Privilege set of a user *U*
PF	Specified privilege set of a file
F ϕ′F;p	Token of file *F* with privilege *p*

TABLE 1

VI. CONCLUSION

In this paper, the notion of authorized data de-duplication was proposed to protect the data security by including differential privileges of users in the duplicate check. We also presented several new de-duplication constructions

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 5, May 2016

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. Security analysis demonstrates that our schemes are secure in terms of insider and outsider attacks specified in the proposed security model. As a proof of concept, we implemented a prototype of our proposed authorized duplicate check scheme and conduct test bed experiments on our prototype. We showed that our authorized duplicate check scheme incurs minimal overhead compared to convergent encryption and network transfer.

VII. ACKNOWLEDGEMENTS

This work was supported by National Natural Science Foundation of China (NO.61100224 and NO.61272455), GRF CUHK 413813 from the Research Grant Council of Hong Kong, Distinguished Young Scholars Fund of Department of Education(No. Yq2013126), Guangdong
Province, China. Besides, Lou's work is supported by US National Science Foundation under grant (CNS-1217889).

REFERENCES

[1].R. S. Sandhu, E. J. Coyne, H. L. Feinstein, and C. E. Youman. Role-based access control models. *IEEE Computer*, 29:38–47, Feb 1996.
[2] J. Stanek, A. Sorniotti, E. Androulaki, and L. Kencl. A secure data deduplication scheme for cloud storage. In *Technical Report*, 2013.
[3] M. W. Storer, K. Greenan, D. D. E. Long, and E. L. Miller. Secure data deduplication. In *Proc. of StorageSS*, 2008.
[4] Z. Wilcox-O'Hearn and B. Warner. Tahoe: the least-authority filesystem. In *Proc. of ACM StorageSS*, 2008.
[5] J. Xu, E.-C. Chang, and J. Zhou. Weak leakage-resilient client-side deduplication of encrypted data in cloud storage. In *ASIACCS*, pages 195–206, 2013

BIOGRAPHY

Venkateswarlu CH is a Assistant Professor in the Department of Master of Computer Applications, Narayana Engineering College, Nellore. His research interest in hybrid cloud secure for de-duplication.

Sravanthi B.L, completed master of computer applications in narayana engineering college, Nellore. Her research interest is hybrid cloud secure for de-duplication.

Rajya lakshmi R, completed master of computer applications in narayana engineering college, Nellore. Her research interest is hybrid cloud secure for de-duplication.

.