

A Hybrid Cloud Approach For Secured Authorised Deduplication

C.Kanagaranjani, S.Karpagam

Abstract — 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. To protect the confidentiality of sensitive data while supporting deduplication, the convergent encryption technique has been proposed to encrypt the data before outsourcing. To better protect data security, this project makes the first attempt to formally address the problem of authorized data deduplication. Different from traditional deduplication systems, the differential privileges of users are further considered in duplicate check besides the data itself. It also present several new deduplication constructions supporting authorized duplicate check in a hybrid cloud architecture. Security analysis demonstrates secure in terms of the definitions specified in the proposed security model. As a proof of concept, implement a prototype of proposed authorized duplicate check scheme and conduct test bed experiments using prototype. In proposed authorized duplicate check scheme incurs minimal overhead compared to normal operations.

Index Terms — Cloud Computing, Bandwidth Allocation, Data Security, Cryptography, Hybrid Cloud Architecture.

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, deduplication has been a well-known technique and has attracted more and more attention recently.

Data deduplication 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, deduplication eliminates redundant data by keeping only one physical copy and referring other redundant data to that copy. Deduplication can

take place at either the file level or the block level. For file-level deduplication, it eliminates duplicate copies of the same file. Deduplication can also take place at the block level, which eliminates duplicate blocks of data that occur in non-identical files.

II. PROBLEM STATEMENT

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. To protect the confidentiality of sensitive data while supporting deduplication, 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.

Although data deduplication brings a lot of benefits, security and privacy concerns arise as users sensitive data are susceptible to both insider and outsider attacks.

- ❖ Traditional encryption, while providing data confidentiality, is incompatible with data deduplication.
- ❖ Identical data copies of different users will lead to different cipher texts, making deduplication impossible.
- ❖ Today's cloud service providers offer both highly available storage and massively parallel computing resources at relatively low costs.
- ❖ To increasing amount of data is being stored in the cloud and shared by users with specified privileges
- ❖ One critical challenge of cloud storage services is the management of the ever-increasing volume of data.

III. PROPOSED SCHEME

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 deduplication. Different from traditional deduplication systems, the differential privileges of users are further considered in duplicate check besides the data itself. We also present several new deduplication constructions supporting authorized duplicate check in a hybrid.

C.Kanagaranjani, PG Scholar, Department of MCA, St. Michael College of Engineering & Technology, Kalayarkoil, TamilNadu.

S.Karpagam, Professor, Department of MCA, St. Michael College of Engineering & Technology, Kalayarkoil, TamilNadu.

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.

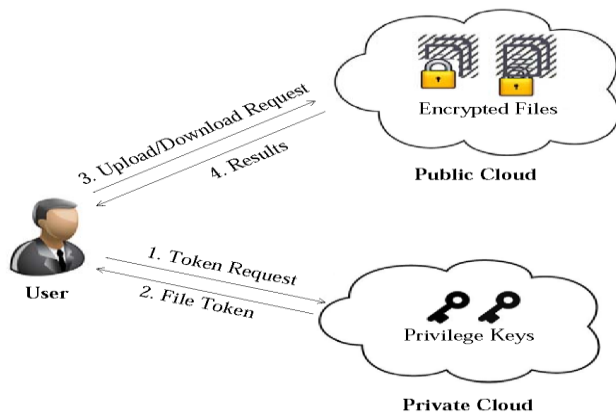


Fig 1. Cloud Authentication Architecture

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 testbed experiments using our prototype. We show that our proposed authorized duplicate check scheme incurs minimal overhead compared to normal operations.

- Differential Authorization - Each authorized user is able to get his/her individual token of his file to perform duplicate check based on his privileges. Under this assumption, any user cannot generate a token for duplicate check out of his privileges or without the aid from the private cloud server.
- Authorized Duplicate Check - Authorized user is able to use his/her individual private keys to generate query for certain file and the privileges he/she owned with the help of private cloud, while the public cloud performs duplicate check directly and tells the user if there is any duplicate.

IV. IMPLEMENTATION

Implementation is the stage of the project when the theoretical design is turned out into a working system. Thus it can be considered to be the most critical stage in achieving a successful new system and in giving the user, confidence that the new system will work and be effective.

The implementation stage involves careful planning, investigation of the existing system and its constraints on implementation, designing of methods to achieve changeover and evaluation of changeover methods.

1. User Module
2. Secure Deduplication System
3. Security of Deduplicate Check Token
4. Send Key

A. User Module:

In this module, Users are having authentication and security to access the detail which is presented in the ontology system. Before accessing or searching the details user should have the account in that otherwise they should register first.

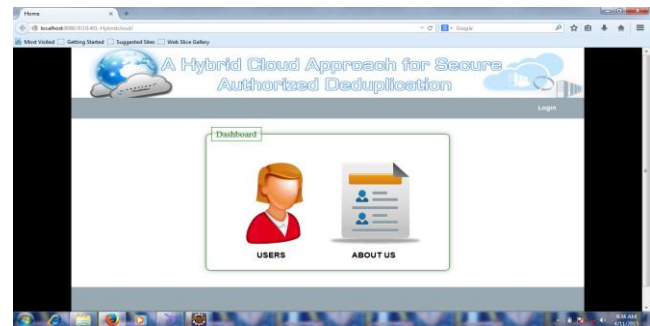


Fig 2. User Module

B. Secure Deduplication System:

To support authorized deduplication, the tag of a file F will be determined by the file F and the privilege. To show the difference with traditional notation of tag, we call it file token instead. To support authorized access, a secret key kp will be bounded with a privilege p to generate a file token. Let $F;p = \text{TagGen}(F, kp)$ denote the token of F that is only allowed to access by user with privilege p . In another word, the token $F;p$ could only be computed by the users with privilege p . As a result, if a file has been uploaded by a user with a duplicate token $F;p$, then a duplicate check sent from another user will be successful if and only if he also has the file F and privilege p . Such a token generation function could be easily implemented as $H(F, kp)$, where $H(_)$ denotes a cryptographic hash function.

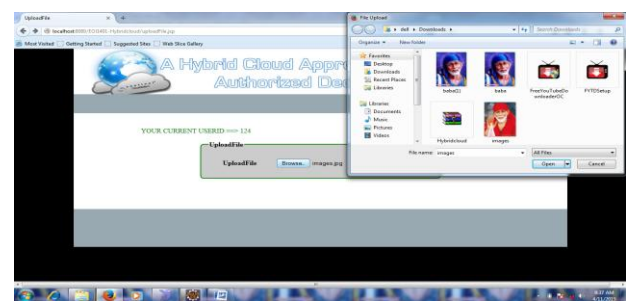


Fig 3. Secure Deduplication System

C. Security of Deduplicate Check Token:

We consider several types of privacy we need protect, that is, i) unforgeability of duplicate-check token: There are two types of adversaries, that is, external adversary and internal adversary. As shown below, the external adversary can be viewed as an internal adversary without any privilege. If a user has privilege p , it requires that the adversary cannot forge and output a valid duplicate token with any other privilege p' on any file F , where p does not match p' .

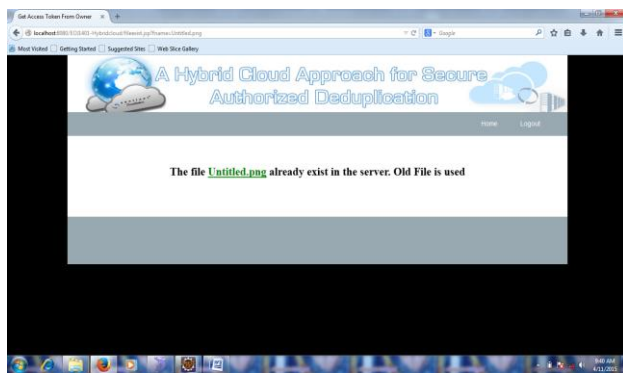


Fig 4. Security of Deduplicate Check Token

Furthermore, it also requires that if the adversary does not make a request of token with its own privilege from private cloud server, it cannot forge and output a valid duplicate token with p on any F that has been queried.

D. Send Key:

Once the key request was received, the sender can send the key or he can decline it. With this key and request id which was generated at the time of sending key request the receiver can decrypt the message.

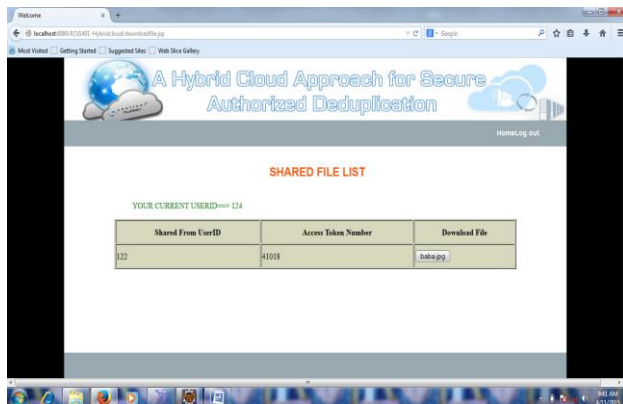


Fig 5. Send Key

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. 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 testbed experiments on our prototype. We showed that our authorized duplicate check scheme incurs minimal overhead compared to convergent encryption and network transfer.

The notion of authorized data deduplication was proposed to protect the data security by including differential privileges

of users in the duplicate check. It 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. Security analysis demonstrates that our deduplication is 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. Authorized duplicate check scheme incurs minimal overhead compared to convergent encryption and network transfer.

VI. FUTURE WORK

Let design and implement a new system which could protect the security for predicable message. The main idea of our technique is that the novel encryption key generation algorithm. For simplicity, To use the hash functions to define the tag generation functions and convergent keys in this section. In traditional convergent encryption, to support duplicate check, the key is derived from the file F by using some cryptographic hash function $kF = H(F)$. To avoid the deterministic key generation, the encryption key kF for file F in our system will be generated with the aid of the private key cloud server with privilege key kp . The encryption key can be viewed as the form of $kF;p = H_0(H(F), kp) H_2(F)$, where H_0, H and H_2 are all cryptographic hash functions. The file F is encrypted with another key k , while k will be encrypted with $kF;p$. In this way, both the private cloud server and S-CSP cannot decrypt the ciphertext. Furthermore, it is semantically secure to the S-CSP based on the security of symmetric encryption with the traditional deduplication architecture in cloud computing, this is a new entity introduced for Proposed system has been designed to solve the differential privilege problem in secure deduplication. The security will be analyzed in terms of two aspects, that is, the authorization of duplicate check and the confidentiality of data. Security of Duplicate-Check Token can consider several types of privacy user need protect.

Unforgeability of duplicate-check token: There are two types of adversaries, that is, external adversary and internal adversary. As shown below, the external adversary can be viewed as an internal adversary without any privilege. If a user has privilege p , it requires that the adversary cannot forge and output a valid duplicate token with any other privilege p' on any file F , where p does not match p' . Furthermore, it also requires that if the adversary does not make a request of token with its own privilege from private cloud server, it cannot forget and output a valid duplicate token with p on any F that has been queried. The internal adversaries have more attack power than the external adversaries and thus we only need to consider the security against the internal attacker.

REFERENCES

- [1] OpenSSL Project. <http://www.openssl.org/>.
- [2] P. Anderson and L. Zhang. Fast and secure laptop backups with encrypted 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 296312, 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 concurrent 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 *15th NIST-NCSC National Computer Security Conf.*, 1992.
 - [10] GNU Libmicrohttpd. <http://www.gnu.org/software/libmicrohttpd/>.