ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

A Novel Hybrid Cloud Approach for Secure Authorized Deduplication

A .Thapan Chandra
M.Tech Student,
Department of CSE,
Global Group of Institutions,
Batasingaram, Ranga Reddy (Dist).

Mr. S Dilli Babu, M. Tech
Assistant Professor,
Department of CSE,
Global Group of Institutions,
Batasingaram, Ranga Reddy (Dist).

Mr. M V Narayana, M. Tech, Ph.D
Associate Professor & HOD,
Department of CSE,
Global Group of Institutions,
Batasingaram, Ranga Reddy (Dist).

Abstract:

“Cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g. networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction.” Data deduplication is a specialized data compression technique for eliminating duplicate copies of repeating data in storage. 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 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 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.

Keywords:

Deduplication, authorized duplicate check, confidentiality, hybrid cloud.

INTRODUCTION:

In cloud computing data de-duplication is a important data compression mechanism for reducing identical copies of same data. this mechanism is used to improve effective use of storage space and also applied to minimize data transmission over network in de-duplication method identical data are find and stored during process of analysis. As process continues other data are matched to the stored copy and whenever matched found the identical data is replaced with a small reference that addressed to stored data. A hybrid cloud is a combination of private cloud and public cloud in which the data which is most critical that resides on a private cloud and the data which is easily accessible is resides on a public cloud hybrid cloud is helpful for reliability, extensibility and fast deployment and cost saving of public cloud with more security with private cloud .

The complex challenge of cloud storage or cloud computing is the arrangement of large volume of data duplication is a process of eliminating of duplicate data in deduplication techniques redundant data removed leaving single instance of the data to be stored. In the previous old system the data is encrypted back to outsourcing it on the cloud or network. This encryption requires maximum time as well as storage space requirement to encode the data if there is large amount of data at that time encryption process becomes complex and critical. By using de-duplication technique in hybrid cloud the encryption technique become simpler. As we all of knows that the network has large amount of data which being shared by many users. Many large networks uses data cloud to store the data and share that data on the network.

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.

Cloud computing is an emerging service model that provides computation and storage resources on the Internet. One attractive functionality that cloud computing can offer is cloud storage. Individuals and enterprises are often required to remotely archive their data to avoid any information loss in case there are any hardware/software failures or unforeseen disasters. Instead of purchasing the needed storage media to keep data backups, individuals and enterprises can simply outsource their data backup services to the cloud service providers, which provide the necessary storage resources to host the data backups. While cloud storage is attractive, how to provide security guarantees for outsourced data becomes a rising concern. One major security challenge is to provide the property of assured deletion, i.e., data files are permanently inaccessible upon requests of deletion.

Keeping data backups permanently is undesirable, as sensitive information may be exposed in the future because of data breach or erroneous management of cloud operators. Thus, to avoid liabilities, enterprises and government agencies usually keep their backups for a finite number of years and request to delete (or destroy) the backups afterwards. For example, the US Congress is formulating the Internet Data Retention legislation in asking ISPs to retain data for two years, while in United Kingdom, companies are required to retain wages and salary records for six years.

LITERATURE SURVEY:

1) Fast and secure laptop backups with encrypted de-duplication:

Many people now store large quantities of personal and corporate data on laptops or home computers. These often have poor or intermittent connectivity, and are vulnerable to theft or hardware failure. Conventional backup solutions are not well suited to this environment, and backup regimes are frequently inadequate. This paper describes an algorithm which takes advantage of the data which is common between users to increase the speed of backups, and reduce the storage requirements. This algorithm supports client-end per-user encryption which is necessary for confidential personal data.

2) Message-locked encryption and secure deduplication:

We formalize a new cryptographic primitive, Message-Locked Encryption (MLE), where the key under which encryption and decryption are performed is itself derived from the message. MLE provides a way to achieve secure deduplication (space-efficient secure outsourced storage), a goal currently targeted by numerous cloud-storage providers. We provide 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. Our work shows that MLE is a primitive of both practical.

3. Security proofs for identity-based identification and signature schemes:

This paper provides either security proofs or attacks for a large number of identity-based identification and signature schemes defined either explicitly or implicitly in existing literature. Underlying these is a framework that on the one hand helps explain how these schemes are derived and on the other hand enables modular security analyses, thereby helping to understand, simplify, and unify previous work. We also analyze a generic folklore construction that in particular yields identity-based identification and signature schemes without random oracles.

4. A reverse deduplication storage system optimized for reads to latest backups:

Deduplication is known to effectively eliminate duplicates, yet it introduces fragmentation that degrades read performance. We propose RevDedup, a deduplication system that optimizes reads to the latest backups of virtual machine (VM) images using reverse deduplication. In contrast with conventional deduplication that removes duplicates from new data, RevDedup removes duplicates from old data, thereby shifting fragmentation to old data while keeping the layout of new data as sequential as possible. We evaluate our RevDedup prototype using a 12-week span of real-world VM image snapshots of 160 users. We show that RevDedup achieves high deduplication efficiency, high backup throughput, and high read throughput.

5. Secure deduplication with efficient and reliable convergent key management:

Data deduplication is a technique for eliminating duplicate copies of data, and has been widely used in cloud storage to reduce storage space and upload bandwidth. Promising as it is, an arising challenge is to perform secure deduplication in cloud storage. Although convergent encryption has been extensively adopted for secure deduplication, a critical issue of making convergent encryption practical is to efficiently and reliably manage a huge number of convergent keys. This paper makes the first attempt to formally address the problem of achieving efficient and reliable key management in secure deduplication. We first introduce a baseline approach in which each user holds an independent master key for encrypting the convergent keys and outsourcing them to the cloud.

However, such a baseline key management scheme generates an enormous number of keys with the increasing number of users and requires users to dedicatedly protect the master keys. To this end, we propose Dekey, a new construction in which users do not need to manage any keys on their own but instead securely distribute the convergent key shares across multiple servers. Security analysis demonstrates that Dekey is secure in terms of the definitions specified in the proposed security model. As a proof of concept, we implement Dekey using the Ramp secret sharing scheme and demonstrate that Dekey incurs limited overhead in realistic environments.

PROBLEM STATEMENT:

* Data deduplication 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.

DRAWBACKS:

* Traditional encryption, while providing data confidentiality, is incompatible with data deduplication.
* Identical data copies of different users will lead to different ciphertexts, making deduplication impossible.

PROBLEM DEFINITION:

In this paper, 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.

ADVANTAGES:

* The user is only allowed to perform the duplicate check for files marked with the corresponding privileges.
* We present an advanced scheme to support stronger security by encrypting the file with differential privilege keys.
* Reduce the storage size of the tags for integrity check. To enhance the security of deduplication and protect the data confidentiality.

IMPLEMENTATION

Cloud Service Provider:

» In this module, we develop Cloud Service Provider module. This is an entity that provides a data storage service in public cloud.

» The S-CSP provides the data outsourcing service and stores data on behalf of the users.
» To reduce the storage cost, the S-CSP eliminates the storage of redundant data via deduplication and keeps only unique data.
» In this paper, we assume that S-CSP is always online and has abundant storage capacity and computation power.

Data Users:

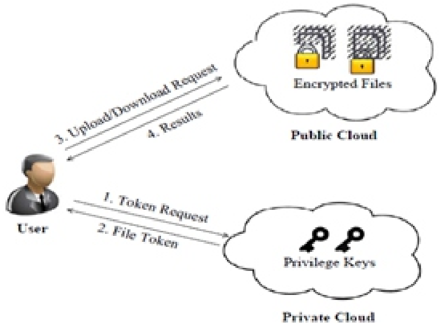
» A user is an entity that wants to outsource data storage to the S-CSP and access the data later.
» In a storage system supporting deduplication, the user only uploads unique data but does not upload any duplicate data to save the upload bandwidth, which may be owned by the same user or different users.
» In the authorized deduplication system, each user is issued a set of privileges in the setup of the system. Each file is protected with the convergent encryption key and privilege keys to realize the authorized deduplication with differential privileges.

Private Cloud:

» Compared with the traditional deduplication architecture in cloud computing, this is a new entity introduced for facilitating user's secure usage of cloud service.
» Specifically, since the computing resources at data user/owner side are restricted and the public cloud is not fully trusted in practice, private cloud is able to provide data user/owner with an execution environment and infrastructure working as an interface between user and the public cloud.
» The private keys for the privileges are managed by the private cloud, who answers the file token requests from the users. The interface offered by the private cloud allows user to submit files and queries to be securely stored and computed respectively.

Secure Deduplication System:

» 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′. 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.

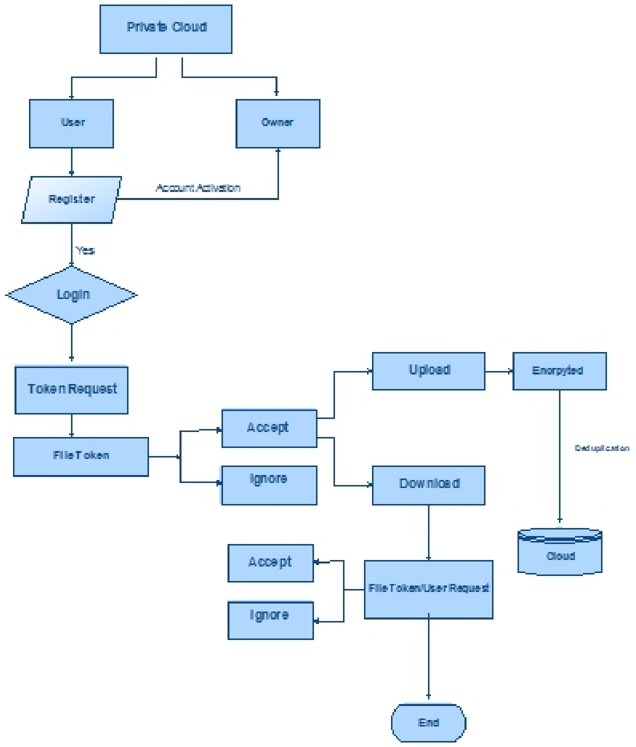


Our implementation of the Client provides the following function calls to support token generation and deduplication along the file upload process.

• FileTag(File) - It computes SHA-1 hash of the File as File Tag;
• TokenReq(Tag, UserID) - It requests the Private Server for File Token generation with the File Tag and User ID;
• DupCheckReq(Token) - It requests the Storage Server for Duplicate Check of the File by sending the file token received from private server;
• ShareTokenReq(Tag, {Priv.}) - It requests the Private Server to generate the Share File Token with the File Tag and Target Sharing Privilege Set;
• FileEncrypt(File) - It encrypts the File with Convergent Encryption using 256-bit AES algorithm in cipher block chaining (CBC) mode, where the convergent key is from SHA-256 Hashing of the file; and
• FileUploadReq(FileID, File, Token) - It uploads the File Data to the Storage Server if the file is Unique and updates the File Token stored.

Our implementation of the Private Server includes corresponding request handlers for the token generation and maintains a key storage with Hash Map.

• TokenGen(Tag, UserID) - It loads the associated privilege keys of the user and generate the token with HMAC-SHA-1 algorithm; and
• ShareTokenGen(Tag, {Priv.}) - It generates the share token with the corresponding privilege keys of the sharing privilege set with HMAC-SHA-1 algorithm



Our implementation of the Storage Server provides deduplication and data storage with following handlers and maintains a map between existing files and associated token with Hash Map.

- DupCheck(Token) - It searches the File to Token Map for Duplicate; and
- FileStore(FileID, File, Token) - It stores the File on Disk and updates the Mapping.

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 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.

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: Serveraided 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 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/.
[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.
[12] J. Li, X. Chen, M. Li, J. Li, P. Lee, and W. Lou. Secure deduplication with efficient and reliable convergent key management. In IEEE Transactions on Parallel and Distributed Systems, 2013.
[13] libcurl. http://curl.haxx.se/libcurl/.
[14] C. Ng and P. Lee. Revdedup: A reverse deduplication storage system optimized for reads to latest backups. In Proc. of APSYS, Apr 2013.