ISSN:0975-9646

Eliminating Redundancy in File System Using Data Compression and Secured File Sharing

Raakesh[1], Varun Raj G[1] , Krishna R[1] Mr.L.Maria Michael Visuwasam[2]M.E.,M.B.A,(Ph.D)

Department Of Computer Science And Engineering
Velammal Institute of Technology

***Abstract*- Cloud computing promises to increase the velocity with which applications are deployed. Data Compression in cloud computing deals with reducing the storage space and providing privacy for users .Each authorized user is able to get an individual token of their file from duplicate check based on the privileges. Authorized user is able to use his/her individual private keys to generate query and hence attributes are attached along with the file. Attributes are found in the private cloud and hence control immediately passes to the private cloud, where duplicate check is performed. Data stored in the public cloud is accessed only by the authorized users by providing different encryption privilege keys. Convergent and symmetric encryption techniques produce identical cipher text that results in minimum overhead. Proof of reliability assures a verifier via a proof that a user's file is available.**

Key words – De-duplication, authorized duplicate check, confidentiality, file sharing .

INTRODUCTION

Cloud computing increases the speed and dexterity which alludes to accessing the internet in a specific data center of different hardware and software. It is a used to describe a class of network based computing that takes place over the internet. It comprises the procurement of dynamically adaptable and virtualized reserves as a indulgence over the internet. This technology allows more efficient computation by centralizing storage memory processing and bandwidth. A censorious confrontation for cloud storage is the management of aggregate volume of accumulating data. In order to manipulate the data management, data compression or data de-duplication technique has been proposed and intrigues more attention. Since the amount of data storage is larger, there may be large amount of duplicate copies. In order to avoid those unwanted data and to save the storage space [9], a peculiar data compression technique has been enabled to remove the redundant data .This helps to reduce the byte storage in cloud. Only one copy of the tautological data is kept ant the remaining data are excluded. Redundant data are replaced with pointers, so that only eminent data can be retrieved. Pointers are provided to users with same file so that there is no obligation to upload the file though there are many privileges, certain security crisis may occur internally and externally. Hence certain encryption techniques are handled and are accompanied by cipher texts. De-duplication can be made possible by formatting contrast cipher texts for divergent users. The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. Any system developed must not have a high demand on the available technical resources.The system must have a modest requirement, as only minimal or null changes are required for implementing this system.Obviously, technologies thereby need to recognizeresults from these areas, just as economical and legalistic views need to acknowledgethe technological capabilities and restrictions. To solve the problem of replication in cloud computing, hybrid cloud is contemplated and it is encompasses public cloud and private cloud. The function of public cloud is to manage the data storage. Private cloud monitors the attribute size. S-CSP is responsible for data storage in public cloud. Attributes are thoroughly checked in the private cloud. Only files with eminent privileges are allowed for duplicate check. There are certain instances where we need to enhance our security in cloud computing. Cloud service providers store more data in the same server and hence it may lead to repetition of data and security complications.

PRELIMINARIES

In this section, we first define the notations used in this paper, review some secure primitives used in our secure deduplication. The notations used in this paper are listed in TABLE 1. Symmetric encryption. Symmetric encryption uses a common secret key κ to encrypt and decrypt information. A symmetric encryption scheme consists of three primitive functions: • KeyGenSE(1λ) → κ is the key generation algorithm that generates κ using security parameter 1λ; • EncSE(κ,M) → C is the symmetric encryption algorithm that takes the secret κ and message M and then outputs the ciphertext C; and • DecSE(κ,C) → M is the symmetric decryption algorithm that takes the secret κ and ciphertext C and then outputs the original message M. 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. 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. Formally, a convergent encryption scheme can be defined with four primitive functions:

• KeyGenCE(M) → K is the key generation algorithm that maps a data copy M to a convergent key K;

• EncCE(K,M) → C is the symmetric encryption algorithm that takes both the convergent key K and the data copy M as inputs and then outputs a ciphertext C;

• DecCE(K,C) → M is the decryption algorithm that takes both the ciphertext C and the convergent key K as inputs and then outputs the original data copy M; and

• TagGen(M) → T(M) is the tag generation algorithm that maps the original data copy M and outputs a tag T(M).

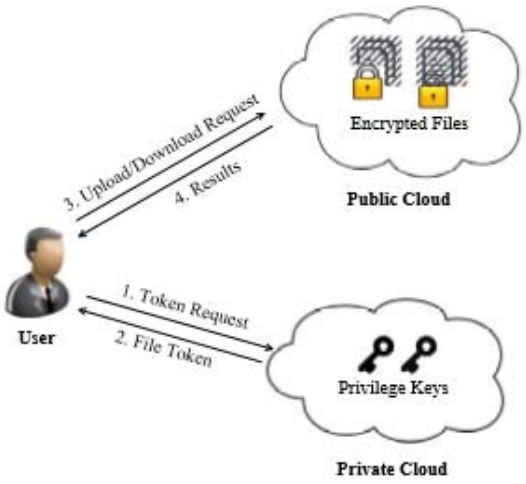

Fig1.a

SYSTEM MODEL

Hybrid Architecture for Secure Deduplication

At a high level, our setting of interest is an enterprise network, consisting of a group of affiliated clients (for example, employees of a company) who will use the S-CSP and store data with deduplication technique. In this setting, deduplication can be frequently used in these settings for data backup and disaster recovery applications while greatly reducing storage space. Such systems are widespread and are often more suitable to user file backup and synchronization applications than richer storage abstractions. There are three entities defined in our system, that is, users, private cloud and S-CSP in public cloud as shown in Fig. 1. The S-CSP performs deduplication by checking if the contents of two files are the same and stores only one of them. The access right to a file is defined based on a set of privileges. The exact definition of a privilege varies across applications. For example, we may define a rolebased privilege [9], [19] according to job positions (e.g., Director, Project Lead, and Engineer), or we may define a time-based privilege that specifies a valid time period (e.g., 2014-01-01 to 2014-01-31) within which a file can be accessed. A user, say Alice, may be assigned two privileges "Director" and "access right valid on 201401-01", so that she can access any file whose access role is "Director" and accessible time period covers 2014-0101. Each privilege is represented in the form of a short message called token. Each file is associated with some file tokens, which denote the tag with specified privileges (see the definition of a tag in Section 2). A user computes and sends duplicate-check tokens to the public cloud for authorized duplicate check.

S-CSP. 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.

DESIGN GOALS

3.3 Design Goals In this paper, we address the problem of privacypreservingdeduplication in cloud computing and propose a new deduplication system supporting for • 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. The security requirements considered in this paper lie in two folds, including the security of file token and security of data files. For the security of file token, two aspects are defined as unforgeability and indistinguishability of file token. The details are given below. • Unforgeability of file token/duplicate-check token. Unauthorized users without appropriate privileges or file should be prevented from getting or generating the file tokens for duplicate check of any file stored at the S-CSP. The users are not allowed to collude with the public cloud server to break the unforgeability

SECURED DUPLICATION SYSTEM WITH SHARING.

To make data management scalable in cloud computing, de-duplication 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 can also be applied to network data transfers to reduce the number of bytes that must be sent. Furthermore, such unauthorized users cannot decrypt the cipher text even collude with the S-CSP. Cloud security controls are enabled to reduce the attack from insiders. These are dynamically aligned to the users to reduce complexity and to increase the performance and utilization.

Hence cloud security platform exaggerates way for virtualization and load balance. The aspect of cloud computing is mostly concerned with data portability and information leakage and legal risks concerning compliance. Cloud computing architecture must involve virtualized infrastructure, scalable and dynamic application for users. It must be structured in a methodological way, so that it helps to streamline the process and other requirements. Analysis of existing and proposed system is represented in the following sections.

DISADVANTAGES OF EXISTING SYSTEM:

Attribute sizes are not constant and hence more memory specifications are required.

Traditional encryption, while providing data confidentiality, is incompatible with data de-duplication.

Identical data copies of different users will lead to different cipher texts, making de-duplication impossible.

PROBLEMS

De-duplication system cannot prevent the privilege private key sharing among users. The users will be issued the same private key for the same privilege in the construction. As a result, the users may collude and generate privilege private keys for a new privilege set that does not belong to any of the colluded user. De-duplication system cannot protect the security of predictable files. One of critical reasons is that the traditional convergent encryption system can only protect the semantic security of unpredictable files.

Each user will be issued private keys for their corresponding privileges. These private keys can be applied by the user to generate file token for duplicate check. However, during file uploading, the user needs to compute file tokens for sharing with other users. To compute these file tokens, the user needs to know the private keys. Such a restriction makes the authorized de-duplication system unable to be widely used and limited.

The construction is inherently subject to brute-force attacks that can recover files falling into a known set. That is, the deduplication system cannot protect the security of predictable files. One of critical reasons is that the traditional convergent encryption system can only

OUR PROPOSAL

A hybrid cloud architecture is introduced to solve the problem paper. 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 run PoW.

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.

ADVANTAGES OF PROPOSED SYSTEM:

The private keys for privileges will not be issued to users directly, which will be kept and managed by the private cloud server.

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 straightforward construction.

To get a file token, the user needs to send a request to the private cloudServer.

Extensive security and performance analysis shows that the proposed scheme is highly effective and resilient to malicious data modification attacks.Symmetric encryption keys are provided to encrypt and decrypt the data respectively. Unlike traditional encryption, convergent encryption is more efficient to practice and implement.

A new concept called tag is introduced. Each data is provided with a tag so that replications can be avoided. Privilege keys are stored in the private cloud and hence the user tries to get access through the keys followed by the corresponding data. After the keys are obtained they are applied to the encrypted data stored in the public cloud Attributes are found in the private cloud and hence the user should get authentication

for accessing the data. Once the user gets the access rights then automatically control passes to the user's data present in the public cloud. These are the main concepts of the cloud architecture. Attribute size should be brief and explicit so that more space is not allocated for storage and attributes. Here Proof of ownership acts as a formal security.

Data compression also known to be Data de-duplication helps to reduce the size of the bits and the amount of data from its original appearance. Compression is appropriate because it supports accurate data transmission. Data compression is carried out mail for backup applications and recovery procedures

Hence public and private cloud acts as two main entities in this architecture. For better understanding let us consider a university that includes principal, Hod and teachers

Credentials

This module authorize user in to the system. This adds security to the user data. The login credentials are secured by encryption and they are decrypted back by the server to avoid eaves dropping. Logging in is usually used to enter a specific page, which trespassers cannot see. Once the user is logged in, the login token may be used to track what actions the user has taken while connected to the site.

Granting Access

User who wants the data to be shared, need to be authorized by the data owner. This is done by requesting for access token and access token is automatically sent to the user. The authorization token is mandatory to access the file. A user with not access token cannot view the file too. If an administrator, support representative, or publisher makes setup changes using your login, the setup audit trail lists those changes, including the username of the delegate user who made the changes.

Multiple Access

The above module is repeated for many times to grant access to many users. Here the data is not replicated but it is shared and also the data is read only for the users, so every user can read single data at a time.

One of the fundamental responsibilities of a site owner is to control who can access the site, who can work with site content, and who can make changes to the pages and functionality on the site. As a site owner, we can give some employees permission to read and change site content, and then give other employees only permission to read site content.

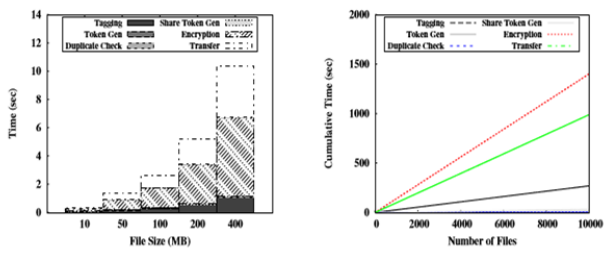

Fig1.b

Access Token Generation

In this module user must log in to the system as data owner and upload the data to the server. This module also has the secure upload facility and request of the user is not recorded to ensure the privacy of the user. The data is transferred from the data owner system to the cloud using http protocol. The access token is generated by the data owner. 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.

IMPLEMENTATION

We implement a prototype of the proposed authorized deduplication system, in which we model three entities as separate C++ programs. A Client program is used to model the data users to carry out the file upload process. A Private Server program is used to model the private cloud which manages the private keys and handles the file token computation. A Storage Server program is used to model the S-CSP which stores and deduplicates files. We implement cryptographic operations of hashing andencryptionwiththeOpenSSLlibrary [1].Wealsoimplement the communication between the entities based onHTTP,usingGNULibmicrohttpd[10]andlibcurl[13]. Thus, users can issue HTTP Post requests to the servers. 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 TagOur 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 .

RELATED WORK

Secure Deduplication. With the advent of cloud computing, secure data deduplication has attracted much attention recently from research community. Yuan et al. [24] proposed a deduplication system in the cloud storage to reduce the storage size of the tags for integrity check. To enhance the security of deduplication and protect the data confidentiality, Bellare et al. [3] showed how to protect the data confidentiality by transforming the predicatable message intounpredicatable message. In their system, another third party called key server is introduced to generate the file tag for duplicate check. Stanek 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 deduplication 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 keymanagement issue in block-level deduplication by distributing these keys across multiple servers after encrypting the files.

Proof of ownership.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 deduplication 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.

CONCLUSION

In this paper, the notion of authorized data decompression 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 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. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make the user familiar with it. Application of cloud computing theory is clear and carefully designed. Security model manipulates that our proposed system is secure in terms of outsider attacks. With the advent of cloud computing, secure data de-duplication has attracted much attention recently from research community. We will try to implement the techniques of data compression, in order to avoid duplicates in videos and images.

REFERENCES

[1] A Hybrid Cloud Approach for Secure Authorized Deduplication Jin Li, Yan Kit Li, Xiaofeng Chen, Patrick P. C. Lee, Wenjing Lou (2014)

[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 encryptionandsecurededuplication. InEUROCRYPT,pages296

[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] GNULibmicrohttpd. 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,andW.Lou. Securededuplication 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.

[15] W. K. Ng, Y. Wen, and H. Zhu. Private data deduplication protocols in cloud storage. In S. Ossowski and P. Lecca, editors, Proceedings of the 27th Annual ACM Symposium on Applied Computing, pages 441–446. ACM, 2012.

[16] R. D. Pietro and A. Sorniotti. Boosting efficiency and security in proof of ownership for deduplication. In H. Y. Youm and Y. Won, editors, ACM Symposium on Information, Computer and Communications Security, pages 81–82. ACM, 2012.

[17] S. Quinlan and S. Dorward. Venti: a new approach to archival storage. In Proc. USENIX FAST, Jan 2002.

[18] A. Rahumed, H. C. H. Chen, Y. Tang, P. P. C. Lee, and J. C. S. Lui. A secure cloud backup system with assured deletion and version control. In 3rd International Workshop on Security in Cloud Computing, 2011.

[19] 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.

[20] J. Stanek, A. Sorniotti, E. Androulaki, and L. Kencl. A secure data deduplication scheme for cloud storage. In Technical Report, 2013.

[21] M. W. Storer, K. Greenan, D. D. E. Long, and E. L. Miller. Secure data deduplication. In Proc. of StorageSS, 2008.

[22] Z. Wilcox-O'Hearn and B. Warner. Tahoe: the least-authority filesystem. In Proc. of ACM StorageSS, 2008.

[23] 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.

[24] J. Yuan and S. Yu. Secure and constant cost public cloud storage auditing with deduplication. IACR Cryptology ePrint Archive, 2013:149, 2013.

[25] K. Zhang, X. Zhou, Y. Chen, X. Wang, and Y. Ruan. Sedic: privacyaware data intensive computing on hybrid clouds. In Proceedings of the 18th ACM conferencen Computer and communications security, CCS'11, pages 515–526, New York, NY, USA, 2011. ACM.