International Journal of Computer Science and Engineering (IJCSE)
ISSN(P): 2278-9960; ISSN(E): 2278-9979
Vol. 6, Issue 2, Feb - Mar 2017; 1-10
© IASET

International Academy of Science, Engineering and Technology
IASET Connecting Researchers; Nurturing Innovations

DEDUPLICATION IN CLOUD COMPUTING USING HYBRID CLOUD

DISHA D N & CHETHANA H R

Department of Computer Science and Engineering, RNS Institute of Technology, Bangalore

ABSTRACT

Data deduplication is the technique which reduces the data size by removing the duplicate copies of identical data and it is extensively used in cloud storage to save bandwidth and minimize the storage space. To avoid this duplication of data and to maintain the confidentiality in the cloud the concept of Hybrid Cloud is used and to secure the confidentiality of sensitive data during deduplication of cloud storage. For better data protection, in this paper different techniques of deduplication are discussed.

KEYWORDS: Deduplication, Hybrid Cloud, Proof of Ownership, Convergent Encryption

INTRODUCTION

In cloud computing, data deduplication is a specialized data compression technique for eliminating duplicate copies of repeating data. Related and somewhat synonymous terms are intelligent (data) compression and single-instance (data) storage. This 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. In the deduplication process, unique chunks of data, or byte patterns, are identified and stored during a process of analysis. As the analysis continues, other chunks are compared to the stored copy and whenever a match occurs, the redundant chunk is replaced with a small reference that points to the stored chunk [1]. Given that the same byte pattern may occur dozens, hundreds, or even thousands of times the match frequency is dependent on the chunk size, the amount of data that must be stored or transferred can be greatly reduced.

To make data management scalable in cloud computing, deduplication has been a well-known technique and has attracted more and more attention recently [2]. 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.

Although data deduplication brings a lot of benefits, security and privacy concerns arise as users' sensitive data are susceptible to both internal and external attacks. Traditional encryption, while providing data confidentiality is incompatible with data deduplication. 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 ciphertexts, making deduplication impossible. Convergent encryption has been proposed to enforce data confidentiality while making deduplication feasible [3]. 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 ciphertext 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 ciphertext. 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 ciphertexts 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 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 deduplication 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 deduplication 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 deduplication based on convergent encryptiontechnique [2]. It seems to be contradicted if we want torealize both deduplication and differential authorization duplicate check at the same time.

LITERATURE SURVEY

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. The overview of the cloud deduplication is as follows [4].

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 Deduplication

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 Deduplication

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 [5], [4]. 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.

EXISTING DEDUPLCATION

Techniques

To achieve deduplication in cloud storage several algorithms are used. Types of algorithm along with the method are explained in this section.

Table 1: Comparison of Deduplication Techniques

Sl. No	Technique Name	Description	Remark
1.	Symmetric encryption Method	3 methods: KeyGen, Encryption, Decryption	**Advantage:**simple **Disadvantage:** Identical copies of different users will lead to different cipher text
2.	Convergent Encryption	4 methods: KeyGen Encryption Decryption TagGen	**Advantage:** Identical copies will generate same cipher text.

			Disadvantage: Customer ownership is not verified.
3.	Proof of ownership	5 methods: KeyGen Encryption Decryption TagGen OwnerProof	**Advantage:** Customer ownership is verified. **Disadvantage:** Differential ownership is not checked.

A. Symmetric Encryption

Symmetric encryption utilizes a regular secret key k to encode the decoded data. A symmetric encryption plan comprises of three basic functions such as [1], [2],

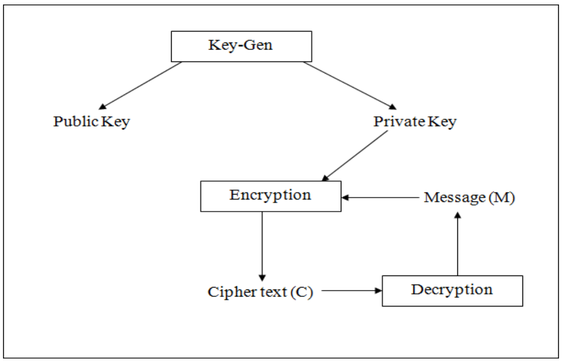

Figure 1: Symmetric Encryption Method

- **Key-Gen:** Key generation algorithm to generate the private and public key pairs.
- **Encryption:** Symmetric encryption algorithm that receives secret key K from Key generation step and message Mand gives ciphertext C.
- **Decryption:** Symmetric decryption algorithm that receives the secret key Kand ciphertext C and gives the original message M.

Disadvantage: Symmetric encryption method requires different users to encrypt their data with their own keys. Thus identical copies of different users will lead to different cipher text. Hence it makes the deduplication impossible.

B. Convergent Encryption

Convergent Encryptiongives information secrecy in deduplication. Customers get a convergent key from each and every unique data copy and encrypt the unique data copy with the convergent key. And also, the customer determines a tag for the unique data copy, which will utilize the tag to recognize duplicate copies. The consideration of the tag accuracy holdsthat means if both the data copies are the same, then the tags of the data copies are same [7]. To discover the duplicate copies, the customer first sends the tag to the server to verify if the duplicate copy has been already available. The convergent key and tags are individually evaluated, and tags cannot understand the convergent key to distract the data security. The encrypted data copy and the respective tag will store on the server. The convergent encryption system can be defined by four basic functions:

- Key-Gen (M) →-key generation algorithm which maps an information data copy M to convergent key K.
- Encce(K,M) →C -symmetric encryption algorithm that receives the input of both data copy M and convergent key K, then gives output cipher text C.

- Decce(K,C) →M –decrypting algorithm which receives the input of the convergent key K and cipher text C, then gives the output of the original data copy M.
- TagGen(M) → T(M) –tags generating algorithm which maps original data copy M and gives output tag T(M).

Advantage: Since the encryption operation is derived from the data content, identical copies will generate same convergent key and hence same cipher text will b produced.

Disadvantage: It does not allow customers to verify the ownership of the information data copies to storage server.

C. Proof of Ownership

The disadvantages of previous two methods can be overcome by this method. The idea of proof of ownership PoW allows customers to verify the ownership of the information data copies to storage server. Particularly, PoW is developed as a communicative algorithm run by a verifier (i.e. customer) and a prover (i.e. storage server). The storage server derives a short term ϕ(M) from an information data copy M. To demonstrate the ownership of information data copy M, the customer needs to send ϕ′ to the storage sever such that ϕ′ = ϕ(M) [8]. The security definition for PoW follows threat system in content distributed network, where the attacker doesn’t knows the whole document, yet has accessories who have the record. The accessories follows "bound retrieval system", that it can help the attacker to get the document, subject to restrict or give limitation that they must send some few bits than the starting min-entropy of the document to the attacker.

Disadvantage: Cannot support differential authorization duplicate check.

What is Differential Authorization Duplicate Check?

It is a system in which each user is issued a set of privileges during system initialization. Each file uploaded to the cloud is also bounded by set of privileges to specify which kind of user is allowed to perform duplicate check and access file. Before submitting his duplicate check request for some file the user needs to take this file and his privilege as input.

The user is able to find duplicate for this file if and only if there is a copy of this file and matched privilege stored in cloud.

Example: In a company many different privileges will be assigned to employee. Data will be moved to the storage server provider in public cloud with specified privileges and deduplication technique is applied to store only one of the same file. Because of the privileges some file will be encrypted and allowed to duplicheck by employees with specified privileges.

HYBRID CLOUD IN DEDUPLICATION

The main aim is to solve the problem of deduplication with differential privileges in cloud computing efficiently. We consider a hybrid cloud architecture consisting of a public cloud and a private cloud. Unlike existing 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. A new deduplication 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 [2].

Furthermore, the system security can be enhanced 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 the system is secure in terms of the definitions specified in the proposed security model.

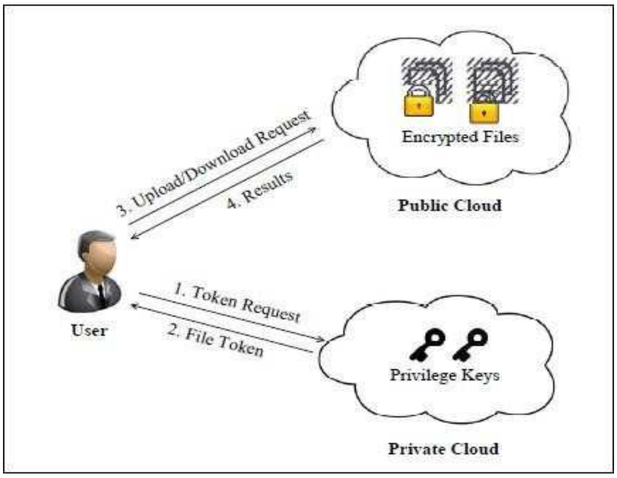

Figure 2: Architecture of Authorized Deduplication [2]

There are three entities defined in this system, that is, users, private cloud and S-CSP in public cloud as shown in Figure 2. 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. 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. A user computes and sends duplicate-check tokens to the public cloud for authorized duplicate check. Users have access to the private cloud server, a semi-trusted third party which will aid in performing deduplicableencryption by generating file tokens for the requesting users. Users are also provisioned with per-user encryption keys and credentials (e.g., user certificates) [9].

In file level deduplication data copy is referred as a whole file and file-level deduplication which eliminates the storage of any redundant files. Actually, block-level deduplication can be easily deduced from file-level deduplication. Specifically, to upload a file, a user first performs the file-level duplicate check. If the file is a duplicate, then all its blocks must be duplicates as well; otherwise, the user further performs the block-level duplicate check and identifies the unique blocks to be uploaded. Each data copy (i.e., a file or a block) is associated with a token for the 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-CS Peliminates the storage of redundant data via deduplication and keeps only unique data. 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 isa 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.

EXAMPLE

HP Store Once

HP Store Once deduplication software simplifies the deployment of deduplication technology across IT infrastructure. Not licensed as standalone software, it is a portable engine that can be easily embedded in multiple infrastructure components, eliminating the complexity seen in earlier-generation deduplication. HP Store Once uses patented innovation and features designed by HP Labs to maximize backup and recovery performance while minimizing management and hardware overhead [10].

HP Store Once deduplication software identifies replicate data inline (upon ingest) with its sparse index-based deduplication approach. This method has two phases:

- HP Store Once algorithms sample large data sequences (approximately 10 MB) to identify the likelihood of duplicates and rapid routing delivers each sequence to the best node for deduplication.
- Store Once uses a SHA-1 hash algorithm on approximately 4 KB variable-length blocks. By using a subset of key values stored in memory, Store Once determines a small number of sequences already stored on disk that are similar to any given input sequence. Then each input sequence is only deduplicated against those few sequences. This minimizes disk IO and uses less disk and little memory, creating more efficiency and enabling faster ingest and, importantly, restoration of data.

As we know, deduplication involves replacing duplicate data with pointers to existing (unique) data. If the unique data is scattered across a storage system (i.e., "fragmented"), then restoring it could take longer because reconstituting it would require many slow random seeks. Store Once avoids this situation by not replacing small amounts of duplicate data with pointers to faraway places with no other related data. This approach greatly improves restore speed with only a bit more extra data stored.

HP's approach has more far-reaching implications. The architecture and design of the deduplication software makes it portable, scalable, and able to deliver global deduplication (within and across independent multiple nodes with a single namespace). The implication is that HP Store Once deduplication can be deployed in a number of iterations; for example, as a virtual machine instance, integrated with HP Data Protector backup and recovery software, and with the HP

X9000 scalable NAS storage. The architecture commonality also means these deployments can extend across the WAN and in ROBO environments without requiring data to be rehydrated and then deduplicated multiple times.

CHALLENGES

However, as deduplication emerges as an answer to the increased demand of storage services in the cloud infrastructures, it introduces the vulnerability of side channel attacks due to cross user deduplication. It has been observed that in spite of various solutions provided, deduplication still suffers from the vulnerability of one or the other side channel attack.

Several attack models have been discovered, which can lead to the exploitation of deduplication towards an insecure storage method. The first attack can be used to predict an already known file possessed by the user. The second attack is related to creating a secret channel for extracting information while the third attack is related to distribution of any file among various users of cloud storage [8], [10].

A. Attack Model I: Predicting Files

This attack can be used to predict whether a particular file is possessed by a specific user 1. Furthermore this attack can be more efficiently used to predict a file if the file contains data with limited possibilities for example yes or no in case of a medical test report. Suppose the attacker wants to find out whether user1 possesses a file, File A. He will upload a copy of file A if the file gets uploaded this will indicate that the file is not possessed by the user1. Whereas in the other case the attacker will be able to find out if the file is possessed by user1.

B. Attack Model II: Creating a Secret Channel

If the attacker manages to install any malicious software on the machine of user1, this software can be used to establish a secret channel between the user1 and the attacker 1. There are several ways of creating this type of channel one of them is to bypass the firewall and communicate with its control server. Consider this example; suppose user1 is using the system with malicious software installed, the software will generate two files in two different conditions. When user1 will backup his files on control server this file will be stored on the server. Now, attacker can easily use the attack described in previous section to find which file was stored by the software.

C. Attack Model III: the Content Distribution Attack

The content distribution attack can be used to distribute a specific file to various users without providing the identity of the distributor. The type of file can be a bootlegged video or a file containing a virus etc. The users in deduplication are enabled to use a file if they are included in the access control list of the file.

CONCLUSIONS

Cloud computing has reached a maturity that leads it into a productive phase. This means that most of the main issues with cloud computing have been addressed to a degree that clouds have become interesting for full commercial exploitation. This however does not mean that all the problems listed above have actually been solved, only that the according risks can be tolerated to a certain degree. Cloud computing is therefore still as much a research topic, as it is a market offering. Though the above solution supports the differential privilege duplicate, it is inherently subject to brute force attacks launched by the public cloud server, which can recover files falling into a known set.

REFERENCES

1. P. Anderson and L. Zhang. “Fast and secure laptop backups with encrypted de-duplication”. In Proc. of USENIX LISA, 2012
2. M. Bellare, S. Keelveedhi, and T. Ristenpart. “Dupless: Server aided encryption for deduplicated storage”. In USENIX Security Symposium, 2013
3. PasqualoPuzio, Refik Molva, MelekOnen,” Cloud Dedup: Secure Deduplication with Encrypted Data for Cloud Storage”, SecludIT and EURECOM, France.
4. Iuon –Chang Lin, Po-ching Chien ,”Data Deduplication Scheme for Cloud Storage” International Journal of Computer and Control(IJ3C), Vol1, No.2 (2012)
5. Shai Halevi, Danny Harnik, Benny Pinkas,” Proof of Ownership in Remote Storage System”, IBM T.J. Watson Research Center, IBM Haifa Research Lab, Bar IIan University, 2011
6. M. Shyamala Devi, V. Vimal Khanna, Naveen Balaji” Enhanced Dynamic Whole File De- Duplication (DWFD) for Space Optimization in Private Cloud Storage Backup”, IACSIT, August, 2014.
7. Weak Leakage-Resilient Client –Side deduplication of Encrypted Data in Cloud Storage” Institute for Info Comm Research, Singapore, 2013
8. TanupriyaChaudhari, Himanshushrivastav, VasudhaVashisht, “A Secure Decentralized Cloud Computing Environment over Peer to Peer”, IJCSMC, April, 2013
9. Mihir Bellare, Sriramkeelveedhi, Thomas Ristenart, “Dup LESS: Server Aided Encryption for Deduplicated storage” University of California, San Diego 2013 BhavanashriShivajiRaut et al / (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 6 (1) , 2015, 89-91
10. 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.ss