



IJIRCCCE

e-ISSN: 2320-9801 | p-ISSN: 2320-9798



INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

Volume 11, Issue 5, May 2023

ISSN INTERNATIONAL
STANDARD
SERIAL
NUMBER
INDIA

Impact Factor: 8.379

 9940 572 462

 6381 907 438

 ijircce@gmail.com

 www.ijircce.com



Hashing Based De-duplication System in Cloud Computing

Poonam Jadhavar, Atharva Girish Kirve, Akshay Kaduba Khandebharad, Shrikant Gokul Patil

Department of Computer Science, Padmabhooshan Vasantdada Patil Institute of Technology, Pune, India

ABSTRACT : Data deduplication, a technique for reducing redundant data by maintaining only one duplicate of a file, saves cloud storage providers a lot of storage and traffic. Individuals and corporations can use cloud storage services to offload data storage to distant servers. Deduplication has become a frequently used technology in cloud data centres to improve the efficiency of IT resources. An attacker can exploit the duplication check to determine whether a file (for example, a pay slip with a given name and salary amount) has already been stored (by someone else), potentially revealing the user's sensitive information. Most preceding systems that rely on a trusted key server (KS) for help are vulnerable and constrained due to information leakage, low attack resistance, high computational cost, and so on. If the reliable KS fails, the entire system fails, resulting in a single point of failure. Data duplication causes security vulnerabilities in existing systems. As a result, we are addressing these difficulties through the use of hash code generation and double encryption approaches. It combines cloud data deduplication and access control. Because only authorised data holders can receive the symmetric keys required for data decryption, encrypted data can be viewed safely. Extensive performance research and testing revealed that, given the described security paradigm, our method is secure and efficient, making it ideal for huge data deduplication. We assess its performance using in-depth study and computer simulations. The results demonstrate the scheme's improved efficiency and efficacy for prospective practical application, particularly for huge data deduplication in cloud storage.

KEYWORDS : Cloud Computing, Data deduplication, Storage Management.

I. INTRODUCTION

Although the cloud storage system has been widely embraced, it does not address some critical new needs, such as the ability of cloud users to verify the integrity of information in the cloud and the detection of duplicate files on cloud servers. Both issues are detailed below by the author. These cloud servers can relieve users of the burden of storage management and maintenance. The primary distinction between cloud storage and typical internal storage is that data is transported over the Internet and stored in an unknown domain that is not under the control of the customer, which inevitably creates serious worries about the integrity of your data. These concerns stem from the fact that cloud storage is affected by security threats both outside and inside the cloud, and servers in the uncontrolled cloud can passively hide some episodes of customer data loss to maintain its reputation. What is more serious is that to save money and space, cloud servers can even exclude an active and deliberate data file that we only have access to and belong to a common customer. Given the large size of outsourced data files and the limited capacity of customer resources, the first problem is widespread so the customer can perform integrity checks effectively, even without a local copy of the data file. Cloud computing is computing in which large groups of remote servers are networked to allow centralized data storage and online access to services or IT resources.

With cloud computing, large groups of resources can be connected via a private or public network. In the public cloud, services (that is, applications and storage space) are available for general use on the Internet. A private cloud is a virtualized data center that operates within a firewall. Cloud computing provides computing and storage resources on the Internet. The increasing amount of data is stored in the cloud, and users with specific privileges share it, which defines special rights to access stored data. Managing the exponential growth of a growing volume of data has become a critical challenge. According to the IDC 2014 cloud report, companies in India are gradually moving from the legacy of premise to different forms of cloud. As the process is gradual, it began during the migration of some cloud application workloads. To perform scalable management of data stored in cloud computing, deduplication has been a well-known technique that has become more popular recently. Deduplication is a specialized data compression technique that reduces storage space and charges bandwidth in cloud storage. In deduplication, only a single instance of data is actually on the server and the redundant data is replaced with a pointer to the copy of the unique data. Deduplication can occur at the file or block level. From the user's point of view, security and privacy issues arise, as



data is susceptible to internal and external attacks. We must properly apply the confidentiality, integrity verification and access control mechanisms of both attacks. Deduplication does not work with traditional cryptography. The user encrypts their files with their own individual encryption key, a different encryption text may also appear for identical files. Therefore, traditional cryptography is incompatible with data duplication. Converged encryption is a widely used technique for combining storage savings with deduplication to ensure confidentiality. In converged encryption, data copy is encrypted with a key derived from the data hash. This converging key is used to encrypt and decrypt a copy of data. After key generation and data encryption, users keep keys and send encrypted text to the cloud. Because cryptography is deterministic, copies of identical data will generate the same convergent key and the same encrypted text. This allows the cloud to duplicate encrypted texts. Cryptographic texts can only be decrypted by the owners of the corresponding data with their converging keys. Differential authorization duplication control is an authorized duplication elimination technique in which each user is granted a set of privileges during system initialization. This privilege set specifies what types of users can perform duplicate checks and access files.

II. RELATED WORK

G. Wallace, F. Douglis, H. Qian, P. Shilane, S. Smaldone, M. Chamness, and W. Hsu has developed Characteristics of backup workloads in production systems The author presents a complete characterization of backup workloads by analyzing statistics and content metadata collected from a large set of EMC Data Domain backup systems in production use. This analysis is complete (it covers the statistics of over 10,000 systems) and in depth (it uses detailed traces of the metadata of different production systems that store almost 700TB of backup data). Author compared these systems with a detailed study of Microsoft's primary storage systems and demonstrated that back-up storage differs significantly from the primary storage workload in terms of data quantities and capacity requirements, as well as the amount of data storage capacity. Redundancy within the data. These properties offer unique challenges and opportunities when designing a disk-based file system for backup workloads [1].

A. El-Shimi, R. Kalach, A. Kumar, A. Ottean, J. Li, and S. Sengupta have developed Primary data deduplication-large scale study and system design The author presents a large-scale study of primary data deduplication and uses the results to guide the design of a new primary data deduplication system implemented in the Windows Server 2012 operating system. The file data were analyzed by 15 servers of globally distributed files that host data for over 2000 users in a large multinational company. The results are used to achieve a fragmentation and compression approach that maximizes deduplication savings by minimizing the metadata generated and producing a uniform distribution of the portion size. Deduplication processing resizing with data size is achieved by a frugal hash index of RAM and data partitioning, so that memory, CPU and disk search resources remain available to meet the main workload of the IO service. [2].

P. Kulkarni, F. Douglis, J. D. LaVoie, and J. M. Tracey, "Redundancy elimination within large collections of files". Propose a new storage reduction scheme that reduces data size with comparable efficiency to the most expensive techniques, but at a cost comparable to the fastest but least effective. The scheme, called REBL (Block Level Redundancy Elimination), exploits the advantages of compression, deletion of duplicate blocks and delta encoding to eliminate a wide spectrum of redundant data in a scalable and efficient way. REBL generally encodes more compactly than compression (up to a factor of 14) and a combination of compression and suppression of duplicates (up to a factor of 6.7). REBL is also coded similarly to a technique based on delta encoding, which significantly reduces the overall space in a case. In addition, REBL uses super fingerprint, a technique that reduces the data needed to identify similar blocks by drastically reducing the computational requirements of the matching blocks: it converts the comparisons of $O(n^2)$ into searches of hash tables. As a result, the use of super fingerprints to avoid enumerating the corresponding data objects decreases the calculation in the REBL resemblance phase of a couple of orders of magnitude [3].

Shweta D. Pochhi, Prof. Pradnya V. Kasture have represents "Encrypted Data Storage with De-duplication Approach on Twin Cloud. The data and the private cloud where the token generation will be generated for each file. Before uploading the data or file to the public cloud, the client will send the file to the private cloud for token generation, which is unique to each file. Private clouds generate a hash and token and send the token to the client. The token and hashes are kept in the private cloud itself, so that whenever the next token generation file arrives, the private clone can refer to the same token. Once the client gets the token for a given file, the public cloud looks for the token similar if it exists or not. If the public cloud token exists, it will return a pointer to the existing file, otherwise it will send a message to load a file. A system that achieves confidentiality and allows block-level deduplication at the same



time. Before uploading the data or file to the public cloud, the client will send the file to the private cloud for token generation, which is unique to each file. The private cloud generates a hash and token and sends them to the client. The token and the hash are kept in the private cloud itself so that whenever the next token generation file arrives, the private clone can refer to the same token.

[4].

Jin Li, Yan Kit Li, Xiaofeng Chen, Patrick P. C. Lee, Wenjing Lou have developed A Hybrid Cloud Approach for Secure Authorized De-duplication[9]. In the proposed system, we are getting data deduplication by providing data evidence from the data owner. This test is used when the file is uploaded. Each file uploaded to the cloud is also limited by a set of privileges to specify the type of users who can perform duplicate verification and access the files. New duplication constructs compatible with authorized duplicate verification in the cloud hybrid architecture where the private cloud server generates duplicate file verification keys. The proposed system includes a data owner test, so it will help implement better security issues in cloud computing [5].

M. Lillibridge, K. Eshghi, and D. Bhagwat represents the improvement in recovery speed for backup systems that use block-based online deduplication. The slow recovery due to the fragmentation of the parts is a serious problem faced by data deduplication systems in one piece: the recovery speeds for the most recent backup can eliminate orders of magnitude during the life cycle of a system. Author have studied three techniques: increase the size of the cache, limit the containers and use a direct assembly area to solve this problem. Limiting the container is a time-consuming task and reduces fragmentation of fragments at the cost of losing part of the deduplication, while using a direct assembly area is a new technique of recovery and caching in the recovery process which exploits the perfect knowledge of the future access to the fragments available during the restoration of a backup to reduce the amount of RAM needed for a certain level of caching in the recovery phase [6].

D. Meister, J. Kaiser, and A. Brinkmann represented caching of data deduplication locations. The author proposes a new approach, called Block Locality Cache (BLC), which captures the previous backup execution significantly better than existing approaches and always uses up-to-date information about the location and is therefore less prone to aging. Author evaluated the approach using a simulation based on the detection of multiple sets of real backup data. The simulation compares the Block Locality Cache with the approach of Zhu et al. and provides a detailed analysis of the behavior and the IO pattern. In addition, a prototype implementation is used to validate the simulation [7].

D. T. Meyer and W. J. Bolosky has represents A study of practical Deduplication. Author collect data from the file system content of 857 desktop computers in Microsoft for a period of 4 weeks. Author analyze the data to determine the relative efficiency of data deduplication, especially considering the elimination of complete file redundancy against blocks. Author have found that full file deduplication reaches about three quarters of the space savings of more aggressive block deduplication for live file system storage and 87% of backup image savings. Author also investigated file fragmentation and found that it does not prevail, and Author have updated previous studies on file system metadata, and Author have found that file size distribution continues to affect very large unstructured files[8].

V. Tarasov, A. Mudrankit, W. Buik, P. Shilane, G. Kuenning, and E. Zadok

having represents generating realistic datasets for the deduplication analysis. The author has developed a generic model of file system changes based on properties measured in terabytes of real and different storage systems. Our model connects to a generic framework to emulate changes in the file system. Based on observations from specific environments, the model can generate an initial file system followed by continuous changes that emulate the distribution of duplicates and file sizes, realistic changes to existing files and file system growth. [9].

P. Shilane, M. Huang, G. Wallace, and W. Hsu

discovered the optimized WAN replication of backup data sets using delta compression reported by the stream. Offsite data replication is critical for disaster recovery reasons, but the current tape transfer approach is cumbersome and error prone. Replication in a wide area network (WAN) is a promising alternative, but fast network connections are expensive or impractical in many remote locations, so better compression is needed to make WAN replication very practical. Author present a new technique for replicating backup data sets through a WAN that not only removes duplicate file regions (deduplication) but also compresses similar file regions with delta compression, which is available as a feature of EMC Data Domain systems. [10].

III. PROPOSED SYSTEM

In this paper, Author propose a confidence scheme in the challenge of data ownership and cryptography to manage the storage of encrypted data with deduplication. Our goal is to solve the problem of deduplication in the situation where the data owner is not available or it is difficult to get involved. Meanwhile, the data size does not affect the performance of data deduplication in our schema. Author are motivated to save space in the cloud and to preserve the privacy of data owners by proposing a scheme to manage the storage of encrypted data with deduplication. Author test safety and evaluate the performance of the proposed scheme through analysis and simulation. The results show its efficiency, effectiveness and applicability.

A. System Architecture

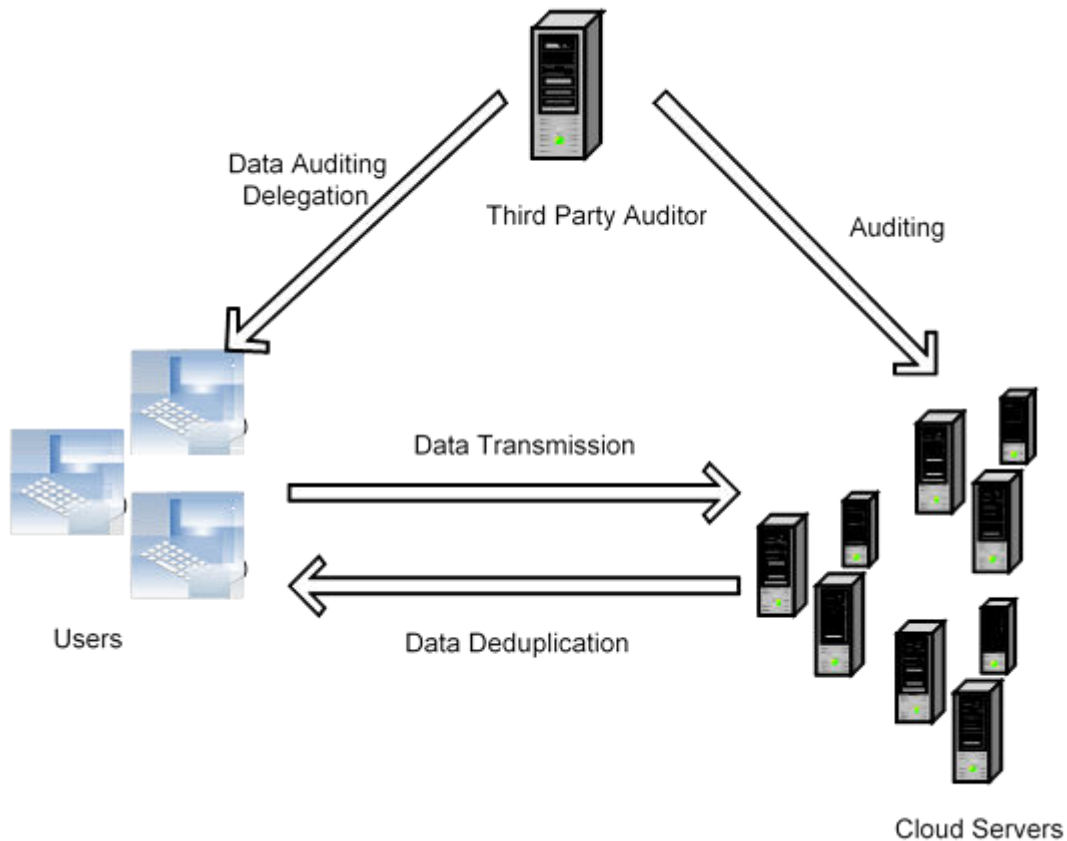


Fig. System Architecture

The primary objective is the achievement of data integrity and deduplication. The first development purpose of this work is to allow for the reliability of remotely stored data to be checked. The integrity check requires two features:

- 1) Public screening, requires any person to carry out the check, not only customers who originally filed the file.
- 2) Stateless monitoring that can remove the need to retain verifier status information between audit activities and data storage.

Another objective of this work is secure deduplication, that is, the cloud server requires a single copy only of the same file to decrease the storage space. The computer-based overhead for secure integrity and deduplication checking should not affect the conventional cloud storage substantially and should not change the way it loads or unloads.



B. Algorithms

1. Encryption Algorithm.

AES (advanced encryption standard).It is symmetric algorithm. It used to convert plain text into cipher text .The need for coming with this algo is weakness in DES. The 56 bit key of des is no longer safe against attacks based on exhaustive key searches and 64-bit block also consider asweak.AES was to be used128-bit block with128-bit keys. Rijendeal was founder. In this drop we are using it to encrypt the data owner file.

Input:

128_bit /192 bit/256 bit input (0, 1)

Secret key (128_bit) +plain text (128_bit).

Process:

10/12/14-rounds for-128_bit /192 bit/256 bit input

Xor state block (i/p)

Final round:10,12,14

Each round consists: sub byte, shift byte, mix columns, add round key.

Output:

cipher text(128 bit)

2.Message-Digest Algorithm

The MD5 message-digest algorithm is a widely used cryptographic hash function producing a 128-bit (16-byte) hash value, typically expressed in text format as a 32 digit hexadecimal number. MD5 has been utilized in a wide variety of cryptographic applications, and is also commonly used to verify data integrity.

Steps:

1. A message digest algorithm is a hash function that takes a bit sequence of any length and produces a bit sequence of a fixed small length.
2. The output of a message digest is considered as a digital signature of the input data.
3. MD5 is a message digest algorithm producing 128 bits of data.
4. It uses constants derived to trigonometric Sine function.
5. It loops through the original message in blocks of 512 bits, with 4 rounds of operations for each block, and 16 operations in each round.
6. Most modern programming languages provides MD5 algorithm as built-in functions

IV. RESULTS AND DISCUSSION

In our experimental setup, in table 1, find out different file upload and time required for time for uploading that file. In our experimental setup, in our system first is uploading file size and time for that file.

Sr.No	File Size(Kb)	Time(ms)
1	10351	226
2	17541	500
3	8500	140

Table1: File Uploading Time and Size

V. CONCLUSION

Data deduplication is critical in the use of cloud data storage, particularly in the administration of large data files. The author suggested a heterogeneous data storage management method in this research, which provides customizable data deduplication in the cloud as well as access control. Our schema is adaptable to various circumstances and application needs, and it provides cost-effective massive data storage management across numerous SPs. Data deduplication can be accomplished with varying levels of security.Our system is safe, advanced, and efficient, according to security analysis, comparison with prior work, and implementation-based performance evaluation.



REFERENCES

- [1] D. Meister, J. Kaiser, and A. Brinkmann, "Block locality caching for data deduplication," in Proc. 6th Int. Syst. Storage Conf., 2013, pp. 1–12.
- [2] M. Lillibridge, K. Eshghi, and D. Bhagwat, "Improving restore speed for backup systems that use inline chunk-based deduplication," in Proc. 11th USENIX Conf. File Storage Technol, Feb. 2013, pp. 183–197.
- [3] V. Tarasov, A. Mudrankit, W. Buik, P. Shilane, G. Kuenning, and E. Zadok, "Generating realistic datasets for deduplication analysis," in Proc. USENIX Conf. Annu. Tech. Conf., Jun. 2012, pp. 261–272.
- [4] D. T. Meyer and W. J. Bolosky, "A study of practical deduplication," ACM Trans. Storage, vol. 7, no. 4, p. 14, 2012.
- [5] G. Wallace, F. Douglis, H. Qian, P. Shilane, S. Smaldone, M. Chamness, and W. Hsu, "Characteristics of backup workloads in production systems," in Proc. 10th USENIX Conf. File Storage Technol., Feb. 2012, pp. 33–48.
- [6] El-Shimi, R. Kalach, A. Kumar, A. Ottean, J. Li, and S. Sengupta, "Primary data deduplication-large scale study and system design," in Proc. Conf. USENIX Annu. Tech. Conf., Jun. 2012, pp. 285–296.
- [7] P. Shilane, M. Huang, G. Wallace, and W. Hsu, "WAN optimized replication of backup datasets using stream-informed delta compression," in Proc. 10th USENIX Conf. File Storage Technol., Feb. 2012, pp. 49–64.
- [8] P. Kulkarni, F. Douglis, J. D. LaVoie, and J. M. Tracey, "Redundancy elimination within large collections of files," in Proc. USENIX Annu. Tech. Conf. Jun. 2012, pp. 59–72.
- [9] Jin Li, Yan Kit Li, Xiaofeng Chen, Patrick P. C. Lee, Wenjing Lou "A Hybrid Cloud Approach for Secure Authorized De-duplication" IEEE Transactions on Parallel and Distributed Systems: PP Year 2014.
- [10] Shweta D. Pochhi, Prof. Pradnya V. Kasture "Encrypted Data Storage with De-duplication Approach on Twin Cloud " International Journal of Innovative Research in Computer and Communication Engineering



INNO  **SPACE**
SJIF Scientific Journal Impact Factor
Impact Factor: 8.379

doi[®]
CROSS **ref**

ISSN INTERNATIONAL
STANDARD
SERIAL
NUMBER
INDIA



INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

 **9940 572 462**  **6381 907 438**  **ijircce@gmail.com**



www.ijircce.com

Scan to save the contact details