International Journal of Scientific Research in Computer Science, Engineering and Information Technology
© 2018 IJSRCSEIT | Volume 3 | Issue1 | ISSN : 2456-3307

Study of Various Mechanisms Used in Data Deduplication in Cloud Storage System

Hema S [1], Dr.Kangaiammal A[2]
[1]Research Scholar, Department of Computer Applications, Govt. Arts College (Autonomous), Salem-7, Tamil Nadu, India
[2]Assistant Professor, Department of Computer Applications, Govt. Arts College(Autonomous), Salem-7, Tamil Nadu, India

ABSTRACT

Cloud computing is an emerging concept that provide different services such as computing, communication and storage resources on demand over the internet. Data deduplication is one of the mainly used techniques in cloud storage, which removes redundant data; reduce network bandwidth and storage utilization. In this paper, the concepts and types of chunk based data deduplication techniques are summarized and also how chunks are uniquely identified by hashing process is discussed.

Keywords: Deduplication, Chunking, Boundary Shift Problem, Convergent Encryption, Cloud Storage Optimization

I. INTRODUCTION

Cloud computing is an internet technology that manages resources and applications using both internet and central remote servers. Many businesses and end users utilize this technology with low cost as well as access application without an installation. Cloud computing gives more effective computing service to clients through centralized memory, high speed processing, large storage capacity and bandwidth. Moreover, It allows remotely access the information and files using the Internet [1,2]

Cloud computing providers offer Infrastructure-as-a-Service (IaaS), Platform-as-a-Service (PaaS), and Software-as-a-Service (SaaS) for users and enterprises to access cloud regardless of time and locality. IaaS is the most basic model from which other higher models abstract the details. PaaS permits developers to generate applications on the provider's platform over the Internet. SaaS is a software distribution model that delivers applications as a service to users and interacts with them through the internet. Organizations or end users do not need to manage software and hardware applications on their own computers. SaaS maintains security, availability and performance.

In recent times a number of organizations develop technology and framework for cloud computing. The current cloud system like Amazon Elastic Compute Cloud (EC2) [3] gives a virtual computing environment that support the users to run their applications based on Linux. Google App Engine[4] supports Application Programming Interfaces (APIs) for the user to effortlessly manage his web applications, data store, Google Account and email services. It supports the user to run their web applications written in Python programming standard. Now, Google App Engine permits up to 500MB of storage to use and 5 million page views per month freely. Sun network.com (Sun Grid) [5] allows the user to run their applications based on Solaris OS, Java, C, C++, and FORTRAN languages.

CSEIT183162| Received:07 Jan2018 | Accepted:20Jan2018 | January-February-2018[(3) 1 : 277-282]

Nowadays, the enormous amount of digital data being generated increases[23], in which a lot of duplicate data are expected to be found. Moreover, backup of duplicate data remarkably increases the storage time and it consumes a lot of network bandwidth. Data deduplication is a technique that plays a vital role in storage, which eliminates redundant data from being, is stored on the backup. In this way, it minimizes cost, reduce storage space and reduce bandwidth usage in network.

II. DEDUPLICATION

For a smart and efficient management of cloud storage, there is a need for de-duplication. Data de-duplication (often called "intelligent compression" or "single-instance storage") is a method, which reduces duplicate data by storing only unique copy of each file or block. Duplicate data is replaced by means of a pointer to the unique copy of data. In Figure1, the data storage holds four unique data elements such as A, B, C and D. Chunk values are used to identify redundant data and pointers are created for redundant data and it is not stored on storage device.

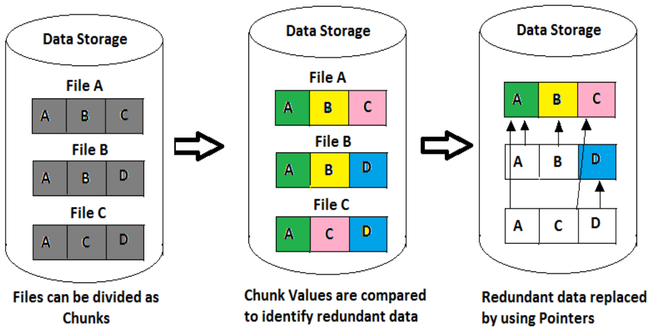
Data Storage
File A
A
B
C
File B
A
B
D
File C
A
C
D
Files can be divided as Chunks
Data Storage
File A
A
B
C
File B
A
B
D
File C
A
C
D
Chunk Values are compared to identify redundant data
Data Storage
A
B
C
A
B
D
A
C
D
Redundant data replaced by using Pointers

Figure 1. Data Deduplication Mechanism

A) Deduplication Process

As mentioned earlier, data deduplication is one of the intelligent data compression techniques, which reduces redundant data storage. Figure2 shows the overview of data deduplication process and the following steps are performed.

(1) Chunking: In this step, files are split into non-overlapping blocks called chunks using chunking algorithms such as SHA -1 or MD5. Most significant Chunking strategies are fixed-size chunking [6] and variable size chunking [2, 11].

(2) Hash Value Generation: The system then calculates unique fingerprint or hash value for each block of data using hash function such as a SHA-1[7] or MD5 [8].

(3) Deduplication checking: In this stage, each chunk hash value is used against the index table to verify the presence of duplicate data. If duplicates found, then the corresponding chunk is not written to disk and eliminated. However, if the hash value does not exist, then the data is written to disk and the index table is updated with the new hash value.

(4) Metadata Management: After indexing, the metadata of the file is recorded or updated that can be used to rebuild the original file.

(5) Data Storage: The ultimate step in this process is to store the unique data blocks. The storage elements in disk are organized by containers [9], which can hold a specific number of data blocks. When the size of the container attains the target size, then it is written from RAM memory to disk. Finally, the metadata of containers are recorded to enhance index-lookup process.

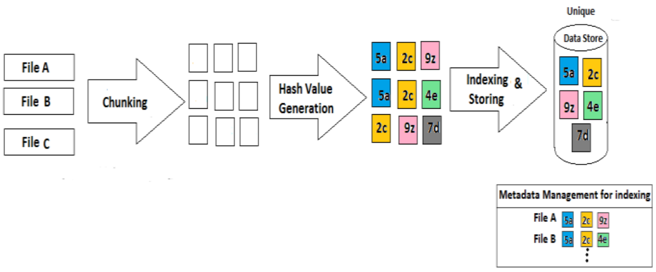

Figure2. Overview of data deduplication Process

B) Deduplication types

Data Deduplication implemented based on location are known as Source Deduplication and Target Deduplication [12,13]. Moreover, Target deduplication is further classified as i) In-Line Deduplication and ii) Post-Process De-Duplication. Figure 3 shows types of deduplication which includes

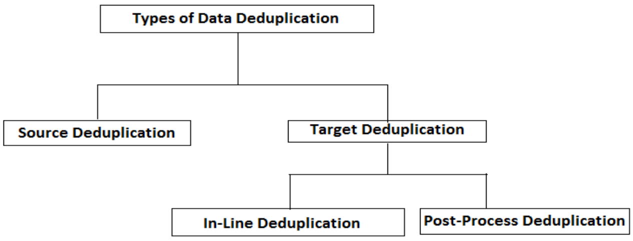

Figure 3. Types of Data Deduplication

1) **Source Deduplication:** Deduplication process functioned at the source side is known as source de-duplication [13]. In this process redundancies are removed before the data is sent to the backup storage. Source deduplication is also recognized as Software based deduplication. Benefits of source deduplication is that it reduce bandwidth usage and time needed to backing up data while the drawback are the consumption of more processor resources and necessity for frequent software version updation.
2) **Target Deduplication:** In Target Deduplication removing of redundant data is performed after sending it to a target [13]. It is also known as hardware based deduplication. Advantages of Target deduplication are that it brings high performance, scalability, and use very less client resources while the disadvantage is using more network bandwidth. Target De-duplication comes in two types i) In-Line De-duplication and ii) Post-Process De-Duplication. In-Line De-duplication takes place at the instant when the data is written to the storage device. The benefit of this method is it needs less storage space than Post-Process method, but it would slow down the speed of data storage. As Post-Process method takes place after the data has been written to disk, it requires a large volume of storage space. It is usually faster than In-Line method.

c) De-duplication Levels

Data deduplication technique is a process of identifying and removing redundant data to increase the storage optimization. While exploring the recent methods, it is found that three levels of detection strategy are used to identify duplicate data. At present, primarily available levels of deduplication are File level deduplication, Block level deduplication, and Byte level deduplication.

1) **File Level Deduplication:** In File level method the entire file is considered as a chunk. It is sometimes called Single Instance Storage (SIS) [14].The hash value of chunk is compared with list of hash index value. If hash value already exists, it stores a reference otherwise; it stores the entire file and enter the chunk hash value into hash index table. This method saves more memory space as well as reduces metadata lookup process and CPU usage. The disadvantage of this method is when a small portion of file is altered, it considers as another version of that file. Furthermore it is not appropriate for large file storage.
2) **Block Level Deduplication:** In Block level [15, 16] approach, the incoming file is divided into non overlapping blocks called a chunk which is either fixed in size or variable in size. Fixed size chunking splits files into equal sized chunks and the negative aspect is boundary shifting problem. In variable size method files can be segmented into non overlapping variable size chunk which gives higher deduplication ratio and save more memory space than fixed size approach. Thus, it is widely used technique for large files.
3) **Byte Level De-Duplication:** In Byte stream approach, the data stream is partitioned into a block which contains some stream of bytes as an element. Hash signature was calculated for this byte stream and compared with hash index to identify duplicate data. This method offers highest deduplication ratio than File level and Block level approach while the problem is that it will creates very large hash index table which leads to performance degradation.

III. MECHANISMS USED IN DEDUPLICATION

Following are some of the mechanisms used in de-duplication to enhance the deduplication process and optimize the storage space.

A) Hash-Based Algorithms

Hash based deduplication methods apply some algorithms on the chunks of data for unique identification. Most commonly used algorithms are Secure Hash Algorithm 1(SHA1) and Message-Digest Algorithm 5(MD5). Both SHA-1(160-bit) and MD5(128 bit) are designed for cryptographic purpose which breaks data into chunks and generate unique hash key for each chunk. SHA-1 is stronger but slower than MD5. It has very less chances of data collision occurrence. MD5 is faster but less secure than SHA-1.

B) The Basic Sliding Window algorithm

One of the hash breaking chunking algorithms named as Basic Sliding Window (BSW) algorithm [20, 21] gives best de-duplication ratio than previous techniques like k-gram and 0 mod p algorithms. In the BSW algorithm, there are three main factors needs to be pre-configured, i) fixed size window(W), ii) integer divisor(D), and iii) integer remainder(R), where R < D. Working of sliding window technique is as follows (in Figure 2):

1) Concept of the BSW Algorithm

(1) A fixed-size window W is shifting one byte at one time from the beginning of the file to end of the file.

(2) If h mod D = R, then that position is a break point for chunk boundary. Sliding window W starts at the break point position and repeats the computation and comparison.

(3) If h mod D ≠ R, the sliding window W keeps shifting one byte and repeats computation and comparison.

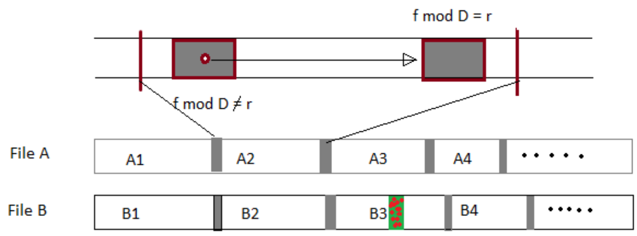

Figure 2: The Sliding Window Technique for the CDC Algorithm

In Figure2, Rabin's algorithm uses a sliding window that helps to identify duplicate chunks (A1, B1) and (A2, B2) present in file A and File B but it fails to find duplicate present in (A3, B3).Hence, this approach cannot identify redundancy between two similar files. Thus, a number of re-chunking schemes such as TTTD, FDC, etc have been suggested to further divide the non- duplicate chunks (e.g., chunks A3 and B3 in Figure 2) into smaller regions to detect more redundancy [8].

2) Problems of the BSW Algorithm.

The BSW algorithm has very poor controls over the chunk-size and hence it may create too small chunk or too large chunk. It is not that much effective to transmits the small data or large data due to only one-byte modification.

C) Two Threshold Two Divisor Algorithm (TTTD)

This algorithm uses same idea as the BSW algorithm does. In addition, the TTTD [11] algorithm uses four parameters. i) The maximum chunk size threshold, ii) The minimum chunk size threshold, iii) The main divisor, and iv) The secondary or backup divisor in order to avoid the problems of the BSW algorithm. The maximum and minimum thresholds are used to remove very large-sized and very small-sized chunks in order to control the variations of chunk-size. The main divisor performs same as the BSW algorithm and second divisor is mainly used to find breakpoint for chunks in case main divisor cannot find any breakpoint. TTTD performs much better than all the existing algorithms, and enhances the performance of applications that uses content based chunking.

D) Content or Application Aware-Based Chunking

Content aware chunking [22] [24] methods understand the format or characteristics of the file which provide good deduplication ratio than the fixed-size and variable-size chunking methods. It compares the incoming file with its index table to identify similarities and relationship (e.g.,mp3 files to mp3 files or mp4 files to mp4 files).This method use delta compression technique to compare very similar files(i.e. different versions of a file). It computes difference (delta) between these similar files and store this delta value rather than to store the whole files.

E) Convergent Encryption

One of the most popular cloud services is data storage service. Cloud users can upload their confidential data to the cloud data centre and permit it to maintain these data. However, intruders may attack to these sensitive data on the cloud server and hack that data. Hence, it is recommended to outsource encrypted form of data to avoid such leakage of sensitive data whereby data security and user privacy is ensured. Here, encrypted data bring new trouble for cloud data deduplication, because deduplication and encryption are two contradictory techniques which turn deduplication more critical on encrypted data. Conventional deduplication methods cannot work properly on encrypted data. This means that if the user applies a standard way to encrypt two identical files the outcome will be different after encryption. Thus, the cloud provider cannot identify the duplicate data while two identical files will be changed after encryption.

A technique recommended to perform deduplication on encrypted data is known as Convergent Encryption [17]. This technique uses an encryption key which is derived from the data content itself to perform encryption on the data and hence, it will generate same identical cipher text for two identical copies of the files. Moreover, this scheme ensures the privacy of user and at the same time it will perform deduplication, unfortunately it is vulnerable to dictionary attacks.

F) Proof of Ownership

Specifically client-side deduplication allows an adversary one who knows a little information about the file can convince the server, as a result the server permits adversary to access the entire file. To overcome such kind of problem, an approach proposed is known as Proof-of-Ownership (PoWs) [18,19], which allows a client to make confirmation to the server that he is the owner of the entire file not just part of the file.

IV.CONCLUSION

Data deduplication plays an important role in cloud storage which eliminates redundant data being stored on cloud storage. This paper, presents various mechanisms used in deduplication to reduce the overhead and enhance the performance of storage. Moreover, this paper discusses how encrypted form of data ensures the security and privacy of user along with pros and cons of some techniques over the other methods.

V. REFERENCES

[1]. D. Irwin, L. Grit, J. Chas, Balancing risk and reward in a market-based task service, in: 13th International Symposium on High Performance, Distributed Computing (HPDC13), June 2004, pp. 160-169.

[2]. C. Yeo, R. Buyya, Service level agreement based allocation of cluster resources: handling penalty to enhance utility, in: 7th IEEE International Conference on Cluster Computing (Cluster 2005), September 2005.

[3]. Amazon Elastic Compute Cloud (EC2), http://www.amazon.com/ec2/ .9 Nov 2017]

[4]. Google App Engine, http://appengine.google.com .9 Nov 2017]

[5]. Sun network.com (Sun Grid), http://www.network.com .9 Nov 2017]

[6]. S. Quinlan and S. Dorward, Venti: A new approach to archival data storage, in

Proceedings of the 1st USENIX Conference on File and Storage Technologies, 2002.

[7]. National Institute of Standards and Technology, FIPSPUB 180-1: Secure hash Standards, Technical Report, 1995.

[8]. R. Rivest, The md5 message-digest algorithm, http://www.ietf.org/rfc/rfc1321.txt, 1992.

[9]. B. Debnath, S. Sengupta, and J. Li, Chunkstash: Speeding up inline storage deduplication using flash memory, in Proceedings of the Annual Conference on USENIX Annual Technical Conference, 2010.

[10]. E. Kruus, C. Ungureanu, and C. Dubnicki, Bimodal content defined chunking for backup streams, in Proceedings of the 8th USENIX Conference on File and Storage Technologies, 2010.

[11]. K. Eshghi and H. K. Tang, A framework for analysing and improving content-based chunking algorithms, Tech. Rep. HPL-2005-30(RI), 2005.

[12]. WikiPedia-online] Available:http://www.wikipedia.com/deduplication 3 nov 2017]

[13]. Q. He, Z. Li, and X. Zhang, ―Data deduplication techniques,‖ in Future Information Technology and Management Engineering (FITME), 2010 International Conference on, vol. 1, 2010, pp. 430-433.

[14]. Bolosky WJ,Corbin S,Goebel D,Douceur JR.Single instance storage in Windows 2000.In:Proc.of the 4th Usenix Windows System Symp.Berkeley: USENIX Association,2000. 13-24.

[15]. S. Quinlan and S. Dorward, “Venti: a new approach to archival storage,” in Proceedings of USENIX Conference on File and Storage Technologies (FAST’02). Monterey, CA, USA: USENIX Association, January 2002, pp. 1-13.

[16]. Liu C, Lu Y, Shi C, Lu G, Du DH, Wang DS. ADMAD: Application-driven metadata aware de-duplication archival storage system. InStorage Network Architecture and Parallel I/Os, 2008. SNAPI'08. Fifth IEEE International Workshop on 2008 Sep 22 (pp. 29-35). IEEE

[17]. J. Douceur, A. Adya, W. Bolosky, D. Simon, and M. Theimer. “Reclaiming space from duplicate files in a serverless distributed file system. In Distributed Computing Systems”, 2002. Proceedings. 22nd International Conference on, pages 617{624. IEEE, 2002.

[18]. Shai Halevi , Danny Harnik , Benny Pinkas , Alexandra Shulman-Peleg, Proofs of ownership in remote storage systems, Proceedings of the 18th ACM conference on Computer and communications security, October 17-21, 2011, Chicago, Illinois, USA

[19]. Z. Yan, W. Ding, X. Yu, H. Zhu and R. H. Deng, "Deduplication on Encrypted Big Data in Cloud," in IEEE Transactions on Big Data, vol. 2, no. 2, pp. 138-150, June 1 2016.

[20]. A. Muthitacharoen, B. Chen, and D. Mazieres, A low-bandwidth network file system. in Symposium on Operating Systems Principles, 2001, page 174-187, 2001.

[21]. Rabin M (1981) Fingerprinting by random polynomials. Center for Research in Computing Technology, Aiken Computation Laboratory, University.

[22]. Mogul J, Douglis F, Feldmann A, Krishnamurthy B (1997) Potential benefits of delta encoding and data compression for HTTP. In: Proceedings of ACM SIGCOMM’97 conference, pp 181- 194, Sept 1997

[23]. X. Zhang, M. Deng, An Overview on Data Deduplication Techniques, Cham:Springer International Publishing, pp. 359-369, 2017.

[24]. Venish, A., and K. Siva Sankar. "Study of Chunking Algorithm in Data Deduplication." Proceedings of the International Conference on Soft Computing Systems. Springer, New Delhi, 2016