ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

Vol. 5, Issue 6, June 2017

Two Level Deduplication Using SHA-256 Algorithm for Resemblance Detection

Rohini S Mahajan, Prof. K. D. Bamane

M. E Student, Dept. of Computer Engineering, Alard College of Engineering and Management, Savitribai Phule Pune University, Pune, India

Professor, Dept of Computer Engineering and Management, D.Y.Patil College Of Engg, Akurdi Savitribai Phule Pune University, Pune, India

ABSTRACT: Data deduplication is a method of reducing storage needs by eliminating redundant data. Only one unique instance of the data is actually retained on storage media, such as disk or tape. Redundant data is replaced with a pointer to the unique data copy. Data deduplication has gained increasing attention and popularity as a space-efficient approach in backup storage systems. Data deduplication not only reduces the storage space requirements by eliminating redundant data but also minimizes the network transmission of duplicate data in the network storage systems.
In proposed system, deduplication is carried out at 2 levels, File level deduplication and Block level deduplication approaches are applied to reduce the data redundancy.
At file-level deduplication, if more than one file is exactly alike, one copy of the file is stored and other repetitions receive pointers to the saved file.
In block deduplication it looks within a file i.e. file is broken down into chunks/blocks. SHA-256 algorithm is applied to each block. The new incoming chunks/blocks are compared with the existing stored chunks saves unique copy of respectively block.
Further delta compression is performed to store the Deltas i.e. difference. Once the unique and deltas are obtained, encryption algorithm is applied to them and store them on different nodes.

KEYWORDS:Data deduplication, Delta Compression, Index-structure, Storage systems, SHA-256

I. INTRODUCTION

"Cloud computing is the computing paradigm that delivers computing services—servers, storage, databases, networking, software, analytics and more—over the net ("the cloud"). Companies which offer this kind of computing resources are known as cloud providers. The charge for using cloud computing services is based on usage, exactly like billed for water or electricity at home. Cloud computing is innovation that uses advanced computational power and improved storage capabilities.

Cloud computing is a pool of configurable computing resources which are shared, and provides on-demand network access. The advantage of cloud is cost savings. The prime disadvantage is security.

In today's digital world, large amount of data is generated every day. To manage such data, we need some techniques. Data reduction is one of them. The basic concept is the reduction of numerous amounts of data down to the meaningful parts. Data reduction increases storage efficiency and reduce costs.

There are several data reduction technologies available. The most widely used data reduction is data deduplication that eliminates data redundancies.

The deduplication process occurs at the storage file level and block level. The system examines the storage to check if duplicate blocks exist, then get rid of any redundant blocks. The remaining block is shared by any file that requires a copy of the block.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

Vol. 5, Issue 6, June 2017

DATA DEDUPLICATION TECHNIQUES

Data deduplication is technique used for reducing the amount of storage space an organization needs to save its data. Most probably, the storage systems of organizations contain duplicate copies of data.
For example, the same file may be saved at different locations by different users, or two or more files that aren't identical may still include much of the same data.

Deduplication discards such extra copies of data by just saving only one copy of the data and replacing the other copies with pointers that lead back to the original copy. Nowadays, companies have started using deduplication in storage systems.

DEDUPLICATION: FILE OR BOCK LEVEL

- **File level deduplication** discards the duplicate copies of the same file. This kind of deduplication is also known as file-level deduplication or single instance storage (SIS). But deduplication can also on the performed at block level, by discarding duplicated blocks of data that occur in non-identical files.
- **Block-level deduplication** saves more space than file level, and a particular type known as variable block or variable length deduplication has become very popular. Often the phrase data deduplication is a synonym for block-level or variable length deduplication.

II. RELATED WORK

In [1], authors had described different chunking models and algorithms with a comparison of their performances. In the deduplication technology, file data is broken down into multiple pieces called “blocks” and every block is identified with a unique hash identifier. These fingerprints are used to compare the blocks with previously stored blocks and verified for duplication. The chunking algorithm is the first step involved for get efficient data deduplication ratio and throughput, it is very important in the deduplication scenario. In [2], author proposed chunking based on frequency for data deduplication. The algorithm makes use of the chunk frequency information from data to improve deduplication.
In [3], Duplicate Data Elimination (DDE) exactly calculates the corresponding hash values of the data blocks before the actual transformation of data at the client side. It works on the combination of copy-on-write, lazy updates & content hashing to identify and coalesce identical blocks of data in SAN system. In [4], Venti is a type of network storage system. The deduplication technique employed by this storage system depends on the recognition of same hash values of the data blocks in order that it will reduce the utilization of the space for storing. It avoids collision of the data using the write once policy. In [5], authors proposed a new scheme for storage reduction that reduces data sizes with an effectiveness comparable to the more expensive techniques, but at a cost comparable to the faster but less effective ones. The scheme, known as Redundancy Elimination at the Block Level (REBL), extracts the benefits of duplicate block suppression, compression, and delta-encoding to discard a large range of redundant data in a scalable and efficient manner. In [6], authors proposed an approach, called Block Locality Cache (BLC) that captures the previous backup that runs significantly better than existing approaches. It also uses up-to date locality information and thus less prone to aging. The authors evaluated the approach using a trace-based simulation of multiple real-world backup datasets.

III. PROPOSED SYSTEM

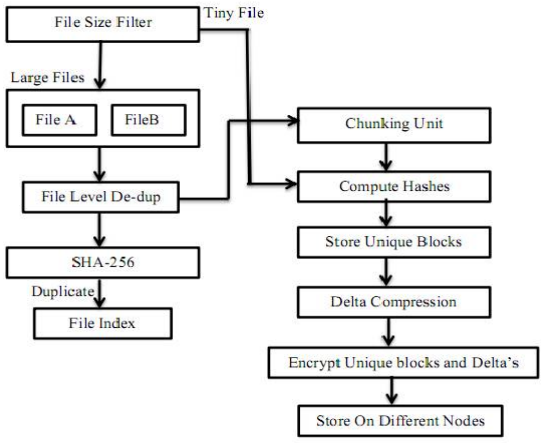

Figure 1: Components Of Deduplication System

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

Vol. 5, Issue 6, June 2017

In this system, the deduplication is performed at two levels;

Root level Deduplication:

Image:

If the uploaded is Image then check if image name already exists. If already available, then show message, "Image already exists, please change Image name". If not available, then calculate RGB value and match RGB value to existing already stored image.

If match is found, then message "Duplicate Image" and provide index (means store unique image).If match not found, then store it.

File (File level Deduplication):

"N" number of files of tiny size or large size is present on system. And deduplication can be performed on this files (txt, word, docx, pdf) text data. The file is tiny if the size of file is less than 8KB. These files are directly considered as block/chunk for deduplication.

At the root level, file level deduplication is performed. If it is file, check if filename exists or not. If exist then message "File name exists". Please change filename.

If file is non_duplicate, then convert it into text file and calculate/generate fingerprint or hash value for the file.

Check file fingerprint, if already exists or match with existing one then perform file-level deduplication. If match is found then it gives message like, "Found duplicate file" and provide index.

Chunk level Deduplication:

If match is not found, then split file into multiple small blocks. (Block level Deduplication).

The fragmentation algorithm splits the large file in small data chunks. If the file size if less than 8kb, it will not be fragmented. It will be considered as a chunk itself, instead of file.

SHA-256 algorithm is used to generate fingerprint (hash value) for blocks/chunks. The reason behind using SHA-256 hash function is that it is one way and collision-resistant hash function. So secure as compared to SHA-1.

The message digest size of hash values produced by SHA-256 is 256 bit. Calculate the hash value of each block. SHA-256 generated fingerprint is used to detect duplicate chunks. Check duplicate block, if found then provide index.

Duplicate chunks checking:

To achieve data deduplication efficiency, fingerprints are compared. If fingerprint matches, data is considered as duplicate and will not be stored, instead link is provided.

If the fingerprint does not match with the existing fingerprint, then it is considered as the new fingerprint.

When a new chunk arrives, it will check against all chunks and store them.

Delta Compression:

In this, if non-duplicate chunk is found, then calculate delta compression is carried out.

Apply encryption algorithm.

Separate out unique data and the difference between two chunks and store them on different nodes in encryption form.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

Vol. 5, Issue 6, June 2017

IV. MATHEMATICAL MODEL

A] Mapping Diagram:

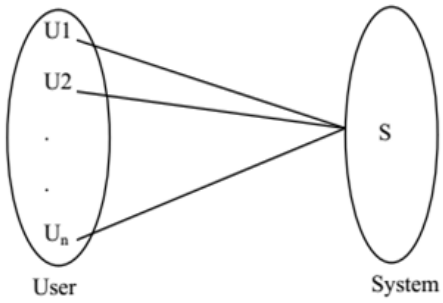

Figure:2 Mapping Diagram

Where,
U1, U2….Un = Users.
S = System

B] Set Theory

Input: File or Image.

Output: Whenever user wants to upload the file on or store the file on system, then check or test the duplication and resemblance identification and removal or not.

Process:

Step 1: Data owner Select File
Step 2: Upload file or store the file on storage.
Step 3: System checks for the duplicate file available on storage system.
Step 4: If found then remove the duplication and maintain index.
Step 5: On non-duplicate data, check for deltas.
Step 6: Store unique and deltas on system in encrypted form.

Mathematical model contains five tuples –

S = {s, e, X, Y, ϕ}

Where, the following conditions are satisfied-

{s} = Start of the program

1. Log in with webpage.

To access the facilities of system such as store on system, user has to log into system.

2. Upload text Files on system.

Upload files on system in text format.

{X} = Input of the program.

Input should be any text file.

{Y} = Output of the program.
{e} = End of the program.
ϕ = Success and failure conditions

File will be first fragmented then it is encoded and the fragments are allocated.

{X, Y € U}

Let U be the Set of System.

{U} = {Client, F, S, T, M, D, R, DC}

Where,
Client, F, S, T, M, D, R, DC are the elements of the set.
{Client} = Data Owner, User.
{F} = Fragmentation
{T} = Generates fingerprints for file and blocks.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

Vol. 5, Issue 6, June 2017

{D} = Check for duplicate file or block.
{R} = Detects similarity by using existing information of a deduplication system.
{DC} = Delta compression module takes each of the blocks detected previously, and reads its base-chunk, and then delta encodes their differences.

Chunking:

Before storing the files on system, Files are broken down into chunks such as,
F = {FC1, FC2….Fcn}

Deduplication Checking:

H (New chunk) = h H (Old n chunks)
If H (New chunk) == H (Old n chunks [])
Chunk is duplicate and do not store it, instead provide link.
Else Chunk is not duplicate, and then stores it.

Encryption:

Each chunk is encrypted before storing on system by using AES algorithm to provide the security over data.
Enc (Fc) = CFC
Where,
C is the cipher text of chunk of file F, (FC).

Success Condition

File splitting and storing it on multiple nodes. User gets result very fast according to their needs.

Failure Condition

- Hardware failure.
- Software failure.
- Maintaining indexing leads to more time consumption to get the proper file stored on system.

Space Complexity

More the storage of data more is the space complexity.

Time Complexity

Time complexity of system depends on following factors: time taken to upload file, time taken during file level and block level deduplication, delta calculating, storing deltas and non-duplicate data in encrypted format on different nodes eq.(3)

Above mathematical model is NP Complete.

V. EXPERIMENTAL RESULTS

In experiments, any number of users can upload the files/Images into the system and those uploaded files are stored in chunk format in the system. Once the file is uploaded, the system first performs file level deduplication and then generates chunks of uploaded non-duplicate file and will check for deduplication and then it check on duplicate detection with the different file for similar chunk and it shows the similar chunks and reference will be generate for particular chunk.

Figure:3 shows time taken for the complete process from file uploading to deduplication checking and storing non-duplicate and deltas on different nodes in encrypted format.

Figure: 4 show the graph of Deduplication on Uploaded Files. The X-axis represents Uploaded Files in KB and Y-axis represents Unique File length i.e. how much unique data is store/found for particular uploaded file.(For Example: The uploaded file size is 9.923KB in which we found 1KB data is unique).

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

Vol. 5, Issue 6, June 2017

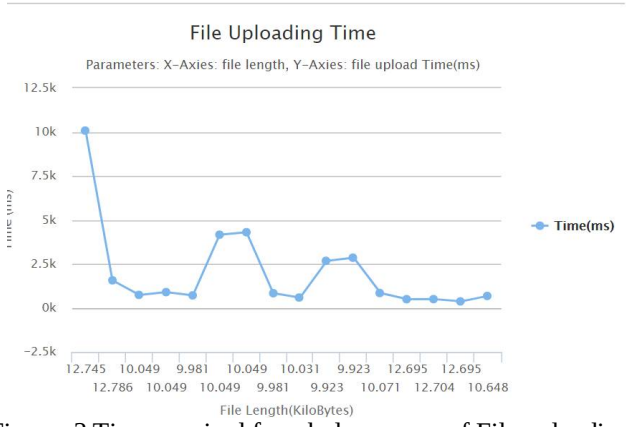

Figure :3 Time required for whole process of File uploading.

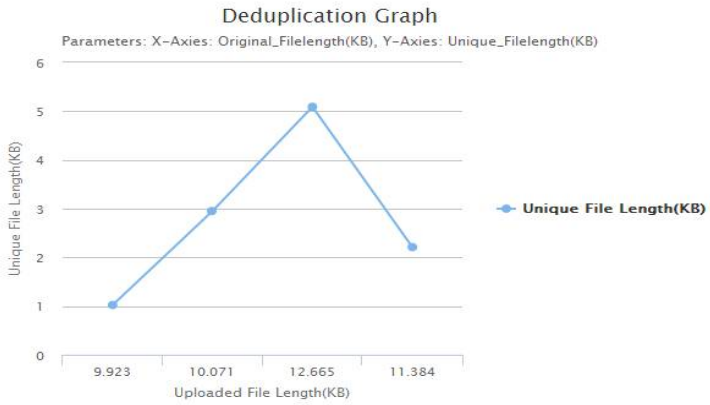

Figure 4: Deduplication graph

VI. CONCLUSION AND FUTURE WORK

The system achieves the storage space management in a secure manner. And therefore the system allows to maximally find and eliminate redundancy at very low overheads by using DARE scheme. A deduplication-aware, low-overhead resemblance detection and elimination scheme is used for data reduction in storage systems. The deduplication is performed on File and Image. The file level deduplication eliminates duplication at root level. In block level deduplication, the incoming data is split into smaller fixed block. Each of these smaller blocks is given a unique identifier which is created by using hashing algorithm SHA-256.

By applying the SHA-256 cryptographic hash function the hash collision is reduced and the security of the data chunks is improved. It compares the data to the already identified blocks and stores in its database. If a block is already available in the database, the new redundant data is removed and a pointer to the existing data is provided. Also, if the block contains new, exceptional information, at that point the block is embedded into the information store (file system).

Delta compression is carried out to find deltas. The unique and deltas are stored on different nodes on system in encrypted format.

The future work is to study and improve the data-restore performance of storage systems based on deduplication and delta compression.

The long term reliability and analysis of duplicated data is still lacking, so is very important in the long term primary or secondary storage systems.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

Vol. 5, Issue 6, June 2017

REFERENCES

[1] G. Lu, Y. Jin, and D .H. Du, Frequency based chunking for data de-duplication, in Proc. IEEE Int. Symp. Model. Anal. Simul. Comput. Telecommun. Syst., Aug.2010, pp. 287296
[2] Venish and K. Siva Sankar, "Study of Chunking Algorithm in Data Deduplication " Proceeding of the International Conference on Soft Computing System, ICSCS , Volume 2.
[3] B. Hong, "Duplicate Data Elimination in a SAN File System," in 21st International Conference on Massive Storage Systems and Technologies (MSST), College Park, MD, 2004.
[4] S. Quinlan and S. Dorward, "Venti: A new approach to archival storage," in Proc. USENIX Conf. File Storage Technol., Jan. 2002, pp. 89–101.
[5] 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.
[6] Meister, J. Kaiser, and A. Brinkmann, "Block locality caching for data deduplication," in Proc. 6th Int. Syst. Storage Conf., 2013, pp. 1–12.
[7] El-Shimi, R. Kalach, A. Kumar, A. Ottean, J. Li, and Sengupta, "Primary data deduplication-large scale study and system design," in Proc. Conf. USENIX Annu. Tech. Conf., Jun. 2012, pp. 285–296.
[8] Zhu, K. Li, and R. H. Patterson, "Avoiding the disk bottleneck in the data domain deduplication file system," in Proc. 6th USE-NIX Conf. File Storage Technol., Feb. 2008, vol. 8, pp. 1–14.
[9] 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.
[10] Jin Li, Yan Kit Li, Xiao feng 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.
[11] T. Meyer and W. J. Bolosky, "A study of practical deduplication," ACM Trans. Storage, vol. 7, no. 4, p. 14, 2012.
[12] 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.
[13] F. Douglis and A. yengar, "Application- specific delta-encoding via resemblance detection, "in Proc. USENIX Annu.Tech.Conf.,GeneralTrack,Jun.2003,pp.113–1264