"METHODOLOGY FOR OPTIMIZING STORAGE ON CLOUD USING AUTHORIZED DE-DUPLICATION" – A Review

[1]Ruchi Agrawal, [2]Prof.D.R. Naidu

[1] M.Tech Student, CSE Department, Shri Ramdeobaba College of Engineering and Management, Nagpur, India
[2]Assistant Professor, CSE Department, Shri Ramdeobaba College of Engineering and Management, Nagpur, India

---***---

Abstract - ***Cloud computing is a transpire computing in which resources are provided as services over the Internet. At the foundation of cloud computing, it is the vast concept of infrastructure services and shared services. The Cloud Storage system will be limited till 2020 . For these data de-duplication plays an important role in the cloud structure. De-duplication is a kind of data compression technique where eliminate the duplicate copies of replicated data. Since only one copy of content is kept in storage, it is often known as single-instance-storage (SIS). In this paper, data de-duplication process will be discussed with latest cryptographic application (i.e. Hash algorithms) in detail. Hashing scheme is firstly on whole file after that on segmented (Chunk/Block) file.***

A comparison between the proposed SHA hash function and the work shows that it occupies optimized area while also achieving a high throughput. We proposed a novel idea of storage optimization method through SHA-512. We illustrate how to use elasticity on de-duplication based systems, which use the ability to dynamically increase memory resources for improvement of de-duplication performance and locate the related chunks i.e. the concept of locality of index. For authentication on public cloud the key concept is used. The correctness of the functionality will verify with certain technique.

Key Words: *De-duplication, Single-instance-storage, Chunking, Secure Hash Algorithm (SHA), Locality.*

1. INTRODUCTION

1.1 Cloud Computing: Cloud computing technique is most widely used technique in today's world. In which, computation is done over the communication network. It is an important for business storage at low cost. Cloud computing provides a large storage in various areas as government, enterprises, and use for storing personal data on cloud. Here user can access and share different resources over cloud. The most important problem in cloud computing is that vast amount of storage and its security issues. One critical challenge of cloud storage to management of day-by-day increasing the volume of data. To improve scalability, storage problem Data De-duplication is most important and more attentive idea. It is an important technique for storage compression, here the duplicate copies of data are identified and merge with references whereas saves the identical copy of data. De-duplication process is done in byte level, block level or file level. In file level, the replicated files are avoided and not to store in disk, similarly in block level idea replicated blocks are not stored. In presented research work, de-duplication is at byte level. Recently focusing issue in data De-duplication is security problem and privacy (authorization) concern for protecting the data from attacker. To upload the file on cloud, first user has to generate the convergent (hash) key then load file to the cloud. For prevention from the unauthorized access, proof of ownership (Pow) protocol is used, that is, user refer the same file if De-duplication found. As the proof completed, server gives a reference pointer to user to accessing same file and the file can be downloaded.

1.2 Data De-duplication : Data De-duplication is defined as elimination of replicated data from the files. The main aim of de-duplication is management of the amount of storage space over cloud, so that information can be singly and simply stored in disk and by that increment of volume of data on network can be easily done. In de-duplication, the single-instance store strategy (identifies and eliminates the repeated instances of identical from whole files and save only single time) is involved. It is suitable with compression systems to compact the data before saving the data blocks to the disk. Any file that present in operating system or the dataset, represent the set of metadata that also contains information about reference pointers of the disk where the blocks of data set are reside. In block level de-duplication, use a technique where a single segment or identical block of data that can be referenced by many pointers from different sets of data. In data De-duplication method, also uses the idea of referring multiple pointers to common blocks.

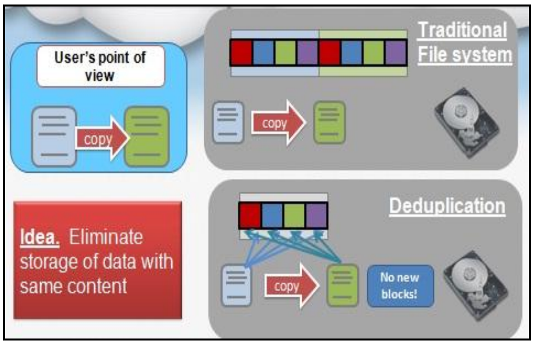

Fig -1: De-duplication Process

How De-duplication works?

Data De-duplication systems, segment all incoming data. In this situation back up the data into "chunks" and calculate a unique hash mark for each chunk. That permits the system to evaluate each chunk's hash against those already backed up. If an incoming chunk's hash is unique, that means that it hasn't been previously backed up, and it is saved. If an incoming chunk's hash matches with a recently backed up chunk, it will be referenced to the recently backed up chunk. Storage demands are decreased by eliminating the storing multiple redundant copies of data.

De-duplication systems can be carried out file level, block level or byte level. The byte level is much more granular than the block level and file level. De-duplication operations performs with inline (duplicates are found during the procedure of creating a backup) or offline (De-duplication occurs after the backup procedure by reading through the backup file and eradicating duplicates).

Steps of De-duplication

1. Chunking - Every file that user uploads will be of a different size. The uploaded whole file can be hashed at once but there is a disadvantage because if its sub-file is duplicated then it can't be determined. Therefore files are chunked into blocks. The chunking size is user dependent. For our convenience we have chosen block size as 4kb. 4kb is found to be most optimized block size by the researches at HP (Hewlett-Packard). The unit chosen for chunking is a single character. The user input file is scanned character by character, and whenever scanned size reaches 4kb, the block is chunked.

Chunk Selection: The selection of a chunk size is the most important characteristic for effective De-duplication. As the chunk size grows, the probability of finding matching chunks decreases.. Experience has shown that the chunk size should be less than 10KB. A 4kb chunk size is common. There are basic methods for breaking up a data into chunks on the basis of size– fixed chunk size and variable chunk size.

2. Secure Hash Algorithms - Hashes are used to ensure data and message integrity as well as password validity. The hash function is a one way function. It means, hash code can be generated from any string, but string cannot be generated from given hash code. Because of this, the integrity of the text is maintained. Each segment block generated in the chunking process is passed as input to the SHA algorithms. Each SHA gives hash codes as output for each string. Hash codes are unique for each identical string. Therefore, for duplicated blocks, the hash code will again be same as previous.

Comparison and Storage

The application contains two databases for this purpose. In the first table, the hash codes are mapped against contents of original file. Here the hash code is set as a unique key constraint. If another file has same contents as of the original file, then the hash codes generated will be identical. But due to the unique key constraint, only unique hash codes are stored in the table.

In second table, the hash codes are mapped against the usernames and filenames. In case of identical hash codes, the duplicate hash codes are also be stored in the table because they are mapped against username and filename which are identical. This is done because it is easier to retrieve a file when a user requests to download a file.

3. For Security Purpose : CPABE

1. **Key generation center:** It is a key authority that generates public and private parameters for CP-ABE. It is used in charge of issuing, revoking, and updating attribute keys for users. It grants access rights to each users based on their attributes. It is assumed to be honest-but-curious. Thus, it should be prevented from accessing the plaintext even if it is honest.
2. **Data-storing center:** It is a center that provides a data sharing service. It is used in controlling the accesses from outside users to the storing data and provides corresponding contents services. The data-storing center has another key authority that generates personalized user key with the Key Generation Center.

3. **Data Owner:** A client who owns data, and wishes to upload it into the external data-storing center for ease of sharing or for cost saving. A data owner is responsible for defining (attribute-based) access policy, and enforcing it on its own data by encrypting the data under the policy before distributing it.

4.**User:** It is an entity who wants to access the data. If a user possesses a set of attributes satisfying the access policy of the encrypted data, and is not revoked in any of the valid attribute groups, then he will be able to decrypt the cipher text and obtain the data.

2. REVIEW OF LITERATURE:

According to the data granularity, De-duplication strategies can be categorized into two main categories: file-level De-duplication and block-level De-duplication, which is nowadays the most common strategy. In block-based De-duplication, the block size can either be fixed or variable. Another categorization criterion is the location at which De-duplication is performed if data are de-duplicated at the client, and then it is called source-based De-duplication, otherwise target-based. In source-based De-duplication, the client first hashes each data segment he wishes to upload and sends these results to the storage provider to check whether such data are already stored: thus only "not de-duplicated" data segments will be uploaded by the user on the cloud. While De-duplication at the client side can achieve bandwidth savings, it unfortunately can make the system vulnerable to side-channel attacks whereby attackers can immediately discover whether a certain data is stored or not. On the other hand, by de-duplicating data at the storage provider.

Many people now store huge amount of personal and corporate data on laptops or home computers. These often have poor connectivity, and are susceptible to theft or hardware failure...Below there is brief about the papers which we referred for our project.

- **Paper by Zhang titled "Fault Tolerant digital signature Scheme"*.**
  It improves the speed of data De-duplication. The Signature is computed for uploaded file for verifying the integrity of files. There is a problem of the worst case in that cloud storage server will regard all blocks as a new blocks and store all of these blocks, resulting in storing duplicate blocks the probability of the worst case is low and won't affect most. To be concluded it improves the speed of data De-duplication phase not only enhances the efficiency of data duplication.

- **Paper by S. Quinlan and S. Forward. Venti titled "A new approach to archival storage"*.**
  In 2002, this paper presents an approach towards De-duplication called write-once policy of data. It provides efficient storage applications such as backup system i.e. logical backup, physical backup, and snapshot file systems.

- **Paper by Bugiel et al titled "Architecture for secure cloud computing".**
  It provided an architecture consisting of twin clouds for securely outsourcing of user private data and arbitrary computations to an untrusted commodity cloud. Privacy aware data intensive computing on hybrid clouds - Zhang et al also presented the hybrid cloud techniques to support privacy-aware data-intensive computing. We used public cloud of elastic infrastructure.

- **Paper by S. Halevi, D. Harnik, B. Pinkas, and A. Shulman.**
  Proposes of POW (proof of ownership) technique is that a user can efficiently prove to the cloud storage server that he/she owns a file without uploading the file itself. It also proposes the Merkle-Hash Tree to enable client-side De-duplication, which include the bounded leakage setting. The proposed scheme is focusing only on the data ownership and not on the data privacy.

- **"A Secure Client Side De-duplication Scheme in Cloud Storage Environment"*.**
  Security and privacy are among top concerns for the public cloud. On Open-Stack Swift, a new

client-side De-duplication scheme. It is originality of proposal is twofold.

-First, it ensures better confidentiality towards unauthorized users.

-Second, by integrating access rights in metadata file, an authorized user can.

It uses Proof of ownership (PoW). The security protocols are designed to guarantee several requirements, namely lightweight of verification and computation efficiency. It based on cloud storage server and merkev tree properties.

- **Paper by Yufeng Wang, Chiu C Tan,Ningfang Mi "Using Elasticity to Improve Inline Data De-duplication Storage Systems"*.**
  In this paper, we illustrate how to use elasticity to improve de-duplication based systems, and propose EAD (elasticity aware de-duplication), an indexing algorithm that uses the ability to dynamically increase memory resources to improve overall de-duplication performance.

3. PROPOSED WORK:

We can take any file as input. Then, we create the chunks of size 4kb and generate fingerprint or hash for each chunk by using SHA1, SHA2, SHA256 and SHA512 to check the efficiency of hashing algorithm for best result to our system. Then, we will implement the Elasticity techniques like if the interdependent chunks are present then we put the single chunks in memory and locate the related chunks i.e. the concept of locality of index.

We will maintain the indexes of each chunk with the relevant user and file and if any duplicate chunks found the we just keep the hash in database but not the chunks. Finally, we recreate the file by mapping all related chunks in database.

With the recent adoption of the data sharing paradigm in distributed systems such as cloud computing, there have been increasing demands & concerns with distributed data security. One of the most challenging issues in data sharing systems is the access policies by user and the support of policies updates. Cipher text policy attribute-based encryption (CP-ABE) is becoming a promising of purity in cryptographic solution to this issue

4. CONCLUSIONS

De-duplication aids in saving the storage space as well as bandwidth. This application helps in easy maintenance of data. The efficiency of the application is dependent on the amount of data present in the storage. Therefore, the data de-duplication application is found to be efficient in eliminating redundant data. All file formats are accepted and deduplicated successfully. Application will test for text, docx, ppt, jpeg, png, sql, ppt, pptx, mp3, mp4, wmv file formats. Since the application will implement on cloud, it can accessed from anywhere, anytime. It gives us a flexibility on how to operate on our data. From future perspective, implementation on cloud gives us the edge to handle BigData. Since, the application is finding duplicate data, large amounts of data have to be processed.

5. REFERENCES

1. J. H. Burrows, "Secure hash standard," DTIC Document, Tech. Rep., 1995.
2. M. Dutch, "Understanding data De-duplication ratios," in *SNIA Data Management Forum*, 2008.
3. D. Geer, "Reducing the storage burden via data De-duplication," *Computer*, 2008.
4. W. Xia, H. d. Jiang *et al.*, "Silo: a similarity-locality based near-exact De-duplication scheme with low ram overhead and high throughput," in *USENIX annual technical conference*, 2011.
5. J. Min, D. Yoon, and Y. Won, "Efficient De-duplication techniques for modern backup operation," *IEEE Transactions on Computers*, 2011.
6. D. Harnik, O. Margalit, D. Naor, D. Sotnikov, and G. Vernik, "Estimation of De-duplication ratios in large data sets," in *Mass Storage Systems and Technologies (MSST), 2012 IEEE 28th Symposium on*, 2012.
7. D. T. Meyer and W. J. Bolosky, "A study of practical De-duplication," *ACM Transactions on Storage (TOS)*, 2012.
8. "Amazon S3, Cloud Computing Storage for Files, Images, Videos," Accessed in 03/2013, http://aws.amazon.com/s3.
9. "OpenfMRI Datasets," Accessed in 05/2013, https://openfmri.org/data-sets.
10. Jin Li, Xiaofeng Chen, Mingqiang Li, Jingwei Li, Patrick P.C. Lee, and Wenjing Lou. *"Secure De-duplication with Efficient and Reliable Convergent Key Management".* In IEEE Transactions On Parallel And Distributed Systems, Vol. 25, No. 6, June 2014.
11. Amrita Upadhyay, Chanchal Dhaker, Pratibha BR, Sarika Hablani, Shashibhushan Ivaturi, *Application of data De-duplication and compression techniques in cloud design,* IIITB, April 2011
12. Jason Buffington, *HP StoreOnce Is "Better Together" with HP Data Protector 7,* Hewlett-Packard, December 2012. Available: http://h20195.www2.hp.com/V2/GetPDF.aspx%2F4A 4-4673ENW.pdf

13. "A SURVEY: DE-DUPLICATION ONTOLOGIES, *International Journal of Computer Applications (0975 – 8887) Volume 109 – No. 1, January 2015.*
14. "A SURVEY ON DE-DUPLICATION METHODS, *International Journal of Computer Trends and Technology- volume3 Issue3- 2012* .
15. "FILE DE-DUPLICATION WITH CLOUD STORAGE FILE SYSTEM, 2013 *IEEE 16th International Conference on Computational Science and Engineering.*
16. "COMPREHENSIVE STUDY OF DATA DE-DUPLICATION, *International Conference on Cloud, Big Data and Trust 2013, Nov 13-15.*
17. 'AN EFFICIENT IMPLEMENTATION OF SHA-1 HASH FUNCTION, *Guoping Wang, Department of Engineering, Indiana University Purdue University, Fort Wayne, IN.*
18. HiTech Whitepaper "Effective Data De-duplication Implementation 05 2011.