ISSN: 2321-7782 (Online)
Impact Factor: 6.047

Volume 4, Issue 6, June 2016

International Journal of Advance Research in Computer Science and Management Studies

Research Article / Survey Paper / Case Study

Available online at: www.ijarcsms.com

Encrypted Deduplication for Efficient Cloud Backup

Pradeep Sarjerao Jagtap[1]
Computer Department
TSSM's Bhivarabai Sawant College of Engineering and Research
Pune – India

Prof. A. D. Gujar[2]
Computer Department
TSSM's Bhivarabai Sawant College of Engineering and Research
Pune – India

Abstract: Data deduplication is a technique for eliminating duplicate copies of data, and has been widely used in cloud storage to reduce storage space and upload bandwidth. Personal computing devices using a cloud storage environment for data backup, facing a challenge where source deduplication for cloud backup service is low deduplication efficiency because of combining resource intensive nature of deduplication and restricted system resources. Now the challenge is to perform secure deduplication in cloud storage. Application aware local global deduplication improves data deduplication efficiency by use of application awareness, and more combines local and global duplicate detection to strike a good balance between cloud storage capacity saving and deduplication time reduction. The prototype implementation to demonstrate that our scheme can significantly improve deduplication efficiency with low system overhead, resulting shortened backup window, increased power efficiency and reduced cost for cloud backup services of personal storage.

Keywords: Cloud computing; Deduplication; chunking scheme; cloud backup; application awareness

I. Introduction

Data is the heart of any organization; hence it is necessary to protect it. Now-a-days, the backup has become the most essential mechanism for any organization. Backing up files can protect against accidental loss of user data, database corruptions, hardware failures, and even natural disasters. Information deduplication a successful innovation for taking out the repetitive information in reinforcement information. Information deduplication innovation partitions the information into little pieces and uses a calculation to allot an extraordinary hash worth to every information lump called unique mark. The calculation takes the lump information as data and produces cryptographic hash esteem as the yield. The most of the time utilized hash calculations are SHA, MD5. These fingerprints are then put away in a file called piece list. The information deduplication framework contrasts each unique mark and every one of the fingerprints as of now put away in the lump record.

The propose methodology of utilization mindful Local-Global source deduplication it figure utilization of use mindfulness as well as consolidates the neighborhood and worldwide copy information identification. ALG-Dedupe plan accomplishes not just higher deduplication effectiveness by lessening deduplication inactivity additionally spares the distributed storage cost. Application mindfulness adjusts distinctive sorts of uses autonomously amid the neighborhood and worldwide copy check process, which lessens the framework. The proposed framework consolidates neighborhood deduplication and worldwide deduplication to adjust the viability and idleness of copy information.

II. Literature Survey

In paper [2], to enhance space use and decrease system blockage, cloud reinforcement merchants (CBVs) dependably execute information deduplication in the source and the destination. Towards coordinating source and destination information deduplication, we mostly propose two recommendations here. One of the vital things of this is advantage taken a toll model for clients to choose in which degree the deduplication executes in customer and in cloud, and let the server farm to choose how to handle the duplications. This will give better unwavering quality, nature of administration and so on. Joining reserving and pre-fetching, and the necessities of various cloud reinforcement benefits, the read execution in the cloud reinforcement frameworks can be moved forward.

Additionally paper [6] lets us know that, Data Deduplication depicts approach that diminishes the capacity limit expected to store information or the information must be exchange on the system. Source Deduplication is valuable in cloud reinforcement that spares system data transfer capacity and decreases system space. Deduplication is the procedure by separating an approaching stream into generally huge fragments and deduplication every portion against just a couple of the most comparable past sections. To recognize comparative portions use square file method The issue is that these plans customarily require a full piece record, which files each lump, so as to figure out which pieces have as of now been put away lamentably, it is unrealistic to keep such a file in RAM and a plate based list with one look for per approaching lump is dreadfully moderate.

In paper [8] the distributed computing is an innovation which is utilized to give assets as an administration. There are numerous administrations gave by cloud supplier. For example: SAAS, IAAS, PAAS. The distributed computing gives the Storage-as-a-Service which is utilized to reinforcement the client's information into cloud. The Storage-as-a-Service is given by Storage Service Provider or Cloud Service Provider. This administration is given by Cloud Service Provider which is viable, dependable and financially savvy. The current reinforcement maintaining so as to plan gives the unwavering quality the same duplicate of the information twice. The current reinforcement booking gives the unwavering quality and reinforcement speed, however the repetition of information is not considered. The current reinforcement booking not considers a significant part of the security issues. The restrictions of the current reinforcement proposing so as to plan calculation is enhanced a reinforcement planning algorithm which goes for decreasing excess without bargaining on accessibility. The IBSD calculation lessens excess by deduplication methods. The deduplication is a method which is utilized to recognize the copy information. The deduplication recognizes the copy information and disposes of it, by putting away one and only duplicate of the first information. On the off chance that the copy happens then the connection will be added to the current information.

III. Proposed System

Overview of the proposed system is shown in Fig. 1 where tiny files are filtered out by file size filter for efficiency, and backup data streams are divided into chunks by intelligent chunker using application aware chunking strategy. Data chunks from the same type of files are deduplicated in the application aware deduplicator by creating chunk fingerprints in hash engine and perform data redundancy check in application aware indices in both local client and remote cloud. Their fingerprints are first looked up in application aware local index that is stored in the local disk for local redundancy check. If the match is found, the metadata for the file containing that chunk is updated at the point location of the existing chunk. When there is no match found, the fingerprint will be sent to the cloud for global duplication check on application aware global index. If match is found in the cloud the corresponding file metadata is updated for duplicate chunks or else the chunk is new.

TABLE 1 File Type respective chunking scheme

File Type	Data Chunking Scheme
Static uncompressed files (Uneditable files) Ex: .pdf, .png, .exe	Static Chunking
Dynamic uncompressed files (Editable files) Ex: .doc, .txt, .ppt	Content Defined Chunking

A. File Size Filter

Many of the files in PC dataset are tiny files that less than 10 KB in file size, that requires small percentage of storage capacity. The storage capacity is very negligible but it increases the overhead on the system. To reduce this overhead, the proposed scheme filters out these tiny files in the file size filter before carrying out the deduplication process. The file size filter then groups data from many tiny files together into larger units of about 1 MB each in segment store to increase data transfer efficiency over WAN.

B. *Intelligent Data Chunking*

The deduplication efficiency of data chunking depends upon the particular data chunking scheme selected. Depending whether the file type is either static or dynamic, the chunking scheme is selected. Following table shows the chunking scheme used for particular file type.

C. *Application Aware Deduplicator*

After data chunking in the intelligent chunker module, data chunks will be deduplicated in the application aware deduplicator by creating chunk fingerprints in the hash engine and detecting duplicate chunks in both the local client and remote cloud ALG-Dedupe requires two application-aware chunk indices: a local index on the client side and a global index on the cloud side.

D. *SHA-1 (Secure Hash Algorithm-1)*

In cryptography, SHA-1 is cryptographic hash function with 160-bit hash value. A SHA-1 hash value is typically rendered as a hexadecimal number, 40 digits long.

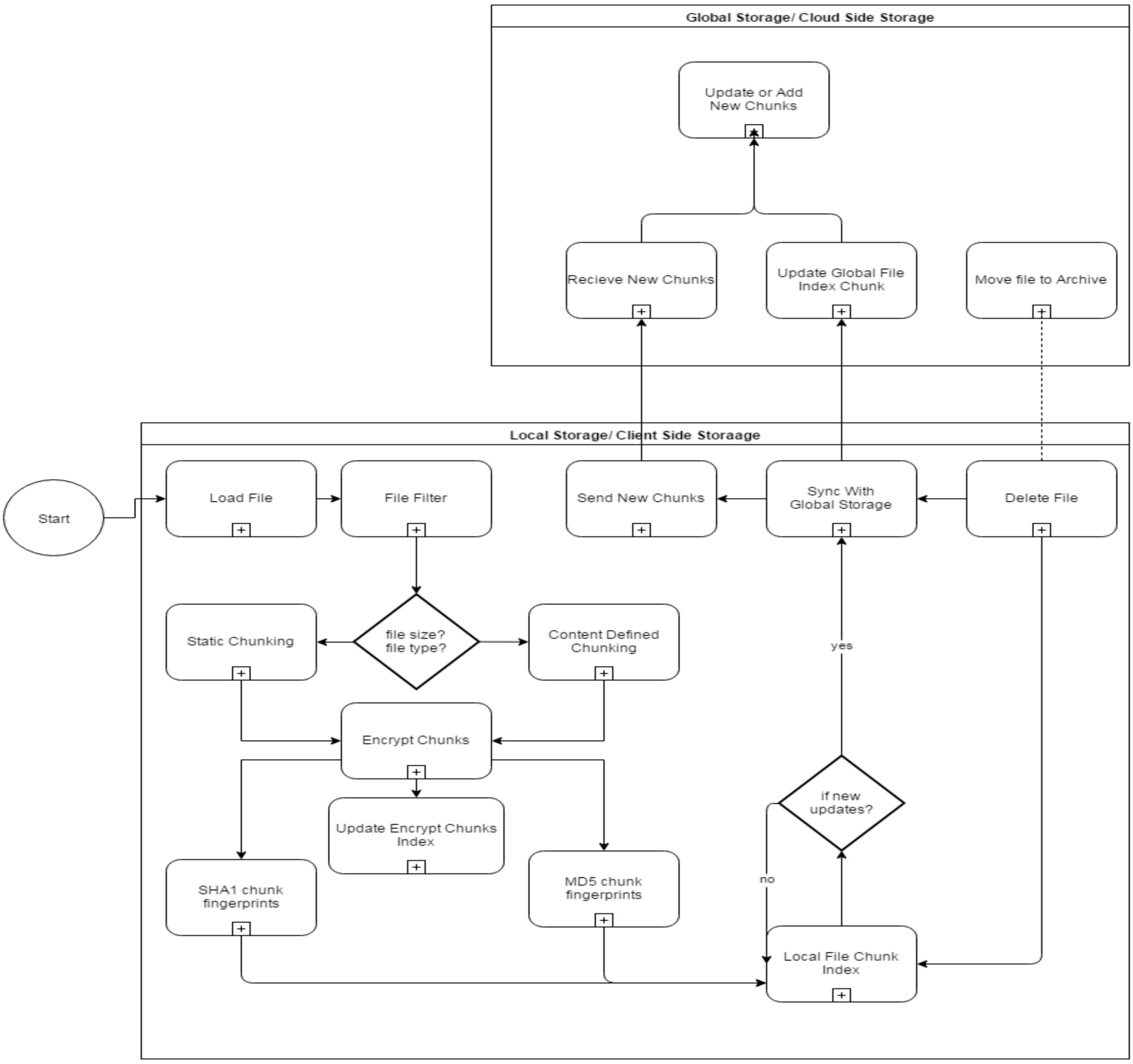

Fig. 1 System Architecture

E. MD5 (Message-Digest 5)

It is widely used cryptographic function with 128- bit hash value. MD5 has been employed in wide variety of security applications. MD5 is commonly used to check the integrity of files. An MD5 hash is typically expressed as a 32-digit hexadecimal number.

To enhance the security of proposed system the data stored on global storage will be kept in encrypted form. For encryption convergent encryption technique is used where the key for encryption will be generated from the data present in the file. The generated keys will be stored on local storage. During decryption, the required encrypted files from global storage are brought to the local storage and then using keys the files are decrypted.

F. Rabin Karp Rolling Hash

The Rabin Karp rolling hash algorithm is efficient in finding a pattern in a large file or stream, but there is a use case: creating content based chunks of file to detect the changed blocks without carrying full byte by byte comparison. The traditional way in computation of the hashes interleaving windows generally works like this:

Input: [a,b,c,d,e]

Window size: 3

Hash#1: hash ([a,b,c])

Hash#2: hash ([b,c,d])

Hash#3: hash ([c,d,e])

The Rabin-Karp alogorithm very efficiently reuses the previous hash to calculate the new one:

Hash#1 = hash([a,b,c])

Hash#2 = Hash#1 - hash(a) + hash(d)

Hash#3 = Hash#2 - hash(b) + hash(e)

So, to calculate the next window's hash remove the window's first element's 'hash value' from the current hash and add the new to it. Rabin-Karp algorithm typically works with the powers of a prime number to do calculate the hash.

$p^{(n)}*a[x] + p^{(n-1)}*a[x-1] + ... + a[0]$

The nice thing about it is it requires constant number of operations and does not depend on the size of the window.

IV. Experimental Result

The optimal combination of chunking and hash fingerprinting methods can reduce system overheads on resource-limited personal computing devices. As shown in Fig. 2, the deduplication strategy based on simpler chunking schemes can achieve a higher throughput because of lower metadata storage and chunking overheads, while deduplication strategies with weaker hash function (example Rabin hash) obtain a higher throughput because of their lower computational overhead. Furthermore, the response time of Rabin hash and MD5 is less than that of SHA-1. This suggests that we can employ the extended Rabin hash value chunk fingerprint for local duplicate detection and MD5 for global duplicate detection on compressed files to reduce the computational overhead with less probability of hash collision.

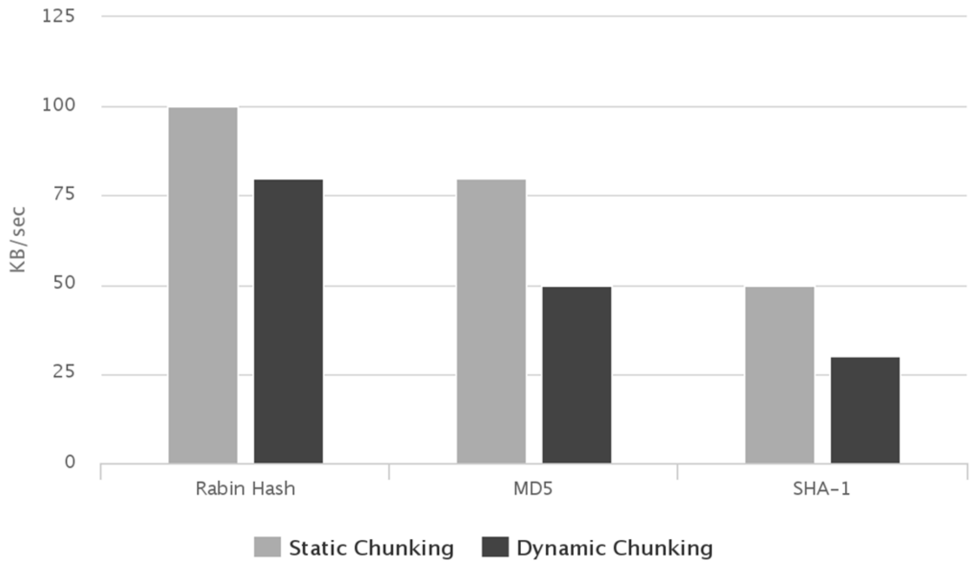

Fig. 2 Throughput of Chunking and fingerprinting.

As Fig. 3 shows the chunking speed comparison as the file size increases the time taken for chunking also increases. The dynamic chunking requires less time as compared to static chunking.

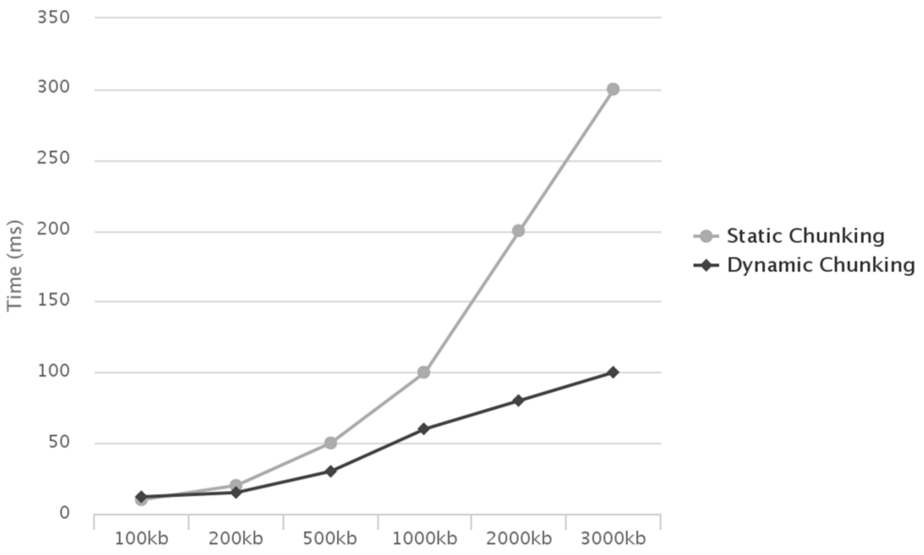

Fig. 3 Chunking speed comparison

Considering the limited system resources in clients, we estimate the system overhead in terms of CPU processing speed and usage of RAM for source deduplication based cloud backup services in personal devices. In our design, we adaptively select the chunking method and hash function for different application data subsets to achieve high deduplication efficiency with low system overhead.

V. Conclusion

The ALG-Dedupe, application aware local global source deduplication for cloud backup in the personal computing environment to improve deduplication efficiency is proposed. An intelligent deduplication strategy in ALG-Dedupe is designed to exploit file semantics to minimize the computational overhead and maximize the deduplication effectiveness using application awareness. It combines the local deduplication and global deduplication to balance the effectiveness and latency of deduplication. The proposed system performs deduplication on static uncompressed files and dynamic uncompressed files. The future scope of the proposed system is that in near future the system can be implemented for compressed files such as .zip, video files, etc. Whole File Chunking can be used for this purpose.

ACKNOWLEDGEMENT

I have taken efforts in this review of deduplication for cloud backup services of personal storage. However, it would not have been possible without the kind support and help of many individuals. I am highly indebted to Prof. A. D. Gujar for his guidance and constant supervision as well as for providing necessary information regarding this approach.

References

1. Y.Fu, H.Jiang,"Application-Aware Local-Global Source Deduplication for Cloud Backup Services of Personal Storage," IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, VOL. 25, NO 5, MAY 2014
2. Xiongzi Ge, Zhichao Cao, "Data Deduplication in Cloud Backup System", Final Report Computer Science and Engineering, University of Minnesota, Twin Cities.
3. A. Katiyar and J. Weissman, ''ViDeDup: An Application-Aware Framework for Video De-Duplication,'' in Proc. 3rd USENIX Workshop Hot-Storage File Syst., 2011.
4. C. Liu, Y. Lu, C. Shi, G. Lu, D. Du, and D.-S. Wang, ''ADMAD: Application-Driven Metadata Aware De-Deduplication Archival Storage Systems,'' in Proc. 5th IEEE Int'l Workshop SNAPI I/Os, 2008, pp. 29-35.
5. S. Kannan, A. Gavrilovska, and K. Schwan, ''Cloud4HomeV Enhancing Data Services with @Home Clouds,'' in Proc. 31st ICDCS, 2011, pp. 539-548.
6. Y. Fu, H. Jiang, N. Xiao, L. Tian, and F. Liu, ''AA Dedupe: An Application-Aware Source Deduplication Approach for Cloud Backup Services in the Personal Computing Environment,'' in Proc. 13th IEEE Int'l Conf. CLUSTER Comput., 2011.
7. P. Anderson and L. Zhang, ''Fast and Secure Laptop Backups with Encrypted De-Duplication,'' in Proc. 24th Int'l Conf. LISA, 2010, pp. 29-40.
8. "Improved Backup Scheduling With Data Deduplication Technique For Saas In Cloud" by Tamilselvi.T, K. Saruladha Department of Distributed Computing System, Pondicherry Engineering College, Pondicherry.

AUTHOR(S) PROFILE

Pradeep Sarjerao Jagtap, is currently pursuing M.E (Computer) from Department of Computer Engineering, TSSM's Bhivarabai Sawant College of Engineering and Research, Pune, India at Savitribai Phule Pune University. He received his B.E (Information Technology) Degree from Annasaheb Dange College of Engineering & Technology, Ashta, India at Shivaji University Kolhapur. His area of interest is cloud computing.

Anil Gujar, received the M.Tech (IT) degree from the Department of Information Technology, Bharati Vidyapeeth Pune, India. He is currently working as Asst. Professor with Department of Computer Engineering, Bhivarabai Sawant College of Engineering and Research, Pune, India.