Analysis of De-Duplication Methods in Cloud Computing

Preetika Singh
Student, M. Tech CSE
Sri Sai college of Engg and Technology ,Badhani.

Meenakshi Sharma
Head of Deptt, CSE
Sri Sai college of Engg and Technology ,Badhani

Deepika Singh
Assistant Professor
Sri Sai college of Engg and Technology ,Badhani

ABSTRACT
Cloud computing has helped in minimizing the need for extra storage at the user's premises. De-duplication has largely contributed to the emergence of cloud computing as an efficient storage medium at small as well as large scale. However, De-duplication has invited serious concerns related to security as well as speed of access. Considering the amount of storage efficiency increased due to this method it becomes undesirable to avoid its use. This paper studies the emergence of these concerns and discusses the solutions proposed. Further the loopholes of solutions proposed and the future work that can be done in this aspect is studied.

General Terms
Cloud computing; De-duplication; speed; efficiency; security**.**

Keywords
Cloud computing; De-duplication; speed; efficiency; security.

1. INTRODUCTION
Over the past few decades the networks have become denser and the data usage, wider. As the world is shrinks, data dependency of human being is increasing. No one has been left untouched with the data storage requirement increase. Scientific institutions, business applications, social networking, and various industries, these have contributed to increasing data storage space requirement. The increased storage space requirement has led to increased energy requirement, leading to increased carbon emission, leading to environmental hazards and so on.

Among the various techniques dedicated towards the solution of this problem, cloud computing is the highest scorer. Cloud computing, is a pay per use method of computing, which is highly elastic, reliable and easy to use. Cloud computing provides the services for storage, platform and infrastructure. However even clouds are suffering with the high data storage space requirement. The De-duplication technique is used with cloud computing to provide efficient storage solutions.

De-duplication was introduced as a solution to the redundant data storage in the clouds. When several users use the same data, its actual storage can be single and the data can be distributed among the users. The user's identity gets added to the access control list of the files containing the data. For example, consider the case of a private company, if there is a set of instructions, given by a commanding authority in the form of a presentation, and it has to be followed by each employee, than each one will save the presentation on the same private cloud used by the company. Even if the presentation's actual size is 50 kb and the company has 100 employees the actual storage requirement counts to be 5000 KB, more than about 4GB. In order to stop the wastage of this vast amount of data storage space, De-duplication was introduced. Figure 1[5] gives an overview of the process of De-duplication.

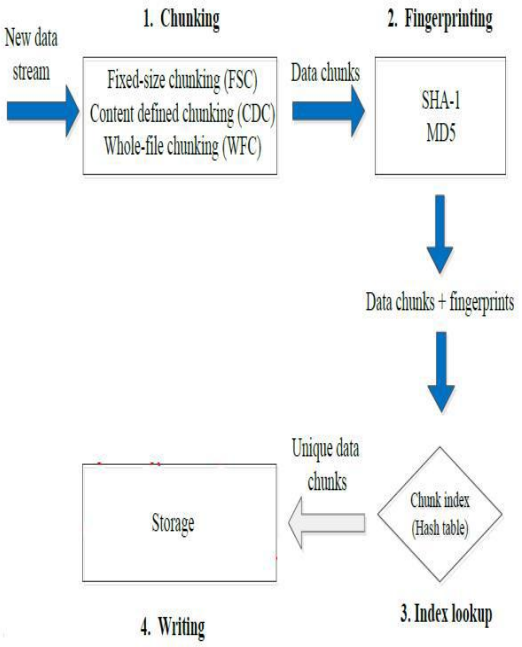

Figure 1[5]: The process of De-duplication

De-duplication when introduced was done at the server premises, rather than on the client's. Further it was observed that the network bandwidth required for sending a particular file again and again was wasted, so client-side De-duplication was introduced.

The unit of De-duplication remained a file unless it was observed that the redundancy of data storage can even be reduced at a much smaller unit, a file divided into blocks. The former was called file level De-duplication while the latter was block level De-duplication. In file level De-duplication, the file is first hashed with the help of any hashing algorithm like SHA or MD5 further an index is created in which the hash value and the corresponding file name is stored. This index is used to find out if any file is present already. If the hash value of the file is already present, the unique user ID of the person uploading the file is added to the access control list of the file otherwise, the file is uploaded. The block level De-duplication does the same with the blocks. The only difference is that the hash value is calculated at the block level. Figure 2 [10] shows the iconic visualization of block based De-duplication system.

This paper studies various approaches to De-duplication and analyses them based on their advantages and disadvantages. Section II addresses various disadvantages associated with the implementation of De-duplication technology. Section III discusses the advancements in the De-duplication to cope up with its disadvantages. Section IV analyses the proposed systems discussed in section IV.

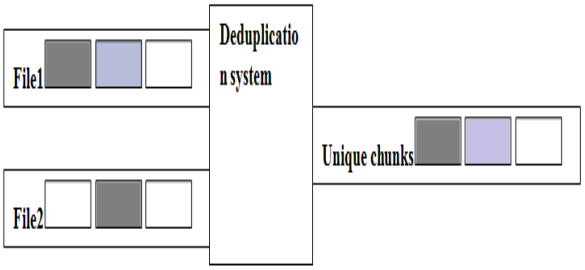

Figure 2[10]: Iconic visualization of block based De-duplication.

2. THE PROBLEM DOMAIN

One side of De-duplication deals with lots of storage space saving promises while the other one show a handful of disadvantages, the systems suffer while using De-duplication. Security, speed and efficiency remain top scorer among them.

Security is the biggest problem domain in cloud computing system, and it increases its value with De-duplication facility. Side channels created while implementing De-duplication let an attacker use the system or any other user's data [3]. The side channels are created when an attacker places his virtual machine on the same hypervisor as that of the victim's. Side channels can be used to extract data while using De-duplication. Various possible attacks were listed [3] which reveal that it is possible to extract confidential information related to a user while using side channels.

It is possible to find out whether a particular file is possessed by a user or not by using side channels. Suppose user 1 uses cloud storage and the attacker places his virtual machine on the same. The attacker wants to know if a confidential file, file1 is possessed by the user 1. He will simply upload the file1 and while keeping an eye on the network bandwidth usage, if the file will be uploaded this means that the file is already present on the server and probably user 1 posses the file. Another variant of this attack is Probing attack; it uses the concept of probability. If file 1 is possessed by user 1 with some variations having limited possibility, attacker can use the possibilities to find the variation by the method described above. An example of this attack can be, finding the report of a medical test where the result has only two possibilities, positive or negative and the format of the test report is already known to the attacker. Secondly, if any attacker calculates the hash value of a file, as the hashing algorithm is already known to the users, he can distribute it among the various users and users can have access to the file, the example could be any bootlegged video. Another possible attack could be related to the injection of software into the user's machine which is designed by the attacker. This software can reveal the information related to the user's work to the attacker if they use same hypervisor. The example could be to find out if any employee had been watching a video or audio file during working hours. This is called as covert channel or secret channel attack. These security concerns had been addressed largely and solutions have been proposed, discussed in the next section.

The next issue related to the De-duplication usage is speed of access of data. The De-duplication method when implemented on a large number of file produces a considerable amount of delay while uploading and downloading the file, as the file has to undergo the process of blocking, hashing, indexing etc. Moreover, fetching a file is much more time consuming than uploading it, as public clouds store innumerable files and De-duplicate them to the highest possible extent.

The De-duplication method injection invites serious problems in cloud computing system, but when we take into concern the amount of storage space saved by this technology, its ignorance becomes avoidable. Keeping this in mind a number of security and efficiency, increase solutions have been proposed and studied in recent years.

3. SOLUTIONS PROPOSED

A rigorous research in the field of De-duplication technology is going on to overcome the disadvantages discussed above. Random De-duplication method [3] was proposed to deal with the possibility of finding the contents of the file. Every file in this method was assigned a threshold value randomly; the file was De-duplicated only when the number of similar files exceeded the threshold value. This approach, however did not promise reliability. So next gateway based De-duplication was proposed [12] which ensured the upload and download of file only through a particular gateway. This method aimed towards providing a transparency to the De-duplication process, here the user remains unaware of any De-duplication taking place and thus the probing attack becomes impossible to implement. But, looking at the need of extra hardware required a proof of ownership [11] method was proposed. This method devised towards asking for a random block of the requested file from the user in order to ensure that the requester is the actual holder of the file. However, proof of ownership method removed the chances of all the attacks described above, but reliability still remains a question in this aspect.

Regarding speeding up De-duplication process the study started with using block level De-duplication in place of file level De-duplication. The blocks were further divided based on their contents in content defined chunking [6] it was used with the low bandwidth file system, LBFS. Further the problem of input, output delay due to large amount of dataset present in RAM as an index (hash values) was addressed. A system [8] was proposed a system was proposed which used SSD (Solid State Drives) to store the index value. This storage system was flash aware and hence produced the lesser delay, but didn't supported scalable system and required extra hardware. A centralized block index was proposed [9] which avoid the usage of full hash values by sparse indexing by using the method of sampling.

4. RESULT AND CONCLUSION

Based on the above study, the following table summarizes various problems and solutions discovered and proposed respectively in the domain of De-duplication, also; the problems discussed above are taken into concern according to their disadvantages. Table 1 formulates that instead of rigorous research done towards De-duplication, the problems still pertain. The setup proposed in the paper, provides a trust based system which can help eradicating the flaws in the system, by increasing the De-duplication speed and efficiency. By using this method, the speed as well as efficiency of the De-duplication system, both can be increased. It was observed that this type of implementation will be more useful in the medical systems, where files like patient reports vary a little. Regarding security, a trust based deduplication system can be proposed to establish trust between the user and the Storage Service Provider, (SSP).

Table 1: Problems discussed and solutions proposed

Concern	Problem	Solution proposed	Disadvantage
Security	Probing attack	Gateway based deduplication [12]	Extra hardware required
	Finding the contents of file (predicting files)	Randomisation solution [3]	Unreliable
	Side channel attack	Randomisation solution [3]	Unreliable
	Content distribution network attack	Proof of ownership [11]	Unreliable
Speed and efficiency	Uploading and Downloading speed of file	Content defined chunking [6]	Can be used in low bandwidth file system (LBFS) only
		Solid state drives (SSD) [8]	Scalability issue
		Sparse indexing [9]	Only for centralized systems.

5. REFERENCES

[1] D. Russell, "Data De-duplication Will Be Even Bigger in 2010", Gartner, February 2010.

[2] M. Dutch. Understanding data De-duplication ratios. White paper, June 2008.

[3] D. Harnik, B. Pinkas, and A. Shulman-Peleg. Side Channels in Cloud Services. De-duplication in Cloud Storage. IEEE Security & Privacy, 8(6), Nov. 2010.

[4] Sun, Z., Shen, J. & Yong, J. (2013). A novel approach to data De-duplication over the engineering-oriented cloud systems. Integrated Computer Aided Engineering, 20 (1), 45-57.

[5] Mkandawire, Stephen, "Improving Backup and Restore Performance for De-duplication-based Cloud Backup Services" (2012). Computer science and Engineering: Theses, Dissertations, and Student Research. Paper 39.

[6] A. Muthitacharoen, B. Chen, and D. Mazieres, "A low-bandwidth network file system," in Symposium on Operating Systems Principles, 2001, pp. 174–187. [Online]. Available: http://citeseer.ist.psu.edu/muthitacharoen01lowbandwidth.html

[7] S. Quinlan and S. Dorward, "Venti: a new approach to archival storage," in First USENIX conference on File and Storage Technologies, Monterey,CA, 2002. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.18.8085

[8] D. Meister and A. Brinkmann. dedupv1: Improving De-duplication throughput using solid state drives (SSD). In IEEE 26th Symposium on Mass Storage Systems and Technologies (MSST), pages 1-6, 2010.

[9] D. Bhagwat, K. Eshghi, D. D. E. Long, and M. Lillibridge. Extreme Binning: Scalable, parallel De-duplication for chunk-based file backup. In IEEE International Symposium on Modelling, Analysis & Simulation of Computer and Telecommunication Systems (MASCOTS), 2009.

[10] Burton H. Bloom. Space/time trade-offs in hash coding with allowable errors. Communications of the ACM, 13(7):422–426, 1970.

[11] S. Halevi, D. Harnik, B. Pinkas, and A. Shulman-Peleg, "Proofs of ownership in remote storage systems," in Proc. of CCS'11, 2011.