Data De-duplication Approach based on Hashing Techniques for Reducing Time Consumption over a Cloud Network

Manreet Kaur
M.Tech (IS) Scholar
Department of Information Technology
CEC Landran, Mohali, Punjab, India

Jaspreet Singh
Assistant Professor
Department of Information Technology
CEC Landran, Mohali, Punjab, India

ABSTRACT
Data de-duplication is a developing and widely engaged method for recent storage systems. Cloud storage is an isolated storage service, where users can upload and transfer their data anytime and anywhere. However, it raises problems regarding privacy and data secrecy because all the data are stored in the cloud storage. This is a focus of concern for users, and it affects their disposition to use cloud storage services. On the other hand, a cloud storage server classically performs a particular data de-duplication to remove duplicate data because the storage space is not infinite. Data de-duplication, which makes it possible for data possessors to share a copy of the same data, can be achieved to reduce the hashing time, memory consumption and detection time and accuracy. This study proposes a novel de-duplication MD5, SHA-1 and SHA-2 Hybridization. Due to the above concerns, there is a research on data de-duplication. In this script, we propose a hashing data de-duplication mechanism which makes the cloud storage server be able to abolish duplicate improves the privacy protection.

Keywords
Data De-duplication, MD-5, SHA1 and Enhanced the SHA-2 Algorithm, Cloud computing and security.

1. INTRODUCTION
In current years, cloud computing has become a hot topic and conveys many benefits through various services. The difficult hardware, database, and operating system can be handled by a cloud server. Users only need some simple devices, which can connect [1] to the cloud server. However, in the environment, the cloud server can attain and control all the uploaded data since all the data are stored or functioned in the cloud. The security and secrecy issues are very significant in cloud computing. In order to defend privacy, users hash their data by some hashing algorithms and upload the hybrid hashing data to the cloud. As a result, widespread research has been performed in cloud computing. Like memory consumption, Detection Time and Accuracy. Users can store their data in the cloud storage and transfer the stored data anywhere. Even if users consume their own storage spaces, the cloud storage server can increase the storage spaces without finishing the stored data. However, the fast growth of storage necessities burdens the cloud storage, which is not infinite. The cloud storage server typically applies the data de-duplication technique [2] to reduce the consumption of memory.

2. DE-DUPLICATION
Data de-duplication is a specific data firmness method which makes all the data owners, which upload the same data, share a particular copy of duplicate data and removes the duplicate copies in the storage. When users upload their data, the cloud storage server will check whether the uploaded data have been deposited or not. If the data have not been stored, it will be really written in the storage; otherwise, the cloud storage server only stores a pole, which points to the first stored copy, instead of storing the whole data. Hence, it can avoid the same data being stored recurrently. Generally, data de-duplication can be divided into two basic approaches [3]:

1) Target-based data de-duplication
2) Source-based data de-duplication.

The two methods of data de-duplication are described as follows:

Target-Based Data De-duplication	Source Based Data De-duplication
Improve the storage operation, and users do not have to change their ways of using cloud storage services [5].	If the data have not been deposited, users need to upload the whole data, and the cloud storage server entirely stores [4] them. Otherwise, users need to upload only the metadata, and the cloud storage server simply creates a pointer, which points to the first stored copy.
Target based approach only focuses on escaping storing duplicate data. Those duplicate data are still uploaded repeatedly. Therefore, it cannot advance the volume of transmissions.	Source-based approach can advance both the application of the storage and the bandwidth. Nevertheless, it changes the familiar process of cloud storage services. When users want to upload their data, they must enquiry the cloud storage server for the reality of the data first [6].

3. SECURITY IN DATA DEDUPLICATION

Malicious User: It is an exceptional enemy in source based data de-duplication. In a source-based data de-duplication scheme, each user inquiries the cloud [7] storage server whether the data have been uploaded by another. The cloud storage server responds ”Yes” or ”No” reliably to avoid the duplicate data being uploaded constantly. Therefore, a malicious user can use the answer of the cloud storage server to obtain isolated information about the existence of data.

Cloud Storage Server: In cloud storage service, the cloud storage server can attain and control all the uploaded data. In addition, if encoded data [8] de-duplication can be achieved, even though all the data are encoded, the cloud storage server can still examine the encoded data by using the encoded data de-duplication and try to obtain information

4. RELATED WORK

K. Saritha et al., 2015 [9]Presented this approved duplicate check in hybrid cloud architecture. The hybrid cloud building suggests about both the public cloud and the private cloud. The private cloud plays an significant role in this system, moreover the security of this scheme is less, as the private cloud of this mechanism is not secure, unsanctioned access of the data resulting in fail in security. In order to offer more security, the private cloud is provided with multilevel authentication. **Backialakshmi.N et a., 2015 [10]**Described to outstandingly reduce the amount of duplicates in one type of No SQL DBs, namely the key-value store, to maximally increase the process presentation such that the backup window is slightly affected, and to design with straight scaling in mind such that it would run on a Cloud Platform competitively. **Vandana Dixit Kaushik et al.,2014 [11]**Proposed definite rules which help to enterprise efficient algorithm for de-duplication which is based on Indian demographic information that comprising two name strings, viz. Given Name and Surname, of individuals. Rules help to diminish all name strings to generic name strings. A bin is formed by the common name which contains all term strings and their Ids. Thus, the folder with demographic information contains of an array of bins and each bin is considered by a singly linked list. At the time of query, top n best matches are resolute by searching all adjoining bins of the reduced query name strings

.

5. SIMULATION MODEL

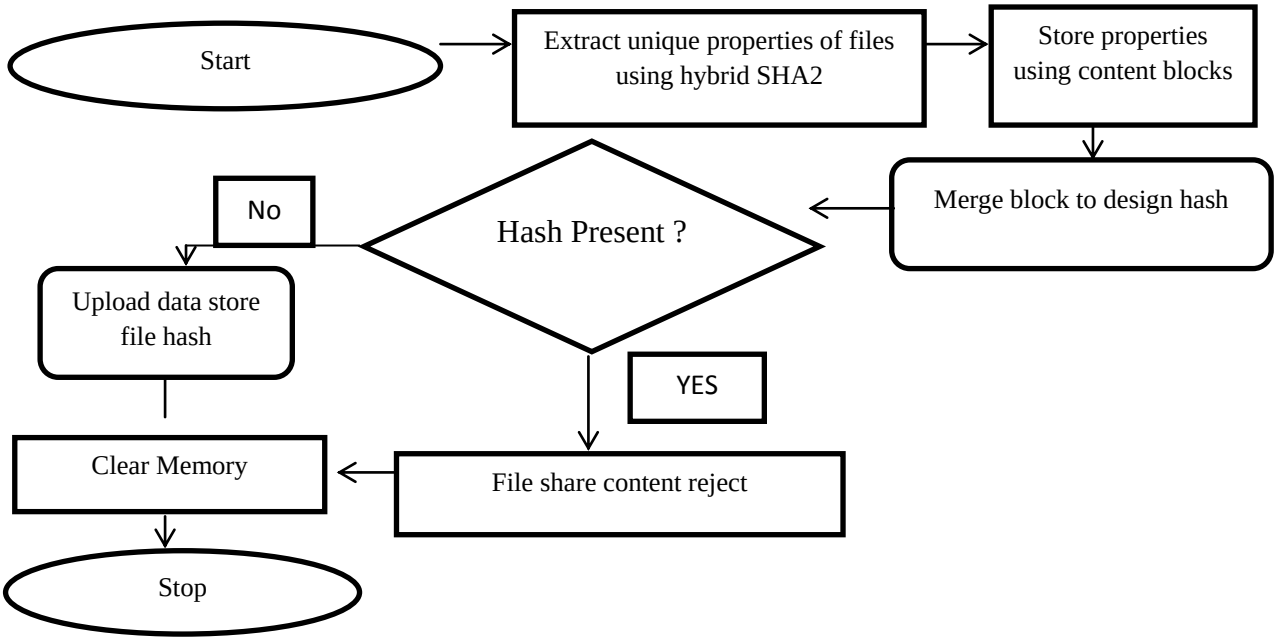
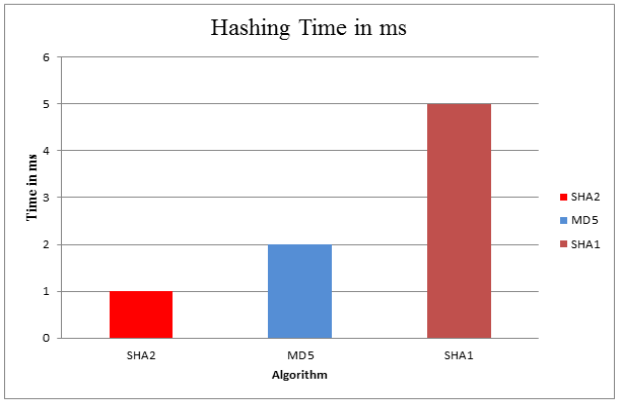


Fig: 1 Proposed Work

In this Simulation Model describe the whole architecture of the de-duplication. First start the process and move to extract unique properties of the files using hybrid SHA-2. To store the Properties using the content blocks. The content block merges to design hash. If hash is present in content block then share the file content reject. If not present then upload the data supply in hash file and to clear the memory.

6. RESULTS AND DISCUSSIONS

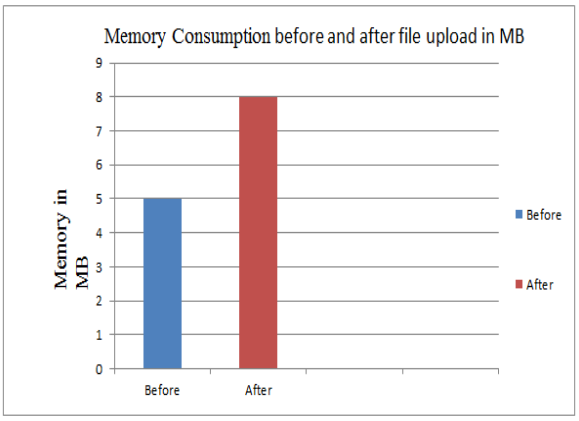
Below graph shows the hashing time by using three different algorithms . it is clear from this graph that SHA2 take less time as compared to MD5 and SHA1. That means SHA2 is better than MD5 and SHA1 as it takes much less time .



SHA2	MD5	SHA1
0.5	0.5	1.0
0.5	1	2.1
0.8	1.5	3.3
1	1.8	4
1.1	2	5

Fig 2 : Hashing Time in milliseconds

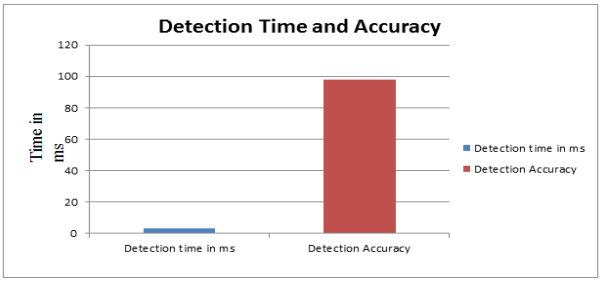
The below graph shows the memory used before file upload and after file upload. It is clear from the graph that the memory space is increased when we upload the new file in database. But when duplicate file is detected by using hashing algorithm then there is no effect on memory space it is same as before. In this way by using De-duplication memory space is less consumed.



Before	After
0.5	0.8
0.9	1.3
1.4	1.8
2.5	2.5
3.7	3.9
4	4.8
5	5.4
0	6.7
0	7.9
0	8

Fig 3 : Memory Consumption Before and After

Accuracy shows that how accurately our system works to detect the duplicate files. From the below graph we can conclude that duplicator detect the duplicate file in less time and perform accurately. Detection time is a time taken to detect a duplicate file and it is also clear from the graph that it takes very less time to detect a file.



Detection Time	Detection Accuracy
0.9	20
3	40
5	60
8	80
10	98

Fig 4: Detection Time and Accuracy

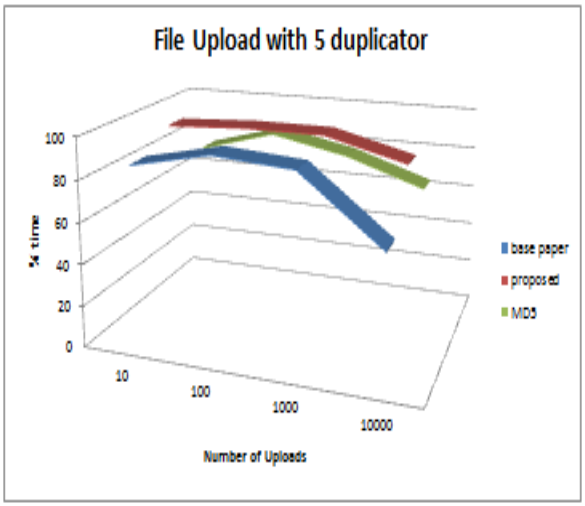


Fig 5 : Upload using 5 duplicators

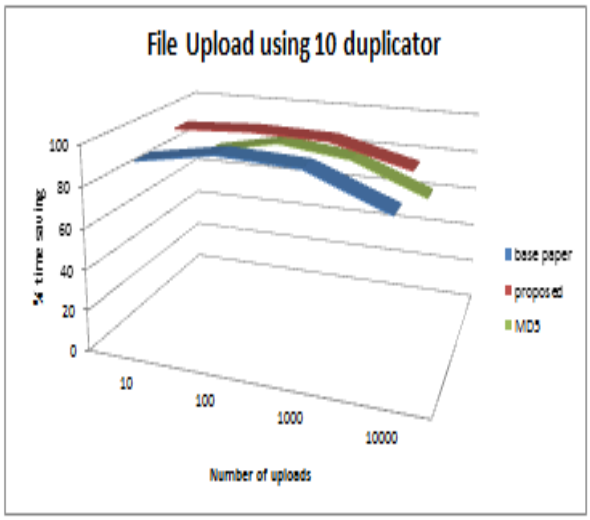


Fig 6 : Upload using 10 Duplicators

These are the graphs of the below table by using 5 and 10 duplicators. In which time saving is shown by using Base paper parameters, MD5, SHA1 and SHA2 algorithms when user uploads 10, 100, 1000 and 10000 files. By using hybrid (SHA1-SHA-2) algorithms time saving is more.

FILE UPLOAD PARAMS

Number of Uploads	5 Duplicators			10 Duplicators		
	md5	Previous	Hybrid SHA2	md5	Previous	Hybrid SHA2
Time saving using 10 uploads	74	85.75	95.17	73	90.85	94.25
Time saving using 100 uploads	84	94.20	97.15	84	97.55	98.11
Time saving using 1000 uploads	77	91.40	97.15	78	95.58	97.28
Time saving using 10000 uploads	65	60.10	88.18	62	79.71	89.12

Fig 7: File upload Parameters

The Result table shows the time required to upload of files by using 5 and 10 duplicators. In which time saving is shown by using 10, 100, 1000 and 10000 Uploads, updates and deletion. By using MD5 and SHA 1 algorithm we show that time saving of these parameters.

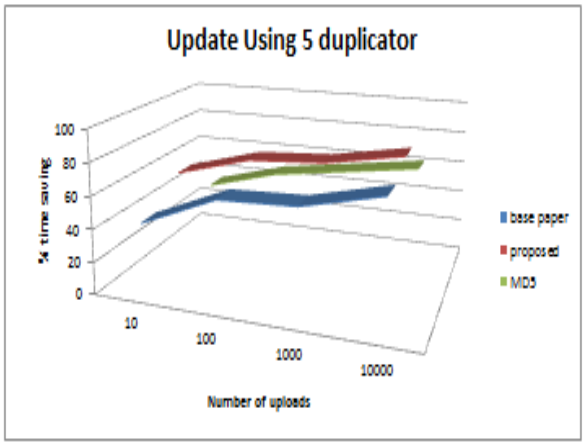


Fig 8: Update using 5 duplicator

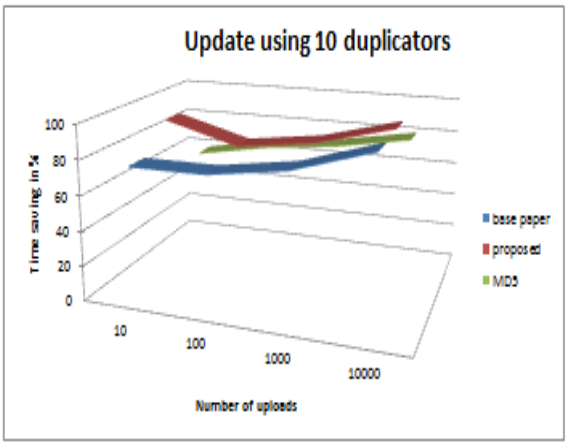


Fig 9: Update using 10 duplicator

The above graphs show the update time by using 5 and 10 duplicators. In which time saving is shown by using Base paper parameters, MD5, SHA1 and SHA2 algorithms when user update 10, 100, 1000 and 10000 files. By using Hybrid algorithms time saving is more.

FILE UPDATE PARAMS

Number of Updates	5 Duplicators			10 Duplicators		
	md5	Previous	Hybrid SHA2	md5	Previous	Hybrid SHA2
Time saving using 10 Updates	34	41.78	60.26	56	75.02	89.17
Time saving using 100 Updates	45	61.79	72.27	66	75.34	77.17
Time saving using 1000 Updates	56	63.78	73.16	75	82.09	85.18
Time saving using 10000 Updates	62	75.25	82.19	75	96.17	97.15

Fig 10: File Update Parameters

The Result Table shows the time required to update of files by using 5 and 10 duplicators. In which time saving is shown by using 10, 100, 1000 and 10000 Uploads, updates and deletion. By using MD5 and SHA 1 algorithm we show that time saving of these parameter

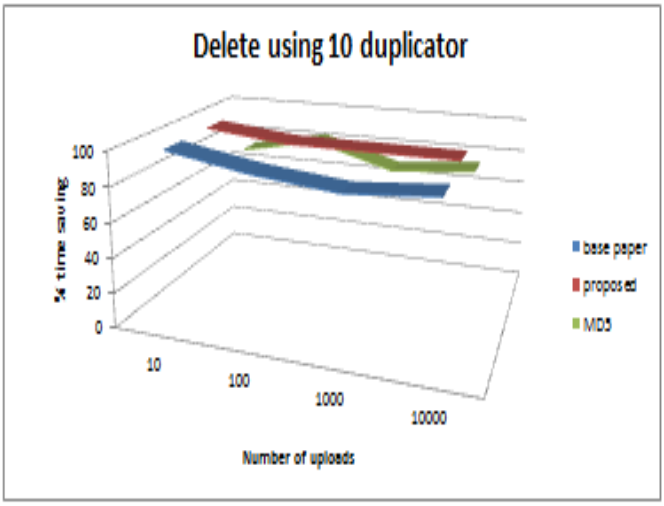


Fig 11: Delete using 5 duplicator

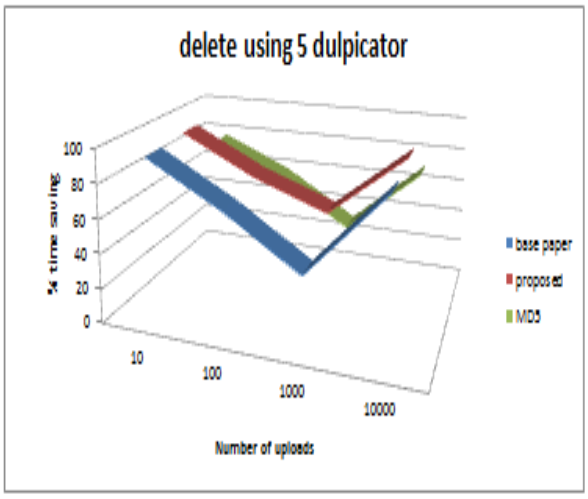


Fig 12: Delete using 10 Duplicator

The above graphs shows the delete time by using 5 and 10 duplicators. In which time saving is shown by using Base paper parameters, MD5, SHA1 and SHA2 algorithms when user upload 10, 100, 1000 and 10000 files. By using Hybrid algorithms time saving is more.

FILE DELETE PARAMS

Number of DELETE	5 Duplicators			10 Duplicators		
	md5	Previous	Hybrid SHA2	md5	Previous	Hybrid SHA2
Time saving using 10 DELETE	76	93.42	95.18	77	98.68	99.15
Time saving using 100 DELETE	59	69.31	72.16	84	90.59	94.17
Time saving using 1000 DELETE	33	40.74	57.17	71	85.87	94.18
Time saving using 10000 DELETE	70	90.28	94.15	75	90.03	96.16

Fig 13: File Delete Parameters

These snapshots shows the time required to delete of files by using 5 and 10 duplicators. In which time saving is shown by using 10, 100, 1000 and 10000 Uploads, updates and deletion. By using MD5 and SHA 1 algorithm we show that time saving of these parameters.

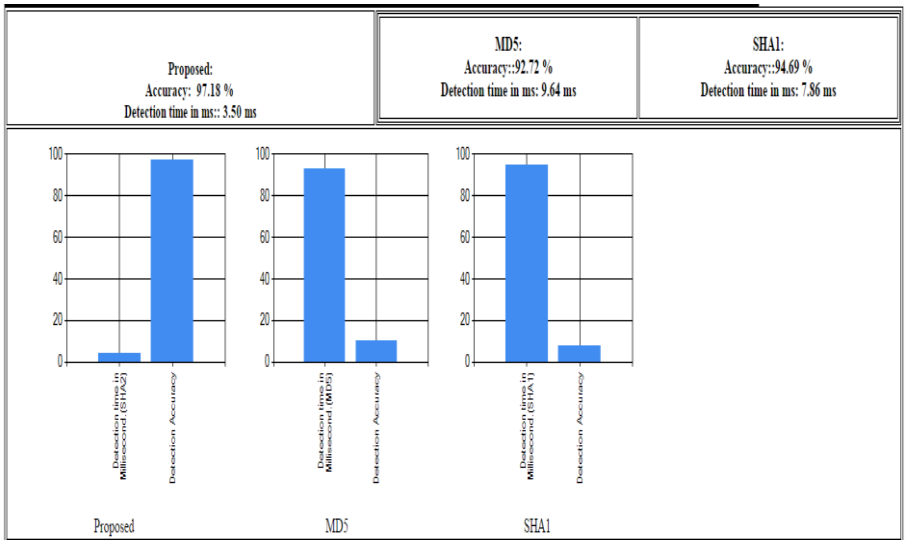


Fig 14 : Proposed Accuracy

This graph shows that the proposed hybrid SHA2 accuracy is 97.18%, MD-5 Accuracy is 92.72% and SHA-1 accuracy is 94.69%.

7. CONCLUSION

Cloud is the costly storage provider, so the motivation is to use its storage area efficiently. De-duplication has been proved to reduce memory consumption by removing the useless duplicate files. So far from the previous studies file level de-duplication is the better approach to be used, the focus of the proposed work will be on file level de-duplication. Our aim is to choose a well-built hybrid algorithm which will generate a good hash value in turn reducing cloud storage. In this proposed work the use of Microsoft azure provides the replica of the cloud computing environment which is used by many companies. Thus the work can easily be accomplished by the use of cloud framework without any cost consumption usage. In Future Scope , an improved technique for storage has been tested only for text, PDF and doc files. In future, it can be further extended to support files of other type's i.e. video and audio files.

8. REFERENCES

[1] J. Harauz, L. M. Kaufman, and B. Potter, "Data security in the world of cloud computing," IEEE Security and Privacy, vol. 7, no. 4, pp. 61–64, 2009.

[2] Q. He, Z. Li, and X. Zhang, "Data deduplication techniques," in Proc. 2010 Int. Conf. on Future Information Technology and Management Engineering (FITME 2010), 2010, pp. 430–433.

[3] D. Harnik, B. Pinkas, and A. Shulman-Peleg, "Side channels in cloud services: Deduplication in cloud storage," IEEE Security and Privacy, vol. 8, no. 6, pp. 40–47, 2010.

[4] C. Liu, D. Ju, Y. Gu, Y. Zhang, D. Wang, and D. Du, "Semantic data de-duplication for archival storage systems," in Proc. 13th IEEE Asia-Pacific Computer Systems Architecture Conference (ACSAC 2008), 2008, pp. 1–9.

[5] Luo, Shengmei, et al. "Boafft: Distributed Deduplication for Big Data Storage in the Cloud."IEEE TRANSACTIONS ON CLOUD COMPUTING, VOL. 61, NO. 11, JANUARY 2015.

[6] Wang, Jibin, et al. "I-sieve: an inline high performance deduplication system used in cloud storage." Tsinghua Science and Technology 20.1 (2015): 17-27.

[7] N. S. A. (NSA), "Secure hash standard (SHS)," United States National Institute of Standards and Technology (NIST), vol. Federal Information Processing Standards Publication 180-4, March 2012, http://csrc.nist.gov/publications/fips/fips180-4/fips-180-4.pdf

[8] C. Wang, Z. Qin, J. Peng, and J. Wang, "A novel encryption scheme for data deduplication system," in Proc. IEEE Int. Conf. on Communications, Circuits and Systems (ICCCAS 2010), 2010, pp. 265–269

[9] Saritha, K., and S. Subasree. "Analysis of hybrid cloud approach for private cloud in the de-duplication mechanism." Engineering and Technology (ICETECH), 2015 IEEE International Conference on.IEEE, 2015.

[10] Backialakshmi, N., and M. Manikandan. "Data de duplication using N0SQL Databases in Cloud." Soft-Computing and Networks Security (ICSNS), 2015 International Conference on.IEEE, 2015.

[11] Kaushik, Vandna Dixit, et al. "Certain Reduction Rules Useful for De-Duplication Algorithm of Indian Demographic Data." Advanced Computing & Communication Technologies (ACCT), 2014 Fourth International Conference on. IEEE, 2014.