De-Duplication Techniques: A Study

S.Usharani, K.Dhanalakshmi, N.Dhanalakshmi

Abstract- De-duplication is the growing technology in storing the data over the cloud. De-duplication means avoiding the duplicate data(i.e) multiple copies of same data can be identified and rectified by de-duplication technique. De-duplication technique can apply to any type of storage data such as cloud storage, primary storage and secondary storage. In this Paper, we are going to see, how the de-duplication techniques can applied to data stored in the cloud storage andwhat are the difficulties will arise while doing in it. Because for security purpose, cloud stored the data in an encrypted or cipher data format. But de-duplication technique can't be applied for ciphered or encrypted data.So how the challenges can overcome and de-duplication technique can be performed over cloud storage data will learn in this paper.

I. INTRODUCTION

Data De-duplication is used to avoid multiple copies of similar data and it is primarily used to moderate spaces in the storage. That means only one unique copy of any data such text, image or video is engaged on storage media and duplicate or multiple copies of data should be reduced by de-dupe methodologies. De-duplication techniques provide the benefits such as increasing storage efficiency and network efficiency, improving speed replication and bandwidth efficiency, reducing backup window, and lastly cost is also reduced.De-duplication can be done by dividing in to three parts such as data comparison, storage place, and timeas shown in the table These three classification criteria can be defined as chunking, hashing and indexing [1].

- Easy and fast to compute the hash value for any block
- Small changes in a block data should completely change the hash value so broadly that the new hash value looks uncorrelated with the previous hash value
- Infeasible to find two dissimilar messages with the same hash value Because the hash value of the file should be unique as we know the properties of hash function.
-

II. CLASSIFICATION OF DE-DUPLICATION

A. Data Comparison based on Granularity

File level De-duplication – The data can be compared at the files or sub files which will be done by sub dividing the files to chunks or blocks. Then the chunk is produced in the cryptographic algorithm to produce a hash value or digital print for the block/chunk. As we know the cryptographic hash functions have some important properties they are:

- Deterministic so the same messages will constantly results in the equal hash valu

Revised Manuscript Received on December 22, 2018.

S.Usharani, Associate Professor, Department of CSE, IFET College of Engineering, Villupuram, ushasanchu@gmail.com

Dr. K.Dhanalakshmi, Professor, Department of CSE, PSNA College of Engineering and Technology, Dindugul, dhanalakshmikrs@gmail.com

Dr. N. Dhanalakshmi, Assistant Professor, Department of CSE, PSNA College of Engineering and Technology, Dindugul ndmugi@gmail.com

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

Table 1.1 De-duplication Classifications

Data Comparison based on sandy	Place	Time
File level– Data in the Files are sub-divided in to	Server based de-duplication	Inline de-duplication
Fixed-length block de-duplication	Client based de-duplication	Offline de-duplication
Variable-length block de-duplication	End to end Redundancy elimination	
	Network wide Redundancy elimination	
Application centre De-duplication	Target De-duplication	
	Global de-duplication	

De-duplication is the growing technology in storing the data over the cloud. De-duplication means avoiding the duplicate data (i.e) multiple copies of same data can be identified and rectified by de-duplication technique. De-duplication technique can apply to any type of storage data such as cloud storage, primary storage and secondary storage. Data De-duplication is used to avoid multiple copies of similar data and it is primarily used to moderate spaces in the storage. That means only one unique copy of any data such text, image or video is engaged on storage media and duplicate or multiple copies of data should be reduced by de-dupe methodologies. De-duplication techniques provide the benefits such as increasing storage efficiency and network efficiency, improving speed replication and bandwidth efficiency, reducing backup window, and lastly cost is also reduced.

Hi I am Mrs.X, I want to show how the hash function works in files level and Block level. For doing that let us save the data to file and perform hashing and save the result. Now take the same data perform hashing for the same but this time, instead of file we have to take the data as blocks and produce hash value save the result. Compare the File level and Block level hashing both results should not be unique it will differ. As we know, Small changes in a block data should completely change the hash value so broadly that the new hash value looks uncorrelated with the previous hash value. So verify that add or delete some content to the same file or block, then the hash value will different from the before block level hashing and file level hashing. We have proven that infeasible to find two dissimilar messages with the same hash value. Cryptographic hashing properties have been verified and proved by us in this paper.

In this algorithm it divides the files in to chunks and the chunks should be in fixed or same size like 4KB, 8KB or 16 KB etc. For e.g, if a file is taken for de-duplication, once the data in the file is sub-divided in to chunk as 4kb size then remaining chunks are all divided in to same chunk size as 4kb for entire file. Then the chunk is produced in the cryptographic algorithm to yield a hash value or digital print for the block/chunk. Because of the hash properties each and every block should maintain a unique id, if and only if there are no any duplication, if the block is repeated which means file is repeated, then the block data is same, so we can identify (file due to the first property) the duplication or reappearance of same file easily.

This de-duplication algorithm is used to divide the Data in to chunks. The divided block may in various sizes such as 4KB, 8KB or 16KB etc accordingly. And the size of the block should change dynamically during the whole process. Then the hash algorithm always calculates a digital print on a variable chunk size and realizes if there is any match. After a chunk of data is processed, it automatically takes another chunk of data and does the same and repeat the processes until it complete the entire data.

This de-duplication performed for the information or data is done at the innermost level (i.e) Bytes. The Applicationcentre de-duplication is differ from hash de-duplication, because hash de-duplication reduce the data redundancy at the chunk level where content-centre looks the data as objects, the redundancy is reduced by comparing the objects of the data or information. In this, the de-duplication realizes the actual objects such as files, database objects and breaks the data in to larger blocks as 8mb to 100mb in size. By knowledge of the data content, similar data are found in this technique.

For example: The following Fig1.1 contains 256 bytes. If the data is stored as a file as test.doc then the file size is 27136 bytes.

Fig1.1 Data (27136 bytes)

If the above test.doc file is given in the cryptographic hash algorithm it produces the digital print value such as

MD5:25EAD6E689A32FAB9A5437F70F6D0717
SHA1:D9B81B5DBC6B1F171DC0A0B5A3D4C3AEE97FA9C8

De-duplication is the growing technology in storing the data over the cloud. De-duplication means avoiding the duplicate data (i.e) multiple copies of same data can be identified and rectified by de-duplication technique. De-duplication technique can apply to any type of storage data such as cloud storage, primary storage and secondary storage. Data De-duplication is used to avoid multiple copies of similar data and it is primarily used to moderate spaces in the storage. That means only one unique copy of any data such text, image or video is engaged on storage media and duplicate or multiple copies of data should be reduced by de-dupe methodologies. De-duplication techniques provide the benefits such as increasing storage efficiency and network efficiency, improving speed replication and bandwidth efficiency, reducing backup window, and lastly cost is also reduced.Hi I am Mrs.X, I want to show how the hash function works in files level and Block level. For doing that let us save the data to file and perform hashing and save the result. Now take the same data perform hashing for the same but this time, instead of file we have to take the data as blocks and produce hash value save the result. Compare the File level and Block level hashing both results should not be unique it will differ. As we know, Small changes in a block data should completely change the hash value so broadly that the new hash value looks uncorrelated with the previous hash value. So verify that add or delete some content to the same file or block, then the hash value will different from the before block level hashing and file level hashing. We have proven that infeasible to find two dissimilar messages with the same hash value. Cryptographic hashing properties have been verified and proved by us in this paper.In this algorithm it divides the files in to chunks and the chunks should be in fixed or same size like 4KB, 8KB or 16 KB etc. For e.g, if a file is taken for de-duplication, once the data in the file is sub-divided in to chunk as 4kb size then remaining chunks are all divided in to same chunk size as 4kb for entire file. Then the chunk is produced in the cryptographic algorithm to yield a hash value or digital print for the block/chunk. Because of the hash properties each and every block should maintain a unique id, if and only if there are no any duplication, if the block is repeated which means file is repeated, then the block data is same, so we can identify (file due to the first property) the duplication or reappearance of same file easily.

Fig1.2 Modified File

And if the above file is modified as fig1.2 and the digital print is calculated shown as below:

MD5:45C2E2CCDDEFDC3A10F9B0E697359DB0
SHA1:8D41C238C20DB3D1FE06B7728394DD67D6E72016
SHA256:F6B65628D328E93B5DBA69AC2873AEF617F870CBFB05714AEFF34ADCC42C63B6

B.Fixed-sized De-duplication – In this algorithm it divides the files in to chunks and the chunks should be in fixed or same size like 4KB, 8KB or 16 KB etc. For e.g, if a file is taken for de-duplication, once the data in the file is sub-divided in to chunk as 4kb size then remaining chunks are all divided in to same chunk size as 4kb for entire file. Then the chunk is produced in the cryptographic algorithm to yield a hash value or digital print for the block/chunk. Because of the hash properties each and every block should maintain a unique id, if and only if there are no any duplication, if the block is repeated which means file is repeated, then the block data is same, so we can identify (file due to the first property) the duplication or reappearance of same file easily.

C.Variable sized De-duplication – This de-duplication algorithm is used to divide the Data in to chunks. The divided block may in various sizes such as 4KB, 8KB or 16KB etc accordingly. And the size of the block should change dynamically during the whole process. Then the hash algorithm always calculates a digital print on a variable chunk size and realizes if there is any match. After a chunk of data is processed, it automatically takes another chunk of data and does the same and repeat the processes until it complete the entire data.

D.Application centre De-duplication – This de-duplication performed for the information or data is done at the innermost level (i.e) Bytes. The Application centre de-duplication is differ from hash de-duplication, because hash de-duplication reduce the data redundancy at the chunk level where content-centre looks the data as objects, the redundancy is reduced by comparing the objects of the data or information. In this, the de-duplication realizes the actual objects such as files, database objects and breaks the data in to larger blocks as 8mb to 100mb in size. By knowledge of the data content, similar data are found in this technique.

For example: Word document is compared with Word document.

E.Hashing De-duplication – Hashing de-duplication uses hash algorithms to check the blocks' of data. Some hashing algorithms are MD4/ MD5, SHA-1/2/256/512, RIPEMD. For example: if the content of the Fig1.1 is used to find the Hash value or digital print then the below table 1.3 shown be produced. It is important to see that the hash value of MD5 for the file test.doc is

MD5: 45C2E2CCDDEFDC3A10F9B0E697359DB0

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

then the hash value should be different

Where instead of file, only the content of the data is hashed

MD5: 2264727882CC9B39FCF89283C69F1B1A

From the above, it is easily known, the file and Fixed Block or variable block de-duplication is different.

Table 1.2 Fixed Sized chunk hash value calculation

00000000	44 65 2D 64 75 70 6C 69 63 61 74 69 6F 6E 20 69	De-duplication i
00000010	73 20 74 68 65 20 67 72 6F 77 69 6E 67 20 74 65	s the growing te
00000020	63 68 6E 6F 6C 6F 67 79 20 69 6E 20 73 74 6F 72	chnology in stor
00000030	69 6E 67 20 74 68 65 20 64 61 74 61 20 6F 76 65	ing the data ove
00000040	72 20 74 68 65 20 63 6C 6F 75 64 2E 20 44 65 2D	r the cloud. De-
00000050	64 75 70 6C 69 63 61 74 69 6F 6E 20 6D 65 61 6E	duplication mean
00000060	73 20 61 76 6F 69 64 69 6E 67 20 74 68 65 20 64	s avoiding the d
00000070	75 70 6C 69 63 61 74 65 20 64 61 74 61 20 28 69	uplicate data (i
00000080	2E 65 29 20 6D 75 6C 74 69 70 6C 65 20 63 6F 70	.e) multiple cop
00000090	69 65 73 20 6F 66 20 73 61 6D 65 20 64 61 74 61	ies of same data
000000a0	20 63 61 6E 20 62 65 20 69 64 65 6E 74 69 66 69	can be identifi
000000b0	65 64 20 61 6E 64 20 72 65 63 74 69 66 69 65 64	ed and rectified
000000c0	20 62 79 20 64 65 2D 64 75 70 6C 69 63 61 74 69	by de-duplicati
000000d0	6F 6E 20 74 65 63 68 6E 69 71 75 65 2E 20 44 65	on technique. De
000000e0	2D 64 75 70 6C 69 63 61 74 69 6F 6E 20 74 65 63	-duplication tec
000000f0	68 6E 69 71 75 65 20 63 61 6E 20 61 70 70 6C 79	hnique can apply

Table 1.3 Calculated Hashes (2569 bytes)

Name	Length	Hash
md2	16	A846FFB0664F140480E0264EEBAD8096
md4	16	4C95BBCC5EECF927F4ECCD32DB96A7F5
md5	16	2264727882CC9B39FCF89283C69F1B1A
sha1	20	1CC685347C5AAD1FCD34B669FE80318B9C3DAD3E
sha224	28	37A74FBD8F5CC1A60E152E563817F645A3859D2F0ED3613AAED409C8
sha256	32	EDB24646961BF5A6A9B5D6E3E4D723B027E8225A6E298E5B925B695AACFFE3EB
sha384	48	34167B4BFEAAE1095C1F9505FBDAEE72D0514671DE122FC482D0A0914B65B006D797C116852CAC7FE942E33FB7BA3C78
sha512	64	CB601F2ADB4238D28B62770E6DFC12EE0E8385F21BD267289BCD4649B5BE17A42AA5EC5ECD018EF564F840BDB98B171EE75606D8A7A4C50FF87C23C6DBD151C1
ripemd128	16	8B897A49AC4909355CBF8291D52FCC6C
ripemd160	20	4783438D417580F5B8A0725FF655EBCE9B3B0A86
ripemd256	32	E0CF06AF6EBC8FF549C0111A018E59BFBA647CE69A71E650F4DB0B45DBF20497
ripemd320	40	C7E25B864108B2798577CB87014DB32CB0D4EF3F2DC5033CC6AE74488E27C82F65BA16C52F5CB667
whirlpool	64	A33B8868BC25FDB7916A86B210A03C789B6F16CA0D18BF813EBBCEE0FA39DEF0610A19780AC2B7527A5F66CA5AE5DB816507E56DC81A4A8E761E5F45976A86DD

F.De-duplication based on Place

Server Based De-duplication – It emerged to store huge data for back-up. Server Based de-duplication are very fast and performances are very high. In server, data are sent by the client for back-up, where data are de-duplicated. In this method, files are transferred to server through client. On server, the files are divided in to chunks and then it processes the de-duplication technique. Server-based de-duplication founds significantly efficient redundancy, but it suffers extremeredundant data stream of traffic flow because redundant data are brought to servers for de-duplication. The main disadvantage is CPU processing reckoning and memory storage overhead for blocking and indexing all the reserve data.

G.Client Based De-duplication – In this de-duplication, clients side itself redundant data are checked in the local or cloud server through a backup agent. It transmits the distinct data to the server. It used to remove network traffic. Even though, it experience CPU processing computation and memory storage overhead for accessing the reserve of cloud agent.

H.Target-based de-duplication – In this de-duplication, it requires that the targeted backup servers or else it need dedicatedHardware target de-dupe application to handles all of the de-duplication. If it happens, then no upstairs on the client or server being reversed up. This solution is apparent to surviving effort flows, so it creates negligible disruption. Even though, it requires more network resources because the unique data, with all its idleness, must go over the network.

I.End to end RE – It removes redundancies in network stream of traffic at dual end points such as data center to another data center.

J.Network wide RE – It removes redundancies at routers on network pretty moderately running it at a cloud side.

K.Global de-duplication – It removes redundancies of Data across a set-up over all transport protocols

L.De-duplication based on Time

Inline Processing – Inline de-duplication is also called online or real time de-duplication because it removes redundancy in real times as the data are stored. For example, it can be smeared to main capacities like email and databases like hold-ups. For main workloads, de-duplication runs on a straight write or read path whereas for secondary workloads, de-duplication runs when data are archived or reversed up on a hold-up server. Inline processing removes redundancies for both main and secondary workloads without acquiring additional storage space or requiring extra disk bandwidth.

A.Offline Processing – Offlinede-duplication removes redundancies after the data are warehoused on the disk. So it requires more storage space. It has several disadvantages such as

- Extra space to grasp data briefly from before de-duplication
- It should run the de-duplication on system shiftless time so it can be postponed if the system is processing nearly all the time
- Data on disk are overloaded to memory for de-duplication, so memory disk bandwidth is unreasonably inspire

III. WORKING OF DE-DUPLICATION

Usually if any user wants to send any data to another user, the data which the user needs to send is first uploaded in the public cloud server and then it is delivered to the actual receiver. If another user wants to send that same data to some other user, he/she again uploads the same data once again in the cloud server. Here the same data is being uploaded twice in the server, which results in data duplication and thereby the storage space of the server is being wasted. In-order to overcome this data duplication problem occurring at the server end, we propose a new data de-duplication technique in which we introduce the concept of Convergent Encryption [2] and proof of rights for the data with secure, more scalable and very efficient solution [3].

IV. CONFLUENT ENCRYPTION

Confluent encryption [2], is an encryption method that provisions de-duplication. With confluent encryption, the key is produced on sale of hashing of Original text. By smearing this technique same plaintexts will create the identical cipher text, and it helps in accomplishing de-duplication further.

V. PROOF OF RIGHTS

After finishing the de-duplication, if a replicate data are found then we have to produce pointer to the file which is already present. But if we produce a pointer to the file, if a file is deleted by the first loaded person, it will cause problem, because the same file is uploaded by another person also. To avoid such confusion we are going to introduce a proof of rights concept. If a person, load a file then the ownership of file should be given to the person. If the same file is pointed by another person for de-duplication then the ownership rights should be shared between two people, like wise if more numbly of people pointed the same file then the ownership rights should be shared by all of them. Ownership of rights is explained in the Table 1.5.

If a person, upload any file in the server then de-duplication will compute hash functionon the file and then the file should be shared in the server. If another person sent the same file, then the file ownership should be shared between two of them. But there are so many challenges are available to share a same file, such as deleting or modifying. If one person deletes the file, then ownership rights of the person should be deleted but not the file should be deleted. If a person modified the file, then hash value should be recalculated and new memory should be allocated to the modified file and ownership rights of the before modification file should be removed from the ownership table.

VI.CONCLUSION

De-duplication is a strategy accessible in distributed cloud storage for sparing data transmission and limitation of volume. Be that as it may, de-duplication is slight attainable with encrypted data, since various encryptions change over similar information into various configurations. In this paper, different de-duplication techniques are studied about where de-duplication strategies are done on encrypted data in an extensive storage area. The greater part of the strategies examined here work based on Confluent encryption, which is a simplemethod that creates de-duplication good with Encoded data. In this data thick world, we can't trade off on both safekeeping and duplication of data crosswise over capacity of storage spaces. A methodology should be planned which will improve capacity of storage advancement without consulting on encryption strategy; by giving de-duplication system in data or information storage servers where the accessible data is scrambled.

REFERENCES

1. D. Kim et al., Data Deduplication for Data Optimization for Storage and Network Systems, Springer International Publishing Switzerland 2017. https://www.springer.com/in/book/9783319422787
2. https://pibytes.wordpress.com/2013/02/02/deduplication-internals-part-1
3. Xiao-Long Liu, Ruey-Kai Sheu, Shyan-Ming Yuan, Yu-NingWang ," A file-deduplicated private cloud storage service with CDMI standard", 2015 Elsevier B.V. https://doi.org/10.1016/j.csi.2015.09.010
4. AkhilaK,AmalGanesh,Sunitha C ,"A Study on Deduplication Techniques over Encrypted Data", Fourth International Conference on Recent Trends in Computer Science & Engineering, Chennai, Tamil Nadu, India.

Retrieval Number:F13700476S519/19©BEIESP

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

https://doi.org/10.1016/j.procs.2016.05.123
5. LorenaGonzález-Manzano, AgustinOrfila, "An efficient confidentiality-preserving Proof of Ownership for deduplication", 2015 Elsevier Ltd. https://doi.org/10.1016/j.jnca.2014.12.004.
6. VivekWaghmare, SmitaKapse, " Authorized Deduplication: an approach of Secure cloud Environment". https://doi.org/10.1016/j.procs.2016.02.063
7. JinLi, XiaofengChen, FatosXhafa, LeonardBarolli,"Secure deduplication storage systems supporting keyword search", Elsevier Inc. https://doi.org/10.1016/j.jcss.2014.12.026
8. Meixia Miao, Jianfeng Wang, Hui Li , Xiaofeng Chen, "Secure multi-server-aided data deduplication in cloud computing" http://dx.doi.org/10.1016/j.pmcj.2015.03.002,1574-1192/© 2015 Elsevier
9. OpenDedup. OpenDedup, Global inline deduplication for Block Storage and Files. [online] 2010 Available from:http://opendedup.org/index.php.
10. R. Di Pietro, A. Sorniotti, Proof of ownership for deduplication systems: A secure, scalable, and efficient solution, Computer Communications (2016), http://dx.doi.org/10.1016/j.comcom.2016.01.011
11. D. Connor, P.H. Corrigan, J.E. Bagley, Cloud Storage: Adoption, Practice and Deployment, An Outlook Report from Storage Strategies NOW2011.http://docs.media.bitpipe.com/io_10x/io_101145/item_436696/Cloud%20Storage%20Adoption,%20Practice,%20and%20Deployment.pdf
12. J. Douceur, A. Adya, W. Bolosky, D. Simon, and M. Theimer. "Reclaiming space from duplicate files in a serverless distributed file system. In Distributed Computing Systems", 2002. Proceedings. 22nd International Conference on, pages 617{624. IEEE, 2002.
13. M. Rouse, DEFINITION-public cloud storage, TechTarget, 2011.
14. http://searchcloudstorage.techtarget.com/definition/public-cloud-storage
15. D. Chen, H. Zhao, Data security and privacy protection issues in cloud computing, Computer Science and Electronics Engineering (ICCSEE), 2012 International Conference on. IEEE2012 647–651.
16. **DOI:** 10.1109/ICCSEE.2012.193https://ieeexplore.ieee.org/document/6187862/
17. SNIA,"AdvancedDeduplication Concepts,"[online] 2011.Available fromhttp://www.snia.org/sites/default/education/tutorials/2011/fall/DataProtectionManagement/ThomasRiveria_Advanced_Dedupe_Concepts_FINAL.pdf
18. I. Ion, N. Sachdeva, P. Kumaraguru, S.C. Apkun, E. Zurich, I. Delhi, Home is safer than the cloud!: privacy concerns for consumer cloud storage, Proceedings of the Seventh Symposium on Usable Privacy and Security. ACM, 2011 13–23.doi>10.1145/2078827.2078845
19. https://dl.acm.org/citation.cfm?id=2078845
20. http://searchdatabackup.techtarget.com/tip/Where-and-how-to-use-data-deduplication-technology-in-disk-based-backup
21. C.Wang, S.S.M. Chow, Q.Wang, K. Ren,W.J. Luo, Privacy-preserving public auditing for secure cloud storage, IEEE Trans. Comput. 62 (2) (2013) 362–375.**DOI:** 10.1109/TC.2011.245.https://ieeexplore.ieee.org/document/6109245/
22. S.Usharani., D.Saravanan, R.Parthiban, An capable facts amalgamation come near through the guess of hazardous patients as of the original phase , International Journal of Pure and Applied Mathematics , Volume 119 No. 14 2018, 603-609
23. R.Parthiban,D.Saravanan,S.Usharani, Worldwide center discovery using surf and sift algorithm, , International Journal of Pure and Applied Mathematics , Volume 119 No. 14 2018, 705-708.
24. P.ManjuBala, J. Kayalvizhi, S. Usharani, D.Jayakumar, A decentralized file shareing& data transmission in peer to peer communication using edonkey protocol, International Journal of Pure and Applied Mathematics, Volume 119 No. 14 2018, 1027-1032
25. K.Jayasri, R.Rajmohan, D.Dinagaran, Analyzing the query performances of description logic based service matching using Hadoop, International Conference on Smart Technologies and Management for Computing, Communication, Controls, Energy and Materials, ICSTM 2015 – Proceedings.
26. D.Saravanan, R.Parthiban, S.Usharani, Precautions and seclusion shield in cloud computing, International Journal of Pure and Applied Mathematics, Volume 119 No. 14 2018, 849-856
27. G.Chandralekha, K.Iswarya, M.Pajany, P. Vimala, A survey on smart exam script evaluation using OCR and ontology, International Journal of Pure and Applied Mathematics, Volume 119 No. 14 2018.
28. M DeivaRagavi, S Usharani, Social data analysis for predicting next event, Information Communication and Embedded Systems (ICICES), 2014 International Conference, IEEE, 2014.**DOI:** 10.1109/ICICES.2014.7033935https://ieeexplore.ieee.org/abstract/document/7033935/
29. Dutch T Meyer and William J Bolosky."A study of practical Deduplication". *ACM Transactions on Storage (TOS)*, 7(4):14, 2012. https://www.usenix.org/legacy/event/fast11/tech/full_papers/Meyer.pdf
30. C.I. Ku, File deduplication with cloud storage file system, Computational Science andEngineering (CSE), IEEE 16th International Conference on. IEEE2013 280–287.**DOI:** 10.1109/CSE.2013.52https://ieeexplore.ieee.org/document/6755230/
31. Halevi, D. Harnik, B. Pinkas, and A. Shulman-Peleg, "Proofs of ownership in remote storage systems," in Proc. ACM Conf. Comput.Commun. Security, 2011, pp. 491–500.doi>10.1145/2046707.2046765
32. https://dl.acm.org/citation.cfm?id=2046765
33. A. Patawari, Getting started with own Cloud, Packt Publishing, 2013.
34. https://owncloud.com/getting-started-with-owncloud/
35. Yang, Chao, Jianfeng Ma, and JianRen. "Provable Ownership of Encrypted Files in De-Duplication Cloud Storage." *Ad Hoc & Sensor Wireless Networks*26.1-4 (2015): 43-72. https://doi.org/10.1002/sec.784https://dl.acm.org/citation.cfm?id=2913526
36. State of Information Global Results, Symantec Corporation, 2012.
37. https://www.symantec.com/content/dam/symantec/docs/reports/2012-state-of-information-global-en.pdf
38. Li, Jin, Yan Kit Li, Xiaofeng Chen, Patrick PC Lee, and Wenjing Lou. "A hybrid cloud approach for secure authorized deduplication." *Parallel and Distributed Systems, IEEE Transactions on* 26, no. 5 (2015): 1206-1216.DOI: 10.1109/TPDS.2014.2318320
39. https://sci-hub.tw/10.1109/TPDS.2014.2318320
40. R.K. Sheu, S.M. Yuan, W.T. Lo, C.I. Ku, Design and implementation of file deduplication framework on hdfs, International Journal of Distributed SensorNetworks, 2014http://dx.doi.org/10.1155/2014/561340
41. Bellare, Mihir, SriramKeelveedhi, and Thomas Ristenpart. "Message-locked encryption and secure deduplication." *Advances in Cryptology–EUROCRYPT 2013*. Springer Berlin Heidelberg, 2013. 296-312.
42. https://link.springer.com/chapter/10.1007/978-3-642-38348-9_18
43. D.J. Abadi, Data management in the cloud: limitations and opportunities, IEEE DataEng. Bull. 32 (1) (2009) 3–12.
44. https://pdfs.semanticscholar.org/a3d1/fb532bf4297ae9608bdc09ae5464ef58ee30.pdf
45. Chen, Rongmao, Yi Mu, Guomin Yang, and FuchunGuo. "BL-MLE: Block-Level Message-Locked Encryption for Secure Large File Deduplication." (2015). *Information Forensics and Security, IEEE Transactions on* 26(2015), no. 12: 2643-2652. DOI: 10.1109/TIFS.2015.2470221https://ieeexplore.ieee.org/document/7210226/
46. S. Dhumbumroong, K. Piromsopa, Personal Cloud File system: a distributed unification file system for personal computer and portable device, Computer Science andSoftware Engineering (JCSSE), Eighth International Joint Conference on. IEEE201158–62.
47. **DOI:** 10.1109/JCSSE.2011.5930094
48. https://ieeexplore.ieee.org/document/5930094/
49. Miguel, Rodel, and KhinMiMiAung. "HEDup: Secure Deduplication with Homomorphic Encryption." In *Networking, Architecture and Storage (NAS), 2015 IEEE International Conference on*, pp. 215-223. IEEE, 2015.

50. DOI: 10.1109/NAS.2015.7255226
51. https://ieeexplore.ieee.org/document/7255226/
52. G. Community, Cloud Storage for the Modern Data Center—An Introduction to Gluster Architecture, 2011.
53. http://moo.nac.uci.edu/~hjm/fs/An_Introduction_To_Gluster_ArchitectureV7_110708.pdf
54. Bellare, Mihir, SriramKeelveedhi, and Thomas Ristenpart. "Dupless: Server-aided encryption for deduplicated storage." *Proceedings of the 22nd USENIX conference on security*. USENIX Association, 2013.
55. https://dl.acm.org/citation.cfm?id=2534782

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

Table 1.5 Proofs of Rights of the Shared File

File Holders	Uploaded file	Index pointed to memory location	File Holder Rights	Separate Memory for Modifying the File (If ownership rights is only read then the person want to modify the file)	Deleting the File(If ownership rights is only read then the person want to delete the file)
Person X	Q	100	Read	File Copied to new memory and modify and store the file as new file	Ownership of File is removed, and file is not deleted if other person is having the ownership of the file.
Person Y	Q	100	Read	File Copied to new memory and modify and store the file as new file	Ownership of File is removed, and file is not deleted if other person is having the ownership of the file.
Person Z	R	200	Read, Write, Delete	-	-
Person T	Q	100	Read	File Copied to new memory and modify and store the file as new file	Ownership of File is removed, and file is not deleted if other person is having the ownership of the file.