A De-Duplication in Distributed System Using File & Block Level De-Duplication Techniques

Mr. T. P. ADHAU[1], Dr. V. M. Deshmukh[2]
[1]Dept of CSE
[2]Professor, Dept of CSE
[1, 2] PRMIT & R Badnera, Amravati, India

Abstract- *For removing replication copies of data we use data De-duplication process. As well as it is used in cloud storage to reduce memory space & upload bandwidth. Only one copy for each file stored in cloud that can be used by number of users. De-duplication process helps to improve storage reliability. One more challenge of privacy for sensitive data also arises when they are outsourced by users to cloud. The aim of this paper is to make the first attempt formalize the idea of distributed reliable De-duplication system. In our proposed system we are going to develop a new distributed De-duplication system which is highly reliable. In De-duplication process data chunks are distributed across multiple cloud servers. Instead of using convergent encryption as in previous De-duplication systems we use deterministic secret sharing scheme in distributed storage systems. So that we can achieve the required concepts for security that are data confidentiality and tag consistency. In the proposed security model, Security analysis demonstrates that our De-duplication systems are secure*

Keywords- De-duplication, secret sharing, distributed storage system, reliability

I. INTRODUCTION

In computing,data De-duplication is a specialized data compression technique for eliminating duplicate copies of repeating data. Related and somewhat synonymous terms are intelligent (data) compression and single-instance (data) storage. This technique is used to improve storage utilization and can also be applied to network data transfers to reduce the number of bytes that must be sent. In the de duplication process, unique chunks of data,or byte patterns, are identified and stored during a process of analysis.By the unpredictable development of digital data, De-duplication techniques are broadly engaged to backup data and decrease network and storage transparency by notice and eradicate redundancy among data. As an alternative of maintaining multiple data copies with the same content, De-duplication reducing redundant data by maintaining only single copy and referring other redundant data to that copy. De-duplication has inward much concentration from both academic world and industry since it can really recover storage utilization and keep storage space, particularly for the applications with high De-duplication ratio such as archival storage systems. A number of De-duplication systems have been projected based on various De-duplication scheme such as client-side or server-side De-duplication, file-level or block-level De-duplications.Specially, with the advent of cloud storage, data De-duplication procedure grow to be more gorgeous and essential for the management of ever-increasing quantity of data in cloud storage services which inspires Endeavour and club to outsource data storage to third-party cloud providers. Today's commercial cloud storage services, such as Dropbox, Google Drive and Mozy, have been applying De-duplication to save the network bandwidth and the storage cost with client-side De-duplication.

Data De-duplication often called intelligent compression or single-instance storage is a process that eliminates redundant copies of data and reduces storage overhead. Data De-duplication techniques ensure that only one unique instance of data is retained on storage media, such as disk, flash or tape. Redundant data blocks are replaced with a pointer to the unique data copy. In that way, data De-duplication closely aligns with incremental backup, which copies only the data that has changed since the previous backup

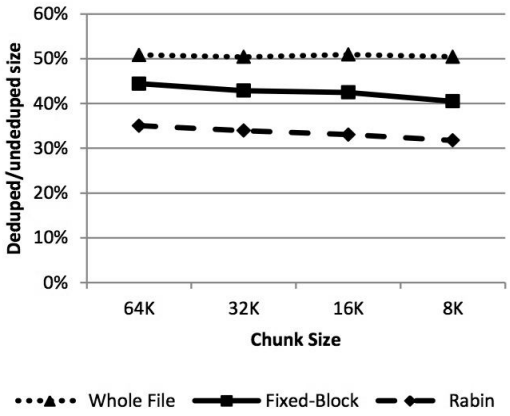
II. RELATED WORK

M. Bellare, S. Keelveedhi, and T. Ristenpart, "Dupless: Serveraided encryption for deduplicated storage," in USENIX Security Symposium, 2013. It introduced the idea of security and scheme for symmetric encryption in concentrate security framework. They give different idea of security and analyze the good involution of reduction among them. They provide method of encryption using a block cipher, cipher block chaining and counter mode. Its have two goals .First is to study the idea of security for symmetrical encryption and second is to provide concrete security analysis of fixed symmetric encryption device. M. W. Storer, K. Greenan, D. D. E. Long, and E. L. Miller, "Secure data de-duplication," in Proc. Of StorageSS, 2008. They developed a

solution that provides both data security and space efficiency in single-server storage and distributed storage systems to solve the problem such that de-duplication exploits identical content, while encryption tries to make all content appear random ,the same content encrypted with two different keys results in very different cipher text. De-duplication and encryption are opposed to one another. De-duplication takes benefit of data similarity to achieve a reduction in storage space & the goal of cryptography is to make cipher text indistinguishable from theoretically random dataP. Anderson and L. Zhang, "Fast and secure laptop backups with encrypted de- duplication," in Proc. of USENIX LISA, 2010.They presents an algorithm which takes benefits of the data which is common between users to reduce the storage requirements, and increase the speed of backups. This algorithm supports client end per-user encryption which is important for confidential personal data, also supports a unique feature that allows immediate detection of common sub trees, avoiding the necessity to query the backup system for every file. This system has shown that a community of laptop users shares a considerable amount of data in between. This gives the potential to significantly decrease backup times and storage requirements. However, they have shown that manual selection of the relevant data -eg, backing up only home directories is a poor strategy; this become fails to take backup of important files, at the same time as unnecessarily duplicating other files.Yinjin Fu, Hong Jiang 2014 proposes ALG-Dedupe, an application aware local-global source-DE-duplication scheme for cloud backup in the personal computing environment to improve DE-duplication efficiency. An intelligent DE-duplication strategy in ALG-Dedupe is designed to exploit file semantics to minimize computational overhead and maximize DE-duplication effectiveness using application awareness. It combines local DE-duplication and global DE-duplication to balance the effectiveness and latency of DE-duplication. The proposed application-aware index structure can significantly relieve the disk index lookup bottleneck by dividing a central index into many independent small indices to optimize lookup performance.Jin Li, Yan Kit Li, Xiaofeng Chen, Patrick P. C. Lee, Wenjing Lou, 2014] In this paper they first attempt to address the problem of authorized data DE-duplication. The system present new DE-duplication constructions to support authorized duplicate checking. This paper shows that authorized duplicate check method incurs minimal overhead as compared to conversion encryption.

III. PROPOSED WORK

To protect private data the secret sharing technique is used which is corresponding to distributed storage systems. In this paper the secret sharing technique is used for protection of private data. In detail a file is divides and encode into sections by using secret sharing technique. These sections will be distributed over many independent storage servers. A cryptanalysis hash value of the content will also be calculated and send to storage server as the mark of the fragment stored at each server. only the data user who first upload the data is required to calculate and distribute such secret shares and following users own same data copy do not need to calculate and stores these shares. Retrieve data copies owner must access a minimum number of storage server by a validation and obtain the secret shares to alter the data. In different way, the authorized uses will access the secret shares data copy. Another distinguishable feature of our proposal is that data completeness encloses tag consistency, can be derived. To explain further if the same value is stored in various cloud storage then de-duplication check by methods. It cannot oppose the collision attack established by many servers. To our knowledge no related work on secure de-duplication can rightly address, the reliability and tag consistency problem. The file level and block level de-duplication is used for higher reliability. The secret splitting technique is used for protect data. Our proposed structure supports both traditional de-duplication methods. Privacy, credibility and integrity can be achieved in our proposed system. In solution to kind of secret agreement attacks are considered. These are the attack on the data and the attack against servers. The data is secure when the opponent control limited number of storage servers.



Proposed Techniques

File Splitting

Data de-duplication involves finding and removing duplication within data without compromising its fidelity or integrity. The goal is to store more data in less space by segmenting files into small variable-sized chunks (32–128 KB), identifying duplicate chunks, and maintaining a single copy of each chunk. Redundant copies of the chunk are replaced by a reference to the single copy. The chunks are

compressed and then organized into special container files in the System Volume Information folder. After de-duplication, files are no longer stored as independent streams of data, and they are replaced with stubs that point to data blocks that are stored within a common chunk store. Because these files share blocks, those blocks are only stored once, which reduces the disk space needed to store all files. During file access, the correct blocks are transparently assembled to serve the data without calling the application or the user having any knowledge of the on-disk transformation to the file. This enables administrators to apply de-duplication to files without having to worry about any change in behavior to the applications or impact to users who are accessing those files.

File-Level Distributed De-duplication System

It support capable duplicate check, tags for each file will be calculated and send to storage cloud service provider. To prevent alignment invasion organized by the cloud based service provider, tag collected at different storage servers. System Setup: In our structure, the storage cloud service provider is considered to be n with identities denoted by id1, id2,…,idn respectively. To upload file F, the client communicate with cloud based service provider to perform the elimination of duplicate data .For downloading file F, the client downloads the secret shares of the file from k out of storage servers.

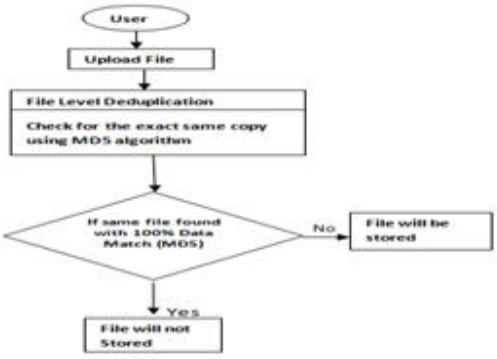


Figure: Working process of File level De-duplication

Block-Level De-duplication System

In this part, we appear how to derive the fine grained block level distributed de-duplication. In this system, the client also demands to perform the file level de-duplication before uploading file. The user partition this files into blocks, if no duplication is found and performs block-level de-duplication system. The system set up is similar to file-level de-duplication and also block size parameter will be defined.

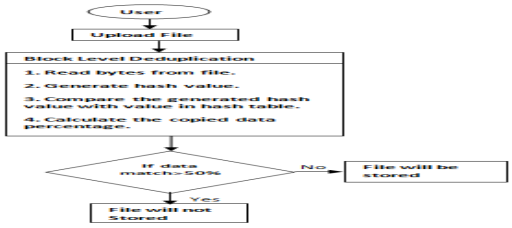


Figure: Working process of Block level De-Duplication

System Architecture

Two kind's entities will be involved in this De-duplication system, including the user and the storage cloud service provider (S-CSP).

User: The user is an entity that wants to outsource data storage to the S-CSP and access the data later. In a storage system supporting De-duplication, the user only uploads unique data but does not upload any duplicate data to save the upload bandwidth. Furthermore, the fault tolerance is required by users in the system to provide higher reliability.

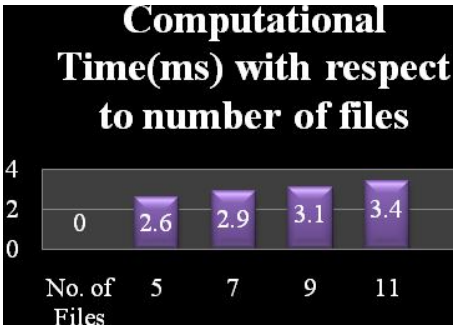
S-CSP: The S-CSP is an entity that provides the outsourcing data storage service for the users. In the De-duplication system, when users own and store the same content, the S-CSP will only store a single copy of these files and retain only unique data. A De-duplication technique, on the other hand, can reduce the storage cost at the server side and save the upload bandwidth at the user side. For fault tolerance and confidentiality of data storage, we consider a quorum of S-CSPs, each being an independent entity.

IV. RESULTS

File level De-duplication graph

.

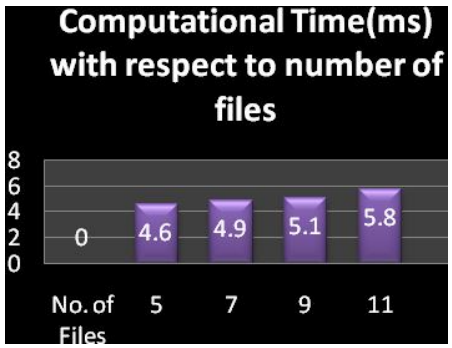
No. of Files	Computational Time for file level approach
5	2.6
7	2.9
9	3.1
11	3.4



In the above graph the x axis indicates the number of files on which the file level de-duplication performed. The y axis indicated the total computational time in m/s to perform file level de-duplication by using MD5 algorithm.

Block level De-duplication graph

No. of Files	Computational Time for Block level approach
5	4.6
7	4.9
9	5.1
11	5.8



In the above graph the x axis indicates the number of files on which the block level De-duplication performed. The y axis indicated the total computational time in m/s to perform block level De-duplication.

V. CONCLUSION

Design of an improved technique for storage in Cloud is De-duplication technique. De-duplication aids in saving the storage space. This application helps in easy maintenance of data on the cloud platform so that no duplicate files are saved in the Cloud. With the evolution of Cloud computing, storage resources of commodity machines can be efficiently utilized. This allows every organization to build its own private cloud for a variety of purposes. In order to better utilize the limited storage available in a private cloud, a suitable approach for optimization has to be used.

REFERENCES

[1] M. Bellare, S. Keelveedhi, and T. Ristenpart, "Dupless: Serveraided encryption for deduplicated storage," in USENIX Security Symposium, 2013

[2] M. W. Storer, K. Greenan, D. D. E. Long, and E. L. Miller, "Secure data de-duplication," in Proc. Of StorageSS, 2008.

[3] P. Anderson and L. Zhang, "Fast and secure laptop backups with encrypted de- duplication," in Proc. of USENIX LISA, 2010

[4] J. S. Plank, S. Simmerman, and C. D. Schuman, "Jerasure: A library in C/C++ facilitating erasure coding for storage applications - Version 1.2," University of Tennessee, Tech. Rep. CS-08-627, August 2008.

[5] J. S. Plank and L. Xu, "Optimizing Cauchy Reed-solomon Codes for fault-tolerant network storage applications," in NCA-06: 5th IEEE International Symposium on Network Computing Applications, Cambridge, MA, July 2006.

[6] C. Liu, Y. Gu, L. Sun, B. Yan, and D. Wang, "R-admad: High reliability provision for large-scale de-duplication archival storage systems," in Proceedings of the 23rd international conference on Supercomputing, pp. 370–379.

[7] M. Li, C. Qin, P. P. C. Lee, and J. Li, "Convergent dispersal: Toward storage-efficient security in a cloud-of-clouds," in The 6th USENIX Workshop on Hot Topics in Storage and File Systems, 2014.

[8] P. Anderson and L. Zhang, "Fast and secure laptop backups with encrypted de-duplication," in Proc. of USENIX LISA, 2010.

[9] Z. Wilcox-O'Hearn and B. Warner, "Tahoe: the least-authority filesystem," in Proc. of ACM StorageSS, 2008.

[10] A. Rahumed, H. C. H. Chen, Y. Tang, P. P. C. Lee, and J. C. S. Lui, "A secure cloud backup system with assured deletion and version control," in 3rd International Workshop on Security in Cloud Computing, 2011.

[11] M. W. Storer, K. Greenan, D. D. E. Long, and E. L. Miller, "Secure data de-duplication," in Proc. of StorageSS, 2008.

[12] J. Stanek, A. Sorniotti, E. Androulaki, and L.Kencl, "A secure data de-duplication scheme for cloud storage," 2014.

[13] BELLARE, M., AND NAMPREMPRE, C. Authenticated encryption: Relations among notions and analysis of the generic composition paradigm. In ASIACRYPT 2000

(Kyoto, Japan, Dec. 3–7, 2000), T. Okamoto, Ed., vol. 1976 of LNCS, Springer, Berlin, Germany, pp. 531–545.

[14] BOWERS, K. D., JUELS, A., AND OPREA, A. HAIL: a highavailability and integrity layer for cloud storage. In ACM CCS 09 (Chicago, Illinois, USA, Nov. 9–13, 2009), E. Al-Shaer, S. Jha, and A. D. Keromytis, Eds., ACM Press, pp. 187–198.

[15] CHAUM, D. Blind signatures for untraceable payments. In CRYPTO'82 (Santa Barbara, CA, USA, 1983), D. Chaum, R. L. Rivest, and A. T. Sherman, Eds., Plenum Press, New York, USA, pp. 199–203.

[16] Z. N. J. Peterson, R. Burns, J. Herring, A. Stubblefield, and A. D. Rubin," Secure Deletion for a Versioning File System", 2005.

[17] G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z. Peterson, and D. Song, "Provable data possession at untrusted stores," in Proceedings of the 14th ACM conferenceon Computer and communications security, ser. CCS '07. New York, NY, USA: ACM, 2007, pp. 598–609

[18] ERWAY, C. C., K¨UPC¸ ¨U, A., PAPAMANTHOU, C., AND TAMASSIA, "R. Dynamic provable data possession", In ACM CCS 09 (Chicago, Illinois, USA, Nov. 9–13, 2009), E. Al-Shaer, S. Jha, and A. D. Keromytis, Eds., ACM Press, pp. 213–222.

[19] C.KavithaSree, CH.Shashikala, Dr.S.PremKumar,"Secure Distributed DE-duplication Systems with Improved Reliability", Volume 2, Issue 12, December-2015.