A Survey on Distributed De-duplication system with improved Dependability

Yogesh Bhagwat, Sunil Minde, Kunal Malve, Harshad Pawar

Abstract— **The elimination of duplicate copies of the file which is present on the distributed server is called as data de-duplication and it is mostly used on the distributed systems for increasing the storage space of the systems and bandwidth. However, there is only one type of copy is stored on server even such a file is used by number of users. As the data de-duplication increases the storage space with improved reliability. Security is also important issue on distributed server which provide privacy for private data. In this paper, we introduce a system that provides with higher reliability data chunks are distributed among the multiple servers using file level distributed system, block level distributed system and secret sharing scheme. Data confidentiality and tag consistency are achieved instead using previous encryption methods. This paper proposes reducing the duplication of the data which is present in the file on distributed servers and improves the storage of the systems with higher data reliability.**

Index Terms— ***Distributed storage system, data de-duplication, secret sharing technique.***

I. Introduction

Data deduplication is technique which is used on the cloud for compression data and also used for removing duplicate copies which are present in cloud and also used in the storage cloud service provider to reduce the amount of storage space and save upload bandwidth. The incredible growth of digital data, this technique used for back up the data and reduce the network bandwidth and storage overhead by detecting and eliminating duplicate copies of the data which is unnecessarily increases the storage space in the cloud. Instead of keeping multiple files with the same content, deduplication deleting same content of file by keeping only one physical copy. Data Deduplication has much aware from both industry and academia because it can largely increases storage fulfilment and save storage space, specially used for the application which have a high De-duplication ratio.

The main purpose of the multiple deduplication systems have been proposed by the various deduplication strategies such as system, this technique is more useful and difficult for the management of reducing the size of data in cloud storage service provider. Data De-duplication provides and motivates industrial and organizational outsource data storage in the cloud [1]. As per the assumption of international data corporation, the volume of data will be reach up-to 40 trillion gigabytes in 2020 [2].Now a days the trade is that, cloud storage services such as google drive, drop-box have been implemented this de-duplication to save the bandwidth of the network and increase the space in the cloud. To make data management scalable in distributed storage server, data deduplication has been a well-known technique and has attracted more and more attention recently in last few decades. The technique is used to improve distributed storage space utilization and can also be applied to data transfers to minimize the number of bytes that must be sent over the network.

Distributed server is widely used service model that provides scalable and storage space on the network. One of the most important functionality is that Storage cloud server provider(S-CSP) can offer cloud storage. The simple principle of deduplication is that repeated data uploaded by huge number of user's are stored only once. Unfortunately, data deduplication is not compatible with encryption because of storage overhead. If different users upload the same file, instead of storing multiple copies of it, the distributed storage provider adds the user unique copy of the file. Costs of storing and transferring data can be greatly smaller. As an example, data deduplication can reduce up to 80% of storage based on the experiments in [5]. The aim of data deduplication is to identify identical data segments and store them only once.

II. RELATED WORK

A. Bellare et al. [3] formalized that message-locked encryption scheme, and explored its application in which include space-efficient secure outsourced storage.

B. Harnik. [14] Represented a number of attacks in cloud storage that can lead to data leakage system supporting client-side deduplication. Solution for these attacks.

C. Halevi et al. [6] designed the system for data deduplication systems is proofs of ownership, so that a client can efficiently

Manuscript received Sept, 2015.

Yogesh Bhagwat *, Computer Engineering Department,, Savitribai phule pune univercity/LGNSCOE). Nashik, Indian, Mobile N;9766812333*
Sunil Minde*, Computer Engineering Department,, Savitribai phule pune univercity/LGNSCOE). Nashik, Indian, Mobile N;9822873960*
Kunal Malve, Computer Engineering Department,, Savitribai phule pune univercity/LGNSCOE). Nashik, Indian, Mobile N;9175999693
Harshad Pawar, Computer Engineering Department,, Savitribai phule pune univercity/LGNSCOE). Nashik, Indian, Mobile N;8657020506

prove to the distributed storage server that user owns a file without uploading the file itself in the distributed storage server.

D. Ateniese et al. [13] introduced the concept of proof of data possession (PDP). This notion was introduced to allow a distributed server client to verify the integrity of its data to the server in a very economical way.

E. Recently, Xu et al. [24] presented a PoW strategy that allows client-side and server-side deduplication in a bounded effusion setting with security in the random oracle model.

F. Ng et al. [25] extended for encrypted file, but they did not mention how to reduce the key management overhead.

Data deduplication techniques are very interesting that are widely used for backup the important data in enterprise environments to minimize network bandwidth and storage capacity by detecting and reducing the redundancy among data which is present on the cloud. There are many reliable deduplication schemes proposed. The reliability in deduplication has also been addressed by [6], [5],[7].Data privacy is ensured by convergent encryption [4] in data deduplication system. There are many types of convergent implementations of different convergent encryption for data deduplication system. [8],[9],[10],[11] Current data deduplication systems that uses single instance storage depends on the three primary goals: w file, fixed-sized chunks, and variable-sized chunks. The first one is the, whole file, typically uses a file's hash value for identifying its identifier. Thus, if two or more files which have same hash value, they are assumed to have same identical contents and only stored at once (not including redundant copies).

III. PROPOSED SYSTEM

This section is devoted to the definitions of the how system model and security threats are work. In deduplication system two types of entities are their one is user and another is s cloud storage service provider(S-CSP).In this system model, to save the bandwidth for data uploading and storage space for data storing in the cloud both client and server side deduplication are supported. In order to save bandwidth of the uploading data and reliable management, the data will be moved to the s cloud server (S-CSP).This technique will be used for the storing only one copy of the same file in the cloud. The user is an entity that wants to store data on the outside outsource data storage and access the data later when user wants. In a cloud storage system deduplication, the user only uploads unique data or but does not upload any same copy of the file to save the upload bandwidth.

Furthermore, the main thing is required by users to provide higher reliability in the system, As part of constructing our security model, it is important to establish a consistent notation. For achieving confidentiality and integrity to storing data in the cloud, the data deduplication system has been proposed. The main objective of this system is avoid duplicate storage of the data across distributed storage servers. To keep the confidentiality of the data and integrity of the data, our new constructions utilize the data splitting technique to divide the data into chunks. These chunks will then be distributed across multiple storage servers. In this paper we try to minimize the storage of the system.

3. Building Blocks

A. S-CSP. The S-CSP is the storage cloud server provider service that provides the outsourcing data storage for the users. In the data deduplication system, when users wants to store the same data , the S-CSP will only store a single copy of these files and store only exclusive data.

3.1 The File-level Distributed Deduplication System.

To support better duplicate check, tags for each chunk of the file which will be allocated and computed are sent to S-CSPs. To avoid collusion attack the S-CSPs, the tags stored at different distributed storage servers are logically independent and different. We now describe the details of the construction as follows.

A. File Upload. To upload a file F on the storage server, the user interacts with S-CSPs to perform the data deduplication. The user first calculates and sends the file tag ϕF = TagGen (F) to Storage-Cloud Server Provider for the file duplicate check. If a duplicate is found, the user processes and sends the result ϕF; idj = TagGen′ (F, idj) to the j-th server with identity idj through the secure channel for 1 ≤ j ≤ n. Therefor logic behind is that an index j is to avoid the server from gaining the shares of other S-CSPs for the same data in a file or block, which will be expressed in detail in the security analysis. If ϕF;idj same as the metadata stored with ϕF , the user will supplied a pointer for the chunk stored at server idj .Otherwise, if no duplication is found, the user will perform the computation as follows. He runs the secret sharing algorithm SS on F to get the {cj} = Share (F), where cj is the j-th chunk of F. He also processes ϕF;idj = TagGen′(F, idj), which provide the tag for the jth S-CSP. Finally, the user get uploads the set of values {ϕF , cj , ϕF;idj} to the S-CSP with identity idj through a secure channel. The S-CSP stores these data values and pointer get back to the user for its regular storage.

B. File Download. To download a file F, the user first get the secret shares {cj} of the data or file from k out of n distributed storage servers. Respectively, the user sends the pointer of F to k out of n Storage -Cloud Service Providers. After getting enough shares, the user rebuild file F by using this algorithm strategy of Recover ({cj}).

This technique provides fault tolerance and grant the user to remain available even if any limited part of storage servers fail.

3.2 The Block-level Distributed Deduplication System

In this section, we express that how to achieve the fine-grained block-level distributed deduplication. In a block-level deduplication system, the user also needs to firstly perform the file-level deduplication before uploading his file. If no repeated data is found, the user divides this file into blocks and performs block-level deduplication. The system setup and the file-level deduplication system both are same, except the block size parameter will be added additionally. Next, File Upload and File Download, this are the two method used in this algorithms. To upload a file F on distributed storage server, the user first performs the file-level deduplication by sending request? F to the storage servers.

Whenever, duplication is occur in a file, then user will perform file-level deduplication on that file F.
Otherwise, user directly perform block-level deduplication on that file F as follows- Firstly File F is divide in into chunks {Ci} where i = 1, ... n). for each chunk Ci, computing? Ci = TagGen(Ci) for performing block level duplication, When the content of block level and file level are same then file is overlapped with the block Ci. Upon receiving block tags {? Ci}, the server with identity idj computes a block signal vector sCi for each i. • i) If sCi=1, the user further computes and sends? Ci;j = TagGen'(Ci, j) to the individuality of S-CSP with idj . If it also same as the corresponding tag stored, S-CSP returns a block pointer of Ci to the user. Then, the user keeps the block pointer of Ci and does not need to upload Ci. • ii) If sCi=0, the user runs the secret sharing algorithm SS over Ci and gets {bij} = Share (Ci), where bij is the j-th secret share of Ci. The user also computes ?Ci;j for 1 = j = n and uploads the set of values {?F , ?F;idj , bij , ?Ci;j} to the server idj via a secure channel. The corresponding pointers back to the user through S-CSP. File Download. To download a file F = {Ci}, the user first downloads the secret shares {bij} of all the blocks Ci in F from k out of n S-CSPs. Specifically, the user sends all the pointers for Ci to k out of n servers. After collecting all the shares, the user reconstructs all the fragments Ci using the algorithm of Recover ({·}) and gets the file F = {Ci}.In this paper, the data which is present in the file is uploaded by the user on the distributed server. After that server compare the chunks of the file by distributing them on to the servers. If any chunk of the file is matches with uploaded chunk of the file then, it will be directly discarded that particular chunk of the file. Using this technique, it will reduces the size of the servers storage and achieve the good reliability.

IV. SYSTEM WORKS:-

This section is introduce to the definitions of the how file upload and file download are works together to achieve a better de-duplication in distributed storage system.

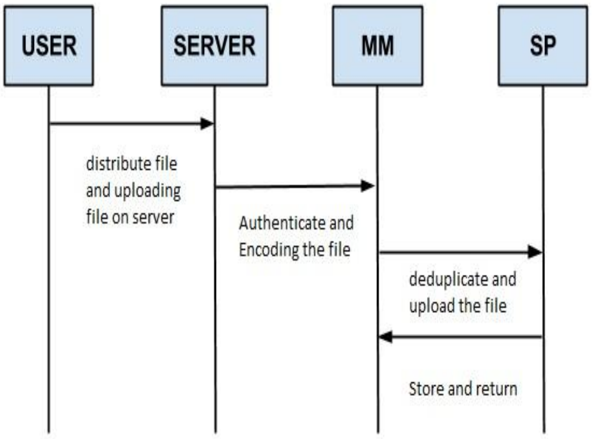
USER
SERVER
MM
SP
distribute file and uploading file on server
Authenticate and Encoding the file
deduplicate and upload the file
Store and return

Fig 1: Structure of the System of uploading File

A. User. User should upload the file on to the distributed storage server.

B. Encoding. In encoding and decoding phase, file is encoded by the system software and all the operations when performed file will stored in distributed storage server.

C. Server. In distributed storage server, Server will authenticate and distribute the file among multiple servers. Storage cloud service provider (S-CSP) is used for the managing the operations which are performed on the file.

D. MM. MM receives the request from the server and for each chunk of the file contained in the request, MM checks if that chunk has already been stored by computing its value and comparing it to the ones already stored. If the chunk of that file has not been stored in the past, MM creates a new node. MM updates the data structure by linking each chunk (block) of file F1 to its successor.

E. Sp. SP get the request to store a chunk of the file and returns pointer to the block.

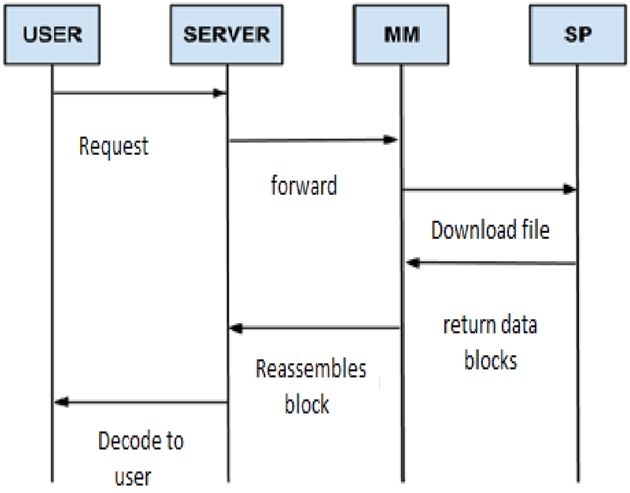

Fig 2: Structure of the System of downloading File

A. USER. User sends a retrieval request to the server in order to retrieve file F1.

B. SERVER. The server get the request from the user and then server forwards the request to MM without performing any encoding.

C. MM. MM receives the request from the server. If the user is authorized, MM search in the file identifier in the file table in order to get the first chunk of the file. Then, MM visits and retrieve all the chunks that compose the file. For each of these chunks, MM retrieves the pointer from the pointer table and sends a request to SP.

D. SP. SP returns the content of the encoded block to the MM.

V. SIMULATION RESULTS

In this simulation section study that examines efficiency of our data de-duplication method .The objective of our practical is to introducing the deduplication and dependability of our approach for estimating the duplicate files which increases the storage space.

VI. SCOPE OF THE SYSTEM

Scope of the proposed system is given below:

- To provide good security and dependability data de-duplication is required.
- It is very useful in large enterprises and industrial environment.
- Data Duplication among the chunks will be taken care while processing the multiple files.

CONCLUSION

In this work we try to achieve that distributed data deduplication systems to improve the reliability of data while accomplishing the confidentiality of the user's data without using encryption. Four constructions strategies were proposed to support file-level and block-level data deduplication. The security of tag consistency and integrity were accomplished. We try to implement our deduplication systems using checksum technique with small encoding and decoding overhead compared to the network transmission in regular upload and download operations on the storage server.

ACKNOWLEDGMENT

Our special thanks to Prof. M.C. Nikam for continuous inspiration and valuable guidance in throughout our dissertation work.

References

1) Amazon,"CaseStudies,"https://aws.amazon.com/solutions/casestudies/# backup.

2) J. Gantz and D. Reinsel, "The digital universe in 2020: Big data, bigger digi tal shadows, and biggest growth in the far east," http://www.emc.com/collateral/analyst reports/idcthe- digital-universe-in-2020.pdf, Dec 2012.

3) M. Bellare, S. Keelveedhi, and T. Ristenpart, "Dupless: Serveraided encryption for deduplicated storage," in *SENIX Security Symposium*, 2013.

4) J. R. Douceur, A. Adya, W. J. Bolosky, D. Simon, and M. Theimer, "Reclaiming space from duplicate files in a server less distributed file system." in *ICDCS*, 2002, pp. 617–624.

5) J. Li, X. Chen, M. Li, J. Li, P. Lee, and W. Lou, Secure deduplication with efficient and reliable convergent key management," in *IEEE Transactions on Parallel and Distributed Systems*, 2014, pp. vol. 25(6), pp. 1615–1625.

6) S. Halevi, D. Harnik, B. Pinkas, and A. Shulman-Peleg, "Proofs of ownership in remote storage systems." in *ACM Conference on Computer and Communications Security*, Y. Chen, G. Danezis, and V. Shmatikov, Eds. ACM, 2011, pp. 491–500.

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

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

9) P. Anderson and L. Zhang, "Fast and secure laptop backups with encrypted de-duplication," in *Proc. of USENIX LISA*, 2010.

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

11) M. W. Storer, K. Greenan, D. D. E. Long, and E. L. Miller, "Secure data deduplication," in *Proc. of StorageSS*, 2008.

12) 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 conference on Computer and communications security, ser. CCS '07. New York, NY, USA: ACM, 2007, pp. 598–609. [Online]. Available: http://doi.acm.org/10.1145/1315245.1315318.

13) Harnik, B. Pinkas, and A. Shulman-Peleg, "Side channels in cloud services: Deduplication in cloud storage." *IEEE Security & Privacy*, vol. 8, no. 6, pp. 40–47, 2010.

14) J. Xu, E.-C. Chang, and J. Zhou, "Weak leakage-resilient client-side deduplication of encrypted data in cloud storage," in ASIACCS, 2013, pp. 195–206.

1 st Author Name :- Yogesh Bhagwat.
Qualification :- Diploma in Computer Engg from Pune University,
B.E Appear of computer Engineering department
from Late G.N College Of Engineering Savitribai Phule Pune University.

2nd Author Name :- Sunil Minde.
Qualification :- Diploma in Computer Engg from Pune University,
B.E Appear of computer Engineering department
from Late G.N College Of Engineering Savitribai Phule Pune University.

3rdAuthor Name :- Kunal Makve.
Qualification :- Diploma in Computer Engg from Mumbai University,
B.E Appear of computer Engineering department
from Late G.N College Of Engineering Savitribai Phule Pune University.

4rdAuthor Name :- Harshad Pawar.
Qualification :- Diploma in Computer Engg from Pune University,
B.E Appear of computer Engineering department
from Late G.N College Of Engineering Savitribai Phule Pune University.