Improved deduplication with keys and chunks in HDFS storage providers

Ramya Ramesh[1], K.O Sumitha[2], Dr. S Subburam[3], R.K Kapila Vani [4]

[1, 2]Student, Department of Computer Science and Engineering, Prince Shri Venkateshwara Padmavathy Engineering College, Chennai, India
[3]Professor, Department, of Computer Science and Engineering, Prince Shri Venkateshwara Padmavathy Engineering College, Chennai, India
[4]Assistant Professor, Department of Computer Science and Engineering, Prince Dr. K. Vasudevan College of Engineering and Technology, Chennai, India

---***---

Abstract - *Cloud computing is one type of computing platform that is Internet based. One important feature of a Cloud service is Cloud storage. The cloud users may upload sensitive data in the Cloud and allow the Cloud server to maintain these data. It is critical to manage large volumes of duplicated data hence many deduplication techniques were developed. Existing solutions of deduplication failed in providing security and reliability efficiently. We propose a new distributed deduplication system in HDFS storage providers that ensures higher reliability with the help of ownership verification and provides reliable distributed key management servers to maintain the keys storage. Thus the new scheme achieves deduplication in HDFS storage easily.*

Key Words: deduplication, data chunks, key management, distributed block servers, tags.

1. INTRODUCTION

Cloud storage is a model of storage of data with a highly virtualized infrastructure made up of many distributed resources. In the cloud storage, the data is remotely maintained and managed. The users can store their data online and are able to access from any part of the world via the Internet connection. The biggest advantage of storing data in Cloud is that the users are provided with a broader range of access to distributed resources in a cost-efficient manner. This is because it eliminates the capital expenses such as buying the hardware, software and setting it up. Hence the cloud storage provides benefits of higher reliability, good performance, better usability, greater accessibility and secured environment. The rise of Cloud Computing led to the emergence of Big Data. Big Data refers to the term of dealing with extremely large amount of data that maybe structured or semi-structured or unstructured. Now-a-days, massive amount of informational data are generated due to digitalization and technological growth, hence it becomes essential to handle such Big Data in an efficient and economical way using Cloud. To process the Big Data, a Hadoop Java-based framework is used that helps to support large data processing in a parallel and distributed environment. Hadoop runs in a master-slave setup where the master distributes the jobs to its cluster and processes the map and reduce tasks sequentially.

The most important requirement of Cloud storage is to manage the large sets of data efficiently. Though the cloud storage is efficient, in order to make it more effective there is a need for eliminating the duplicate copies of data that are stored in Cloud. The duplicate copies may arise in scenarios where the data are shared among multiple users. For example, consider an email server system, where 200 instances of data are sent to all the members of an organization. If all the members back-up their inboxes each of their instances are saved creating duplicate copies of the same file which increases the storage space in Cloud. This greatly wastes the network bandwidth, complicates management of data and energy consumption. This introduces the concept of deduplication in Cloud storage. Deduplication is a compression technique where it eliminates the duplicate copies of data by storing only one physical copy of the data and referring other duplicate data to that copy. The technique achieves minimized network and storage overhead by improvising the storage capacity. Deduplication schemes have proved to reduce 90-95 percent storage for backup and up to 68 percent in file systems. Thereby, the overall storage needs has been reduced by 80 percent for both files and backups. Existing solutions on deduplication suffered from brute-force attacks [1], [2] and also it cannot ensure security, reliability, data privacy, data access control and data revocation.
In this paper, we propose a new scheme for deduplication with higher reliability in which the data chunks are distributed across HDFS and a reliable key management in secure deduplication using slave nodes. The rest of the paper has the details as follows. Section 2 gives an overview of related work. Section 3 discusses about the current methodology. Section 4 introduces the system model. Section 5 gives detailed description of our proposed scheme and finally conclusion is provided in the Section 6.

2. RELATED WORK

Many traditional encryption techniques for deduplication were introduced but they are not suitable since the file suffers from dictionary attacks [3].The data privacy is an important constraint in dealing with Cloud storage. Hence to ensure the data security and privacy, a good way is to outsource encrypted data. DeDu [4] is a system that was introduced to solve duplication efficiently but it was not able to handle encrypted data. In order to resist brute-force attacks, a secure deduplicated storage called DupLESS [5] was proposed. But the drawback of the system is that it cannot control over the data access of other data users in a convenient way.

To improve restore performance, History Aware Rewriting (HAR) algorithm [6] was proposed to accurate identifies and rewrite fragmented chunks. To protect the data security, an authorized data deduplication [7] was proposed by including differential privileges of users in the duplicate check in hybrid cloud architecture. But, it cannot flexibly support the data access control by data holders especially for data revocation process. ClouDedup [8], a secure and efficient storage service which assures block-level deduplication and data confidentiality was proposed. This scheme failed since it cannot solve the issue caused by data deletion where the data holder can access the data as it still knows the data encryption key if the data is not completely removed from the Cloud.

A new proposal was developed based on Provable Ownership of File (POF) [9]. This is a cryptographically secure and efficient scheme for a client to prove to the server based on actual possession of the entire file instead of only partial information about it. Thus, it helps in reducing the burden of the client. To have a secure and constant cost public Cloud storage [10], a new scheme introduced supported data integrity as well as storage deduplication at the same time. But, the drawback being that it doesn't discuss about the feasibility of supporting deduplication with Big Data. For reducing the workloads due to duplicate files, index name server (INS) [11] was proposed but the work doesn't concentrate on deduplication of encrypted data.

In this paper, we perform both file-level and block-level deduplication and use MD5 algorithm to generate signatures to perform the ownership verification. We also apply Triple Data Encryption algorithm (3DES) using convergent key for the data to be more secure and reliable against hackers. Thus, we achieve deduplication with higher security and reliability in an efficient manner in Cloud.

3. EXISTING METHODOLOGY

Most of the previous methodologies have only considered in a single-server system. Another important factor to be considered is that deduplication systems must provide higher reliability especially while handling data that are sensitive or critical. At times, it is required that these data should be preserved over longer time periods. Therefore the deduplication system has to be such, it achieves higher reliability as well as higher security at the same time compared to that of the other high-available systems. In the existing methodology, each user has a unique master key for preserving the data privacy.

But, this approach is unreliable, since each user has to dedicatedly protect his own master key. Accidentally, if the master key is lost then the user can never be recovered and hence it paves a way for attackers to involve in data leakage. Yet another problem is that the number of keys gets increased with increasing number of users and it becomes difficult to manage not only for the storage of content as well as for the storage of keys.

Disadvantages

- Deduplication is not scalable as enormous numbers of keys are required with the increasing number of users.
- Cost increases to the storage of content as well as storage of keys.
- Lack of proper key management.
- Security becomes crucial if the master key is not protected.
- Easily prone to data leakage by attackers if the master key is lost.
- Less reliability due to lack of key management.

4. SYSTEM MODEL

We propose a new scheme to deduplicate the encrypted data ensuring higher reliability by ownership verification of the file with the help of a unique tag generated using MD5 algorithm.

In the figure 1, a detailed overview of system architecture of our proposed scheme is shown.

As described in the figure 1, the system consists of four different entities namely the User, Cloud server, N-key management servers and N-distributed block servers.

User:

User is an entity that makes use of the Cloud service for storage, retrieval and management of data in a cost-efficient environment. Cloud service allows the user to remotely store the data online and can access the data from any part of the world via the internet connection. The users are provided with a priority of data security, enjoy a cost-effective service and also take the advantage of an unlimited storage space.

Cloud server:

Cloud server is an important entity built with a logical server over the internet. Cloud server is also known as the virtual server. It possess the capabilities similar to that of a normal server but can be remotely accessed from anywhere in the world via the internet access.

An Example of a Cloud server that stores Electronic health records for doctors to access the patient's records instantly in a Military based health system.

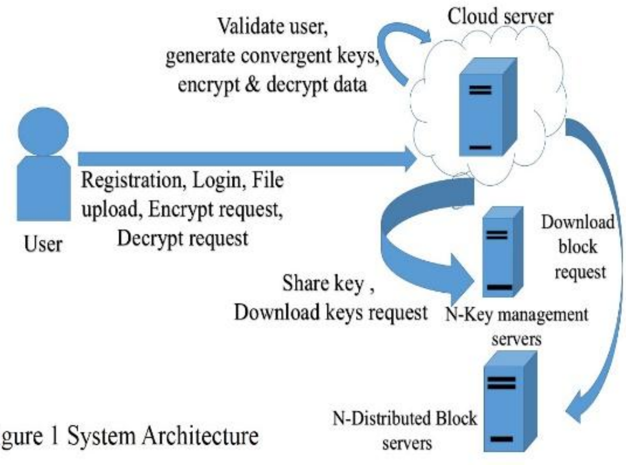

Figure 1 System Architecture

N-Key management servers:

Distributed key management servers are helpful in managing the keys storage in Cloud. Since the workload of storing keys in Cloud storage space adds overhead, the concept of distributed key management server is introduced. It provides improved scalability of key management with increasing number of keys and helps to protect sensitive data against the attackers.

N-Distributed block servers:

These servers are used to store the files as blocks in a distributed environment and the blocks are requested by the Cloud server in behalf of the Cloud user after successful ownership verification.

These are the components that are majorly involved in our system. The Cloud user will initially register to the Cloud server and logins to their account to perform storage, retrieval or deletion of the data. When the data is uploaded onto the storage, each file and its block will be provided with a unique tag generated by MD5 algorithm. The concept of ownership verification is performed to identify a valid user. The Cloud server generates convergent keys based on the hash and stores it in the key management server. These keys are later used to encrypt and decrypt the data content. The Cloud server in behalf of the Cloud user requests the blocks to the distributed block servers and provides it to the user. Thus providing an efficient deduplication system that improves scalability and reliability.

5. PROPOSED METHODOLOGY

The proposed methodology aims at providing a new deduplication system with higher reliability by splitting the file into several blocks and performing both file-level and block-level duplication checking.

Our scheme ensures the security by means of an ownership verification. The ownership verification is done by the Cloud server to identify the valid user when uploading or downloading the file from preventing against attackers. A unique tag is assigned for each file and each block that is generated using an MD5 algorithm.

Using the hash value, convergent key is created that performs encryption and decryption of the data content using Triple Data Encryption algorithm (3DES). Triple DES ensures high security and consistency to the data stored. Convergent encryption suffers from offline brute-force attacks, hence we generate the convergent key alone for our scheme and use it in another secure encryption 3DES instead of using convergent encryption for the entire data. An efficient distributed key management server helps in managing the convergent keys stored.

A csv file that contains the metadata on the file such as hash value is also maintained to check the duplication. One of the main key features of using this new deduplicated system is that the data deletion will only delete the reference to that copy but not the entire content and can be accessed by other users.

File Upload:

Suppose that a user wants to upload its file to the Cloud storage. The user initially has to make a registration to the Cloud and login with their account then he/she chooses an option to upload the file to the Cloud. Once after the file is uploaded, a unique signature tag is generated using MD5 algorithm. The produced hash tag is unique to only unique data contents.

Suppose that another user wants to upload the same file into the Cloud storage. Initially registration is done and then he/she chooses an option to upload the same file. Once after the upload process is completed, a unique signature is generated that matches with the previous signature indicating it's a duplicate copy. Now, the Cloud server stores only one physical copy of the same file after performing the ownership verification. Thus, file-level deduplication is achieved.

If the ownership verification becomes pass, the server provides the reference of the file to the requested user. If it fails, the upload operation is aborted.

For block-level deduplication, the file uploaded is split into several blocks and stored in the distributed block servers. Each block is associated with a unique tag generated by the algorithm. If the same block is being uploaded, the server checks the ownership of the user to guarantee the validity of the owner and provides access rights to the users for the same block. The server fails to upload if the verification fails. Thus, block-level deduplication is also achieved.

File Download: Consider the user who wants to download the data from the Cloud. The user requests for the download option to the server. Upon ownership verification, the Cloud provides the user to download its data contents. If the user requests for a block, the ownership is verified and blocks are provided from the distributed block servers to the Cloud servers which in turn serves the users with the requested block. The user upon receiving the cipher text decrypts the data with its private key to get the original data.

If the ownership verification fails, the download operation is aborted.

Data deletion: If the user requests for the data to be deleted from the Cloud storage, the server only deletes the reference of that user to the data but the entire file is not deleted for providing accesses for other users.

Algorithm 1: checking duplicate files:

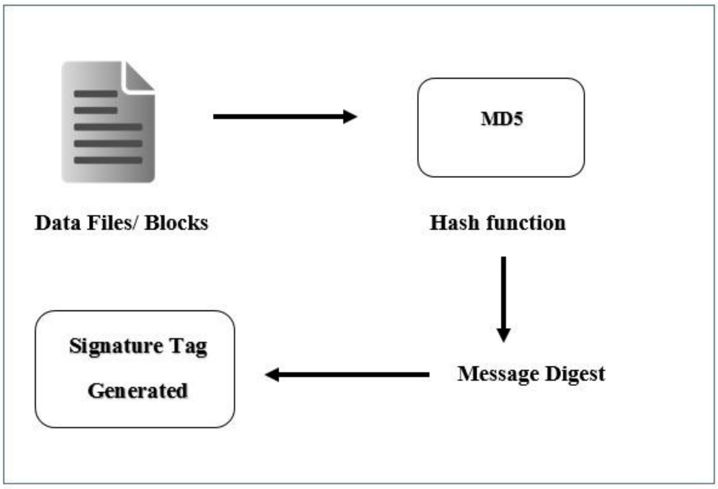

Figure 2: MD5 algorithm

--The input file uploaded is processed by the server.

--Generates a unique tag for the file using MD5.

--Server maintains a csv metadata file with stored tags.

--It becomes a Duplicate data, if the tag matches.

-- Not a Duplicate data, otherwise.

Algorithm 2: performing encryption:

--Generating convergent keys using the stored hashes of file.

--Perform 3DES encryption using convergent keys.

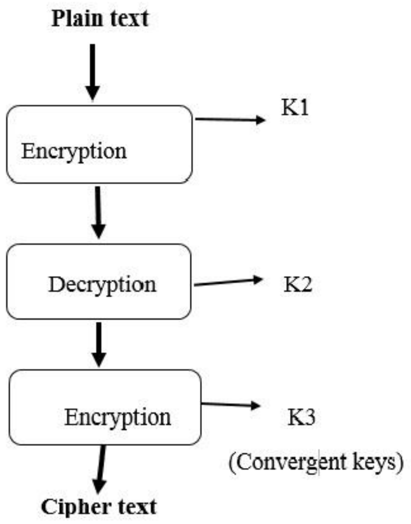

Figure 3: Triple data encryption algorithm

6. CONCLUSION

We propose a new scheme that efficiently supports Big Data in HDFS storage providers. An effective approach is implemented to verify data ownership through verification process. We integrate the scheme with data access control and deduplication in a simple way over a secure challenging environment.

The privacy of the data is also preserved by making use of strongly effective algorithms. The data chunks are distributed across the HDFS storage ensuring higher reliability. A reliable distributed key management is accomplished where the overhead of storing the keys in the Cloud storage is lessened.

Thus a new distributed deduplication system is achieved with higher reliability and security of data in HDFS storage.

6. REFERENCES

[1] Z. O. Wilcox, “Convergent encryption reconsidered,” 2011. [Online]. Available: http://www.mailarchive.com/cryptography@metzdowd.com/msg08949.html

[2] The Freenet Project, Freenet. (2016). [Online]. Available: https:// freenetproject.org/

[3] Atul Adya, William J Bolosky, Miguel Castro, Gerald Cermak, Ronnie Chaiken, John R Douceur, Jon Howell, Jacob R Lorch, Marvin Theimer, and Roger P Wattenhofer. Farsite: Federated, available, and reliable storage for an incompletely trusted environment. ACM SIGOPS Operating Systems Review, 36(SI):1–14, 2002.

[4] Z. Sun, J. Shen, and J. M. Yong, “DeDu: Building a deduplication storage system over cloud computing,” in Proc. IEEE Int. Conf. Comput. Supported Cooperative Work Des., 2011, pp. 348–355, doi:10.1109/CSCWD.2011.5960097

[5] M. Bellare, S. Keelveedhi, and T. Ristenpart, “DupLESS: Server aided encryption for deduplicated storage,” in Proc. 22nd USENIX Conf. Secur., 2013, pp. 179–194.

[6]M.Fuetal, “Acceleratingrestoreandgarbagecollectionindeduplication-based backup systems via exploiting historical information,” inProc. USENIX Annu. Tech. Conf., 2014, pp.181–192.

[7] J. Li, Y. K. Li, X. F. Chen, P. P. C. Lee, and W. J. Lou, “A hybrid cloud approach for secure authorized deduplication,” IEEE Trans. Parallel Distrib. Syst., vol. 26, no. 5, pp. 1206–1216, May 2015, doi:10.1109/TPDS.2014.2318320.

[8] P. Puzio, R. Molva, M. Onen, and S. Loureiro, “ClouDedup: Secure deduplication with encrypted data for cloud storage,” in Proc. IEEE Int. Cof. Cloud Comput. Technol. Sci., 2013, pp. 363–370, doi:10.1109/CloudCom.2013.54.

[9] C. Yang, J. Ren, and J. F. Ma, “Provable ownership of file in deduplication cloud storage,” in Proc. IEEE Global Commun. Conf., 2013, pp. 695–700, doi:10.1109/GLOCOM.2013.6831153.

[10] J.W.Yuanand S.C.Yu, “Secure and constant cost public cloud storage auditing with deduplication,” in Proc. IEEE Int.Conf. Communic. Netw.Secur.,2013,pp.145–153,doi:10.1109/CNS.2013.6682702.

[11] T. Y. Wu, J. S. Pan, and C. F. Lin, “Improving accessing efficiency of cloud storage using de-duplication and feedback schemes,” IEEE Syst. J., vol. 8, no. 1, pp. 208–218, Mar. 2014, doi:10.1109/ JSYST.2013.2256715.