Novelty Journals

ISSN 2394-7314

International Journal of Novel Research in Computer Science and Software Engineering
Vol. 3, Issue 1, pp: (231-236), Month: January-April 2016, Available at: www.noveltyjournals.com

Block Wise Data Deduplication with Integrity Auditing

Mamatha L[1], Dhananjaya M.K[2]

[1]2nd Year M.Tech, [2]Assistant Professor, Dept of CSE, RRIT, Bangalore, India

Abstract: **Data deduplication is one of important data compression techniques for eliminating duplicate copies of repeating data, and has been widely used in cloud storage to reduce the amount of storage space and save bandwidth. To protect the confidentiality of sensitive data while supporting deduplication, the convergent encryption technique has been proposed to encrypt the data before outsourcing.. In the proposed security model, Security analysis demonstrates that our deduplication systems are secure. The distributed deduplication systems future aim is to reliably store data in the cloud while achieving privacy and consistency. For achieving data deduplication we propose two types of deduplication methods: block wise data deduplication and file level deduplication with integrity auditing.**

Keywords: **Deduplication, Integrity auditing, Cloud Computing.**

I. INTRODUCTION

Data deduplication is a technique for eliminating duplicate copies of data, and has been widely used in cloud storage to reduce storage space and upload bandwidth. Promising as it is, an arising challenge is to perform secure deduplication in cloud storage. Although convergent encryption has been extensively adopted for secure deduplication, a critical issue of making convergent encryption practical is to efficiently and reliably manage a huge number of convergent keys. One critical challenge of today's cloud storage services is the management of the ever increasing volume of data. To make data management scalable deduplication we are use convergent Encryption for secure deduplication services.

Cloud computing provides seemingly unlimited "virtualized" resources to users as services across the whole Internet, while hiding platform and implementation details. Today's cloud service providers offer both highly available storage and massively parallel computing resources at relatively low costs. As cloud computing becomes prevalent, an increasing amount of data is being stored in the cloud and shared by users with specified privileges, which define the access rights of the stored data. One critical challenge of cloud storage services is the management of the ever-increasing volume of data.

Two types of deduplication in terms of the size : (a) block-level deduplication, which find out and eliminate redundancies among data blocks.(b)file-level deduplication, which determine redundancies between different files and eradicate these redundancies to decrease ability demands, and The file can be separated into lesser fixed-size. Using fixed-size blocks shorten the calculation of block bound-arise, even as using variable-size blocks .[ii]Despite the fact that deduplication method can accumulate the storage space for the cloud storage service providers, it decreases the consistency of the system.

Furthermore, the challenge for data privacy also arises as more and more sensitive data are being outsourced by users to cloud. Encryption mechanisms have usually been utilized to protect the confidentiality before outsourcing data into cloud. Most commercial storage service provider is reluctant to apply encryption over the data because it makes deduplication impossible. The reason is that the traditional encryption mechanisms, including public key encryption and symmetric key encryption, require different users to encrypt their data with their own keys. As a result, identical data copies of different users will lead to different cipher text. To solve the problems of confidentiality and deduplication, the notion of

convergent encryption has been pro-posed and widely adopted to enforce data confidentiality while realizing deduplication. However, these systems achieved confidentiality of outsourced data at the cost of decreased error resilience. Therefore, how to protect both confidentiality and reliability while achieving deduplication in a cloud storage system is still a challenge.

We implement new distributed deduplication system, which has more reliability. In that data chunks are distributed across multiple cloud servers. Deduplication technique can save the memory space for the cloud storage service providers; it reduces the reliability of the system. Security analysis indicate that our deduplication systems are secure in terms of the definitions specified in this security model. As a proof of concept, we implement the proposed systems that indicate the acquired aerial is very limited in actual environments. Deduplication process improves storage utilization & it saves storage space .That why it is useful in industry as well as in academic. It is useful in such application which has high deduplication ratio like as archival storage system. Furthermore, for the data privacy challenge is also arises more. The more sensitive data are redistributed by the users to cloud. Encoding have been usually Utilized, for to provide protection confidentiality before the redistributed data into cloud. Most commercial storage No of service providers are oppose to apply encryption over the data because it is impossible to make deduplication. The reason of that is the traditional encryption mechanism. In which including the public key encryption and symmetric key encryption have require number of users to encrypt their data with own key. For the result of similar data copy of the number of users will lead to the different Data has been encrypted. To solve the problems of confidentiality and deduplication, for solving the problem of deduplication we implement notation of the convergent encryption.

II. CONTRIBUTIONS

In this paper, we show how to design secure deduplication systems with higher reliability in cloud computing. We introduce the distributed cloud storage servers into deduplication systems to provide better fault tolerance. To further protect data confidentiality, the secret sharing technique is utilized, which is also compatible with the distributed storage systems. In more details, a file is first split and encoded into fragments by using the technique of secret sharing, instead of encryption mechanisms. These shares will be distributed across multiple independent storage servers. Furthermore, to support deduplication, a short cryptographic hash value of the content will also be computed and sent to each storage server as the fingerprint of the fragment stored at each server. Only the data owner who first uploads the data is required to compute and distribute such secret shares, while all following users who own the same data copy do not need to compute and store these shares any more. To recover data copies, users must access a minimum number of storage servers through authentication and obtain the secret shares to reconstruct the data. In other words, the secret shares of data will only be accessible by the authorized users who own the corresponding data copy.

Another distinguishing feature of our proposal is that data integrity, including tag consistency, can be achieved. The traditional deduplication methods cannot be directly extended and applied in distributed and multi-server systems. To explain further, if the same short value is stored at a different cloud storage server to support a duplicate check by using a traditional deduplication method, it cannot resist the collusion attack launched by multiple servers. In other words, any of the servers can obtain shares of the data stored at the other servers with the same short value as proof of ownership. Furthermore, the tag consistency, which was first formalized by [5] to prevent the duplicate/cipher text replacement attack, is considered in our protocol. In more details, it prevents a user from uploading a maliciously-generated cipher text such that its tag is the same with another honestly-generated cipher text. To achieve this, a deterministic secret sharing method has been formalized and utilized. To our knowledge, no existing work on secure deduplication can properly address the reliability and tag consistency problem in distributed storage systems.

This paper makes the subsequent contributions.

a) Four new secure deduplication systems are planned to provide economical deduplication with high reliability for file-level and block-level deduplication, respectively. The key rending technique, instead of ancient secret writing ways, is employed to protect knowledge confidentiality. Specifically, data are split into fragments by exploitation secure secret sharing schemes and keep at totally different servers. Our proposed constructions support each file-level and block-level deduplications.

b) Security analysis demonstrates that the planned deduplication systems are secure in terms of the definitions specified in the planned security model. In more details, confidentiality, responsibleness and integrity can be achieved in our planned system. Two kinds of collusion attacks are thought-about in our solutions. These are the collusion attack on the info and also the collusion attack against servers. Especially, the data remains secure notwithstanding the oppose controls a restricted range of storage servers.

c) We tend to implement our deduplication systems exploitation the Ramp secret sharing theme that permits high responsibleness and confidentiality levels. Our analysis results demonstrate that the new planned constructions are economical and also the redundancies are optimized and comparable the opposite storage system supporting identical level of responsibleness.

In previous deduplication systems cannot support differential authorization duplicate check, that is vital in several and applications. In such a licensed deduplication system, every user is issued a group of privileges throughout system data formatting.

III. PROPOSE MODEL

The distributed deduplication systems future aim is to reliably store data in the cloud while achieving privacy and consistency. Its main objective is to allow deduplication and distributed storage of the data diagonally multiple storage servers. As an alternative encrypting the data to keep the privacy of the data, new structures put on the top-secret intense technique to split data into shards. These shards will then be distributed transversely in multiple storage servers.

In our previous data deduplication systems, the non-public cloud is bothered as a proxy to allow knowledge owner/users to firmly perform duplicate talk over with differential privileges. Such style is sensible and has attracted lush attention from researchers. The data homeowners exclusively source their information storage by utilizing public cloud whereas the data operation is managed privately cloud. data deduplication is one among necessary data compression techniques for eliminating duplicate copies of repetition knowledge, and has been wide used in cloud storage to chop back the quantity of cabinet house and save system of measurement. To safeguard the confidentiality of sensitive data whereas supporting deduplication, Cloud computing provides ostensibly unlimited ‚virtualized' resources to users as services across the whole internet, whereas activity platform and implementation details. Today's cloud service suppliers offer every extra ordinarily offered storage and massively parallel computing resources at comparatively low costs. As cloud computing becomes rife, Associate in Nursing increasing amount of knowledge is being keep inside the cloud and shared by users with nominal privileges, that define the access rights of the keep data.

INTEGRITY AUDITING:

Integrity auditing refers to maintaining and assuring the accuracy and consistency of data over its entire life-cycle and is a critical aspect to the design, implementation and usage of any system which stores, processes/retrieve data. The definition of provable data possession (PDP) was introduced by Ateniese et al. [5][6] for assuring that the cloud servers possess the target files without retrieving or downloading the whole data. Essentially, PDP is a probabilistic proof protocol by sampling a random set of blocks and asking the servers to prove that they exactly possess these blocks, and the verifier only maintaining a small amount of metadata is able to perform the integrity checking.

Another line of work supporting integrity auditing is proof of retrievability (POR) [12]. Compared with PDP, POR not merely assures the cloud servers possess the target files, but also guarantees their full recovery. The first design goal of this work is to provide the capability of verifying correctness of the remotely stored data. The integrity verification further requires two features: 1) public verification, which allows anyone, not just the clients originally stored the file, to perform verification; 2) stateless verification, which is able to eliminate the need for state information maintenance at the verifier side between the actions of auditing and data storage.

SECURE DEDUPLICATION:

The second design goal of this work is secure deduplication. In other words, it requires that the cloud server is able to reduce the storage space by keeping only one copy of the same file. Notice that, regarding to secure deduplication, our objective is distinguished from previous work [3] in that we propose a method for allowing both deduplication over files

Novelty Journals

ISSN 2394-7314

International Journal of Novel Research in Computer Science and Software Engineering
Vol. 3, Issue 1, pp: (231-236), Month: January-April 2016, Available at: www.noveltyjournals.com

and tags. Data deduplication may be a specialized knowledge compression technique for eliminating duplicate copies of repetition knowledge. Connected and somewhat synonymous terms square measure intelligent (data) compression and single-instance (data) storage. This method is employed to boost storage utilization and might even be applied to network knowledge transfers to cut back the quantity of bytes that has to be sent. Within the deduplication method, distinctive chunks of information, or computer memory unit patterns, square measure known and hold on throughout a method of study. because the analysis continues, alternative chunks square measure compared to the hold on copy and whenever a match happens, the redundant chunk is replaced with a tiny low reference that points to the hold on chunk.

The file level and block level deduplication is used for higher reliability. The secret splitting technique is used for protect data. Our proposed structures support both traditional deduplication 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.

Block Diagram/Architecture of Proposed System:- When the user wants to upload and download the file from cloud storage at that time first user request to the web server for uploading file. It means only approved user can upload the file to web server for that purpose it use the proof of ownership algorithm . User to prove their relation of an owner to the thing possessed of data copies to the storage server. When file is uploaded it splits into blocks i.e by default size of block is 4KB.According to file size the block occurs. After that deduplication detection occurs.

In this paper, we show how to design secure deduplication systems with higher reliability in cloud computing. We introduce the distributed cloud storage servers into deduplication systems to provide better fault tolerance. To further protect data confidentiality, the secret sharing technique is utilized, which is also compatible with the distributed storage systems. In more details, a file is first split and encoded into fragments by using the technique of secret sharing, instead of encryption mechanisms. These shares will be distributed across multiple independent storage servers.

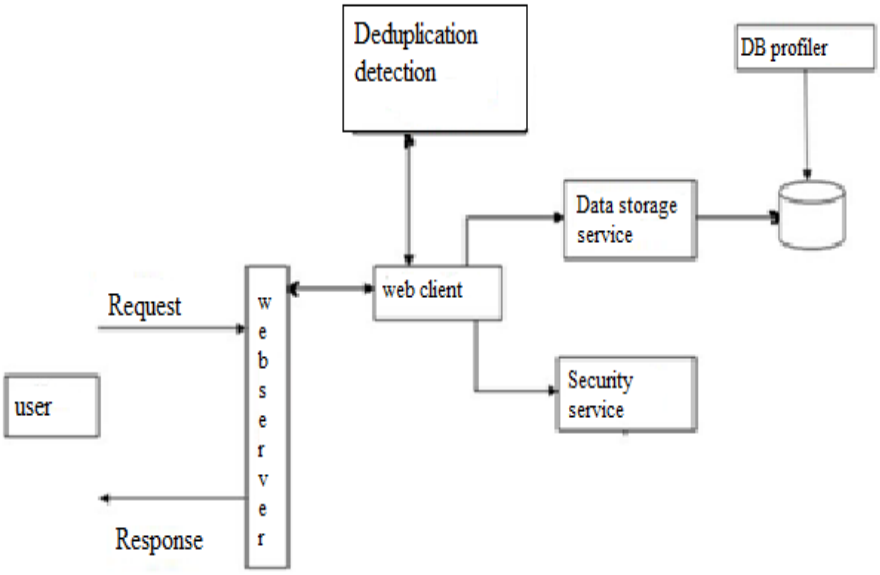

Figure 1: System Architecture of Secure deduplication

Furthermore, to support deduplication, a short cryptographic hash value of the content will also be computed and sent to each storage server as the fingerprint of the fragment stored at each server. Only the data owner who first uploads the data is required to compute and distribute such secret shares, while all following users who own the same data copy do not need to compute and store these shares any more. To recover data copies, users must access a minimum number of storage servers through authentication and obtain the secret shares to reconstruct the data. In other words, the secret shares of data will only be accessible by the authorized users who own the corresponding data copy. Two new secure deduplication systems are proposed to provide efficient deduplication with high reliability for file-level and block-level deduplication,

respectively. The secret splitting technique, instead of traditional encryption methods, is utilized to protect data confidentiality. Specifically, data are split into fragments by using secure secret sharing schemes and stored at different servers [1].

Modules:

File Level Deduplication Systems:

Figure2: File level duplication

Data deduplication can generally operate at the file, block or byte level thus defining minimal data fragment that is checked by the system for redundancy. The file-level deduplication can be performed most easily. It requires less processing power since files' hash numbers are relatively easy to generate. However, there is the reverse side of a medal: if only one-byte of a file is changed, its hash number also changes. As a result both file versions will be saved to storage.

Block Level Deduplication Systems:

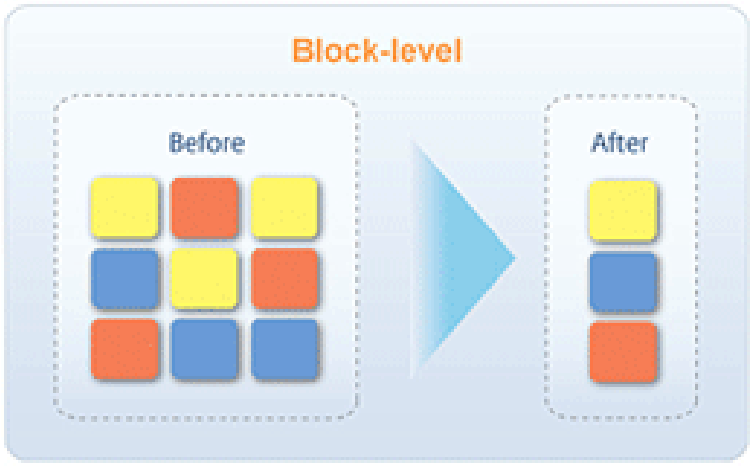

Figure3: Block Level duplication

Block deduplication requires more processing power than the file deduplication, since the number of identifiers that need to be processed increases greatly. Correspondingly, its index for tracking the individual iterations gets also much larger. Using of variable length blocks is even more source-intensive. Moreover, sometimes the same hash number may be generated for two different data fragments, which is called hash collisions. If that happens, the system will not save the new data as it sees that the hash number already exists in the index.

International Journal of Novel Research in Computer Science and Software Engineering
Vol. 3, Issue 1, pp: (231-236), Month: January-April 2016, Available at: www.noveltyjournals.com

IV. CONCLUSION

We implement the secure distributed deduplication systems to improve the reliability of data while achieving the secret of the clients outsourced data. Four constructions were proposed to support file-level and fine-grained block-level data deduplication. The security of tag consistency and integrity were achieved. We implemented our deduplication systems using the Ramp secret sharing scheme and demonstrated that it incurs small encoding/decoding overhead compared to the network transmission overhead in regular upload/download operations.

ACKNOWLEDGMENT

We would like to thank Management of RRIT for providing such a healthy environment for the successful completion of this work and express my gratitude to Mr. Dhananjaya M.K (Assistant Professor, RRIT, Bangalore) for providing continuous support and encouragement. Last but not the least I thank all my friends who has continuous support in all my works.

REFERENCES

[1] "Secure System with Improved Reliability Using Distributed Deduplication". A. G. Gangathade, Prof. Ms. V. D. Jadhav,

[2] Li, Yan Kit Li, Xiaofeng Chen, Patrick P. C. Lee, Wenjing Lou, ,A Hybrid Cloud Approach for Secure Authorized Deduplication', IEEE Transactions on Parallel and Cloud Systems,2014.

[3] Jin Li, Xiaofeng chen,xinyi huang,mohammadmehedi Hassanmember ,ieee and abdulhameed alelaiwi member "Secure distributed deduplication system with improved reliability" 2015

[4] J. R. Douceur, A. Adya, W. J. Bolosky, D. Simon, and M. Theimer, "Reclaiming space from duplicate files in a serverless distributed file system," in Proc. Int. Conf. Distrib. Comput. Syst.,2002, pp. 617–624.

[5] D. Ferraiolo and R. Kuhn, "Role-based access controls" in Proc.15th NIST-NCSC Nat. Comput. Security Conf., 1992, pp. 554–563.

[6] GNU Libmicrohttpd, (2012). [Online]. Available: http://www. gnu.org/software/libmicrohttpd/

[7] S. 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. Er. Navdeep Kochhar and Er. Arun Garg,"ECO

[8] W. K. Ng, Y. Wen, and H. Zhu, "Private data deduplication protocols in cloud storage," in Proc. 27th Annu. ACM Symp. Appl. Comput., 2012, pp. 441–446.

[9] A. Juels and B. S. Kaliski, Jr., "Pors: proofs of retrievability for large files," in Proceedings of the 14th ACM conference on Computer and communications security, ser. CCS '07. New York, NY, USA: ACM, 2007,