Authorized Secure Deduplication in Cloud Computing

R. V. Argiddi[1], **Prachi Sontakke**[2]

[1, 2]Walchand Institute of Technology, Solapur, India

Abstract: ***A service model where data is provided to many users over the world, is called as Cloud storage. Data in this model is stored in several logical pools. Cloud storage is a concept of storage beyond an interface where the storage is controlled and managed on demand. This technology gives benefits by many features like data backup and archival, no need of maintaining hardware resources, greater data accessibility, etc. Data deduplication is an important feature in cloud storage. Deduplication process recognizes and gets rid of the repeated data in the backup storage, indirectly improving the network bandwidth. Providing security along with the deduplication process is a challenging task. In this paper, we survey the Ontologies related to deduplication techniques to give future direction. The key aspects used for maintaining security in cloud is convergent encryption. To overcome certain drawbacks in this encryption technique, we will describing new techniques- cryptographic tuning and domain separation. Maintaining the authorized access for the user's confidential data, the concept called 'Proof of Ownership' is used for recognizing the respective user along with his access privilege.***

Keywords: Secure deduplication, Convergent Encryption, Proof of Ownership, Identification Protocol

1. Introduction

The Cloud computing is a model to provide access to computing resources and applications available on the Internet Cloud computing platform offers the network resources and storage space to the remote users. Information can be accessed by the user at anytime and from anywhere via Internet. So the user and his data need not to be on same physical location. Moreover, user even does not requires to manage the actual resources. Cloud computing enables the users to access shared resources by providing services as per user requirement over the network to perform operations. Managing and deploying certain applications developed by the users can be done through cloud computing services. The use of cloud computing has widespread in the IT industry. Companies like Microsoft, Google and IBM deliver their services to its users using cloud. Because of the cloud computing high scalability and availability it increases response time, which results in high performance and provides services to its users on a large scale.

Cloud computing era have lots of research issues. Deduplication is one of them. It is a compression technique which identifies and locates the duplicate data. It then eliminates duplicate copies of repeating data and saves the space for data that needs to be physically store. Hence, the two main advantages of data deduplication are Reduction in Storage Allocation and Efficient Volume of Replication.

1.1 Types of deduplication strategies

According to the operational area, data de-duplication strategies can be classified into two approaches.

1.1.1 File-level De-duplication

File level de-duplication, as the name suggests, is always performed over a single file. Identification of same hash value of two or more files determines that the files are similar

.

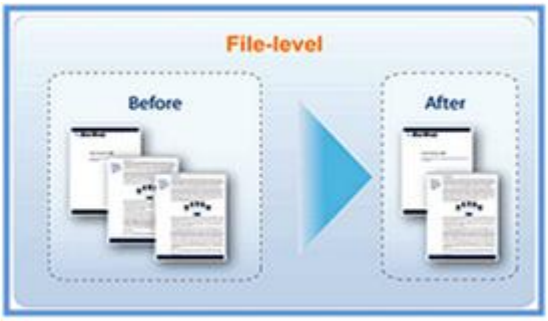

Figure 1: File-level Deduplication

1.1.2 Block-level Deduplication

Block level deduplication is performed over blocks. Firstly it divides the files into blocks and stores just a single copy of each block. Fixed-sized blocks or variable-sized chunks can be used with block-level deduplication.

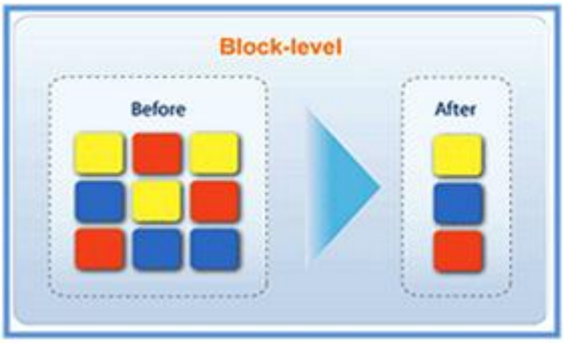

Figure 2: Block-level Deduplication

2. Literature Survey

Q. He, Z. Li, and X. Zhang talks about various deduplication techniques. The basic principle of deduplication is to maintain only one copy of the duplicate data provided with a pointer to point to the the duplicate blocks. This can be done at file level, block level or byte level. The new data are compared with old data at byte level and if they match, they are marked as duplicate. Data pointers are updated and the redundant copy is deleted.

Z. Li, X. Zhang, and Q. He, discusses various cloud storage techniques. With respect to data deduplication, they suggest to retain only the unique instance of the data, thus, reducing data storage volumes. An index of the digital signature is created by the data deduplication engine for the data segment along with the signature of a given repository to identify data blocks. To check whether the data block is already present, a pointer is provided by the index. In the copy operation, the data deduplication software found in a block of data inserts a link to the original data block index location instead of storing the data block again. Appearance of the same block more than once would generate more pointers to the indexing table. Moving data from one storage system to another which are at different geographical locations is referred to as data migration in cloud storage. It aims at cooperating and keeping load balance in cloud storage system. Migration of data into other cloud storage units should occur and the pointers to be kept in the old stored positions intact, or modify and update the index as changes occur. But this may bring overhead to network bandwidth and access bottleneck to concurrent clients.

S. Halevi, D. Harnik, B. Pinkas, and A. Shulman-Peleg proposed the notion of “proofs of ownership” (PoW) for deduplication systems in which a client can prove to a server based on Merkle trees and the error-control coding that it indeed has a copy of a file without actually uploading it. However, their scheme cannot guarantee the freshness of the proof in every challenge. Furthermore, their scheme has to build Merkle Tree on the encoded data, which is inherently inefficient. This scheme do not consider about data privacy.

The Proof of Ownership (PoW) concept is introduced by Halevi, a challenge-response protocol enabling a storage server to check if a requesting entity is the data owner, which is based on a short value. In other words, while uploading a data file (D) to the cloud, user first computes and sends a hash value hash = H(D) to the storage server. This later maintains a database of hash values of all received files, and looks up hash. If a match is found, then data file D is already outsourced to cloud servers. With respect to these cloud tags, there is no need to upload the file to remote storage servers. If there is no match found, then the user has to send the file data (D) to the cloud.

Douceur et al study the problem of deduplication in a multitenant system in which deduplication has to be reconciled with confidentiality. The authors propose the use of convergent encryption, i.e., deriving keys from the hash of the plaintext, so that two users will produce the same ciphertext from the same plaintext block, and the ciphertext can then be deduced.

M. W. Storer, K. M. Greenan, D. D. E. Long, and E. L. Miller point out some security issues with convergent encryption, while, proposing a security model and two protocols for secure data deduplication. There are two approaches to secure deduplication, viz., authenticated and anonymous. While the two models are similar, they each slightly differ in security properties. These both can be applied to single server storage as well as distributed storage. In the former, single server storage, clients interact with a single file server which stores both data and metadata. In the later, metadata is stored on an independent metadata server, and data is stored on a series of object-based storage devices (OSDs).

D. Harnik, B. Pinkas, and A. Shulman-Peleg discusses the shortcomings of client-side deduplication, and attacks to privacy and confidentiality that can be addressed without a full-edged POW scheme by triggering deduplication only after a small, but random, number of uploads.

J. Yuan and S. Yu. Here, the data owner outsources the erasure-coded file to the cloud server along with its corresponding authentication tags. The integrity of the outsourced file is audited by a user (who may not be the owner) who challenges the cloud with a challenging message. On receiving this message, the cloud generates the information proof based on the public key and sends it to the user. The user verifies the data integrity with the proof information, using our verification algorithm. In order to deduplicate the data, when a user wants to upload a data file that already exists in the cloud, the cloud server executes a checking algorithm to check if this user actually possesses the whole file. If the user passes the checking, he/she can directly use the file existed on the server without reuploading it.

Zhu et al try to solve the problem of disk bottlenecks while deduplicating. A significant challenge is to identify and eliminate duplicate data segments at a high rate on a low-cost system that cannot afford enough RAM to store and access a large metadata of the stored blocks.

Camble et al. proposes the “Sparse Indexing” deduplication system which uses a different approach to avoid the chunk lookup disk bottleneck. Here, the chunks are sequentially grouped into segments. These segments are then used to search similar existing segments using a RAM based index, which stores only a small fraction of the already stored chunks. In contrast to other approaches, Sparse Indexing allows to store a chunk multiple times if the similarity based system is not able to detect the segments, which already have stored the chunk. Therefore, Sparse Indexing is a member of the class of approximate data deduplication systems.

Stanek et al. presented a novel encryption scheme that provides differential security for both popular data and unpopular data. For popular data that are not particularly sensitive, the traditional conventional encryption is performed. For unpopular data, another two-layered encryption scheme with stronger security, while supporting deduplication is proposed. In this way, they achieved better tradeoff between the efficiency and security of the outsourced data.

M. Bellare, S. Keelveedhi, and T. Ristenpart. Dupless gives the Server aided encryption for deduplicated storage proposes different security mechanisms. Confidentiality can be preserved by transforming predictable message into unpredictable form. One new concept introduced as Key server (Third party auditor) to generate the file tag for duplicate check.

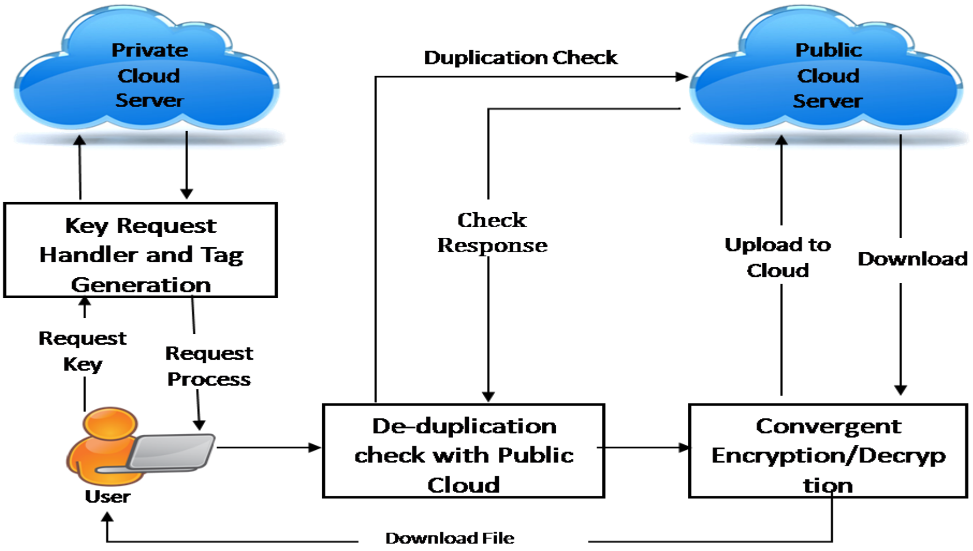

Figure 3: Architecture

3. Methodology

Figure 3 shows the architecture of our system model which includes main components such as key request handler and tag generation, deduplication check, convergent encryption and decryption. A new deduplication system obtained for differential duplicate check is proposed under this hybrid cloud architecture where in the public cloud resides the Secure Client Service Provider (S-CSP). The user is only allowed to perform the duplicate check for files marked with the corresponding privileges**.** To support authorized deduplication, the tag of a file *F* will be recognized by the file *F* along with the privilege. To show the difference with traditional notation of tag, we call it file token instead. To achieve authorized access, a secret key K_P will be bounded with a privilege *p* to generate a file token. Let $\emptyset_{F,P}$ = TagGen(*F*, K_P) be the token of *F* which is only allowed to access by user with privilege *p*. In another word, the token$\emptyset_{F,P}$, can only be computed by the users with the privilege *p*. As a result of this, if a file is been uploaded by a user as a duplicate token $\emptyset_{F,P}$ then a duplicate check that is sent from another user will be delivered successful if and only if he also has the file *F* and privilege *p*. This kind of token generation function could be quickly implemented as *H*(*F*, K_P), where *H*(.) denotes a cryptographic hash function.

Table 1: Comparison of State of Art Methods

Author	*Method*	*Feature*	*Result*
S. Halevi, D. Harnik, B. Pinkas, A. Shulman Peleg [5]	Proofs of Ownership in Remote Storage Systems	• Rigorous security • Identify attacks • Bandwidth saving • Time saving	Performance measurements indicate that the scheme incurs only a small overhead as compared to naive client-side deduplication.
M.Bellare, S.Keelveedhi, T.Ristenpart [8]	DupLESS ServerAided Encryption for Deduplicated Storage	• Saves space • The cross user is resolved • Security of deduplication: Strong security against External attacks is provided. • High Performance	Simple storage Interface.
J. Li, X. Chen, M. Li, J. Li, P. Lee, W. Lou [9]	A Secure deduplication is achieved with efficient and reliable convergent key management	• It reduces bandwidth & storage space • Efficient • Reliable key management • Provide confidentiality	Convergent key is shared across multiple server.
S. Nurnberger, S. Bugiel, A. Sadeghi, T. Schneider [10]	Twin clouds: An architecture for secure cloud computing	• Secure computation Stores large amount of data • Low latency • Secure execution environment	The trusted cloud is used as a proxy that provides a clearly defined interface for managing the outsourced data, queries, and programs.
W. K. Ng, Y. Wen, H.Zhu [11]	Private data deduplication protocols in cloud storage	• Improve speed of data duplication • Fault tolerant • Reduce cloud storage capacity.	Enhance the efficiency of data.

4. Results

Cloud storage and data deduplication techniques have pulled the attention of many researchers recently. He et al. [4] have researched over various cloud storage techniques and have recommended some techniques for reducing the storage volumes. The authors have proposed data deduplication engine that generated an index of digital signatures. This index also provides pointers for knowing the presence of data blocks.

A new concept of data migration is emerging now-a-days. It is nothing but the relocation of data from one storage to other storage. Both the storages are geographically separated. New concept known as Proof of Ownership is implemented for deduplication process by Halevi et al. [5]. Here a client can manifest based on Merkle-Hash Tree [5]. But the proposed system doesn’t take into attention providing the data security. The client-side deduplication has many limitations which are discussed by Harnik et al. [6]. An excellent survey on various deduplication techniques is done in [7]. The concept of deduplication is very simple, it says only a single copy of the duplicate data must be maintained and there should be a pointer for pointing the duplicate blocks. And this process can be attained in three levels: file level, byte level and block level.

The comparison of state of art methods are described in Table 1 based on methods, features and result.

5. Conclusion

The advanced research work in the fields of Cloud Computing has taken the field on a maturity level which is leading towards a very productive phase. This means the topics of cloud computing have become interesting to work on and also the issues of cloud computing are handled and addressed. But yet, cloud computing is still as much as a research topic to many researchers. In this paper we reviewed the deduplication techniques for better confidentiality and security in cloud computing. The detection of redundant data and removal of this redundant data is an important task for keeping the cloud storage clean and scalable. This duplicate data elimination has a great advantage for cloud storage. We have surveyed various techniques for deduplication.

6. Future Scope

The traditional approach of convergent encryption cannot be suited in this secure deduplication as it is susceptible to brute-force attack. To overcome this drawback, modified version of convergent encryption can be used by introducing two approaches - domain separation and cryptographic tuning. This gives a better authorized deduplication approach.

References

[1] K. Jin and E. Miller, "The effectiveness of deduplication on virtual machine disk images" In Proc. SYSTOR 2009: The Israeli Experimental Systems Conference..

[2] 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, 2013.Fröhlich, B. and Plate, J.

[3] S. Halevi, D. Harnik, B. Pinkas, and A. Shulman-Peleg. Proofs of ownership in remote storage systems. In Y. Chen, G. Danezis, and V. Shmatikov, editors, ACM Conference on Computer and Communications Security, pages 491–500. ACM, 2011.

[4] Z. Li, X. Zhang, and Q. He, Analysis of the key technology on cloud storage, in International Conference on Future Information Technology and Management Engineering, 2010, pp. 427428.

[5] S. Halevi, D. Harnik, B. Pinkas, and A. Shulman-Peleg. Proofs of ownership in remote storage systems. In Y. Chen, G. Danezis, and V. Shmatikov, editors, ACM Conference on Computer and Communications Security, pages 491500. ACM, 2011.

[6] D. Harnik, B. Pinkas, and A. Shulman-Peleg. Side channels in cloud services: Deduplication in cloud storage. IEEE Security & Privacy, 8(6), 2010.

[7] Q. He, Z. Li, and X. Zhang, Data deduplication techniques, in International Conference on Future Information Technology and Management Engineering,pp.431-432,2010.

[8] M. Bellare, S. Keelveedhi, and T. Ristenpart. Dupless:Server aided encryption for deduplicated storage. In USENIX Security Symposium, 2013.

[9] 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, 2013.

[10] S. Bugiel, S. Nurnberger, A. Sadeghi, and T. Schneider. Twin clouds: An architecture for secure cloud computing. In Workshop on Cryptography and Security in Clouds (WCSC 2011),2011.

[11] W. K. Ng, Y. Wen, and H. Zhu. Private data deduplication protocols in cloud storage. In S Ossowski and P. 2012.

http://dx.doi.org/10.21275/v5i6.NOV164306