Survey on Deduplication and Encrypted storage on Cloud Computing Systems

Vikram Arikath
Department of Computer Science and Engineering
SRM Institute of Science and Technology
Chennai, India.

T.L.Harshitha
Department of Computer Science and Engineering
SRM Institute of Science and Technology
Chennai, India.

Mrs.S.Sharanya
Assistant Professor
Department of Computer Science and Engineering
SRM Institute of Science and Technology
Chennai, India.

Dr.Revathi Venkataraman
Professor
Department of Computer Science and Engineering
SRM Institute of Science and Engineering
Chennai,India.

Abstract – This paper addresses the different types of deduplication and encrypted storage methods used for smart file storage and management in cloud computing systems. Cloud computing has had a vast impact on several industries over the last decade and is a preferred method for file sharing and storage. The stored files are susceptible to cyber-attacks and hence they are encrypted and stored. Other than the security, smarter management system within the cloud storage needs to be ensured to prevent wastage of storage space and bandwidth also ensuring faster and better access to files on the cloud. This can be done by promoting deduplication of files while uploading by comparing different parameters of the file being uploaded. The different strategies of encryption and how deduplication can be incorporated simultaneously are discussed as per the survey.

Keywords: Deduplication, Encryption, Hashing

I. Introduction

Cloud computing is a concept that has had a major impact on several industries over the last decade. Cloud computing can be simply defined as an infrastructure that shares its resources among multiple users using the internet as its medium. Users around the globe can access data or services that are stored and provided by the cloud. Over the past decade, the data that is being processed by companies worldwide is increasing at an exponential rate. It is not possible to store all these data in a single database due to the high costs of maintainability and the immobility of data when it is required by another user at a different location. Cloud computing helps to bridge the gap by providing a platform for online virtual storage that ensures data mobility and is also a cost-effective alternative. Now, since vast amounts of data are processed and stored in different clouds, the stored data is also susceptible to cyber-attacks. A third party might try to access the data stored in the cloud. To ensure that the data that is uploaded and stored in the cloud is safe, it is encrypted. The encryption ensures that the data uploaded cannot be easily read by other users and sensitive information does not get leaked. Another problem faced by cloud storage systems is

the duplication of files that contribute to wastage of file storage and bandwidth in the cloud system. There are methods to ensure deduplication[8][9] of files in the system by checking different parameters of the file and comparing it with pre-existing files on the cloud to check for duplicates. This process along with encryption helps to provide a smart storage system. Several methods of deduplication and encryption are already present, but the aim to maintain its coexistence in a cloud system has been a challenge for many years. The primary reason being that if deduplication and encryption existed on a single cloud server, for every file that is uploaded a special key would be generated to encrypt it resulting in a ciphertext. For each file hence, unique keys will be generated resulting in different ciphertexts. Even for similar files, since the key generated will be different the ciphertext will be different as well and thus this prevents the check of duplication within the system due to the randomization of ciphertexts for the same files. This problem has been circumvented by using different methods but a permanent solution for it is yet to be found. The structure of the paper is such that in Section II the existing methods are discussed followed by Section III that describes the Literature Survey with Conclusion and Future work forming the Section IV.

II. Existing Methods

From the fundamentals of encryption, the most basic form of encrypting a file is based on Symmetric Key Cryptography. There are mainly 3 components to a symmetric key encryption:

- Plaintext
- Key
- Ciphertext

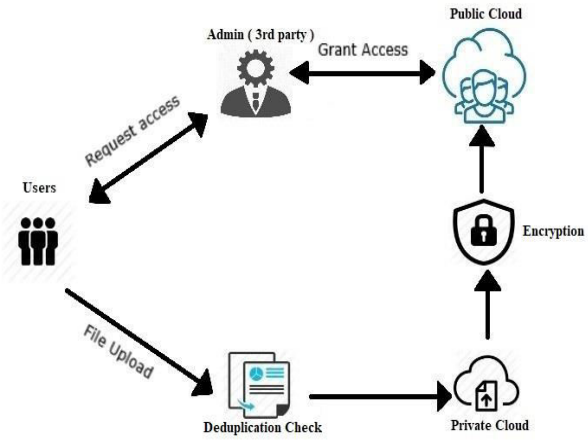
The plaintext along with the key generates the required ciphertext for the file that is being uploaded. So, the ciphertext is dependent on the plaintext and key always. Changing either one of them will result in a new cipher being generated altogether. The initial systems depended on keeping the same key for every file that is being uploaded. This resulted in similar ciphertext generated for similar files and hence the deduplication check was possible. This, however had a glaring security issue that once if the key is discovered by a third party, every file of the system would be compromised. So the notion of a better method to generate unique keys for each file and check for deduplication arose.

This challenge was solved by using Convergent Key Encryption [6] where it was based on an additional component compared to symmetric key encryption which was the procedure of Hashing. Hashing is a process that generates a unique identity of each file that is being uploaded into the system. The secret key generated for each file that is being uploaded was now the hash value of the file. This resulted in similar ciphertexts of similar files because similar plaintexts would generate the same hash values. There are several different hashing algorithms such as Rabin hash, MD4, MD5 (weak algorithms) and SHA-1, SHA-2 (strong algorithms) to generate the hash values of files that are being transferred to the cloud system. Hashing did not offer a permanent solution since the weaker algorithms which were faster, when used resulted in a lot of hash collisions and the stronger algorithms generated the required output but increased the computational complexity and the workload on the system causing time delays. However, to an extent on single cloud systems, the method of convergent key encryption prevailed. When the size of data that was being uploaded grew, the complexity of generating hashes grew as well.

The Hashing system[15] was implemented for File Level Deduplication check. This was followed by Chunk Level Deduplication check where a given file was broken into chunks. Each chunk was then made to generate a unique hash value which was then used to check for deduplication with existing chunks in the cloud. Static chunking[13] and Content defined chunking[10][11][12] were the two chunking methods

used on files. The former splits the files into same sized chunks whereas the latter divides the files into chunks of variable sizes. Static chunking ensured that the process is faster but however faces the Boundary shift problem if a single byte of data in the file is changed or deleted. Content defined chunking helped in overcoming this problem and also provides a more efficient method of checking for data deduplication but increased the complexity and caused time delays.

The methods discussed above are only with respect to a single cloud system. Hence a Hybrid Cloud System[2] comprising of both a public and private cloud was proposed to improve the efficiency of deduplication check methods and also ensure the co-existence of encryption as well. The system dealt with splitting the functions of deduplication and encryption by doing so on separate clouds. The private cloud helped in checking deduplication whereas the public cloud ensured in storing the encrypted file. The private cloud where the user first uploads the file is checked using different deduplication methods for a file already present in the private cloud and then later after checking, encrypts and stores the data in the public cloud. Encrypting and storing the data in the public cloud ensured a better and secure file sharing system. Since the private cloud was only accessible by the user there was no fear of outside indulgence in viewing or downloading the files. Deduplication strategies such as hash generation, simple content matching and chunking could be performed on the private clouds. The architecture of a hybrid cloud system is a shown.



III. Literature Survey

Mark W. Storer et al. in [1] proposed the need for deduplication in a cloud-based system and also the different strategies or models that can be implemented. The two specified models were Authenticated and Anonymous. The Authenticated Model ensure that the users were registered and their credentials were checked and monitored by an admin whereas the Anonymous Model ensured that the identities of the users were private but also gave room to malicious users registering within the system. Both systems follows the deduplication checking method of convergent key encryption. The key was only known to the user so even if the complete system was compromised the data could not be retrieved from the chunks that were stored. The method proposed could be implemented in a single server or distributed system.

The Twin Clouds architecture proposed in [2] set the base for the concept of the hybrid cloud architecture. Sven Bugiel et al. conceptualized a trusted cloud which performed the Security critical operations and

Commodity Cloud or the untrusted cloud that took care of the Performance critical operations. The query operations were done on the Commodity cloud that offered more storage space than the Trusted Cloud. This system could, in turn, be implemented to check deduplication as well.

In [3] the system has been implemented where a private cloud checks the files that are being uploaded for duplication with files already present in the system and then the file is encrypted with symmetric key AES encryption and stored in the cloud for sharing with other users. The implementation of both private and public cloud is to ensure that both deduplication and encryption can exist effectively in a single system.

The DupLESS method suggested in [4] is an architecture for a strong message locked encryption storage that is resistant to brute force attacks. Message locked encryption is a prominent form of convergent key encryption. The files are encrypted under the message-based keys which are received from a key server to ensure more security and resistance to attacks.

The DeKey[5] proposed by Jin Li et al. is responsible for managing all the convergent keys in the system. The working of the system was initially such that the user possessed a master key that encrypts the convergent keys before sending them to the cloud. The DeKey system ensures that the user does not have to manage the keys. The DeKey is also implemented in a realistic environment and the results are promising to show a more effective key management system.

The implementation of both encryption and deduplication is possible in a cloud with the use of convergent key encryption, in a single cloud system and also implementing the DeKey or the DupLess system adds to being effective cryptographic techniques. In a hybrid cloud system as well the processes can exist with the private cloud performing the deduplication check and the public cloud performing the encryption and storage.

Sn.no	Paper	**Author**	**Features**
1.	Secure Data Deduplication	Mark W. Storer, Kevin Greenan, Darrell D. E. Long, Ethan L. Miller	• Need for Deduplication • Authenticated Model – Users known and registered, more security no anonymity. • Anonymous Model – Unknown users, anonymity of users but less security.
2.	Twin Clouds: An Architecture for Secure Cloud Computing	Sven Bugiel, Stefan N¨urnberger, Ahmad-Reza Sadeghi, Thomas Schneider	• Twin cloud architecture • Trusted Cloud – Security Critical Operations. • Commodity Cloud – Performance Critical Operations.
3.	A Novel Approach for Securing Data-Deduplication Methodology in Hybrid Cloud Storage	Kameswari Bhaskar, Jayashree R, R. Sathiyavathi, L. Mary Gladence, V. Maria Anu	• Architecture for deduplication and encryption in a hybrid cloud environment. • Private Cloud – Deduplication check • Public Cloud – Encrypted Storage
4.	DupLESS: Server-Aided Encryption for Deduplicated Storage	Mihir Bellare, Sriram Keelveedhi, Thomas Ristenpart	• Message Locked Encryption • Secure architecture against brute force attacks

5.	Secure Deduplication with Efficient and Reliable Convergent Key Management	Jin Li, Xiaofeng Chen, Mingqiang Li, Jingwei Li, Patrick P.C. Lee, and Wenjing Lou	• Secure Key Management • DeKey – Distribute Convergent Keys Securely

IV. Conclusion and Future Work

In this paper, the different techniques of deduplication and how encryption and deduplication can co-exist in a single system are discussed. There are definite methods at present that cater to the needs of file deduplication and encryption. Deduplication ensures a better storage system of valuable cloud space and encryption ensures the stored data is not tampered with and its integrity is maintained. However, there are limitations to the methods as well.

A generalized chunking or hash generation algorithm is used on files to check the duplicate copies. In the future algorithms specific to the type of file that is being uploaded can result in a more effective deduplication strategy. For text files, text mining algorithms for plagiarism checking can be implemented for faster processing. Big Data files can use mapping algorithms and multimedia files can have algorithms implemented specific to their type. All these can contribute to building a secure and effective cloud system.

References

[1] Mark W. Storer Kevin Greenan Darrell D. E. Long Ethan L. Miller. Secure Data Deduplication. In StorageSS 2008, Fairfax, Virginia, USA.

[2] 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.

[3] Kameswari Bhaskar, Jayashree R, R. Sathiyavathi, L. Mary Gladence, V. Maria Anu. A Novel Approach for Securing Data-Deduplication Methodology in Hybrid Cloud Storage. In International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS), 2017.

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

[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, 2013.

[6] M. Bellare, S. Keelveedhi, and T. Ristenpart. Message locked encryption and secure deduplication. In EUROCRYPT, pages 296– 312, 2013.

[7] 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.

[8] D. Meyer and W. Bolosky, "A Study of Practical deduplication," in 9th USENIX conference on File and Storage Technologies (FAST' 11), SAN JOSE, California, 2011.

[9] J. Malhotra and J. Bakal, "A survey and comparative study of data deduplication techniques," in International Conference on Pervasive Computing (ICPC), Pune, 2015.

[10] W. Xia, Y. Zhou, H. Jiang, D. Feng, Y. Hua, Y. Hu, Q. Liu and Y. Zhang, "Fast CDC: A Fast and Efficient Content-defined Chunking Approach for Data Deduplication," in 2106 USENIX Conference on USENIX Annual Technical Conference, Berkeley, CA, USA, 2016.

[11] Y. Zhang, D. Feng, H. Jiang, W. Xia, M. Fu, F. Huang and Y. Zhou, "A Fast Asymmetric Extremum Content Defined Chunking Algorithm for Data Deduplication in Backup Storage Systems," IEEE Transactions on Computers, vol. 66, no. 2, pp. 199-211, February 2017.

[12] J. Wei, J. Zhu and Y. Li, "Multimodal Content Defined Chunking for Data Deduplication," Huawei Technologies, 2014.

[13] A. Li, S. Jiwu and L. Mingqiang, "Data Deduplication Techniques," Journal of Software, vol. 2, no. 9, pp. 916-929, 2010.

[14] B. Cai, Z. F. Li and W. Can, "Research on Chunking Algorithms of Data Deduplication," in International Conference on Communication, Electronics and Automation Engineering, Berlin, Heidelberg, 2012.

[15] G. Raj, "Deduplication Internals – Hash based deduplication: Part-2," [Online]. Available: https://pibytes.wordpress.com/2013/02/09/deduplication-internalshash-based-part-2/.

[16] J. Paulo and J. Pereira, "A survey and classification of storage deduplication systems," ACM Computing Surveys (CSUR), vol. 47, no. 1, 2014.