Volume No:1, Issue No:7 (December-2015) ISSN No : 2454-423X (Online)

International Journal of Research in Advanced Computer Science Engineering

A Peer Reviewed Open Access International Journal

www.ijracse.com

Distribution Storage Deduplication With Improved Reliability Systems

A.Soumya

Department of Information Technology

ABSTRACT:

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. However, there is only one copy for each file stored in cloud even if such a file is owned by a huge number of users. As a result, deduplication system improves storage utilization while reducing reliability. Furthermore, the challenge of privacy for sensitive data also arises when they are outsourced by users to cloud. Aiming to address the above security challenges, this paper makes the first attempt to formalize the notion of distributed reliable deduplication system. We propose new distributed deduplication systems with higher reliability in which the data chunks are distributed across multiple cloud servers. The security requirements of data confidentiality and tag consistency are also achieved by introducing a deterministic secret sharing scheme in distributed storage systems, instead of using convergent encryption as in previous deduplication systems. Security analysis demonstrates that our deduplication systems are secure in terms of the definitions specified in the proposed security model. As a proof of concept, we implement the proposed systems and demonstrate that the incurred overhead is very limited in realistic environments.

Keywords:

Deduplication, reliability, distributed storage system, secret sharing, encryption.

I.INTRODUCTION:

The various kinds of data for each user stored in the cloud and the demand of long term continuous assurance of their data safety, the problem of verifying correctness of data storage in the cloud becomes even more challenging. Cloud Computing is not just a third party data warehouse.

The data stored in the cloud may be frequently updated by the users, including insertion, deletion, modification, appending, reordering, etc. One critical challenge of today's cloud storage services is the management of the ever-increasing volume of data. According to the analysis report of IDC, the volume of data in the wild is expected to reach 40 trillion gigabytes in 2020. The baseline approach suffers two critical deployment issues. First, it is inefficient, as it will generate an enormous number of keys with the increasing number of users. Specifically, each user must associate an encrypted convergent key with each block of its outsourced encrypted data copies, so as to later restore the data copies. Although different users may share the same data copies, they must have their own set of convergent keys so that no other users can access their files. Second, the baseline approach is unreliable, as it requires each user to dedicatedly protect his own master key. If the master key is accidentally lost, then the user data cannot be recovered; if it is compromised by attackers, then the user data will be leaked. We propose Dekey, a new construction in which users do not need to manage any keys on their own but instead securely distribute the convergent key shares across multiple servers. Dekey using the Ramp secret sharing scheme and demonstrate that Dekey incurs limited overhead in realistic environments we propose a new construction called Dekey, which provides efficiency and reliability guarantees for convergent key management on both user and cloud storage sides. A new construction Dekey is proposed to provide efficient and reliable convergent key management through convergent key Deduplication and secret sharing. Dekey supports both file-level Deduplication. Security analysis demonstrates that Dekey is secure in terms of the definitions specified in the proposed security model. In particular, Dekey remains secure even the adversary controls a limited number of key servers. We implement Dekey using the secret sharing scheme that enables the key management to adapt to different reliability and confidentiality levels. Our evaluation demonstrates that Dekey incurs limited overhead in normal upload/download operations in realistic cloud environments.

Volume No:1, Issue No:7 (December-2015) ISSN No : 2454-423X (Online)

International Journal of Research in Advanced Computer Science Engineering

A Peer Reviewed Open Access International Journal

www.ijracse.com

The advantages of placing decoys in a file system are threefold:

(1) The detection of masquerade activity.

(2) The confusion of the attacker and the additional costs incurred to distinguish real from bogus information, and

(3) The deterrence effect which, although hard to measure, plays a significant role in preventing masquerade activity by risk-averse attackers.

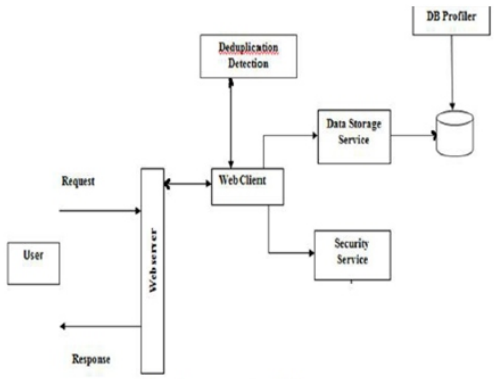

Fig: System Architecture

II.RELATED WORK:

Data deduplicationused for removing replication copies of data. Data deduplication techniques are very interesting techniques. The reliability produces stable and consistent results. They only focused on files without encryption, without considering the reliable deduplication over ciphertext. Ciphertext is also known as encrypted or encoded information. In 1997 M. Bellare introduced the idea of security and scheme for symmetric encryption in concentrate security framework [8]. They give different idea of security and analyze the good involution of reduction among them.

They provide method of encryption using a block cipher, cipher block chaining and counter mode. Its have two goals .First is to study the idea of security for symmetrical encryption and second is to provide concrete security analysis of fixed symmetric encryption device. Convergent encryption provides data security in deduplication [10]. Bellare explains the message locked encryption system and give it's application in secure outsourced storage [8]. Encryption is used to achieve the data privacy. Li et al incline in block level deduplication having some key management problems, through several servers [10].

Bellare et al. displayed how to protect private data through the conversion of predictable message into the unpredictable message [8]. In their system, another third party knew the key server. It was introduced to produce the file tag to check the replicate copies. Stanek et al. accomplish better efficiency and security of outsourced data storage [9]. They provide differential security for all types of data. D Harnik explains proof of ownership in Remote storage systems [5]. They identify attack that can lead to dropout data leakage and save time with the help of client side deduplication..

III. SYSTEM PREMELIRES:

A.Secure Deduplication:

Data deduplication is a specialized data compression technique for eliminating duplicate copies of repeating data. Related and somewhat synonymous terms are intelligent (data) compression and single-instance (data) storage. This technique is used to improve storage utilization and can also be applied to network data transfers to reduce the number of bytes that must be sent. In the deduplication process, unique chunks of data, or byte patterns, are identified and stored during a process of analysis.

As the analysis continues, other chunks are compared to the stored copy and whenever a match occurs, the redundant chunk is replaced with a small reference that points to the stored chunk. Given that the same byte pattern may occur dozens, hundreds, or even thousands of times (the match frequency is dependent on the chunk size), the amount of data that must be stored or transferred can be greatly reduced. This type of deduplication is different from that performed by standard file-compression tools, such as LZ77 and LZ78.

Whereas these tools identify short repeated substrings inside individual files, the intent of storage-based data deduplication is to inspect large volumes of data and identify large sections – such as entire files or large sections of files – that are identical, in order to store only one copy of it. This copy may be additionally compressed by single-file compression techniques. For example a typical email system might contain 100 instances of the same 1 MB (megabyte) file attachment. Each time the email platform is backed up, all 100 instances of the attachment are saved, requiring 100 MB storage space.

Volume No:1, Issue No:7 (December-2015) ISSN No : 2454-423X (Online)

International Journal of Research in Advanced Computer Science Engineering

A Peer Reviewed Open Access International Journal

www.ijracse.com

B.User Behavior Profiling:

We monitor data access in the cloud and detect abnormal data access patterns User profiling is a well known Technique that can be applied here to model how, when, and how much a user accesses their information in the Cloud. Such 'normal user' behavior can be continuously checked to determine whether abnormal access to a user's information is occurring. This method of behavior-based security is commonly used in fraud detection applications. Such profiles would naturally include volumetric information, how many documents are typically read and how often. We monitor for abnormal search behaviors that exhibit deviations from the user baseline the correlation of search behavior anomaly detection with trap-based decoy files should provide stronger evidence of malfeasance, and therefore improve a detector's accuracy.

C.Decoy documents:

We propose a different approach for securing data in the cloud using offensive decoy technology. We monitor data access in the cloud and detect abnormal data access patterns. We launch a disinformation attack by returning large amounts of decoy information to the attacker. This protects against the misuse of the user's real data. We use this technology to launch disinformation attacks against malicious insiders, preventing them from distinguishing the real sensitive customer data from fake worthless data the decoys, then, serve two purposes:

(1) Validating whether data access is authorized when abnormal information access is detected, and
(2) Confusing the attacker with bogus information.

IV. CONCLUSIONS:

We proposed the distributed deduplication systems to improve the reliability of data while achieving the confidentiality of the users' outsourced data without an encryption mechanism. 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.

REFERENCES:

[1] Amazon, "Case Studies," https://aws.amazon.com/solutions/casestudies/# backup.

[2] J. Gantz and D. Reinsel, "The digital universe in 2020: Big data, bigger digital 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. O. Rabin, "Fingerprinting by random polynomials," Center for Research in Computing Technology, Harvard University, Tech. Rep. Tech. Report TR-CSE-03-01, 1981.

[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 ICDCS, 2002, pp. 617–624.

[5] M. Bellare, S. Keelveedhi, and T. Ristenpart, "Dupless: Serveraided encryption for deduplicated storage," in USENIX Security Symposium, 2013.

[6] ——, "Message-locked encryption and secure deduplication," in EUROCRYPT, 2013, pp. 296–312.

[7] G. R. Blakley and C. Meadows, "Security of ramp schemes," in Advances in Cryptology: Proceedings of CRYPTO '84, ser. Lecture Notes in Computer Science, G. R. Blakley and D. Chaum, Eds. Springer-Verlag Berlin/Heidelberg, 1985, vol. 196, pp. 242–268.

[8] A. D. Santis and B. Masucci, "Multiple ramp schemes," IEEE Transactions on Information Theory, vol. 45, no. 5, pp. 1720–1728, Jul. 1999.

[9] M. O. Rabin, "Efficient dispersal of information for security, load balancing, and fault tolerance," Journal of the ACM, vol. 36, no. 2,pp. 335–348, Apr. 1989.

[10] A. Shamir, "How to share a secret," Commun.ACM, vol. 22, no. 11,pp. 612–613, 1979.

[11] J. Li, X. Chen, M. Li, J. Li, P. Lee, and W. Lou, "Secure deduplicationwith efficient and reliable convergent key management," inIEEE Transactions on Parallel and Distributed Systems, 2014, pp. vol.25(6), pp. 1615–1625.

Volume No:1, Issue No:7 (December-2015)
ISSN No : 2454-423X (Online)
International Journal of Research in Advanced Computer Science Engineering
A Peer Reviewed Open Access International Journal
www.ijracse.com

[12] S. Halevi, D. Harnik, B. Pinkas, and A. Shulman-Peleg, "Proofsof ownership in remote storage systems." in ACM Conference onComputer and Communications Security, Y. Chen, G. Danezis, andV. Shmatikov, Eds. ACM, 2011, pp. 491–500.

[13] J. S. Plank, S. Simmerman, and C. D. Schuman, "Jerasure: Alibrary in C/C++ facilitating erasure coding for storage applications- Version 1.2," University of Tennessee, Tech. Rep. CS-08-627,August 2008.

[14] J. S. Plank and L. Xu, "Optimizing Cauchy Reed-solomon Codesfor fault-tolerant network storage applications," in NCA-06: 5thIEEE International Symposium on Network Computing Applications,Cambridge, MA, July 2006.

[15] C. Liu, Y. Gu, L. Sun, B. Yan, and D. Wang, "R-admad: Highreliability provision for large-scale de-duplication archival storagesystems," in Proceedings of the 23rd international conference onSupercomputing, pp. 370–379.

[16] M. Li, C. Qin, P. P. C. Lee, and J. Li, "Convergent dispersal: Toward storage-efficient security in a cloud-of-clouds," in The 6thUSENIX Workshop on Hot Topics in Storage and File Systems, 2014.

[17] P. Anderson and L. Zhang, "Fast and secure laptop backups with encrypted de-duplication," in Proc. of USENIX LISA, 2010.

[18] Z. Wilcox-O'Hearn and B. Warner, "Tahoe: the least-authorityfilesystem," in Proc. of ACM StorageSS, 2008.

[19] A. Rahumed, H. C. H. Chen, Y. Tang, P. P. C. Lee, and J. C. S.Lui, "A secure cloud backup system with assured deletion andversion control," in 3rd International Workshop on Security in CloudComputing, 2011.

[20] M. W. Storer, K. Greenan, D. D. E. Long, and E. L. Miller, "Securedata deduplication," in Proc. of StorageSS, 2008.