IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 7, Issue 2, February 2018

A Comparitive Analysis of Data Deduplication in Cloud Storage

Reena Johnson[1], Prof M.V. Bramhe[2]

Department Of Computer Science and Technology, St.Vincent Pallotti College Of Engineering & Technology, Nagpur, India[1,2]

Abstract: With an increase in the usage of cloud storage, effective methods need to be employed to reduce hardware costs, meet the bandwidth requirements and to increase storage efficiency. This can be achieved using Data Deduplication. Data Deduplication is a method to reduce the storage need by eliminating redundant data. Thus by storing less data you would need less hardware and would be able to better utilize the existing storage space. Based on this idea, we design an encryption scheme that guarantees semantic security for unpopular data and provides weaker security and better storage and bandwidth benefits for popular data. This way, data deduplication can be effective for popular data, whilst semantically secure encryption protects unpopular content. We show that our scheme is secure under the Symmetric External Decisional Diffie-Hellman Assumption in the random oracle model.

Keywords: Cloud, Deduplication, Security, Bandwidth.

INTRODUCTION

Everyone is talking about the benefits of storing data to the cloud for sharing information among friends, to simplify moving data between different mobile devices, and for small businesses to back up and provide Disaster Recovery (DR) capabilities. But what about the massive amounts of data in enterprise data centers? How do cloud providers protect your data? How is the entire Internet protected?

Let's face it; backing up the data from your cellphone to the cloud is fairly routine. The hard job is on the back end, where service providers and large companies need to move, protect and store the massive amounts of data they have within and between their datacenters.

If you intend to move large amounts of data over a network and provide access to that data as a service, you need to be cognizant of network bandwidth requirements, data security and the total IT costs of providing those services to end users, especially when providing services for data storage and DR protection.

As an example, a basic requirement for any cloud-based data protection solution needs to be the ability to reduce the overall costs of providing the same service that clients could do themselves in their own data centers. One method being used to achieve this goal is data deduplication across multiple end-user clients, where the costs to provide the service is amortized over the number of paying clients. There are multiple methods to deduplicate data, so service providers and their customers need to be cognizant of the differences between the available solutions and the impact they may have on security and the ability to efficiently move, protect and store data in the cloud in a cost-effective manner.

Data deduplication is one of the recent trends in storage technologies as it enables companies to save a lot of money on storage costs to store the data and on the bandwidth costs to move the data when replicating it offsite for DR. This is great news for cloud providers, because if you store less, you need less hardware. If you can deduplicate what you store, you can better utilize your existing storage space, which can save money by using what you have more efficiently. If you store less, you also back up less, which again means less hardware and backup media. If you store less, you also send less data over the network in case of a disaster, which means you save money in hardware and network costs over time. The business benefits of data deduplication include:

- Reduced hardware costs;
- Reduced backup costs;
- Reduced costs for business continuity / disaster recovery;
- Increased storage efficiency; and
- Increased network efficiency.

IJARCCE
ISSN (Online) 2278-1021
ISSN (Print) 2319 5940
International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 7, Issue 2, February 2018

PROBLEM STATEMENT

Storage efficiency functions such as compression and deduplication afford storage providers better utilization of their storage backends and the ability to serve more customers with the same infrastructure. Data deduplication is the process by which a storage provider only stores a single copy of a file owned by several of its users. There are four different deduplication strategies, depending on whether deduplication happens at the client side (i.e. before the upload) or at the server side, and whether deduplication happens at a block level or at a file level. Deduplication is most rewarding when it is triggered at the client side,as it also saves upload bandwidth. For these reasons, deduplication is a critical enabler for a number of popular and successful storage services (e.g. Dropbox, Memopal) that offer cheap, remote storage to the broad public by performing client-side deduplication, thus saving both the network bandwidth and storage costs. Indeed, data deduplication is arguably one of the main reasons why the prices for cloud storage and cloud backup services have dropped so sharply.

Unfortunately, deduplication loses its effectiveness in conjunction with end to-end encryption. End-to-end encryption in a storage system is the process by which data is encrypted at its source prior to ingress into the storage system. It is becoming an increasingly prominent requirement due to both the number of security incidents linked to leakage of unencrypted data [1] and the tightening of sector-specific laws and regulations. Clearly, if semantically secure encryption is used, file deduplication is impossible, as no one| apart from the owner of the decryption key|can decide whether two ciphertexts correspond to the same plaintext. Trivial solutions, such as forcing users to share encryption keys or using deterministic encryption, fall short of providing acceptable levels of security.

As a consequence, storage systems are expected to undergo major restructuring to maintain the current disk/customer ratio in the presence of end-to-end encryption. The design of storage efficiency functions in general and of deduplication functions in particular that do not lose their effectiveness in presence of end-to-end security is therefore still an open problem.

LITERATURE SURVEY

Several deduplication schemes have been proposed by the research community [2{4] showing how deduplication allows very appealing reductions in the usage of storage resources [5, 6].Most works do not consider security as a concern for deduplicating systems; recently however, Harnik et al. [7] have presented a number of attacks that can lead to data leakage in storage systems in which client-side deduplication is in place. To thwart such attacks, the concept of proof of ownership has been introduced [8, 9]. None of these works, however, can provide real end-user confidentiality in presence of a malicious or honest-but-curious cloud provider.

Convergent encryption is a cryptographic primitive introduced by Douceur et al. [10, 11], attempting to combine data confidentiality with the possibility of data deduplication. Convergent encryption of a message consists of encrypting the plaintext using a deterministic (symmetric) encryption scheme with a key which is deterministically derived solely from the plaintext. Clearly, when two users independently attempt to encrypt the same file, they will generate the same ciphertext which can be easily deduplicated. Unfortunately, convergent encryption does not provide semantic security as it is vulnerable to content-guessing attacks. Later, Bellare et al. [12] formalized convergent encryption under the name message-locked encryption. As expected, the security analysis presented in [12] highlights that message-locked encryption offers confidentiality for unpredictable messages only, clearly failing to achieve semantic security.

Xu et al. [13] present a PoW scheme allowing client-side deduplication in a bounded leakage setting. They provide a security proof in a random oracle model for their solution, but do not address the problem of low min-entropy files.
Recently, Bellare et al. presented DupLESS [14], a server-aided encryption for deduplicated storage. Similarly to ours, their solution uses a modified convergent encryption scheme with the aid of a secure component for key generation. While DupLESS offers the possibility to securely use server-side deduplication, our scheme targets secure client-side deduplication.

The Proof of Ownership (PoW) is introduced by Halevi [8]. It is challenge-response protocol enabling a storage server to check whether a requesting entity is the data owner, based on a short value. That is, when a user wants to upload a data file (D) to the cloud, he first computes and sends a hash value hash = H(D) to the storage server.
This latter maintains a database of hash values of all received files, and looks up hash. If there is a match found, then D is already outsourced to cloud servers. As such, the cloud tags the cloud user as an owner of data with no need to upload the file to remote storage servers. If there is no math, then the user has to send the file data (D) to the cloud.
This client side deduplication, referred to as hash-as-aproof [16], presents several security challenges, mainly due to the trust of cloud users assumption. In 2002, Douceur et al. [4] studied the problem of deduplication in multi-tenant environment. The authors proposed the use of the convergent encryption, i.e., deriving keys from the hash of plaintext.

IJARCCE
ISSN (Online) 2278-1021
ISSN (Print) 2319 5940
International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 7, Issue 2, February 2018

Then, Storer et al. [13] pointed out some security problems, and presented a security model for secure data deduplication. However, these two protocols focus on server-side deduplication and do not consider data leakage settings, against malicious users. In order to prevent private data leakage, Halevi et al. [8] proposed the concept of Proof of Ownership (PoW), while introducing three different constructions, in terms of security and performances. These schemes involve the server challenging the client to present valid sibling paths for a subset of a Merkle tree leaves [11].

The first scheme applies erasure coding on the content of the original file. This encoded version is the input for construction of the Merkle tree. The second purpose pre-possesses the data file with a universal hash function instead of erasure coding. The third construction is the most practical approach. Halevi et al. design an efficient hash family, under several security assumptions. Unfortunately, the proof assumes that the data file is sampled from a particular type of distribution. In addition, this construction is given in random oracle model, where SHA256 is considered as a random function.

Recently, Ng et al. [12] propose a PoW scheme over encrypted data. That is, the file is divided into fixed-size blocks, where each block has a unique commitment. The hash-tree proof is then built, using the data commitments. Hence, the owner has to prove the possession of a data chunk of a precise commitment, with no need to reveal any secret information. However, this scheme introduces a high computation cost, as requiring generation of all commitments, in every challenging proof request.

In [3], the authors presented an efficient PoW scheme. They use the projection of the file into selected bit-position as a proof of ownership. The main disadvantage of this construction is the privacy violation against honest but curious storage server. In 2013, Jia et al. [16] address the confidentiality preservation concern in cross-user client side deduplication of encrypted data files. They used the convergent encryption approach, for providing deduplication under a weak leakage model. Unfortunately, their paper does not support a malicious storage server adversary.

TABLE 1 COMPARITIVE ANALYSIS OF DEDUPLICATION TECHNIQUE, APPROACH AND ADVANTAGES

S.N	Year	Deduplication Technique	Approach	Advantages
1	2016	Server side deduplication scheme for encrypted data	To solve issues of deduplication in situation where the data holder is not available or difficult to get involved	The performance of data deduplication is not influenced by the size of data, thus applicable for big data.
2	2015	proof of ownership - a protocol used to prove that the user indeed owns the same file when a duplicate is found	It makes use of hybrid cloud i.e. two clouds are used for handling data deduplication	i)enforce data confidentiality • Allows user to effectively prove his ownership to server • Incurs small overhead compared to naive client-side deduplication • Identifies attacks & saves bandwidth
3	2014	Dekey- provides reliable convergent key management through convergent key deduplication and secret sharing. It supports both file-level and block-level deduplications	Users do not need to manage keys on their own instead securely distribute the convergent key shares across multiple severs.	i)Secure deduplication and reliable management of conversion keys ii) The block level and file level deduplications are supported by Dekey iii) Limited overhead in normal upload/download operations in realistic cloud environment.
4	2013	Convergent encryption along with additional layer of encryption	A metadata manager and an additional server is defined the server adds additional encryption layer to prevent attacks against convergent encryption.	Prevents curious cloud storage providers from inferring the original contents of stored data.

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 7, Issue 2, February 2018

CONCLUSION

This paper discusses about data deduplication for the cloud based systems. It consists the methods that are used to achieve cost effective storage and effective bandwidth usage by deduplication.The core concept involves eliminating duplicate files using hashing algorithms. However, reliability and speed are at stake. The next challenge lies in identifying the most effective hashing algorithms for improving the speed of data storage and security. However, data deduplication is an important element for improving efficiency of the cloud system. This technique will play a major role in the cloud based services for storing backup data by both medium and large enterprises.

REFERENCES

[1] Open Security Foundation: DataLossDB (http://datalossdb.org/).
Meister, D., Brinkmann, A.: Multi-level comparison of data deduplication in a backup scenario. In: SYSTOR '09, New York, NY, USA, ACM (2009) 8:1{8:12 Mandagere, N., Zhou, P., Smith, M.A., Uttamchandani, S.: Demystifying data deduplication. In: Middleware '08, New York, NY, USA, ACM (2008)
[2] H. Miranda and L. Rodrigues, "Preventing Selfishness in Open Mobile Ad Hoc Networks," Proc. Seventh CaberNet Radicals Workshop, 2002.
[3] Aronovich, L., Asher, R., Bachmat, E., Bitner, H., Hirsch, M., Klein, S.T.: The design of a similarity based deduplication system. In: SYSTOR '09. (2009) 6:1{6:14
[4] Dutch, M., Freeman, L.: Understanding data de-duplication ratios. SNIA forum (2008) Http://www.snia.org/sites/default/files/Understanding_Data_ Deduplication_Ratios-20080718.pdf.
[5] Harnik, D., Margalit, O., Naor, D., Sotnikov, D., Vernik, G.: Estimation of deduplication ratios in large data sets. In: IEEE MSST '12. (april 2012)
[6] Harnik, D., Margalit, O., Naor, D., Sotnikov, D., Vernik, G.: Estimation of deduplication ratios in large data sets. In: IEEE MSST '12. (april 2012) 1 -11
[7] Harnik, D., Pinkas, B., Shulman-Peleg, A.: Side channels in cloud services: Deduplication in cloud storage. Security Privacy, IEEE 8(6) (nov.-dec. 2010) 40 -47
[8] Halevi, S., Harnik, D., Pinkas, B., Shulman-Peleg, A.: Proofs of ownership in remote storage systems. In: CCS '11, New York, NY, USA, ACM (2011) 491-500
[9] Di Pietro, R., Sorniotti, A.: Boosting e_ciency and security in proof of ownership for deduplication. In: ASIACCS '12, New York, NY, USA, ACM (2012) 81-82
[10] Douceur, J.R., Adya, A., Bolosky, W.J., Simon, D., Theimer, M.: Reclaiming space from duplicate _les in a serverless distributed _le system. In: ICDCS '02, Washington, DC, USA, IEEE Computer Society (2002) 617-632
[11] Storer, M.W., Greenan, K., Long, D.D., Miller, E.L.: Secure data deduplication.
[12] In: StorageSS '08, New York, NY, USA, ACM (2008) 1{10
[13] 12. Bellare, M., Keelveedhi, S., Ristenpart, T.: Message-locked encryption and securededuplication. In: Advances in Cryptology{EUROCRYPT 2013. Springer 296{312
[14] Xu, J., Chang, E.C., Zhou, J.: Weak leakage-resilient client-side deduplication of encrypted data in cloud storage. In: 8th ACM SIGSAC symposium. 195{206
[15] Bellare, M., Keelveedhi, S., Ristenpart, T.: DupLESS: server-aided encryption for deduplicated storage. In: 22nd USENIX conference on Security. (2013) 179{194
[16] Douceur, J.R.: The sybil attack. In: Peer-to-peer Systems. Springer (2002) 251{260
[17] Goldwasser, S., Micali, S.: Probabilistic encryption. J. Comput. Syst. Sci. (1984)
[18] Fahl, S., Harbach, M., Muders, T., Smith, M.: Condentiality as a service{usable security for the cloud. In: TrustCom 2012. 153{162
[19] Fahl, S., Harbach, M., Muders, T., Smith, M., Sander, U.: Helping johnny 2.0 to encrypt his facebook conversations. In: SOUPS 2012. 11{28
[20] Ateniese, G., Blanton, M., Kirsch, J.: Secret handshakes with dynamic and fuzzy matching. In: NDSS '07
[24] A. Pelc. Detecting errors in searching games. Journal of Combinatorial Theory Series A, 51(1):43–54, 1989...