DDCloud: A Data Deduplication for Storage System in a Dynamic Cloud

K. S. Suvidha
M.tech Dept. Of CSE
SVIT,Rajankunte
Bangalore-India

Dr. H .S. Ramesh Babu
Principal, Dept. Of CSE
SVIT, Rajankunte
Bangalore-India

Abstract **-Cloud computing has received a lot of popularity in the last few years and market observers believe it to be the future, but not if security problems persist. Cloud computing has accelerated with the wide use of cloud-based services for large scale content storage,processing, and distribution. Security and privacy are the major concerns for the public cloud environments. Cloud storage services commonly use deduplication,which eliminates redundant data by storing only a single copy ofeach file or block. Deduplication reduces the space and bandwidth requirements of data storage services, and is most effective when applied across multiple users, a common practice by cloud storage offerings. To protect the confidentiality of sensitive data while supporting deduplication, the convergent encryption technique has been proposed to encrypt the data before outsourcing. To better protect data security, this paper makes the first attempt to formally address the problem of authorized data deduplication .We propose a system service which assures block-level deduplication and data confidentiality at the same time. Although based on data encryption scheme, DDCloud remains securewith an additional encryption operation and an access control mechanism with higher authority privilege. To perform deduplication at block-levelraises an issue with respect to key management, in order to implement the keymanagement for each block together with the actual deduplicationoperation.**

Keywords – Cloud Storage, Data Security, Advanced Encryption Standard(AES), Deduplication, Confidentiality, Proof of Ownership.

I INTRODUCTION

This paper proposes a general architecture of cloud storage system, analyzes the functions of the components, and discusses the key technologies, etc. Cloud storage is a novel storage service mode which the service providers supply storage capacities and data storage services through the Internet to the clients; meanwhile, the clients needn't know the details and loweredstructures and mechanisms. The proposed architecture of cloud storage is layered and cooperative, and the discussed key technologies involve deployment, storage virtualization, data organization, migration, security, etc. The operation mechanism including ecology chain, game theory, ant colony optimization, data life cycle management, maintenance and update, convergence and evolution mechanisms are analyzed too. So an overall and new viewpoint to cloud storage system is illustrated.The market for cloud backup services in the personal computing environment is growing due to large volumes of valuable personal and corporate data being stored on desktops, laptops and smart phones. Source deduplication has become a mainstay of cloud backup that saves network bandwidth and reduces storage space.Nowadays, the explosive growth of digital contentscontinues to rise the demand for new storage and networkcapacities, along with an increasing need for more costeffectiveuse of storage and network bandwidth for datatransfer. As such, the use of remote storage systems is gaining an expanding interest, namely the cloud storage based services, since it provides cost efficient architectures.For saving resources consumption in both networkbandwidth and storage capacities, many cloud services,namely Dropbox, wuala and Memopal, apply client sidededuplication ([3], [5]). This concept avoids the storageof redundant data in cloud servers and reduces networkbandwidth consumption associated to transmitting the same contents several times. Despite thesenamely Dropbox, wuala and Memopal, apply client sidededuplication ([1], [2]). Client data deduplication brings many security issues,considerably due to the multi-owner data possessionchallenges [4]. For instance, several attacks target eitherthe bandwidth consumption or the confidentiality and theprivacy of legitimate cloud users. For example, a user maycheck whether another user has already uploaded a file, bytrying to outsource the same file to the cloud.Recently, to mitigate these concerns, many efforts have been proposed under different security models([3],[8], [5],). These schemes are called Identity based authentication. They allow the storageserver check a user data ownership, based on a static and short value (e.g. hash value). Along with low ownership costs and flexibility, users requirethe protection of their data and confidentiality guaranteesthrough encryption. Unfortunately, deduplication and encryptionare two conflicting technologies. While the aim of deduplicationis to detect identical data segments and store themonly once, the result of encryption is to make two identical datasegments indistinguishable after being encrypted. This meansthat if data are encrypted by users in a standard way, the cloudstorage provider cannot apply deduplication since two identical data segments will be different after encryption. On the other hand, if data are not encrypted by users, confidentiality cannot be guaranteed and data are not protected against curious cloud storage providers.A technique which has been proposed to meet these two conflicting requirements is data encryption [7], [8],[6] whereby the encryption key is usually the result of the hash of the data segment. In this paper, we cope with the inherent security exposures of convergent encryption and propose DD, which preserves the combined advantages of deduplication and data encryption. The security of DD relies on its new architecture whereby in addition to the basic storage provider, a metadata manager and an additional server are defined: the server adds an additional encryption layer to prevent well-known attacks against data encryption and thus protect the confidentiality of the data; on the other hand, the metadata manager is responsible of the key management task since block-level deduplication requires the memorization of a huge number of keys. Therefore, the underlying deduplication is performed at block-level and we define an efficient key management mechanism to avoid users to store one key per block.

II. SECURITY ISSUES

The attacks we describe can be applied to deduplication that is performed either at the file level or at the block level (to be concrete, we assume from now on that deduplication is performed at the file level). There are, however, two features of the deduplication service that are crucial for the attacks:

- Source-based deduplication. That is, deduplication mustbe performed at the client side. As mentioned above, this version of deduplication saves bandwidth and is therefore commonly used. The result of applying this approach is that the client can observe whether a certain file or block was deduplicated (or “deduped” in short). This can be done by either examining the amount of data transferred over the network, or by observing the log of the storage software, if that software provides this type of report. The second feature which is crucial for the attack is
- cross-user deduplication. That is, each file or block is compared to the data of other users, and is deduped if an identical copy is already available at the server. This approach is popular since it saves storage and bandwidth not only when a single user has multiple copies of the same data, but also when different users storecopies of the data. (Enterprise clients often store multiple copies of identical, or similar, data. We found out that this is true even for private customers: almost every common software manual or media file that we tried to backup using popular backup services was found to be already available on the servers and was therefore deduped). Identifying storage providers susceptible to the attack We performed the following test to identify services that perform source-based and cross-user deduplication (the test can be repeated by any reader, on the storage service of his ofher choice): (1) We installed the client software of the service on two different computers and created two different useraccounts; (2) We used one account to upload a file (in our tests this file was Sun’s VirtualBox software of size almost 73M); (3) We used the second account to upload the same file again, checking whether it is indeed uploaded. When the file was not re-transmitted over the network we concluded that the backup service performed source-based, cross-user deduplication. (Infact, when checking popular storage services there is no need to use two accounts, since, as described above, any popular file found on the web is likely to exist on the servers, as it was previously uploaded by other users. Therefore the test can consist of downloading a popular file from the web, uploading it to the service and checking whether deduplication occurs.)We identified services of three leading backup and file synchronizationproviders that perform cross-user, source-baseddeduplication. These services were (1) DropBox, a popular file sharing and backup service which crossed the 3 million user) Mozy, which is a leading provider of online backup for consumers and businesses, providing backup to over one million customers and 50,000 business users, andstoring more than 25 petabytes;3 and (3) Memopal which was ranked backup service in Europe.

III. RELATED WORK

Identity based authentication protocol enables astorage 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 [12], presents several security challenges, mainly due to the trust of cloud users assumption.

Many systems have been developed to provide secure storagebut traditional encryption techniques are not suitable for deduplication purposes. Deterministic encryption, in particular convergent encryption, is a good candidate to achieve both confidentiality and deduplication [10], [12] but it suffers from well-known weaknesses which do not ensure protection of predictable files against dictionary attacks [12], [9]. In order to overcome this issue, a secret value S is added to the encryption key. Deduplication will thus be applied only to the files of those users that share the secret. The new definition of the encryption key is K = H(S||M) where || denotes an operation between S and M. However, this solution overcomes the weaknesses of convergent encryption at the cost of dramatically limitingdeduplication effectiveness. Most importantly, learning the secret compromises the security of the system. Our approach provides data confidentiality without impacting deduplication effectiveness. Indeed, DD is totally independent from the underlying deduplication technique.

IV PROPOSED SYSTEM

Figure 1 illustrates a descriptive network architecture for cloud storage. It relies on the following entities for the good management of client data:

- Storage Cloud Service Provider (S-CSP): a S-CSP resides in the public cloud in which the user uploads the file, before uploading the files to the S-CSP the files are encrypted and outsourced to the cloud .
- Private cloud: Private cloud a new entity introduced for facilitating user's secure usage of cloud servicethe private cloud is involved as a proxy to allow data owner/users to securely per-form duplicate check with differential privileges. Such architecture is practical and has attracted much attention from researchers..
- Users/data owners: the users are able to perform the operations on the cloud, the authorized users can perform operations on cloud depending on their access rights which are granted by the Private cloud , like the rights to read, write or re-store the modified data in the cloud. These access rights serve to specify several groups of users. Each group is characterized by an identifier IDGand a set of access rights. In practice, the S-CSP provides a web interface for the users to store data into a set of cloud servers, which are running in a cooperated and distributed manner. In addition, the web interface is used by the users to retrieve, modify and restore data from the cloud, depending on their access rights. Moreover, the S-CSP relies on database servers to map useridentities to their stored data identifiers and group identifier

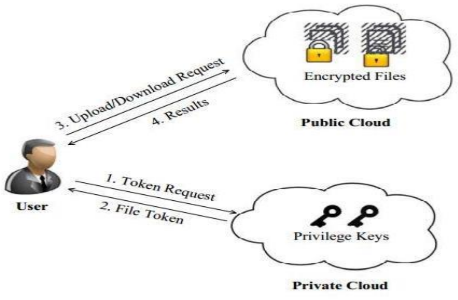

Fig. 1: Architecture of Authorized Deduplication check.

A. The Server

A simple solution to prevent the attacks against convergent encryption (CE) consists of encrypting the ciphertexts resulting from CE with another encryption algorithm using the same keying material for all input. This solution is compatible with the deduplication requirement since identical ciphertexts resulting from CE would yield identical outputs even after the additional encryption operation.

B.File-level Deduplication and Key Management

Even though the mechanisms of the server cope with the security weaknesses of CE, the requirement for deduplication at block-level further raises an issue with respect to key management. As an inherent feature of CE, the fact that encryptionkeys are derived from the data itself does not eliminate the need for the user to memorize the value of the key for each encrypted data segment. Unlike file-level deduplication, in case of block-level deduplication, the requirement to memorize and retrieve CE keys for each file in a secure way, calls for a fully-fledged key management solution. We thus suggest to include a new component, the metadata manager (MM), in the new DD system in order to implement the key management for each file together with the actual deduplication operation.

C.Security

The security requirements considered in this paper lie in two folds, including the security of file token and security of data files. For the security of file token, twoaspects are defined as unforgeability and indistinguishability of file token. The details are given below.

• Unforgeability of file token/duplicate-check token.Unauthorized users without appropriate privileges or file should be prevented from getting or generating the file tokens for duplicate check of any file stored at the S-CSP. The users are not allowed to collude with the public cloud server to break the unforgeability of file tokens. In our system, the S-CSP is honest but curious and will honestly perform the duplicate check upon receiving the duplicate request from users. The duplicate check token of users should be issued from the private cloud server in our scheme.

• Indistinguishability of file token/duplicate-check token.It requires that any user without querying the private cloud server for some file token, he cannot get any useful information from the token, which includes the file information or the privilege information.

•Data Confidentiality. Unauthorized users without appropriate privileges or files, including the S-CSP and the private cloud server, should be prevented from access to the underlying plaintext stored at S-CSP. In another word, the goal of the adversary is to retrieve and recover the files that do not belong to them. In our system, compared to the previous definition of data confidentiality based on convergent encryption, a higher level confidentiality is defined and achieved.

V CONCLUSION

We designed a system which achieves confidentiality and enables file-level deduplication at the same time. Our system is built on top of convergent encryption. We showed that it is worth performing file-level deduplication instead of blocklevel deduplication since the gains in terms of storage space are not affected by the overhead of metadata management, which is minimal. The solution is based on Advanced Encryption Standard used for enciphering the data file and asymmetric encryption for metadata files, due to the highest sensibility of these information towards several intrusions. As part of future work, DD may be extended with more security features such as proofs of retrievability [10], data integrity checking [12] and search over encrypted data [11]. In this paper we mainly focused on the definition of the two most important operations in cloud storage, that are storageand retrieval. We plan to define other typical operations such as edit and delete. After implementing a prototype of the system, we aim to provide a full performance analysis. Furthermore, we will work on finding possible optimizations in terms of bandwidth, storage space and computation.

REFERENCES

[1] Jia Xu, Ee-Chien Chang, and Jianying Zhou. Weak leakage-resilient client-side deduplication of encrypted data in cloud storage. InProceedings of the 8th ACM SIGSAC symposium on Information, computer and communications security, pages 195–206. ACM, 2013.

[2]Chuanyi Liu, Xiaojian Liu, and Lei Wan. Policy-based de-duplicationin secure cloud storage. In Trustworthy Computing and Services, pages 250–262. Springer, 2013.

[3]J. Xu, E.-C. Chang, and J. Zhou. Weak leakage-resilient client-sidededuplication of encrypted data in cloud storage. In Proceedings of the 8th ACM SIGSAC symposium on Information, computer and communications security, ASIA CCS '13, pages 195–206, New York, NY, USA, 2013. ACM.

[4] Dutch T Meyer and William J Bolosky. A study of practical deduplication. ACM Transactions on Storage (TOS), 7(4):14, 2012.

[5]W. K. Ng, Y. Wen, and H. Zhu. Private data deduplication protocols in cloud storage. In Proceedings of the 27th Annual ACM Symposium onApplied Computing, SAC '12, pages 441–446, New York, NY, USA, 2012. ACM.

[6]C. Wang, Z. guang Qin, J. Peng, and J. Wang. A novel encryption scheme for data deduplication system. In Communications, Circuits and Systems (ICCCAS), 2010 International Conference on, pages 265–269, 2010.

[7]D. Russell, Data Deduplication Will Be Even Bigger in 2010, Gartner, February 2010.

[8]Atul Adya, William J Bolosky, Miguel Castro, Gerald Cermak, Ronnie Chaiken, John R Douceur, Jon Howell, Jacob R Lorch, Marvin Theimer, and Roger P Wattenhofer. Farsite: Federated, available, and reliable storage for an incompletely trusted environment. ACM SIGOPS Operating Systems Review, 36(SI):1–14, 2002.

[9]John R Douceur, Atul Adya, William J Bolosky, P Simon, and Marvin Theimer. Reclaiming space from duplicate files in a serverless distributed file system. In Distributed Computing Systems, 2002. Proceedings. 22nd International Conference on, pages 617–624. IEEE, 2002.

[10]Luis Marques and Carlos J Costa. Secure deduplication on mobile devices. In Proceedings of the 2011 Workshop on Open Source and Design of Communication, pages 19–26. ACM, 2011.

[11] Perttula. Attacks on convergent encryption. http://bit.ly/yQxyvl.

[12] Jia Xu, Ee-Chien Chang, and Jianying Zhou. Weak leakage-resilient client-side deduplication of encrypted data in cloud storage. In Proceedings of the 8th ACM SIGSAC symposium on Information, computer and communications security, pages 195–206. ACM, 2013.