International Journal of Recent Trends in Engineering & Research
ISSN(ONLINE) : 2455-1457
IMPACT FACTOR : 3.344

A Secure Deduplication System at Client Side in Cloud Computing

Payal A.Mulik[1], Prof. Vivekanand Reddy[2]
[1]Student, Department of Computer Science and Engineering,
Visvesvaraya Technological University, Belagavi, Karnataka, India
[2]Faculty in Department of Computer Science and Engineering,
Visvesvaraya Technological University, Belagavi, Karnataka, India

Abstract: With the tremendous growth in data and number of users for cloud storage providers, data de-duplication becomes more important. De-duplication helps to store a unique copy of redundant data which results in reduce storage space and low bandwidth consumption. But the de-duplication result in new privacy and security challenges. We propose a new Secure Client Side de-duplication which provides secure de-duplication at client side with the help of convergent encryption and merkle tree.
Keywords- Cloud storage, Deduplication, Security, Privacy, Proof of Ownership

I. INTRODUCTION

Recent years the growth of digital contents is growing in large amount whether its enterprise world, business world or home. The major problem is able to store the data that is currently being generated. Cloud Computing allows to share various resources to the users. One critical challenge of cloud service providers is management of exponentially increasing volume of data. They use de-duplication technique to reduce storage space. It is a technique to store replicate data only once. It has proved to achieve high space and cost savings. It can reduce storage space upto 90-95% for backup applications and upto 68% in standard file systems.

Although de-duplication provides low storage space and less bandwidth consumption, users require the protection of their data and confidentiality measures to be taken. But, Deduplication and security are two conflicting technologies. The Deduplication is used to detect replicate copies of data and store them only once while security needs encryption. The encryption results in two data segments not identical after being encrypted. The cloud storage providers cannot apply Deduplication if the data are encrypted by users in a standard way, since two identical data will be different after encryption. On the other hand, if data blocks are not encrypted by users, security cannot be provided and data blocks are not protected against curious cloud storage providers.

A convergent encryption is used to encrypt the data blocks. It provides identical data blocks after encryption. It uses hash of the data as an encryption key.

In this paper, we propose a secure de-duplication at client side. It uses technique such as convergent encryption and Merkle tree to provide data security in cloud storage systems and efficient data Deduplication. The use of Merkle tree over encrypted data provides a unique identifier over data. This identifier helps to check the same data is present or not in cloud services. To prevent unauthorized access a secure proof of ownership protocol is used.

II. LITERATURE SURVEY

Douceur et al [2] proposed the use of convergent encryption i.e. deriving encryption keys from hash of plaintext.

Storer et al [8] find out some security problems and proposed secure data Deduplication but it focus on server side and do not consider security against malicious users.

Halevi et al [4] proposed a concept of proof of ownership (POW). It enables a server to check whether a request is from data owner based on hash value. A user can efficiently prove to cloud storage server that he owns the file by providing hash value. It uses Merkle-tree for de-duplication which includes the bounded leakage setting. The proposed concept focus only on data ownership, it does not consider data privacy.

Bellare et al [3] showed how to prevent the security of data by transforming the predicable message into unpredictable message to enhance the security of de-duplication and protect the confidentiality of the data. A new third party called key server was introduced to generate the file tag for duplicate check.
Xu et al [10] proposed a weak leakage resilient client side de-duplication model. They show a secure convergent encryption. The proposed technique only focuses on file level de-duplication. The issue of key management and block level de-duplication is not considered.

III. SECURITY ANALYSIS

PoW schemes bring several security challenges that may lead to sensitive data.
1) Exposure to data confidentiality– hash-as-a-proof schemes introduces an important data confidentiality concern, mainly due to the static proof client side generation. For instance, if a malicious user has the short hash value of an outsourced data file, he could fool the storage server as an owner trying to upload the requested data file. Then, he gains access to data, by presenting the hash proof. As such, an efficient PoW scheme requires the use of unpredictable values of verifications.

2) Privacy contravention – sensitive data leakage is a critical challenge that has not been addressed by Halevi et al. in [4]. That is, cloud users should have an efficient way to ensure that remote servers are unable to access outsourced data or to build user profiles.

3) Poison attack – when a data file D is encrypted on the client side, relying on a randomly chosen encryption key, the cloud server is unable to verify consistency between the uploaded file and the proof value hash. In fact, given a pair (hashD;Enck(D)), the storage server cannot verify, if there is an original data file D, that provides a hash value hash. As such, a malicious user can replace a valid enciphered file with a poisoned file. So, a subsequent user loses his original copy of file, while retrieving the poisoned version.

IV. SYSTEM ARCHITECTURE

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

Cloud Service Provider (CSP): a CSP has significant resources to govern distributed cloud storage servers and to manage its database servers. It also provides virtual infrastructure to host application services. These services can be used by the client to manage his data stored in the cloud servers.
Client: a client makes use of provider's resources to store, retrieve and share data with multiple users. A client can be either an individual or an enterprise.

Users: the users are able to access the content stored in the cloud, depending on their access rights which are authorizations granted by the client, like the rights to read, write or re-store the modified data in the cloud.

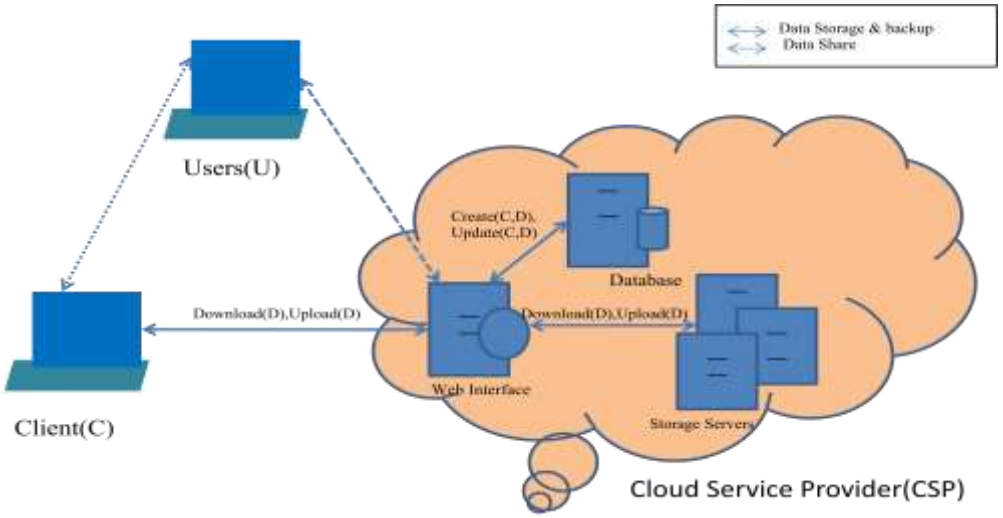

Fig. 1: Architecture of cloud data storage

V. METHODOLOGY

Our Secure Client Side data Deduplication scheme is based on original use of convergent encryption. Figure 2 describes the block diagram of proposed model. When the data owner wants to store a new enciphered file in cloud server he has to first generate enciphering key. The key can be derived by applying a one way hash function such as SHA-256. After key derivation, asymmetric encryption (ECC) is applied to original data file and then merkle tree is run over encrypted file to extract the unique identifier of the data file. The unique id, user permission and encrypted file are stored in cloud database. If the data owner wants to upload the duplicate file the cloud server checks whether the unique id is present in database. The server stops the transferring of encrypted file to cloud database if unique id is present in database. It helps to save the less consumption of network bandwidth. Data owner cannot upload the same file to the cloud again and he has to prove his ownership by providing the root value and a sibling value of Merkle tree along with his private key whenever he wants to access the data file that has been already outsourced into cloud. It ensures the highest level of privacy to cloud clients.
The flow of proposed work is as follows-

1) Apply hash function (SHA-256)
2) Asymmetric encryption on data blocks
3) Merkle based tree run over encrypted data and unique identifier derived
4) Cloud database maintain unique id, encrypted file and user permission
5) Authorized user decrypt the file and get the required content

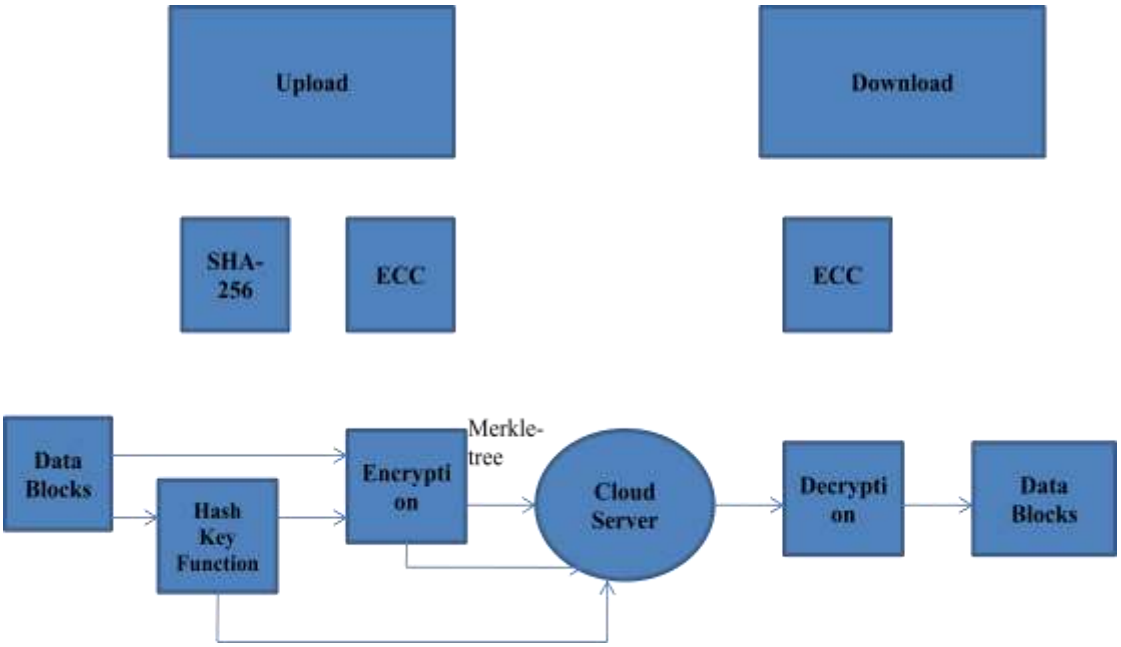

Fig 2: Block level diagram of proposed work

VI. EXPERIMENTAL RESULTS

We are implementing using Java and running it on a Pentium – IV with 1GB of RAM and Hard disk of 200 GB. The Operating framework utilized is Windows XP. Oracle 10g is used for backend data storage. Dropbox is used for cloud storage.

Fig 4: Client Login

Fig 7: File uploaded to cloud storage

Fig 8: Duplicate file not uploaded to cloud storage

VI. CONCLUSION

Secure client side Deduplication result in an efficient use of resources such as reduces storage space and less bandwidth consumption. It provides security to the content of data owner. It helps to eliminate storing same data repeatedly. Deduplication has benefits such as reduced infrastructure costs, reduce downtime and management cost. Secure Deduplication is achieved by convergent encryption and merkle based tree. Unauthorized users cannot access the data and highest level of privacy is achieved.

REFERENCES

1. https://github.com/openstack/swift.
2. 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 In Proceedings of 22nd International Conference on Distributed Computing Systems (ICDCS, 2002.
3. Mihir Bellare, Sriram Keelveedhi and Ristenpart, "Message-locked encryption and secure deduplication." in EUROCRYPT, 2013, pp. 296312.
4. S. Halevi, D. Harnik, B. Pinkas, and A. Shulman-Peleg. Proofs of ownership in remote storage systems. In Proceedings of the 18th ACM conference on Computer and communications security, CCS '11, pages 491–500, New York, NY, USA, 2011. ACM.
5. D. Hankerson, A. J. Menezes, and S. Vanstone. Guide to Elliptic Curve Cryptography. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2003.
6. D. Harnik, B. Pinkas, and A. Shulman-Peleg. Side channels in cloud services: Deduplication in cloud storage. IEEE Security And Privacy, 8(6):40–47, 2010.
7. R. C. Merkle. A digital signature based on a conventional encryption function. In A Conference on the Theory and Applications of Cryptographic Techniques on Advances in Cryptology, CRYPTO '87, pages 369–378, London, UK, UK, 1988. Springer-Verlag.
8. M. W. Storer, K. Greenan, D. D. Long, and E. L. Miller. Secure data deduplication. In Proceedings of the 4th ACM International Workshop on Storage Security and Survivability, StorageSS '08, pages 1–10, New York, NY, USA, 2008. ACM.
9. 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.
10. J. Xu, E.-C. Chang, and J. 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, ASIA CCS '13, pages 195–206, New York, NY, USA, 2013. ACM.

