International Journal of Scientific Research and Engineering Development-– Volume X Issue X, Year

Available at www.ijsred.com

RESEARCH ARTICLE OPEN ACCESS

KEYD: SECURE KEY-DEDUPLICATION WITH IDENTITY-BASED BROADCAST ENCRYPTION

Mr.Naveen Durai K[1], Dhivya bharathi S[2], Shobana sri S[3], Monisha S[4]

[1]*Assistant Professor, Department of Computer Science and Engineering, Sri Eshwar College of Engineering, Coimbatore, Tamilnadu-641202*

[2.3,4,]*UG Students, Department of Computer Science and Engineering, Sri Eshwar College of Engineering, Coimbatore, Tamilnadu-641202*

-----***-----

Abstract – Deduplication is a technique used for removing duplicate copies of data in the cloud in order to reduce the storage space and upload bandwidth. Before outsourced, the data which is about to be upload in the cloud will be encrypted for ensuring data confidentiality. Traditional encryption will produce different ciphertexts which are produced from the same plain text by different user's secret key, which makes difficult for deduplication. To overcome this problem, we go for Convergent Encryption which naturally encrypts the same plain texts into same ciphertexts. This project explains the problem of achieving reliable key management in secure deduplication. since we use the baseline approach for key management for maintaining an enormous number of keys with the increasing number of users where user have to protect their master key from the third party. So we designed a novel client-side deduplication protocol named KeyD by using Identity-based broadcast Encryption (IBBE) instead of the independent key management server. The user will interact with the cloud service provider(CSP) while uploading files and downloading it. Security Analysis explains that KeyD ensures data confidentiality and convergent key security at the same time it provides ownership privacy. Our scheme makes better tradeoff among storage cost, communication and computation overhead.

Key Words: Deduplication, proof of ownership, convergent encryption, key management.

1.Introduction

The system in which “KeyD secure Deduplication” is a web application which is used for avoiding the deduplication in the cloud where it relatively increases the storage space. By the year 2020, the volume of data will reach up to 40 trillion gigabytes. To make data management scalable, deduplication has been introduced. The process of providing ownership for files is to remove duplicates or replica information which is done automatically while uploading files. In our existing system, the encryption process of a single file for the number of times with different keys so that encrypted files are different in a different manner. In order to avoid multiple data copies with the same content, we go for both file level and block level granularities.

2. Convergent Encryption

Convergent encryption is a Cryptosystem which produces identical plaintexts files without encryption keys mainly used in deduplication for storing data in the cloud. In this, the user will compute cryptographically hash value from the data and this will be used to encrypt data, where the hash value is

used as a convergent key along with that user, will derive a tag to find duplication. ciphertext and the tag is sent to the server to check whether the file already exists in the cloud or not. This scheme is used for four algorithm

KeyGen(D) K is the key generation algorithm which map with the data files D to a convergent key K;

Encrypt (K, D) Symmetric encryption algorithm C which uses convergent key K and Data copy D as inputs to produce ciphertext as output C.

Decrypt(K, C) This algorithm takes convergent key and ciphertext as input and produces the plain text or original data copy as Output D.

TagGen(D) This algorithm produces a tag for each data copies.

3. Working Principle with Detailed Design

Process 1: User Authentication using the username and IP address provided.

Process 2: File Upload.

File upload process has certain steps to be followed

Step 1: Files will be categorized into block level and file level splitting so that it is easy to check whether the same data exists in the cloud.

Step 2: Convergent Encryption and initialization of convergent Key are done.

Step 3: Using the SHA algorithm, the hash value is generated and performs Identity based broadcast encryption.

Step 4: The user will interact with the cloud service provider to check whether the file already has ownership or not.

Step 5: If the file is already uploaded by the same owner or by other owners then the file will be uploaded or else the file will be uploaded in the cloud.

Process 3: we can View the file and Download it.

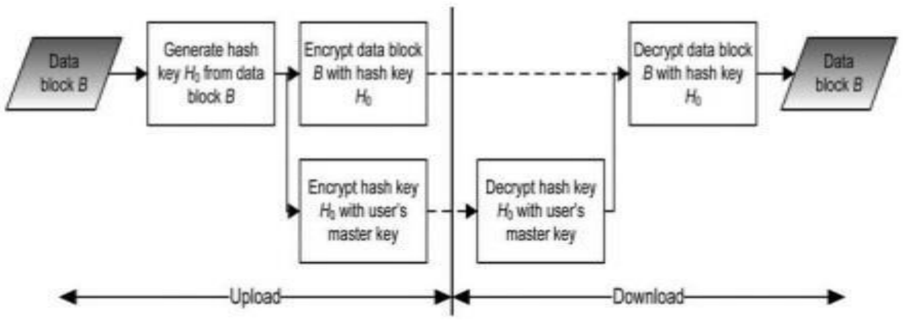
Data block B
Generate hash key H0 from data block B
Encrypt data block B with hash key H0
Encrypt hash key H0 with user's master key
Decrypt hash key H0 with user's master key
Decrypt data block B with hash key H0
Data block B
Upload
Download

Fig1. Block diagram

4. Module Description:

4.1 User Module

4.1.1 Register Module:

In a distributed system user wants to register with system IP, name and other details which can be stored in databases that can be maintained by admin.

4.1.2 Login Module:

The User can log in with IP address and name and also with secured captcha so that user allows to User Home page or otherwise it stays in the same page.

4.1.3 User Home Module:

In this module, the process of the project is explained through you can clearly know the concept of working.

4.1.4 Upload file Module:

In this module we are uploading the file in a database which can be secured by the convergent encryption process, now we are only generating a hash key.

4.1.4.1 File-level deduplication

Generating hash value for a file and comparing with the hash value of other files stored in the database.

4.1.4.2 Block level deduplication

Splitting the files into different blocks. And Generating hash value for each block and comparing with the hash value of other file blocks stored in the database.

4.1.5 View file Module:

This module shows the complete files that can be stored in the database by this particular User that can be downloaded for person usage

4.2 Admin module

4.2.1 Login Module:

The cloud storage spaces which are maintained by the administrator can authenticate to this module.

4.2.2 Deduplication Module

The duplication is a module which shows the data which tries to upload in the system. So that we can predict which user is supposed to the fake data in the cloud space.

4.2.3 Proof of Ownership

The Ownership of each user which is provided by the owner that is an admin. So that we can avoid duplicates. The admin is which gives single ownership for the single user.

5. Existing System

In the system, in which the data stored cloud must be encrypted as before stored in the cloud space, that can be encrypted by using the DES data encryption standard. So that while user fetching for the data it changes the ciphertext into the plain text.There occur some defects in storing data that may store some duplicate data for the number of times. Storing the same data needs huge storage.

5.1 Disadvantages

- There exists any duplication while storage into it.
- Occupy a number of storage for the same amount of data.

6. Proposed System

In our proposed system, the file uploaded in the cloud should avoid duplicated files. For this, we are using convergent key encryption which completely avoids the duplicated files stored for a number of times. There, we provide certain proof of ownership so that for single there only provide ownership for a single owner.

6.1 Advantages

- Reduces the Storage space and avoids duplicate files.
- Maintains Single Ownership for each file.
- Confidentiality of files while Upload/Download.
- Reduces the Storage space and exact retrieval of files.

7. Conclusion

In this paper, we propose a secure client-side deduplication scheme KeyD to effectively manage convergent keys. Data deduplication in our design is achieved by interactions between data owners and the Cloud Service Provider (CSP), without the participation of other trusted third parties or Key Management Cloud Service Providers. The security analysis shows that our KeyD ensures the confidentiality of data and security of convergent keys, and well protects the user ownership privacy at the same time. Experimental results demonstrate that the security of our scheme is not at the expense of the performance. For our future work, we will try to seek ways to protect the identity privacy of data owners, which is not considered in our scheme.

References

[1] Amazon Web Services, [Online]. Available:
https://aws.amazon.com/cn/.

[2] D.A. Sarma, X. Dong, and A. Halevy Bootstrapping pay-as-you-go data integration systems[C]. ACM SIGMOD International Conference on Management of Data, SIGMOD 2008, Vancouver, Bc, Canada, June. DBLP, 2008:861-874.

[3] J.R. Douceur, A. Adya, W.J. Bolosky, D. Simon, and M. Theimer, Reclaiming Space from Duplicate Files in a Serverless Distributed File System[C]. Distributed Computing Systems, 2002. Proceedings. 22nd International Conference on. IEEE, 2002: 617-
624.

[4] S. Ghemawat, H. Gobioff, and S. Leung, The Google File System[M].SOSP '03 Proceedings of the nineteenth ACM symposium on Operating systems principles, 2003, 37(5): 29-43.

[5] D. Borthakur, HDFS architecture guide[J]. Hadoop Apache Project,2008, 53.