ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 2, February 2016

Block-Level Encryption for Avoidance of Duplication in Large File System

R.Rajeshwari, S.Hemalatha, P.Esthercatherinpushpam

Assistant Professor, Dept of Information Technology, Velammal Institute of Technology, Panchetti, Tamilnadu, India

B.Tech, Dept of Information Technology, Velammal Institute of Technology, Panchetti, Tamilnadu, India

B.Tech, Dept of Information Technology, Velammal Institute of Technology, Panchetti, Tamilnadu, India

ABSTRACT: Many organizations have chosen to outsource data storage to cloud storage providers. This makes data management a critical challenge for the cloud storage providers. To achieve optimal usage of storage resources, many cloud storage providers perform de-duplication, which exploits data redundancy and avoids storing duplicated data from multiple users. File-level de-duplication is the data redundancy is exploited on the file level and thus only a single copy of each file is stored on the server. Block-level de-duplication, in which each file is divided into blocks, and the sever exploits data redundancy at the block level and hence performs a more fine-grained de-duplication.. In this work, we focus on the block-level de-duplication with fixed block size. Redundancy checking through Source-based de-duplication, unlike target-based de-duplication.

KEYWORDS: block-level redundancy checking in encrypted file for secure de-duplication,RSA algorithm is used for encryption**.**

I. INTRODUCTION

The main aim of the project is to achieve secure De-duplication of encrypted files for optimal usage of storage resources in cloud storage, which exploits data redundancy and avoids storing duplicated data from multiple users.So that, unique instance of the data is actually retained on cloud storage and also those redundant data is replaced with a pointer to the unique data copy.this paper addresses these issues by proposing Block-level de-duplication, in which each file is divided into blocks, and the sever exploits data redundancy at the block level and hence performs a more fine-grained de-duplication

Block-level de-duplication, in which each file is divided into blocks, and the sever exploits data redundancy at the block level and hence performs a more fine-grained de-duplication. It is worth noting that for block-level de-duplication, the block size can be either fixed or variable in practice, and each method has its advantages and disadvantages. In this work, we focus on the block-level de-duplication with fixed block size. Redundancy checking through Source-based de-duplication, unlike target-based de-duplication, the user first sends an identifier/tag of the data before uploading the data to the server for redundancy checking and thus duplicated data would not be sent over the network, if the file already exists then it securely de-duplicates the File level and block level of encrypted data.

II. RELATED WORKS

In [2] authors used the encryption system for securing the message before it is being checked for redundancy in file system,that why it is termed as secure de-duplication,where in[3] the comparisons between the file-level data de-duplication technology vs block level data de-duplication technology,which clearly states the drawback in file-level data de-duplication technology and then reason/advantage of block level data de-duplication technology.In [4] authors states the new algorithms in securing the file system before it has been checked for redundancy after that, the file level redundancy checking will be done in that encrypted files.In[5] authors proposed a new system of reclaiming the space from duplicate files by means of serverless distributed file system which means the concept of distributed file system is used here.In[6] ,authors describe cryptographic schemes for the problem of searching on encrypted data and provide

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 2, February 2016

proofs of security for the resulting crypto systems.there techniques have a number of crucial advantages. They are provably secure: they provide provable secrecy for encryption, in the sense that the untrusted server cannot learn anything about the plaintext when only given the ciphertext.

III. PROPOSED SYSTEM

This paper addresses these issues by proposing Block-level de-duplication, in which each file is divided into blocks, and the sever exploits data redundancy at the block level and hence performs a more fine-grained de-duplication also it can eliminate chunks of data smaller than afile. It is worth noting that for block-level de-duplication, the block size can be either fixed or variable in practice. In this work, we focus on the block-level de-duplication with fixed block size. Fortunately, file-level de-duplication and block-level de-duplication are not incompatible with each other. In this paper, we present a technique that can achieve both of them (i.e., dual-level de-duplication).Redundancy checking through Source-based de-duplication, unlike target-based de-duplication, the user first sends an identifier/tag of the data before uploading the data to the server for redundancy checking and thus duplicated data would not be sent over the network, if the file already exists then it securely de-duplicate the File level and block level of encrypted data.
For example, suppose the server performs file-level de-duplication, which means only one copy of File F will be saved. Later, user downloads F, appends several new pages to it, and uploads the modified file (denoted by F') to the server. Since F'is different from F, the server needs to store the whole file F. However, if block-level de-duplication is used, the server only needs to store the appended Pages.

IV. PSEUDOCODE

The steps involved are:
1.the user should be authorized user,for which authentication will be done for each user.thus the authorization will be done by administrator.
2.once the user is identified as authorized user then user can make request for the key generation,based upon the request given by the user it will generate key.
3.user can ensure whether to perform file level or block level redundancy checking which depends upon the size of the file user wants to upload.
4.if it has to perform file-level de-duplication then usual file level redundancy checking will be done.
5.else,block-level de-dupulication will be done.

V. SUB -SYSTEM DESCRIPTION

SUB-SYSTEMS ARE:

Authentication and Key Generation
Redundant File Checking and uploading.
File level de-duplication
Block level de-duplication

Authentication and Key Generation

A dishonest user who has learnt a piece of information about a file may claim that he/she owns the file. To overcome such an attack, the user proves to the server that he/she indeed owns the entire file through authentication. In reality, end users may not entirely trust the cloud storage servers. In order to protect data privacy, files may be encrypted first before being uploaded to the server. Here user request key the Admin to encrypt file, Admin will generate the key to user.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 2, February 2016

Redundant File Checking and uploading

Dual-Level Source-Based (DLSB) De-duplication for large files system, the user firstly sends an encrypted file identifier to the server for file redundancy checking. If the file to-be-stored is not exist in cloud storage then the user uploads the encrypted file. Otherwise the user uploads the identifiers/tag of all the encrypted file blocks to the server for block-level de-duplication checking and the user uploads encrypted data blocks which are not stored in the server.

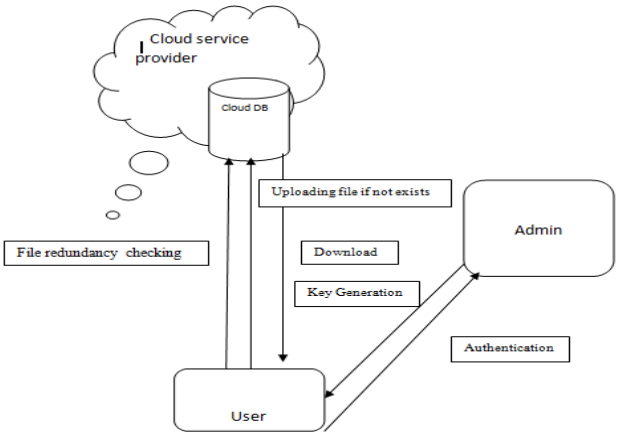
File level de-duplication

While redundancy checking if the file already exist then File-level de-duplication will be perform in the storage. In which the data redundancy is exploited on the file level and those redundant data is replaced with a pointer to the unique data copy.

Block level de-duplication

While redundancy checking if the block already exist then Block-level de-duplication will be perform in the storage. In which the data redundancy is exploited on the block level and those redundant block is replaced with a pointer to the unique data copy.
Block-level de-duplication, in which each file is divided into blocks, and the sever exploits data redundancy at the block level and hence performs a more fine-grained de-duplication also it can eliminate chunks of data smaller than a file. It is worth noting that for block-level de-duplication, the block size can be either fixed orvariable in practice.

VI. ARCHITECTURE DIAGRAM



VII. DESIGN AND IMPLEMENTATION CONSTRAINTS

Constraints in Analysis

- Constraints as Informal Text
- Constraints as Operational Restrictions
- Constraints Integrated in Existing Model Concepts
- Constraints as a Separate Concept
- Constraints Implied by the Model Structure

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 2, February 2016

Constraints in Design

- Determination of the Involved Classes
- Determination of the Involved Objects
- Determination of the Require Clauses
- Global actions and Constraint Realization

Assumptions and Dependencies

A hierarchical structuring of relations may result in more classes and a more complicated structure to implement. Therefore it is advisable to transform the hierarchical relation structure to a simpler structure such as a classical flat one. It is rather straightforward to transform the developed hierarchical model into a bipartite, flat model, consisting of classes on the one hand and flat relations on the other. Flat relations are preferred at the design level for reasons of simplicity and implementation ease. There is no identity or functionality associated with a flat relation. A flat relation corresponds with the relation concept of entity-relationship modeling and many object oriented methods.

System Features

In the system of protocol sequences for broadcasting messages in VANETs, because of mobility, two mobile users within their hearing range may use the same protocol sequences. In this case, they may have two or more collided packets within a period. However, if the relative delay offsets are uniformly distributed, the probability that the Hamming autocorrelation is nonzero is $(2p - 1) / (p\ q)$. The chance for having collided packets between two users with the same protocol sequence is not high if q is large enough. As the protocol sequences can be reassigned at roadside nodes, the effect of duplicate sequences can be further mitigated. If necessary, additional measures may be taken to avoid total blocking. We introduce a hybrid scheme called the *random hopping* scheme. A user makes a random "delay shift" after a certain period of time. The random hopping scheme is between the non-persistent scheme and the deterministic protocol sequence- based scheme. There is a design parameter T. If T is set to 1, the random hopping scheme is the same as the non-persistent scheme, and when $T \rightarrow \infty$, the random hopping scheme is the protocol-sequence-based scheme

VIII. SIMULATION RESULT

The simulation results brought the following levels of sub-systems,which could be explained with data flow diagrams.

LEVELS OF DESIGN

DFD LEVEL 0(Authentication and Key Generation)

i. A dishonest user who has learnt a piece of information about a file may claim that he/she owns the file.
ii. To overcome such an attack, the user proves to the server that he/she indeed owns the entire file through authentication.
iii. In reality, end users may not entirely trust the cloud storage servers.
iv. In order to protect data privacy, files may be encrypted first before being uploaded to the server.
v. Here user request key the Admin to encrypt file, Admin will generate the key to user.

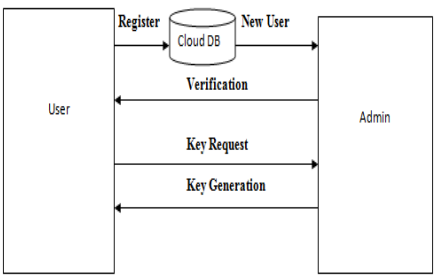


Fig A:DFD level 0

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 2, February 2016

DFD LEVEL 1(Redundant File Checking and uploading)

I. Dual-Level Source-Based (DLSB) De-duplication for large files system, the user firstly sends an encrypted file identifier to the server for file redundancy checking.
II. If the file to-be-stored is not exist in cloud storage then the user uploads the encrypted file.
III. Otherwise, the user uploads the identifiers/tag of all the encrypted file blocks to the server for block-level de-duplication

Then, checking and the user uploads encrypted data blocks which are not stored in the server.

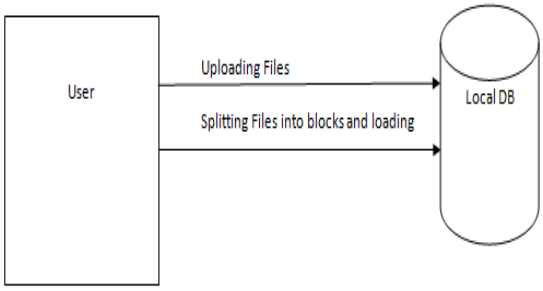


Fig B:DFD level 1

DFD LEVEL 2(File level de-duplication)

i. While redundancy checking if the file already exist then File-level de-duplication will be perform in the storage.

ii. In which the data redundancy is exploited on the file level and those redundant data is replaced with a pointer to the unique data copy.

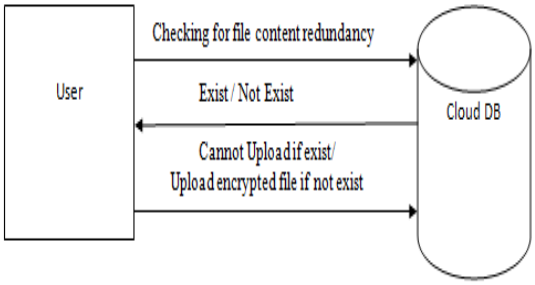


Fig C: DFD level 2

DFD LEVEL 3 (Block level de-duplication)

i. In which each file is divided into blocks, and the sever exploits data redundancy at the block level.
ii. While redundancy checking if the block already exist , the data redundancy is exploited on the block level and those redundant block is replaced with a pointer to the unique data copy. hence performs a more fine-grained de-duplication also it can eliminate chunks of data smaller than a file.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 2, February 2016

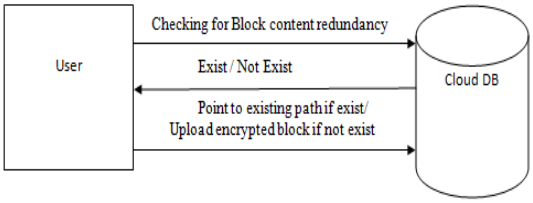


Fig D :DFD level 3

OVERALL ARCHITECTURE DIAGRAM

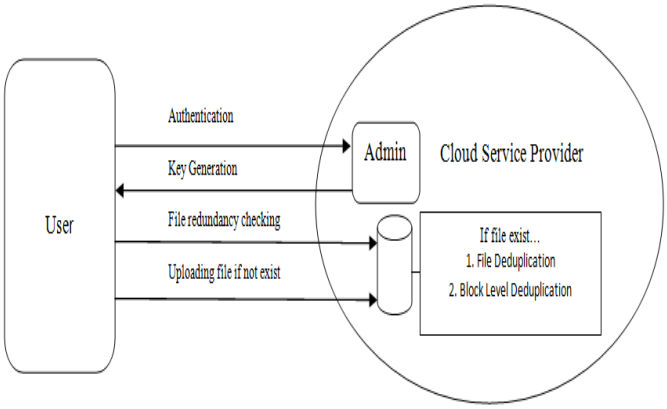


Fig E: describes about the overall functionality of the system

IX. CONCLUSION

In this work, we have proposed an new idea of block-level encryption and de-duplication technique where we could eliminate the duplicate copies in an efficient and secure manner .Block-level de-duplication, in which each file is divided into blocks, and the sever exploits data redundancy at the block level and hence performs a more fine-grained de-duplication also it can eliminate chunks of data smaller than a file. It is worth noting that for block-level de-duplication, the block size can be either fixed or variable in practice. In this work, we focus on the block-level de-duplication with fixed block size.

Fortunately, file-level de-duplication and block-level de-duplication are not incompatible with each other. In this paper, we present a technique that can achieve both of them (i.e., dual-level de-duplication).Redundancy checking through Source-based de-duplication, unlike target-based de-duplication, the user first sends an identifier/tag of the data before uploading the data to the server for redundancy checking and thus duplicated data would not be sent over the network, if the file already exists then it securely de-duplicate the File level and block level of encrypted data.

REFERENCES

[1] J. Gantz and D. Reinsel. (2012). *The Digital Universe in 2020: Big Data,Bigger Digital Shadows, and Biggest Growth in the Far East*. [Online].Available: http://www.emc.com/collateral/analyst-reports/idc-the-digitaluniverse-in-2020.pdf
[2] M. Bellare, S. Keelveedhi, and T. Ristenpart, "Message-locked encryption and secure de-duplication," in *Proc. EUROCRYPT*, 2013, pp. 296–3
[3] *The Pros and Cons of File-Level Vs. Block-Level Data De-duplication Technology*. [Online]. Available: http://searchdatabackup.techtarget.com/tip/The-pros-and-cons-of-file-level-vs-block-level-data-deduplicationtechnology,accessed Jan. 2, 2015.
[4] M. W. Storer, K. M. Greenan, D. D. E. Long, and E. L. Miller, "Secure data de-duplication," in *Proc. 4th ACM Int. Workshop Storage Secur. Survivability (StorageSS)*, 2008, pp. 1–10.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 2, February 2016

[5] J. R. Douceur, A. Adya, W. J. Bolosky, P. Simon, and M. Theimer, "Reclaiming space from duplicate files in a serverless distributed file system," in *Proc. 22nd ICDCS*, 2002, pp. 617–624.
[6] D. X. Song, D. Wagner, and A. Perrig, "Practical techniques for searches on encrypted data," in *Proc. IEEE Symp. Secur. Privacy (S&P)*, May 2000, pp. 44–55.
[7] D. Harnik, B. Pinkas, and A. Shulman-Peleg, "Side channels in cloud services: Deduplication in cloud storage," *IEEE Security Privacy*, vol. 8, no. 6, pp. 40–47, Nov./Dec. 2010.
[8] S. Halevi, D. Harnik, B. Pinkas, and A. Shulman-Peleg, "Proofs of ownership in remote storage systems," in *Proc. 18th ACM Conf. Comput.Commun. Secur.*, 2011, pp. 491–500.
[9] D. Boneh, G. Di Crescenzo, R. Ostrovsky, and G. Persiano, "Public key encryption with keyword search," in *Proc. EUROCRYPT*, 2004, pp. 506–522.