Deduplication on Encrypted Data with Secure Cloud Storage

Vishakha Mohite, Mahima Tanwar, Sushmita Kirtane, Manisha Thirwani.
Guide Name : Prof. M. A. Rane
Bharati Vidyapeeth's College of Engineering for women

Abstract:

Cloud offers the provision for storing huge amount of data of clients, which would otherwise occupy a large amount of disk space in client's machine. The previous schemes of deduplication cannot work on data which is in encrypted format. The current deduplication scheme on encrypted data suffers from security weakness. Deduplication is necessary to avoid duplicate copies of users on cloud. Duplicate data on the cloud wastes a lot of network resources, leads to consumption of energy and complication in data management. So we propose a scheme to deduplicate the encrypted data present in the cloud. Data ownership challenge and verification will also be provided and access will be granted to the authorized user through proxy re-encryption. In this project we are focusing mainly on the data saving on cloud by saving our storage cost and energy.

Keywords:

Deduplication, encrypted data, data ownership challenge, proxy re-encryption.

Introduction

Cloud computing is the new and exciting thing on the internet all over the world. Various industries have already started throwing their workload over the cloud. The most important services provided by cloud are storage and data processing.

Cloud storage provides the users/clients an opportunity to store their data on the cloud and get rid of the expensive headache of maintaining the data warehouse of own. This also means that the additional cost which the clients had been spending from years and years on the disks/machines will be saved.

Another most important benefit of the cloud storage is that the 3rd party, also referred as CSP (cloud service provider) who is providing the space on cloud will whole and solely be responsible for the management and the security of the data.

The existing schemes available in the market either fail to store the data securely on the cloud or they fail to remove the duplicate data on the cloud. The reason behind this is that once the data is encrypted considering the security aspect it becomes impossible with the existing schemes to remove the duplicate data on the cloud. And if in order to not keep/store the duplicate copies on the cloud, it than requires unencrypted data thus compromising on the security.

This proposed paper eventually solves both the problems i.e. it stores the encrypted data on the cloud thus no concerns with the security of data and at the same time it also removes the duplicate data from the cloud i.e. saves more space on the cloud.

This is implemented/proposed with help of data ownership challenge and Proxy re-encryption. As the name itself suggest, the data is first encrypted using symmetric encryption and send to CSP for storing it on the cloud. Here CSP will throw a ownership challenge to the user, if the user proves the ownership to the data than the CSP will not store that data again on the cloud instead it will give access to the existing copy of data to the user by proxy re-encryption as the data ownership challenge was positive. On the other hand if the CSP finds that this data is not present on the cloud it gets considered as the original/1st time upload request and hence the data will be stored on the cloud.

Problem Definition:

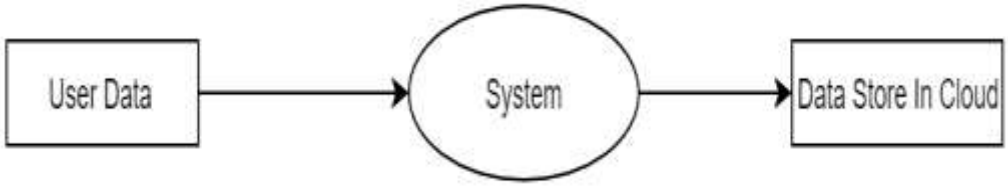
To deduplicate encrypted data at CSP by applying PRE to issue keys to different authorized data holders based on data ownership challenge. It is applicable in scenarios where data holders are not available for deduplication control.

Objectives

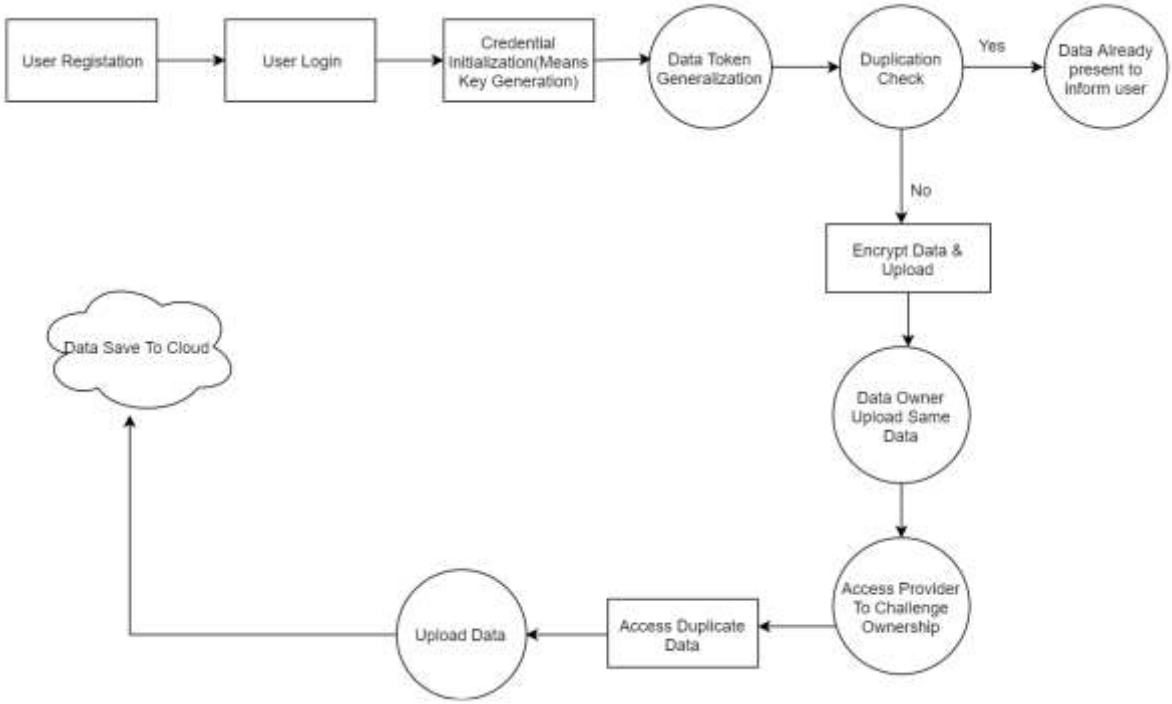
The main objective of the project is to deduplicate the data using various encryption standards, and to save the data in an encrypted format and give access to every user without leaking the main private key.

System design:

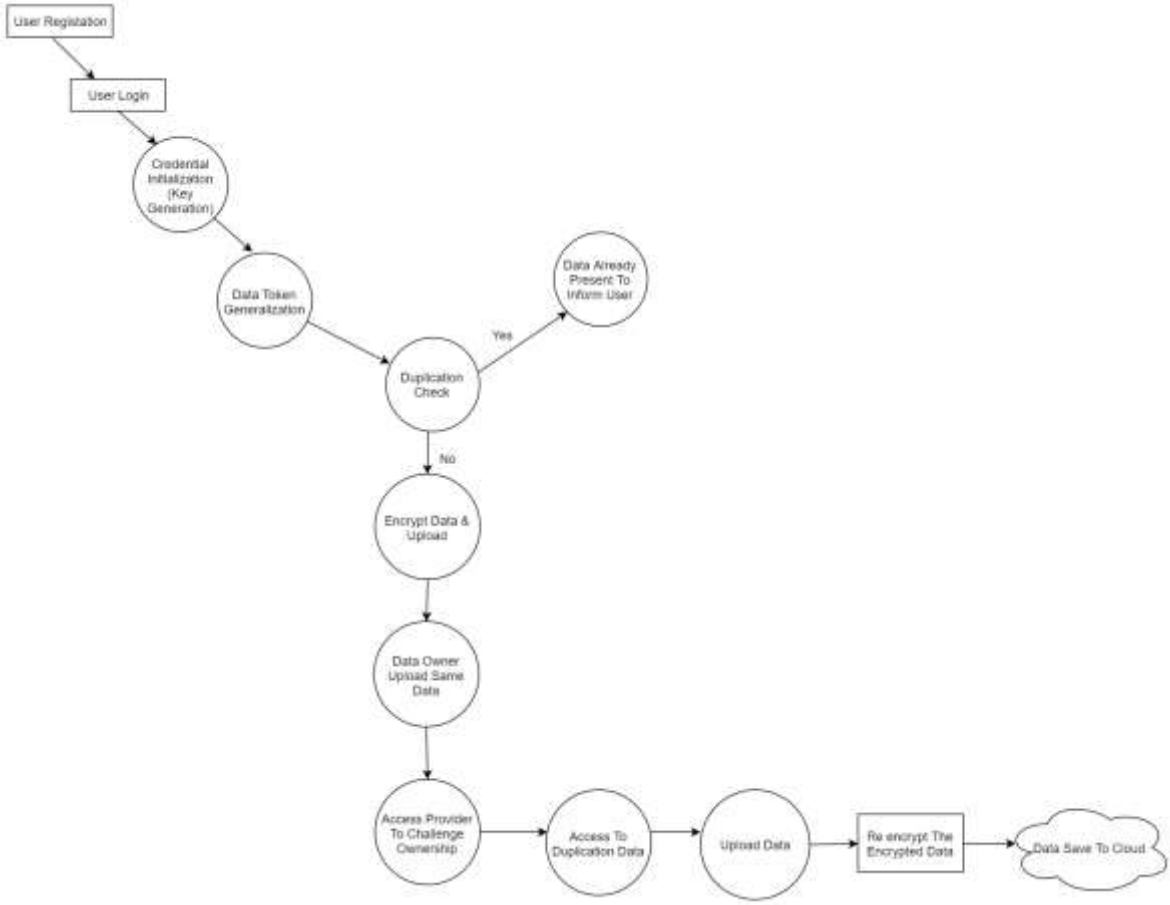
Data Flow Diagram 0:



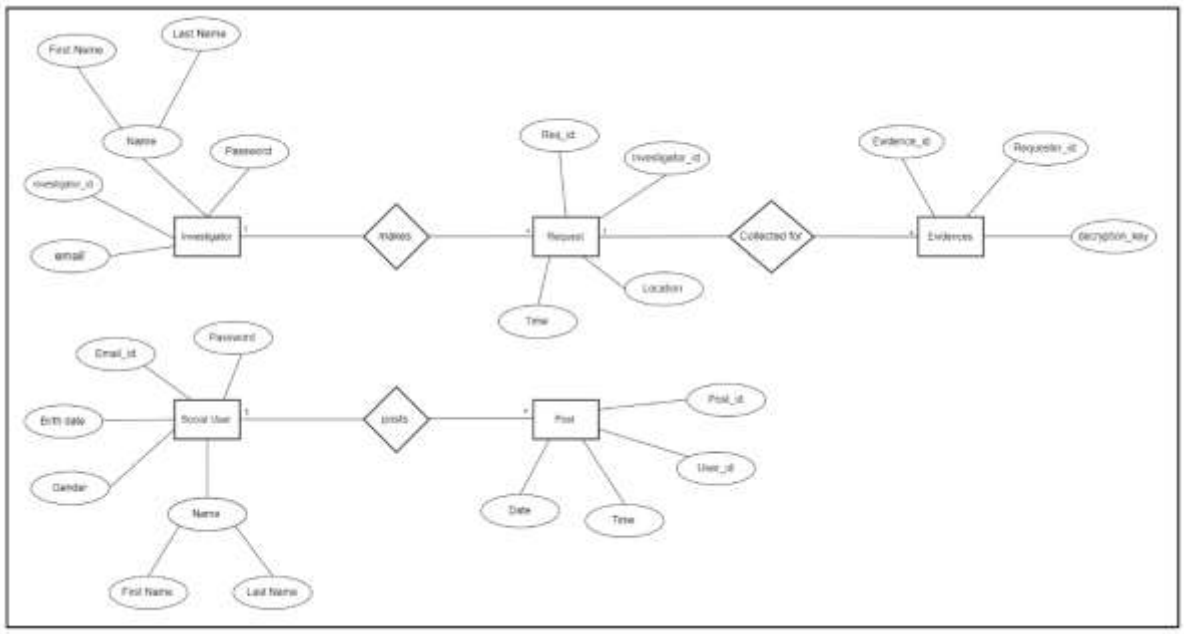
Data Flow Diagram 1:



Data Flow diagram 2:



ER Diagram:



System Architecture

In the proposed system we are going to develop cloud based system where our data is saved on cloud. As there is privacy and authentication issue there is need to save our data securely. For that we are using different algorithms of cryptography. Here our system helps us to deduplicate our data so that the space on cloud can be saved. And our aim is to reduce time complexity of our project. In the following diagram there is one user who uploads the file on the cloud. AP is nothing but the admin which keeps a look on the users files. CSP is cloud storage provider which allows you to store the data on the cloud. Deduplication is checked on encrypted data. The encrypted data is saved on cloud. And user will get the file by performing decryption.

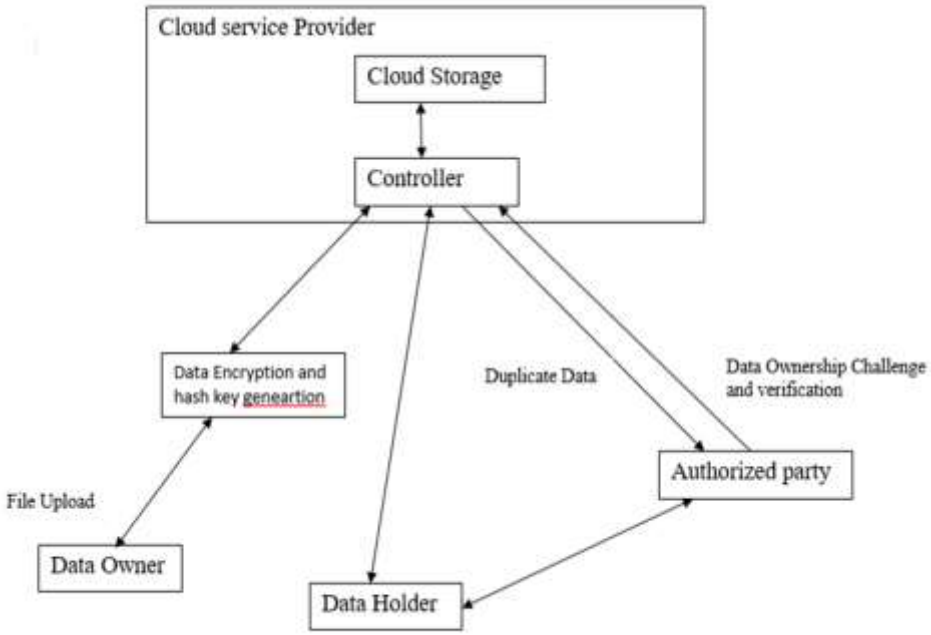


Fig 1: System Overview

Related Works

1) **DupLESS Server-Aided Encryption for Deduplicated Storage.**

Year: 2013

Author Name:

- Mihir Bellare
- Sriram Keelveedhi
- Thomas Ristenpart.

Description: This paper implement a new system called DupLESS (Duplicateless Encryption for Simple Storage) that provides a more secure, easily-deployed solution for encryption that supports deduplication. In DupLESS, a group of affiliated clients (e.g., company employees) encrypt their data with the aid of a key server (KS) that is separate from the SS.

Limitations: This system only for security and unpredictable data may be a limitation for, and threat to, user privacy this paper shows unpredictable data. and this is very difficult to access data.

2) **ClouDedup: Secure Deduplication with Encrypted Data for Cloud Storage**

Year: 2013

Author Name:

- Pasquale Puzio
- Refik Molva
- Melek Onen
- Sergio Loureiro.

Description: The new ClouDedup system in order to implement the key management for each block together with the actual deduplication operation. This system additional HSM can be implemented by taking advantage of Amazon CloudHSM which provides secure,durable, reliable, replicable and tamper-resistant key storage.

Limitations: This system finding possible optimizations in terms of bandwidth, storage space and computation.

3) **A Policy-based De-duplication Mechanism for Securing Cloud Storage**

Year: 2015

Author Name :

- Zhen-Yu Wang1
- Yang Lu1
- Guo-Zi Sun

Description: The new policy-based deduplication proxy scheme using the security proxy and random storage , which separate storage services and security services to ensure the security of user data and improve the system efficiency at the same time.

Limitations: This paper we not consider duplicated data management (e.g., deletion and owner management) and did not evaluate scheme performance.

4)Reducing impact of data fragmentation caused by in-line deduplication

Year**:** 2012

Author Name:

- Michal Kaczmarczyk
- Marcin Barczynski
- Wojciech Kilian
- Cezary Dubnicki

Description: This paper focused on inter-version duplication and proposed Context-Based Rewriting (CBR) to improve the restore performance for latest backups by shifting fragmentation to older backups.

Limitations: The full backup of only one filesystem is saved every week to a system with backward pointing deduplication.

5) DeDu: Building a Deduplication Storage System over Cloud Computing.

Year: 2011

Author Name**:**

- Zhe SUN
- Jun SHEN
- Jianming YONG.

Description: This paper presents a deduplication storage system over cloud computing. Our deduplication storage system consists of two major components, a front-end deduplication application and Hadoop Distributed File System.

Limitations**:** We developed efficient deduplication system, but it cannot handle encrypted data.

6) A Verifiable Data Deduplication Scheme in Cloud Computing.

Year: 2014

Author Name:

- Zhaocong Wen
- Jinman Luo
- Huajun Chen
- Jiaxiao Meng
- Xuan Li‡ and Jin Li.

Description: This system presents image deduplication scheme adopts two servers to achieve Verifiability of deduplication.

Limitations: This system is not flexible to support data access control by data holders, especially for data revocation process, since it is impossible for data holders to generate the same new key for data re-encryption.

7) Survey and Classification of Storage Deduplication Systems.

Year: 2014

Author Name:

- JO˜ AO PAULO
- JOS´E PEREIRA.

Description: This paper is offline deduplication systems. Then provide reliability, security and privacy should be taken into considerations when designing a deduplication system.

Limitations: This paper we not consider duplicated data management (e.g., deletion and owner management) and did not evaluate scheme performance.

8)A Hybrid Cloud Approach for Secure Authorized Deduplication.

Year: 2013

Author Name **:**

- Jin Li
- Yan Kit Li
- Xiaofeng Chen
- Patrick P. C. Lee,
- Wenjing Lou

Description**:** This paper are also several implementation of convergent of different convergent encryption variants for secure deduplication. This system provide reliability, security and privacy with sound performance.

Limitations: This system cannot flexibly support data access control and revocation at the same time.

9)Efficient Hybrid Inline and Out-of-Line Deduplication for Backup Storage.

Year: 2015

Author Name:

- YAN-KIT LI
- MIN XU
- CHUN-HO NG
- PATRICK P. C. LEE

Description: This paper we design and implement *RevDedup*, an efficient hybrid inline and out-of-line deduplication system for backup storage.

Limitations: This is high performance in essential operations of deduplication backup storage systems, including backup, restore, and deletion, while maintaining high storage efficiency.

10) Improving Restore Speed for Backup Systems that Use Inline Chunk-Based Deduplication

Year: 2013

Author Name:

- Mark Lillibridge
- Kave Eshghi
- Deepavali Bhagwat

Description: This paper improves the restore performance for latest backups by shifting fragmentation to older backups.

Limitations: We developed to forfeit deduplication to reduce the chunk fragmentation by container capping. In our previous work we developed using PRE for cloud data deduplication.

Tools Used

- **Software Requirement:**
  - Operating System : windows 8 and above.

- Application Server : Tomcat5.0/6.X
- Language : Java
- Front End : HTML, JSP
- Database : MySQL

- **Hardware Requirement:**
  - Processor : Intel i3/i4/i5
  - RAM : 4 GB (min)
  - Hard Disk : 20 G/B(min)

Algorithm

- **Data Encryption Standard(DES)**: This Algorithm is used to encrypt your data so that your data will not be hacked by third party.
- **Elliptic Curve Cryptography(ECC)**: In our system, we have used ECC to generate a specific key for different files.
- **Proxy Re-encryption(PRE)**: In cryptography PRE is used to generate different keys for different users to access single file, so that key of one user will not get leaked.

.

DES:

```
Cipher(byte in[16], byte out[16], key_array round_key[Nr+1])
begin
byte state[16];
state = in;
AddRoundKey(state, round_key[0]);
for i = 1 to Nr-1 stepsize 1 do
SubBytes(state);
ShiftRows(state);
MixColumns(state);
AddRoundKey(state, round_key[i]);
end for
SubBytes(state);
ShiftRows(state);
```

AddRoundKey(state, round_key[Nr]);

end

ECC:

1. add some extra data to the end of the input

set the initial sha-1 values

for each 64-byte chunk do

extend the chunk to 320 bytes of data

perform first set of operations on chunk[i] (x20)

perform second set of operations on chunk[i] (x20)

perform third set of operations on chunk[i] (x20)

perform fourth set of operations on chunk[i] (x20)

end

return value as a key

PRE :

Input : file

Output : Decrypted file download

Step 1: Take file as a input

Step 2 : Generate key

Step 3 : get recepeint id

Step 4:send key to mail

Step 5: exit

Our Approach:

The system will work in three operating modes:

1. User :

The user is the holder and owner of the data file. The data will be uploaded by the user, this file will be saved on cloud in encrypted format.

2. Admin :

The admin is an intermediate between user and cloud. Here you can see who uploaded which file.

3. View Files :

You can see list of files here for downloading.

4. Delete Files :

Here you can delete a file.

Experiment Result:

The system will collect the files by downloading the user's file. The files will be decrypted after downloading. Authorized user will get a private key on his mail. This key will be different for different users so that main key will not get hacked.

Future scope:

A more flexible system can be developed in future which will work with the real time data. Mp3, Mp4 data and pdf files will also be considered.

Conclusion:

In the recent days the data storage with deduplication is very important topic. In the existing system the data storage on encrypted data including deduplication is very hard to achieve. So here there is need to store encrypted data on cloud with data security. In this paper we proposed a model to deal with the data storage with encryption that will deduplicate the data. Here we are going to manage the encryption of the data with ownership challenge and proxy reencryption. Our scheme can be flexibly support the data sharing among different users. Encrypted data can be securely accessed without leaking the original data and its key. And one more thing is that the encrypted data will be accessible only to the authorized data holder. Here we are going to use symmetric keys for data decryption which will be sent to the authorized users mail id. Futurwork includes optimizing our design and implementation for practical deployment and studying verifiable computation to ensure that CSP behaves as expected in deduplication management.

References:

[1] M. Bellare, S. Keelveedhi, and T. Ristenpart, "DupLESS: Server aided encryption for deduplicated storage," in Proc. 22nd USENIX Conf. Secur., 2013, pp. 179–194.

[2] Dropbox, A file-storage and sharing service. (2016). [Online]. Available: http://www.dropbox.com

[3] Google Drive. (2016). [Online]. Available: http://drive.google.com

[4] Mozy, Mozy: A File-storage and Sharing Service. (2016). [Online]. Available: http://mozy.com/

[5] 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 Proc. IEEE Int. Conf. Distrib. Comput. Syst., 2002, pp. 617–624, doi:10.1109/ICDCS.2002.1022312.

[6] G. Wallace, et al., "Characteristics of backup workloads in production systems," in Proc. USENIX Conf. File Storage Technol., 2012, pp. 1–16.

[7] Z. O. Wilcox, "Convergent encryption reconsidered," 2011.n[Online]. Available:http://www.mailarchive.com/cryptography@metzdowd.com/msg08949.html

[8] G. Ateniese, K. Fu, M. Green, and S. Hohenberger, "Improved proxy re-encryption schemes with applications to secure distributed storage," ACM Trans. Inform. Syst. Secur., vol. 9, no. 1, pp. 1– 30, 2006, doi:10.1145/1127345.1127346.

[9] Opendedup. (2016). [Online]. Available: http://opendedup.org/

[10] D. T. Meyer and W. J Bolosky, "A study of practical deduplication," ACM Trans. Storage, vol. 7, no. 4, pp. 1–20, 2012, doi:10.1145/2078861.2078864.