RESEARCH ARTICLE OPEN ACCESS

IMPLEMENTATION OF SECURE DATA DEDUPLICATION ON CLOUD WITH OTP GENERATION

Payal Banode [1], ***Sapna Meshram*** [2], ***Prof. Shweta Bhelonde*** [3]

1. *Computer engineering, S.R.P.C.E., Nagpur,India,*
2. *Computer engineering, S.R.P.C.E., Nagpur,India,*
3. *Computer engineering, SR.P.C.E.,Nagpur,India,*

Abstract:

Generally cloud computing offers a storage service, in this paper experimented the de-duplicate data in encrypted form if unauthorized person occurred while accessing data. However we need to secure our data, we proposed a scheme to deduplicate Encrypted data stored on cloud and download the encrypted file in decryption form using OTP on email. It integrates to cloud data reduplication with access control. It redundant of data with data security. We enhanced its performance based on extensive analysis and comparison of various structured file. The result does not shows the deduplication data and save the cloud storage space. We provide the email OTP system to authenticate the valid register user. Which make cloud storage more secure and efficient

Keyword: ***Dropbox.***

I. INTRODUCTION

In today's world everyone need to store the data with security. If the data is not secure, then the problem can be create in that time and it is so harmful for that person. A structured file is consist of metadata and object like, text and image (.zip, .rar) .Any type of file we can store in it [5] [13]. It provides flexibility and the benefits of deduplication storage facility. SAFE is a fast processing time as file level or fixed size block deduplication using structure based with save the space of cloud. It activate the highly providing email security and the email of one time password service and this service gives data of authenticate person[17][21]. It is using for official business work that save the office data with individual id and the admin is use one email id for the security then admin provide OTP and authenticate person easily access user data. This project aimed is to save the large amount of space of cloud storage system with reduce deduplicate data and the security service with using dropbox application. [7][15]

LITERETURE SURVEY

Deduplication has proved to achieve high cost savings, e.g., reducing up to 90-95 percent storage needs for backup applications [9] and up to 68 percent in typical file systems [10]. Observably, the savings, which can be passed back directly or indirectly to cloud users, are important to the economics of Cloud Company. How to deal with encrypted data storage with deduplication in an efficient way is a realistic issue. However, current work deduplication solutions cannot handle encrypted data. Existing solutions for deduplication suffer from brute-force attacks [7], [11], [12], [13], and [14].

We propose a novel Structure-Aware File and Email deduplication (SAFE) scheme that achieves both efficient and effective data deduplication at a source site for cloud-based storage [7]

Reconciling deduplication and client-side encryption is an active research topic [1]. Message-Locked Encryption (MLE) intends to solve this problem [5]. The most prominent manifestation of MLE is Convergent Encryption (CE), introduced by Douceur et al. [6] and

others [7], [11], [12]. CE was used within a wide variety of commercial and research storage service systems. Letting M be a file's data, a client first computes a key K H(M) by applying a cryptographic hash function H to M, and then computes cipher text C K;MÞ via a deterministic symmetric encryption scheme. A next client B encrypting the same file M will generate the same C, enabling deduplication. However, CE is subject to an inherent security restriction, namely, susceptibility to offline brute-force dictionary attacks [13], [14].

However, it does not compact with data sharing after deduplication among dissimilar users. Clouded up [16] also aims to cope with the inherent security exposures of CE, but it cannot solve the issue caused by data deletion. A data holder that removes the data from the cloud can still access the same data since it still knows the data encryption key if the data is not completely removed from the cloud.

Bellaire et al. [1] proposed duplicate that provides secure deduplicated storage to resist brute-force attacks. In Dup- LESS, a group of affiliated clients (e.g., company employees) encrypt their data with the aid of a Key Server (KS) that is separate from a Storage Service (SS), but do not leak any information about their data to it. As long as the KS remains inaccessible to attackers, high security can be ensured. Obviously, Dup- LESS cannot control data access of other data users in a flexible way. Alternatively, a policy-based deduplication proxy scheme [15]

SYSTEM OVERVIEW

Diagram:

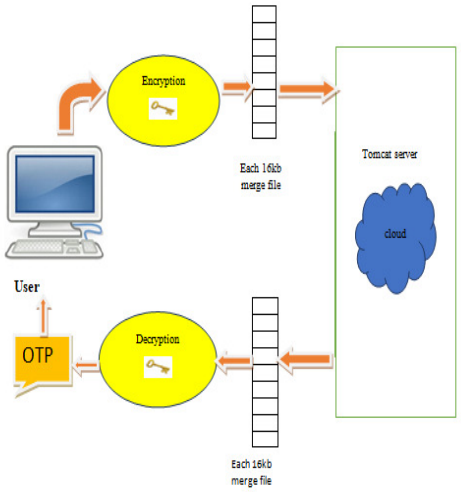


Fig 1.1 A merge files sending data with OTP to the cloud

A shown in fig 1.1 Each file object chunk is hashed into a unique object and then merging with cloud storage system. Depend upon size then it will handle the distributed data and time. A cloud data capacity depends on the data size. When the user download the data then they will received decrypt data with OTP on their email_id.

When user create new registration then the admin approval the authenticate person. After that procedure user can easily login and upload, download the data with the security. Each 16kb merge file compare with the unique and object file and separated the blogs of each block. Large amount of data stored to the cloud with the OTP concept generation. We are using tomcat server because it is an open source for the implementation of java servlet, the server powers number of large scale,

critical web based application. The email OTP authentication method send to the email address of the user with one time password user can authenticate their stored data from the server. This authentication play the role of security a perform automatically and it's not possible to removed it.

Flowchart:

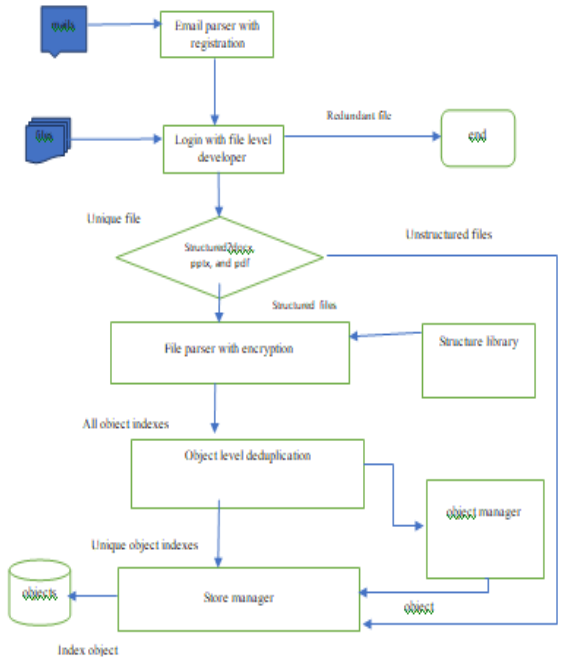


Fig 1.2 Flowchart

EMAIL PARSER:

An Email parser bridge the space between the emails you received and documentation files, database or any other tools or file format , it can parse the text from the email body or the attachments, the confirmation registration to allow the approval from admin for authorization. Registration details, user_id, password and all other information include the email parser of the connect with dropbox application details.

LOGIN WITH FILE LEVEL DEVELOPER

File level deduplication work at the file level by eliminating redundant file, object level deduplication work at a object level deduplication. Which may be fixed size block or variable size block by eliminating duplicate block. The advantage of file level deduplication is that its takes less resources those may be deployed over the larger amount of physical storage.

STORE MANAGER

When you add and object stored or upgrade in a cloud storage, you must configured the production target object stored indexes.

OBJECT MANAGER

Object manager can help you handle object as they progress through your project life cycle using object manager, you can copy object with in a project or across project. Object manager **can** copy application, schema and configuration objects when you copy object across the project with object manager, if an object with same id of source object exist anywhere in the destination project a conflict occurs. Object manager helps you resolve to conflict.

METHODOLOGIES/PROPOSED SYSTEM

In this method, the unique object compare with other object and the similar data doesn't exist. It is main thing because of the processing speed become fast. And the save the storage space. For the security purpose we take the AES and DES algorithm it generates encrypted key.

AES is cryptographic cipher that is responsible for large amount of data files security that makes our important data more secure by converting user data in non-readable form. There are so many common uses of AES algorithm which have common compression tools like WinZip, RAR allow you to compress and the decompress data files in order to improve storage space on cloud and nearly all of those used AES to ensure file security.

The features of Advanced Encryption Standard are as follows –

1. Symmetric key symmetric block cipher
2. 128-bit data, 128/192/256-bit keys
3. Stronger and faster than Triple-DES
4. Provide full specification and design details
5. Software implementable in C and Java

Operation of AES:

The block of encryption process is just a sequence of 128 bites. So fist we covert these 128 bits into 16 bytes, we say that we convert these 128 bits into bytes but in reality it is almost certainly stored this way already. This array. Include the four types of operations which are known as sub byte, shift Row, MixColoumn, XorRoundKey.

The DES (data encryption standard) is symmetric key algorithm for encryption of electronic data files. DES based on feistel construction while the one way function is used and one way function you don't need to reversed it at all to "Decrypt".

In DES plaintext are broken into blocks of the length which is 64 bits. Encryption is done block wise. The message block is first going through an initial permutation IP. Then the block is divided into two parts L0. Where L0 is the left part of 32 bits block of data and R0 is the right part of the 32 bit block of data.

Round I have input L_{i-1}, Ri_{-1} and output L_i, Ri

$L_i = R_{i-1}$, $Ri = L_{i-1} \oplus f(R_{i-1}, Ki)$

And K_i is the sub key for the 'it's where $1 \leq i \leq 16$

$L_1 = R_0$, $R_1 = L_0 \oplus f(R_0, K1)$

$L_2 = R_1$, $R_2 = L_1 \oplus f(R_1, K2)$

$L_3 = R_2$, $R_3 = L_2 \oplus f(R_2, K3)$

....

....

....

$L_{16} = R_{15}$, $R_{16} = L_{15} \oplus f(R_{15}, K16$

Later round 16 the L_{16} and R_{16} are exchanged, so that decryption algorithm has the same structure as the encryption algorithm.

Finally, the block is go through the opposite the permutation IP^{-1} and then we get the output.

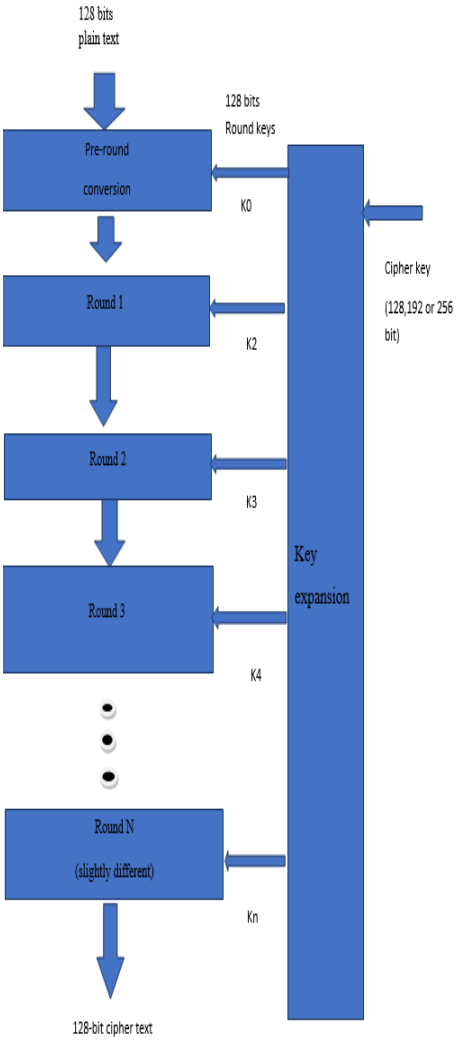
- Opinion: In encryption, we have

$L_i = R_{i-1}, R_i = R_i = L_{i-1} \oplus f(R_{i-1}, K_i)$

- and K_i is the subkey for the 'i'through.Hence

$R_{i-1} = L_i, L_{i-1} = R_i \oplus f(L_i, K_i)$ for each 'i'

- because of swap operation after the 16th round encryption,the output of the encryption is $IP^{-1}(R_{16}, L_{16})$



R	Key size
10	128
12	192
14	256
16	320
18	384

Relationship between number of round(R) and cipher key sizes

Fig 2.1 AES operation

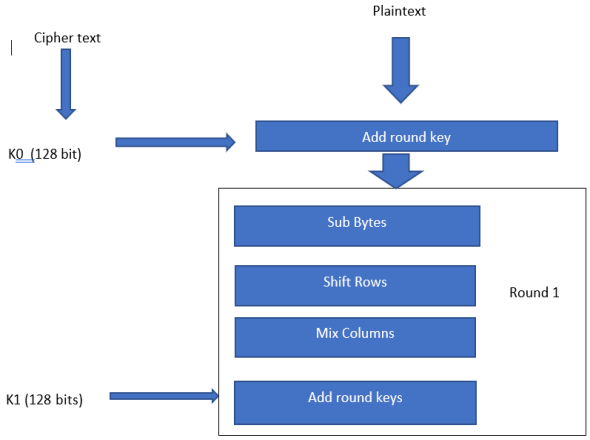


Fig 2.2 Round1 process expansion

ALGO ---PSEUDO CODE

We have c-bit message as input, and that we have to find its message digest. Here c is a random nonnegative integer; c may be zero, it need not be a multiple of eight, and it may be randomly large. It visualize the bits of the message as follows:

n_0 n_1 ... n_{c-1}

The following five steps are perform to calculate the message digest of the message.

Step1: Add padding bits

Step2: Add length

Step3: Initialize MD buffer

Step 4: Processing 16 words block

Step 5: Output

Hx (Z1) = Hx (Z2) ⟺ Y1=Y2

Hy (Z2) = Hy (Z2) ⟺ Y1=Y2

A (K1, E (K2, Y)) Y⟺ K1=K2

With this method doesn't waste a space when chunk occur multiple time around of the blocks of unique object.

TABLE COMPARISION OF SIZE

#	size	Hash	ID	Length	Start
0.doc	28KB	Ac062ddf-afoa	31	4556	0
1.jpg	20.1MB	6c1fba89-cdd7	51	1978	4556
2.txt	55 Byte	4cac3c92-71a4	13	2843	6535
3.pdf	272KB	81fda942-f479	45	3794	9379
4.video	15MB	97ff73b42-cf42	54	15751	13174
5.audio	5.3MB	971gf5rf3-ju63	76	5283	28926

table1: The hash value refers the id of data and depending on length takes the size and time.

Id = = start value + length;

It restoring the deduplicate search data with the calculate value of the allocation area. Data of particular format get break into chunks i.e. The separated object value with the blocks of unique value and index value. In the simple way we restore k-byte slices takes in allocated area and then send or received the user in a single piece.

EXPERIMEDNTAL RESULT

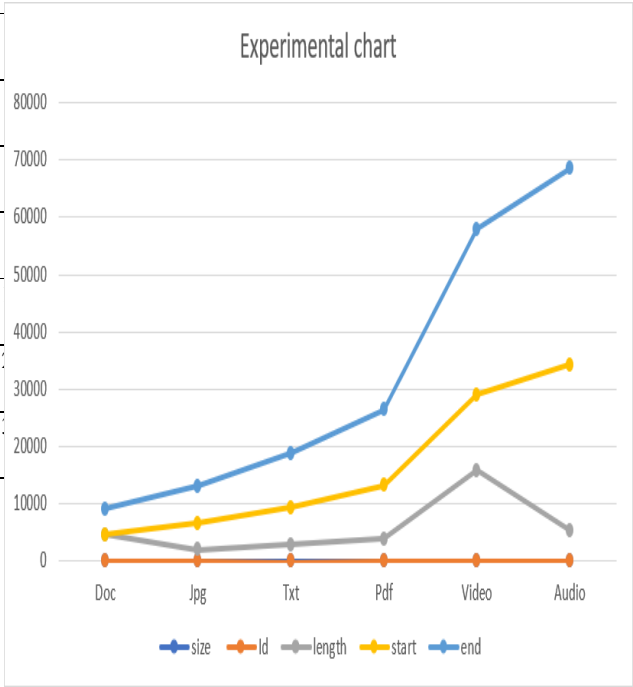


Fig 3.1 shows the results ratio of length of data size file level deduplication and block level deduplication. It shows the start and end with the flow of certainty value and it contains the hash value with the size of data. If the takes time depends on the size of data variable.

FUTURE SCOPE

Data deduplication techniques guarantee that only one unique example of data is maintained on storage media, such as mobile devices, flash or disk. Speed of flexibility to efficient for storage the data this technique i.e. data deduplication is must for efficiency, it is add more ideas in our basic future implementation

We include An OTP system which is more secure than a static password, especially a user-created password, which is typically weak. OTPs replace authentication login information or used in addition to it, to add another layer of security. In future we can add the barcode scanning and finger print by using bio-metric system.

CONCLUSION

This paper discusses the information about data deduplication for the dropbox cloud based systems. It includes the methods that are used to achieve cost effective storage and effective bandwidth usage by deduplication. However, reliability and speed are at stake. Therefore lies in identifying more effective hashing algorithms for improving the speed of storing data and security. However, data deduplication is the most crucial element for improving efficiency of the cloud system. This technique will play a major role in the cloud based services for storing backup data by both medium and large enterprises.

REFRENCES:

[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] 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.

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

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

[5] 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.

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

[7] 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.

[8] J. Pettitt, "Hash of plaintext as key?" (2016). [Online]. Available: http://cypherpunks.venona.com/date/1996/02/msg02013.html

[9] The Freenet Project, Freenet. (2016). [Online]. Available: https:// freenetproject.org/

[10] M. Bellare, S. Keelveedhi, and T. Ristenpart, "Message-locked encryption and secure deduplication," in Proc. Cryptology—EUROCRYPT, 2013, pp. 296–312, doi:10.1007/978-3-642-38348-9_18.

[11] D. Perttula, B. Warner, and Z. Wilcox-O'Hearn, "Attacks on convergent encryption." (2016). [Online]. Available: http://bit.ly/ yQxyvl

[12] C. Y. Liu, X. J. Liu, and L. Wan, "Policy-based deduplication in secure cloud storage," in Proc. Trustworthy Comput. Serv., 2013, pp. 250–262, doi: 10.1007/978-3-642-35795-4_32.

[13] P. Puzio, R. Molva, M. Onen, and S. Loureiro, "ClouDedup: Secure deduplication with encrypted data for cloud storage," in Proc. IEEE Int. Cof. Cloud Compute. Technol. Sci., 2013, pp. 363–370, doi:10.1109/CloudCom.2013.54.

[14] Z. Sun, J. Shen, and J. M. Yong, "DeDu: Building a deduplication storage system over cloud computing," inProc. IEEE Int. Conf. Comput. Supported Cooperative Work Des., 2011, pp. 348–355, doi:10.1109/CSCWD.2011.5960097.

[15] Z. C. Wen, J. M. Luo, H. J. Chen, J. X. Meng, X. Li, and J. Li, "A verifiable data deduplication scheme in cloud computing," in Proc. Int. Conf. Intell. Netw. Collaborative Syst., 2014, pp. 85–90, doi:10.1109/INCoS.2014.111.

[16] J. Li, Y. K. Li, X. F. Chen, P. P. C. Lee, and W. J. Lou, "A hybrid cloud approach for secure authorized deduplication," IEEE Trans. Parallel Distrib. Syst., vol. 26, no. 5, pp. 1206–1216, May 2015, doi:10.1109/TPDS.2014.2318320.

[17] P. Meye, P. Raipin, F. Tronel, and E. Anceaume, "A secure twophase data deduplication scheme," in Proc. HPCC/CSS/ICESS, 2014, pp. 802–809, doi:10.1109/HPCC.2014.134.

[18] J. Paulo and J. Pereira, "A survey and classification of storage deduplication systems," ACM Comput. Surveys, vol. 47, no. 1, pp. 1–30, 2014, doi:10.1109/HPCC.2014.134.

[19] Y.-K. Li, M. Xu, C.-H. Ng, and P. P. C. Lee, "Efficient hybrid inline and out-of-line deduplication for backup storage," ACM Trans. Storage, vol. 11, no. 1, pp. 2:1-2:21, 2014, doi:10.1145/2641572.

[20] M. Fu, et al., "Accelerating restore and garbage collection in deduplication- based backup systems via exploiting historical information," in Proc. USENIX Annu. Tech. Conf., 2014, pp. 181–192.

[21] M. Kaczmarczyk, M. Barczynski, W. Kilian, and C. Dubnicki, "Reducing impact of data fragmentation caused by in-line deduplication," in Proc. 5th Annu. Int. Syst. Storage Conf., 2012, pp. 15:1–15:12, doi:10.1145/2367589.2367600.

[22] M. Lillibridge, K. Eshghi, and D. Bhagwat, "Improving restore speed for backup systems that use inline chunk-based deduplication," in Proc.USENIX Conf. File Storage Technol., 2013, pp. 183–198. [19] Fahlman, Scott E. and Lebiere, Christian: The Cascade-Correlation Learning Architecture, Neural Information Processing Systems 2, page 524-532, 1990