ISSN(Online) : 2319-8753
ISSN (Print) : 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

*(An ISO 3297: 2007 Certified Organization)*

**Vol. 5, Issue 7, July 2016**

# Secured Distributed De-duplication System on Cloud

Deepali Choudhari [1], Rashmi Deshpande [2]

P.G. Student, Department of Information Technology, Sidhhant College of Engineering, Pune, India[1]

Professor, Department of Information Technology, Sidhhant College of Engineering, Pune, India[2]

**ABSTRACT**: Cloud Computing is getting more and more popularity day by day due to its provided good services. But ever increasing volume of data on cloud storage is going to be a challenge. Data management and data security are the main problems in cloud computing. Encryption is used for the purpose of data security and data privacy of the outsourced data. For the data management and maintenance de-duplication concept is used. De-duplication is a technique of keeping only one unique instance of data copy by detecting identical data copies and eliminating those so that it could improve storage utilization, system performance of storage system. We present the appropriated distributed storage servers into de-duplication frameworks to give better adaptation to internal failure. For the classification of tags which are derived after encryption of the file to distribute across multiple cloud servers, k nearest neighbour algorithm is used along with simple password encryption key exchange protocol for the safe and secure communication among the end users. This protocol provides more security than previous one. It prevents man in middle attack. Data confidentiality and reliability are achieved.

**KEYWORDS**: de-duplication, convergent encryption, cryptographic hash, cloud storage.

## I. INTRODUCTION

Distributed computing gives unfathomable virtualized arrangement of activity to customer as organizations over the whole web while hiding the stage and executing unpretentious components. Today by expanding the volume of data in distributed storage made issue for copy information as information is gathered from heterogeneous sources. Distributed storage is getting famous more as it is ease and on-interest utilization of substantial stockpiling. To make information administration versatile in distributed computing, the de-duplication idea is utilized. As per the examination report of IDC, the volume of information on the planet is relied upon to achieve 40 trillion gigabytes in 2020 [1]. Today's business distributed storage administrations, for example, Dropbox, Google Drive and Mozy, have been applying de-duplication to spare the system data transfer capacity and the capacity cost with customer side de-duplication. De-duplication implies copy information is wiped out, a pointer is made to reference an information that is moved down. So as to have a protected stockpiling of de-copied information over a distributed computing we utilize the encryption/unscrambling procedure.

De-duplication can occur at 1) File level, in this it recognizes repetitive information inside the documents or 2) Block level, in this it identifies excess information over the records and evacuates those information of indistinguishable information records. In document level de-duplication, framework first check inside the records away for copies and if there is no copies in record then square level de-duplication is finished. It works on premise of sub-record level in which check copies or indistinguishable information over the documents. Record is being broken into pieces or lumps or sections which are then contrasted with already put away information.

Before transfer the information to the server client need to encode the information and afterward transfer their information on to the cloud. Encryption is the procedure of changing over a plaintext into a cipher text [2]. Customary encryption is opposing to de-duplication as it requires diverse clients to encode their information with their own keys. United encryption is for the most part utilized for the de-duplication process [3]. To keep the classification of touchy information while supporting the de-duplication, to scramble the information before outsourcing focalized encryption procedure has been proposed. Joined encryption is surely understood idea. It checks the copy by label estimation of

ISSN(Online) : 2319-8753
ISSN (Print) : 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

*(An ISO 3297: 2007 Certified Organization)*

**Vol. 5, Issue 7, July 2016**

information document delivered by clients. A Proof of proprietorship convention is executed to recognize information duplicate proprietor that is which client possesses the information document [7]. A number of de-duplication systems have been proposed based on various de-duplication strategies are explained in survey.

## II. PRELIMINARIES

Following are the some basic methods used in de-duplication:

### *A. Symmetric Encryption*

Symmetric encryption uses a common secret key *k* to encrypt and decrypt information. A symmetric encryption scheme made up of three primary functions.

*1)* KeyGen SE (1λ)→ : k is the key generation algorithm that generates k using security parameter 1λ;

*2)* Enc SE (k, M)→ C: is the symmetric encryption algorithm that takes the secret k, and message M and then outputs the cipher text C, and Dec SE (k, C) → *M:* is the symmetric decryption algorithm that takes the secret *k* and cipher text *C* and then outputs the original message *M*.

Each user encrypts the data with their own encryption algorithm. In this, identical data copies that produce the different cipher text, this make the de-duplication process impossible.

### *B. Convergent Encryption*

Convergent encryption encrypts a data copy with a convergent key, which is derived by cryptographic hash value of the content of the data itself [3]. In addition user derives a tag for the data copy, such that tag will be used to detect duplicates. After key generation and data encryption, users retain the keys and send the tag to the server side to check the identical copy. It is assumed that if two copies are identical then their corresponding tag values are also identical. Since encryption is deterministic, identical data copies will generate the same convergent key and same cipher text. This allows the cloud to perform de-duplication on cipher texts can only be decrypted by corresponding data owners with theirs convergent keys.

### *C. Proof of ownership*

Proof of ownership allows ownership of data copies to the server side. When tag value is same in storage then it should be proven that which user, owner owns that file.

## III. RELATED WORK

In [3], focalized encryption is utilized by means of A cryptographic primitive, Message-Locked Encryption (MLE), is a symmetric encryption plan where the key under which encryption and deciphering are performed is itself drawn from the message.

To ensure information classification, [4] chipped away at it by changing the predictable message into unpredictable message. In their framework, another outsider called the key server was acquainted with create the record tag for the copy check. It gives secure de-copied capacity opposing savage power assaults, and acknowledge it in a framework called DupLESS (Duplicateless Encryption for Simple Storage).

A novel encryption conspires that gave in [5] that bolster differential security for prominent and disagreeable information. For prominent information that are not especially delicate, the customary traditional encryption is performed. PoW, in light of the joint utilization of concurrent encryption and the Merkle-based Tree, for enhancing information security in distributed storage frameworks, giving element sharing amongst clients and guaranteeing productive information de-duplication. They proposed and actualized, on OpenStack Swift, another customer side de-duplication plan for safely putting away and sharing outsourced information by means of general society cloud. They assembled a reproduced distributed storage structure, taking into account OpenStack Storage framework (Swift). It is a cloud based capacity framework, which stores information and permits compose, read, and erase operations on them.

ISSN(Online) : 2319-8753
ISSN (Print) : 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 7, July 2016

The dependability in de-duplication has been tended to in [6] by Jin Ji et al. It demonstrates to accomplish solid key administration in de-duplication. Notwithstanding, they didn't say about the utilization of solid de-duplication for encoded records. It addresses the key-administration issue in square level de-duplication by disseminating these keys over different servers in the wake of encoding the records. Dekey is intended to effectively and dependably keep up joined keys. It is another development in which clients don't have to deal with any keys all alone yet rather safely disperse the united keys offers over various key administration separates that is KM-CSPs.
The conventional de-duplication strategies can't be straightforwardly developed and connected in disseminated and multi-server frameworks. In the event that it is utilized, it can't avoid the agreement assault dispatched by numerous servers. In [7] a document is first part and encoded into sections by utilizing the strategy of Ramp mystery sharing, rather than encryption systems. These shares are then conveyed over different free stockpiling servers. Tag, a short cryptographic hash estimation of the substance will likewise be figured and sent to every capacity server as the unique mark of the piece put away at every server. Just the information proprietor who first transfers the information is required to process and circulate such mystery offers, while every single after client who possess the same information duplicate don't have to figure and store these shares any more. To recoup information duplicates, clients must get to a base number of capacity servers through validation and get the mystery shares to recreate the information.
For the proof of ownership [8] enables users to prove their ownership of data copies to the storage server. Specifically, Proof of ownership is implemented as an interactive algorithm run by a user and a storage server. To solve the problem of using a small hash value as a proxy for the entire file, they designed a solution where a client proves to the server that it indeed has the file. We call a proof mechanism that prevents such leakage amplification a proof of ownership. Cross user client side de-duplication is focused in [9] of encrypted files in the cloud storage: confidentiality of users' sensitive files against both outside adversaries and the honest-but-curious cloud storage server in the bounded leakage model. [10] extended PoW for encrypted file, but they did not address how to minimize the key management overhead.
The secret sharing technique is utilized To protect data confidentiality [11]. A file is divided into pieces say n by using the technique of secret sharing in which data can be reconstructable using some pieces say k only but cant be reconstruct by k – 1 pieces. Such scheme is called as (k, n) threshold scheme. This scheme is based on polynomial interpolation.

## IV. IMPLEMENTATION DETAILS

### A. System Overview

Due to the popularity of cloud storage system, volume of back up data in cloud storage is increasing rapidly. This leads to use de-duplication concept which keeps only one unique instance of data copy by detecting identical data copies and eliminating those so that it could improve storage utilization, system performance of storage system. Following figure Fig.1 represents architecture of the system.
Architecture mainly includes entities:
_ User
_ Storage cloud service provider(S-CSP)
Users are entities that want to outsource their data on cloud remotely for the use of next time. On cloud there should be only unique of data copy uploaded by users, no other duplicates to save upload bandwidth.
The S-CSP is an entity that provides the outsourcing data storage service for the users who wanted to outsource their data on cloud. In de-duplication system at the timing of uploading data, the S-CSP will only store a single copy of files and retain only one data file due to which can reduce the storage cost at the server side and increase storage utilization. For fault tolerance and confidentiality of data storage, we consider a quorum of S-CSPs, each being an independent entity. The user data is distributed across multiple S-CSPs.

ISSN(Online) : 2319-8753
ISSN (Print) : 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 7, July 2016

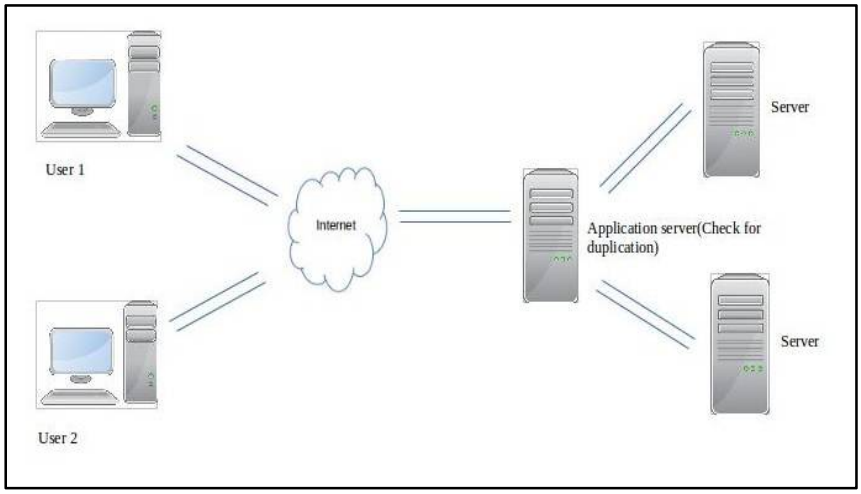


Fig. 1 System Overview

*B.* Algorithm Used

In this system, we are intended to demonstrate to plan secure de-duplication frameworks with higher dependability in distributed computing. We present the appropriated distributed storage servers into de-duplication frameworks to give better adaptation to internal failure.

The k Nearest Neighbours (k-NN) algorithm is to use a database in which the data points are separated into several separate classes to predict the classification of a new sample point. It classifies tags (derived after encryption) to distribute on server. In k-NN classification, the output is a class membership. An object is classified by a majority vote of its neighbours, with the object being assigned to the class most common among its $k$ nearest neighbours ($k$ is a positive integer, typically small). If $k = 1$, then the object is simply assigned to the class of that single nearest neighbour. For the encryption process Advances encryption standard (AES) symmetric key cryptosystem is used. It is used to protect classified information. It is considered that AES is impervious to all attacks.

Along with these algorithms SPEKE (Simple Password Encryption Key Exchange) protocol is used that is the method for password authenticated key agreement for the communication of end users. This protocol nothing but little more than a Diffie-Hellman key exchange where the Diffie-Hellman generator g is created from a hash of the password. By incorporation of the password SPEKE can prevent man in middle attack which Diffie-Hellman can not do.

*C.* Workflow of The System

The distributed de-duplication system allows both file level and block level de-duplication. At the time of uploading file, it will first check at file level de-duplication. If file is duplicate, then obviously blocks of the file are same as a result no need to go for block level de-duplication otherwise system will check at block level. Each data copy is associated with tag (cryptographic hash values of data) which is stored at cloud for the duplicate check. When users want to upload file on cloud, tags are derived from the data to check duplicates. New file is stored if and only if tags are not available on cloud otherwise pointer of original file is referred to that user. The workflow of the system is shown in figure 2.

ISSN(Online) : 2319-8753
ISSN (Print) : 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 7, July 2016

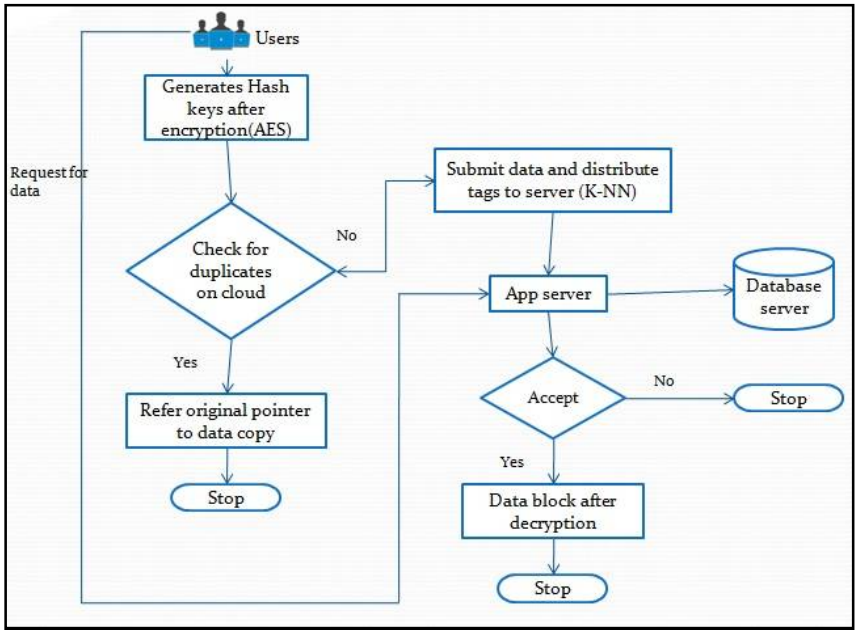


Fig. 2 Workflow of System

## V. EXPERIMENTAL RESULTS

*A.* Data Set

Distributed de-duplication system works on input data and draws conclusion whether provided data file will be stored on cloud storage or not. Input data files can be text file, doc file or pdf file. Memory size of data to be uploaded can be unlimited.

*B.* Results

De-duplication system should be reduce the storage space and upload bandwidth as much as possible. With that data security and data privacy should also be maintained as user wanted to keep their data safe. Proposed system not only focus on duplicates files but also end user security. Performance of de-duplication system with different storage system is different. It depends on parameter like data size, encryption speed, hashing function, number of servers etc. Speed performance with increasing data sizes of encryption and decryption is given below. Fig. 3 shows encoding and decoding overhead.

| Data size (kb) | Enc Time (ms) | Dec Time (ms) |
|---|---|---|
| 100 | 15 | 8 |
| 200 | 18 | 11 |
| 300 | 23 | 13 |
| 400 | 26 | 17 |
| 500 | 31 | 20 |
| 600 | 35 | 24 |

Table 1 Encryption and Decryption Time of Data

ISSN(Online) : 2319-8753
ISSN (Print) : 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

*(An ISO 3297: 2007 Certified Organization)*

**Vol. 5, Issue 7, July 2016**

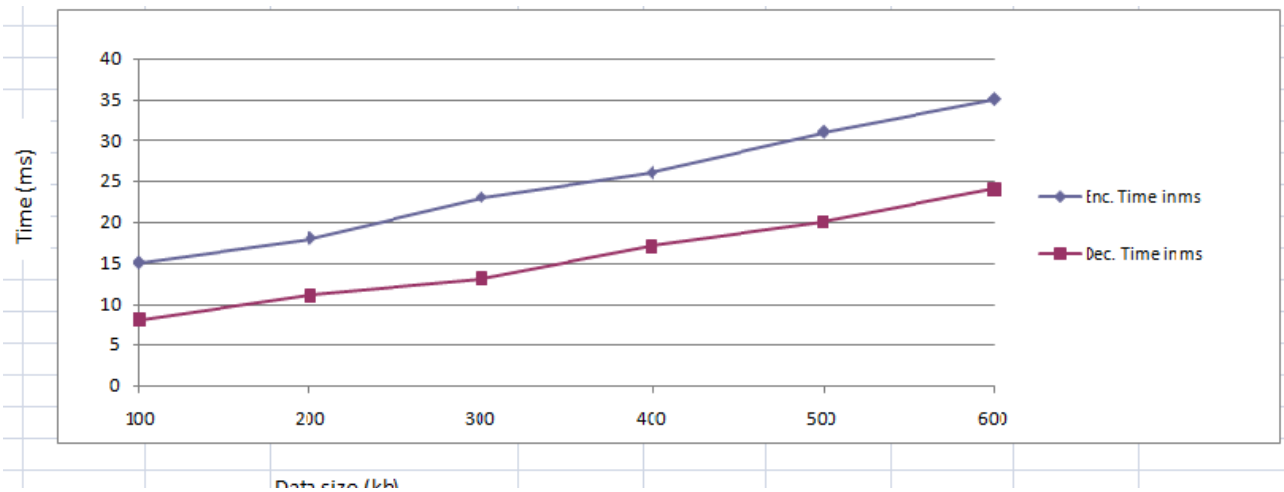


Fig. 3 Graphical Representation of Encryption and Decryption Time with incresed data size

## VI. CONCLUSION

The proposed work has outlined how to provide more security. The fundamental thought is that we can restrain the harm of stolen information on the off chance that we diminish the estimation of that stolen data to the assailant. K-NN algorithm is used to distribute the tags and SPEKE is used for secured key exchange to the end users. We can accomplish this through a "preventive" disinformation assault by SPEKE protocol. We set that protected de-duplication administrations can be actualize given extra security highlights insider assailant on De-duplication and outcast aggressor by utilizing the recognition of masquerade action. The perplexity of the assailant and the extra costs brought about to recognize genuine from fake data, and the discouragement impact which, albeit difficult to quantify, assumes a noteworthy part in averting masquerade action by danger loath aggressors. We place that the mix of these security components will give uncommon levels of security to the de-duplication.

## REFERENCES

[1] J. Gantz and D. Reinsel, "The digital universe in 2020: Big data, bigger digital shadows, and biggest growth in the far east," http://www.emc.com/collateral/analyst-reports/idcthe-digital-universe-in-2020.pdf, Dec 2012.
[2] S. Kamara and K. Lauter, ''Cryptographic Cloud Storage,'' in Proc. Financial Cryptography: Workshop Real-Life Cryptograph. Protocols Standardization, 2010, pp. 136-149.
[3] "Message-locked encryption and secure de-duplication," in EUROCRYPT, 2013, pp. 296–312.
[4] M. Bellare, S. Keelveedhi, and T. Ristenpart, "Dupless: Server- aided encryption for deduplicated storage," in USENIX Security Symposium, 2013.
[5] J. Stanek, A. Sorniotti, E. Androulaki, and L. Kencl, "A secure data de-duplication scheme for cloud storage," in Technical Report, 2013.
[6] J. Li, X. Chen, M. Li, J. Li, P. Lee, and W. Lou, "Secure de-duplication with efficient and reliable convergent key management," in IEEE Transactions on Parallel and distributed Systems, 2014, pp. vol. 25(6), pp. 1615–1625.
*[7]* Jin Li, Xiaofeng Chen, Xinyi Huang, Shaohua Tang and Yang Xiang, "Secure Distributed System with Improved Reliability" in IEEE Transaction on Computers Volume: PP Year 2015.
[8] S. Halevi, D. Harnik, B. Pinkas, and A. Shulman-Peleg, "Proofs of ownership in remote storage systems." in ACM Conference on Computer and Communications Security, Y. Chen, G. Danezis, and V. Shmatikov, Eds. ACM, 2011, pp. 491–500.
[9] J. Xu, E.-C. Chang, and J. Zhou, "Weak leakage-resilient client-side deduplication of encrypted data in cloud storage," in ASIACCS, 2013, pp. 195–206.
[10] W. K. Ng, Y. Wen, and H. Zhu, "Private data deduplication protocols in cloud storage." in *Proceedings of the 27th Annual ACM Symposium on Applied Computing*, S. Ossowski and P. Lecca, Eds. ACM, 2012, pp. 441–446.
[11] A. Shamir, "How to share a secret," *Commun. ACM*, vol. 22, no. 11, pp. 612–613, 1979.