IJSEAT International Journal of Science Engineering and Advance Technology

Self-Assured Formal Deduplication In FusionCloud Methodology

P Lakshmi Tejaswi #1, N.Srinu#2

#1Student of M.Tech (CSE) and #2 Asst. Prof, Department of Computer Science and Engineering, QIS College of Engineering and Technology, Ongole.

Abstract:

Data deduplication is a critical system in support dispose of excess information as an option of enthralling records; it provisions simply distinct copy of file. Together with the whole associations stockpiling pattern and numerous associations encase more bits of copy in sequence. holder in point Different clients stores comparative documents in a few superior spaces. Deduplication nullify these extra duplicates by sparing as single duplicate of information and restore alternate duplicates alongside pointers that flipside to the first duplicate. It be the information pressure procedure for to build the data transfer capacity proficiency and capacity misuse. Information deduplication is tremendously utilizing as a part of distributed computing currently. Feature information administration versatile and stockpiling setback in distributed computing. Information deduplication protect the privacy of touchy data. It lives Up to expectations with merged encryption system to scramble the information before convey. Administrations frequently utilize Deduplication in reinforcement and disaster recuperation functions. Here we attempt the approved deduplication ensure, meet with simultaneous encryption to supervise the cost of security for touchy information with half and half distributed computing.

Keywords: Cloud computing, Deduplication, Convergent Encryption, Proof of Ownership Protocol, Differential Authorization.

I. Introduction:

Distributed computing strategy is widely utilized now a days. In that, processing is utilized as a part of the extensive correspondence arrange in the vein of Internet. It is the critical answer for business stockpiling in minimal effort. Distributed computing manage the cost of colossal measure of capacity in all divisions like government. In this endeavor additionally utilized for putting away our mystery information on cloud, without any setting usage fine essentials. every phase clients can get to and offers disparate chattels on cloud. Largely significant issue in distributed computing is mass measure of storage room and protection trepidation. Solitary perceptive test of distributed storage to overseeing of perpetually expanding volume of in order. advance stockpiling , versatility issue information deduplication is the most basic system and concerned more consideration now a days. It is the noteworthy procedure for information pressure , It sidestep the copy duplicates of information and store one duplicate of information. Information deduplication happens in neither piece level nor document level. In document level methodology copy records dispose of, and in piece level methodology copy squares of information happens in non-indistinguishable records. Deduplication decrease the capacity needs prepared to 90-95% for reinforcement application and 68% in standard document framework. The fundamental issue in information deduplication is security and protection for to shield the information from insider or outcast assault. For information classifiedness and encryption, distinctive client really encode there information or records. For this client utilizing a discharge key to perform encryption and unscrambling operation. For transferring a document to a cloud client he first produce concurrent key and encryption of the record then at last loads the document to the cloud. To counteract unapproved access control the possession convention is utilized to give verification that the client to be sure claims the same document when deduplication happens. After the verification, server bear the cost of a pointer to resulting client for getting to same record without expecting to transfer same document. At the point when client needs to download those documents, he will download scrambled record from cloud and next unscramble's that document utilizing concurrent key.

II. Related Work

Information deduplication innovation is technique for expanding the utilization of given information stockpiling. Deduplication distinguishes likenesses among diverse records to spare circle space. Point of interest of deduplication is to minimize stockpiling cost by putting away more information on circles. Deduplication decrease info/yield proportion and serves to bring down the expense of capacity and vitality. Deduplication likewise serves to recoup record or whole framework at certain point in time. If there should be an occurrence of deduplication on records, whole document is utilized for approval if whatever other record with comparative information is available or not. In the event that comparable duplicate is discovered then another duplicate of same document is not put away. Point of preference of document level deduplication is it needs less metadata data and nearly simple to actualize and keep up. In the event of piece level deduplication, document is isolated into lump of same sizes or different sizes. Amid deduplication, every piece is utilized for approval. On the off chance that comparable piece (of same or other record) is discovered then deduplication just stores a reference to this lump rather than its genuine substance.

Customary or conventional encryption does not work for deduplication as client encode their information with their individual keys and therefore indistinguishable of duplicate of information will have diverse figure content and deduplication is incomprehensible. Merged encryption [2] is generally used to accomplish deduplication and not too bad privacy of information. In merged key encryption same key is utilized to encode and unscramble the information as key is produced utilizing crypto realistic hash estimation of information. Since focalized key is gotten from information, it will produce indistinguishable figure content for comparative information. This serves to accomplish deduplication on cloud. Downside of focalized encryption is it is liable to animal power assault for information or records falling into known sets. In run of the mill stockpiling framework with deduplication, first customer will just send the hash estimation of the record then server will check if that hash esteem as of now exists in its database. In the event that document is as of now present on the server it requests that customer not send the record and imprints customer extra proprietor of the record. In this way customer side deduplication prompts security issue and could uncover other customer has same document of touchy data. This issue can be tended to by confirmation of proprietorship convention (PoW)[3]. PoW is in two sections and it's between two players on basic info document. In first step verifier rundowns to itself and create sort data "v". In later step prover and verifier take part in intelligent convention where verifier has sort data "v" and prover has record "F" toward the end verifier either acknowledges o rejects it. Restriction with PoW convention is it can't bolster differential approval copy check, which valuable in numerous applications. In framework with approval and deduplication, client is appointed a situated of benefits when uses are included. Every record added to cloud is likewise relegated situated of benefits that indicate which sort of client is permitted to execute copy check and permitted to get to the documents.

III. Methods Used In Secure Deduplication

Following are the secure primitive used in the secure deduplication

3.1 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.

- KeyGen SE (1λ) → k is the key generation algorithm that generates k using security parameter 1λ ;
- Enc SE (k, M) → C is the symmetric encryption algorithm that takes the secret k, and message M and then outputs the ciphertext C, and
- Dec SE (k, C) → M is the symmetric decryption algorithm that takes the secret k and ciphertext C and then outputs the original message M.

3.2 Convergent Encryption

Convergent encryption provides data confidentiality in Deduplication. A user derives a convergent key from each original data copy and encrypts the data copy with the convergent key. In addition, the user also derive tag for the data copy, such that to detect duplicates tag will be used Here, we assume that the tag holds the property of correctness , i.e., if two data copies are the same, the tags of the data also same. The user first sends the tag to the server side to check if the identical copy has been already stored for detect duplicates.[4].

3.3 Proof of Ownership

The notion of proof of ownership (PoW) 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.

III. System Model:

Architecture for Secure Deduplication

At a high level, our setting of interest is an enterprise network, consisting of a group of affiliated clients (for example,employees of a company) who will use the S-CSP and store data with deduplication technique. The S-CSP performs deduplication by checking if the contents of two files are the same and stores only one of them. Each privilege is represented in the form of a short message called *token*.

- *S-CSP*. This is an entity that provides a data storage service in public cloud. The S-CSP provides the data outsourcingservice and stores data on behalf of the users.
- *Data Users.* A user is an entity that wants to outsource data storage to the S-CSP and access the data later. In a storage system supporting deduplication, the user only uploads unique data but does not upload any duplicate data to save the upload bandwidth, which may be owned by the same use or different users. Every single file is protected with the convergent encryption key and privilege keys to realize the authorized deduplication with differential privileges.
- *Private Cloud.* Compared with the traditional deduplication architecture in cloud computing, this is a new entity introduced for facilitating user's secure usage of cloud service. Private Keys are managed by private cloud in order to give them privileges as per their designation.

IV. Design Goals :

In this paper, we address the problem of privacy preserving deduplication in cloud computing and propose a new deduplication system supporting for:

- **Differential Authorization**. Each authorized user is able to access its individual token of his file to perform duplicate check based on authority.Under this assumption, any user cannot generate a token for duplicate check out of his access or without the aid from the private cloud server.

- **Authorized Duplicate Check.** Authorized user is able to access his/her own token from private cloud,while the public cloud performs duplicate check directly and tells the user if there is any duplicate. The security requirements considered in this paper lie in two folds, including the security of file token and security of data files. For the security of file token, two aspects are defined as unforgeability and indistinguishability of file token. The details are given below.
- **Unforgeability of file token/duplicate-check token.** User make registration in private cloud for generating file token.Using respective file token he/she upload or download files on public cloud. The users are not allowed to collude with the public cloud server to break the unforgeability of file tokens. In our system, the S-CSP is honest but curious and will honestly perform the duplicate check upon receiving the duplicate request from users. The duplicate check token of users should be issued from the private cloud server in our scheme.

Indistinguishability of file token/duplicate-check token.

It requires that any user without querying the private cloud server for some file token, he cannot get any useful information from the token, which includes the file information and key information.

- **Data Confidentiality**. Unauthorized users without appropriate token , including the S-CSP and the private cloud server, should be prevented from access to the underlying plaintext stored at S-CSP. In another word, the goal of the adversary is to retrieve and recover the files that do not belong to them. In our system, compared to the previous definition of data confidentiality based on convergent encryption, a higher level confidentiality is defined and achieved.

V. Security Analysis:

Proposed system has been designed to solve the differential privilege problem in secure deduplication. The security will be analyzed in terms of two aspects, that is, the authorization of duplicate check and the confidentiality of data.

Security of Duplicate-Check Token

We consider several types of privacy we need protect, that is, i) unforgeability of duplicate-check token: There are two types ofadversaries, that is, external adversary and internal adversary. As shown below, the external adversary can be viewed as an internal adversary without any privilege. If a user has privilege p, it requires that the adversary cannot forge and output a valid duplicate token with any other privilege p' on any file F, where p does not match p'. Furthermore, it also requires that if the adversary does not make a request of token with its own privilege from private cloud server, it cannot forget and output a valid duplicate token with p on any F that has been queried. The internal adversaries have more attack power than the external adversaries and thus we only need to consider the security against the internal attacker, ii) indistinguishability of duplicate check token : this property is also defined in terms of two aspects as the definition of unforgeability. First, if a user has privilege p, given a token ϕ', it requires that the adversary cannot distinguish which privilege or file in the token if p does not match p'.

VI. Proposed System:

In our system we implement a project that includes the public cloud and the private cloud and also the hybrid cloud which is a combination of the both public cloud and private cloud. In general by if we used the public cloud we can’t provide the security to our private data and hence our private data will be loss. So that we have to provide the security to our data for that we make a use of private cloud also. When we use a private clouds the greater security can be provided.In this system we also provides the data deduplication . which is used to avoid the duplicate copies of data.User can upload and download the files from public cloud but private cloudprovides the security for that data.that means only the authorized person can upload and download the files from the public cloud.for that user generates the key and storeed that key onto the private cloud. at the time of downloading user request to the private cloud for key and then access that Particular file.

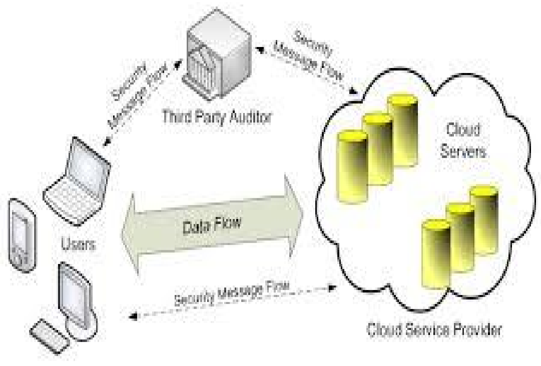


Fig2: Architecture of Authorized Deduplication

VII. Implementation:

We implement a prototype of the proposed authorized deduplication system, in which we model three entities as separate C++ programs. A *Client* program is used to model the data users to carry out the file upload process. A *Private Server* program is used to model the private cloud which manages the private keys and handles the file token computation. A *Storage Server* program is used to model the S-CSP which stores and deduplicate files.

Our implementation of the **Client** provides the following function calls to support token generation and deduplication along the
file upload process.

• FileTag(File) - It computes SHA-1 hash of the File as File Tag;

• TokenReq(Tag, UserID) - It requests the Private Server for File Token generation with the File Tag and User ID;

• DupCheckReq(Token) - It requests the Storage Server for Duplicate Check of the File by sending the file token received
from private server;

• ShareTokenReq(Tag, {Priv.}) - It requests the Private Server to generate the Share File Token with the File Tag and
Target Sharing Privilege Set;

• FileEncrypt(File) - It encrypts the File with Convergent Encryption using 256-bit AES algorithm in cipher block chaining
(CBC) mode, where the convergent key is from SHA-256 Hashing of the file;

• FileUploadReq(FileID, File, Token) – It uploads the File Data to the Storage Server if the file is Unique and updates the
File Token stored. Our implementation of the **Private Server** includes corresponding request handlers for the token
generation and maintains a key storage with Hash Map.

• TokenGen(Tag, UserID) - It loads the associated privilege keys of the user and generate the token with HMAC-SHA-1
Algorithm

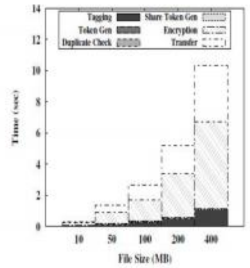


Fig. 2. Time Breakdown for Different File Size

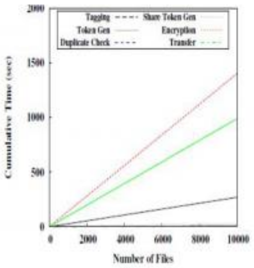


Fig. 3. Time Breakdown for Different Number of Stored Files

VIII. Assessment:

Our evaluation focuses on comparing the overhead induced by authorization steps, including file token generation and share
token generation, against the convergent encryption and file upload steps. We evaluate the overhead by varying different
factors, including 1) File Size 2) Number of Stored Files 3) Deduplication Ratio 4) Privilege Set Size. We break down the
upload process into 6 steps, 1) Tagging 2) Token Generation 3) Duplicate Check 4) Share Token Generation 5) Encryption 6)
Transfer . For each step, we record the start and end time of it and therefore obtain the breakdown of the total time spent. We present the average time taken in each data set in the figures

File Size

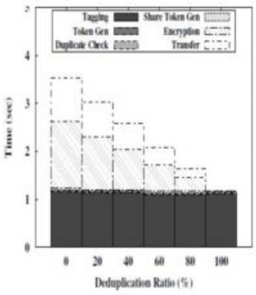
To evaluate the effect of file size to the time spent on different steps, we upload 100 unique files (i.e., without any deduplication opportunity) of particular file size and record the time break down. Using the unique files enables us to evaluate the worst-case scenario where we have to upload all file data. The average time of the steps from test sets of different file size are plotted in Figure 2. The time spent on tagging, encryption, upload increases linearly with the file size, since these operations involve the actual file data and incur file I/O with the whole file.

Number of Stored Files

To evaluate the effect of number of stored files in the system, we upload 10000 10MB unique files to the system and record the breakdown for every file upload. From Figure 3, every step remains constant along the time. Token checking is done with ahash table and a linear search would be carried out in case of collision.

Deduplication Ratio

To evaluate the effect of the deduplication ratio, we prepare two unique data sets, each of which consists of 50 100MB files. We first upload the first set as an initial upload. For the second upload, we pick a portion of 50 files, according to the given deduplication ratio, from the initial set as duplicate files and remaining files from the second set as unique files. The average time of uploading the second set is presented in Figure.



g. 4. Time Breakdown for Different Deduplication Ratio

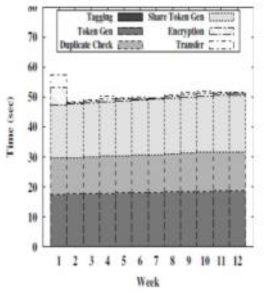


Fig. 6. Time Breakdown for the VM dataset.

IX. Conclusion:

In this paper, the idea of authorized data deduplication was proposed to protect the data security by including differential authority of users in the duplicate check. In public cloud our data are securely store in encrypted format,and also in private cloud our key is store with respective file.There is no need to user remember the key.So without key anyone can not access our file or data from public cloud. As a proof of concept, we implemented a prototype of our proposed authorized duplicate check scheme and conduct testbed experiments on our prototype. We showed that our authorized duplicate check scheme incurs minimal overhead compared to convergent encryption and network transfer.

X. References:

[1] OpenSSL Project. http://www.openssl.org/.

[2] P. Anderson and L. Zhang. Fast and secure laptop backups with encrypted de-duplication. In *Proc. of USENIX LISA*, 2010.

[3] M. Bellare, S. Keelveedhi, and T. Ristenpart. Dupless: Serveraided encryption for deduplicated storage. In *USENIX Security Symposium*, 2013.

[4] M. Bellare, S. Keelveedhi, and T. Ristenpart. Message-locked encryption and secure deduplication. In *EUROCRYPT*, pages 296–312, 2013.

[5] M. Bellare, C. Namprempre, and G. Neven. Security proofs for identity-based identification and signature schemes. *J. Cryptology*, 22(1):1–61, 2009.

[6] M. Bellare and A. Palacio. Gq and schnorr identification schemes: Proofs of security against impersonation under active and concurrent attacks. In *CRYPTO*, pages 162–177, 2002.

[7] S. Bugiel, S. Nurnberger, A. Sadeghi, and T. Schneider. Twin clouds: An architecture for secure cloud computing. In *Workshop on Cryptography and Security in Clouds (WCSC 2011)*, 2011.

[8] 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 *ICDCS*, pages 617–624, 2002.

[9] D. Ferraiolo and R. Kuhn. Role-based access controls. In *15th NIST-NCSC National Computer Security Conf.*, 1992.

[10] GNU Libmicrohttpd. http://www.gnu.org/software/libmicrohttpd/.

[11] S. Halevi, D. Harnik, B. Pinkas, and A. Shulman-Peleg. Proofs of ownership in remote storage systems. In Y. Chen, G. Danezis, and V. Shmatikov, editors, *ACM Conference on Computer and Communications Security*, pages 491–500. ACM, 2011.

[12] J. Li, X. Chen, M. Li, J. Li, P. Lee, andW. Lou. Secure deduplication with efficient and reliable convergent key management. In *IEEE Transactions on Parallel and Distributed Systems*, 2013.

[13] libcurl. http://curl.haxx.se/libcurl/.

[14] C. Ng and P. Lee. Revdedup: A reverse deduplication storage system optimized for reads to latest backups. In *Proc. of APSYS*, Apr 2013.

[15] W. K. Ng, Y. Wen, and H. Zhu. Private data deduplication protocols in cloud storage. In S. Ossowski and P. Lecca, editors, *Proceedings of the 27th Annual ACM Symposium on Applied Computing*, pages 441–446. ACM, 2012.

[16] R. D. Pietro and A. Sorniotti. Boosting efficiency and security in proof of ownership for deduplication. In H. Y. Youm and Y. Won, editors, *ACM Symposium on Information, Computer and Communications Security*, pages 81–82. ACM, 2012.

[17] S. Quinlan and S. Dorward. Venti: a new approach to archival storage. In *Proc. USENIX FAST*, Jan 2002.

P LAKSHMI TEJASWI **is** Pursuing M.Tech (Computer Science and Engineering), QIS College of Engineering and Technology Ongole, Prakasam Dist, Andhra Pradesh, India.

N Srinu Received M.Tech(CSE), MISTE. He is currently working as Asst. Professor in QIS College of Engineering and Technology, in the Department of Computer Science and Engineering, Ongole, Prakasam Dist, Andhra Pradesh, India. His Research interests are Network Security,Image processing.