Biological and Applied Sciences

Vol.61: e18160609, 2018
http://dx.doi.org/10.1590/1678-4324-2018160609 ISSN 1678-4324 Online Edition

BRAZILIAN ARCHIVES OF BIOLOGY AND TECHNOLOGY

AN INTERNATIONAL JOURNAL

# Secure Deduplication for Cloud Storage Using Interactive Message-Locked Encryption with Convergent Encryption, To Reduce Storage Space

**Jayapandian N[1]*, Md Zubair Rahman A M J[2].**

*[1]Knowledge Institute of Technology - Computer Science and Engineering, Knowledge Institute of Technology NH-47,KIOT Campus,Kakapalayam Salem, Salem, India; [2]Al-Ameen Engineering College,Erode - Computer Science and Engineering, India.*

## ABSTRACT

*The digital data stored in the cloud requires much space due to copy of the same data. It can be reduced by dedupilcation, eliminating the copy of the repeated data in the cloud provided services. Identifying common checkoff data both files storing them only once. Deduplication can yield cost savings by increasing the utility of a given amount of storage. Unfortunately, deduplication has many security problems so more than one encryption is required to authenticate data. We have developed a solution that provides both data security and space efficiency in server storage and distributed content checksum storage systems. Here we adopt a method called interactive Message-Locked Encryption with Convergent Encryption (iMLEwCE). In this iMLEwCE the data is encrypted firstly then the cipher text is again encrypted. Block-level deduplication is used to reduce the storage space. Encryption keys are generated in a consistent configuration of data dependency from the chunk data. The identical chunks will always encrypt to the same cipher text. The keys configuration cannot be deduced by the hacker from the encrypted chunk data. So the information is protected from cloud server. This paper focuses on reducing the storage space and providing security in online cloud deduplication.*

**Key words:** Data compression, Database systems, Cloud computing, Encryption, Secure storage

* Author for correspondence: njayapandian@gmail.com

## INTRODUCTION

Organizations and buyers are turning out to be progressively aware of the estimation of secure, recorded information stockpiling. In modern business world without profit any one can't get deduplication. Information conservation is regularly commanded by law and information mining has turned out to be a help in molding business methodology. For people, recorded capacity is being called upon to save sentimental and authentic ancient rarities, for example, photographs, films and individual reports. Further, whereas few would contend that business information calls for security, protection is vital for people information. For example, medicinal records and authoritative archives must be kept for stretches of time yet should not be openly available. To that end, deduplication, otherwise called single-occasion stockpiling, has been used as a strategy for augmenting the utility of a given server storage of capacity. Deduplication distinguishes regular arrangements of bytes both inside and between documents ("pieces"), and just stores a solitary occurrence of every lump paying little mind to the number of times it happens.

Thus, deduplication can significantly lessen the space required to store an expansive information set. Information security is another region of expanding significance in advanced stockpiling frameworks but deduplication and encryption are contradicted to each other. Deduplication exploits information similitude so as to accomplish a diminishment of space. Conversely, the objective of cryptography is to make cipher text indistinct from hypothetically irregular information. Along these lines, the objective of a safe deduplication framework is to give data security, against both inside and outside foes, with the space efficiency achievable through single instance stockpiling procedures. To this end, we display two ways to deal with secure deduplication confirmed. Whereas both the models are comparable, they offer somewhat distinctive security properties. Both can be applied to single server stockpiling without addition to circulated capacity. In the first, single server stockpiling, customers interface with a solitary document server that stores both information and metadata. However, metadata is put away on an autonomous metadata server, and information is put away on a progression of article based stockpiling gadgets.

Both models of our safe deduplication procedure depend on various fundamental security systems. In the first place, we use focalized data to empower encryption whereas permitting deduplication on regular pieces. Joined encryption utilizes an element of the plain text a lump as the encryption key: any customer encoding a given piece will utilize the same key to do as such, so indistinguishable plain text values will encrypt to indistinguishable cipher text values, paying little mind to who scrambles them. Whereas this procedure indicates that a specific cipher text, and along these lines plaintext, as of now exists. Foe with no information of the plaintext can't conclude the key from the scrambled lump. Second, all information chunking and encryption happens on the customer information, plain text information is never transmitted, reinforcing the framework against both interior and outer foes (1,2). At last, the guide that partners pieces to a given record is encrypted utilizing a unique key, controlling the outcome of a key trade off to a solitary document. Further, the keys are put away inside the framework in a manner that clients just need to keep up a solitary private key paying little mind to the quantity of documents to which they have admittance.

Cloud computing is one of the emerging technology; main aim is to that how effectively usage of datacenter resources (3). The duplicate files can be analyzed two ways one is server side and another one is client side. Server side duplicates once file upload in server it automatically detected, but in client side duplicate before upload file client check it manually using hash file sending operation (4). The main

advantage of deduplication is to reduce storage and improve the bandwidth (5). The level of deduplication attainable is determined by number of problems. The modern business backgrounds, deduplication percentages in the range of 4:1 (75%) to 500:1 (99.8%) are typical. Although deduplication benefits storage providers, it also creates a privacy threat for users. The randomized threshold method used in brute-force attacks. It can be maintained both client and server side deduplication system (6). Private deduplication is also one of the major problems in cloud data storage, the structure of private deduplication is constructed on the normal cryptography system (7). Here we use two encryption techniques iMLE and Convergent encryption. Convergent encryption is content hash key. It is the cryptographic algorithm that creates the identical cipher text from identical plain text. This is one type of system used to remove duplicate files in cloud storage (8). The client encrypts its plain text V with a deterministic interactive Message-Locked Encryption (IMLE) scheme under a B that is itself derived as a deterministic hash of the plain text m. Any Message-Locked Encryption (MLE) scheme specifies algorithm A,P,R,S. MLE can only provide security for unpredictable data within this range two data dimension emereged (9,10). Two types of security for MLE one is correlation and parameter dependence. Correlation means security holds when messages being encrypted and individually unpredictable are related it each other parameter dependence (11). The security holds even for messages that depends on the public parameter. IMLE turns out to be interesting in its own files and yields some other benefits to provide the first secure deduplication schemes they permits incremental updating.

**Hash Collisions**

Collision means the same hash value generated in two different data. In data storage process, whereas data corruption occurs there hash collision will raised (12). The main drawback of deduplication is computational source of power system. This is major problem of individual system also it affects the system programs and applications. Here the device overhead linked with computing the hash vales it's a main role of deduplication process.

**Target Vs Source Deduplication**

Another approach to consider information deduplication is where it happens. The data can be verified step by step, automatically deduplication data identified. This is named as target deduplication. Source deduplication guarantees that information on the information source is deduplicated (13). This for the most part happens straightforwardly inside a document framework. The record framework will occasionally examine new documents making hashes and contrast them with hashes of existing records. Target deduplication is the elimination of redundancies from a backup communication because it passes through an application sitting between the source and also the backup goal (14).

**Data Deduplication Types**

File-level Deduplication is regularly known as single-case stockpiling. Document level information deduplicates a record that must be chronicled or reinforcement that has as of now been put away by checking every one of its properties against the list. The file is overhauled and put away if the document is novel, if not then just a pointer to the current record put away as referenced. Just the single case of document is spared in the outcome and applicable duplicates are supplanted by "stubs" which

indicate the first record (15). Block-level Deduplication is square level information deduplication works on the premise of sub-document level. As the name suggests, the record is broken into portions, squares or pieces that will be inspected for already put away data repetition. The well-known way to deal with excess information is by allocating identifier to lump of information, by utilizing hash calculation, For instance it creates an ID to that specific piece. The specific ID will be contrasted in the focal record. In that event the ID is present. Then a pointer reference is made before the information put away. It creates a chance that the ID is new and does not exist, then that piece is special. The novel lump is put away and the special ID is upgraded in the Index. The span of the piece which should be checked differs from seller to merchant. Some will have altered square sizes, whereas a few others use variable sizes in the same manner few might likewise change the extent of settled square size for purpose of befuddling. Square sizes of altered size might fluctuate from 8KB to 64KB however the primary contrast with it is the smaller the part, than it will be prone to have chance to recognize it as the copy information. In the event that less information is put away then it clearly implies that information is present in the server. The main significant problem by utilizing altered size pieces is that in the event that if the document is adjusted. Deduplication result utilizes the same previous assessed result can lead to a risk of not recognizing the same repetitive information section. The squares of record would be moved or changed, and then they will move downstream from change, by balancing whatever remains of correlation (15). Variable block level Deduplication is compare the various sizes of data blocks that can reduce the chances of collision. Variable block level deduplication involves using different algorithms to decide a variable block size. The data is divided based on the algorithm's determination. Then, those blocks are stored in the subsystem.

## MATERIAL AND METHODS

Deduplication system formalized to assign individual ID for unique data, so the same ID identified means automatically it eliminate the duplicate data (16). The Information deduplication has several structures. Ordinarily, ideal approach is to actualize information deduplication over the whole of an association. Rather, to expand the advantages, associations might convey more than one deduplication methodology. It is extremely necessary to comprehend the reinforcement requirements and challenges, whereas selecting deduplication as an answer. Information deduplication has basically three structures. In spite of the fact that definitions shift, a few types of information deduplication, for example, pressure, have been around for a considerable length of time. Of late, single stockpiling has empowered the expulsion of repetitive documents from capacity situations, for example, chronicles. These three sorts of information deduplication are depicted beneath.

### Data Compression

Information pressure is a technique for reducing the extent of records. For example information pressure works inside of a document to recognize and uproot void space. This type of information deduplication is neighborhood to the record and does not mull over different documents and information sections inside of those documents. Information pressure has been accessible for a long time. However being disconnected to every specific record, the advantages are restricted. For instance, information pressure won't be helpful in perceiving and disposing of copy documents, but will freely pack each of the records.

**Single-Instance Storage**

Uprooting numerous duplicates of any record is one type of deduplication. Single-case stockpiling (SIS) situations can identify and evacuate excess duplicates of indistinguishable documents. After a record is put away in a solitary occasion stockpiling framework then, the various references to same document will allude to the first, single duplicate. Single-occurrence stockpiling frameworks contrast the substance of documents which figure out whether the approaching record is indistinguishable to a current document in the capacity framework. Content-tended to capacity is normally furnished with single-occasion stockpiling usefulness. Whereas document level deduplication abstains from putting away records that are a copy of another record. For instance, it would just take one little component (e.g., another date embedded into the title slide of a presentation) for single-example stockpiling to see two huge records as being distinctive and be put them away without further deduplication.

**Sub-file Deduplication**

Sub-document deduplication distinguishes excess information inside and crosswise over records instead of discovering indistinguishable documents as in SIS executions. Utilizing sub-record deduplication, excess duplicates of information are distinguished and wiped out even after the copied information exists, inside of isolated documents. This type of deduplication finds the one of kind information components inside of an association and distinguishes when these components are utilized inside different documents (17). Subsequently, sub-document deduplication takes out the capacity of copy information over an association. Sub-record information deduplication has enormous advantages even where documents are not indistinguishable, but rather have information components that are perceived some place in the association. Sub-document deduplication execution has two structures. Altered length sub-record deduplication utilizes a subjective settled length of information to look for the copy information inside of the documents. Albeit basic in outline, settled length sections miss numerous chances to find repetitive sub-record information. Consider the situation where addition of a man’s name to an archive’s cover sheet, the entire record is taken as new and deduplication does not take place duplication. Variable-length usage match information portion sizes to the actually happening duplication inside of records, limitlessly expanding the general deduplication proportion (In the case above, variable-length deduplication will get every single copy section in the archive, regardless of where the progressions happen). So a large portion of the associations broadly utilize information depulication innovation, which is additionally called as, single-occasion stockpiling, shrewd pressure, and limit enhanced capacity.

Deduplication System is costly to execute, keep up and buy said Rob Sims, CEO of Crossroads Frameworks (18). Moreover, Companies need more information to de-copy for sparing more cash, than with essential pressure method. Execution of Comparing so as to hash calculations and hashes makes deduplication to utilize parcel of power for handling and energy expressed. The greater part of the organizations are utilizing deduplication as a part of little apparition that can handle up to 100 terabytes information as per data connection's, this is insufficient for substantial machines. Associations can keep up execution and expansion limit by utilizing numerous machines it must be utilized if machines which bolster grouping, databases, work with same hash tables. The same number of deduplication framework won't offer excess, incase huge apparatus crashes, capacity exhibit that is working with it gets to be distracted incidentally to client.

**Security**

The more extensive deduplication framework gets to be, Sims said, the more an organization will endure if its hash database finds problems. Clients need to keep the reinforcement of the database consistently, he noted. Distinctive items won't offer the same sort of security as deduplication innovation, has not been institutionalized, Sims expressed. Security is additionally undermined by innovation since clients can't scramble information which they need to de-copy. Encryption will keep frameworks from precisely recognizing and perusing the put away data that is for deduplication, expressed Curtis Preston, an IT-framework administrations seller (19).

**Information Trustworthiness**

By breaking information into squares, deduplication clears the limits that different all information bunches (20). This makes problems for associations that consent to government related regulations that oblige organizations to independently keep sorts of various budgetary records. Since information is broken into pieces, reassembled and de-copied he noted, Lawyers will be covering up with security and uprightness questions when an organization need to demonstrate that information created is the really put away data (21). Innovation related commercial enterprises, for example, pharmaceuticals, information transfers and money related administrations have officially embraced deduplication, clarified Scott Gidley, boss officer with an information administration, Data flux and combination merchant. However innovation can devour part of assets for preparing, vitality, furthermore exorbitant is not suited to all end clients. All things considered deduplication will get to be basic component same like pressure in coming five years in the event that, it will be less unreasonable, anticipated David Russell, VP for techniques and capacity innovations with statistical surveying firm Gartner Inc. He expressed, it is no more innovation that is rising yet it is something that is observed to be in right on time standard stage. The financial matters are additionally convincing to be ignored.

**Deduplication with Data Encryption Methodology**

The main aim of the proposed system is to resolve the security problems in deduplication method in an efficient manner. The deduplication consists of three different types. They are File level deduplication, Block level deduplication and Variable deduplication. The File level deduplication method is based on the file name. It looks for multiple copies of same file. If there are a multiple copies, it stores the first copy of the file and links the references of other files. The Block level deduplication provide some backup and storage solutions. It split each data into a smaller block and creates checksum for every block uniquely. The blocks with same checksum are stored only once. It has an advantage that it handles various types of duplication. Variable dedupulication compares the size of the blocks and it will decrease the collisions of data. The deduplication method has some security problems. In the proposed system we will combine two algorithms to solve the security problems called iMLEwCE. In this iMLEwC the data is encrypted firstly then the cipher text is again encrypted. The following is the description of deduplication process: For instance, assume a set of data such as data 1, data2 etc., with respective names in the file-level deduplication. Each data individually accounts to about 27kb. This data is converted into cipher text using an algorithm called iMLE then the encrypted data is further converted into a cipher text. Thus the iMLEwCE technique enhances the security level of the proposed system. Now the original data is about 135kb is in the block-level deduplication. Here, the block of data is segmented into chunks each of 27kb. There are five segments including the

repeated data. The data which is stored twice is given an ID and is stored only once. Therefore it reduces the storage space. Convergent encryption is the concept of encrypting the data by using the cipher algorithm. The convergent encryption encrypts data by accessing the key. It gets the encryption, it will generate the same cipher text and they are encrypted with the same key. For an illustration, let us consider the data D, we will obtain a key called R=H(D),where H is the cryptographic hash function and it will encrypt the data with the above key, hence C=F(R,D)=F(H(D),D), where F is a block cipher. It has applications in cloud computing to eliminate duplicate files from storage without accessing the encryption. Here same Cryptography key is the progression of encrypting and decrypting data when we transfer the data from one host to another host by keeping the data sage. Fig. 1 proposed methodology determinations the security problems in deduplication method by using both iMLE and convergent encryption which produce the best result.

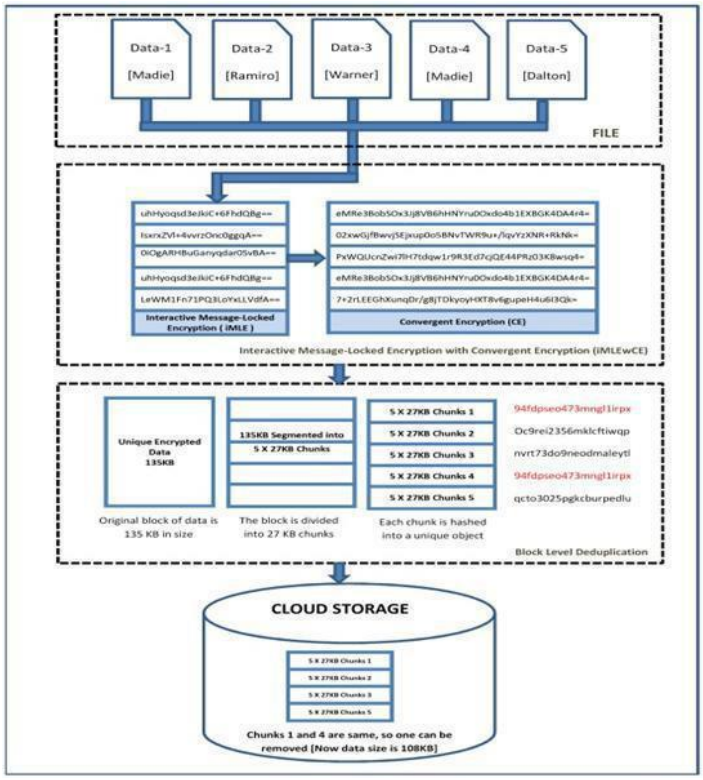



**Figure 1.** Architecture of iMLEwCE with deduplication

In the encryption process first step is key generation process after generating the key the encryption and decryption process is performed. The KEYGEN algorithm under the random key assumption is the impotent process in the deduplication encryption process keys can be constructed as follows:

1. $\epsilon$ is chosen at random from $Z^*_p$ such that the order of is k.
2. $\vartheta_1, \vartheta_2, \ldots, \vartheta_m$ are chosen at random from $Z^*_q$.
3. $\epsilon$ and $\vartheta_j (\vartheta_1 \leq j \leq m)$ are assigned as follows:

   $v_1 = \varepsilon$; $v_2 = \varepsilon^{-1}$;

   $s_{j1} = s_{j2} = \vartheta_j (1 \leq j \leq m)$.
4. The rest of key components g, u, w, $z_1, z_2, \ldots, z_m$ and $z_g$ are generated correctly according to the KEYGEN algorithm.
5. The keys are formed as:

PKint =PKdup = {q,p,g,u, $\vartheta_1$, $\vartheta_2$,$z_1$,$z_2$,…,$z_m$,$z_{sg}$}
SKint = {($s_{11}$, $s_{12}$),….,($s_{m1}$; $s_{m2}$),ω}
SKdup = null.

An auditor will be fooled into assuring the integrity of an outsourced F even if a storage server does not have the correct file but an arbitrary F'(≠F). Let us denote auxiliary audit information computed from the keys as $Tag_{int}$. Suppose an adversary A, which acts as a client, generates a key $PK_{int}$ and $SK_{int}$. A begins uploading a tuple (fid, F',$Tag_{int}$), where F' ≠F and $Tag_{int}$,fid is for F, to the storage server. An auditor who is to verify the integrity of the file F will sending the storage server a challenge (I,β), where I={α1, α 2….. αc} and β = { β **1**, β 2,…, βc . pon receiving the challenge, the storage server computes the following with F and $Tag_{int}$.

$$\mu = \sum_{i=I} \beta_i \, F'_{ij} \bmod q \; (1 \le j \le m)$$
$$Y_1 = \sum_{i=I} \beta_i \, F'_{ij} \, Y_{i1} \bmod q$$
$$Y_2 = \sum_{i=I} \beta_i \, F'_{ij} \, Y_{i2} \bmod q$$
$$T = \prod_{i \in I} t^{\beta i} \qquad (1)$$

Now getting the KEY and TAG by using this we can encrypt and decrypt the user file. EncCE = $\{\mu'_j\} 1 \le j \le m, \{x_i\}_{i=I}, Y_1, Y_2, T)$ to encrypt the data of the user. This Equation 2 holds as follow:

$$\begin{aligned}
&v_1 y_1 + v_2 y_2 + \prod_{j=1}^{m} z_j \mu = v_1 \sum_{i \in I} \beta_i Y_{i1} + v_2 \sum_{i \in I} \beta_i Y_{i2} + (v_1^{-sj1} v_2^{-sj2}) \prod_{i \in I} t^{\beta i} \\
&= v_1 \sum_{i \in I} \beta_i Y_{i1} + v_2 \sum_{i \in I} \beta_i Y_{i2} + \varepsilon^{(\delta i - \delta j)} \prod_{i \in I} t^{\beta i} \\
&= v_1 \sum_{i \in I} \beta_i (r_{i1} + \sum_{j=1}^{m} F_{ij} s_{j1}) v_2 \sum_{i \in I} \beta_i (r_{i2} + \sum_{j=1}^{m} F_{ij} s_{j2}) \\
&= v_1 \sum_{i \in I} \beta_i \, v_1 \sum_{i \in I} \beta_i (\varepsilon \sum_{i \in I} \sum_{j=1}^{m} \beta_i F_{ij} \delta i - \sum_{i \in I} \sum_{j=1}^{m} \beta_i F_{ij} \delta j) \\
&= (v_1^{ri1} v_2^{ri2}) \sum_{i \in I} \beta_i \\
&= \prod_{i \in I} x_i^{\beta i} \\
&= X
\end{aligned} \qquad (2)$$

Similarly DecSE = $\{\mu'_j\} 1 \le j \le m, \{x_i'\} i \in I, Y_1', Y_2', T')$ to decrypt the data of the user. This Equation 3 and 4 holds as follow:

$$\begin{aligned}
&v_1 y_1' + v_2 y_2' + \prod_{j=1}^{m} z_j \mu = X. \\
&= \prod_{i \in I} x_i^{\beta i'} \\
&= (v_1^{ri1} v_2^{ri2}) \sum_{i \in I} \beta_i'
\end{aligned} \qquad (3)$$

$$\begin{aligned}
&= v_1 \sum_{i \in I} \beta_i' \, v_2 \sum_{i \in I} \beta_i' (\varepsilon \sum_{i \in I} \sum_{j=1}^{m} \beta_i' F_{ij} \partial j' - \sum_{i \in I} \sum_{j=1}^{m} \beta_i' F_{ij} \partial j') \\
&= v_1 \sum_{i \in I} \beta_i' (r_{i1}' + \sum_{j=1}^{m} F_{ij} s_{j1}') v_2 \sum_{i \in I} \beta_i' (r_{i2}' + \sum_{j=1}^{m} F_{ij} s_{j2}') \\
&= v_1 \sum_{i \in I} \beta_i' y_{i1}' + v_2 \sum_{i \in I} \beta_i' y_{i2}' + \varepsilon^{(\partial i - \partial j)} \prod_{i \in I} t^{\beta i'} \\
&= v_1 \sum_{i \in I} \beta_i' y_{i1}' + v_2 \sum_{i \in I} \beta_i' y_{i2}' + (v_1^{-sj1} v_2^{-sj2}) \prod_{i \in I} t^{\beta i'}
\end{aligned} \qquad (4)$$

The file F will sending the storage server a challenge (α), where α= {α**1**,α2,…, α c-1}. Upon receiving the challenge, the equation 5 storage server computes the following with Ḟ and $Tag_{int}$.

$$v = U\alpha F'_{ij} \bmod q (1 \le j \le m)$$
$$Y_1 = \mathrm{U}\ \alpha\ Y_1 \bmod \mathrm{q}$$
$$Y_2 = \mathrm{U}\ \alpha\ Y_2 \bmod \mathrm{q}$$
$$\mathrm{T} = \prod K^{\alpha} \tag{5}$$

Now we can complete the encrZpt and decrZpt the user file. EncCE ={ $\upsilon'_j$}1≤ j ≤ m,{$y_i$} $i \in I$ , $Z_1, Z_2$, K) to encrZpt the data of the user. This Equation 6 holds as follow:

$$w_1 z_1 + w_2 z_2 + \prod_{j=1}^{n} z_j \mu = W_1 \cup \alpha\ Z_{i1} + W_2 \cup \alpha\ Z_{i2} + (w_1^{-sj1}\, w_2^{-sj2}) \prod_{i \in I} t^{\beta i}$$
$$= W_1 \cup \alpha\ Z_{i1} + W_2 \cup \alpha\ Z_{i2} + \varepsilon^{(\delta i - \delta j)} \cup \alpha$$
$$= W_1 \cup \alpha_i (r_{i1} + \sum_{j=1}^{n} F_{ij} S_{j1})\, W_2 \cup \alpha_i (r_{i2} + \sum_{j=1}^{n} F_{ij} S_{j2})$$
$$= W_1 \sum_{i \in I} \beta_i\ W_2 \sum_{i \in I} \beta_i (\varepsilon \cup \alpha F_{ij}\, \vartheta j - \cup \alpha F_{ij}\, \vartheta j)$$
$$= (w_1^{ri1}\, w_2^{ri2}) \sum_{i \varepsilon I} \beta_i$$
$$= \prod Y^{\alpha}$$
$$= \mathrm{Y}. \tag{6}$$

SimilarlZ DecSE = { $\vartheta'_j$}1≤ j ≤ m,{$y_i'$} $i \in I$ , $Z_1'$, $Z_2'$, K') to decrZpt the data of the user. This Equation 7 and 8 holds as follow:

$$w_1 z_1' + w_2 z_2' + \prod_{j=1}^{n} z_j \mu = \mathrm{Y}$$
$$= \prod Y^{\alpha'} \tag{7}$$

$$= (w_1^{ri1'}\, w_2^{ri2'}) \cup \alpha'$$
$$= w_1 \cup \alpha'\ w_2 \sum_{i \varepsilon I} \beta_i' (\varepsilon \cup \alpha' F_{ij}\, \vartheta j' - \cup \alpha' F_{ij}\, \vartheta j')$$
$$= w_1 \cup \alpha' (r_{i1}' + \sum_{j=1}^{n} F_{ij} S_{j1})\, w_2 \cup \alpha'\, r_{i2}' + \sum_{j=1}^{n} F_{ij} S_{j2}$$
$$= w_1 \cup \alpha'_i Z_{i1}' + w_2 \cup \alpha'_i Z_{i2}' + \varepsilon^{(\vartheta i - \vartheta j)} \prod_{i \varepsilon I} t^{\beta i'}$$
$$= w_1 \cup \alpha'_i Z_{i1}' + w_2 \cup \alpha'_i Z_{i2}' + (w1^{-sj1}\, w_2^{-sj2}) \tag{8}$$

Convergent encryption provides information privacy in deduplication. A user derives a convergent key from each unique data copy and encrypts the data copy with the convergent key. The client likewise infers a tag for the data duplicate, such that the tag will be utilized to identify copies. Here, we accept that the label accuracy property holds, that is if two data duplicates are the same, then their labels are the same. To distinguish copies, the client first sends the tag to the server side to check if the indistinguishable duplicate has been as of now put away. Note that both the focalized key and the tag are freely inferred and the tag can't be utilized to conclude the united key and trade off data privacy. Both the encoded data duplicate and its comparing tag will be put away on the server side.

## RESULTS AND DISCUSSION

The experiment was setup by using the 8GH processor and 2TB hard disk in intel core i Pentium and the code were 00000022200…… 00. Done in java and simulated in Cloudsim. We considered 100 different data sets; each data set consists of various sizes and multiple files. Various data sets are tested by using the deduplication with convergent encryption method which is in the proposed technique. After the testing we get the best result. The result has slight variation when associate with conventional result of deduplication

**Table I.** Different file execution time and size

| File Name | File Size (KB) | Encryption Time [iMLEwCE] (Sec) | After Deduplication File Size(KB) | File Name | File Size (KB) | Encryption Time [iMLEwCE] (Sec) | After Deduplication On File Size(KB) |
|---|---|---|---|---|---|---|---|
| File001 | 135 | 0.16463 | 108 | File051 | 3840 | 4.68293 | 3810 |
| File002 | 204 | 0.24878 | 189 | File052 | 204 | 0.24878 | 191 |
| File003 | 174 | 0.21220 | 153 | File053 | 593 | 0.72317 | 570 |
| File004 | 192 | 0.23415 | 192 | File054 | 6922 | 8.44146 | 6392 |
| File005 | 794 | 0.96829 | 708 | File055 | 249 | 0.30366 | 210 |
| File006 | 843 | 1.02805 | 814 | File056 | 5023 | 6.12561 | 4920 |
| File007 | 472 | 0.57561 | 406 | File057 | 5504 | 6.71220 | 5120 |
| File008 | 1003 | 1.22317 | 985 | File058 | 302 | 0.36829 | 294 |
| File009 | 382 | 0.46585 | 362 | File059 | 483 | 0.58902 | 483 |
| File010 | 529 | 0.64512 | 502 | File060 | 5912 | 7.20976 | 5820 |
| File011 | 753 | 0.91829 | 753 | File061 | 9583 | 11.68659 | 9420 |
| File012 | 189 | 0.23049 | 182 | File062 | 5810 | 7.08537 | 5720 |
| File013 | 472 | 0.57561 | 456 | File063 | 132 | 0.16098 | 123 |
| File014 | 948 | 1.15610 | 932 | File064 | 1295 | 1.57927 | 1295 |
| File015 | 294 | 0.35854 | 287 | File065 | 5912 | 7.20976 | 5720 |
| File016 | 430 | 0.52439 | 427 | File066 | 1250 | 1.52439 | 978 |
| File017 | 948 | 1.15610 | 918 | File067 | 5024 | 6.12683 | 4928 |
| File018 | 1501 | 1.83049 | 1478 | File068 | 5923 | 7.22317 | 5880 |
| File019 | 492 | 0.60000 | 480 | File069 | 5820 | 7.09756 | 5730 |
| File020 | 2039 | 2.48659 | 2002 | File070 | 9404 | 11.46829 | 9350 |
| File021 | 482 | 0.58780 | 470 | File071 | 1043 | 1.27195 | 978 |
| File022 | 948 | 1.15610 | 932 | File072 | 492 | 0.60000 | 482 |
| File023 | 6293 | 7.67439 | 6287 | File073 | 599 | 0.73049 | 570 |
| File024 | 902 | 1.10000 | 889 | File074 | 392 | 0.47805 | 380 |
| File025 | 4832 | 5.89268 | 4710 | File075 | 4920 | 6.00000 | 4840 |
| File026 | 1034 | 1.26098 | 1001 | File076 | 3059 | 3.73049 | 3010 |
| File027 | 442 | 0.53902 | 410 | File077 | 124 | 0.15122 | 124 |
| File028 | 1923 | 2.34512 | 1802 | File078 | 4245 | 5.17683 | 4139 |
| File029 | 129 | 0.15732 | 129 | File079 | 3948 | 4.81463 | 3820 |
| File030 | 414 | 0.50488 | 401 | File080 | 693 | 0.84512 | 680 |
| File031 | 102 | 0.12439 | 102 | File081 | 359 | 0.43780 | 329 |
| File032 | 391 | 0.47683 | 329 | File082 | 6860 | 8.36585 | 6728 |
| File033 | 4929 | 6.01098 | 4910 | File083 | 394 | 0.48049 | 360 |
| File034 | 5932 | 7.23415 | 5910 | File084 | 5930 | 7.23171 | 5820 |
| File035 | 404 | 0.49268 | 380 | File085 | 5368 | 6.54634 | 5359 |
| File036 | 393 | 0.47927 | 404 | File086 | 367 | 0.44756 | 348 |
| File037 | 5923 | 7.22317 | 5820 | File087 | 607 | 0.74024 | 590 |
| File038 | 583 | 0.71098 | 542 | File088 | 7003 | 8.54024 | 7003 |
| File039 | 8829 | 10.76707 | 8730 | File089 | 9489 | 11.57195 | 9328 |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| File040 | 435 | 0.53049 | 435 | File090 | 3296 | 4.01951 | 3129 |
| File041 | 839 | 1.02317 | 810 | File091 | 6903 | 8.41829 | 6850 |
| File042 | 1294 | 1.57805 | 1120 | File092 | 9340 | 11.39024 | 9310 |
| File043 | 4910 | 5.98780 | 4820 | File093 | 547 | 0.66707 | 547 |
| File044 | 596 | 0.72683 | 510 | File094 | 347 | 0.42317 | 322 |
| File045 | 239 | 0.29146 | 210 | File095 | 7950 | 9.69512 | 7813 |
| File046 | 9483 | 11.56463 | 9320 | File096 | 647 | 0.78902 | 629 |
| File047 | 2389 | 2.91341 | 2239 | File097 | 5755 | 7.01829 | 5621 |
| File048 | 598 | 0.72927 | 520 | File098 | 327 | 0.39878 | 327 |
| File049 | 842 | 1.02683 | 842 | File099 | 8644 | 10.54146 | 8512 |
| File050 | 5920 | 7.21951 | 5280 | File100 | 458 | 0.55854 | 420 |
| **Overall** | **85224** | **103.93** | **82598 KB** | **Overall** | **179291** | **218.64** | **175322 KB** |

The result of the proposed methodology has minor variation like execution time, storage space. Sometime the data has multiple files, so it will take little more additional time. Even the result of the proposed technique has slender variation when associated with ordinary result it is the best result. The result of the proposed technique is shown below with an illustration. In the Table I consider File001. Its file size is 135kb and the execution time accounts to about 0.16463 now after deduplication (DD) the size of the file is reduced to 108kb.

**Execution Time Analysis**

The execution time of a particular data without deduplication (Without DD) is 0.4527. After deduplication the execution time is about 0.4912 (With DD). Although there is a slight variation in the execution time of the data, but it is highly secured only when it is deduplicated. This is depicted in the Fig. 2.

**Encryption with Deduplication Time Analysis**

From the experimental study, a data iMLE execution time is about 0.03829. Its Convergent encryption (CE) execution time is about 0.12634. Therefore, the execution time of iMLEwCE is 0.16463. Now the execution time with DD is 0.4912 then the execution time of Encryption with DD is 0.82046. Even though the execution time is varied a bit, it does not affect the CPU performance. The Fig. 3 gives a clear representation.

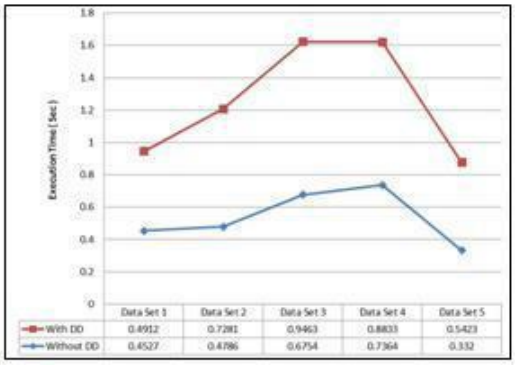



**Figure 2**. With DD and without DD execution time

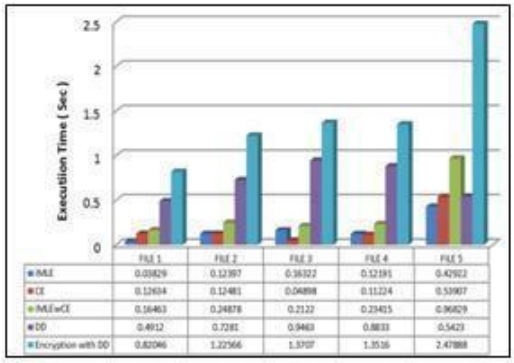



**Figure 3.** iMLEwCE with deduplication

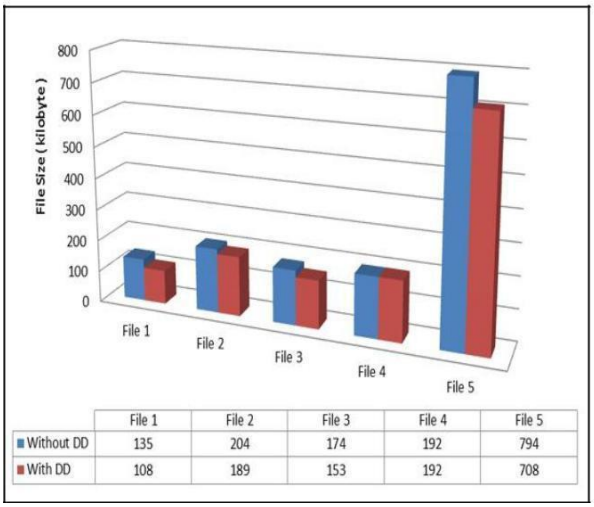
**Storage Space Comparison**

Consider a file size without DD of about 135kb. After DD the file size is extracted to about 108kb. Thereby it reduces the storage space. In concern with the file 4 in the

Fig. 4, the execution time with DD and without DD is the same because the data is unique. Among the 100 data sets the average storage size has also decreased in 2.49%.

**Attackers Ratio Comparison**

The attacker's ratio is reduced in iMLEwCE methodology when compared to iMLE and CE algorithms. This test case data is gathered during dynamic run time. This is shown in the Fig. 5.




| | File 1 | File 2 | File 3 | File 4 | File 5 |
|---|---|---|---|---|---|
| Without DD | 135 | 204 | 174 | 192 | 794 |
| With DD | 108 | 189 | 153 | 192 | 708 |

**Figure 4**. With and without DD storage space comparison

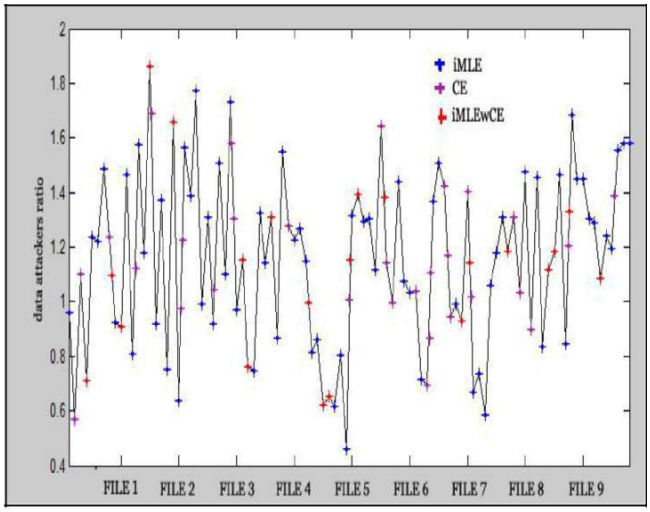



**Figure 5.** Attacker's ratio Analysis

## CONCLUSION

Cloud computing has achieved a development that leads to a beneficial stage. Yet the data stored here are not that much secured. To overcome this we have adopted a methodology named iMLEwCE and deduplication. Thus the data stored in the system are highly secured due to cipher texts and it reduces the storage space due to block level deduplication. Although the execution time has been slightly increased in comparison with the existing systems, it does not affect the overall performance of the system rather it has only enhanced the security level.

## ACKNOWLEDGMENT

The authors wish to thank the management of Knowledge Institute of Technology Salem for providing all the computational facilities to carry out this work.

## REFERENCES

1. Storer M, Greenan K, Long D, Miller E. Secure data deduplication. Proceedings of the 4th ACM international workshop on Storage security and survivability - StorageSS '08. 2008;.
2. Clarke I, Sandberg O, Wiley B, Hong T. Freenet: A Distributed Anonymous Information Storage and Retrieval System. Designing Privacy Enhancing Technologies. 2001;:46-66.
3. PATRASCU APATRICIU V. Digital Forensics in Cloud Computing. Advances in Electrical and Computer Engineering. 2014;14(2):101-108.
4. Xu J, Chang E, Zhou J. Weak leakage-resilient client-side deduplication of encrypted data in cloud storage. Proceedings of the 8th ACM SIGSAC symposium on Information, computer and communications security - ASIA CCS '13. 2013;.

5. Liu J, Asokan N, Pinkas B. Secure Deduplication of Encrypted Data without Additional Independent Servers. Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security - CCS '15. 2015;.
6. Harnik D, Pinkas B, Shulman-Peleg A. Side Channels in Cloud Services: Deduplication in Cloud Storage. IEEE Security & Privacy Magazine. 2010;8(6):40-47.
7. Ng W, Wen Y, Zhu H. Private data deduplication protocols in cloud storage. Proceedings of the 27th Annual ACM Symposium on Applied Computing - SAC '12. 2012;.
8. Douceur J, Adya A, Bolosky W, Simon P, Theimer M. Reclaiming space from duplicate files in a serverless distributed file system. Proceedings 22nd International Conference on Distributed Computing Systems.
9. Bellare M, Keelveedhi S, Ristenpart T. Message-Locked Encryption and Secure Deduplication. Advances in Cryptology – EUROCRYPT 2013. 2013;:296-312.
10. Bellare MKeelveedhi S. Interactive Message-Locked Encryption and Secure Deduplication. Lecture Notes in Computer Science. 2015;:516-538.
11. Abadi M, Boneh D, Mironov I, Raghunathan A, Segev G. Message-Locked Encryption for Lock-Dependent Messages. Advances in Cryptology – CRYPTO 2013. 2013;:374-391.
12. Wu B, Chen J, Wu J, Cardei M. A Survey of Attacks and Countermeasures in Mobile Ad Hoc Networks. Wireless Network Security. :103-135.
13. Rahumed A, Chen H, Tang Y, Lee P, Lui J. A Secure Cloud Backup System with Assured Deletion and Version Control. 2011 40th International Conference on Parallel Processing Workshops. 2011;.
14. Clarke I, Sandberg O, Wiley B, Hong T. Freenet: A Distributed Anonymous Information Storage and Retrieval System. Designing Privacy Enhancing Technologies. 2001;:46-66.
15. Li J, Chen X, Li M, Li J, Lee P, Lou W. Secure Deduplication with Efficient and Reliable Convergent Key Management. IEEE Trans Parallel Distrib Syst. 2014;25(6):1615-1625.
16. Li J, Li Y, Chen X, Lee P, Lou W. A Hybrid Cloud Approach for Secure Authorized Deduplication. IEEE Trans Parallel Distrib Syst. 2015;26(5):1206-1216.
17. Bessonov M, Heuser U, Nekrestyanov I, Patel A. Open Architecture for Distributed Search Systems. Intelligence in Services and Networks Paving the Way for an Open Service Market. 1999;:55-69.
18. Geer D. Reducing the Storage Burden via Data Deduplication. Computer. 2008;41(12):15-17.
19. Puzio P, Molva R, Onen M, Loureiro S. ClouDedup: Secure Deduplication with Encrypted Data for Cloud Storage. 2013 IEEE 5th International Conference on Cloud Computing Technology and Science. 2013;.
20. Liu C, Liu X, Wan L. Policy-Based De-duplication in Secure Cloud Storage. Trustworthy Computing and Services. 2013;:250-262.
21. Shacham HWaters B. Compact Proofs of Retrievability. Advances in Cryptology - ASIACRYPT 2008. :90-107.

Received: February 03, 2016;
Accepted: July 14, 2016