WWW.IJRPR.COM

International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Enhance Cloud Data Compression using Duplication

Sharik Ahmad and Ms. Sneha Deokate

Department of Computer Science and Engineering
School of Research and Technology, People' University, Bhopal [MP], India

A B S T R A C T

Online social networks have become the standard in recent years. Social media networks allow users to communicate with their family, friends, and colleagues. Data published on social networks and other forms of media spreads rapidly and thoroughly, making it enticing to attackers to gain information. Community networking environments may not come without risk. There are several security and privacy concerns with the user's shared information, particularly when the user uploads personal content like photographs, videos, and audios. The scale of data storage is a big issue in the cloud. This topic focuses on reducing storage space. The deduplication method is used for storage space compression. The reproduction of data during the relocation process is referred to as duplication. Overall execution time is lowered by 34%. Storage space is reduced by 33.38% on average when using the deduplication approach. The method proposed compares storage space and execution time. This proposed approach yields superior results for reducing storage data size in a cloud setting.

Keywords:iMLE, Message Locked Encryption Algorithm, **Deduplication**

1. Introduction

Cloud computing establishes a trend of extracting values from delivery services while increasing speed and integrity. It reduces the time required to create an application product to real preparation. Cloud computing includes visualisation, on-demand deployment, internet processes, computing utility delivery, and open source software. The servers provide a pool of resources that may be managed as needed, as well as the relationship of applications to computation or storage. As virtualization enables a dynamic data centre, network resources change regularly to meet each workload and business requirement.

2. Objective of the Research

The primary goal is to reduce the amount of data stored in the cloud. To data storage compression, the proposed algorithm named as iMLEwCE and deduplication method is used.

3. Research Methodology

The primary goal of cloud computing is the online sharing of software and hardware resources. Some steps have been taken to make sense of the above remark, but there are still some problems to be overcome in the domain of cloud networking. The most significant difficulty that providers and customers confront in the cloud environment is security. Some of the key benefits of cloud technology are lower costs and resource re-provisioning. To secure online documents, a common encryption method is utilized, and the characteristic that may suit well with security demands in the cloud environment is discovered.

3.1. Message Locked Encryption Algorithm

The most serious issue in cloud computing is data storage space. Storage space reduction is a significant task for service providers. Because it has a direct impact on the cost and performance of the service. The deduplication method is used to tackle this problem. The two methods for analyzing duplicate files

Coresponding Author Contact: +91 88406 25634
E-mail address: shariqahmed180@gmail.com

are server-side and client-side. Due to duplicates, the submitted file on the server will be erased automatically, whereas on the client side before uploading the file. When a hash file is used for a transmitting action, it is manually verified for duplicates. The primary function of deduplication is to increase network bandwidth and reduce storage requirements. File storage space in the private and public cloud environments is an important consideration. The two encryption algorithms utilized here are iMLE and Convergent encryption.

Convergent encryption is the content of the hash key. Using a cryptographic technique, identical cipher text can be generated from identical plain text. This method is used in cloud storage to delete duplicate files. For the uncertain data, a specific range of two data dimensions emerges. Only MLE (Message Locked Encryption) will provide better security. MLE provides two types of security: correlation and parameter dependence. Correlation occurs when messages are encrypted that are related to other parameter dependence and individually unpredictable in the security holds. Even in the public parameter, the reliance is secure. The iMLE provides benefits such as a secure deduplication approach in its personal file and certificates incremental apprises.

3.2. Data Storage Reduction Methods

Hash Collisions: - Collision occurs when two separate data sets generate the same hashing value. When data corruption occurs, a hashing collision happens during the storing process. The biggest disadvantage of deduplication is the computational source of power system. The system programs and applications are affected, which is the most serious problem with any particular system. The primary function of the deduplication procedure is to reduce hash value overhead.

Data Compression: - The information pressure technique can be used to minimize the size of records. The document's information pressure operates within it and detects the empty space (void space). Over time, information pressure will also be available. During the disconnect, the benefits are limited to those specific records.

Data Deduplication Types: - File-level Deduplication is another name for single case stocking. Individual documents should be reinforced, which is referred to as file level deduplication. Subfile reinforcement is referred to as block level deduplication. Allocating identifiers and employing hashes are the best approaches to deal with large amounts of data. For example, an ID is assigned to that specific piece of data. The previous assessed data result is employed by the deduplication result, increasing the chance of not recognizing the same repetitive information section. The data will be pushed away from the alteration by balancing whatever remains in correlation. Variable level duplication refers to the analysis of data block variables. The primary benefit of this strategy is that it reduces collisions.

Sub-file Deduplication: - Individual data is reviewed in the database using deduplication, and duplicate data should be deleted. Sub-document deduplication engages the potential to copy information across an organization. Because information is viewed in association, deduplication of sub-record information has significant advantages over other techniques. The sub-document deduplication has two structures. A subjective fixed length of information technique is employed by sub-record deduplication, which has a changed length, to examine the copy information inside of the documents. Although method can detect repeating sub-record information.

3.3. iMLEwCE Algorithm with Deduplication Architecture

The suggested interactive Message Locked Encryption (iMLE) and Convergent Encryption (CE) techniques improve data security. The purpose of deduplication is to overcome the storage space problem. There are three types of deduplication processes. Deduplication at the file, block, and variable levels. The file level system examines the entire file; all file names and file types must be saved and preserved in the address pointer link. The files that are repeated are easily identifiable. At the block level, a single block should be divided into numerous subblocks while maintaining the checksum notion.

Figure 1 depicts the architecture of the iMLEwCE algorithm with deduplication. iMLEwCE is an abbreviation for interactive Message Locked Encryption with Convergent Encryption. A file is a storage unit in a computer system. The computer can perform read and write operations on files that include data and applications. The file contains data and refers to characters from any set that are being gathered and translated for analysis. The character is made up of numbers, graphics, sounds, video, and text. The first portion of the above-proposed diagram is file, which contains five data files in notepad. The information is denoted as data1, data2, data3, data4, and data5. Dalton, Ramiro, Madie, Warner, and Madie are their names.

These data act as an input for the next block. The block is named as interactive messaged lock encryption with convergent encryption. An MLE scheme is a five-tuple of PT algorithms, the last two of which are deterministic. The parameter generation method P returns a public parameter P on input 1"A. The key-generation method K, given an input P and a message M, returns a message-derived key K ←$KP (M). The encryption method E returns a cipher string C ←$EP (K, M) for inputs P, K, and M. The decryption method D returns DP (K, C) E 0*U l for inputs P, K, and a cipher text C. The tag generation algorithm produces T←TP(C) for inputs P, C. Associated to the scheme is a message space MsgSpMLE that associates to any "A € N a set MsgSpMLE(λ) ⊆{0,1}*. it requires that there is a function C l such that, for all " λ € N, all P € [P(1"λ)] and all M €{0,1}*, any output of EP(KP(M),M) has length Cl(P," λ,|M|), meaning the length of a cipher text depends on nothing about the message other than its length. The decryption correctness condition requires that DP(K,C) = M for all " λ € N, all P € [P(1" λ)], all M € MsgSpMLE("λ), all K € [KP(M)] and all C € [EP(K,M)].

The tag correctness condition requires that there is a negligible function δ: N→[0,1], called the false negative rate, such that Pr[TP(C) 6= TP(C0)] ≤ δ(λ) for all λ€ N, all P € [P(1λ)] and all M € MsgSpMLE(λ), where the probability is over C←$EP(KP(M),M) and C0←$EP(KP(M),M).hence the MLE is deterministic if K and E are deterministic. The data in the MLE will be converted as unreadable. It is being observed that if MLE is deterministic then it has perfect tag correctness, meaning a false negative rate of 0.

Convergent encryption occurs when the cryptosystem generates identical ciphertext from identical text. The system generates a cryptographic hash of the question's plaintext. The plaintext is then encrypted by the system using its hash as the key. The hash is encrypted with a key chosen by the user and saved. The input is often obtained from the previous block, where the data is in an unreadable format. Unreadable data is provided as input, which is

encrypted in the previous state, and encryption also occurs in the present state. The user creates a tag for information copy, specifying that the tag will be used for sight duplicates.

Deduplication could be implemented as a device driver for the block layer, which sits between the file system layer and the underlying block device. This technique has the virtue of being universal and applicable to any file system. Finding all deduplication chances will potentially save additional space. If a strong hash function with a low collision rate, such as SHA-1, is employed, it may be safe to presume that matches actually relate to the same data and forgo a byte-by-byte comparison.

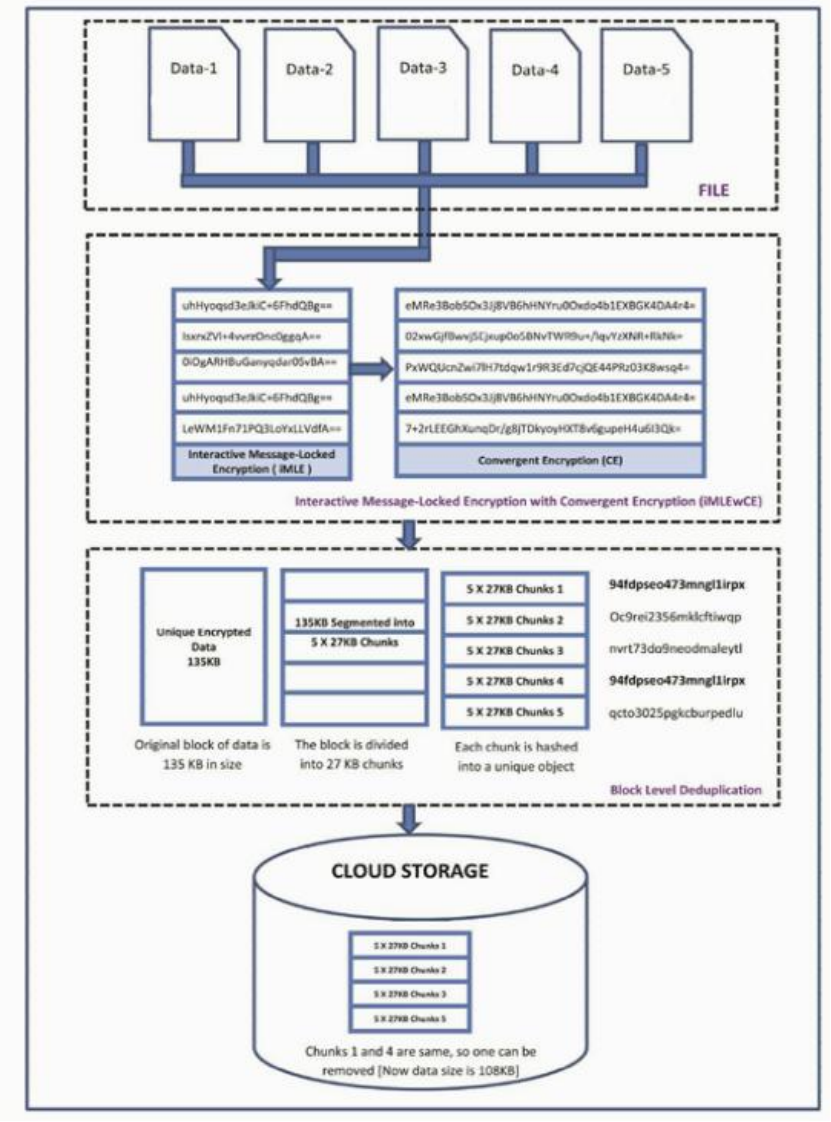

Figure 1 Proposed iMLEwCE Architecture with Deduplication

Furthermore, the saved hashes could be used to verify data on reads and locate duplicate blocks. The problem of this strategy is that significant amounts of space must be used to store block hashes. To avoid re-hashing all the blocks on mount, the data must be persistent on disk. Because the disk is reasonably sized, storing all hash data in memory would use a significant amount of RAM. As a result, the disk block cache's effectiveness is considerably reduced. It is also difficult to cache this type of information because hashes are practically random and might potentially match any block on the disk.

As in some content-addressable systems, file system block pointers could be substituted with hashes in written data. This method has various drawbacks. A block pointer should be bigger than hash to avoid clashes. To store and retrieve blocks by hash, additional hardware is required. Finally, this method is incompatible with existing file systems. The proposed technique employs block level duplication. Digital data is often stored in logical pools and controlled by a hosting provider, which is referred to as Cloud storage.

The proposed graphic contains four data points where the same data is repeated. Each data object is produced, however only four objects are created because the data is repeated and they save memory. The memory assigned for the five sets of data is 135kb, but when the data repeats, the memory capacity is lowered to 108kb in the cloud storage system. The file in question comprises five data components. The encryption layer receives these data pieces. These encryption layers include two algorithms that are employed in the hybrid technique. The interactive Message Locked Encryption algorithm is the first. The fundamental advantage of this technique is that it does not keep the secret key value forever. The hackers attempt to hack the secret key; however, they are unsuccessful because the secret key is dynamically altered.

The data will be encrypted using this iMLE algorithm. For example the input data is ramiro that will be encrypted in the form of IsxrxZVI+4vvszOnc0ggqA== using iMLE algorithm. The second operation of encryption process is using convergent encryption algorithm. The basic principle of the convergent encryption is hashing the content, while using the cryptography method to change the plaintext to ciphertext. Here the plaintext of convergent encryption is IsxrxZVI+4vvszOnc0ggqA==, based on the convergent encryption algorithm that plaintext will be converted to some ciphertext. That ciphertext is O2xwGjfBwwjSEjxup0o5 BNvEWR9u+/lqvyzxRN+RkNk=. The two-layer encryption that is hybrid encryption produce the better security compares to the existing model.

The proposed architecture's next procedure is to use deduplication to solve the storage problem. Deduplication, in general, implies removing duplicate values or repetitive items from a file. This problem can be solved via block level deduplication. Block level refers to dividing a single activity into many blocks and storing data pieces with reference addresses. In each data, the input data element size is 27 KB. Data here refers to ciphertext, which is O2xwGjfBwwjSEjxup0o5BNvEWR9u+/lqvyzxRN+RkNk=. This information will be saved in individual blocks. Using the hashing table, each block has an address for locating the specific contents. If the same data appears, the hash value address is also the same, making it simple to detect the repeated data. That repeating data is known as duplicate data. It removes duplicate data at the block level based on the hash value. The proposed method's final step is to store all of the data in the cloud. The size of each data element is 27 KB. At this time, only one data piece is repeated and eliminated from the file. The original file size of 27KB can be decreased to 108 KB. The original file is 135 KB in size. The suggested solution addresses two main cloud problems: security and storage. The encryption algorithm solves the security problem, while the deduplication method solves the storage problem. It outperforms the present method in terms of security.

3.4. Pseudocode Structure of Deduplication

The deduplication process dynamic and random key generations are generated.

1. The $\mathcal{E}$ order is k, the random order Z*p, then choose the €.
2. Z*q is also random order, then choose S1, S2,…..,Sm
3. Sj(S1 S Y S m)& € are mentioned like this:

R1= $\mathcal{E}$ & R2 = $\mathcal{E}$ -1; Tj1 = Tj2 = Sj($1 \leq Y \leq m$)

4. The remaining key mechanisms are u,g,w, H1,H2,…., Hm & Hg . The KEYGEN algorithm is generated these key.
5. The key are structured:

DUint =DUdup = {p,q,w,u, R1, R2,H1,H2,…,Hm,Hsg}

EUint = {(T11, T12),….,(Tm1; Tm2),co} EUdup = empty.

The primary goal of convergent encryption is to guarantee data security and privacy. All information is to be derived from a single convergent key. This unique key should be used in a randomized fashion. Each data set should have a distinct name and address. If the same file appears in many lists, the same key value is used. This proposed solution employs the deduplication method and the encryption algorithm; it maintains data security while reducing storage space.

4. Results and Discussion

Cloud computing is an internet-based technology that connects circulated and similar systems while also providing a virtual system. Three computing resources are provided by the internet cloud computer. The cloud's hardware and software components are handled by a third party on a remote server. It is legal to utilize for both corporations and end users. The dependable services deliver next-generation data centers built on compute and storage virtualization. They are reassured by the most recent cloud computing innovations. The fundamental reason is that users fail to run complex programs and communicate sensitive data over cloud. Trust is commonly used to assess data integrity, reliability, availability, and turnaround efficiency.

4.1. iMLEwCE Algorithm and Storage Space Analysis

Cloudsim is used to create and test the proposed algorithm. The suggested technique accepts data set of 40 various sizes. Table 1 depicts the sample data size and execution time. For 1 Kb, the average execution time is 0.001219 seconds. The input file is 537 KB in size and takes 0.654855 seconds to execute. The file size is reduced to 521 KB after deduplication. Using the same approach, the execution time for a huge file of 9284 KB is 11.32 seconds, and the file compression size is 914 KB. Data 1, Data 2, and Data 40 are all listed as file names. The top 20 files have an average file size of 39012 KB and an execution time of 47.5739seconds. The file size is reduced to 11243 KB after deduplication. The identical strategy is used in additional 20 files, with an average execution time of 72.8597 seconds. The size of a 40 file before duplication is 98757 KB; following deduplication, the file size is gradually reduced to 65789 KB. The average file size reduction is 33.38%.

Table 1 Different Size of File Execution and Storage Space Comparison

File Name	File Size (KB)	After Deduplication File Size(KB)	Encryption Time [iMLEwCE](Sec)	File Name	File Size (KB)	After Deduplication File Size(KB)	Encryption Time [iMLEwCE] (Sec)
Data01	294	282	0.3585	Data21	3563	3124	4.34512
Data02	421	412	0.5134	Data22	135	121	0.16463

Data03	743	731	0.9061	Data23	643	592	0.78415
Data04	136	124	0.1658	Data24	2464	2143	3.00488
Data05	723	711	0.8817	Data25	124	114	0.15122
Data06	962	942	1.1731	Data26	6743	6243	8.22317
Data07	1356	129	1.6536	Data27	8372	8003	10.2097
Data08	2546	243	3.1048	Data28	246	231	0.3
Data09	568	550	0.6926	Data29	754	729	0.91951
Data10	953	941	1.1621	Data30	2124	2005	2.59024
Data11	211	196	0.2573	Data31	4621	4237	5.63536
Data12	5786	563	7.0561	Data32	4685	4134	5.71341
Data13	568	545	0.6926	Data33	324	301	0.39512
Data14	831	820	1.0134	Data34	5732	5353	6.99024
Data15	400	389	0.4878	Data35	4666	4232	5.69024
Data16	5147	500	6.2768	Data36	2346	2131	2.86097
Data17	9284	914	11.321	Data37	2678	2312	3.26585
Data18	932	918	1.1365	Data38	2368	2152	2.8878
Data19	719	701	0.8768	Data39	2589	2175	3.15732
Data20	6432	632	7.8439	Data40	4568	4214	5.57073
Overall	**39012 KB**	**11243 KB**	**47.5739 Sec**	**Overall**	**59745 KB**	**54546 KB**	**72.8597 Sec**

Execution Time Analysis: - The execution time analysis is explained in table 2 and figure 2. The encryption and deduplication processes can be used to calculate the execution time analysis. Consider data set 1, which took 0.4527 seconds to complete without the deduplication process. With the deduplication procedure, it takes 0.4912 seconds. In data set 2, the execution time is 0.4786 seconds without the deduplication method, and it is 0.7281 seconds with the deduplication method. Consider data set 4, where the execution time without the deduplication process is 0.7364 seconds and the execution time with the deduplication process is 0.8833 seconds. Consider data set 5: without deduplication, it takes 0.332 seconds to complete, while with deduplication, it takes 0.5423 seconds. This demonstrates that the execution time grows during the deduplication process since repeated data may be repaired rapidly, but massive amounts of data cannot be fixed quickly. Despite the increased execution time, storage space and security have been increased. Figure 2 depicts the execution time analysis.

Table 2 Execution Time Comparison

Execution Method	File Set 1	File Set 2	File Set 3	File Set 4	File Set 5
Without DD (Sec)	0.4527	0.4786	0.6754	0.7364	0.332
With DD (Sec)	0.4912	0.7281	0.9463	0.8833	0.5423

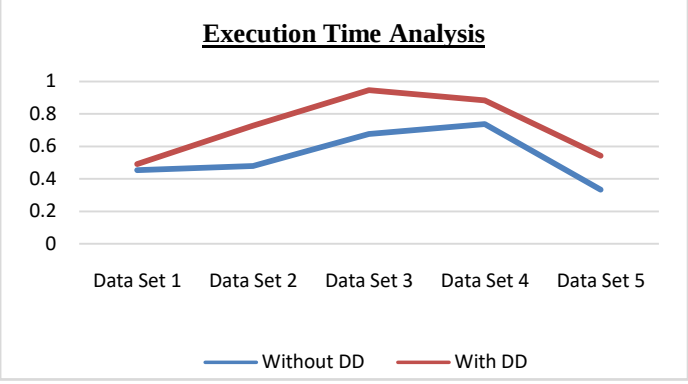

Figure 2 Deduplication Execution Time Analysis

Encryption with Deduplication Time: - The encryption with deduplication time is explained in Table 3 and Figure 3. The execution time of many algorithms has been discussed. The file 1, which takes 0.03829 seconds to process with the iMLE method, takes 0.12634 seconds with CE. The suggested combined iMLE and CE algorithms have an execution time of 0.16463 sec. The deduplication method takes 0.4912 seconds to execute, while encryption with deduplication techniques takes 0.82046 seconds. In iMLE, file 2 takes 0.12397 seconds; with the CE, it takes 0.12481 seconds. In file 2, the combined algorithm takes 0.24878 seconds to execute. The deduplication approach takes 0.7281 seconds to execute and 1.2256 seconds to encrypt utilizing the deduplication method. Consider file 5, where the iMLE algorithm takes 0.42922 seconds to execute while the CE approach takes 0.53907 seconds. It takes 0.96829 seconds to run the combined algorithms. The deduplication algorithm takes 0.5423 seconds to execute. The combined encryption and deduplication processes take 2.47888 seconds. Despite the fact that the goal of integrated algorithms is to boost security, iMLE and CE do so. As security and storage are enhanced, the encryption with deduplication algorithms execution time increases when compared to the deduplication approach. The graphic below explains encryption with deduplication time analysis.

Table 3Encryption and Deduplication Time Comparison

Algorithms	FILE 1	FILE 2	FILE 3	FILE 4	FILE 5
iMLE	0.03829	0.12397	0.16322	0.12191	0.42922
CE	0.12634	0.12481	0.04898	0.11224	0.53907
iMLEwCE	0.16463	0.24878	0.2122	0.23415	0.96829
DD(Deduplication)	0.4912	0.7281	0.9463	0.8833	0.5423
Encryption with DD(Deduplication)	0.82046	1.22566	1.3707	1.3516	2.47888

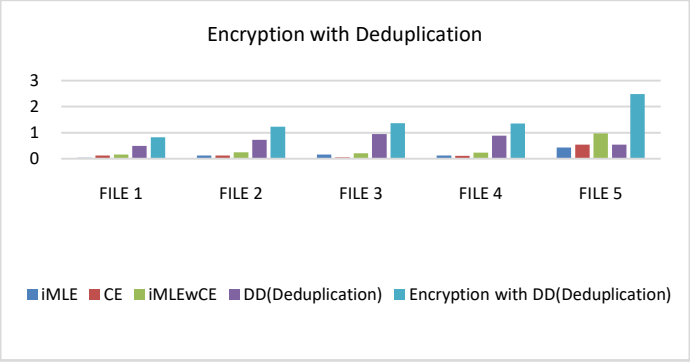

Figure 3 Encryption with Deduplication Time Analysis

Storage Space Comparison: - Figure 4 and table 4 show a comparison of storage space. Consider file 1, which has a size of 135KB without the deduplication procedure. The same file with the deduplication approach has a file size of 108KB. The file size in file 2 is 204KB without the deduplication technique, and 189KB using the deduplication approach. Consider file 3, which has a size of 174KB without the deduplication process. The same file utilizing the deduplication method has a file size of 153KB. Consider file 5, which has a file size of 794KB without the deduplication approach. The deduplication approach is 543KB in size. The size of file 4 remains unaltered, indicating that the deduplication procedure never performed in file 4.

Table 4 Deduplication Storage Space Comparison

Compression Method	File 1	File 2	File 3	File 4	File 5
Without DD	135 KB	204 KB	174 KB	192 KB	794 KB
With DD	108 KB	189 KB	153 KB	192 KB	543 KB

All of the above examples illustrate that the deduplication method reduced file size. As a result, storage space might be reduced. Figure 4 depicts a storage space comparison that has been thoroughly detailed.

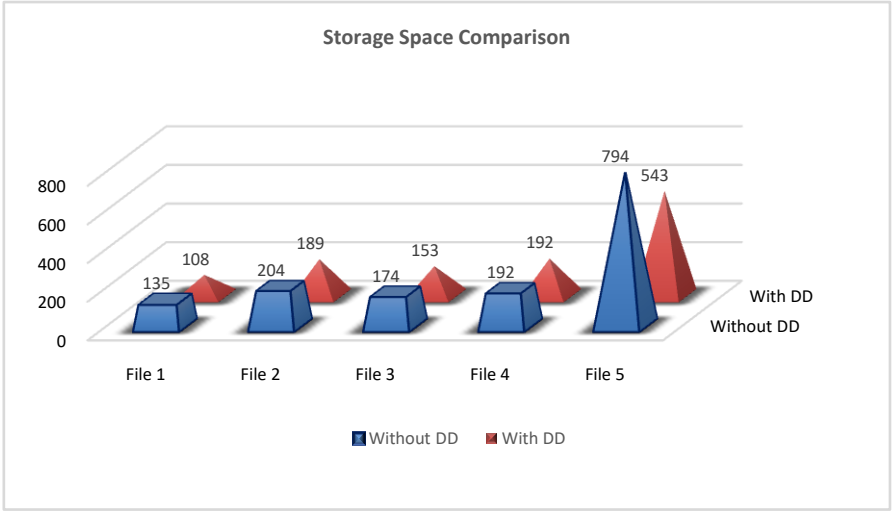

Figure 4 Storage Space Comparison Analysis

The data security and Storage size are major problems in cloud. These are all the consequences occurring at the time of data transmission in cloud server. The data security here refers to the theft of data by attackers. The data security is achieved by these AES algorithm in an efficient way. The first module of this chapter is using anonymization method with advanced encryption standard algorithm. Anonymization is an alternative method of encryption process. The input data is anonymized in some format that should be stored in cloud server. Here anonymized output is input plaintext of AES algorithm. The AES algorithm is again encrypting the data and gives some ciphertext output. All this ciphertext data is stored in cloud server dynamically. The overall execution time is reduced by 0.6%. But CPU utilization is increased by 0.2%. The security level has been increased. The input data is simulated and get better outcome using Cloudsim tool. The second module of this chapter is to concentrate storage space reduction and increase security. The security improvement here uses hybrid algorithm of iMLEwCE. The storage space compression uses deduplication method. The duplication refers to the replication of data while the relocation process. The overall execution time is reduced to 34%. Using deduplication method storage space is reduced averagely 33.38%. The proposed method is to compare storage space and execution time. This proposed algorithm provides the better result for secured data in cloud environment.

5. Conclusion

The proposed method employs the iMLEwCE algorithm to address the issue of deduplication. The execution of the 40 different example files. Without using the deduplication approach, the average storage space is 98757 KB. The average file size after using this deduplication approach is 65789 KB. In this proposed solution, the 32968 KB file size was reduced, saving 33.38% of storage space. The existing approach takes 0.53502 seconds to execute. The proposed technique takes 0.71824 seconds to execute. The only distinction is 0.18322 seconds. The execution time is slightly longer than in the previous system, but it provides better security and saves storage space. The proposed methodology reduces the average system implementation cost.

Cloud storage is now one of the most useful features of cloud computing. It provides a place on the cloud for data warehousing with the user's flexibility and time bound access, similar to how a user has its own local storage, but with substantially better reliability. Cloud service providers are compensated by the user for accessing the service that they supply. Cloud service providers maintain datacenters to host cloud computing applications and assure resilience in the event of a data center disaster.

The proposed model provides storage space savings. Storage space reduction is used to lower storage costs. Quality and profit are always required by the industry for their services. The decrease in storage is to satisfy the profit factor. In terms of business, customers expect more dependable service at a lower cost. With consumer pleasure, the industry anticipates lower investment and better profit. The proposed solution is intended to meet the expectations of the service provider and the customer.

6. Future Work

Users' data can be stored in the cloud's data centers. This classification technique should take into account numerous factors such as access frequency and data size. Once the data has been categorised and tagged, the level of security associated with each tagged data element can be applied. The level of security chosen depends on the type of data includes confidentiality, encryption, integrity, and storage.

REFERENCES

[1] Atayero, AA &Feyisetan, O, 'Security issues in cloud computing: The potentials of homomorphic encryption'. Journal of Emerging Trends in Computing and Information Sciences, vol. 2, no. 10, pp. 546-552, 2011.

[2] Bellare, M, Keelveedhi, S &Ristenpart, T, 'Message-Locked Encryption and Secure Deduplication'. In Theory and Applications of Cryptographic Techniques, International Conference of the Springer, Lecture Notes in Computer Science, vol. 7881, pp. 296-312, 2013.

[3] Celesti, A, Tusa, F, Villari, M &Puliafito, A, 'Security and cloud computing: Intercloud identity management infrastructure'. In Enabling Technologies: Infrastructures for Collaborative Enterprises (WETICE), 19th International Workshop of the IEEE, pp. 263-265, 2010.

[4] Che, J, Duan, Y, Zhang, T & Fan, J, 'Study on the security models and strategies of cloud computing'. In Power Electronics and Engineering Application, International Conference of the Procedia Engineering Elsevier, pp. 586-593, 2011.

[5] Chen, Y & Sion, R, 'On Securing Untrusted Clouds with Cryptography'. In Privacy in the Electronic Society, 9th Annual Workshop of the WCE, pp. 109-114, 2012.

[6] Harnik, D, Pinkas, B & Shulman-Peleg, A, 'Side channels in cloud services: Deduplication in cloud storage'. IEEE Security & Privacy, vol. 8, no. 6, pp. 40-47, 2010.

[7] Hojabr, M, 'Ensuring data storage security in cloud computing with effect of kerberos'. International Journal of Engineering Research & Technology, vol. 1, no. 5, pp. 22-25, 2012.

[8] Junaid Hassan, Danish Shehzad, Usman Habib, Muhammad Umar Aftab , Muhammad Ahmad , RamilKuleev , and Manuel Mazzara, ' The Rise of Cloud Computing: Data Protection, Privacy, and Open Research Challenges-A Systematic Literature Review (SLR)' , Computational Intelligence and Neuroscience, Hindawi, Volume 22, June 2022, pp. 1-26, 2022. https://doi.org/10.1155/2022/8303504

[9] Kouatli, I, 'Managing Cloud Computing Environment: Gaining Customer Trust with Security and Ethical Management'. In Information Technology and Quantitative Management (ITQM 2016), International Conference of the Elsevier, Procedia Computer Science, vol. 91, pp. 412-421, 2016.

[10] Lee, K, 'Security threats in cloud computing environments'. International Journal of Security and Its Applications, vol. 6, no. 4, pp. 25-32, 2012.

[11] Lewko, A & Waters, B, 'Decentralizing Attribute-Based Encryption'. In Theory and Applications of Cryptographic Techniques, Annual International Conference of the Springer, Lecture Notes in Computer Science, vol. 6632, pp. 568-588, 2011.

[12] Li, J, Li, YK, Chen, X, Lee, PP & Lou, W, 'A hybrid cloud approach for secure authorized deduplication'. IEEE Transactions on Parallel and Distributed Systems, vol. 26, no. 5, pp. 1206-1216, 2015.

[13] Liu, Z, Chen, X, Yang, J, Jia, C & You, I, 'New order preserving encryption model for outsourced databases in cloud environments'. Journal of Network and Computer Applications, vol. 59, no. 1, pp. 198-207, 2016.

[14] Lombardi, F & Di Pietro, R, 'Heterogeneous Architectures: Malware and Countermeasures'. Secure System Design and Trustable Computing, pp. 421-438, 2016.

[15] Mahajan, P & Sachdeva, A, 'A Study of Encryption Algorithms AES, DES and RSA for Security'. Global Journal of Computer Science and Technology, vol. 13, no. 15, pp. 01-07, 2013.

[16] Maluleka, SM &Ruxwana, N, 'Cloud Computing as an Alternative Solution for South African Public Sector: A Case for Department of Social Development'. In Advances in Intelligent Systems and Computing, New Advances in Information Systems and Technologies Springer, vol. 444, pp. 481-491, 2016.

[17] Mansukhani, B & Zia, TA, 'An empirical study of challenges in managing the security in cloud Computing'. In Information Security Management, 9th Australian International Conference of the Research Online, pp. 172-181, 2011.

[18] Manzoor, A, 'Cloud Computing Applications in the Public Sector'. Cloud Computing Technologies for Connected Government, pp. 215-245, 2016.

[19] Paul, M & Mandal, JK, 'A Novel Symmetric Key Cryptographic Technique at Bit Level Based on Spiral Matrix Concept'. In Information Technology, Electronics and Communications (ICITEC–2013), International Conference of the IAIRS, pp. 06-11, 2013.

[20] Priteshkumar Prajapati and Parth Shah, 'A Review on Secure Data Deduplication: Cloud Storage Security Issue' Journal of King Saud University - Computer and Information Sciences, Science direct, Volume 34, Issue 7,Pages 3996-4007, 2022.

[21] Rocha, F & Correia, M, 'Lucy in the sky without diamonds: Stealing confidential data in the cloud'. In Dependability of Clouds, Data Centers and Virtual Computing Environments, 1st International Workshop of the IEEE, pp. 129-134, 2011.

[22] Sharma, S, Verma, A, Singh, S & Pandey, V, 'Data Protection in the Cloud: Dynamic Password Authentication and Certificate-Based Authorization'. In Cloud,Big Data and Trust, International Conference of the Cloud, pp. 13-15, 2013.

[23] Shivani Sengar and Ruchika Mishra, 'Secure Sharing & Data Deduplication over Cloud: A Survey,' International Journal of Scientific & Engineering Research, Volume 8, Issue 1, pp. 354-362, 2017.

[24] Sujaritha , Akshara D , Ashfak Ahamed A , Chandhinishri V S, 'Privacy Preserving Verification Scheme for Cloud Platform Using DML' ICCCEBS 2021, Journal of Physics: Conference Series, pp. 1-5, 2021. doi:10.1088/1742-6596/1916/1/012154.

[25] Ubale, SA, Apte, SS &Bokefode, JD, 'Developing Secure Cloud Storage System Using Access Control Models'. In Data Engineering and Communication Technology, Springer International Conference of Advances in Intelligent Systems and Computing, vol. 469, pp. 141- 147, 2017.

[26] Wang, C, Wood, LC, Abdul-Rahman, H & Lee, YT, 'When traditional information technology project managers encounter the cloud: Opportunities and dilemmas in the transition to cloud services'. International Journal of Project Management, vol. 34, no. 3, pp. 371- 388, 2016.