ISSN 2321-807X
Volume 12 Number13
Journal of Advances in chemistry

SEQUENTIAL ENCRYPTION FOR MULTIPLE CHUNKS OF DATA IN CLOUD ENVIRONMENT

Ashok George[1], Dr. A.Sumathi[2]

[1]Research Scholar, [2]Professor

georgeashokphd@gmail.com

[1,2] Department of ECE, Adhiyamaan College of Engineering

Anna University, Tamil Nadu, India

ABSTRACT

Cloud computing is a next generation computer paradigm for IT firm. The vital service of cloud computing is cloud storage, that permits owner to move data from their native computing systems to the cloud. Storing our confidential data to a public cloud is a challenging issue in cloud computing because of unauthorized access to the data. The proposed algorithm splits data and encrypts it using two different algorithms to improve the security level of the confidential data than the various techniques of existing encryption algorithms. The data is first partitioned into multiple chunks and then efficient encryption algorithms such as RSA algorithm and Blowfish algorithm is used for data encryption. It further proposes an efficient data access using indexing technique to retrieve the confidential data from cloud. Finally, it needs to decrypt multiple chunks to get actual data from public cloud. The objectives of the proposed techniques are to store confidential data in public cloud and ensure more security than the existing techniques.

Keywords: Cloud computing, RSA, Blowfish, Partition, Indexing

1. INTRODUCTION

Cloud computing as “Group of IT resources (database, server and applications) which are available on an on-demand basis provided by a service company, available through the internet, and provide resource pooling among Various Clients" [1]. It has ability to run a program or application on many connected computers at the same time in distributed network. Cloud computing is connected with another standard for the procurement of operation of computer infrastructure. The computing infrastructural paradigm changes the area of network to decrease the expenses connected with software and hardware resources maintenance [2-10]. Progress in networking technology and expansion in the requirement for processing assets has incited numerous individuals and enterprises to outsource their processing storage requirements and composite information management method to the cloud owing to its more stupendous adaptability and less expense proficiency [3-4]. In commercial public cloud current outsourcing practices uncovers both processing results and information. Distributing sensitive information and critical application to public clouds raises enormous security concerns, particularly when the outsourced data contain more confidential information. For example, military information, geographical areas, enemy positions in defence, organization financial records, research data, personally recognize health record and so on. Assuming that these information succumb to the wrong hands, then such a security breach could prompt declination of war, wrong behaviour etc. Improvement in demand for data outsourcing in enterprises leads to key administration of enterprise information [1,3-4].

The cloud users must be guaranteed that information facilitated on the cloud will be more secure. Preserve sensitive data is a legitimate and ethical need in todays cloud environment. A lot of organizations move their information to the cloud and experience numerous progressions and there are various tests to succeed. To be successful, data security in cloud depends upon more than essentially applying suitable security information techniques and counter measures. Authentication and Authorization are computer based security efforts to establish. To make the information in cloud more secure from different assaults, encryption of sensitive data to be carried out before it is stored or transmitted.

This paper deals with effective practices than the existing cryptography techniques and secure data by encoding it into an unreadable protected format. The proposed system model works by breaking information into multiple chunks, and encrypt chunks with solid RSA [4-9, 5-20] and Blowfish algorithm [6-16]. The encrypted piece information is archived in cloud server. The idea is to partition the dataset into a number of small pieces called data shards. CG-index (Cloud Global-Index) is intended to be conscious of and enhanced for such form of partitioning. Rather than building a record for the entire dataset, CG-index constructs a neighbourhood B+-tree file for every information shard called an index shard. It is a distribution unit in CG-record, which is archived and supported on a special index server. CG-index depends on this record distribution method for desired scalability [7-18].

Queries are served by looking all qualified index shards and the handles are gathered by their information shard identifiers. An optimization is to recover an assembly of handles from the same information shard in a batch mode. To recover information from the cloud, worker appeals suitable identifiers. The client then sends the identifiers to the cloud supplier who utilizes it to discover and furnish a proportional return, the appropriate encrypted records which the user decrypts using his private key. The essential building design of the proposed system is described in following Figure 1.

ISSN 2321-807X
Volume 12 Number13
Journal of Advances in chemistry

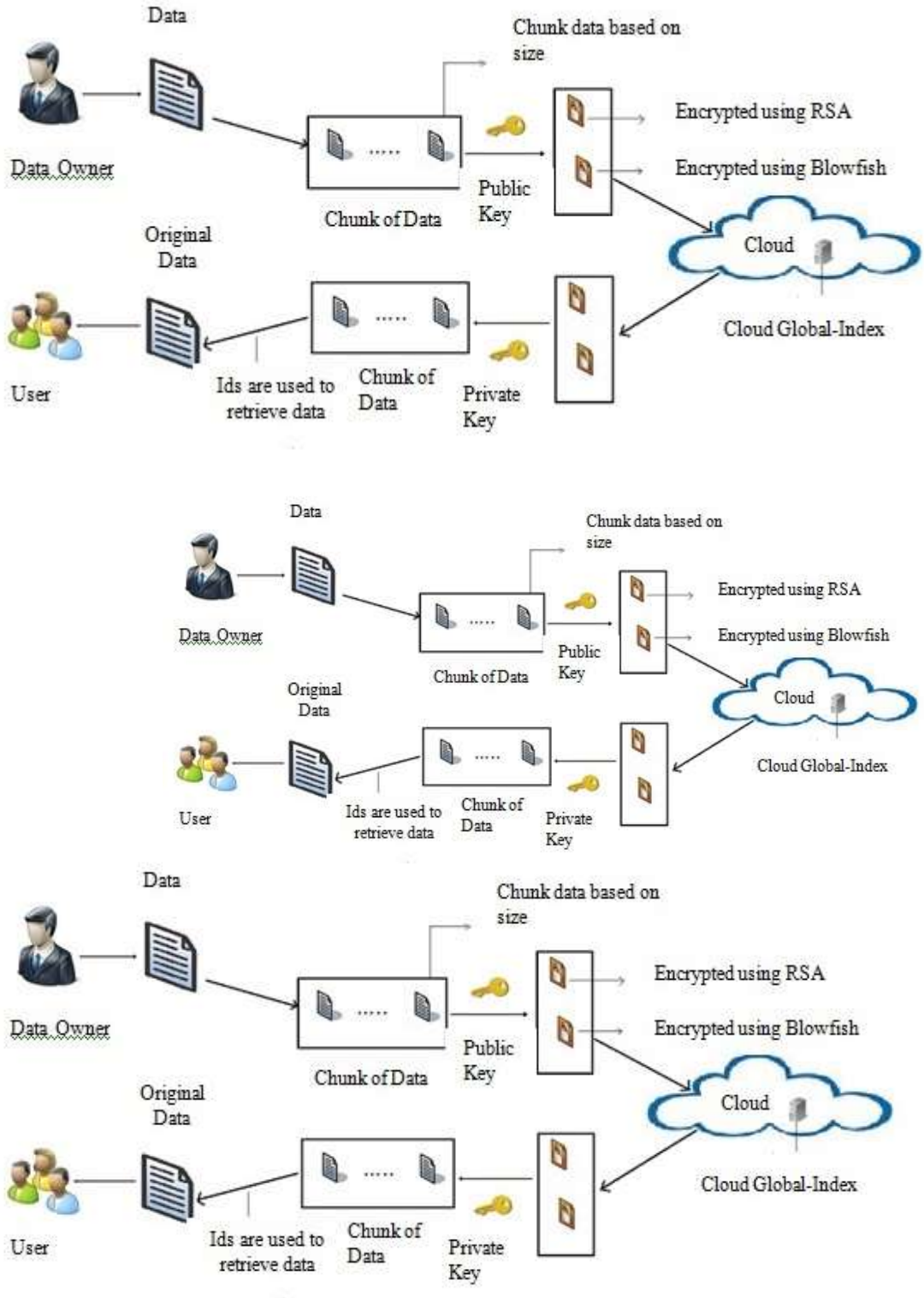

Fig. 1 : Framework of Proposed Method

2. RELATED WORK

The purpose of this paper work mainly depends on identifying security mechanisms in cloud computing, so that enterprises can store large amount of data in cloud more securely. The cloud computing increasers the sharing of resources effectively and it attains intelligibility, economies of scale in a network. In cloud, the organisation interacts with a large number of connected distributed networks to depict a variety of computing idea [8]. This benefit makes an organization to move their services to the cloud environment. To move the services and high sensitive information to the

cloud must be conscious of the dangers and risks. Cloud storage has biggest issue of confidentiality and integrity [5-20]. Different tests incorporate managing with Integrity, Security, accessibility issues, loss of data, and unavailable services in systems because of attacks from unauthorized users outstanding that while the profits of utilizing an infrastructure in open cloud are clear, it presents extremely genuine security and protection from dangers. The study and research of the work facilitates privileged user to access the data from the cloud by utilizing RSA and Blowfish encryption algorithm and extended analysis to efficient retrieval from cloud by utilizing B+ indexing algorithm.

The first step towards in cloud computing is constructing a scalable method for data storage. If a data owner or enterprises wants to share sensitive information with users, they will upload data in cloud storage after an encryption of data. Encryption is outstanding amongst the best data security controls technique in cloud environment. Encryption is dependent upon the methods and techniques of the cryptography security solution with this kind of cryptographic security service steps, cloud user will not be able to access encrypted data. There are many research associated to ensuring protection for data in cloud computing environment using various encryption techniques [7-9].

Ron Rivest, Adi Shamir, and Leonard Adleman developed an encryption algorithm in 1977 called RSA Algorithm [10-15] . Another algorithm used in this paper for secure encryption is Blowfish, which was introduced in 1993 by Bruce Schneier, an American cryptographer [6-16]. The proposed technique of sequential encryption in cloud using RSA and Blowfish algorithm enhance the security than other existing algorithms. The fundamental numerical structure of the RSA process is very basic and it is dependent upon essential mathematical operations on substantial numbers, this could be reason for individuals do feel better on working with a calculation. Blowfish is not subject to any licenses and hence openly accessible for anybody to utilize [10-15, 6-16]. Thus RSA and Blowfish benefit has contributed to its reach in cryptographic method. An encryption method regularly comprises of a general technique and an encryption key. Assuming that clients need to safely correspond with one another, they must impart a key, which is utilized to both encryption and decryption of messages.

In Cloud, data is divided into multiple chunks and encrypted sequentially before store into cloud storage environment. The chunks of data are encrypted and distributed randomly in cloud environment, thus distributed chunks are retrieved efficiently by using B+ tree based indexing method [1, 7-18]. B-tree is a quite usually utilized indexing method in storage management frameworks, and prior deal with B-tree normally kept on ones archived in a solitary workstation's memory space. The large scale clustered dataset was index by proposed B+ tree algorithm in distributed environment. The B+-tree index and hash index which are used to exhibit the effective framework was proposed by K.-L. Wu and S. Wu. The B+-tree is dispersed around the accessible nodes by randomly dispersing every B+- tree node to a processing node [1]. Conversely with that, the disseminated index can efficiently help different queries and give high upgrade and retrieve rate.

3. SYSTEM OVERVIEW

The proposed model is designed to achieve secure and scalable access control on outsourced data in cloud storage environment. The techniques used in this proposed system is as follows: (i) divide the data into multiple chunks for sequential encryption (ii) RSA and Blowfish algorithm for data security and (iii) B+ tree algorithm for efficient retrieval process. At present guaranteeing security in cloud storage has turned one of the hugest concerns for the analysts. These issues are embraced in proposed model to furnish some result corresponded with security. Cloud computing has various users, for example, the education community, enterprises and individual users who have diverse inspiration to move data and services to cloud. The partition of data, encryption and indexing are the main functionalities of this system.

In the proposed model, data's are dividing into multiple chunks for sequential encryption. Assume that *k* size data need to store in cloud securely. Here *k* size data are divided into multiple chunks by *k/n,* i.e., n is fixed size of dividend. The data's are divided into multiple chunks is illustrated in Figure 2.

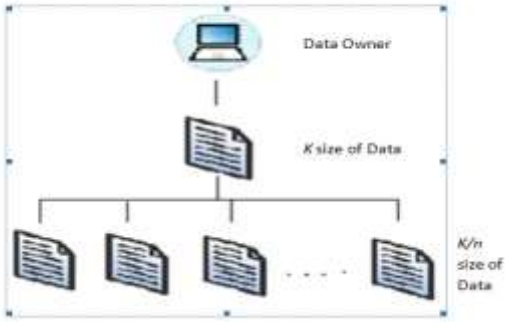

Fig. 2 : Data's are divided into Multiple Chunks

3.1 RSA Encryption Scheme

In the proposed model RSA and Blowfish encryption algorithm is used sequentially for making the data and communication secure. RSA is a block cipher and generally used asymmetric algorithm, in which integer representation for an every messages. Initially, the sensitive data is encrypted before store into cloud storage by comprises of Private-Key and Public-Key. In cloud storage environment the private-key is known only to the data owner, whereas public-key is

known to all. Thus encryption is carried out and decrypt by users in the cloud. If the data is encrypted with the Public-Key and corresponding private key can be used to decrypt data from cloud.

The strongest RSA algorithm holds three steps, encryption, decryption and key generation. The generation of key process is carried out by first picking two prime numbers randomly, such as p and q.

At that point the number n ought to be calculated:

$$n = pq \quad (1)$$

Then, a function $\varphi(n)$ is calculated:

$$\varphi(n) = (p-1)(q-1) \quad (2)$$

However, integer e is chosen:

$$1 < e < \varphi(n) \quad (3)$$

At last, the d is calculated: $d = e\text{-}1 \bmod \varphi(n)$, Such that: $de \bmod \varphi(n) = 1$, and $\varphi(n)$ and e are co-prime. Finally, the result got is (n, d) is the private key and (n, e) is the public key.

Assume that M is a plain text of some integer and encrypted cipher text as C, encryption key as a e , decryption key as a D, and modulo number as a n. The following equations show an encryption of message.

$$C = Me \bmod n \quad (4)$$

The plain text M is obtained by decrypting the cipher text C is shown in following equation.

$$M = C_D \bmod n \quad (5)$$

If the clients need to expand the effectiveness of RSA by picking a less value for the public key exponent, e or for the private key exponent, d, but this technique cause an attack that might break the entire cryptography, In order to avoid this attack, d must not be less than 256 bits in length, when n has a length of 1024 bits.

3.2 Blowfish Encryption Scheme

The next encryption algorithm used in the proposed system is Blowfish. It is a symmetric public domain algorithm with a 64 bits block size and variable key lengths are used in this algorithm. Blowfish algorithm security has been widely tried and demonstrated. It has been liable to an important value of cryptosystem and full Blowfish encryption has never been broken. Blowfish is likewise one of the quickest block ciphers. Symmetric block cipher divides a message up into fixed-length blocks of 64-bit and length of key up to 448 bits, during encryption and decryption processes. Messages that are not a different of eight bytes in size should be padded. Its principle outline criteria are to be quick, minimized, straightforward and having variable security. Blowfish makes utilization of cycles/rounds. It comprises of 16 Rounds and uses fixed and large key-dependent S-boxes, It has full 16 rounds usage; no ways are known to break the security at this point.

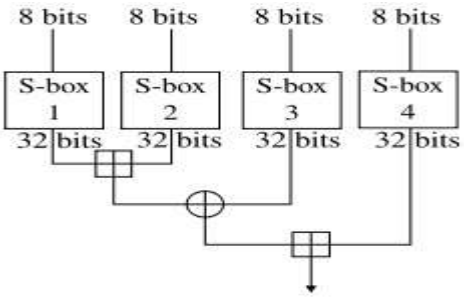

Fig. 3 : Blowfish Fiestal Structure

The illustration of following Figure 3 shows the action of Blowfish. Each line entitled to 32 bits. The step by step procedure keeps two sub key arrays: the 18-entry P-array and four 256-entry S-boxes. The S-boxes acknowledge 8-bit input and 32-bit output. Every round uses one entry of P-array, up to the final round, every half of the block is XORed with one of the two remaining unused P-entries. The figure 3 to the right shows F-function of Blowfish. The function divides the 32-bit data into four eight-bit quarters and it uses input to the S-boxes. The outputs are added modulo 232 and XORed to prepare the last 32-bit output.

The Feistel system of Blowfish algorithm is one that uses a framework that makes encryption and decryption. The Fiestal network basic working structure is shown in the figure 4; it divides each block into halves and right half becomes

new left half than the final result is new right half when the left half is XOR with the result of applying f to the right half and the key.

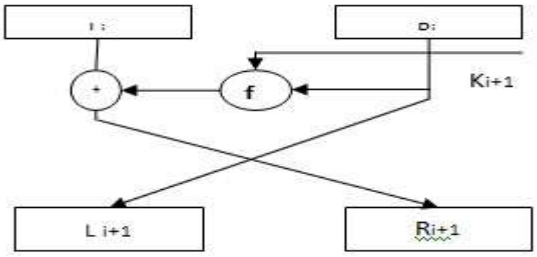

Fig. 4 : Basic Structure of Fiestel Network

Thus the proposed technique of sequential encryption using RSA and Blowfish algorithm to secure sensitive data is shown in the Figure 5. The multiple data chunks uses identity to make the indexing for efficient retrieval process.

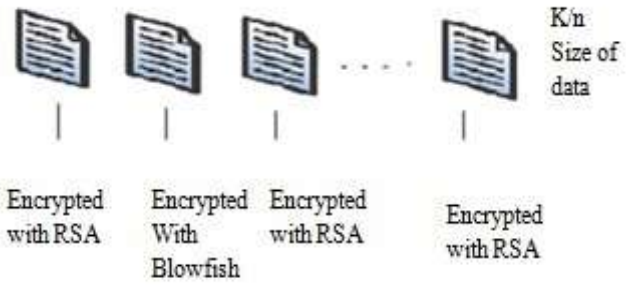

Fig. 5 : Sequential Encryption using RSA and Blowfish Algorithm.

3.3 Efficient Retrieval Scheme

Cloud environment holds hundreds and thousands of computer nodes, and they handle tasks and workloads in parallel. In this case proficient indexing structure is needed to avoid consuming time of query processing in cloud computing platform. The proposed retrieval techniques used in this paper prevents time consuming from cloud. The efficient retrieval process involves cloud Global (CG)-index. The design of CG-index must be aware of optimized form of partitioning, which avoids building a whole data-set index. The CG- index setup a local B+-tree index for data shard. B-tree is a widely used data-structure indexing technique for efficient retrieval data. In this paper it is shown that data retrieval can be performed correctly with much weaker consistency criteria among the replicated parts of the distributed B-tree. The B+-tree variant of the B-tree, is most widely used variant of the data structure.

In proposed technique, data are split into multiple data chunks, which are having its unique identity key and randomly distributed in cloud for storage. To facilitate search, each data chunks builds a B+-tree to index. The figure 6 illustrates the Cloud Global indexing strategy with B+-tree algorithm. To retrieve data from cloud using query, first locate the compute data responsible with the identity key. After locating the compute data, retrieve the indexed B+-tree data in the CG-index.

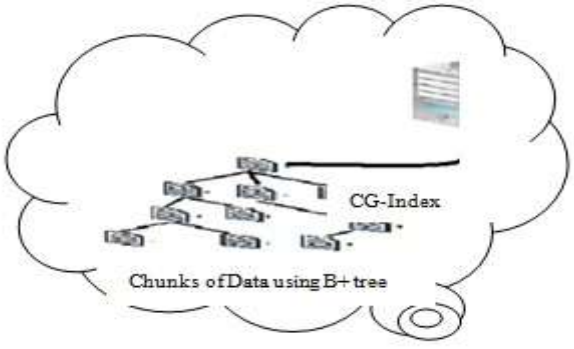

Fig. 6 : CG - Index with B+ Tree Algorithm

4. IMPLEMENTATION SETUP

In this section, the implementation setup is presented along with the final result of proposed technique and explanation is started by giving a description of the experiment. The research work will be conducted using Jboss 6.1 to create data centre and Eclipse IDE to run implementations. The proposed Blowfish and RSA encryption algorithms ensure the data security in cloud. Different key sizes are used for Blowfish and RSA algorithm. Thus, Blowfish used secret key with 448 bit size and RSA used public and private key with 1024 bit size for encryption and decryption.

Fig. 7 : Implementation of Sequential Encryption using Blowfish and RSA

The generated key for blowfish and RSA are maintained separately to access data. 1GB data is chosen for encryption; hence, it's divided into multiple chunks. The multiple chunks of data are encrypted sequentially using Blowfish and RSA algorithm as shown in figure 7. The cloud global index uses Id's of data chunks to retrieve it in correct manner and it avoids data loss or mismatch of arrangement.

5. PERFORMANCE EVALUATION

This section evaluates performance of proposed system based on encryption and decryption. It is evaluated by utilizing following encryption algorithm: DES, AES, Blowfish and RSA. These algorithms are executed in Eclipse tool. Eclipse is a integrated tool which is used to construct and deploy the applications by using java and its used in eclipse, the algorithms are run on local similar to Google app engine. The speed of encryption is measured and different algorithm accessible as standard in Eclipse, and afterward give a different aspects of those algorithms. The encryption algorithms considered are AES (with 128-bit key), DES (with 128-bit key), Blowfish (with a 256-bit key) and RSA (with a 1024 bit key). By acknowledging diverse sizes of data chunks the algorithm were assessed regarding the time needed to encrypt and decrypt the data chunks.

The Speed-Up ratio and mean processing time are also characterized. Speed-Up ratio is distinction between the mean processing time of cloud network and the single system. Speed-Up ratio helps to know how quick the information has been encrypted. It will provide the thought regarding speed of encryption. Mean processing time is the distinction between the beginning time taken to encrypt the information and the resolution time. It is also assessed both on cloud network and the single system. It is the contrast between the times taken to encrypt the data. The encryption time will increase if the size of input increases and with the increase in time speed-up ratio decreases. The Table 1 shows mean processing time of algorithm on local system and cloud network.

Table 1 : Mean Processing Time

Input	AES	DES	Blowfish	RSA
100 kb	115	75	40	60
130 kb	147	100	47	72
390 kb	210	315	82.5	126
560 kb	245	205	157	174

Although the proposed scheme yields some security that the aggregate security scores are less than other encryption algorithms, the security scores of proposed technique are much higher than other encryption AES and DES algorithm with 256-bits key size. It implies that proposed technique can give better security when contrasting and other encryption techniques. Since the basic utilization of symmetric cryptosystem is AES, DES with 256-bits key size, which is still less secure. In this way, it shows that the technique can give enough security to client to ensure their data in cloud environment.

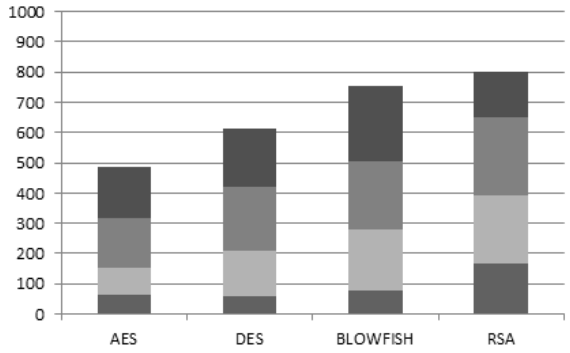

Fig. 8 : Average Security Score

All the executions were accurate to determine that the outcomes will be moderately reasonable and correct. The evaluation illustrate in Figure 8. Acknowledges algorithm and data chunks size. After an accomplishment of execution, the chunks of data are created, encrypted and decrypted. An alternate correlation is made after the successful encryption/decryption method to verify that all the chunks of data are prepared in the correct way by contrasting the produced data with the original data blocks.

The evaluation of proposed indexing scheme which concentrates on how the query functions with the assistance of record which is outsourced to the cloud server is extended. To show the viability of the system, indexing schema for the Cloud framework is developed. In this Cloud framework, every node constructs a nearby B+-tree list for its multiple data chunks. The cloud Global index is made by a portion of local B+-tree. In this framework, every node has 100k data chunks local database. The time taken to retrieve data from cloud using CG-index is illustrated. The begin of the test, the node will persistently acquire another query from the test system after it finishes its present one. The real measurements in the evaluation are redesign throughput and query throughput. To test the adaptability of the proposed technique, Cloud environment with diverse numbers of computing nodes are created.

Table 2 : Time Taken to Retrieve Data from using CG-Index

Chunk Size:100 kb	
Time/Sec	Size/MB
0.145	0,01
0.161	0.1
1.41	1
30	10

6. CONCLUSION AND FUTURE WORK

The innovative architecture proposed enhance and guarantees security of sensitive data stored in public cloud environment and have presented the design and implementation of a two strongest encryption algorithms and high-throughput indexing technique in the Cloud environment.

Thus, in this innovative work; sequentially encryption of data using Blowfish and RSA encryption algorithm is done and extended with efficient retrieval of data from cloud environment using Cloud Global (CG)-Index and assume a local B+-tree is built for the dataset stored in each compute data. To enhance the throughput of the system, compute nodes are organized and build a Cloud Global index for the Cloud. The implementation results show that proposed approach is secure for data storage and efficient for retrieval. In this work, using different encryption algorithms cause a drawback of key management. In future, key management algorithms will be implemented for different algorithm in same dataset producing results to the most efficient one in cloud.

REFERENCES

1. Aguilera M., Merchant., Shah M., Veitch A., and Karamanolis C. 2007. Sinfonia: A New Paradigm for Building Scalable Distributed Systems, In Proceeding of the 7th SOSP, Washington, 159-174.
2. Azua Himmel M., and Grossman F., 2014. Security on Distributed Systems: Cloud Security versus Traditional IT, IBM Journal of Research and Development, Vol. 58, No.3, 1-3.
3. Bojanova., and Samba A., 2011. Analysis of Cloud Computing Delivery Architecture Models, In Proceeding of the IEEE International Conference on Advanced Information Networking and Applications (WAINA), Biopolis, 453-458.
4. Cheng-Kang Chu., Sherman S., Chow M., Wen-Guey Tzeng., Jianying Zhou., and Robert H. Deng., Key-Aggregate Cryptosystem for Scalable Data Sharing in Cloud Storage, available at: http://doi.ieeecomputersociety.org/10.1109/TPDS.2013.112
5. Cheng-Chi Lee., Pei-Shan Chung., and Min-Shiang Hwang., 2013. A Survey on Attribute-based Encryption Schemes of Access Control in Cloud Environments, International Journal of Network Security, Vol.15, 231-240.
6. I-Hsun Chuang., Syuan-Hao Li., Kuan-Chieh Huang., and Yau-Hwang Kuo., 2011. An Effective Privacy Protection Scheme for Cloud Computing, In Proceeding of the 13th International Conference on Advanced Communication Technology (ICACT), Seoul, 260-265.
7. Jin Sun., Yupu Hu., and Leyou Zhang., 2013. A Key-Policy Attribute-Based Broadcast Encryption, The International Arab Journal of Information Technology, Vol.10, 444-453.
8. Luca Ferretti., Michele Colajanni., and Mirco Marchetti., 2014. Distributed, Concurrent, and Independent Access to Encrypted Cloud Databases, IEEE Transactions on Parallel and Distributed System, Vol. 25, 437-446.
9. Leena Khanna ., and Anant Jaiswal., 2013. Cloud Computing: Security Issues and Description of Encryption Based Algorithms to Overcome Them, International Journal of Advanced Research in Computer Science and Software Engineering, Vol. 3, 279-283.
10. Mandeep Kaur., and Manish Mahajan., 2013. Using Encryption Algorithms to Enhance the Data Security in Cloud Computing, International Journal of Communication and Computer Technologies, Vol.1, 56-59.
11. Michel Abdalla., Mihir Bellare., Dario Catalano., Eike Kiltz., Tadayoshi Kohno., Tanja Lange., John Malone-Lee., Gregory Neven., Pascal Paillier., and Haixia Shi., 2008. Searchable Encryption Revisited: Consistency Properties, Relation to Anonymous IBE, and Extensions, Journal of Cryptology, Vol. 21, 350-391.
12. Mather T., Kumaraswamy S., and Latif S., 2009. Cloud Security and Privacy: An Enterprise Perspective on Risks and Compliance, (O'Reilly Media, Inc. 2009).
13. Nabeel M., and Bertino E., 2012. Privacy Preserving Delegated Access Control in the Storage as a Service Model, In proceeding of the 13th IEEE International Conference on the Information Reuse and Integration (IRI), Las Vegas, 645-652.
14. Parsi Kalpana., and Sudha Singaraju., 2012. Data Security in Cloud Computing using RSA Algorithm, International Journal of Research in Computer and Communication Technology, Vol.1, 143-146.
15. Rivest R., A. Shamir., and L. Adleman., 1978. A Method for Obtaining Digital Signatures and Public-Key Cryptosystems, Communications of the ACM, Vol.21, 120-126.
16. Schneier, B., The Blowfish Encryption Algorithm, available at: https://www.schneier.com/blowfish.html
17. Shadi Abudalfa., and Mohammad Mikki., 2013. A Dynamic Linkage Clustering using KD-Tree, The International Arab Journal of Information Technology, Vol.10, 283-289.
18. Sai Wu., Dawei Jiang., Beng Chin Ooi., and KunLung Wu., Efficient B-Tree Based Indexing for Cloud Data Processing, available at: http://dx.doi.org/10.14778/1920841.1920991
19. Sahadeo Padhye., 2006. On DRSA Public Key Cryptosystem, The International Arab Journal of Information Technology, Vol.3, 334-336.
20. Trushna S Khatri., and G B Jethava., 2013. Improving Dynamic Data Integrity Verification in Cloud Computing, In proceeding of the 4th International Conference on the Computing, Communications and Networking Technologies (ICCCNT),1-6.