RESEARCH ARTICLE OPEN ACCESS

Cloud Security by overcoming issues of Bloom filter through Deduplication and DNA encryption

Sangeeta Kumari [1], Harjinder Kaur [2]
Mtech Scholar [1], Assistant Professor and Hod [2]
SSIET – Dinanagar

ABSTRACT
Cloud provides resources to the user on pay per use basis. Although it appears that cloud possess infinite resources but due to mass community of users using recourses offered by cloud hence resources become limited. In addition, intention of the users within cloud is uncertain. Thus, there can be attacking users who can corrupt the resources provided by Cloud service provider. For this purpose, encryption strategies along with deduplication can be used. This means same file is not allowed to upload on the cloud. In addition, the proposed system also uses DNA encryption-based mechanism to provide extra security while transmitting the data towards destination nodes. The results is expressed in the form of encryption time, decryption time, total execution time, throughput and reliability. Almost all the metric shows improvement in terms of the specified metrics.
Keywords: Cloud, DNA encryption, Throughput, reliability, execution time

I. INTRODUCTION

Cloud computing provides resources to clients. The clients and service providers are bound by the service level agreements. Clients as well as service providers cannot violates the service level agreements. As the users will be of uncertain intention, they can corrupt the resources provided by the cloud. Thus, some security procedure must in place to tackle the issue of security procedure is resolved through bloom filter (Geneiatakis et al., 2009). Bloom filters calculate the amount of data transferred from source towards destination(Changsong et al., 2020). Bloom filter will perform the task of cycle stealing which mean word counter will reduce by one as the word is transmitted. As the word counter becomes zero, control will be taken back and given to the next user in sequence.

Figure 1: Cloud services and security

The cloud computing ensures that that user can access the resources provided by the cloud without causing too much pressure of buying the resources. The security procedure in terms of deduplication and DNA encryption proposed through the proposed system can provide better security procedure as compared to procedure with bloom filter. The security procedure along with DNA encryption is described in the section 3.

II. LITERATURE SURVEY

This section presents the encryption mechanism that is used to secure cloud. The security of cloud computing is considered at distinct levels. At first level, we will consider least secure mechanisms and continue towards more secure mechanisms of security toward cloud computing.

2.1 Level 1 Security mechanisms

Level1 security mechanisms include security systems that can be further improved or in other-words are least secure. Comparison table 1 provide insight into the level1 security.

Technique	Security metrics	Merits	Demerits	Considered in
RSA	Key Size Execution time	Key that is formed depends upon random prime numbers. Encryption using random mechanisms are more secured.	Public Key formed can be hacked easily that is transmitted over the public medium.	(Rani & Sagar, 2018)
DES	Key Size	Key size is small and is a building block for most of secure encryption standards in modern era	Encryption standards are poor using 56 bit key size	(Shaukat & Hassan, 2017)
Homomorphic Encryption	Key Size Execution time	Key size is reduced but complex by the use of random number generator	Encryption standards uses both public as well as private key mechanisms that makes entire mechanism complex	(Singh et al., 2015)

Table 1: Level 1 security standards

Random number generator is missing in almost all the studied literature given in table 1. In addition, security of data is at stakes by the use of public key mechanisms.

2.2 Level2 Security Mechanisms

Level 2 security is advanced as compared to level 1 security levels. The random number generator is accommodated within most of the mechanisms to provide security among encryption mechanisms.

Considered in	Technique	Description	Advantage	Disadvantage
(Song et al., 2015)	Dynamic DNA coding	It firstly generate the Chaotic sequence of the image pixels. Then it uses DNA encoding to encrypt the image pixels .	It gives better security to image and can effectively resist statistical, shear, and other attack,	Reliability is low
(Miguel, 2015)	BCD based coding	In this the BCD code is used to encode the data and the key is used once. It also used 1's complement and 2's complement for encoding.	It gives efficient and performs well for text crypto. Throughput of system is high	CPU time is more

(Andem, 2003)	RSA based	In this firstly image is input and then sum of the ASCII values of each character of the string input by the user is stored as x. It selects prime number as encryption key.	Gives better cyber security	It is very computationally intensive and time consuming
(Khan et al., 2019)	Playfair algorithm using XOR	It converts image pixels into matrix and then calculated key value using XOR	It makes it hard for the attacker to perform a frequency analysis based on the used pixel digraphs	Complexity is high
(Awad et al., 2018)	Hamming code along with arithmetic operation	In this firstly checked the received data is error free or not using hamming code and then subtraction is done to find encrypted.	Security is high	Time consuming process

Table 2: Level 2 Encryption mechanisms

These mechanisms are more secured as compared to strategies without random number generator. The mechanism employed could be further enhanced by employing strategies to tackle the issue regarding key generation.

2.3 Level 3 Security mechanism

Level 3 security mechanism are more secure as compared other two levels. To manage the abnormalities within the dataset, pre-processing mechanisms are employed. In addition to manage the storage, deduplication is also utilized. The mechanisms ate discussed in table 3.

Considered in	Technique	Description	Merit	Demerit
(Zhao & Chow, 2017)	Proactive Fault tolerance for cloud security	Problem is tackled before the occurrence of problem	Pre-processing mechanism employed tackled the issue of cloudlet execution within prescribed time interval	Overhead in determining the problem is extremely high
(Gupta et al., 2021)	Homomorphic encryption	This encryption determine complexity of key and space conservation is applied	Less space but more complex key	Space conservation may cause additional overhead
(Li et al., 2016)	Deduplication with RSA encryption	This encryption is performed at server side and encryption along with space conservation is employed	Complex operation is simplified by accommodating space conservation along with random key generation	Essential attributes can be lost using this mechanism
(Hussain, 2015; Seo et al., 2016)	Security of cloud storage using BLOOM FILTER approach	This approach is of encryption based on DNA with high security	Complex key generation is included	Collision problem is not tackled

Table 3: Level 3 Security mechanisms for security of cloud

There is **problem of collision** within discussed literature. Collision resolution can be resolved by the use of chaining in proposed system. The problem discovered in BLOOM FILTER approach that is discovered to be most secured than rest of the literature is given in table 4

FEATURE	QUANTITY AND DESCRIPTION	PROBLEM
Number of phases	5 • Uploading • File Checking • Encryption • Downloading • Decryption	Phases causes execution time to increase in case uploaded file is large in size
Encryption	One algorithm BDNA	Division method employed within DNA encryption generate key that is prone to collision
Key Size	1 key with 32 bits	Key size can be extended to 64 bits for increasing complexity
Execution time	It is a metric defining least time for key generation	Duplicate contents within file could cause high execution time during translation
Future Enhancement	Additional phases with duplicate content handling and collision detection	----

Table 4: BDNA problems

The proposed system accommodates BDNA with excess 3 code rather than binary codes.

III. Cloud Security with Deduplication and DNA encryption

Security of cloud is critical and to accomplish the security process we need to ensure same contents are not uploaded on cloud again and again. Cloud deduplication process indicates that same contents cannot be uploaded again and again by the user within the cloud. The deduplication process encounters the mild security procedure as well but that is not that effective in establishing high form of security. Existing system uses bloom filter that is capable of ensuring only countable contents can be transmitted within the cloud but sometimes multiple users may required to transmit large amount of contents over the cloud. The proposed system working is described as under

3.1 Deduplication

With deduplication we means that proposed system does not allow multiple data to be uploaded on the cloud again and again. The block level deduplication mechanism is used in this case. The repeated content within the block level deduplication will be replaced with the index number of the word. This mechanism will reduce the length and complexity of overall operation.

3.2 DNA Encryption

The problem of collision is significant in BDNA encryption along with binary encoding operation. This problem can be rectified using chaining and excess 3 encryption. The overall procedure of proposed Binary encoding scheme employed in BDAN approach is less secure since plane encoding can simply be hacked and security is at stake. To resolve the issue binary encoding scheme is further made secured using excess 3 encoding. The length of key is increased and hence security is also enhanced. system starts with initialization of queue that will store the generated key.

The proposed model converts the generated key with BDNA approach into more secured form by the application of excess 3 encoding scheme. The model also incorporated pre-processing by handling collision at early stage and hence reliable key generated in least execution time.

The proposed model works in phases. Pre-processing mechanism is considered that convert the data file into normalized form by eliminating collision from generated keys. The procedure that is used for collision resolution is known as chaining. After the collision is resolved, excess 3 mechanisms is applied on the generated key though the DNA approach. In addition, Binary code mechanism is replaced with excess 3 code in proposed approach to enhance the security further. The worth of proposed system is proved in the next section considering different parameters.

3.3 Proposed system flow

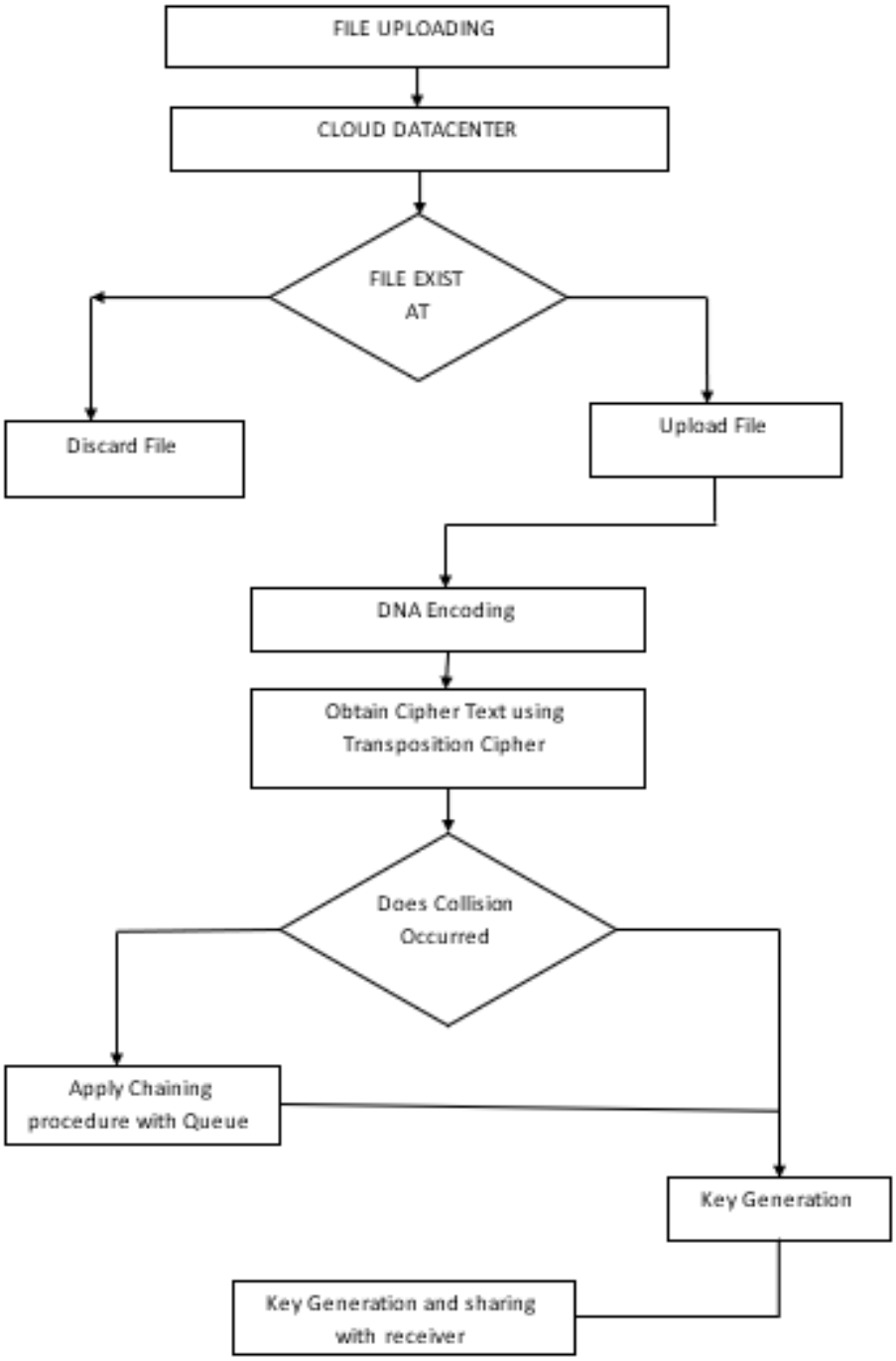

Figure 2: Proposed System

IV. PERFORMANCE ANALYSIS AND RESULT

The proposed system makes the changes in the BDNA approach at two distinct levels. First, key generation is modified by accommodating chaining with queue. In addition, binary codes at encryption phase is replaced with the excess 3 mechanism to ensure more secured encryption as compared to binary encoding scheme. The result is given in terms of execution time and key length. The size of the file being uploaded is varied and result is obtained. The result is given within the table 6 and table 7.

File length(Bytes)	Bloom Filter(ms)	Proposed DNA(ms)
1000	10	9
2000	19	11
3000	28	15.5
4000	39.98	18.98
5000	45.63	21.45

Table 5: Execution time comparison of BLOOM FILTER and Proposed approach
The execution time with different file size with proposed approach is minimized. The result improvement of nearly 10% is observed that Is significant and plots for the same is given in figure

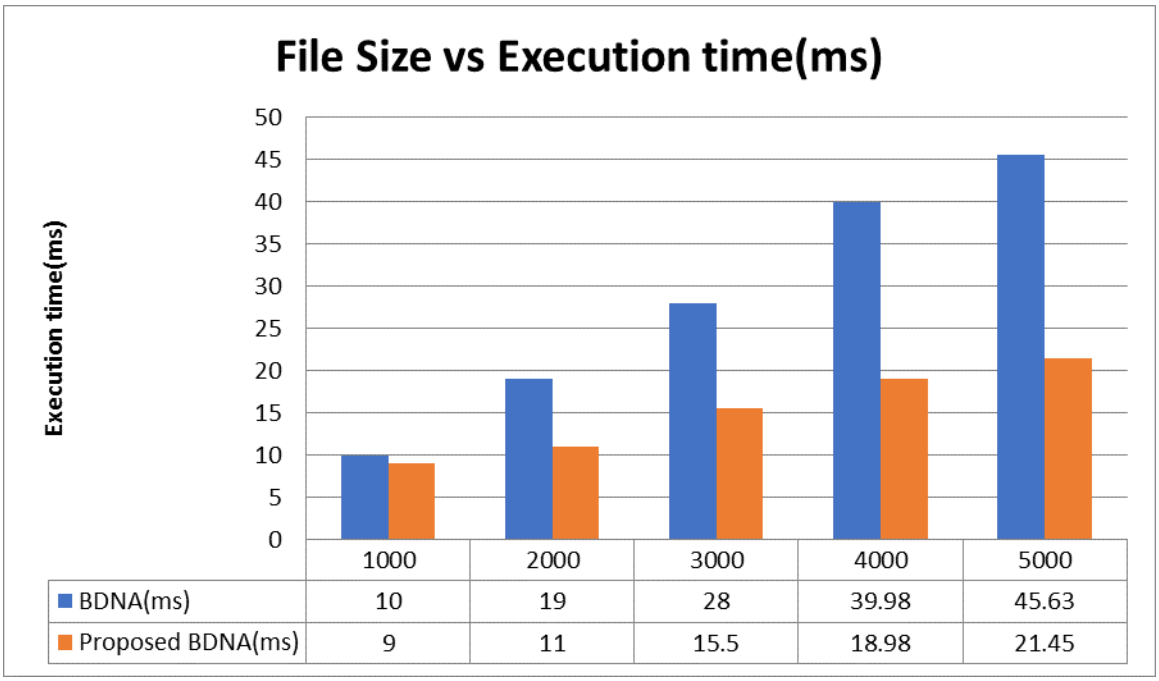

Figure 3: Execution time comparison with BLOOM FILTER and proposed approach
The complexity of key is also enhanced by the use of excess 3 code and Channing mechanism. Key complexity is increased significantly that is demonstrated with the table 6.

File length(Bytes)	BLOOM FILTER(bits)	Proposed BDNA(bits)
1000	14	64
2000	14	64
3000	32	128
4000	32	128
5000	32	256

Table 6: Key size with different file size
The key length in case of BLOOM FILTER is maximum of 32 bits. The proposed model is based upon larger key size since multiple data entries are contained within same index values. This is also demonstrated within figure 4

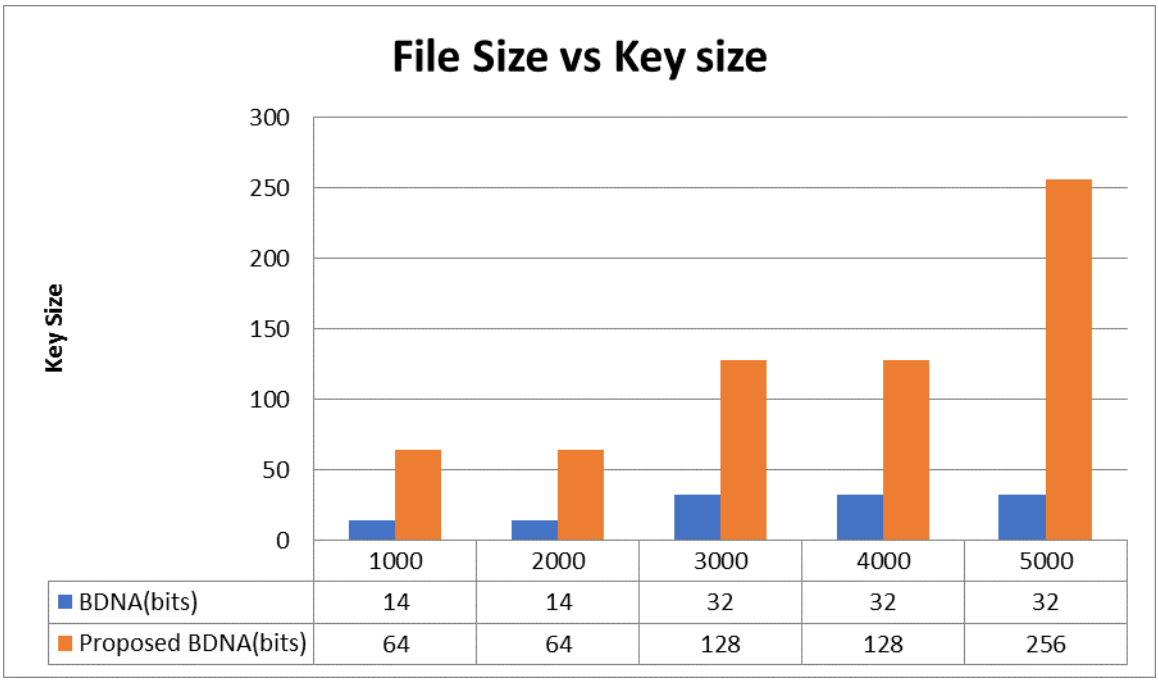

Figure 4: Key size for BLOOM FILTER and proposed approach

The results indicates that by accommodating chaining and excess 3 mechanism security is enhanced and result in terms of execution time is improved.

The encryption time and decryption time is next parameter that is compared to determine the validity of result. The result is reflected within table 7

File length(Bytes)	Bloom Filter(ms) Encryption and Decryption time	Proposed DNA(ms) encryption and decryption time
1000	120	90
2000	198	112
3000	288	150.5
4000	390.98	188.98
5000	405.63	201.45

Table 7: Encryption and Decryption time

The reliability corresponding to the proposed approach is also better as compared to the existing bloom filter. The reliability is given within the table 8

File length(Bytes)	Bloom Filter	Proposed DNA
1000	6	9
2000	6	8
3000	5	9
4000	4	8
5000	3	8

Table 8: Comparison of reliability

The reliability in the existing system decreases with file size. In the proposed system however, result does not vary much with the increase in file size.

V. CONCLUSION AND FUTURE SCOPE

The DNA based encryptions is most secured that is based upon human DNA structure. The DNA

based encryption although is secured but prone to collision problem. The problem exists as division method is employed in key generation. This means distinct data item can yield same keys that results in collision. To tackle the issue, queue is placed at the encryption end. In case same key is generated that must be stored at same index value is placed within queue. In addition to improve the security, excess 3 code is accommodated in BDNA rather than binary coding scheme. This scheme ensures that more secured key is generated that has increased length as compared to binary encoding scheme. The result due to least collision is improved in terms of execution time and complexity. Key becomes more complex and execution time is improved by the margin of 10%.

In future, security mechanism of BDNA with excess 3 code can be demonstrated within ECG signals. The application is most suitable within Internet of Things with medical provisioning application.

REFERENCES

Andem, V. (2003). *A cryptanalysis of the tiny encryption algorithm*. University of Alabama.

Awad, A., Matthews, A., Qiao, Y., & Lee, B. (2018). Chaotic Searchable Encryption for Mobile Cloud Storage. *IEEE Transactions on Cloud Computing*, *6*(2), 440–452. https://doi.org/10.1109/TCC.2015.2511747

Changsong, Y., Changsong, Y., Xiaoling, T., Feng, Z., & Yong, W. (2020). Secure data transfer and deletion from counting bloom filter in cloud computing. *Chinese Journal of Electronics*, *29*(2), 273–280. https://doi.org/10.1049/cje.2020.02.015

Geneiatakis, D., Vrakas, N., &Lambrinoudakis, C. (2009). Utilizing bloom filters for detecting flooding attacks against SIP based services. *Computers and Security*, *28*(7), 578–591. https://doi.org/10.1016/j.cose.2009.04.007

Gupta, S., Jain, S., & Agarwal, M. (2021). Ensuring Data Security in Databases Using Format Preserving Encryption. *Proceedings of the 8th International Conference Confluence 2018 on Cloud Computing, Data Science and Engineering, Confluence 2018*, 214–218. https://doi.org/10.1109/CONFLUENCE.2018.8442626

Hussain, A. K. (2015). A Modified RSA Algorithm for Security Enhancement and Redundant Messages Elimination Using K-Nearest Neighbor Algorithm. *IEEE Access*, *2*(1), 159–163.

Khan, S. A., Aggarwal, R. K., & Kulkarni, S. (2019). Encryption Schemes of Cloud Computing: A Review. *2019 5th International Conference on Advanced Computing and Communication Systems, ICACCS 2019*, 23–26. https://doi.org/10.1109/ICACCS.2019.8728313

Li, J., Lin, X., Zhang, Y., & Han, J. (2016). KSF-OABE : Outsourced Attribute-Based Encryption with Keyword Search Function for Cloud Storage. *IEEE*, *1374*(c), 1–12. https://doi.org/10.1109/TSC.2016.2542813

Miguel, R. (2015). HEDup: Secure Deduplication with Homomorphic Encryption. *2015 IEEE International Conference on Networking, Architecture and Storage (NAS)*, 215–223. https://doi.org/10.1109/NAS.2015.7255226

Rani, K., & Sagar, R. K. (2018). Enhanced data storage security in cloud environment using encryption, compression and splitting technique. *2nd International Conference on Telecommunication and Networks, TEL-NET 2017*, *2018-Janua*, 1–5. https://doi.org/10.1109/TEL-NET.2017.8343557

Seo, H., Liu, Z., Großschädl, J., & Kim, H. (2016). Efficient arithmetic on ARM-NEON and its application for high-speed RSA implementation. *Security and Communication Networks*, *9*(18), 5401–5411. https://doi.org/10.1002/sec.1706

Shaukat, K., & Hassan, M. U. (2017). Cloud computing security using encryption technique. *Transylvanian Review*, *25*(12), 74–82.

Singh, J. P., Mamta, & Kumar, S. (2015). Authentication and encryption in Cloud Computing. *2015 International Conference on Smart Technologies and Management for Computing, Communication, Controls, Energy and Materials, ICSTM 2015 - Proceedings*, *May*, 216–219. https://doi.org/10.1109/ICSTM.2015.7225417

Song, C., Park, Y., Gao, J., Nanduri, S. K., & Zegers, W. (2015). *Favored Encryption Techniques for Cloud Storage*. 267–274. https://doi.org/10.1109/BigDataService.2015.75

Zhao, Y., & Chow, S. S. M. (2017). Updatable Block-Level Message-Locked Encryption. *ACM*, 449–460.

Satish, Karuturi S R V, and M Swamy Das. "Multi-Tier Authentication Scheme to Enhance Security in Cloud Computing." IJRAR (International Journal of Research and Analytical Reviews) 6, no. 2 (2019): 1-8, 2019.