DNA Based Hybrid Approach for Data Integrity on Integrated Cloud Based Smart Environment: An Analysis

Prof. Sunil Raj Y[1], Prof. Lucase L[2], Mrs. E. Helen Parimala[3]

[1,2]Assistant Professor, [3]Research Scholar
Department of Computer Science,
St. Joseph's College (Autonomous), Trichy, India

Abstract: **Cloud computing is outsourcing IT's costly infrastructures over the web. Increasing the convenience of IT users, one of its major contributions is storage. Though storage's over the internet make cost efficiency in IT tasks, there exist risks. Fixing the risks relies on the bed of the algorithms that can also keep cloud storage safe and secure. Few security issues that threaten cloud include, theft of data, Integrity of data, data availability, and security at vendor level. As algorithms could determine the complexity and execution time, it also has the property of identifying the efficiency of the system. As the algorithm is presently used highly in the security of data and storage and there is a lack of data integrity in the cloud environment, this paper is intended to make a detailed study on the existing algorithms used to enhance the security of the data on storage. The study is conducted in various dimensions to explore the facts, such as level of security, efficiency, and speed. The algorithms considered for the study include symmetric, Asymmetric, hashing, and hybrid including DNA cryptography. It is believed that, the work would be helpful for the researchers intending to work on securing data at rest.**

***Index Terms*: Cloud Storage, Data Integrity, SHA256, RSA, ECC, Blowfish, DNA, RIPEMD160**

1. INTRODUCTION

Cloud computing (CC) is acting as a backbone for the survival of every service over the globe. It pretends that its absence could largely affect the existence of the business. Support it provides its followers includes IaaS, SaaS, and PaaS at a low cost on time. One of the most essential services provided is storage, considered mostly as an infrastructure.

The existence of IoT has an increased amount of data generated. This data will be directed to storage as it is flexible to scale the storage when the data size increases. Here the issue is data thus stored is on the internet this leads to the security of data [1]. Major security risks found in CC could be categorized as storage, access control and identity risks, contractual and legal issues.

To enhance the security of storage, it is essential to deal with the issues such as risks on data at rest and data in transit. Data is transmitted through Transport Layer Security may lead to risks like data lineage and provenance. As data is accessed available publicly [7], data leakage could cause data issues.

To overcome risks on data at rest, cryptographic measures are essential to achieve privacy [22]. Various risks include, i) Recovery of data from the damage of storage and deletion. ii) sanitizing of data, iii) Data Backup, iv) Data isolation v) Data segregation, vi) Data Lock-in and vii) Data Location.

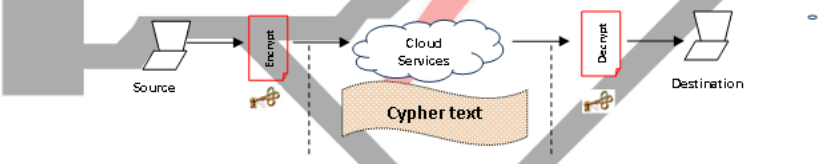

Fig. 1. Basic Crypto System

Certain other security risks include the Integrity of data which allows modification by authorized users alone. Next is data provenance includes computational accuracy along with integrity. As integrity is the most essential parameter for the data at rest, it is possible to achieve using cryptographic algorithms. A better algorithm could keep the data safe from threats by hiding it from intruders. The hashing technique can be used to convert the given message into a fixed-length unique string of characters and it is hard to retrieve back the original digital message. Crypto algorithms can be classified as Public-Key Encryption, Identity-Based Encryption, Attribute-Based Encryption, and Functional Encryption.

The advantages of DNA cryptography have maximized the use of it to a larger extent in the CoT world. DNA based algorithms have the potential of storing huge amount of data generated by the move of IoT enabled devices [9]. DNA cryptography uses modern biotechnology as a measure to transfer ciphertext into plaintext [10].

The paper is organized as follows: section 2 reviews of literature, Section 3 provides a detailed study on the algorithms, Section 4 presenting the comparative analysis and Section 5 concludes with the results after the analysis.

2. REVIEW OF LITERATURE

Data masking is best for enhancing the security of storage on cloud. Major technique is categorized as static and dynamic. The techniques include shuffling, substitution, random substitution, number, nulling, and encryption [7].

The encryption algorithms studied by [11] include symmetric, Asymmetric, and hashing algorithms. The symmetric algorithms considered are DES, 3DES, AES, and Blowfish. Asymmetric algorithms such as RSA, Elgamal, and Diffi-hellmen algorithms and MD5, SHA hashing algorithms were included in the analysis. Parameters used were key size, security rate, throughput, block size,

scalability, speed, and encryption/decryption. It was inferred that the blowfish algorithm was efficient among the algorithms selected for the study.

[14] have introduced a role-based encryption mechanism for securing the EAR. This provides access to the storage based on the role in a flexible manner. Authors have used ECC symmetric encryption algorithm with a 160-bit key length along with ISAAC [22] symmetric encryption algorithm.

[19] have proposed a storage authentication and data encryption schemes for protecting the data storage. AES cryptographic algorithm is used for encryption. The validity of the user is verified to deny unauthorized access. Authors have focused on enhancing the privacy of data, avoid data loss, and enhance the availability of data, thus enhancing the security of remote storage.

In [3] authors have proposed an algorithm for enhancing security by increasing the complexity in key generation. Authors also have suggested using double security by the RSA algorithm could enhance the security of data, which protects the data from security threats.

[9], discusses the symmetric and asymmetric algorithms for enhancing security in cloud, while introducing a new approach of public key cryptosystem. Enhancing the RSA algorithm authors have obtained the better result while improving the security concerns in cloud.

[13] Have proposed an auditing technique in order to overcome the existing security threats. Here author have suggested continuous auditing technique along with the creation of signature for blocks using MD5 algorithm. Encryption of algorithm is done using Base64 algorithm. In order to verify the data integrity, author proposes, two step checking process they are block and file check.

The efficiency of security algorithms have been compared by [16], to provide solution for the existing security threats in cloud. Authors have done a study on the known algorithms such as RSA, DES and AES. They have suggested that homomorphic algorithms may best suit the cloud environment for enabling higher level of security.

[26] have done a Bi-Serial DNA Encryption Algorithm convert text message into hexadecimal and binary code. Message is split in two parts, one is used as key and other as message. Adding XOR operation increase the compression factor. This increases the security, while increasing the computational complexity as it uses two prime numbers for amplification.

Algorithm proposed by [27] uses DNA coding and Binary (OTP) Scheme. This algorithm uses keys; i) for encryption on the sender side and ii) decryption on the receiver side. One is a random string of nucleotides forming a DNA Sequence and the length depends on the length of the plaintext. The other key is a Binary Sequence that is used for OTP. Length of binary key is twice that of the DNA sequence key [27].

[28] put forward an idea using Pseudo DNA Cryptography. The approach adopted is to convert the data using genetic code table and key send to receiver via a secure channel. The length of cipher text is higher than plaintext and information that exists after encryption can be compromised easily [27].

[29] have proposed converting the plaintext into its ASCII code then converted into binary form. These are encoded in to sequences and one of the sequences is selected as a key and grouped in blocks of 8 characters each. With respect to positions of characters in key, table is created and with the help of table and key, data gets converted in the encrypted form [29].

[30] One of the most emerging techniques in the world of cryptography is DNA cryptography which works on the concepts of DNA computing. Multiple DNA algorithms have been studied along with symmetric and asymmetric algorithms that are using DNA system. Authors used a blend of BREA and DNA algorithm for guaranteeing the security.

In [2] authors have proposed a hybrid encryption technique, where homographic encryption and blowfish are the participants making the technique unique. The flexibility of homographic algorithm and security measures of blowfish algorithm combines together to make the storage more efficient.

[4], have proposed a model for enhancing the security of data in cloud storage, while introducing a proxy re-encryption mechanism for the same. The mechanism stands unique as it could transform cipher text of IBE to PKE type text. The user can share private data in more secure manner. Other issue such as performance and dynamic data privacy protection is to be handled.

[34] In order to enhance the integrity of data being transferred the author have used AES along with RSA for key distribution and have encoded with DNA crypto algorithm. Whereas a message digest have been generated using SHA 256 hashing algorithm. The message digest confirms the integrity of data being transferred over the network.

[8] have suggested an auditing technique using TPA for verifying the integrity of data. The proposel makes use of AES for verification of data and Secure Hashing Algorithm (SHA - 2) for generating verification metadata and message digest for checking integrity.

[21] have proposed architecture to ensure the integrity of the query results. This uses Counting Bloom Filters to generate proof for users and also proposes a hybrid model for guarantee the search efficiency.

[20] have introduced an architecture for inter-cloud data transfer based on cryptographic algorithm. The technique proposed uses a two phase encryption/ decryption for file upload and download, i.e., encrypting the data using AES algorithm and then encrypting the key using Elgamal algorithm.

3. SECURITY ALGORITHMS – Data Integrity

The term Cryptography means ‘hidden secret’, often involved in ensuring the data security. This includes integrity, non-repudiation, confidentiality and authentication. Applications of cryptography include authorization, commerce over online, and more [5]. Cryptography was effectively used in encryption and the encrypted message is decoded to the selected recipients. Modern cryptology is centered largely on mathematical theories and computer science.

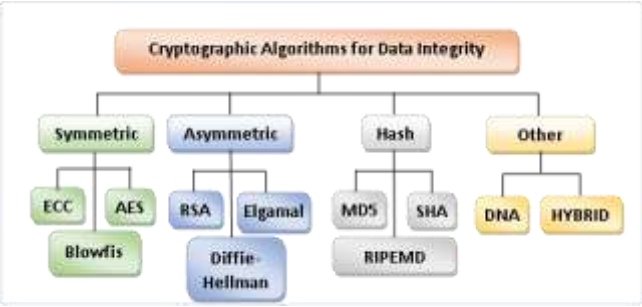

Fig. 2. Crypto Algorithms that can be used for handling data integrity

Therefore, computational firmness may be used in algorithms which makes the algorithms unbreakable their by enhancing the data security. Algorithms that exists are large in number. A few randomly chosen algorithms on broader category are considered for the study. The algorithms chosen are presented below as in the Fig 2.

3.1 Symmetric Algorithms

Symmetric algorithms work under a public key. Three algorithms were chosen randomly for the study. The algorithms chosen are ECC, AES and Blowfish.

3.1.1. ECC algorithm

[32] Elliptical curve cryptography (ECC) algorithm is a public key based technique that could create smaller and efficient keys faster. Key is generated using properties of elliptic curve equation producing a large prime number. Since ECC helps maintaining security with low computing power and battery usage, it is widely used in mobile applications. Following is an algorithm considered for analysis.

Algorithm: ECC
Begin
k1 ← 0; k2 ← 0; v1 ← 0; T ← Y;
while Y > 0 do
    if T = Y then
        k2 ← X;
    else
        k2 ← 2X;
    end if;
    if y0 = 1 then
        v1 ← k1 + k2;
    else
        v1 ← k1;
    end if;
    k1 ← v1 mod p; X ← k2 mod p;
    Y ← '0' || Y (n − 1 downto 1);
end while;
return Z;

3.1.2. AES cryptographic algorithm

[33] Advanced Encryption Standard (AES) is one of the common and widely used symmetric algorithm. It has its own structure to encrypt and decrypt data. It is also applied in hardware and software. Such nature of AES algorithm makes it difficult to get the data. This is flexible to work with different magnitudes of keys such as 256, 192 and 128 bit. It allows predefined rounds of 10 for 128-bit keys, 12 for 192-bit keys and 14 for 256-bit keys. All of these rounds uses a different 128-bit round key that is used to calculate from the original AES key.

Encryption procedure using AES, it consisted of four different steps as stated below

i) Bytes substitution
ii) Shift row
iii) Mix columns
iv) Add round key the final step consisted of exclusive-OR (XOR) of the output for the first three steps with key schedule of four word.

The steps of decryption are: Each round consists of the four processes conducted in the reverse order:

i) Add round key
ii) Mix columns
iii) Shift rows
iv) Byte substitution

After decryption the user will able to access the original data or file.

3.1.3 Blowfish

[31] Blowfish is a common public encryption algorithms which allows variable length key. This algorithm can be habituated to both hardware and software based applications. It undergoes pain of weak keys, yet no attack is recognized to be prolific against it.

Key generation

Key generation is necessary, it should be computed before entering the encryption process. Once key is generated the encryption algorithm can be triggered.

Algorithm: Blowfish Encryption
Divide x into two 32-bit halves: xL, xR
For i = 1to 16:
xL = XL XOR Pi
xR = F(XL) XOR xR
Swap XL and xR
Swap XL and xR (Undo the last swap.)
xR = xR XOR P17
xL = xL XOR P18
Recombine xL and xR
End

This introductions to each algorithm are to provide the minimum information to distinguish the main differences between them.

3.2 Asymmetric Algorithms

The process of double encryption and decryption significantly increase the security of the RSA algorithm. The study have included various algorithms such as RSA, Elgamal and Diffie-Hellman. The encryption and decryption mechanisms followed are discussed here below.

3.2.1. RSA algorithm

The usage of more than 2 keys by the suggested system upsurges the RSA algorithm security. Thus, making the algorithm further intricate for the one who try to access data by decoding. Digital signatures and key exchange both can use the RSA algorithm. RSA practices conventionally accelerative mathematics while using numbers with hundreds of digits. [24] RSA based private and public key is created by subsequent steps:

Algorithm: Encryption
a. *Sender has to attain the public key*
b. *Signify the plain message by a positive figure.*
c. *C = Me Mod n*
d. *Sending the C cipher text*
End
Algorithm: Decryption
Private key-> d
a. M= Cd Mod n
b. M is extracted for plain text.
End

3.2.2. Elgamal

Elgamal encryption could be looked off as the extension of Diffie-Hellman (DH) algorithm. This algorithm aimed at reducing the complexity of problems namely, DH problem and discrete logarithm. Here homomorphic multiplication is done for encrypting the data.

Algorithm: Key Generation
i. *q, a large prime number and α, primitive root of q*
a. *Choose X, a random integer*
ii. *Compute Y = αx mod q*
iii. *Private Key: X, Public Key {q, α , Y}*
End
Algorithm: Encryption
i. *Represent message*
ii. *Choose a random integer k*
iii. *Compute one-time key, K=Yk mod q*
iv. *Encrypt message 'm' as a pair of integers (C1, C2) where C1 = αk mod q, C2 = K, m mod q*
End
Algorithm: Decryption
i. *Recover key by computing K= C1x mod q*
ii. *The message, m can be retrieved as*
iii. *m=C2, K-1 mod q*
End

The algorithms for both encryption and decryption scheme are depicted here, also algorithm for key generation is included as it enhances the key strength.

3.2.3. Diffie–Hellman

This is a public key system, largely used in key distribution. Its ability in hiding the key in secured way is required by the IoT environment to have an integrated security. As security in Cloud and IoT integrated environment is very minimum, security threats still exists. The algorithm is as follows:

Algorithm
Begin
p (prime) and g (primitive root)
A = ga mod p
B = gb mod p
s = Ba mod p
s = Ab mod p
End

Being the first algorithm with public key scheme this is still a performing algorithm in the public cloud environment.

3.3 Hashing Algorithms

Cryptographic hashing algorithms could map messages of random length to series of fixed length string. The algorithms considered in this study are described as follows:

3.3.1. MD5

Message Digest algorithm is cryptographically strong, 128-bit hash algorithm. A successful collision attack against the MD5, hasn't resulted in practical weaknesses.

Algorithm
Begin
i. *Padding bits and Append Length Padding of the bits with '0' and '1' first and last respectively*
ii. *Divide the input into 512-bit blocks*
iii. *Initialize Channing variables Initialization of 32-bit number in the form of chaining variables*
iv. *Process blocks: four buffers are joined with input words, 4 rounds are performed (involving 16 basic operations)*
v. *Hashed Output: There are 4 rounds performed which is of 128 bits.*
End

3.3.2. SHA

Secure Hash Algorithm was designed for using it with Digital Signature Standard. It produces 160-bit hashes and have no known attacks against it. It is stronger for it has longer hash value.

Algorithm: SHA
Begin
1) Initialize random strings of hex characters
2) Append a 1 with actual message, followed by enough 0s until the message is 448 bits.
3) Add the length of the message (64 bits), therefore 512 bits is the resultant message
4) The padded input, is then divided into 512-bit chunks, and each chunk is further divided into sixteen 32-bit words
5) For each chunk, begin the 80 iterations, necessary for hashing, and execute.
End

3.3.3. RIPEMD-160

RIPE Message Digest (160 bits) is expanded as RACE Integrity Primitives Evaluation, which was the program for Research and Development in Advanced Communications Technologies.

Algorithm:
Begin
i. *Operations in one step.*
ii. *Ordering of the message words.*
iii. *Boolean functions.*
iv. *Shifts.*
v. *Constants.*
End

3.4 DNA Algorithm

DNA cryptography could help process, store and transfer data in most secure way. DNA can be classified as two types, they are: i) nDNA that allow sequencing genome region which do not encode proteins. ii) cDNA allows sequencing in regions of genome which encode proteins. In cDNA modification of a sequence to embed information must preserve its translation to proteins.

Algorithm: Encryption
Input: p, k, Output: dc
Begin
i. *Text: 'p', Key: 'k'.*
ii. *Convert the original text into binary text 'b'.*
iii. *Now covert binary text into corresponding DNA base pair (amino acid group).*

iv. Now we finally found the DNA cipher 'dc'.

End

Data received at the receivers end will be unreadable so decrypt the formatted data. This is converting retrieve plain text from the cipher text. Following steps can be used to decrypt the data.

Algorithm: Decryption

Input: dc, k, Output: P

Begin

i. Take the cipher text 'dc'.

ii. Now use key 'k' that provide the time of encryption.

iii. Convert DNA cipher 'dc' into binary text 'b'.

iv. Binary text is converted into original text 'p'.

v. Finally original message is received.

End

These are the algorithms considered for studies which are commonly known to be useful securing the IoT and Cloud architecture. Very specifically such cryptosystems could provide more security to the data stored on the cloud. It is intended that such algorithms could add more strength to the CSP's involved in providing better services by lending their remote storage for the unknown remote user.

4. RESULT AND ANALYSIS

As several amount of algorithms exists, comparison on those algorithms is very essential in order to strengthen the security in Cloud Storage. Here a comparison is made on a broader category to make clear decision on the problems and strengths of existing algorithms. The study made on the algorithms is conducted as in table 1.

4.1. Performance & Efficiency

The comparison of the strength of each and every algorithm in Table 1, shows that every algorithms are capable to perform in different levels with different parameters.

Table -1: Study on the Characteristics of Algorithms

Type	Algorithm	Characteristics
Symmetric	ECC	Obtain security on Low computing power
	AES	Applied to hardware and software
	Blowfish	Optimized to hardware applications
Asymmetric	RSA	Digital Signatures and Key exchange
	Elgamal	Reducing the complexity by homomorphic multiplication
	Diffie-Hellman	exchanging cryptographic keys
Hash	MD5	Faster
	SHA256	Secure, No Attacks reported
	RIPEMD-160	Data integrity
Medical Encoding	DNA	Speed, Minimal Power and Storage Required

It shows that ECC could perform well even when the computing power is less. Whereas AES and Blowfish have the ability of functioning with both hardware and software's. As their capability these can be employed in IoT environment and also in Cloud environment to ensure the security in device level, may be at both mobile devices and high performing devices.

As the asymmetric algorithms have their part in keeping the privacy even in the key generated, RSA and Diffie Hellman are employed in key exchange while Elgamal could perform well by the simple homomorphic multiplication. These could help securing the data form eavesdroppers by hiding the key used for revealing the actual data. Therefore these can be used in Cloud based IoT environment for protecting the data generated by the devices.

Now the master peace of the Cloud Service Providers would be hashing algorithms, the algorithms chosen for the study are MD5, SHA256 and RIPEMD. Among this all are generating a digital signature for the given input message. Whereas MD5 is the faster algorithm while compared with other two. But SHA256 is more reliable to security attacks such as brute force attacks and collision

attack. RIPEMD is highly secure that it could ensure integrity of data. These can be used with cloud storage to keep the integrity of data on it. As the devices generate data in a larger extend and as these data are stored in cloud, it is risky to keep the data without digital signatures.

Finally, DNA based encryption scheme is found to be more reliable and provides various advantages. It is ensuring the speed, while it require minimal power of computation and minimum storage. Therefor the DNA encryption can help the current computing era to keep the data security. As an individual module if it could provide the security provided by both symmetric, asymmetric hand hash algorithms. It can be concluded that it could replace other technologies which requires extra computation.

4.2. Computing Capacity

Cryptography is widening its boundary as the technology is growing, at each and every point of need. Analyzing and providing the exact statistics may be very essential at this situation. Here the theoretical aspects are compared, so as to provide a required amount of confidence for the beginners. The parameters considered for study are Confidentiality, Integrity and Key Strength. As data should be hidden from the eavesdroppers, cryptography can be used to a certain level while integrity cannot be assured. As user's data are with the third party, confidentiality will never help saving users sensitive data.

The efficiency of each and every category of algorithms have been simulated using PHP and the results are recorded for further analysis. The study have been carried out in the basis of four broader categories of algorithms such as Symmetric, Asymmetric, Hash and DNA.

Table 2. Encryption -Symmetric Algorithms (ms)

Input Size	AES	ECC	Blowfish
50	0.52	1.11	1.003
100	0.715	1.36	1.013
150	1.365	1.89	1.067
200	1.5	2.10	1.178
250	2.155	2.34	1.240

As the above table 2 and table 3, shows the results generated by simulation of encryption and decryption of algorithms. It shows that blowfish is reliable presenting a reliable results even after the data size is increased.

Table 3. Decryption – Symmetric Algorithms

Input Size	AES	ECC	Blowfish
50	0.52	1.34	1.101
100	0.70	1.47	1.22
150	1.235	1.78	1.301
200	1.695	2.02	1.321
250	6.370	2.44	1.60

Though ECC and AES are better algorithms the performance is comparatively less as the data size is increased. Therefore among the symmetric encryption algorithms Blowfish can be picked for implementing security at the device level.

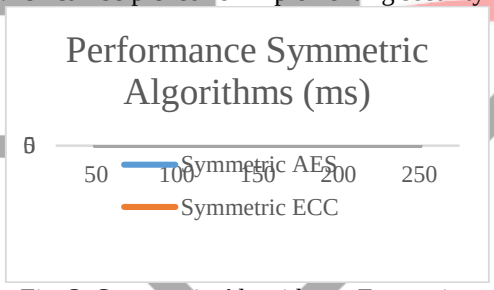

Fig. 3. Symmetric Algorithm – Encryption

The figure 3 and 4., shows clearly the performanc of th blowfish algorithm while comparing with the other two as they take moretime to do the same job when the data size is increased.

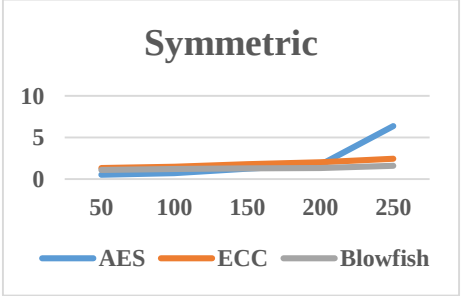

Fig. 4. Symmetric Algorithm - Decryption

The dataset chosen for comparing this is social media data. It can be concluded that blowfish can be used for better device security in the Cloud environment.

Table 4. Encryption - Asymmetric Algorithms (ms)

Input Size	RSA	Diffie-Hellman	Elgamal
50	1.67	1.64	1.52
100	1.93	1.77	1.82
150	2.12	1.93	2.01
200	2.34	2.12	2.14
250	2.67	2.45	2.27

With respect to the asymmetric algorithms, RSA, Diffie-Hellman and Elgamal have been considered. The performance in the basis of its speed Diffie-Hellman is encrypting the given set of data at an average speed.

Table 5. Decryption – Asymmetric Algorithms

Input Size	RSA	Diffie–Hellman	Elgamal
50	1.37	1.33	1.82
100	1.78	1.87	2.02
150	2.23	2.07	2.54
200	2.46	2.35	2.73
250	2.67	2.71	2.92

Among these three algorithms Diffie-Hellman could be considered the best suited in Cloud based IoT environment.

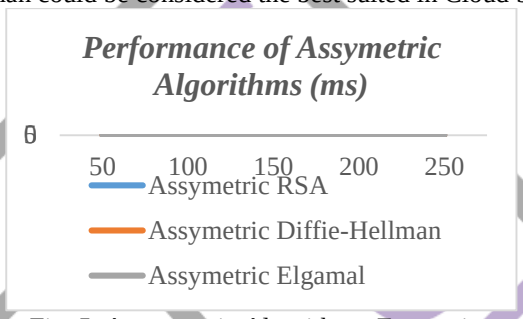

Fig. 5. Asymmetric Algorithm - Encryption

There fig. 5 and fig 6., represents the performance on encryption and decryption done by asymmetric algorithms, with which it can be concluded that Diffie-Hellman could be used to transfer the data in secure way to the cloud.

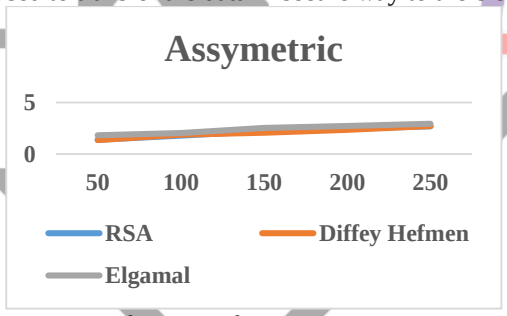

Fig. 6. Performace of asymmetric Decryption

Thus data in transit can be protected from eavesdropping using such secure key transfer mechanism which adds privacy to the data.

Table 6. Hashing Algorithms (ms)

Input Size	MD5	SHA1	RIPEMD-160
50	1.12	.175	.560
100	1.87	.235	.761
150	2.59	.250	1.21
200	2.67	.270	1.38
250	2.98	.285	1.76

Hashing schemes used in signature generation are large in number. This is because of the insecure nature of Cloud and IoT. Very specifically Cloud Storages are prone to attack as they are just devices. These hashes could provide little life to cloud, in such a way that it could identify wheter the data under its control is being modified or not.

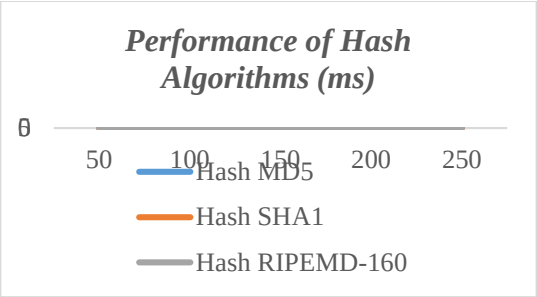

Fig. 7. Hashing Algorithm

The algorithm under study and data received after simulation are tabulated in table 4, the MD5 is considered to be non-reliable and the advisable reliable technique could be SHA or RIPEMD. Among these two SHA is perform the task faster comparing with RIPEMD. Therefore as Fig 7, depicts the results appropriately, it is better to use SHA while enforcing security to the Cloud Storages.

Table 7. Encryption - DNA and Hybrid

Input Size	DNA	Hybrid (ms)
50	.67	0.83
100	.83	0.95
150	.96	1.05
200	1.02	1.14
250	1.11	1.28

Inorder to prove that security can also be enhanced with biological techniques, DNA a worderful performer in IoT world is enhancing the security of data. The encryption mechanism is as simple as ceaser cypher and elemenating complex mathematical computations. So this have been considered for the analysis here.

Table 8. Decryption – DNA & Hybrid Algorithms

Input Size	DNA	Hybrid
50	1.05	1.20
100	1.14	1.33
150	1.37	1.48
200	1.44	1.67
250	1.67	1.72

When DNA is used with some complex mathematical techniques it could not only provide highest level of integrity, but can also do the computations more faster. Thus combining two or more such technique to obtain integrity is considerd as hybrid technique.The results are tabulated in Table 5.

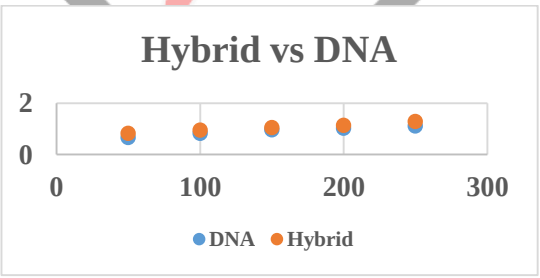

Fig. 8. DNA and Hybrid Algorithm on Encryption

Fig. 8 and Fig 9, shows the actual ability of DNA algorithm without discriminating the performance of Hybrid algorithm. DNA if combined with other algorithms such as symmetric, an asymmetric and a hashing algorithm could keep the data unchanged on the cloud storage.

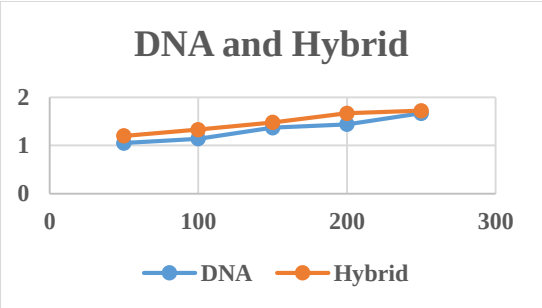

Fig. 9. DNA & Hybrid Algorithm on Decryption

4.2. Analysis on the cryptographic techniques suitable for ensuring DI

Cloud Storage is holding different data from various platforms, various domains and different kind of users. The data used by various devices using MQTT protocol, or HTTP or POP and many such protocols. Therefore it the very essential decision to be taken to choose the wright algorithm for ensuring the integrity of data on the cloud. The study here intended on this problem and the analysed results have been tabulated in Table 9.

Table 9. Comparison of all the Methods

Cryptographic Technique	Encryption (ms)	Decryption (ms)	Performance
AES	1.25	2.10	1.675
ECC	1.76	1.81	1.785
Blowfish	1.10	1.31	1.205
RSA	2.15	2.10	2.125
Diffie-Hellman	1.98	2.07	2.025
Elgamal	2.09	2.41	2.25
MD5	2.25	-	2.25
SHA256	0.24	-	0.24
RIPEMD160	1.13	-	1.13
DNA	0.92	1.33	1.125
HYBRID	1.60	1.48	1.54

While considering the overall performance of symmetric algorithms, blowfish is found to be faster as it could work in device level. This is very suitable in both Cloud and IoT environments. On considering the performance of asymmetric algorithms Diffie-Hellman is considerably performing well as it is employed in secret key transfer. Among the hashing algorithms SHA 256 is doing well as it is safe form attacks. The figure 10, represents the performance of every algorithms which makes the researchers to understand the difference.

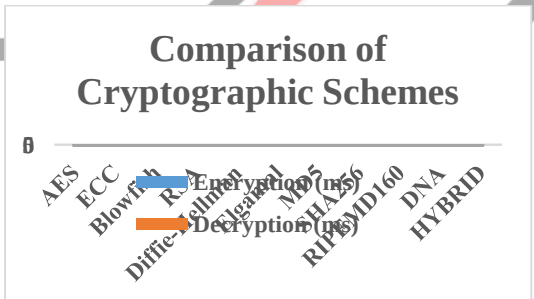

Fig. 10. Performace of Cryptograpic Algorithms

The detailed analysis shows that the combination of algorithms could strengthen the data integrity level. Thus to have a secure cloud storage system the hybrid technique would work better.

5. CONCLUSION

Security threats being the major issue in Cloud storage, this work analyses the techniques implemented in enhancing the integrity of data. It was found that algorithms play a major role in protecting the data. Here the cryptographic algorithms are classified as symmetric, asymmetric, hash, DNA and hybrid. As DNA cryptography is more promising in securing the data, it is considered in the analysis to identify its smartness. As it uses different coding patterns, if merged with the traditional methodologies of cryptography could provide better security while enhancing the efficiency. It is identified that if DNA crypt is used along with traditional cryptographic techniques it not only could enhance the integrity of DS but also could minimize the usage of storage. Also to keep the integrity hashing techniques such as SHA could go hand in hand with DNA. After the analysis it was found that hybrid approach with the inclusion of DNA cryptography could ensure the integrity of the data on the Cloud Storage could be more reliable.

REFERENCES

[1] Sunil Raj Y, Albert Rabara S., An Integrated Architecture For IoT Based Data Storage In Secure Smart Monitoring Environment, International Journal of Scientific & Technology Research, Vol 8(10), 2019, pp. 2213 – 2216.

[2] Sajay, Suvanam Babu, Yeliepeddi, "Enhancing the security of cloud data using hybrid encryption algorithm", Journal of Ambient Intelligence and Humanized Computing, Springer, 2019, DOI: 10.1007/s12652-019-01403-1.

[3] Israa Al Barazanchi, Shihab Shawkat, Moayed Hameed, Khalid Saeed, "Modified RSA-based algorithm: a double secure approach", TELKOMNIKA, 2019, pp. 2818 – 2825.

[4] Jinan Shen, Xuejian Deng, Zhenwu Xu, "Multi-security-level cloud storage system based on improved proxy re-encryption", EURASIP Journal on Wireless Communications and Networking, Springer, 2019, pp. 1-12.

[5] Changzi Yu, Hengjian Li, Xiyu Wang, "SVD-based image compression, encryption and identity authentication algorithm on cloud", IET Image Processing, The Institute of Engineering and Technology, 2019, pp. 2224 – 2232.

[6] Irfan Mohiuddin, Ahmad Almogren, Mohammed Qurishi, Mohhammad Mehedi, Iehab Rassan, Giancarlo Fortino, "Secure Distributed Adaptive Bin pa cking Algorithm for Cloud Storage", Future Generation Computer Systems, 2018, DOI: 10.1016/j.future.2018.08.013.

[7] K.Sharmila, S. Borgia Anne Catherine, Sreeja V.S, "A comprehensive Study of Data Masking Techniques on cloud", International Journal of Pure and Applied Mathematics, Vol. 119 (15) 2018, pp. 3719-3727, ISSN: 1314-3395.

[8] Soumya Shinde, Ramya V. Shinde, Priyanka Kamadhenu, "A Novel Data Auditing Approach to Achieve Data Privacy and Data Integrity in Cloud Computing", International Conference on New Horizons in Science Engineering Technology (NHSET-2018), IJSRCSEIT, 2018.

[9] Thirupaula, Spandhana, Kesavulu Reddy, "Security Analysis of Cryptographic Algorithms in Cloud Computing", International Journal of Engineering Research & Technology, Vol 7 (10), 2018, pp. 208 – 212.

[10] Kajal Rani, Raj Kumar Sagar, "Enhanced Data Storage Security in Cloud Environment using Encryption, Compression and Splitting technique", 2017 2nd International Conference on Telecommunication and Networks (TEL-NET 2017), 2017.

[11] Kumari Sarita, Jawahar Thakur, "Data Centric Security Algorithms In Cloud Computing - A Review", International Journal for Research in Applied Science & Engineering Technology (IJRASET), 2017.

[12] Sugnaya, Durai Raj, "Improving Cloud Security by Enhancing Remote Data Integrity Checking Algorithm", International Conference on Innovations in Power and Advanced Computing Technologies, IEEE, 2017, pp. 1 – 6.

[13] Swetha M., "Creating Secure Cloud by Continuous Auditing Using Dbm Algorithm", International Journal of Computer Science & Engineering Technology (IJCSET), 2017, ISSN: 2229 – 3345. Pp. 38-43.

[14] Lan Zhou, Vijay Varadharajan, K. Gopinath, "A Secure Role-Based Cloud Storage System For Encrypted Patient-Centric Health Records", The Computer Journal, 2016. DOI: 10.1093/comjnl/bxw019

[15] Weiwei Zhong, Zhusong Liu, "Efficient proof of ownership for cloud storage systems", AIP Conference Proceedings 1864, 2017; DOI:10.1063/1.4992867

[16] Nasarul Islam.K.V, Mohamed Riyas.K.V, "Analysis of Various Encryption Algorithms in Cloud Computing", IJCSMC, 2017, pg.90 – 97.

[17] Akashdeep Bhardwaj, GVB Subrahmanyam, Vinay Avasthi, Hanumat Sastry, "Security Algorithms for Cloud Computing", International Conference on Computational Modeling and Security (CMS 2016), ELSEVIER, 2016.

[18] Ashalatha R, Jayashree Agarkhed, Siddarama Patil, "Data Storage Security Algorithms for Multi Cloud Environment", International Conference on Advances in Electrical, Electronics, Information, Communication and Bio-Informatics (AEEICB 16), IEEE, 2016.pp 695 – 699.

[19] Mohamed Ismail, Badamasi Yusuf, "Ensuring Data Storage Security In Cloud Computing With Advanced Encryption Standard (AES) And Authentication Scheme (AS)", International Journal of Information System and Engineering, Vol. 4 (1), 2016.

[20] Ali Azougaghe, Zaid Kartit, An efficient algorithm for data security in cloud storage, IEEE, 2015.

[21] Xiaoyan Zhu, Ripei Hao, Shunrong Jiang, Haotian Chi, Hongning Li, Verification of Boolean Queries over Outsourced Encrypted Data Based on Counting Bloom Filter, IEEE, 2015.

[22] Robert C, Karasinski P, Natowicz R, Limoge A. Adult rat vigilance state discrimination by artificial neural network using a single EEG channel. Physiol and Behav 1996;59(6):1051-1060.

[23] Adleman, L. Molecular computation of solutions to combinatorial problems. Science 266, 1994, pp. 1021-1024.

[24] Israa Al_Barazanchi, Shihab A. Shawkat, Moayed H. Hameed, Khalid Saeed Lateef Al-badri, "Modified RSA-based Algorithm: A Double Secure Approach", Telkomnika, Vol.17(6), 2019, pp.2818~2825. DOI: 10.12928/TELKOMNIKA.v17i6.13201.

[25] Micheal Ernest Taylor, David Aboagye-Darko), "Security Approaches and Crypto Algorithms in Mobile Cloud Storage Environment to Ensure Data Security", Springer Nature Switzerland AG 2019, ICAIS 2019, LNCS 11632, pp. 516–524, 2019.

[26] D.Prabhu, M.Adimoolam, "Bi-serial DNA Encryption Algorithm". Available: https://pdfs. Semanticscholar.org/ 1754/f0eb5852500598a70af 4002e186cd2f3c6ce.pdf

[27] Shreyas Chavan, "DNA Cryptography Based on DNA Hybridization and One Time pad scheme", International Journal of Engineering Research & Technology, Vol. 2 (10), 2013.

[28] Kang Ning, "A Pseudo DNA Cryptography Method", arXiv:0903.2693 [cs.CR], Cornell University Library, 2009.

[29] Kritika Gupta, Shailendra Singh, "DNA Based Cryptographic Techniques: A Review", International Journal of Advanced Research in Computer Science and Software Engineering, Vol. 3 (3), 2013.

[30] Mansi Rathi, Shreyas Bhaskare, Tejas Kale, Niral Shah, Naveen Vaswani, "Data Security Using DNA Cryptograph", International Journal of Computer Science and Mobile Computing, Vol.5, 2016, pg. 123-129.

[31] Hamed Aghili, "Improving Security Using Blow Fish Algorithm on Deduplication Cloud Storage", Fundamental Research in Electrical Engineering, Lecture Notes in Electrical Engineering 480, Springer Nature Singapore Pte Ltd., 2019, DOI: 10.1007/978-981-10-8672-4_54.
[32] C. Nithiya, R. Sridev, ECC Algorithm & Security in Cloud, International Journal of Advanced Research in Computer Science & Technology (IJARCST 2016), Vol. 4 (1), 2016.
[33] Ako Muhammad Abdullah, "Advanced Encryption Standard Algorithm to Encrypt and Decrypt Data", Cryptography & Network Security, 2017.
[34] M. Kumar, "Implementation of DNA cryptosystem using Hybrid approach", Research Journal of Computer and Information Technology Services, Vol. 6(3), 2018.