Partition Ciphering System: A Difficult Problem Based Encryption Scheme

Ziani Fatima Ezzahra[1], Omary Fouzia[2]
Computer Science Department
University Mohammed V
Rabat, Morocco

***Abstract*—In this article, a new encryption scheme called Partition Ciphering System is proposed to adapt and process the message according to the partition problem. The objective of this system, that can be applied as a standalone system or as a building block in a bigger system, is to achieve confidentiality, and maintain a balance between ones and zeros in the output so that attacks like frequency cryptanalysis is avoided and good entropy is met. At first, the authors describe the partition problem together with an adapted version. Secondly, the encryption and the decryption processes are provided. Next, a comparison, in terms of the statistical properties using the DIEHARDER battery, security analysis and performance, with other encryption schemes is presented. From the results, the proposed cryptosystem is resistant to frequency analysis and shows good entropy in the output. Moreover, compared to the Advanced Encryption Standard, it has a random behavior and good confusion and diffusion(Avalanche effect). Also, it displays better performance and resistance to brute force attack on the key.**

Keywords*—Encryption scheme; partition problem; frequency analysis; avalanche effect; confusion; diffusion; statistical properties

I. Introduction

One of the principal concerns in cryptography is confidentiality which includes two concepts [1]: the first one is data confidentiality that would not make personal or confidential data accessible or revealed to unauthorized parties. The second one is privacy that helps concerned entities to control and check who can collect or store their data and who is not allowed to have accessibility to these data in any way.

This paper is concerned by the first concept. There are two types of cryptographic algorithms: the first category includes symmetric systems requiring the same key for encryption and decryption. The second category comprises asymmetric systems demanding two different keys for encryption and decryption [2].

This article's contribution lies in the context of symmetric encryption. It was supposed to be combined with the Symmetrical Evolutionist-based Ciphering (SEC) scheme [3] to make it robust against frequency analysis. In brief, SEC [3] is a symmetric encryption system that substitutes the plaintext's characters and consequently changes their appearance frequency using an evolutionary algorithm. The principal purpose is to make them appear at the same rate. As a result, frequency analysis does not reveal any information. Therefore, the partition problem is the source of inspiration to design the proposed algorithm with this constraint. The authors studied the partition problem and defined an adapted version called the Card-Partition problem to accomplish the goal stated before. They have concluded that the scheme can be considered as a standalone system or as a part of a bigger system. Therefore, they decided to present it alone in this paper.

The rest of this article is organized as follow: Section 2 presents related works. Section 3 defines the partition problem. Next, Section 4 describes the Card-Partition problem and the proposed scheme in detail. In Section 5, some symmetric encryption systems follow. Finally, in Section 6, the results are displayed and discussed.

II. Related work

The partition problem, also called Equal Piles Problem, was defined and first studied by Jones and Beltramo [4]. Different genetic algorithms were proposed to resolve this problem, as Jones and Beltramo's genetic algorithm, Falkenauer's one [5], and by 2000 William A.Green proposed a better one named Eager Breeder Greene [6]. Later, genetic and evolutionist algorithms were used in the design of cryptographic algorithms to reach better security like in [7-11]. In [3], [12], [13], and [14], Omary proposed an evolutionist-based encryption scheme and extended versions that aim to substitute the plaintext's characters to change their occurrence frequency. Later, in [15], Trichni proposed an improved version that comprises a new mutation operator based on the partition problem to provide resistance to frequency analysis and brute force attacks. Afterward, in [16], Bougrine proposed a new encryption scheme inspired by SEC [3] based on the same problem to achieve the same objectives, moreover, in [17] and [18] Kaddouri proposed a revised version of SEC[3]. These works produce a variation of the appearance frequency. But in this article, the purpose is not only to change it, but it is also to produce a balanced output. The authors were inspired by the SEC and the Equal Piles Problem to achieve their goal.

III. Background on the Partition Problem

The authors studied SEC [3] and the partition problem, which is also termed The Equal Piles Problem [4], to design this encryption scheme. The idea is to represent the message as a partition that will be processed by the algorithm. This section defines the partition problem.

A. Partition Problem (Equal Piles Problem)

The partition problem (Equal Piles Problem) purpose is to partition a set into subsets(piles) evenly. Formally, it is defined as follow:

Definition

Given a set S of integer numbers, and an integer k. Divide the set S into k subsets such that:

$$\sum_{e\in S_1} e = \sum_{e\in S_2} e = \ldots = \sum_{e\in S_k} e$$

and$S_1 \cup S_2 \cup \ldots \cup S_k = SandS_1 \cap S_2 \cap \ldots \cap S_k = \emptyset$

where e are elements of the subset S_i, and S_i is the i^{th} subset.

The problem is known to be hard. Also, this problem represents the motivation of this article [4]. The authors described a revised version, that leads to the concerned objective, in the next section.

IV. Partition Ciphering System (PCS) Description

In this section, the authors proposed a revised version of the partition problem called the Card-Partition Problem. Moreover, they presented a detailed description of the proposed scheme.

A. Card-Partition Problem

Given a set S of integer numbers, and an integer k. Divide the set S into k subsets such that:

$$Card(S_1) = Card(S_2) = \ldots = Card(S_k)$$

and$S_1 \cup S_2 \cup \ldots \cup S_k = SandS_1 \cap S_2 \cap \ldots \cap S_k = \emptyset$

where Card(S_i) is the cardinal of the subset S_i.

This definition is the main idea to achieve the objective of the article. Therefore, the authors represented the plaintext by a partition. Furthermore, a new partition, that satisfies the constraint of the subsets cardinalities' equality, is constructed (an instance of the card-partition problem). Thus, the resulting ciphertext is resistant to frequency analysis.

B. PCS (Partition Ciphering System) Encryption

The objective of this system is to get a partition in which all the subsets have the same cardinality. The encryption scheme consists of 3 steps:

- At first, the authors define the plaintext partition.
- Secondly, the authors compute the ideal cardinality.
- Finally, the authors add or delete some blocks to construct the ciphertext partition depending on the ideal cardinality value.

The secret key is constructed during the algorithm process.

K={{k},{NumberOfAddedBlocks}, ListOfDeletes, Permutation}

Where, k is the size of the blocks and NumberOfAddedBlocks is the number of added blocks. The ListOfDeletes represents the deleted blocks and their corresponding positions of deletes. Finally, the permutation is the transformation mapping between the plaintext partition and the ciphertext partition.

1) Step 1: The plaintext partition construction: Let the binary message M be the input. The plaintext partition is formed as follow:

At first, an integer $k \geq 2$ is randomly chosen, then the message is split into blocks of size k (B_0, B_1, … , B_{m-1}). Thereafter, to each block B_i, a list of occurrence Li is associated with.

Let n be the number of blocks in M. In other words, n is the size of M. The L_is form a partition of {0, 1, ..., n − 1} such that $0 \leq i < m$ (m is the number of L_is). This partition subsets do not have the same cardinality. Next, the resulting partition is constructed in the next steps.

To reach the scheme's aim, the ideal cardinality, representing the occurrence number of each block in the ciphertext, is specified.

2) Step *2: Ideal Cardinality (IC) definition:* Let $c = \frac{n}{m}$, where n is the number of blocks in M, and m is the number of distinct blocks in M. If c is an integer, then IC=c. Otherwise, $IC = \lceil c \rceil$.

3) Step 3: The ciphertext partition construction: The ciphertext partition is a partition of the set {0, ..., n'- 1}, where n' is the number of blocks in the ciphertext.

Let n'=IC × m, and NumberOfAddedBlocks = 0 at first.

According to the ideal cardinality value, the authors conclude the next step for each list Li.

For each block B_i where ($0 \leq i < m$), the cardinality of the L_i Card(L_i) is compared with the ideal cardinality.

a) Case 1: If Card(L_i) < IC, then the corresponding block B_i is appended to the message M, and the NumberOfAddedBlocks is incremented. Also, the plaintext partition is updated by the position where the block B_i is inserted in L_i.

b) Case 2: If Card(L_i) > IC, then the corresponding block B_i is removed from a randomly chosen position from L_i. And the ListOfDeletes is updated as follows: at first the index i of B_i is inserted in the ListOfDeletes (if i does not exist in the ListOfDeletes), next the position from where it was deleted is inserted. The key is updated during the encryption progressively together with the plaintext partition to produce the ciphertext partition (the deleted block B_i must be removed also from the L_i in the partition).

At the end, the final partition representing the ciphertext is reached.

The secret key is denoted by:

K={{k},{NumberOfAddedBlocks},{IndexOf(B_i)→{PositionsOf(B_i)},...,IndexOf(B_l)→{PositionsOf(B_l)}},Permutation}

Algorithm 1 provides the pseudocode of the PCS encryption process. Where, the ListOfDeletes and the permutation are initially empty.

Algorithm 1: PCS encryption

Input : The message M
Output: The ciphertext C and the secret key K

Begin
M'← EncodeToBinary (M)
k← randomNumber()
NumberOfAddedBlocks ← 0
K← {{k},{NumberOfAddedBlocks},ListOfDeletes, Permutation}
M'← DivideIntoBlocks(M',k)
n← sizeOf(M')
m← NbOfDiffBlocks(M')
PlaintextPartition← ToPartition(M')
ListOfBlocks← DiffBlocks(M')
IC← ComputeIdealCardinality(n,m)
For I from 0 to m-1 **do**
While Card(L_i)<IC **do**
M'← add(B_i, M')
NumberOfAddedBlocks← NumberOfAddedBlocks+1
K← Update(K, NumberOfAddedBlocks)
PlaintextPartition← updatePartition(PlaintextPartition)
EndWhile
While Card(L_i)>IC **do**
M'← Delete(B_i, randomPosition(L_i), M')
K← Update(K, ListOfDeletes)
PlaintextPartition← updatePartition(PlaintextPartition)
EndWhile
EndFor
C← M'
CiphertextPartition← PlaintextPartition
Permutation←GeneratePermutation(PlaintextPartition, CiphertextPartition)
K← Update(K, Permutation)
End

Fig. 1 summarizes the encryption process detailed before.

C. PCS (Partition Ciphering System) Decryption Algorithm

The decryption algorithm consists of two steps: the ciphertext is first split, and then the inverse actions of the encryption process are done.

1) Step 1: Given the ciphertext C and the secret key *K = {{k}, {NumberOfAddedBlocks}, ListOfDeletes, Permutation}*, C is split into blocks of size k. The *ListOfDifferentBlocks* is defined to be the list of different blocks in the ciphertext.

2) Step 2: At first, each of the inserted blocks is removed from the last position in the message and the *NumberOfAddedBlocks* is decreased by 1 each time.
Next, the *ListOfDeletes, ListOfDifferentBlocks, and Permutation* are used to insert each of the deleted blocks in the position it was removed from.

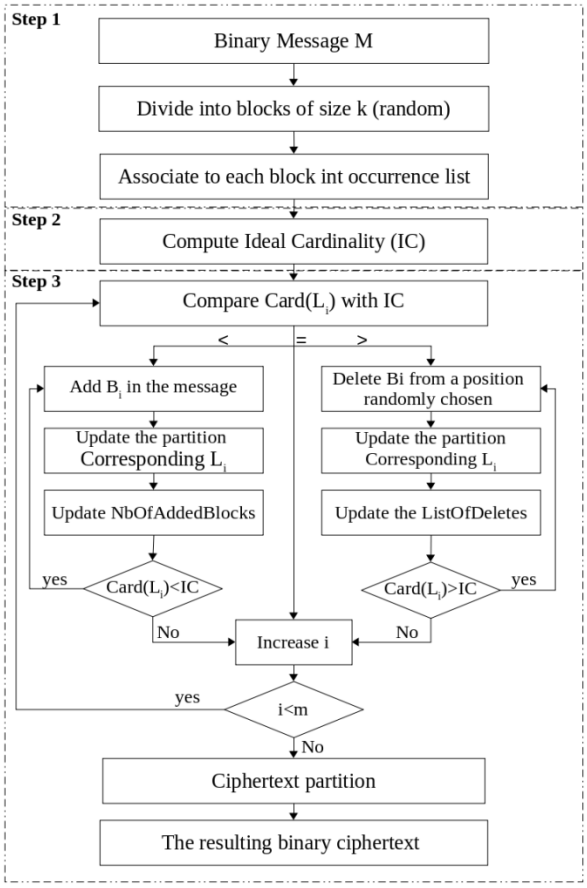

Fig. 1. PCS Encryption Process.

Algorithm 2 describes the decryption algorithm.

Algorithm 2: PCS decryption

Input : The ciphertext C and the secret key K
Output: The message M

Begin
C←DivideIntoBlocks(C,k)
ListOfDiffBlocks← DifferentBlocks(C)
While NbOfAddedBlocks > 0 **do**
C← DeleteFromLast(C)
NbOfAddedBlocks← NbOfAddedBlocks-1
EndWhile
For i from 0 to sizeOf(ListOfDeletes) **do**
Add(Permutation, ListOfDiffBlocks,ListOfDeletes, C)
EndFor
M← C
End

Fig. 2 displays the PCS decryption process.

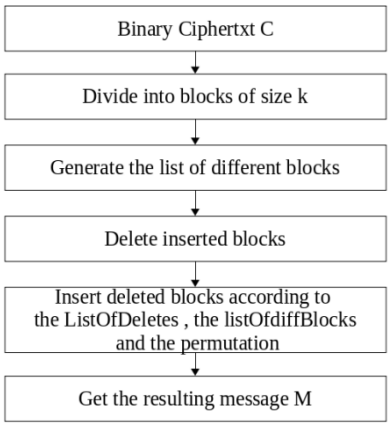

Fig. 2. PCS Decryption Process.

V. Some Symmetric Encryption Schemes

The most famous symmetric encryption schemes are DES (Data Encryption Standard), 3DES (3-Data Encryption Standard) and AES (Advanced Encryption Standard). This section describes them briefly.

A. DES (Data Encryption Standard)

The Data Encryption Standard is a symmetric block cipher that encrypts 64-bit data blocks by using a 56-bit key. It consists of 16 Feistel iterations surrounded by two permutations, IP at the input, and its inverse IP^{-1} at the output. The 48-bit sub keys are formed from the 56-bit key using a permutation [19].

B. 3DES (Triple Data Encryption Standard)

The 3DES is a symmetric block cipher that encrypts 64-bits data block. Using three keys of size 56-bits. It is an enhancement of DES, which consists of 48 Feistel rounds. It is vulnerable to differential cryptanalysis [20].

C. AES (Advanced Encryption Standard)

The Advanced Encryption Standard is a symmetric block cipher that encrypts 128-bit data blocks. It uses symmetric 128-bit, 192-bit or 256-bit keys. It consists of 10, 12 and 14 rounds depending on the key size. Brute force attack is the only effective attack known against this algorithm. AES encryption is fast and flexible [21]. Table I presents the characteristics of DES, 3DES and AES.

TABLE. I. Characteristics of DES, 3DES and AES

Parameters	**Encryption systems**		
	DES	***3DES***	***AES***
Key length	56 bits	168 bits	128,192, or 256 bits
Block size	64 bits	64 bits	128 bits
Developed	1977	1978	2000
Cryptanalysis resistance	Vulnerable to differential and linear cry-ptanalysis	Vulnerable to differential cryptanalysis	Strong against differential, tr-uncated differe-ntial, linear, int-erpolation and square attack

VI. Results and Security Analysis

This section presents the statistical tests and the test of confusion and diffusion properties. Also, PCS and some symmetric encryption schemes are compared.

A. Dieharder Test

Dieharder battery was developed to test the behavior of the pseudo-random number generators and other cryptographic features like encryption schemes and hash functions. This battery consists of 32 tests [22]. A file that contains a sequence of 10 Mb is generated using the PCS algorithm. The algorithms of the battery compute the p-values. The significance level for Dieharder is 0.005, and if the p-values are on the range [0.005, 0.995], then the results are good enough[21]. Fig. 3 shows that PCS passed all the tests of Dieharder battery, as 0.2<p-values(PCS)<0.9. Also, the AES outputs p-values 0.05<p-values(AES)<1. All the p-values of PCS are good enough to conclude that the behavior of the scheme is random. Also, the AES has a random behavior even if some p-values are near to 0.995 and 0.005. Fig. 3 shows that PCS results are better than the AES results.

B. Confusion and Diffusion Properties

In this part, the confusion and diffusion properties are tested for the PCS scheme and compared to the AES. From Shannon's view point, to decide if an encryption scheme is secure against statistical analysis, it is required to satisfy the confusion and diffusion properties [23]. AES is known to have good confusion and diffusion properties. Confusion represents the relation of the ciphertext with the key that must be complex. Moreover, diffusion represents the relation between the plaintext and the ciphertext (changing one character /bit in the ciphertext/plaintext should influence a significant number of the plaintext/ciphertext characters).

The avalanche effect is the best tool to check these properties. The diagram in Fig. 4 illustrates the confusion property for PCS. From Fig. 4, the portion of the changed bit in the ciphertext is approximately 50 % for PCS(Avalanche effect) and AES. These values mean that this scheme satisfies the confusion property.

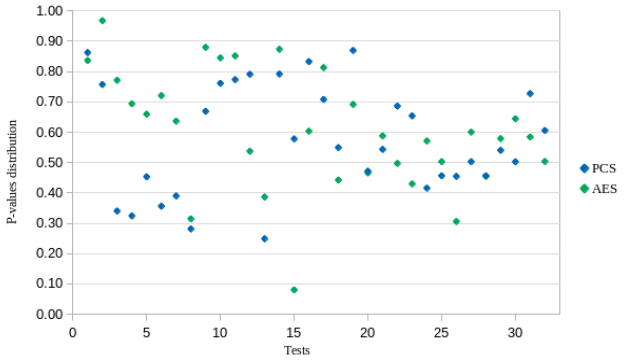

Fig. 3. The Dieharder Results of PCS Encryption Algorithm and AES.

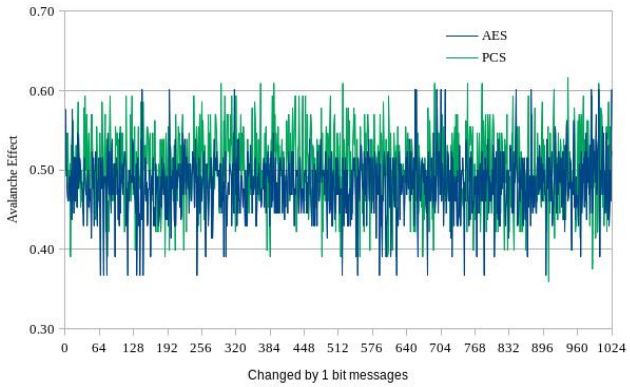

Fig. 4. The Confusion Property of PCS Encryption Algorithm Compared to AES (Avalanche Effect).

Table II illustrates the diffusion property of PCS. The average percentage of changed bit in the ciphertext is around 50%. Furthermore, since PCS has a session key, in case of a compromised state, it will not affect other pairs of plaintexts and ciphertexts. It is related to its encryption process randomness. Also, the same message is differently encrypted each time. To conclude, from the results of the avalanche effect and the statistical tests provided, the PCS has good confusion and diffusion properties.

C. Comparison of PCS with AES, DES and 3DES

1) Encryption and decryption time: In this part, the authors compared the encryption and decryption time of the proposed scheme PCS with DES, 3DES, and AES (see Fig. 5).

The authors noticed that:

- DES has a higher encryption and decryption time compared to the AES and PCS. Parallel computing made breaking DES quite simple. 3DES is not as vulnerable as DES, but it is too slow compared to the other schemes.
- AES has a shorter encryption time compared to DES and 3DES encryption schemes. And it is equivalent to the PSC. Even if the structure simplicity might be inconvenient, AES is faster, more flexible, and stronger than DES and 3DES from the Table I.
- PCS has an encryption time equivalent to the AES and better than DES and 3DES. It has a shorter decryption time comparatively with the others.

2) Security comparison: The keys length is a useful metric when it comes to the cryptographic strength. Because if a longer key is used to encrypt a text, it is hard to decrypt without the appropriate key.

TABLE. II. DIFFUSION PROPERTY OF THE PCS ENCRYPTION ALGORITHM

	Ciphertext pairs								
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
% of change	51	47	56	59	41	49	52	60	56

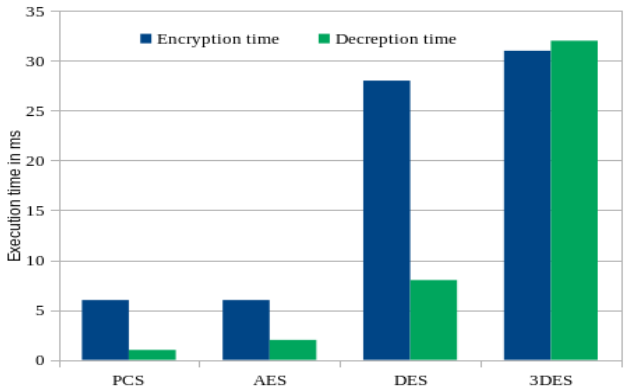

Fig. 5. Encryption and Decryption Time of PCS, AES, DES and 3DES.

Table III shows the strength of the PCS compared to 3DES, DES, and AES, depending on the key length. Moreover, PCS encryption is resistant to linear and differential cryptanalysis. Further, it is robust against frequency analysis. Additionally, the key used in the PCS is a session key, which is not the case of AES, DES, and 3DES.

D. Frequency Analysis

In this subsection, the authors performed the frequency analysis before and after the encryption algorithm. Fig. 6 presents the results of the frequency analysis. As stated before, the purpose of PCS is to have a balanced output. In other words, each block appears with the same frequency. The articles [15],[16],[17], and [18] did not achieve this objective. In PCS, the encryption did not use an evolutionary algorithm to solve the problem.

TABLE. III. KEY LENGTH OF THE 3DES, DES, AES AND PCS FOR M_1

	Encryption Schemes			
	DES	***3DES***	***AES***	***PCS***
Key length	56 bits	168 bits	128 bits	952 bits
# possiblle keys	2^{56}	2^{168}	2^{128}	2^{952}

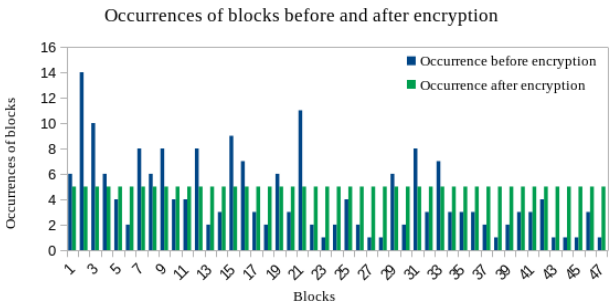

Fig. 6. Occurrences of Blocks before and after Encryption using PCS.

VII. CONCLUSION

In this article, the Partition Ciphering System (PCS), which is inspired by a previously designed system SEC and the partition problem, is proposed. The purpose of PCS is to generate a balanced ciphertext to avoid frequency analysis. Yet, the standard definition of the partition problem did not

provide the objective of PCS. Consequently, the authors proposed a revised partition problem definition called the Card-Partition problem, in which the subsets have the same frequency. The PCS encryption algorithm is a possible way to solve the problem. They performed a comparison of the PCS with AES, DES, and 3-DES. Thus, according to the Dieharder test battery, the results display the random behavior of PCS. Compared to the AES, the PCS provided better results. The results of the brute force attack and frequency analysis are promising. Moreover, the confusion and diffusion properties are satisfied in PCS.

In further work, a new version of the PCS scheme is going to be done to make it suitable for the security of the wireless body area networks and IoT devices in general.

REFERENCES

[1] Stallings, W. (2017). Cryptography and Network Security. 7th ed. Harlow, United Kingdom: Pearson Education Limited.

[2] Douglas ,R. Stinson , Cryptography theory and practice,Chapman & Hall/CRC, U.S .2006. 19-20.

[3] Omary, F., Mouloudi, A. , Tragha, A., Bellaachia, A., A New Ciphering Method Associated with Evolutionary Algorithm,Lecture Notes in Computer Science – Publisher: Springer Berlin / Heidelberg -ISSN: 0302-9743 -Subject: Computer Science-Volume 3984 ,pp 346-352(2006).

[4] Jones,D.R., Beltramo,M.A.," Solving Partitionning Problems with Genetic Algorithms", in Belew, K. R. & Booker,L. B. (Eds.),Proceedings of the Fourth International Conference on Genetic Algorithms;Maurgan Kaufmann Publ.,San Francisco(1991).

[5] Emannuel, F., Solving Equal Piles with the Grouping Genetic Algorithm, in Eshelman, L. J. (Ed.), Proceedings of the Sixth Intenational Conference on Genetic Algorithms; Morgan Kaufmann Publ., San Francisco (1995).

[6] Greene, W.A., Partitioning Sets with Genetic Algorithms, in J. Etheredge and B. Manaris(eds.), Proceedings of the Thirteenth International Florida Artificial Intelligence Research Society (FLAIRS) Conference, May 22-24 , 102-106. (2000).

[7] Kumar, A. and Chatterjee, K. (2016). An efficient stream cipher using Genetic Algorithm. 2016 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET).

[8] Putera, A., Siahaan, U. and Rahim, R. (2016). Dynamic Key Matrix of Hill Cipher Using Genetic Algorithm. International Journal of Security and Its Applications, 10(8), pp.173-180.

[9] Pareek, N. and Patidar, V. (2014). Medical image protection using genetic algorithm operations. Soft Computing, 20(2), pp.763-772.

[10] Khare, M. and Yadav, C. (2017). Secure data transmission in cloud environment using visual cryptography and genetic algorithm: A review. 2017 International Conference on Innovations in Control, Communication and Information Systems (ICICCI).

[11] Sethi, P. and Kapoor, V. (2016). A Proposed Novel Architecture for Information Hiding in Image Steganography by Using Genetic Algorithm and Cryptography. Procedia Computer Science, 87, pp.61-66.

[12] F.Omary, A.Tragha, A.Lbekkouri, A.Bellaachia, A.Mouloudi, "An Evolutionist Algorithm to Cryptography"-Brill Academic Publishers-Lecture Series And Computational Sciences, Volume 4, 2005, pp.1749-1752.

[13] A.Mouloudi, F.Omary, A.Tragha, A.Bellaachia, "An Extension of evolutionary Ciphering System". 2006 International Conference on Hibrid Information Tchnology,Novembre 9th – 11th,2006.

[14] F.omary, A.Tragha, A.Bellaachia, A.Mouloudi. « Design and Evaluation of Two Symmetrical Evolutionist-Based Ciphering Algorithms ». International Journal of Computer Science and Network Security (IJCSNS) February 28, 2007 pp 181-190.

[15] S.Trichni, F.Omary, B.Boulahiat, M.Bougrine,"A new approach of mutation's operator applied to the ciphering system SEC", 6th ICCIT: International Conference on Computer Sciences and Convergence Information Technology (ICCIT 2011), Jeju,Corea,pages:680-685,Novembre 2011.http://www. aicit.org/iccit "

[16] M. Bougrine, F. Omaiy, S. Trichni and B. Boulahiat, "New evolutionary tools for a new ciphering system SEC version," 2012 IEEE International Carnahan Conference on Security Technology (ICCST), Boston, MA, 2012, pp. 140-146. doi: 10.1109/CCST.2012.6393549.

[17] Kaddouri Z, Omary F, Abouchouar A, Daari. M. Balancing Process to the Ciphering System Sec. Journal of Theoretical and Applied Information Technology 2013; 52: 092-093.

[18] Kaddouri Z, Omary F, Abouchouar A. Binary Fusion Process to the Ciphering System "Sec Extension to Binary Blocks". Journal of Theoretical and Applied Information Technology 2013; 48: 067-075.

[19] Biryukov, A., De Cannière, C., Encyclopedia of Cryptography and security, 295-300. Springer, Heidelberg (2011).

[20] Hamdan,O.A., Zaidan, B.B., Zaidan, A.A., Hamid, A.J., Shabbir, M., and Al-Nabhani, Y. , New Comparative Study Between DES, 3DES and AES within Nine Factors,Journal Of Computing, Volume 2, Issue 3, March (2010).

[21] G. Brown, R. (2019). Robert G. Brown's General Tools Page. [online] Webhome.phy.duke.edu. Available at: http://webhome.phy.duke.edu/~rgb/General/dieharder.php.

[22] Daemen, J., Rijmen, V., Encyclopedia of Cryptography and security,pp 1046-1049. Springer, Heidelberg (2011).

[23] Shannon,C. E.. A mathematical theory of communication. The Bell System Technical Journal, 27(3): 379–423 ,(1948).