Preserving Authenticity and Integrity of Distributed Networks through Novel Message Authentication Code

Gurpreet Kour Sodhi[1], Gurjot Singh Gaba[2], Lavish Kansal[3], Eduard Babulak[4], Mohammed AlZain[5], Sandeep Kumar Arora[6], Mehedi Masud[7]
[1,2,3,6]School of Electronics and Electrical Engg., Lovely Professional University, Jalandhar, India – 144411
[4]Zigurat School of Business and Technology Innovation, Spain - 08018
[4]National Science Foundation, Washington, D.C., USA – 22314
[5]Department of Information Technology Taif University, Saudi Arabia – 21974
[7]Department of Computer Science, Taif University, Saudi Arabia – 21974

Article Info

Article history:

Received Aug 28, 2018
Revised Oct 15, 2018
Accepted Oct 29, 2018

Keywords:

Blum blum shub random number generator
DNA
Hash
Integrity
Message Authentication Code

ABSTRACT

In this era of universal electronic connectivity, communication is no more confined to transfer of data from one end to the other; rather it aims at secure data transfer. Communication sector has developed beyond this traditional boundary of data transfer and is now working on ways to provide data from the intended senders to the intended receivers in an unaltered form. Considering all these conditions, the data transfer needs to follow the principles of authentication, confidentiality and integrity. The former two have been addressed using digital signatures and encryption schemes respectively, while the solution to the later is the use of Message Authentication Code. This paper presents a Message Authentication Code scheme, which uses the biological characteristics represented by Deoxyribonucleic acid combined with the output of Blum Blum Shub Random Number Generator, as a secret key along with a novel hash algorithm. This Message Authentication Code structure is evaluated on the basis of National Institute of Science and Technology test suite for random numbers, avalanche criteria and network attacks. The results reveal that the proposed scheme performs well under all the criteria and thus is capable of preserving integrity; this increases its applicability in any data sensitive environment.

Corresponding Author:

Sandeep Kumar Arora,
School of Electronics and Electrical Engg.,
Lovely Professional University,
Jalandhar, India – 144411.
Email: sandeep.16930@lpu.co.in

1. INTRODUCTION

One form of Communications security is preventing unauthorized interceptors from accessing the data while being transferred to the intended receivers. With the advancements in the field of electronic commerce, the data being transferred over Networks should be kept confidential and require to be prevented from any unauthorized access or modification [1]. Thus, data integrity is the need of the hour when it comes to the present form of communication. Each bit of information carries a certain value which needs to be retained and any modification by the intruder can lead to disasters [1].

Various methods have been applied to authenticate data in the past. A data authentication scheme is proposed which helps in privacy preservation and is based on encryption scheme, 'pseudonym technology' and Message Authentication Code [1]. Authentication refers to confirmation, that the data has been received from the projected sender and this is usually verified using secret keys which are known to either ends only [2]. This eradicates the chance of data being received from an advisory since the key has not been shared to anyone. Biometrics serves the best purpose when it comes to authentication; it involves the use of characteristic features of the user which are unique for an individual and cannot be replicated. Sofia et al. in [3], pointed out to that feature of human body which can be used as the authentication identity. The authentication process is carried out by using this unique feature as a key to develop a secure system [4]. A multi biometric user authentication scheme is proposed by Koong et al. [5] which utilize both the physiological and behavioral biometrics. Finger movements on multi touch devices are used for security level. In addition to this, the user credential is made replaceable to prevent any privacy leakage. Dilli and Chandra [6] present another scheme involving the use of HMAC SHA 256 Algorithm for message authentication and data integrity [6]. Verma and Prajapati [7] present a novel SHA which executes in less execution time and possesses higher bit difference, this can be implanted so as to increase the security.

In this paper, a scheme has been presented to design, MAC (Message Authentication Code) which is used to verify the integrity of a message. It assures that the data received is unaltered and not modified by any means. As observed from Equation 1, MAC uses a secret key & hash algorithm and takes the message as input to produce a tag, also known as cryptographic checksum. This tag is appended to the message and compared at the receiver end to conclude if the message is in its original form. Here the assumption is that the secret key is shared between the sending and the receiving party only.

$$MAC = C(K, M) \tag{1}$$

Where, C is the MAC function, K is the secret key, M is the message input, MAC is the message authentication code [8].

The secret key to be used in the presented scheme is generated using DNA (Deoxyribonucleic acid) which is present in living beings and is unique to every individual. Thus, DNA can be efficiently used for authentication provision.

Further, to increase the strength of the algorithm BBSG (Blum Blum Shub Random Number Generator) has been used to further increase the complexity of the algorithm. This generator takes a seed value as the input and produces a random sequence as the output. Apart from the secret key, a hash algorithm has also been used in MAC designing. This novel hash algorithm is a result of the integration of 'f' function in the existing SHA-160 algorithm. The proposed work has been explained with the help of a flow chart in Figure 1. The detailed description is present in Section 2 followed by results in Section 3.

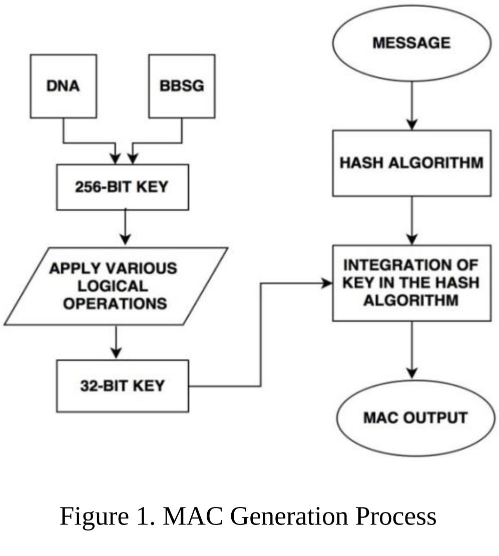
DNA
BBSG
256-BIT KEY
APPLY VARIOUS LOGICAL OPERATIONS
32-BIT KEY
MESSAGE
HASH ALGORITHM
INTEGRATION OF KEY IN THE HASH ALGORITHM
MAC OUTPUT

Figure 1. MAC Generation Process

2. RESEARCH METHOD

Data integrity is maintained using MAC which constitutes of a message input, a secret key and a hash algorithm. The proposed scheme uses a novel hash algorithm which follows the basic structure of SHA-160, to enhance its strength an 'f' function has been integrated along with a secret key which is produced using DNA sequence and BBSG produced random sequence. The proposed MAC scheme is compatible with the existing security paradigms and does not add any more complexity. The existing systems can replace the traditional versions of MAC with the new proposed MAC. The detailed structure is explained in the following subsections.

2.1. Novel Hash Algorithm

The novel hash algorithm used in the proposed algorithm is a result of the incorporation of ‘f’ function in the basic structure of SHA-160. The SHA-160 algorithm constitutes of total 80 rounds and for every 20 rounds a constant ‘K’ is used as input. There are four ‘K’ values, each of which is 8 digit hexadecimal values. The message digest (MD) produced is of 160 bits. The ‘f’ function constitutes of three operations; Expansion (EXP), Substitution using S-box (S) and modulo 2^{48} addition (+) applied on the five register values (A, B, C, D, E) [4]. The structure of the hash algorithm is explained using Figure 2.

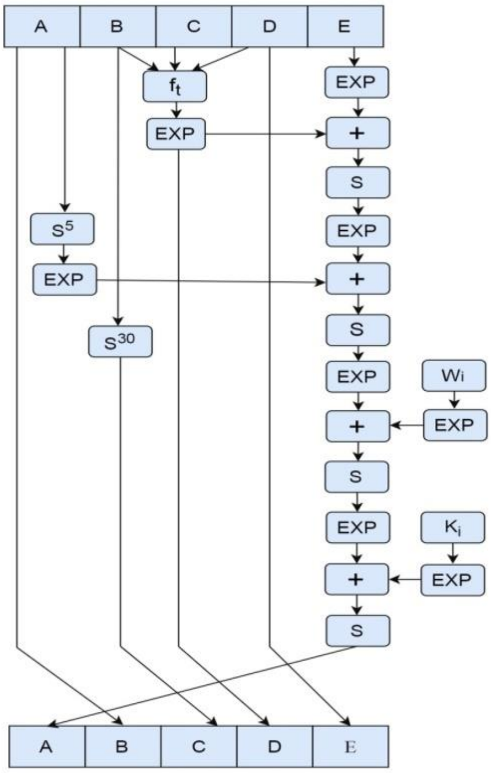

Figure 2. A Novel Hash Algorithm

2.2. DNA-BBSG Based Secret Key

The novel hash algorithm is applied on the message inputs along with the secret key. The secret key used in the proposed scheme is framed using the DNA which represented in the form of a sequence comprising of ‘agct’ characters following a unique pattern for every individual. The characteristic uniqueness of a DNA sequence makes it impossible to be replicated or stolen [8].

To strengthen the security, BBSG is used to produce an output random sequence which is the result of the secret seed value given to the random number generator. The DNA sequence is converted into its binary form and exclusive-or operation is applied in between DNA and BBSG sequence. This results into a 256-bit key, which is further used in the formation of MAC [9].

2.3. Formation of MAC

MAC is also known as keyed Hash due to its composition having a secret key and a hash algorithm [10]. The DNA-BBSG key is of 256 bits and this key has to be further split into four 32-bit keys in order to be used for MAC, therefore various operations are applied on the 256-bit key in order to convert it into four 32-bit keys. The various applied operations are explained through pseudocode and graphical representation for better understanding.

1) Apply 6-bit circular shift on the 256-bit DNA-BBSG key four times and store the sequence every time it’s shifted.

2) Then split the 256 bit shifted key into four parts of 64-bit each.
3) Apply Exclusive-or operation on first two parts, followed by repeating it for the last two parts, resulting in a sequence of 64-bits.
4) Split the obtained 64-bit sequence into two parts of 32-bit each.
5) Apply Exclusive-or operation between the obtained sequences.
6) Convert the obtained 32-bit sequence into hexadecimal form, forming the 8-bit hex key.
7) Repeat steps 2 to 6 every time the key is shifted by 6-bits, thus obtaining final four keys.

A pseudo code giving a better description of the operations involved is given under.

```
Y= initial key of 256-bits
for i=0:4                           //repeating the operation four times
Y(i)= circshift (Y,6)               //circular shift of 6-bits and store the bit stream
end
for i=0:3                           //splitting the key into 4 parts of 64-bit each
x(i+1,:)= Y(64*i+1:64*(i+1))
end
for j=1:4
k(j)= ~xor(x(2*j-1,:), x(2*j,:))    //Applying exclusive-or operation between consecutive pair and then
between their results forming one 62-bit sequence.
end
for i=0:3                           //splitting the key into 2 parts of 32-bit each
x(i+1,:)= Y(32*i+1:32*(i+1))
end
for j=1:2
k(j)= ~xor(x(2*j-1,:), x(2*j,:))    //Applying exclusive-or operation
end
//k represents the key//
//The four key values are converted into hexadecimal form and then used in MAC//
```

A graphical representation is given in Figure 3.

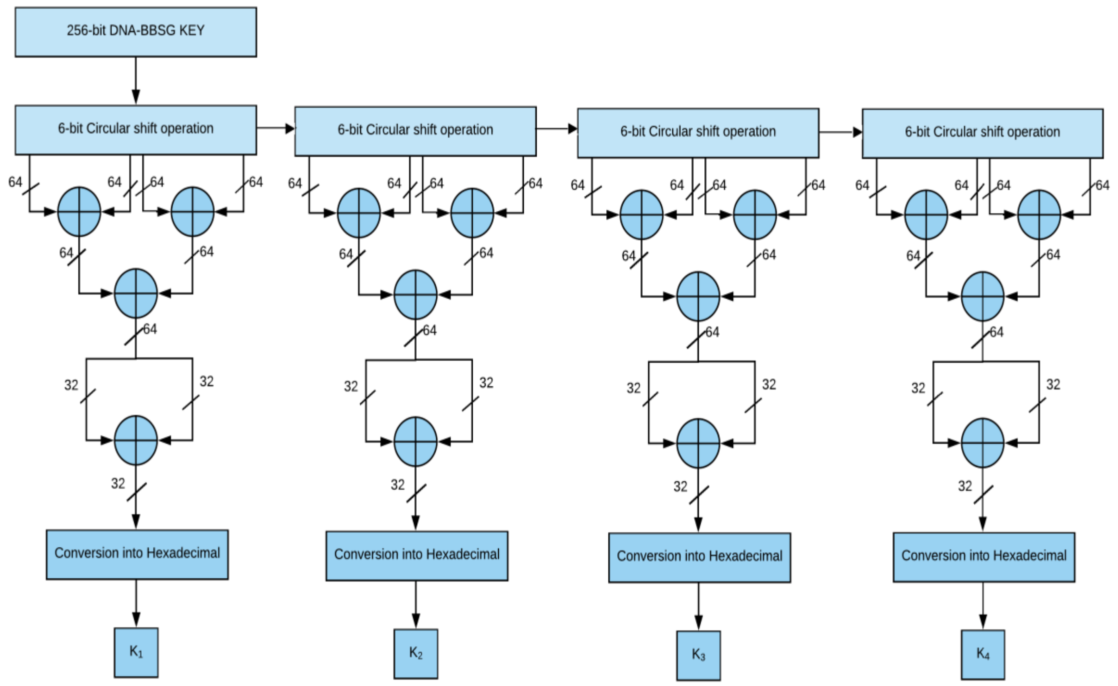
256-bit DNA-BBSG KEY
6-bit Circular shift operation
6-bit Circular shift operation
6-bit Circular shift operation
6-bit Circular shift operation
64
32
Conversion into Hexadecimal
Conversion into Hexadecimal
Conversion into Hexadecimal
Conversion into Hexadecimal
K_1
K_2
K_3
K_4

Figure 3. Key Generation Process

The final four keys are represented in hexadecimal form as shown in Table 1. The SHA-160 algorithm uses four 32-bit constant values [11] which are replaced with the four keys which were framed using the operations. The software tool MATLAB has been used for the simulation.

Table 1. Security Keys

S.No.	32-bit Keys (Hexadecimal)
K_1	CD2740EB
K_2	AF349D03
K_3	0EBCD274
K_4	D03AF349

This forms a novel MAC algorithm. The MAC values obtained, using the proposed technique are given in Table 2.

Table 2. MAC Values

Input	Hex form	MAC Values (Hexadecimal)
G	67	d4be6361479e2705954fbe21641c92d9a37c65e3
Sodhi	6f646869	9ced64e520ccfd741b8ed38326e33a6b88a65b52
Unitedstates	756e69746564 737461746573	f29d903139058701e39e3ceb1c1776316454ef0c

The computed MAC values are then converted into binary form for evaluation on randomness and avalanche criteria.

3. RESULTS AND ANALYSIS

The proposed technique is analyzed using NIST test suite of randomness and the strict avalanche criteria. These tests are performed for three different input values each having a different length. These tests compute the P-value for a binary sequence; which must be greater than 0.01 for a sequence to be considered as random [12].

In order to validate the efficiency of our proposed technique, the NIST results of the proposed MAC scheme are compared with those of the existing techniques. The eight digit hexadecimal key used for these HMAC techniques is '3A54E26B', which is kept constant for all the schemes.

A brief overview of the various NIST tests is given as:

1) Frequency Test

Frequency test calculates the ratio of the number of ones and zeros in the entire sequence. It checks the adjacency between the number of ones and the number of zeros. A sequence is considered to be random if the ratio of both is close to each other [12]. The results in Table 3 depict that the proposed algorithm produces better proximity between the count of ones and zeros as compared to other techniques.

Table 3. NIST test results for Frequency test

MAC Technique	**P-values**		
	G	Sodhi	unitedstates
HMAC MD2	0.3768	0.3768	0.4795
HMAC MD5	0.8597	0.5959	0.8597
HMAC SHA-160	0.8744	1.0000	0.8744
HMAC SHA-256	0.2606	0.9005	0.0801
HMAC SHA-384	0.2207	0.1258	0.3074
HMAC SHA-512	0.7909	0.9296	0.9296
Proposed Technique	0.8774	0.9389	0.9219

2) Binary Derivative Test

The Binary Derivative Test proceeds by applying exclusive-or operation between all the consecutive bits of the sequence until only one bit is left. Then, the ratio of the number of ones to the total number of elements in the sequence is calculated for each case. Finally, the average of the ratio for all the sequences is calculated, if this value lies near to 0.5, then the sequence is random [12]. The results in Table 4 illustrate that the output of the proposed scheme is random.

Table 4. NIST test results for Binary Derivative test

MAC Technique	P-values		
	G	Sodhi	unitedstates
HMAC MD2	0.4952	0.5126	0.5016
HMAC MD5	0.5129	0.4901	0.5149
HMAC SHA-160	0.5069	0.4924	0.5026
HMAC SHA-256	0.5046	0.5007	0.5040
HMAC SHA-384	0.5005	0.4964	0.4993
HMAC SHA-512	0.5026	0.5034	0.4987
Proposed Technique	0.5160	0.5136	0.5092

3) Discrete Fourier Transform Test (DFT)

The focus of the DFT test is to find the peak heights in the Discrete Fourier Transform of a sequence. It determines the presence of similar patterns in the sequence which further indicates a deviation from the expected randomness. The purpose is to check if more than 5% of the peaks exceed the 95% threshold [12]. The results for DFT test are summarized in Table 5.

Table 5. NIST test results for DFT test

MAC Technique	P-values		
	G	Sodhi	unitedsates
HMAC MD2	0.1443	0.0940	0.3304
HMAC MD5	0.8711	0.5164	0.0744
HMAC SHA-160	0.1468	0.4682	0.0295
HMAC SHA-256	0.4220	0.4220	0.1359
HMAC SHA-384	0.7787	0.7787	0.5121
HMAC SHA-512	0.3723	0.2561	0.6265
Proposed Technique	0.8740	0.7798	0.6318

4) Approximate Entropy Test

This test calculates the frequency of all the overlapping bit patterns present in the sequence. It compares the frequency of overlapping blocks of two subsequent lengths with the expected outcome for a random sequence [12]. The results are given in Table 6.

Table 6. NIST test results for Approximate entropy test

MAC Technique	P-values		
	G	Sodhi	unitedstates
HMAC MD2	0.7464	0.7727	0.7310
HMAC MD5	0.4533	0.8983	0.8863
HMAC SHA-160	0.9288	0.8835	0.9883
HMAC SHA-256	0.8330	0.9440	0.9587
HMAC SHA-384	0.9817	0.9836	0.9865
HMAC SHA-512	0.9949	0.9891	0.9855
Proposed Technique	0.9403	0.9889	0.9874

5) Maurer’s “Universal Statistical” Test

This test is used to find out if a sequence can be compressed without any loss of information. A sequence is said to be random if it isn’t compressible [12]. The results are summarized in Table 7.

Table 7. NIST test results for Maurer test

MAC Technique	P-values		
	G	Sodhi	unitedstates
HMAC MD2	0.9268	0.9528	0.9553
HMAC MD5	0.9831	0.9833	0.9951
HMAC SHA-160	0.9713	0.9600	0.9255
HMAC SHA-256	0.9912	0.9599	0.9705
HMAC SHA-384	0.9774	0.9909	0.9913
HMAC SHA-512	0.9865	0.9909	0.9765
Proposed Technique	0.9987	0.9939	0.9993

As it is observed from Table 3 to Table 7 the proposed scheme performs better by passing the NIST criteria. Thus, indicating its efficiency as a MAC technique.

The objective of MAC is to preserve the integrity of the data and to significantly detect any modification in the received data [13]. Also, every MAC represents specific data content and thus it can significantly notify a change in the data [14]. To study this characteristic, Avalanche test is applied to the MAC values. This test calculates the change in the output with respect to the change in the input, which is known as the avalanche effect and is calculated using the formula as given in Equation (2) [16].

$$\text{Avalanche Effect} = \frac{\boldsymbol{No.of\ bits\ flipped}}{\boldsymbol{Total\ no.of\ bits\ in\ the\ seqence}} \times 100 \tag{2}$$

To apply this test, a single character of the input value is altered, and avalanche effect is calculated. The Avalanche Test results are summarized in Table 8.

Table 8. Avalanche Test analysis

Original Input	Altered Input	No. of bits flipped	Avalanche Effect (%)
G	P	79	49.37
Sodhi	Sodhb	81	50.62
Unitedstates	unitedstraten	76	47.50

It can be clearly observed that the proposed technique performs well under this criteria too, thus demonstrating its efficiency. The proposed MAC technique has higher complexity, which makes it highly resistive towards various attacks on integrity, thus increasing its applicability in networks demanding security [17].

4. CONCLUSION

This paper presents an efficient MAC technique; designed using a novel hash algorithm and a secret key generated using DNA and BBSG. The proposed technique is tested using NIST statistical test suite for random and pseudorandom number generators for cryptography applications and the avalanche criteria. MAC also known as cryptographic checksum is an authentication technique which uses a hash technique along with a secret key to protect integrity of a message and to validate the message. The proposed technique involves the use biometric characteristics along with a novel hash algorithm to frame the MAC. The analysis on the basis of different test results demonstrates that the proposed algorithm performs better than the existing HMAC schemes such as MD2, MD5, SHA-160, SHA-256, SHA-384 and SHA-512. This scheme uses a secret key which involves DNA characteristics of the user, thus making it less susceptible to attacks. The combination of biological characteristics along with mathematical operations make this technique efficient enough to be used under intense security requiring areas, such as military, research, banks, etc.

The proposed MAC can be applied in various cryptographic scenarios for enhanced security and prevention against attacks on data integrity. Furthermore, key generation can be done using other biological characteristics in the form of image, voice, gestures, facial expressions etc. which would improve the uniqueness and originality of the key produced, thus leading to enhanced security.

REFERENCES

[1] Zhong H, Shao L. A Lightweight and Secure Data Authentication Scheme with Privacy Preservation for Wireless Sensor Networks, International Conference on Networking and Network Applications (NaNA). 2016; 210 – 217.

[2] Abdullah N, Saleh, Mohammad A, Ahmad A. Security of a New Cryptographic Hash Function – Titanium, Indonesian Journal of Electrical Engineering and Computer Science, Kuwait, 2018; 10(3): 1244~1250.

[3] Sofia NR, Hmad R, Abdollah MF, Dutkiewicz E. A biometric-based security for data authentication in Wireless body Area Network (WBAN), 15th International Conference on Advanced Communications Technology (ICACT), 2014; 998 – 1001.

[4] Uma M, Anand S. Survey of Source Anonymous Message Authentication Scheme using Analytical and Computational Approach, International Journal of Electronics, Electrical and Computational System, Lakshmangarh, 2015; 7(2).

[5] Koong GS, Yang T, Tseng C. A User Authentication Scheme Using Physiological and Behavioral Biometrics for Multitouch Devices, The Scientific World Journal Research Article, Tung University, Hsinchu, Taiwan, 2014.

[6] Dilli R, Chandra S. Implementation of HMAC-SHA 256 algorithm for hybrid routing protocols in MANETs, IEEE International Conference on Electronic Design, Computer Networks & Automated Verification (EDCAV), 2015; 154 – 159.

[7] Verma S, Prajapati GS. Robustness and Security Enhancement of SHA with Modified Message Digest and Larger Bit Difference. IEEE Symposium on Colossal Data Analysis and Networking (CDAN), 2016; 1-5.

[8] Sodhi GK, Gaba GS. "An efficient hash algorithm to preserve data integrity," Journal of Engineering Science and Technology (ESTEC), 2018; 13(3); 778-789.

[9] Sodhi GK, Gaba GS. "DNA and Blum Blum Shub Random Number Generator Based Security Key Generation Algorithm", International Journal of Security and its Applications (IJSIA), 2017; 11(4): 1-10.

[10] Stallings W. Cryptography and Network Security: Principles & Practices," New York, NY: Pearson Education. 2014; 752.

[11] Eastlake D, Hansen T, RFC, Network Working Group, SHA-160. 2016.

[12] Rukhin A, Soto J, Nechvatal J, Smid M, Barker E, Leigh S, Levenso M, Vange ML, Banks D, Heckert A, Dray J, BasshamIII LE. A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications. 2010.

[13] Hans P, Christian L, Ulrich L, UMAC —A Universal MAC architecture for heterogeneous home networks, IEEE International Conference on Ultra Modern Telecommunications & Workshops, 2009; 1-6.

[14] Mavromati C, Key-recovery attacks against the MAC algorithm Chaskey, Sprnger, International Conference on elected Areas in Cryptography, 2015; 205-216.

[15] Adeshina A, Hashim R, Computational Approach for Securing Radiology-Diagnostic Data Connected Health Network using High-Performance GPU-Accelerated AES, Interdisciplinary Sciences: Computational Life, Springer, 2017; 140–152.

[16] Khalilian R, Rezai A, Abedini E, Khalilian R. An efficient method to improve WBAN security. Advanced Science and Technology Letters. 2014; 43–46.

[17] Maram B, Gnanasekar JM, A Block Cipher Algorithm to Enhance the Avalanche Effect Using Dynamic Key-Dependent S-Box and Genetic Operations. International Journal of Pure and Applied Mathematics. 2018; 399-418.