Journal of Engineering Science and Technology
Vol. 13, No. 2 (2018) 505 - 514

A COMPETENT MAC SCHEME DESIGNED USING A UNIQUE DNA-BRNG KEY AND A NOVEL HASH ALGORITHM

GURPREET K. SODHI, GURJOT S. GABA*

School of Electronics & Communication Engineering, Lovely Professional University, Jalandhar, India - 144411
*Corresponding Author: er.gurjotgaba@gmail.com

Abstract

The current communication sector demands security over anything else. Considering the confidential data transmission and involvement of technology in everyday payment modes, many schemes have been employed in order to achieve this goal. Authentication and integrity form the base of a secure system, and various research techniques have been employed for primarily prevention and secondarily detection of data modification. MAC serves the best purpose when it comes to retention of integrity, and its efficiency is increased on pairing it up with a secret key. In this paper, a new message authentication code (MAC) is proposed based on feature extraction of the user's DNA and Bernoulli Random Number Generator's sequence used as key along with a novel hash algorithm. This scheme ensures integrity and the evaluation is done on the basis of NIST test suite for random numbers, which evaluates the outputs by calculating their P value which has to be greater than 0.01 and strict avalanche criteria. Further the performance of the proposed algorithm is analyzed by comparing it with other existing HMAC schemes available and evaluating its behavior towards various network attacks. The proposed scheme has strong security attributes since it involves the fusion of unique biological characteristics along with technical aspects thus making it applicable for data sensitive environments.

Keywords: Bernoulli random number generation, Data integrity, DNA, Hash, Message authentication code, Security.

1. Introduction

In recent years, with the development of information digitalization techniques, data security becomes the area of prime concern. Due to the requirement of data security when two parties are communicating over an open environment, efficient and robust methods are required to validate the contents of the received messages.

Nomenclatures

C	MAC Function
K	Constant Input, (Fig. 1)
f_t	Function, (Fig. 1)
M	Input Message
S	Substitution, (Fig. 1)
W	Expanded Message Word, (Fig. 1)
$+$	Modulation Addition, (Fig. 1)

Abbreviations

AES	Advanced Encryption Standard
BRNG	Bernoulli Random Number Generator
CMAC	Cipher-based Message Authentication Code
DES	Data Encryption Standard
DFT	Discrete Fourier Transform
DNA	Deoxyribonucleic Acid
HMAC	Hash Based Message Authentication Code
MAC	Message Authentication Code
MD	Message Digest
NIST	National Institute of Standards and Technology
SHA	Secure Hash Algorithm
UMAC	Universal Message Authentication Code

However, data integrity has been addressed significantly and much advancement has been done in this field [1].

MAC (Message Authentication Code) is the basic component of cryptographic technology, it is used to verify the integrity of a message and that the proclaimed identity of the sender is valid [l]. MAC algorithm takes a message and a secret key as input and produces an authentication code as shown in Eq. 1.

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

where, *M*: Input message, *C*: MAC function, *K*: Shared secret key, and *MAC*: Message authentication code.

The receiver is in possession of the secret key and it thus generates the authentication code to verify the integrity of the received message. One way of framing a MAC is to combine a cryptographic hash function with a secret key as an input.

The current paper, presents an efficient MAC (Message Authentication Code) scheme for prevention of data from being manipulated or tampered in between transmission and reception. The algorithm integrates a novel crypto-hash function along with a biometric technique which involves the use of the DNA (deoxyribonucleic acid) characteristics and Bernoulli Random Number Generator (BRNG) sequence [2].

In recent years, different research works on MAC algorithms have been reported, such as Ukkrit [3], presented the implementation of AES (Advanced Encryption Standard) algorithm, on the microcontroller operated on the real-time operating system for securing data in a small scale network. Dworkin [4],

specifies a message authentication code (MAC) algorithm based on a symmetric key block cipher. This block cipher-based MAC algorithm, called CMAC, may be used to provide assurance of the authenticity and, hence, the integrity of data. Dilli et al. [5] implemented the HMAC (Keyed-Hash Message Authentication Code) -SHA 256 Algorithm for the message authentication and Data Integrity. This algorithm was introduced in hybrid routing protocol for Mobile network environment. Also, Loeb et al. [6] introduced the heterogeneity flexible computing platform for the network nodes, i.e., the Universal MAC (UMAC) using Universal Hashing (UMAC) [6]. Verma et al. [7] proposed an algorithm to enhance the security of SHA scheme by modifying the message digest and introduction of larger bit difference. Most of these algorithms were based on block ciphers and hash functions. Recently, a new algorithm named Chaskey has been presented [8].which is a permutation-based MAC algorithm for 32-bit microcontrollers.

Security involves three requirements Authenticity, Confidentiality and Integrity. While authenticity indicates that the source and message are authorized and intended, confidentiality prevents unauthorized access of the data and integrity is about detecting if an adversary has modified the contents of the data received. Most common methods of providing integrity are based on using a shared key and a hash algorithm forming a Message Authentication Code or MAC. In this case, the sender sends data with an appended tag which is computed as a function of the data called the message digest or hash value and the secret key. The receiver then re-computes the hash from the received data and compares it with the one received. If both are same then integrity is verified else data is considered as forged. Message Authentication Code may be constructed from block cipher like DES (Data Encryption Standard), for example MAC, or cryptographic hash function like SHA, for example HMAC [5]. Aimed to this motivation, a new message authentication code is presented which utilizes biomedical features and BRNG sequence as the key integrated with a unique hash algorithm. This scheme finds its applicability in army, banks and e-commerce sectors.

2. Proposed Algorithm

Data integrity can be maintained by using MAC which constitutes of a message input, a secret key and a hash algorithm. The presented technique uses a novel hash technique which follows the basic structure of SHA-160 integrated with an 'f' function along with a secret key produced using DNA sequence and BRNG output random sequence. The details are explained in the following subsections:

2.1. Novel hash algorithm

The novel hash algorithm used in this scheme is a result of the incorporation of 'f' function in the basic structure of SHA-160. There are total 80 rounds in SHA-160 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 resister values (A, B, C, D, E).

The structure of the hash algorithm is explained using Fig. 1.

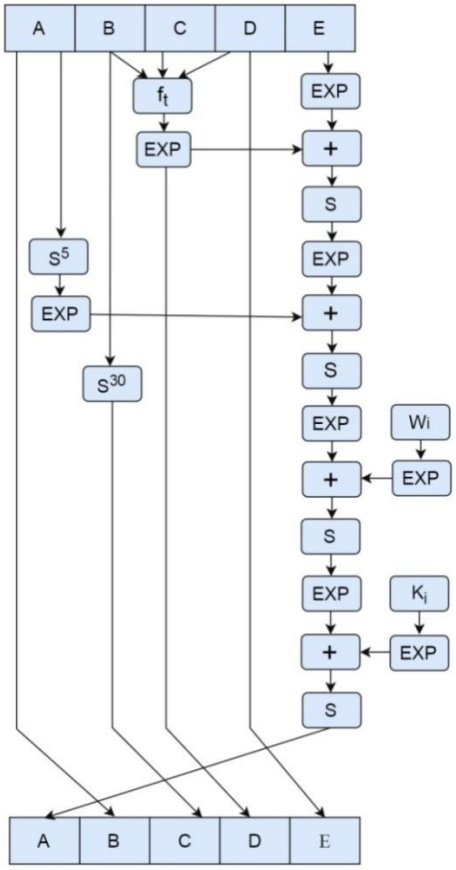

Fig. 1. A Novel Hash Algorithm.

2.2. DNA-BRNG based secret key

The hash algorithm is applied on the message inputs along with the secret key. The secret key used in the proposed scheme is derived from the characteristics of DNA. The DNA is represented in the form of a sequence constituting of 'agct' characters following a unique pattern for every individual. The characteristic uniqueness of a DNA sequence makes it impossible to be duplicated or stolen.

Further to increase the efficiency of the secret key, BRNG is used to produce an output sequence which is a 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 BRNG sequence. This results into a 256-bit key, which is used in the MAC [2].

2.3. Formation of MAC

MAC is also known as keyed Hash in reference to its components, i.e., a hash algorithm and a secret key. The DNA-BRNG key here is of 256 bits and this key has to be further used in the form of four 32-bit keys, therefore the operations applied on the 256-bit key to convert it into four 32-bit keys are explained using Fig. 2.

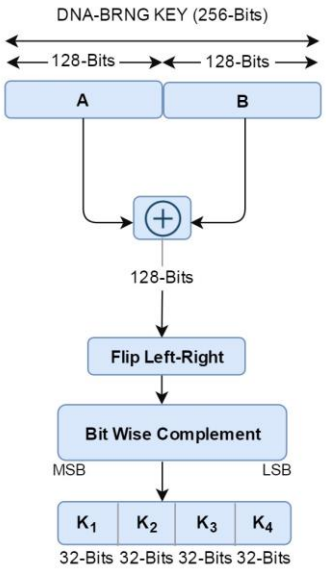

Fig. 2. Operations applied on DNA-BRNG key.

The four final keys are in hexadecimal form as shown in Table 1. The SHA-160 algorithm uses four 32-bit constant values [9] which are replaced with the four keys that were framed using the operations. This forms a novel MAC algorithm. The MAC values obtained, using the proposed technique are presented in Table 2.

Table 1. Security Keys (Hexadecimal)

S.no	32-bit Keys
K_1	E052642F
K_2	9ED92359
K_3	57EDCC22
K_4	92A58D8D

Table 2. MAC Values.

Input	Hex form	MAC Values (Hexadecimal)
g	67	a15118eef9ee8f1159656890c691c7979164110a
Sodhi	536f646869	11ec223fea6b36c2d93893d65ef1369cc8e21864
unitedstates	756e69746564 737461746573	ae1d2843dcc97133003918c0ebd47c4f583e01c9

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

3. Results and Discussions

The presented scheme is accessed on the basis of NIST tests of randomness and avalanche criteria. This is done for three different inputs values having varying lengths. These tests are used to compute the P-value for a binary sequence; this value must be greater than 0.01 for a sequence to be random [10].

In order to validate the efficiency of our proposed algorithm, the NIST results of the proposed MAC scheme are compared with those of the existing ones. The key used for these algorithms is '3A54E26B', which is kept constant for all the traditional techniques.

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

i) Frequency test

Frequency test analyses the ratio of the number of ones and zeros in the entire sequence. It checks the proximity between the number of ones and number of zeros. A sequence is said to be random if the proportion of both is close to each other [10]. The results in Table 3 illustrate that the proposed algorithm produces better proximity between the count of ones and zeros as compared to other schemes.

Table 3. NIST test results for Frequency test.

MAC	**P-values**		
Technique	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	0.8917	0.9744	0.9303

ii) Binary derivative test

The Binary Derivative Test proceeds by applying exclusive-or operation between consecutive bits of a sequence until only one bit is left. Then, the ratio of number of ones to the total length of the sequence in each case is calculated. Finally, the average of the ratio for all the sequences is calculated, if this value lies near to 0.5, then the sequence is said to be random [10]. The results in Table 4 depict that the output of the proposed algorithm is random.

Table 4. NIST test results for Binary Derivative test.

MAC	**P-values**		
Technique	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	0.5240	0.5186	0.5112

iii) Discrete Fourier transform test (DFT)

The objective of DFT test is to find the peak heights in the Discrete Fourier Transform of a sequence. It determines the presence of identical patterns in the sequence which further indicates a deviation from the assumed randomness. The

purpose is to check if more than 5% of the peaks exceed the 95% threshold. The results for DFT test are summarized in Table 5.

Table 5. NIST test results for DFT test.

MAC Technique	P-values		
	g	Sodhi	Unitedstates
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	0.8730	0.8068	0.6437

iv) Approximate entropy test

The aim of this test is to calculate the frequency of all the overlapping bit patterns existing in the sequence. It compares the frequency of overlapping blocks of two sequential lengths with the expected outcome for a random sequence. 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	0.9390	0.9893	0.9880

v) Maurer's "Universal statistical" test

This test focuses on finding out if a sequence can be compressed without any loss of information. A sequence is said to be random if it is not compressible [10]. 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	0.9967	0.9930	0.9968

As clearly observed from Table 3 to Table 7, the proposed technique performs better by passing the NIST criteria of generating a random MAC. Thus, indicating its significance as a MAC Scheme.

The objective of MAC is to protect the integrity of the data and to significantly detect any alteration in the message. Also, a particular MAC is in accordance with particular data content and thus it can significantly indicate a change in the data. Thus, there is a change in a particular MAC value following a change in the data file. To study this parameter, another test has been applied to the MAC values, this is the Avalanche Test. This test calculates the change in the output with respect to a change in the input, which is referred to as the avalanche effect and is calculated using the formula as given in Eq. (2).

$$\text{Avalanche Effect} = \frac{\textit{No.of bits flipped}}{\textit{Total no.of bits in the seqence}} \times 100 \quad (2)$$

The higher the avalanche effect better is the efficiency of the technique. This test has been applied by altering a single character of the input value, it is applied on the traditional techniques and the outcomes are compared with the ones obtained for the presented scheme. 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	70	43.75
Sodhi	Sodhb	83	51.87
Unitedstates	Unitedstraten	87	54.37

It can be clearly noticed that the recommended technique performs well under this criteria too, thus demonstrating its efficiency.

The increased complexity of the technique makes it highly resistive towards various network attacks on data integrity, a brief summary analysing the behaviour of the technique is presented in Table 9.

Table 9. Resistance against attacks.

Network Attacks on Integrity	Preventive Features
Salami attacks	As observed from the avalanche test analysis, a small change in the input results in a major change in the output. Hence, even the minute modification in the data would be detected.
Data diddling attacks	Since the proposed scheme uses biological characteristics to frame the secret key, therefore it is not possible for the data to be modified by an unauthorised party.
Man-in-the-middle attacks	The proposed technique is hash based, thus it is highly resistive towards any changes in the data by an unauthorised party.
Seed Attacks	Keys generated using only the random number generator outputs are susceptible towards seed attacks, the key used in the proposed MAC is a result of fusion of DNA and random number generator output, thus increasing its resistance towards seed attacks.

The proposed MAC algorithm is complex and therefore is highly resistive towards various attacks on integrity, thus increasing its applicability in a data sensitive environment.

4. Conclusion

This paper presents an efficient MAC Scheme which is a result of a novel hash algorithm and a secret key of 256-bits generated using DNA and BRNG. The recommended technique has been analysed using NIST statistical test suite for random and pseudorandom number generators for cryptography applications and strict avalanche criteria. The conclusions drawn are listed as:

- 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 validate the sender.
- The proposed scheme involves the use biometric characteristics along with a novel hash algorithm to frame the MAC and thus enhances its reliability.
- The analysis of the different test results concludes 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 considerably more reliable against attacks.
- The proposed MAC is resistive towards various attacks on integrity and can be applicable in various cryptographic techniques for better security and provision of integrity in data sensitive areas like army, banks etc.

References

1. Stallings, W. (2014). *Cryptography and network security: Principles & practices*, New York, NY: Pearson Education.
2. Sodhi, G.K.; and Gaba, G.S. (2017). DNA and Bernoulli random number generator based security key generation algorithm. *Pertanika Journal of Science and Technology*, 25(3), 891-904.
3. Ukrit, A. (2017). An AES cryptosystem for small scale network. *Proceedings of the IEEE Defense Technology* (ACDT), 49-53.
4. Dworkin, M.J. (2015). Recommendation for block cipher modes of operation: the cmac mode for authentication. *National Institute of Standards and Technology (NIST), Special Publication*.
5. Dilli, R.; and Chandra, S.R.P. (2015). Implementation of HMAC-SHA 256 algorithm for hybrid routing protocols in MANETs, *Proceedings of the IEEE* International Conference on Electronic Design, Computer Networks & Automated Verification (EDCAV), 154 – 159.
6. Leob, H.-P.; Liß, C.; Rückert, U.; and Sauer, C. (2009). UMAC - A Universal MAC architecture for heterogeneous home networks. Proceedings of the IEEE International Conference on Ultra Modern Telecommunications & Workshops, 1-6.
7. Verma, S.; and Prajapati, G.S. (2016). Robustness and security enhancement of sha with modified message digest and larger bit difference. *Proceedings of the IEEE, Symposium on Colossal Data Analysis and Networking* (*CDAN)*, 1-5.

8. Mavromati, C. (2015). Key-recovery attacks against the MAC algorithm Chaskey, *Springer, International Conference on elected Areas in Cryptography*, 205-216.
9. Eastlake, D.; and Hansen, T. (2006). RFC, Network Working Group, SHA-160.
10. Rukhin, A.; Soto, J.; Nechvatal, J.; Smid, M.; Barker, E.; Leigh, S.; Levenson, M.; Vange, M.; Banks, D.; Heckert, A.; Dray, J.; and Vo, S. (2010). A statistical test suite for random and pseudorandom number generators for cryptographic applications. *National Institute of Standards and Technology (NIST), Special Publication* 800-22, Revision 1a