International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Index Copernicus Value (2016): 79.57 | Impact Factor (2015): 6.391

MMAC: Fast and Secure Message Authentication

Mohammed Ali Mohammed[1], Loay K. Abood[2], Makki Maliki[3]

[1, 2, 3]Baghdad University, College of Science, Baghdad, Iraq

Abstract: ***The Mathematical Message Authentication Code (MMAC) is a well-known method to provide integrity of message. This works aims to proof that the MMAC is more reliable than exciting/slandered algorithms. The comparison will take place based on describing the analysis result of the aspects of: fast, complexity, and security with other algorithms. These algorithms are: CMAC-AES (Cipher-Based Message Authentication Code-Advanced Encryption Standard), MAC-Triple-DES (Message Authentication Code-Triple-Data Encryption Standard) and CMAC-DES (Cipher-Based Message Authentication Code-Data Encryption Standard) MACs algorithm. While we give HMAC-SHA-128 (Secure Hash Algorithm) HMAC algorithm more comparable attention for it's important. The dataset is a text which written in English language. There are three dataset have a different size: the first one is "The Irish Penny Journal" book, while the second dataset is "Notes of hospital life" book, and third dataset is "The Life of Robert" book. The result of the comparison shows without any doubt that the MMAC algorithm is faster, more secure and less complexity.***

Keywords: MMAC, Mathematical Message Authentication Code, Message Integrity, Message Authentication, MAC

1. Introduction

Message Authentication Code (MAC) is a keyed hash function, which is used to provide the message authentication. By using the Message authentication service, the receiver can ensure that the message received doesn't have any modification, insertion, or deletion during the transmit process.

The properties and requirements of the MAC algorithms are: accept any length of input message, the MAC (tag) generated should be Fixed length, consist shared secret key between the sender and receiver, Strict Avalanche Criterion (SAC), the algorithm should be one way (many to one), weak collision resistant, easy to compute the MAC value, and finally Pseudo Randomness value [1] [2].

The sender computes the MAC value which can be defined as:

$$MAC = C(K, M)$$

Where:
MAC: Message Authentication Code,
C: MAC Function,
K: Shared Secret Key,
M: message.

Then the sender transmits the message and MAC to the receiver. In receiver side, the MAC` is computed for the received message and then the MAC and MAC` are matched. Match process is to clarify that the received message is modified or not during the transition phase. Figure 1: Sender and Receiver Flow Processshows the flow system of the send/receive processing.

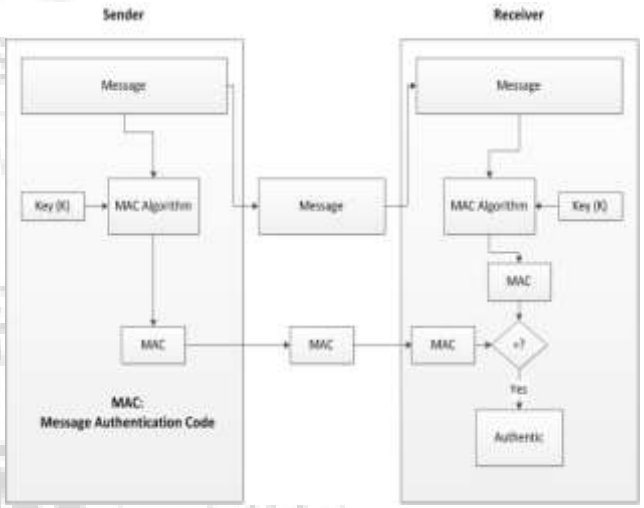

Figure 1: Sender and Receiver Flow Process

To test/prove the efficient result of the MMAC algorithm, three datasets are used. These dataset have different size: the first one is "The Irish Penny Journal" book when the file size is 90KB and the number of block is 5502. While the second dataset "Notes of hospital life" book when the file size is 273KB and the number of block is 16782. Third dataset "The Life of Robert" book when the file size is 567KB and the number of block is 35068. The dataset is a text which written in English language, while the block size is 128bits (16bytes).

The next section is organized as follow: the literature review of previous works is presented in next section. Then the MMAC algorithm will be describes in the section 3. The result and discussion are discussed in section 4. Finally, conclusion will be presented in section 5.

2. Related Work

Most interested researchers who presented many MAC algorithms give careful attention to one or more of the aspects of security, fast, and/or complexity. We present in this paper most related work done so far in this field:

Licensed Under Creative Commons Attribution CC BY

Wadhwa et al. [2] proposed an algorithm called Combined Method for Confidentiality, Integrity, Availability and Authentication (CMCIAA) which is based on the key and AES algorithm. This method does not generate a fixed length of MAC for all variable length of a message as well. It generates ¼ of the length of original message. While the standard properties of MAC algorithm obtain to generate a fixed output length.

Petrank and Rackoff [3] proposed EMAC (Encrypted Message Authentication Code) algorithm which is require two shared secret key. in this algorithm, the output value of CBC-MAC algorithm will be encrypting to increase the performance. EMAC is secure if the length of data is a positive multiple of the block size; else the EMAC is not secure.

Kurosawa and Iwata [4] describe the Two-key CBC-Message Authentication Code (TMAC). TMAC generated from the XCBC with two keys rather than three keys. TMAC change (K2,K3) with (k2 multiplication with some non-zero, K2). TMAC algorithm is vulnerable to second pre-image attack. Therefore, the TMAC is not secure.

Also the Kurosawa and Iwata [5] describe the One-Key CBC-Message Authentication Code (OMAC) which take only one key (K). They are two types of OMAC: OMAC1 and OMAC2 which are getting from the XCBC type by changed the keys number. Also the OMAC algorithm is vulnerable to second pre-image attack.

Minematsu and Tsunoo [6] this paper proposed PC-MAC and MT-MAC algorithm to increase the performance. These algorithms are secure only if the block cipher used the: pseudorandom function and small differential probability for additional permutation. These algorithms are secure and faster than any MAC algorithm because, of using the AES algorithm with 4-rounds.

Mouha et al. [7] present the Chaskey algorithm, which is Message Authentication Code algorithm, for 32-bit microcontrollers. Chaskey algorithm is used for any service that needs 128-bit security. Chaskey is derived from the Chasqui (or Chaski). It is secure in the standard model and faster than MAC algorithm.

Ferguson [8] presented multi algorithms: Black, John, et al. [9] present UMAC (Universal MAC), Bernstein, Daniel J [10] present Poly1305-AES, and Dworkin Morris J [11] present GMAC. These MAC algorithms based on the universal hash function and based on nonce number (one time used or forgery attack happened). If the MAC value are truncated then the GMAC and Ploy-1305 is insecure.

Nandi [12] builds efficient and secure Cipher Block Chaining-Message Authentication Code (CBC-MAC) algorithms called GCBC1 and GCBC2. The length extension attack can happen when the input message is variable length.
Jia et al. [13] proves that the CBC-MAC without truncation function is insecure for variable length messages. Also prove that the: OMAC, TMAC, MT-MAC, PC-MAC, XCBC, EMAC, ECBC, three-key encipher CBC mode, CMAC, CBC-MAC, FCBC are vulnerable to second pre-image attack.

3. Mathematical Message Authentication Code

MMAC is new algorithm which design and implementation to provide the message integrity. MMAC based on the S-Box as shared secret key between sender and receiver. The generation of the S-Box based on the Pseudo-Random Number Generators and shared secret keys. MMAC consist of four steps shown in Figure 2*: MMAC algorithm Steps*: Generate the S-Box Key step, Pre-processing step, substitution step, and mathematical step [14].

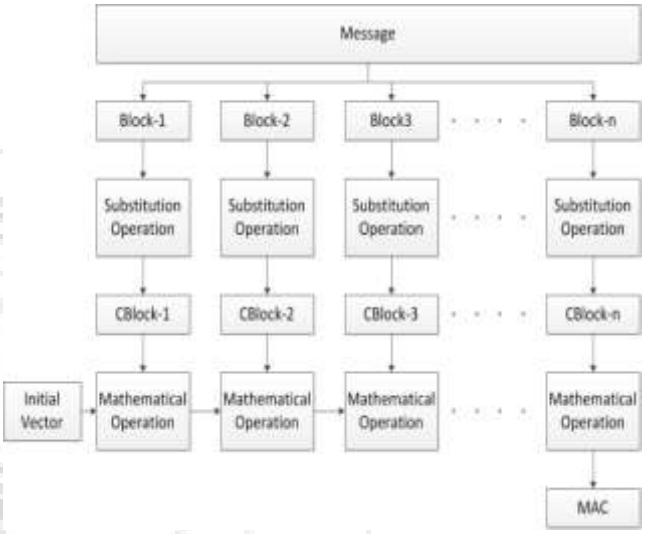

Figure 2: MMAC algorithm Steps

The main characteristics of the MMAC are: simple mathematical equation rather than using complex equation and the MAC value generated from the cipher-text not from the plain-text (original text). The substitution operation is based on the: S-Box table, block number, index of character and original character.

The conclusion of MMAC is faster and secure with the properties: simple equation in the mathematical step, without using AES or DES algorithm, and single round. Finally, the MMAC is also used to ensure that the message received was sent by the authorize part which called User Authentication.

4. Result and Discussion

To prove the efficient and robust of the MMAC algorithm, the result should be analysis according to the three aspects: Fast, Secure, and Complexity.

4.1 Fast

The execution time of algorithm based in the many parameters such as dataset size and number of steps of the algorithm. We calculate the execution time from 10-times execute of the program.

We used three dataset which are different in a size to compare the execution time for these algorithms: MMAC,

Licensed Under Creative Commons Attribution CC BY

CMAC-DES (Cipher- based Message Authentication Code – Data Encryption Standard), CMAC-AES (Cipher – based Message Authentication Code – Advanced Encryption Standard), MAC-Triple-DES (Message Authentication Code – Triple – Data Encryption Standard), and compare with HMAC-SHA-128 (Secure Hash Algorithm) HMAC algorithm.

The execution time of these algorithms when use the small dataset (size = 90KB, 5502 block number) are shown in table 1.

Table 1: execution time of small dataset

No	Algorithm Name	Execution Time (MS)
1	MMAC	8.9
2	CMAC-DES	10.3
3	CMAC-AES	25.8
4	MAC-Triple-DES	20.3
5	HMAC-SHA-128	15.1

Figure 3 present the result of table 1.

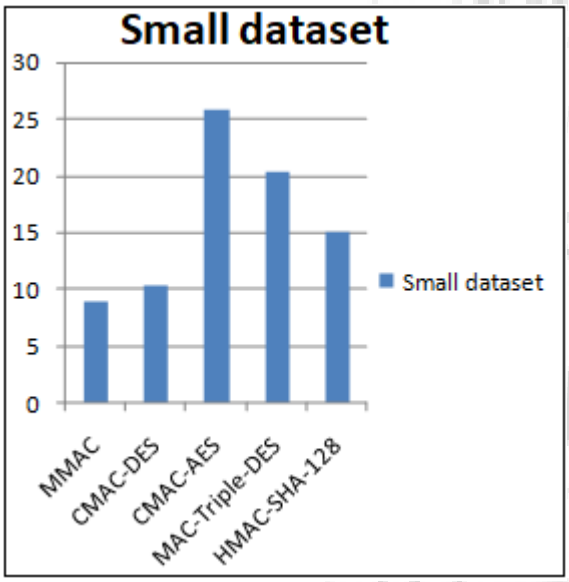

Figure 3: shows the execution time of small dataset

And the execution time of these algorithms when use the medium dataset (size = 273KB, 16782 block number) are shown in table 2.

Table 2: execution time of medium dataset

No	Algorithm Name	Execution Time (MS)
1	MMAC	28.2
2	CMAC-DES	32.2
3	CMAC-AES	52.3
4	MAC-Triple-DES	52.4
5	HMAC-SHA-128	36.5

Figure 4 present the result of table 2.

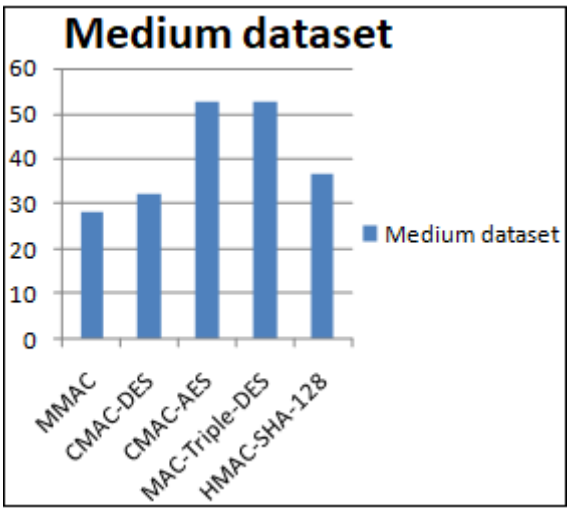

Figure 4: shows the execution time of medium dataset

Finally, the execution time of these algorithms when use the large dataset (size = 567KB, 35068 block number) are shown in table 2.

Table 3: Execution time of large dataset

No	Algorithm Name	Execution Time (MS)
1	MMAC	69.1
2	CMAC-DES	72.7
3	CMAC-AES	110.9
4	MAC-Triple-DES	112.6
5	HMAC-SHA-128	77.5

Figure 5 present the result of table 3.

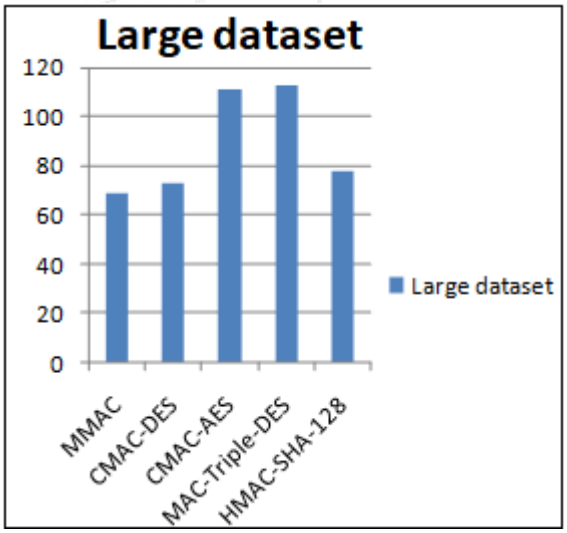

Figure 5: shows the execution time of large dataset

The significant less time execution for MMAC came as a result using:

a) In CMAC-DES and MAC-Triple-DES algorithms contain rounds and complex steps because these algorithms used DES algorithm in one step. So, the execution time of these algorithms is higher than MMAC algorithm.
b) The execution time of CMAC-AES algorithm is also higher than the MMAC algorithm due, the CMAC-AES contain the AES algorithm in one step.
c) The execution time of MMAC algorithm is less than HMAC-SHA-128 algorithm due, the HMAC-SHA-128

Licensed Under Creative Commons Attribution CC BY

includes two executions of the hidden hash function for every output block.

4.2 Complexity

The MMAC algorithm has less complexity than other algorithms because:

a) CMAC-AES algorithm use AES algorithm in one step. The throughput of AES algorithm is less compared with other algorithms.
b) MAC-Triple-DES and CMAC-DES algorithms use DES algorithm in one step. DES consist complex steps.
c) HMAC-SHA-128 algorithm has two hidden hash function in each block. Therefore, the HMAC-SHA-128 is large complex than MMAC algorithm.
d) MMAC algorithm has less complexity compared with other algorithms because, simple generate S-Box table and simple substitution operation which based on the index of character, block number, and character.

4.3 Secure

The security of MMAC algorithm represented in the:

a) Generate secure sub-key (S-Box table).
b) Use shared secret key rather than use 0's or 1's to complete the final block if it is not completed in the preprocessing step. The reason of use shared secret key to ensure: changes from socket to another, and not repeated.
c) Also use the shared secret key to initial the Initial Vector (IV) in CBC-Mode.
d) The MMAC algorithm was successfully tested from these attacks: Block re-ordering attacks, truncation attack, mix and match attack, brute-force attack, cryptanalysis attack, birthday attack, second pre-image resistance attack, and man in the middle attack.

5. Conclusion

After compared the result of MMAC algorithm with the result of CMAC-AES, CMAC-DES, MAC-Triple-DES, and HMAC-SHA-128 algorithm from the security, complexity, and execution time we conclude is that the MMAC algorithm is efficient (more reliable) than other algorithms.

References

[1] William Stallings, Cryptography and network security: principles and practices.: Pearson Education India, 2006.
[2] Neeta Wadhwa, Syed Zeeshan Hussain, and S. A. M. Rizvi, "A Combined Method for Confidentiality, Integrity, Availability and Authentication (CMCIAA)," in Proceedings of the World Congress on Engineering, vol. 2, 2013.
[3] Erez Petrank and Charles Rackoff, "CBC MAC for real-time data sources," Journal of Cryptology, vol. 13, pp. 315-338, 2000.
[4] Kaoru Kurosawa and Tetsu Iwata, "Tmac: Two-key cbc mac," in CT-RSA, vol. 2612, 2003, pp. 33-49.
[5] Tetsu Iwata and Kaoru Kurosawa, "omac: One-key cbc mac," in FSE, vol. 2887, 2003, pp. 129-153.
[6] Kazuhiko Minematsu and Yukiyasu Tsunoo, "Provably secure MACs from differentially-uniform permutations and AES-based implementations," in International Workshop on Fast Software Encryption, 2006, pp. 226-241.
[7] Nicky Mouha et al., "Chaskey: an efficient MAC algorithm for 32-bit microcontrollers," in International Workshop on Selected Areas in Cryptography, 2014, pp. 306-323.
[8] Niels Ferguson, "Authentication weaknesses in GCM," Comments submitted to NIST Modes of Operation Process, 2005.
[9] John Black, Shai Halevi, Hugo Krawczyk, Ted Krovetz, and Phillip Rogaway, "UMAC: Fast and secure message authentication," in Annual International Cryptology Conference, 1999, pp. 216-233.
[10] Daniel J. Bernstein, "The Poly1305-AES Message-Authentication Code.," in FSE, vol. 3557, 2005, pp. 32-49.
[11] Morris J. Dworkin, "SP 800-38D. Recommendation for block cipher modes of operation: Galois/Counter Mode (GCM) and GMAC," National Institute of Standards & Technology, 2007.
[12] Mridul Nandi, "Fast and Secure CBC-Type MAC Algorithms.," in FSE, vol. 5665, 2009, pp. 375-393.
[13] Keting Jia, Xiaoyun Wang, Zheng Yuan, and Guangwu Xu, "Distinguishing and Second-Preimage Attacks on CBC-Like MACs.," in CANS, vol. 5888, 2009, pp. 349-361.
[14] Mohammed Ali Mohammed, Loay K. Abood, and Makki Maliki, "Mathematical Message Authentication Code Using S-Box key," IJCSNS, vol. 17, p. 31, 2017.

Author Profile

Mohammed Ali received the B.S. in computer science from university of Baghdad in 2013. master student currently. He has patent reward in 2016. He has appreciation letter from the president of Karkh University of science. Holds the first place in the scientific competition sponsored by the ministry of youth and sports held at the University of Baghdad. Also he holds 3 books of thanks and appreciation from the Dean of the Faculty of Science.

Loay K. Abood received the M.S. and Ph. D. degrees in Physics Science from University of Baghdad in 1993 and 1999, respectively. During 2007-2010 work as head of computer science department. Currently he works as Assistant President for Scientific Affairs, Karkh University of science.

Makki Maliki received the M.S. in computer science from University of Jordan in 2003. and received the Ph. D in computer science from university of Buckingham UK in 2015. Currently *instructor in the University of Baghdad. Iraq.* Interesting Area: Image processing, Pattern recognition, Writer identification, OCR, Medical Image, and Biometrics.

Licensed Under Creative Commons Attribution CC BY