I.J. Mathematical Sciences and Computing, **2018, 4, 34-41**
Published Online November 2018 in MECS (http://www.mecs-press.net)
DOI: 10.5815/ijmsc.2018.04.03

Available online at ***http://www.mecs-press.net/ijmsc***

Modification on AES-GCM to Increment Ciphertext Randomness

Ahmad S. Bader [a], Prof Dr. Ali Makki Sagheer [b]

[a] *Technical institute of Anbar, Middle Technical University, Baghdad, Iraq*
[b] *Al-Qalam University College, Kirkuk, Iraq*

Received: 20 April 2018; Accepted: 06 August 2018; Published: 08 November 2018

Abstract

Today, there are many cryptographic algorithms that are designed to maintain the data confidentiality, from these algorithms is AES. In AES-GCM, the key in addition to the IV are used to encrypt the plaintext to obtain the ciphertext instead of just the key in the traditional AES. The Use of the IV with the key in order to gain different ciphertext for the same plaintext that was encrypted more than ones, with the same key. In this paper, the mechanism of change the IV each time in AES-GCM was modified to get more randomness in the ciphertext, thus increase the difficulty of breaking the encrypted text through analysis to obtain the original text. NIST statistical function were used to measure the randomness ratio in the encrypted text before and after modification, where there was a clear rise in the randomness ratio in the encoded text which obtained by using the modified algorithm against ciphertext by using the normal AES_GCM.

Index Terms: AES, GCM, AES-GCM, Ciphertext Randomness.

1. Introduction

Nowadays, Cryptography enables individuals to extend the certainty found in the physical world to the electronic world, hence enabling individuals to work together electronically without stresses of deception. Regular a large number of individuals' associates electronically, regardless of whether it is through email, web based business, ATM machine or mobile phones. The unending increment of data transmitted electronically has prompted an expanded dependence on cryptography.

There are two types of encryption algorithms that are asymmetric and symmetric. In asymmetric a pair of keys (private key and public key) are used, one for encryption and the other for decryption, such as the RSA algorithm. In symmetric, the same key is used for encryption and decryption, for example, DES and AES algorithm [1].AES is one of the most popular block cipher encryption algorithms, where it has not yet been proven that this algorithm has been broken [2].

* Corresponding author.
E-mail address: ahm.salim@uoanbar.edu.iq, dean@alqalam.edu.iq, prof.ali@alqalam.edu.iq

There are many attacks that aim to break encryption algorithms to get the original text through ciphertext. Some of these attacks analyze the encrypted text in order to obtain the plaintext, and the factors that help the success of these attacks weak key used in the process of encryption or repetition of certain text within the plaintext more than once. The cipher values of an encryption algorithm are randomized using several diffusion elements such as addition, rotation, transposition, etc. Such operations on diffusion elements are repeated several times or several rounds for achieving sufficient diffusion level.

In order to obtain different encrypted texts for the same plaintext encrypted with same key, the stream cipher modes were used, such as OFB, CFB and CTR [3].One of the modifications introduced to AES by using the stream modes is the AES-GCM algorithm, where which uses the IV Xor with the key each new block in order to obtain new cipher text even if the plaintext and the key used in previous block encryption are same. This change occurs because added one to the IV in each encryption process for a new block.

There are many modification that have been proposed on the AES which aim to enhance algorithm security or to improve time complexity of the algorithm. In this paper, a new mechanism has been proposed for the change of the IV in each encrypting process for a new block, where in addition to the increase by one, the IV is rotate shift by one and this gives more randomization in the ciphertext even if the whole text is composed of the same character and encrypted with the same key.

2. Related Works

Yue et. al., in 2011[4], proposed mathematical model to increase the randomness in the number of changing pixel rate (NPCR) and the unified averaged changed intensity (UACI). The suggested work consider both scores as random variables under the ideally encrypted image model and derive their expectations and variances. The proposed model was applied on a set of images and the results were good.

Vandanav, in 2012 [5], suggested replace the MixColumn process in traditional AES with the permutation process depending on the permutation table. The proposed modified was aimed to get more speed to use the modified algorithm in multimedia encryption systems in effective manner.

Vaidehi et. al., in 2015 [6], proposed used Common Sub-expression Elimination (CSE) algorithm to improve the MixColumn process in order to reduce the hardware complexity and energy consumption and thus improve the performance of the algorithm.

Soukaena et. al., in 2016 [7], proposed a modified RC4 encryption algorithm for greater confidentiality. In the modified algorithm, the randomization ratio of the key which used in the encryption process is increased by 20%. This has helped to increase the randomness ratio of encrypted text and thus obtain greater security when using the modified algorithm in data encryption.

Ammar et. al., in 2017 [8], suggested to design a secure chatting application using a modified AES to encrypt secret data. The modified algorithm used Cipher Block Changing (CBC) encryption Mode by inserting the ciphertext of each block as an Integrated Vector (IV) with the key in the next block encryption process to get different text for similar text encoded with the same key and also to obtained more randomness.

3. Advanced Encryption Standard (AES)

In 1997, specifically on September 12 announced the National Institute of Standards and Technology (NIST) a “Request for Candidate Algorithm Nominations for the Advanced Encryption Standard.” The aim of AES was to replace the Data Encryption Standard (DES) as a new cipher standard. Roughly the more processes associated with the AES selection were different from DES operation. Pre-conditions have been setting for the size of the block to be encrypted and the key size. The aim of the algorithm was to work with several key sizes (128, 192, and 256) in order to meet the security requirements at that time and even in the future. In addition, regulators set several criteria for the proposed algorithm to be selected: security, flexibility, simplicity, cost, and work on all hardware and software. It was invited the contestants from all over the world to participate either as reviewers or submitters [9, 10].

On June 15, 1998, 21 algorithms were submitted on the day of the competition, 15 of algorithms which were said to have met the specific conditions of the contest. 10 of these 15 algorithms were established outside the United States, and there was at least one designer in each algorithm non-US. In the eighth month of 1999, NIST restricted the contest between 5 algorithms for final decomposition. The winning algorithm was known as "Rijndael", it was introduced by two Belgian researchers, Vincent Rijmen and Joan Daemen. The algorithm was formally adopted on the 26th of the 5th month in 2002 [10]. In fact, there are no serious security gaps in any of the 5 algorithms in the finals, but the winning algorithm was selected based on criteria such as efficiency, flexibility and performance on various devices, in addition to other characteristics [11].

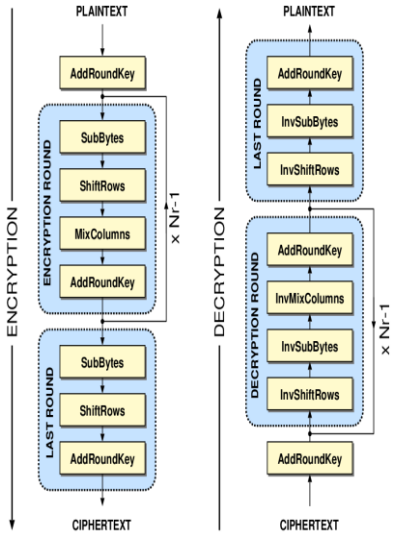

Fig.1. AES Algorithm Steps [12].

AES is a block cipher, the length of the input block is 128 bits while the key length is 128, 192 or 256 bits. The key length determines the number of rounds required for encryption but does not affect the overall structure of each round. Unlike DES which depends on the Feistel structure fully, AES is essentially a permutation-substitution network. At the processes in the AES, a 4*4 byte array called the state is entered and modified in a sequence of rounds. Where the state consists of partitioning the 128-bit entrance block to 16 partitions each 8-bit partition (16 bytes). The following is an explanation of the operations that conduct on the state in each round since the introduction of the plaintext until the ciphertext is accessed [11]:

Stage 1). AddRoundKey: In the AES algorithm, in each round there is an XOR process between the state array and (128-bit) the assigned key for each round, where a round key is derived for each round from the main key which is used in the first round only.

Stage 2). SubBytes: At this point, each byte is replaced in the state array with new byte, this process is done depending on a custom table for this.

Stage 3). ShiftRows: In shiftrows, all the bytes in the state array are shifted except the first row bytes, where the bytes are in the first row stay as they are, the bytes in the second row shift to the left one time, the third row shift left twice and the fourth row shifts left three times. The conversion process is periodic so most of the bytes in the state array are changed.

Stage 4). MixColumns: One of the most important steps in each round in AES algorithm is MixColumn, in this step, a transform is applied in order to affect each column present in the state array (The transformation resulting from this process is a linear transformation).

In AES algorithm, because the last three steps in each round can be inverted, MixColumns is replaced with AddRoundKey in the final round, this prevents the attacker from reversing the last three steps.

In the cryptographic algorithms of block cipher type, in the AES algorithm, there are four different steps, three of which are substitution and one is permutation. [9]:

- Substitute bytes: Used for the substitution process.
- ShiftRows: Helps to make the permutation process.
- MixColumns: make the substitution through the arithmetic operation over GF (28).
- AddRoundKey: XOR process for the current block with the key derived from main key.

4. Galois Counter Mode (GCM)

GCM is a block cipher mode that provides data authentication in addition to data encryption, uses one of the block cipher encryption algorithms in addition to a counter mode (CTR). The authentication process is done by using Hash Functions through binary Galois Field to authenticate the encrypted message [13].

AES-GCM algorithm is a collection between the AES Counter Mode encryption and the Galois Hash authentication algorithm, produces encrypted text as well as an authentication tag. AES-GCM consists of three stages: Pre-processing (encryption, authentication); Processing Loop and Post Processing. AES-GCM is described in (Fig. 2) [14, 15].

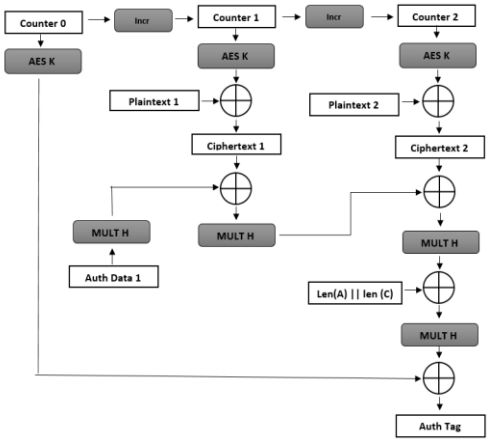

Fig.2. AES-GCM Model [9].

5. Proposed Modification

In a normal AES, if the same key is used to encrypt the same plaintext, it produces the same encrypted text, which makes it easier to analyze and break the ciphertext So IV is used to ensure that the ciphertext is changed each time the same text is encrypted with the same key.

In AES-GCM, adding one to IV every new encryption changes the encrypted text, but to a small extent. So, there is a need to find a method to ensure the greatest possible change to the encrypted text. Therefore, to ensure the largest possible change, the IV is rotated shift by one with each incremental operation, this helps to use a very different IV in each encryption process, thus, a different ciphertext. Figure (3) shows the modified AES-GCM algorithm.

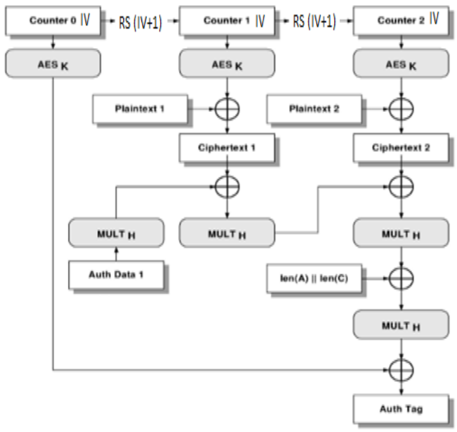

Fig.3. The modified AES-GCM.

6. Results and Discussion

By default if tries hackers attack the ciphertext resulting from the method suggested in this work using brute force attack, the number of possible attempts to break the secret message:

$$\text{Number of possible attempts} = 2^k * 2^{128}$$

(Where k equal to 128,192 or 256). This is because the number of possible attempts to break the ciphertext using original AES with key size K is equal to 2^K, with the use of 128-bit IV, which changes the encrypted text each time the attacker must guess the 128-bit IV in addition of the key.

In case the attack is the frequency of letters analysis, the use of the IV helps to obtain different ciphertext for the same plaintext encrypted with the same key, that if the repetition of a certain character or even a word and was encrypted with the same key will get different ciphertext every time. Table (1) shows the use of the normal AES and the adjusted AES-GCM where the IV is increased by one and then rotated shifted to left is increase the percentage of change in the ciphertext. Where increase the IV by one in addition to the shifting in order not to be periodic no repeat itself after 128 shift.

Table 1. AES-GCM with IV.

Plaintext	**Key_{128}**	**IV**	**Ciphertext**
"aaaaaaaaaaaaaaaa"	"aaaaaaaaaaaaaaaa"	Non	"9b9b9b9b9b9b9b9b9b9b9b9b9b9b9b9 be8e8e8e8e8e8e8e8e8e8e8e8e8e8e8"
"aaaaaaaaaaaaaaaa"	"aaaaaaaaaaaaaaaa"	"1111111111111111"	"8c8c8c8c8c8c8c8c8c8c8c8c8c8c8c7 67676767676767676767676767676676"
"aaaaaaaaaaaaaaaa"	"aaaaaaaaaaaaaaaa"	(IV+1)<<1: "2222222222222224"	"d3c22d7cf54a2fc55a14e336e03758ae2 c06088e41e6e3787a19b00ccabf0515"

Practically, to make sure that the proposed modification on AES-GCM which aims to increase the randomization rate in the resulting ciphertext achieved the target properly, the test functions of NIST randomness was applied. The test was performed using the following parameters as an inputs for encryption by using AES algorithm, where the key was "computer college", IV was "1111111111111111" and the text "ahmad salim badr" was repeated more than once in plaintext, where the plaintext size was 2560 bit in binary. The goal was to increase randomization when a particular plaintext was encrypted with the same key more than once even in the case of a little randomization in the used IV.

Table (2) shows the results of the test of three cases of encryption process by using AES. The first column in the table includes the test functions adopted, the second column represents results of using the traditional AES encryption without a IV, the third column shows the results of the randomization when using the normal AES-GCM which depends on the addition by one on the IV in each new encryption process while the last column displays the random ratios in the proposed method in this work, which uses the shift process plus the IV increase by one each time. As is evident when observing random probability values in the testing table as well as their own cases, the proposed method indicated in the last column achieves a more random probability than the other two methods. Increasing the randomization in encrypted text makes it harder for the attacker to parse the ciphertext and fetch the original text across it.

Table 2. NIST Randomness Test for AES-GCM Ciphertext.

Test Function	**AES**		**AES-GCM (IV+1)**		**AES-GCM (R_Shift(IV+1)**	
	pValue	State	pValue	State	pValue	State
BLOCK FREQUENCY	0.000000	FAILURE	0.220444	SUCCESS	0.968555	SUCCESS
FFT	0.207026	SUCCESS	0.207026	SUCCESS	0.207026	SUCCESS
FREQUENCY	0.000000	FAILURE	0.139333	SUCCESS	0.277333	SUCCESS
LEMPEL-ZIV COMPRESSION	1.000000	SUCCESS	1.000000	SUCCESS	1.000000	SUCCESS
LONGEST RUNS OF ONES	1.000000	SUCCESS	1.000000	SUCCESS	1.000000	SUCCESS
NONPERIODIC TEMPLATES	0.999998	SUCCESS	0.999998	SUCCESS	0.999998	SUCCESS
OVERLAPPING TEMPLATE OF ALL ONES	1.000000	SUCCESS	1.000000	SUCCESS	1.000000	SUCCESS
RANK	0.000000	SUCCESS	0.000000	SUCCESS	0.000000	SUCCESS
RUNS	0.000000	FAILURE	0.087222	SUCCESS	0.968444	SUCCESS
Serial	1.000000	SUCCESS	1.000000	SUCCESS	1.000000	SUCCESS

Encryption algorithms are usually designed to find the greatest effect of the key and the plaintext on the ciphertext (diffusion and confusion), and thus get a more and more randomness in the ciphertext to make it effective against the attacks aimed to cryptanalysis and obtained the original text. The table below shows the effective of the proposed change on the AES-GCM algorithm, where the randomization ratio was measured by using NIST randomness functions. The column PValue represents the randomness ratio for each function, and the optimal randomness ratio is one. The ratios of the three main functions (block frequency, frequency and runs) in the table shows a clear increase in the randomization rate in column of the proposed adjustment on AES-GCM compared with the traditional AES or AES-GCM.

Figure 4 shows the clear increase in the randomization ratio of the three main test functions of the NIST randomness test functions. The figure illustrates that even if a little randomized plaintext and a little randomness in key are used in encryption process, the modified algorithm gives a large randomness proportion in the ciphertext, which makes the cryptanalysis process very difficult.

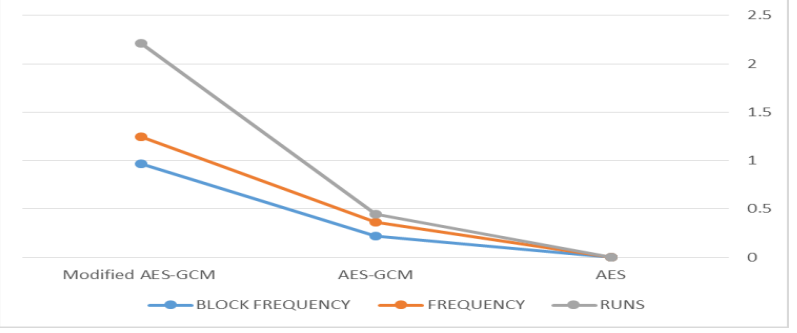

Fig.4. The Proposed Modification effect on Ciphertext Randomness.

7. Conclusion

The AES-GCM algorithm is an encryption and authentication algorithm, which used today in many applications and systems to help keep confidential information safe. In this paper, the AES-GCM encryption algorithm was modified by rotated shift the IV after added one instead of add one only in traditional algorithm. The modification increase the randomness ratio in the ciphertext, thus make modified algorithm more difficult to break the encrypted text by analysis it in order to obtain the original text.

Acknowledgment

Thanks presented for College of Computer Sciences & Information Technology, University of Anbar, by aiding to bring out the research. Also, special thanks to Assist Prof Dr. Salim Bader for assistance this research.

References

[1] W. Stallings and M. P. Tahiliani, Cryptography and network security: principles and practice, vol. 7. Pearson London, 2017.
[2] Kawle, Pravin, et al. "Modified Advanced Encryption Standard." International Journal of Soft Computing and Engineering (IJSCE) 4 (2014).
[3] Mohan, H. S., and A. Raji Reddy. "Revised AES and Its Modes of Operation." International Journal of Information Technology 5.1 (2012): 31-36.
[4] Wu, Yue, Joseph P. Noonan, and Sos Agaian. "NPCR and UACI randomness tests for image encryption." Cyber journals: multidisciplinary journals in science and technology, Journal of Selected Areas in Telecommunications (JSAT) 1.2 (2011): 31-38.
[5] Koradia, V. C. "Modification in Advanced Encryption Standard." Journal Of Information, Knowledge And Research In Computer Engineering 2.02 (2012).
[6] Vaidehi, M., and B. Justus Rabi. "Enhanced MixColumn Design for AES Encryption." Indian Journal of Science and Technology 8.35 (2015).
[7] Hashem, Soukaena H. "A Proposed Modification on RC4 Algorithm by Increasing its Randomness." Al-Rafidain University College for Sciences 39 (2017): 349-372.
[8] Ali, Ammar H., and Ali M. Sagheer. "Design of an Android Application for Secure Chatting." International Journal of Computer Network and Information Security 9.2 (2017): 29.
[9] W. Stallings and M. P. Tahiliani, Cryptography and network security: principles and practice, vol. 6. Pearson London, 2014.
[10] J. Holden, The Mathematics of Secrets: Cryptography from Caesar Ciphers to Digital Encryption. Princeton University Press, 2017.
[11] J. Katz and Y. Lindell, Introduction to modern cryptography. CRC press, 2014.
[12] F. K. Gürkaynak, “GALS system design: side channel attack secure cryptographic accelerators,” ETH Zurich, 2006.
[13] D. McGrew and J. Viega, “The Galois/counter mode of operation (GCM),” Submission to NIST Modes of Operation Process, vol. 20, 2004.
[14] K. Jankowski and P. Laurent, “Packed AES-GCM algorithm suitable for AES/PCLMULQDQ instructions,” IEEE transactions on computers, vol. 60, no. 1, pp. 135–138, 2011.
[15] B. Buhrow, K. Fritz, B. Gilbert, and E. Daniel, “A highly parallel AES-GCM core for authenticated encryption of 400 Gb/s network protocols,” in ReConFigurable Computing and FPGAs (ReConFig), 2015 International Conference on, 2015, pp. 1–7.

Authors' Profiles

Ahmad S. Bader has received his B.Sc. in Computer Science (2011) from the University of Anbar, Iraq. He is a master student (2016, till now) in the Computer Science Department, College of Computer Sciences and Information Technology at University of Anbar. He is interested in the following fields: Information Security, Biometrics, Network Security, Image Processing and Coding Systems.

Ali M. Sagheer is a Professor in Al-Qalam University College. He received his B.Sc. in Information System (2001), M.Sc. in Data Security (2004), and his Ph.D. in Computer Science (2007) from the University of Technology, Baghdad, Iraq. He is interested in the following fields; Cryptology, Information Security, Number Theory, Multimedia Compression, Image Processing, Coding Systems, and Artificial Intelligence. He has published many papers in different scientific journals.

How to cite this paper: Ahmad S. Bader, Ali Makki Sagheer,"Modification on AES-GCM to Increment Ciphertext Randomness", International Journal of Mathematical Sciences and Computing(IJMSC), Vol.4, No.4, pp.34-41, 2018.DOI: 10.5815/ijmsc.2018.04.03