Engineering and Technology Journal

Journal homepage: engtechjournal.org

A Modification on Key Stream Generator for RC4 Algorithm

Suhad M. Kareem[a*], **Abdul Monem S. Rahma**[b]

[a] Collage of Computer Science and Information Technology, University of Basrah, Basrah, Iraq.
Suhad_althaher@yahoo.com

[b] Department of Computer Sciences, University of Technology, Baghdad, Iraq.
110003@uotechnology.edu.iq

*Corresponding author.

Submitted: 27/06/2019 Accepted: 18/09/2019 Published: 25/0٧/2020

KEYWORDS

Information Security, Encryption, stream cipher, RC4, random generator.

ABSTRACT

Rivest Cipher 4 (RC4) is one of most common stream cipher, but it experience many problem, such as, there is little combination between the plaintext and cipher-text. For this reason RC4 Cipher is vulnerable to a number of attacks. Thereupon, this paper proposes a new modification of the RC4 to strong it. This achieved by modifying key-stream generator based on linear equation with four state tables for generating random numbers. Key control is used for selecting one state table to apply # operation, then performing forward and backward effects to generate the key-stream that used in encryption and decryption. Being evaluated, the results obtained from the statistical probabilities prove that our proposed algorithm is more complex than the standard algorithm by using different key lengths. Also, our proposed RC4 pass the randomness in most metrics in NIST.

How to cite this article: S. M. Kareem and A. S. Rahma, “A modification on key stream generator for RC4 algorithm,” Engineering and Technology Journal, Vol. 38, Part B, No. 0٢, pp. 54-60, 2020.
DOI: https://doi.org/10.30684/etj.v38i2B.404

1. Introduction

In a globalized world characterized by the rapid growth of electronic communication via insecure networks and an ever-growing number of attacks on computers and users, information security-related measures, such as integrity, confidentiality and authentication have been offered. As illustration encryption algorithms may be viewed as one of the most robust tools which provides secure environment to protect the sensitive data. The main purpose of encryption is to prevent the “un-authorized users to retrieve the original text [1, 2].

There are two types of encryption algorithm: Symmetric-key and Asymmetric-key encryption. Symmetric algorithms use the same key to encrypt and decrypt message and there are basically classified into two types: Block Ciphers and Stream Ciphers. When encrypting the entire messages as block at the same time, this is called block ciphers such as DES, AES, and Blowfish...etc. [3].

In stream cipher, the term of key-stream is defined as a pseudorandom sequence of bits to encrypt with sequence bit of plaintext by using XOR operation. The cipher text produced from the" stream cipher comprises sequence of bits with a length equal to that of the plaintext. However, there may be a number of various stream ciphers but the most widespread one is RC4 algorithm [4, 5] which has been proposed in this work.

Most modern cryptographic algorithms depend on functions with two states (0, 1) for encryption and decryption. RC4 is the most popular stream which depends on the classical operation (XOR) with two states: simply (0, 1) which has several weak points, such as being simple where it can be deciphered easily by attackers. In addition, there exists a problematic overlap among the public known outputs of the internal state. These weaknesses have been exploited by attackers to discover the key and retrieve the data [6, 7]. Consequently researchers have attempted to replace the two states with four ones (0, 1, 2 & 3) as shown in Fig.1 in the following sections for increasing key space [8]. This research proposes a modification on RC4 based on linear random generator using four states in order to increase the security level.

This paper is organized as follows. Section 2reviews the relevant literature about the modification of RC4. Section3 and Section 4 are introduces an overview about the work and the proposed RC4 respectively. Section 5 evaluates the complexity of the proposed algorithm. Finally, section5 presents our conclusions.

2. Related Work

Attacks on encryption “algorithms” could be considered as the main challenges, In consequence, several scholars have produced a plethora of papers in order to increase the level of algorithms security. Thus, this section offers the most related studies in the field of RC4 modification.

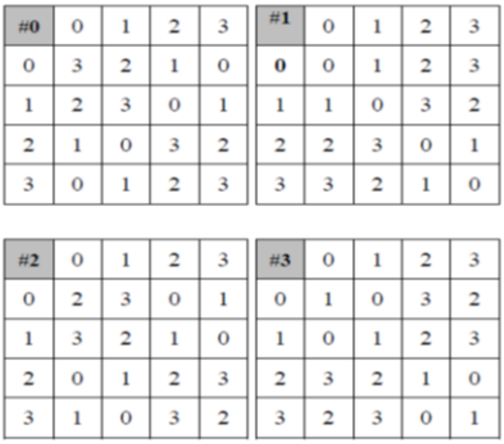
In 20٠٩ [8], the research has presented the work by combining the curve security methods with quantum cryptography concepts to increase the security and key space in order to make encryption operation more secure and robustness. In this work, the proposed modification focuses on the use of four different states (0,1,2,3) instead of (0,1) to make variations in the polarized angles which have been used in quantum description which have been encoded in these four tables, in addition to the output description which have used polarized states angles according to the tables. Then manipulation ciphers convert plaintext into cipher text by changing the actual state pattern of each character by using a logical operator (#). The (#) has four truth tables that shown in Figure 1.

In 2012[9] the authors have proposed anew stream cipher called a Pardeep cipher (PC-RC4) as extension to RC4 algorithm. The PC-RC4 increase the randomness in both KSA and PRGA to make it stronger, but it also increases the time taken for algorithm execution.

In 2015[10], the authors have suggested a new improvement on the RC4 based on the irreducible polynomial by using dual key: the first one is used for encryption, while the second key is generated randomly to determine the Polynomial used in the encryption and decryption operations. The mathematical operations multiplication in this work based on the mathematical theory of Galois field GF (28).

In 2016[11], the researchers have proposed RC4+S Algorithm as extension of standard RC4. RC4+S algorithm also raises the randomness in both KSA and PRGA stages by using double permutation on the state matrix in these stages. (Proposed PRGA+S) tends to overcome the whole overlap problem between the key stream outputs and state-matrix(S) of PRGA.

In 2013[12] Hammood et al. have proposed an RRC4 random initial state algorithm. In this algorithm, a new enhancement and improved randomness of RC4 are introduced compared with the traditional RC4. However, speed was not addressed.



#0	0	1	2	3
0	3	2	1	0
1	2	3	0	1
2	1	0	3	2
3	0	1	2	3

#1	0	1	2	3
0	0	1	2	3
1	1	0	3	2
2	2	3	0	1
3	3	2	1	0

#2	0	1	2	3
0	2	3	0	1
1	3	2	1	0
2	0	1	2	3
3	1	0	3	2

#3	0	1	2	3
0	1	0	3	2
1	0	1	2	3
2	3	2	1	0
3	2	3	0	1

Figure 1: The truth tables for # operation

3. An Overview of RC4 Algorithm

RC4 can be viewed as one of the most widespread symmetric encryption stream in the field of cryptography. It is also known as ARC4 or ARCFOUR. The algorithm was designed by Ron Rivest of RSA security in 1987 and then was anonymously released in 1994 on mail productions. The algorithm has many applications and has been used in many common protocols such as Secure Sockets Layer (SSL)/TLS and WEP (Wireless Equivalent Privacy) to provide confidentially, encrypt files products via e-mail to provide security. This algorithm uses a variable key size stream from 1 to 256 bytes which are completely independent of plaintext [13, 14].

The principle work of RC4 algorithm has two stages: key generation stage and encryption stage. Both stages must be performed for every new key. The first step in RC4 and the most difficult one is key generation. To generate stream of key used in encryption and decryption many steps are to be performed which can be summarized as follows: firstly, initialize the two state variables (S1 with value from 0 to 255 and S2 with number of repeated chosen key), then perform permutation on S1. In key generation two state variables as S1 (initial with a numbers from 0 to 255) and S2 (fill with chosen key) are used. The second step is conducted by performing many operations such as (swapping, modulo and other formulas) on the S1 and S2. After generating stream bit of key, the encryption process is carried out exored bit with bit of plaintext to produce the cipher text and the cipher-text is exored with key-stream to retrieving the plaintext in decryption. The overall work of RC4 is explained in the Pseudo-code [5, 2].

4. Proposed RC4

As noted earlier, RC4 cipher has many weaknesses. This makes it vulnerable to several attacks by hackers. Since RC4 consists of two stages: KSA and PRGA, therefore, the add level of security to this stage would make the algorithm stronger against the hackers. This paper proposes a new modification on RC4 algorithm for both stages which is proposed to increase their randomness in order to enhance its security.

In KSA stage, our modification is proposed by making the key generator work based on linear equation with particular prime number. Instead of filling the state variable (S1) with values from 1 to 256, it will be filled with values calculated from the following equation:

$$Y_{n+1} = (a * Y_n + b\ )\quad mod\ p \qquad (1)$$

Where, “a” is multiplier; “b” is increment; “Y_n ” is the start value; and “p” is the large prime number as modulus. In this work, the add operation in this equation is replaced with # operation by selecting randomly one of the tables from the 4 tables shown in the Fig.1. Moreover, another key is called control key which generates randomly such as (103210…) saves it in the array, and in each step we use (2 bit) from the control key have been used to determine one of the tables among these four tables. The # operation works as an intersection between the row and the column in the selected table. After generating each number, it will convert it to binary and select only 8 bits as a value to be saved in the state variable (S1). The purpose behind using the generator and then the selective process is to increase the randomness of the KSA stage.

In the case of PRGA stage, our proposal attempts to increase the randomness by performing permutation on S1 as a forward effect between the elements of S1 to produce the key stream (X_i), then to use the result of the key stream as a feedback affect the elements of the S1. This leads to break the correlation between the values of the S١ as shown in Figure 2.

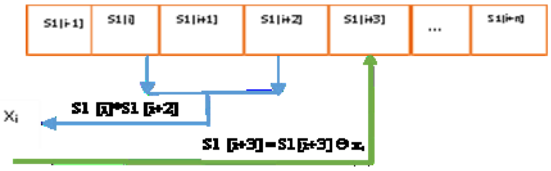


Figure 2: Application of the process of forward and backward in the ith of PRGA stage

In this algorithm, the encryption and the decryption have been taken place in the same manner as that of the well-known RC4 algorithm as xored between the bits of key stream and plaintext in encryption and between the key stream and cipher text in the decryption process. The overall work of our proposed RC4 algorithm is shown in the following algorithim1:

Algorithim1: Pseudo code for modified RC4

```
Input: Plaintext, key and control key
Output: Cipher text
Begin
  Step 1: Initialize S1 and S2
    Step1.1 select a,b and y as prime numbers
            consist of five digits.
    For i from 0 to 255
      Step1.1.a: use (2-bit) from control key to
                 Determine one table among these
                 four tables.
      Step1.1.b: generate number $y_i$ by using
                 The equation (1), replace operation
                 (⊕) with operation (#).
      Step1.1.c: convert $y_i$ into binary form.
      Step1.1.d: select only 8 bits from the
                 output of (step1.1c) and save the
                 result in variable (X).
      Step1.1e: S1 [i] = X.
    End for
    Step1.2 initialize S2
      For i from 0 to 255
        S2 [i] = Key[i mod keylength] …
                 Repeated key.
      End for
  Step 2: Perform permutation on S-box1
    Initialize j = 0
    For i from 0 to 255
      j = (j + S1 [i] + S2 [i]) mod 256
      Swap values of (S1[i] and S1[j])
    End for
  Step 3: generate the key stream (KS)
  While (true)
    i=1
    Apply forward effect:
    $x_i$ = S1 [i]*S1 [i+2]
    Apply backward effect:
 S1 [i+3] = S1[i+3]⊕$x_i$
  KS = S1[$x_i$]
```

```
    i=i+1
  End while
  Step 4: encryption and decryption
    Step 4.1: encryption: Cipher-text =
          plaintext ⊕KS
    Step 4.2: decryption: plain-text =
          Ciphertext ⊕KS
End begin
```

5. Evaluation

Cryptography is the art of the domain which is used for secure information. This process is practiced by using encryption algorithms. One important type of encryption algorithm is the symmetric key which uses the same key (K) between two parties for encryption and decryption the message (M). Therefore, the attacker tries a number of probabilities to retrieve the plain-text (P) from cipher-text(C). Where the K, P and C are represented “as a sequence of bits”. The attacker cannot discover the C from P without the knowing K.

One of the most common type of attacks against symmetric key system is the brute-force attack. In which the attacker estimates which key has been used. The strength of encryption is attached to the difficulty of discovering the key, which in its turn depends on both the cipher used and the length of the key [15]. There is a number of methods for evaluating encryption algorithm, however, just two types of evaluation have been introduced in this paper as shown later in this section:

I. Complexity computing based-evaluation

Our modified RC4 compared with original RC4 which is related to different key sizes in order to compute the complexity of the proposed algorithm against the attackers. This complexity is computed based on the calculating the probability of plaintext is multiply by probability of key.

The number of possibilities of the key which has the length of 256 bits to decrypt well-known RC4 algorithm is: 28 possibilities. The following example illustrates the number of possibilities of keys which the attacker needs to decrypt the modified algorithm with the key which has length of 256 bits: to select one table from the four tables in # operation is 22 , to find the probability of the 8 bits which has been selected from the generated value is: (23)8. Consequently, the number of possibilities of the key to decrypt the modified RC4 algorithm is 28 ×22 × (23)8.

The below Table1 and Figure 3 show the statistical computation to represent the complexity of the modified RC4 which has more complex security than that of the original RC4.

Table 1: Comparison the complexity with different key size

Key length /bits	Standard RC4	Modified RC4
64	2^6	$2^6 \times 2^2 \times (2^3)$
256	2^8	$2^8 \times 2^2 \times (2^3)^8$
1028	2^{10}	$2^{10} \times 2^2 \times (2^3)^8$

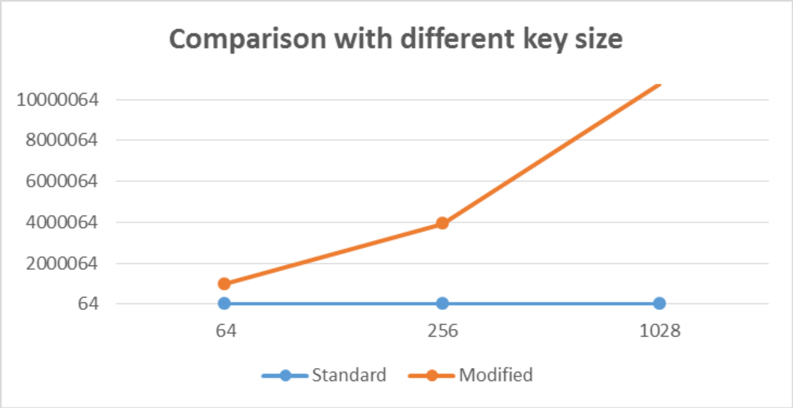


Figure 3: Comparison with different key size

II. Randomness computing based-evaluation

Encryption algorithm should produce the cipher text with more random and text unpredictable. There are several methods for computing the randomness such as NIST, Diehard tests, TestU01. For example, NIST statistical test consist of a number of measures for testing the randomness in binary sequences. In this paper, frequency test and run test are used for the evaluation shown in table (2).
This test is calculated over five random cipher-text which are produced from the well-known RC4 and the modified ones. The probability value (p-value) is set to 0.01 to decide whether the cipher-text is random or not. The results in Table 2 and Figure 4 show that computed average tests.

Table 2: Results of NIST on the outputs by the standard RC4 and the modified of RC4

	Frequency test		Run test	
Key length / bits	p-value for Standard RC4	p-value for Modified RC4	p-value for Standard RC4	p-value for Modified RC4
64	0.175	0.193	0.412	0.305
256	0.530	0.204	0.686	0.792
1028	0.566	0.793	0.741	0.901

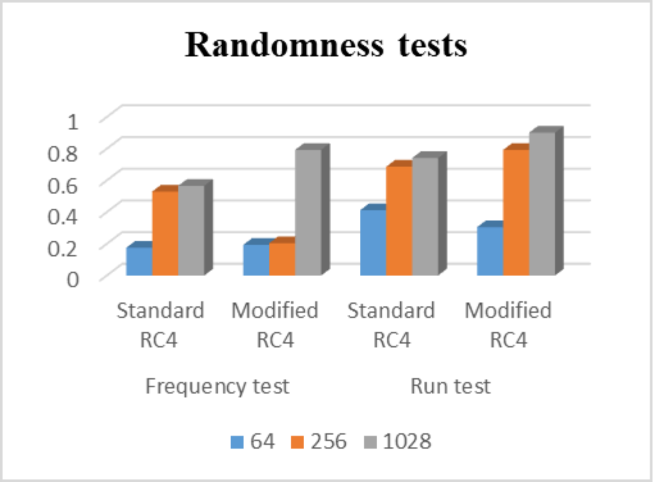


Figure 4: Results of randomness tests with different key size

6. Conclusion

In this paper, a modified version of RC4 is proposed to increase the security of the well-known RC4 by increasing the randomness in both KSA and PRGA stages. This is realized by filling the state variables (S1 and S2) with values depend on a strong random generator starting with a prime number which consists of 5 digits. Then is to select only 8 bits. Similarly, it can be realized by utilizing the key control to identify which table used to apply the # operation. This lead to increase the randomness and add complexity level into algorithm. In addition, the modification on RC4 algorithm adds more complexity in computing the key and it decreases the probabilities of break against differential analysis from the brute force attacks. Conversely it saves time taken during the mathematical computation. The results from the NIST analyses, the effect of the apply # operation in the encryption and decryption provides good randomness.

References

[1] S. V. Swathi, P. M. Lahari and B. A .Thomas, “Encryption algorithms: a survey,” *International Journal of Advanced Research in Computer Science & Technology,* Vol. 4, No.2, pp. 81-88, 2016.

[2] A. Mousa and A. Hamad, “Evaluation of the RC4 algorithm for data encryption,” *international journal of computer science and application*, Vol. 3, No.2, pp.44-56, June 2006.

[3] William Stallings, “Cryptography and network security: Principles and practice,” Pearson Education/Prentice Hall, 5th Edition, 2011.

[4] S. S. Gupta, "Analysis and implementation of RC4 stream cipher," Ph.D. thesis presented to Indian Statistical Institute, Kolkata, West Bengal, India, 2013.

[5] P. Jindal and B. Singh, "RC4 encryption-a literature survey," *International Conference on Information and Communication Technologies (ICICT 2014)*, 2015.

[6] I. Sumartono, A. P. U. Siahaan and N. Mayasari, "An overview of the RC4 algorithm," *IOSR Journal of Computer Engineering (IOSR-JCE)*, Vol. 18, No. 6, pp.67-73, 2016.

[7] I. Mantin and A. Shamir, "A practical attack on broadcast RC4," In: 8th international workshop, FSE, pp. 152–164, 2002.

[8] H. Bahjat, A. S. Rahma, "Proposed new quantum cryptography system using quantum description techniques for generated curves," The 2009 international conference on security and management, SAM2009, 13-16 2009, LasVegas, USA, SAM 2009.

[9] P. Pardeep and P.K. Pateriya "PC-RC4 algorithm: an enhancement over standard RC4 algorithm," *International Journal of Computer Science and Network (IJCSN)*, Vo. 1, No. 3, pp.1-6, 2012

[10] A. S. Rahma and Z. .M.Hussein**,** "Modified RC4 dual key algorithm based on irreducible polynomial," *International Journal of Emerging Trends & Technology in Computer Science (IJETTCS),* Vol. 4, No. 2, 2015.

[11] S. H. Hashem and A. B. Jasim, **"**A proposed modification on RC4 algorithm by increasing its randomness," *Journal of Al Rafidain University College*, Vol. 1, No.39, 2016.

[12] M. M. Hammood, K. Yoshigoe, and A.M. Sagheer, **"**RC4 stream cipher with a random initial state," In: Proceedings of 10th FTRA international conference on secure and trust computing, data management, and applications (STA'13). Lecture notes in electrical engineering, Springer, Heidelberg, 2013.

[13] M. K. Pehlivanoğlu and N. Duru, "Email encryption using RC4 algorithm," *International Journal of Computer Applications (0975 – 8887)*, Vol.130, No.14, 2015.

[14] S. Kadry and M. Smaili, "An improvement of RC4 cipher using vigenère cipher," *International Journal of Computational Intelligence and Information Security*, Vo. 1, No. 3, 2010.

[15] J. Talbot and D. Welsh*,* "*Complexity and cryptography an introduction,*" Cambridge University Press, 2006.