Available online at www.qu.edu.iq/journalcm
JOURNAL OF AL-QADISIYAH FOR COMPUTER SCIENCE AND MATHEMATICS
ISSN:2521-3504(online) ISSN:2074-0204(print)

Enhancement RC4 Algorithm Based on Logistic Maps with Multi-Parameters

Noora Shihab Ahmed [a] , ***Salam Hussein Ahmed*** [b]

[a] *Computer Science Department, University of Halabja, Kurdistan,Iraq.Email: noora.ahmad@uoh.edu.iq*

[b] *Computer Science Department, University of Sulaimani, Kurdistan, Iraq.Email: salam.ahmed@univsul.edu.iq*

ARTICLE INFO

Article history:
Received: 01 \09 \ 2019
Rrevised form: 10\11\2019
Accepted : 17/11/2019
Available online: 12\01\2020

Keywords:

Average Secrecy test, Key Scheduling Algorithm, RC4, Logistic Maps.

ABSTRACT

This paper aims at overcoming the shortcomings of RC4 (Rivest Cipher 4) algorithm that mainly resides in the "key scheduling algorithm" (KSA) of RC4. This paper is a trial to enhance key generation of RC4 on the basis of logistic maps with multi-parameters named (EKSA), the permutation of array of S improved to base of the generator for the random numbers that depend on three logistic maps with two parameters, three parameters and four parameters, the suggested algorithm result the follow : outcome = T ⊕ generated key ⊕ value that is random from EKSA (Lm4p (w)) The secrecy is tested for the enhancement RC4 with EKSA, in addition to the arbitrariness and variable key size effectiveness and different size of the plaintext regarding to those of the original RC4.The outputs display that original RC4 with KSA is less powerful than the the enhancement RC 4 with EKSA.

DOI : 10.29304/jqcm.2019.11.4.650

1 . Introduction

The most commonly stream cipher is RC4, and worked in many internet protocols for example wired equivalent privacy (WEP), Skype, Wireless protected access (WPA) and secure socket layer, Transport layer security (SSL/TLS) [Craincu, B.2015]. The critical components in RC4 algorithm over such a broad space of applications have been its speed and clarity; efficient implementation in both s/w and h/w were exceptionally simple to create. RC4 is simple and fast compared to other encryption methods.

Fluhrerand and others . in [Fluhrer, S. etal 2001] they analyzed the KSA which derives the starting state from a variable measurementfs key and explain two noteworthy shortcomings of this process. The shortcoming is within the presence of a huge number of bits of the initial permutation (KSA output). The 2nd shortcoming is related to key powerlessness, which applies when a portion of the key displayed to the KSA in uncovered to the attacker.

T.D.B Weerasinghr in [T.D.B Weerasinghe, 2012] displayed the analysis of an essentially adjusted RC4 algorithm, and attempted out a basic alteration of RC4 PRGA, where we can mention it like this: Out Put=M XOR Generated key XOR j.

S. M. Hameed and I. N. Mahmood in [Sarab M. Hameed etal 2016] display a unused form of KSA recommended in an endeavor to extend the security of RC4 and get freed of the shortcoming related to the elementary permutation of the S array and the permutation process of the S array.

Naji, Ali and Noora (2018), Present improved RC4 key generation using Multi-Chaotic Maps (IKSA), the results of improved RC4 with IKSA is better than RC4 with KSA.

Corresponding author: Noora Shihab Ahmed
Email addresses: noora.ahmad@uoh.edu.iq
Communicated by Qusuay Hatim Egaar

This paper show a new enhancement of the KSA depend on the randomness of the three logistic maps (With (two parameters, three parameters and four parameters)). The logistic maps have many good features such as allergy on primary condition and system parameter, periodicity and mixing properties. In this paper, we invest these interesting properties of logistic maps to generation random number. The S array permutation is proposed to depend on the generated random key.

2. Overview

2.1 RC4 Algorithm

In 1987 , Ron Rivest [Stallings W., 2011], that consider to be member that create RSA put the RC4. RC4 is a shortened form for" Rivest Cipher 4", it is similarly recognized as "Ron's Code 4". The algorithm is depending on theutilize of a random variation. The RC4 algorithm is straightforward and moderately simple to clarify. [Mao W., 2003][Abdul M.S. Rahma etal, 2015]. Figure 1 shows the "Pseudo random number generation algorithm" (PRGA) and "key scheduling algorithm" (KSA)

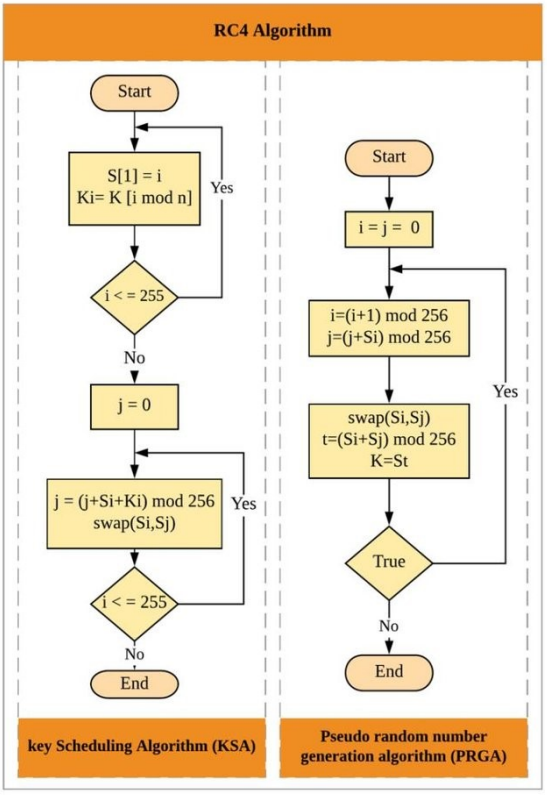


Figure 1: RC4 Flowchart

2.2 Logistic maps with One and Multi-Parameters

In different methods for the encryption of the information , The one-dimensional chaotic maps have been massively used as it recognize for its high-level simplicity and skill. In spite of its features it possess some defects, for instance small key space and security deficiency. Consequently, there is deficiencies in the use of logistic maps with one and multi-parameters.

A. Logistic chaotic Map with one parameter

Logistic chaotic map is the simplest nonlinear model of the chaotic map occurs in real systems. The logistic map chaotic scheme is signified as in the following:

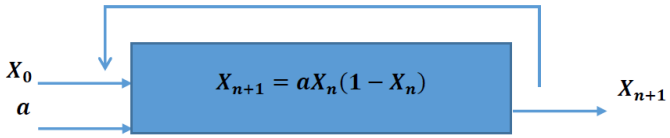


Figure 2: The logistic map with one parameter

Where $\alpha = 3.975$, and $0 < X_0 < 1$

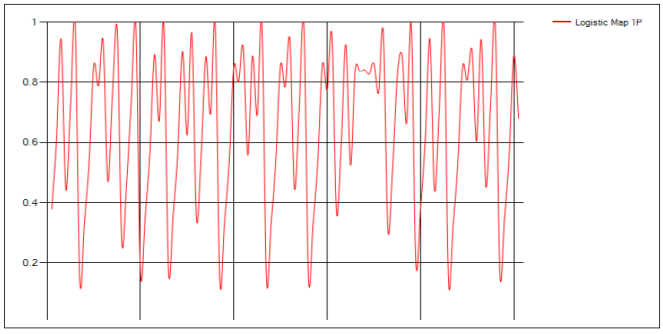


Figure (3) The logistic map with one parameter

B. Logistic Map with Two Parameters

The two parameters logistic map with is signified as:

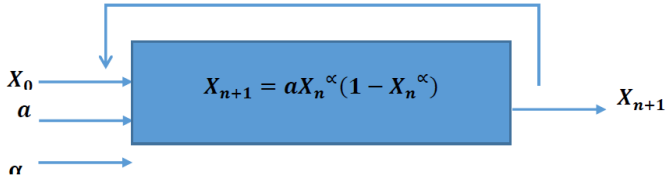


Figure (4) The two parameters logistic map.

Where $a = 2, \alpha = 0.5$, and $0 < X_0 < (0.5)^2$.

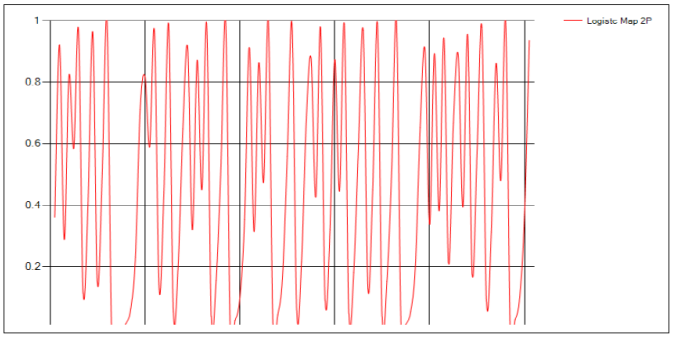


Figure (5) The two parameters logistic map with.

C. Logistic Map with Three Parameters

The three parameters logistic map is signified as:

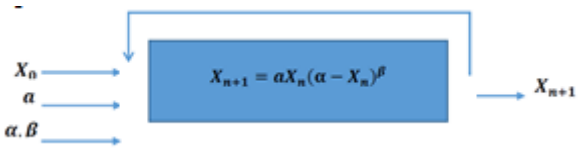


Figure (6) The three parameters logistic map.

Where a = 1.5, α = 3, β = 0.5 and $0 < X_0 < 2$.

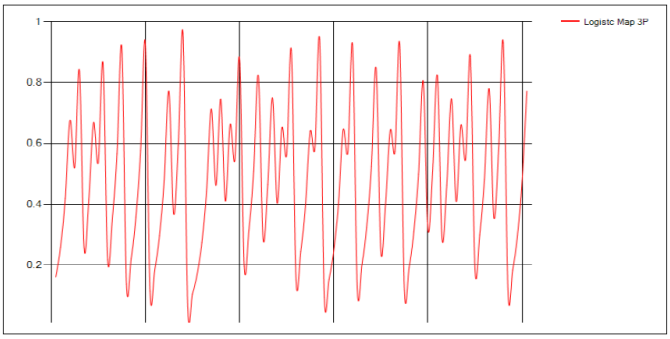


Figure (7) The logistic map with three parameters.

The four parameters logistic map with is signified as:

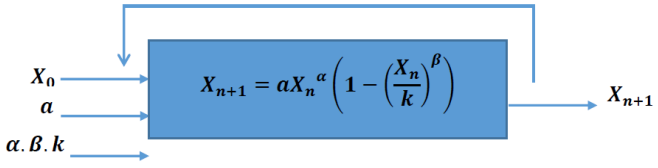


Figure (8) The logistic map with four parameters.

Where a = 0.5, α = 2, β = 3, k = 2 and $0 < X_0 < \left(\frac{1.6}{3.5}\right)^{\frac{1}{3}}$.

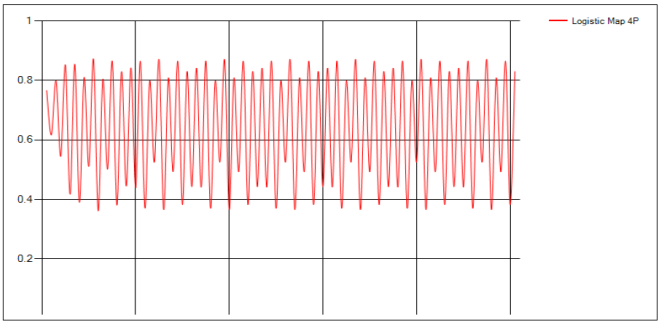


Figure (9) The logistic map with four parameters.

3 Proposed Algorithm

The aim of this part is to generate the enhancement RC4 algorithm (ERC4) fundamentally through two stages.
A) enhancement key scheduling algorithm , a new version of KSA called EKSA is proposed. In this proposal as shown in Fig 10, we choose three Logistic maps (Logistic Map with Two , three and four Parameters) and their figures are (4), (6), and (8), correspondingly. The key which is secret is SEED, is the initial condition of every map. Every

iteration produce algorithm by (w=0 to 255: the iterations number) arrangements of 24 bits (8-bit blocks for every chaotic maps). Lm2p (w), Lm3p (w) and Lm4p (w) are taken from chaotic maps as follows:

Logistic Map with Two Parameters generate Lm2p (w)
Logistic Map with Three Parameters generate Lm3p (w)
Logistic Map with four Parameters produce Lm4p (w)
In the following way:

$$F_n(t_{m+1}) = \begin{cases} 0 & if \quad 0 < t_{m+1} \leq 0.5 \\ 1 & if \quad 0.5 < t_{m+1} < 1 \end{cases}, n = 1,2,3$$

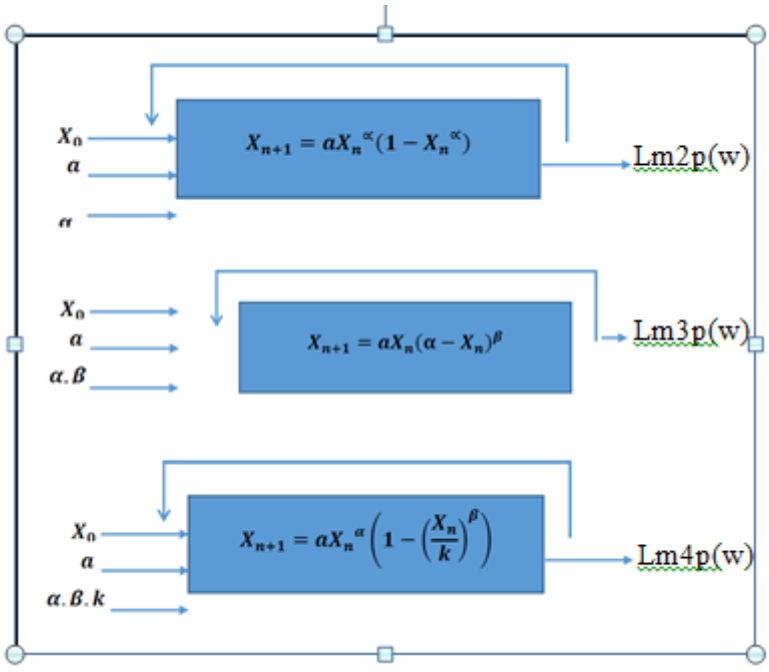


Figure (10) The proposed algorithm EKSA

B) Encryption / Decryption process
Encryption: C=(M⊕Generated key⊕ Lm4p (w))mod_{256}
Decryption: M=(C⊕ Generated key ⊕Lm4p (w))mod_{256}

Algorithm ERC4

Input [plaintext] and [key]
Output [cipher text]
Step 1: /Initialize /
for i = 0 to 255
S[i] = i;
T[i] = K[i mod key];
Next i;
Step 2: / Perform IP of S /
for w=0 to 255
Lm2p (w) = Location: generate from the Logistic map with two parameters
Lm3p (w) = Location: generate from the Logistic map with three parameters
j = (Lm3p (w0 + S [Lm2p (w)] + T [Lm2p (w)]) mod256
Swap (j, S[Lm2p (w)])
Next w;
Step 3: /Stream Generation/

Set [i, j] = 0;
while (true)
i = (i + 1) mod 256;
j = (j + S[i]) mod 256;
Swap (S[i], S[j]);
t = (S[i] + S[j]) mod 256;
k = S[t];
Lm4p (w): generate from the Logistic map with four parameters
Step 4:/The process/
Encryption C = (M ⊕ K ⊕ Lm4p (w)) mod256
Decryption M = (C ⊕ Generation key ⊕ Lm4p(w))
Step 5:/End/

4. RESULTS AND DISCUSSION

4.1 Secrecy of ciphers

Secrecy of ciphers is calculated in terms of the key prevarication (conditional entropy of key given cipher)

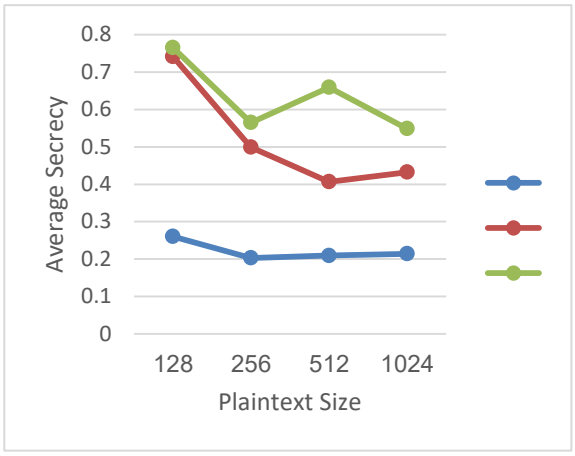
$$H(k/c) = \sum_{j=1}^{L} \sum_{i=1}^{n} q_i P_{ij} \log P_{ij} \quad (4)$$

Where
qi = Pr (C = ci)
Pij = Pr (K=ki / C = ci)

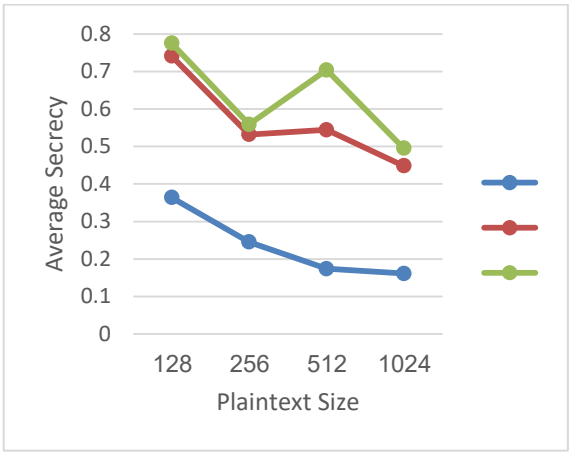
L is the key length
n is the cipher text length
1-**Average secrecy test: A variable plaintext size, Fixed key length.**

Table 1: Average Secrecy Value vs. Plaintext size

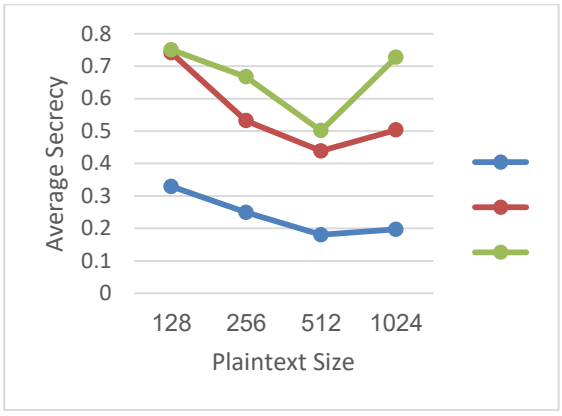
Keys Length\Bits	Plaintext Size\Bits	Algorithms		
		Original RC4 With KSA	Improvement RC4 With IKSA [Naji, M. et al. 2018]	Enhancement RC4 With EKSA The proposed algorithm
32	128	0.260459373	0.740856729	0.764854458
	256	0.203040633	0.498915456	0.564987012
	512	0.20944977	0.406738053	0.658792222
	1024	0.214365643	0.43235869	0.548723198
64	128	0.363815483	0.740856729	0.775201986
	256	0.245275562	0.531832803	0.558974120
	512	0.174139579	0.544481966	0.703900556
	1024	0.161067288	0.448314224	0.495308810
128	128	0.329087567	0.740856729	0.750036481
	256	0.249318629	0.531832803	0.666810303
	512	0.180433057	0.43880481	0.501473029
	1024	0.197202989	0.503334883	0.726940197
256	128	0.295187289	0.740856729	0.761200138
	256	0.247261403	0.74999756	0.697562540
	512	0.153576778	0.585268432	0.588654231
	1024	0.177807869	0.455159783	0.599046308



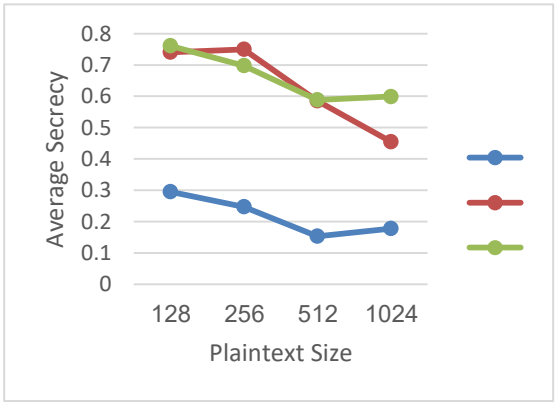
(a) Key = 32 bits



(b) Key = 64 bits



(c) Key = 128 bits



(d) Key = 256 bits

Figure (11) Average Secrecy Value vs. plaintext: (a) key=32 bits (b) key=64 bits (c) key=128 bits (d) key=256 bits

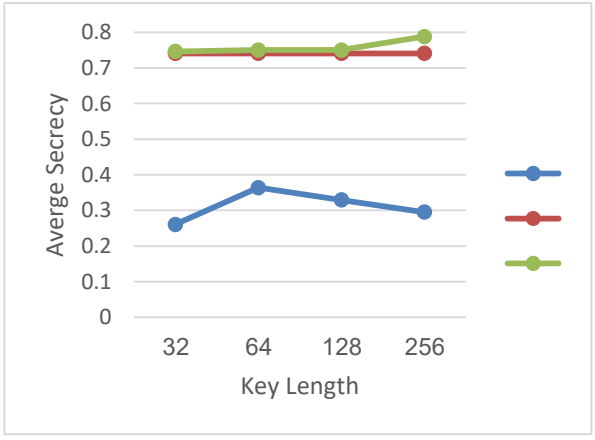
As shown by the Table:1 and figure 11 (a), (b), (c), and (d), improvement RC4 algorithm with EKSA has operative average secrecy than the original RC4 algorithm with KSA, using a variable plaintext size (128,256,512 and 1024 bits), and fixed key length for each phase(32,64,128 and 256 bits).

2. Average secrecy test: A variable key length, Fixed plaintext size.

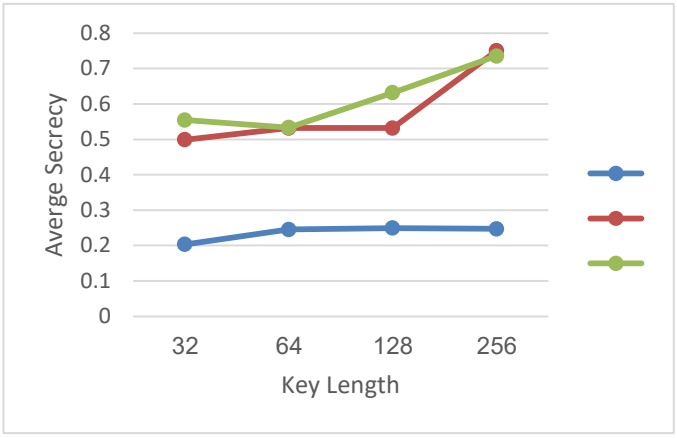
Table 2: Average Secrecy Value vs. Key length

Plaintext size/Bits	Keys Length/Bits	Algorithm		
		Rc4	IRC4 [Naji, M et al. 2018]	ERC4 The Proposed algorithm
128	32	0.260459373	0.740856729	0.745972319
	64	0.363815483	0.740856729	0.750128973
	128	0.329087567	0.740856729	0.749986521
	256	0.295187289	0.740856729	0.787660158
256	32	0.203040633	0.498915456	0.5542876911
	64	0.245275562	0.531832803	0.5330975210
	128	0.249318629	0.531832803	0.6318015831
	256	0.247261403	0.74999756	0.7354801976
512	32	0.20944977	0.406738053	0.6387103681
	64	0.174139579	0.54448966	0.5369997126

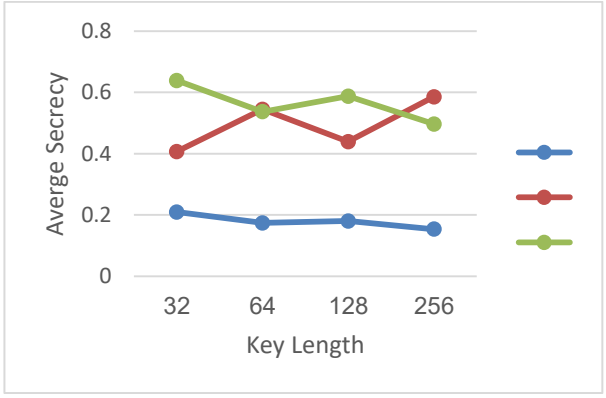
	128	0.180433057	0.43888481	0.5873046951
	256	0.153576778	0.585268432	0.4963542100
1024	32	0.214365643	0.43235869	0.6219854301
	64	0.161067288	0.448314224	0.5019368168
	128	0.197202989	0.50334883	0.6603816294
	256	0.177807869	0.455159783	0.5873099144



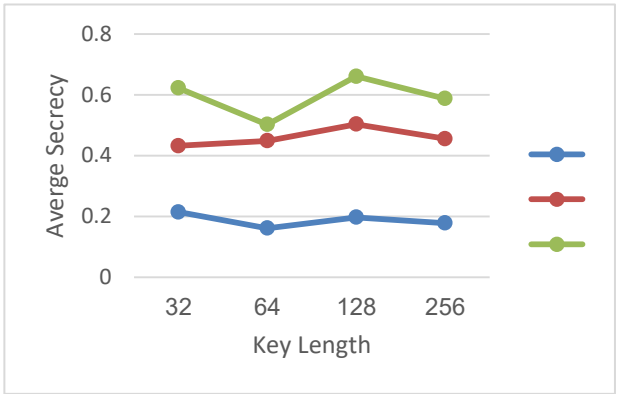
(a) Plaintext = 128 bits



(b) Plaintext = 256 bits



(c) Plaintext = 512 bits



(d) Plaintext = 1024 bits

Figure (12) Average Secrecy Value vs. Key length: (a) Plaintext = 128 bits (b) Plaintext = 256 bits (c) Plaintext =512 bits (d) Plaintext = 1024 bits

As shown by the Table 2 and figure 12 (a), (b), (c) and (d), improved RC4 algorithm with IKSA has better average secrecy than the original RC4 algorithm with KSA, using a variable key length (32.64,128 and 256 bits), and fixed plaintext for each phase (128,256,512 and 1024 bits).

4.2 Analysis of Randomness

The next step , numerous diverse trials are performed to test the statistical properties of the cipher text that created from improved RC4 algorithm with EKSA and it is sensitiveness to elementary conditions. The four different statistical tests (frequency test, serial test, poker test and run test) on a lot of binary sequences of key size (32, 64, 128 and 256 bits) and plaintext size (128 and 1024 bits). These binary sequences succeed in all four tests successfully.

Results are shown in Table 3.

Table 3: Randomness Test for Enhancement RC4 Algorithm

key size\bits	plain size\bits	statistical tests	Degree of freedom	value test	value table	outcome
32	128	Frequency test	1	0.4193	3.8415	success
		Serial Test	2	4.028568425	5.9915	success
		Poker Test	7	11.00941056	15.5073	success
		Run test	2	5.3850197	9.4877	success
	1024	Frequency test	1	0.3999164	3.8415	success
		Serial Test	2	4.0184527	5.9915	success
		Poker Test	31	27.2694130	82.5287	success
		Run test	8	19.810842	82.5287	success
64	128	Frequency test	1	3.308453	3.8415	success
		Serial Test	2	2.6489523	5.9915	success
		Poker Test	7	8.14678029	15.5073	success
		Run test	2	0.394587	9.4877	success
	1024	Frequency test	1	2.015976	3.8415	success
		Serial Test	2	4.994103	5.9915	success
		Poker Test	31	42.058542	82.5287	success
		Run test	8	23.81305	82.5287	success
128	128	Frequency test	1	0.51386	3.8415	success
		Serial Test	2	3.050648	5.9915	success
		Poker Test	7	12.39875	15.5073	success
		Run test	2	5.964231	9.4877	success
	1024	Frequency test	1	0.05289	3.8415	success
		Serial Test	2	4.008153	5.9915	success
		Poker Test	31	51.386106	82.5287	success
		Run test	8	19.472037	82.5287	success
256	128	Frequency test	1	0.95681	3.8415	success
		Serial Test	2	6.94130565	5.9915	success
		Poker Test	7	8.25701	15.5073	success
		Run test	2	1.094529	9.4877	success
	1024	Frequency test	1	0.335791	3.8415	success
		Serial Test	2	7.35961	5.9915	success
		Poker Test	31	36.245923	82.5287	success
		Run test	8	12.924567	82.5287	success

5. CONCLUSION

The Enhancement RC4 algorithm with EKSA based on Logistic Maps with Multi-Parameters is proposed (ERC4 algorithm). This algorithm overcome the weakness of the tradition RC4 with KSA.The average secrecy test for the proposed ERC4 algorithm is more advantaged than the tradition RC4 algorithm. Because of the permutation of the S array are modified to depend on the key random generation based on three chaotic maps (logistic map with two parameters, logistic map with three parameters and logistic map with four parameters) the proposed ERC4 algorithm is distinguished by secrecy, performance and efficiency.

6. CONFLICT OF INTEREST

This work aims to provide a suitable solution for avoid the weakness of the original RC4 by presenting Enhancement RC4 key generation (ERC4) that depending on Logistic Maps with Multi-Parameters. The performance criteria secrecy and randomness were used to compare between the proposed improved algorithm and the original variant.

7. REFERENCES

Journal Articles: Abdul M.S. Rahma, and Zainab M. Hussein, 2015,"Modified RC4 Dual key algorithm based on Irreducible Polynomial". IJETTCS, Vol.4, Issue 2, p: 79-85

A Book: Craincu, B., 2015 "On Invariance Weakness in the KSA Algorithm". Procardia Technology, Elsevier, 19.pp:850-857.

Journal Articles: Fluhrer.S., Mantin, I. and Shamir, 2001, A." Weaknesses in the key scheduling algorithm of RC4".Selected Areas cryptography, 2259, pp: 1-24.

A Book: Mao W., 2003, "Modern Cryptography: Theory and Practice". Prentice Hall PTR.

Journal Articles: Noora Shihab Ahmed, 2016, "Multi-Encryption Technique Based on Permutation of Chaotic System". IJVIPNS-IJENS Vol: 16 No: 01, p:

Journal Articles: K. Prasadh, K. Ramar and R. Gnanajeyaraman, 2009, "Public Key Cryptosystem based on Chebyshev Polynomials", JETR Vol. (7), pp: 122-128.

Journal Articles: Sarab M. Hameed, and Israa Nafea Mahmood, 2016,"A Modified Key Scheduling Algorithm of RC4".Selected Areas in Cryptography. 2259, Iraqi Journal of Science, (ISSN: 0067-2904), Vol. 57, No.1A, pp: 262-267.

A Book: Stallings W., 2011,"Cryptography and Network security Principles and Practices, Fifth Edition". Pearson Education, Inc. Pearson Prentice Hall, USA.

Journal Articles: T.D.B Weerasinghe, 2012, "Analysis of a Modified RC4 Algorithm". IJCA (0975-c xczs8887) Vol.51-No.22, p:

Journal Articles: Naji M. , Ali H. and Noora S. 2018 "Improved RC4 Algorithm Based on Multi-Chaotic Maps". Research Journal of Applied Sciences, Engineering and Technology, Vol 15, Issue 1, (1-6).