Journal of Engineering and Applied Sciences 13 (Special Issue 1): 2317-2321, 2018
ISSN: 1816-949X

S-RADG: A Stream Cipher RADG Cryptography

[1]Salah A.K. Albermany, [2]Duha Amer and Sawsan [2]Kamal
[1]College of Computer Science and Mathematics, University of Kufa, Kufa, Iraq
[2]Department of Computer Science, College of Sciences, Al-Nahrain University, Baghdad, Iraq

Abstract: This study is attempt to develop keyless cryptography exiting algorithm called RADG algorithm into symmetric key cryptography algorithm called S-RADG (Stream RADG (Reaction Automata Direct Graph) algorithm. The result from S-RADG are different cipher texts with same plaintext. The key is generated randomly by using one of stream Cipher algorithms which is LFSR (Linear Feedback Shift Register) method. The random key make S-RADG is difficult to break by attacker. The new algorithm uses to encrypt data in many environments like a cloud computing environment.

Key words: Stream cipher, LFSR, RADG, cryptography, security, computing environment

INTRODUCTION

Cryptography is an essential tool for security. Previously was the purpose of cryptography is to hide text messages during the war. But it has recently become necessary to secure the transfer of information between online networks with complete secrecy. cryptography is the science is used for encryption and decryption, so that, the information be secured and the phenomenon for the sender and the addressee only. Cryptography goals are to satisfy confidentiality, integrity, authentication, non-repudiation. There are two types of cryptography, symmetric key cryptography which use one key for encryption and decryption and public key cryptography which use different two keys one for encryption is called public key, another for decryption is called private key (Paar and Pelzl, 2010; Som and Ghosh, 2012). In symmetric key there are two types of ciphering, stream cipher and block cipher (Paar and Pelzl, 2010). In our study, we will talk about stream cipher, stream cipher encrypts one bit at time, in strong security. The new method is merged between stream cipher and exiting method is called Reaction Automata Direct Graph (RADG) method (Albermany and Safda, 2014) by use LFSR to generate key of new method, we convert keyless RADG method into symmetric key method with keeping RADG design properties.

This attempt increase efficient of encryption, analysis of security for a new method show this by satisfy confidentiality and integrity (Fig. 1 and 2).

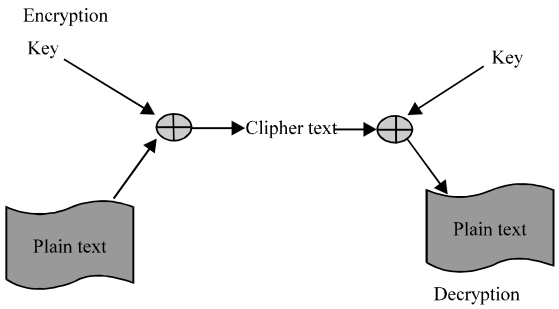

Fig. 1: Encryption/Decryption in steam cipher

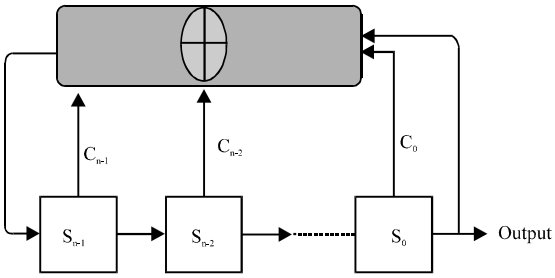

Fig. 2: Basic diagram of LFSR

MATERIALS AND METHODS

Stream cipher RADG (S-RADG): RADG is algorithm that represent by set of tuples {R, Q, Σ, Ψ, J, t} where R is reaction state that have λ of length which have λ of values, Q is standard design state that have n length, also have λ of values, λ represent input data, Σ represent

Corresponding Author: Salah A.K. Albermany, College of Computer Science and Mathematics, University of Kufa, Kufa, Iraq

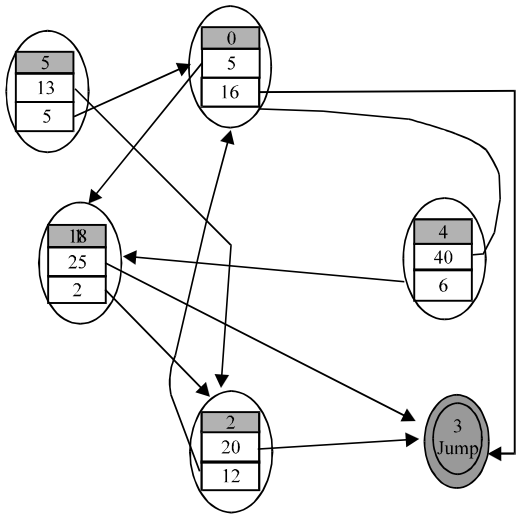

Fig. 3: Implementation RADG algorithm

output transition, J is jump state which is proper of Q state that have K length which is have no value, just transmit from one state to another in Q state, T represent transition function (Safdar, 2014). The following example in Fig. 3 illustrated how RADG is research where m = 2, n = 2, k = 1, suppose λ = 2. Let state number 4, 5 number are R states and state number 0, 1, 2, 3 are Q states.

Encryption in RADG algorithm which is:

- Suppose the encrypted message or plain text is 0111
- T(0, 0) = (1, 5)
- T(1, 1) = (2, 22)
- T(2, 1) = (0, 12)
- T(0, 1) = (4, 16)
- The cipher text is 5, 22, 12 and 16

The study tried to design a new algorithm based on original RADG. The new algorithm is called S-RADG (stream RADG) that used linear feedback shift register to generate key (Fig. 4). Therefore, the S-RADG algorithm is key scheme on the contrary of RADG algorithm which is keyless scheme. SRADG contain 6 tuples as original RADG which are {R, Q, Σ, Ψ, J, T}. Each single state in R or Q states have λ of values, the value either 0 or 1 in each state. The proposed algorithm used message text in stream cipher as data and also use key generated by Linear Feedback Shift Register (LFSR) as a S-RADG key. The current state in R and Q is transmit by certain function is called transition function. To determine the next state be done by a certain function.

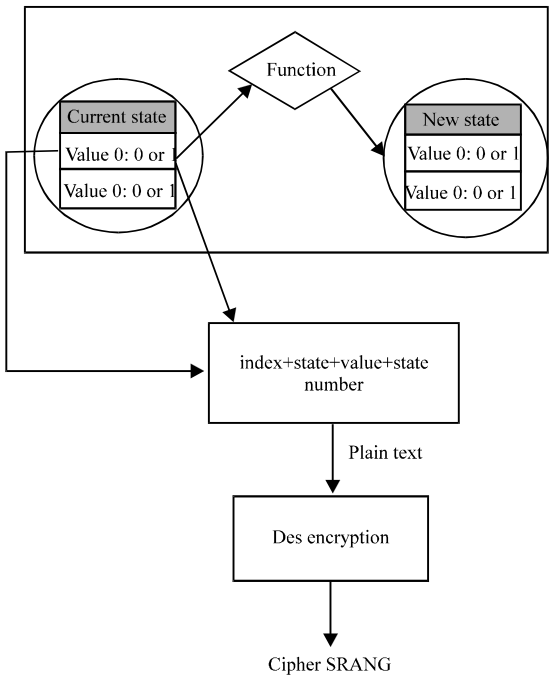

Fig. 4: S-RADG algorithm

Table 1: LFSR implement notations

Notations	Details
L	The length of the stages
S_n	The sequences of the output
C_i	The initial value

Key generation: S-RADG key is sequence that generated by LFSR algorithm. LFSR is linear function in Eq. 1 (Klein, 2013; Paar and Pelzl, 2010).

$$f(\vec{S}) = \sum_{i=0}^{n-1} C_i S_i \quad (1)$$

When, the initial value generated is called seed, then used to determine the output. The input sequence with length n is $S_0, S_1, .., S_{n-1}$. Equation 2 that used for linear function to determine out put is (Klein, 2013; Paar and Pelzl, 2010):

$$S_n + L = \sum_{i=0}^{L-1} C_i S_n, \forall n \geq 0 \quad (2)$$

Table 1 explain the coefficients of Eq. 2. We can know number of stage by the primitive polynomial.

Number of stage = polynomial degree

Suppose we have LFSR with degree d = 4, the path shown in Fig. 5 (Klein, 2013; Paar and Pelzl, 2010). The

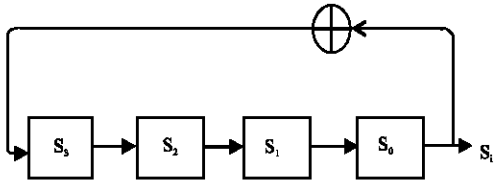

Fig. 5: LFSR with degree 4

Table 2: Sequence of stages of LFSR

t	S_2	S_2	S_1	S_0
0	1	0	0	1
1	0	1	0	0
2	0	0	1	0
3	1	0	0	1

sequence of inner state is S_t, it is shifted one bit to right, the leftmost is the output. The output is computed by XOR for summation of previous stage. If we suppose seed = 1001, Table 2 illustrates how LFSR sequence is generated.

Transition function: Data is message text from stream cipher which is converted into binary bits. The generated Key XORded with data to form transition function T. Figure 6 illustrated transition function.

S-RADG implementation: Implementation include provide algorithm of key generation, encryption and decryption (Alogrithm 1-3).

Alogrithm 1; Key generation:

Step 1: seed←random initial bit
Step 2: shift all value over one place and dropping the last value
Step 3: set the computed value in first position
Step 4: repeat process
Step 5: k sequences←k_i

Alogrithm 2; Encryption algorithm:

Input: Sequence of data M = {m_0, m_1,..., $m_{|M|}$} Sequence of key K = {K_0, K_1, ..., $K_{|M|}$}
Output: sequence of cipher blocks C←{C_0, C_1, ...}

Step 1: C←⌀, $q_{current} = q_{rand}$, L←0
Step 2: while (L<|M|)
Step 3: index←F(m_L,K_L)
Step 4: (state value)$_L$←getvalue (q| old,inde)
Step 5: b_n←block (index, state value, q_{old}), n = 8
Step 6: C_1←cipher[block]
Step 7: Add (C_1, C_1 (L))
Step 8: if (q| oldϵJump):
q_{new}←R
Step 9: else
Step 10: q_{new}←q_{old}
Step11: return C

Alogrithm 3; Decryption algorithm:

Input: sequence of cipher text C←{C_0, C_1,..., $C_{|L|}$}, sequence of key K←{K_0, K_1,..., $K_{|C|}$}
Output: sequence of plain text p←{p_0, p_1,..., }

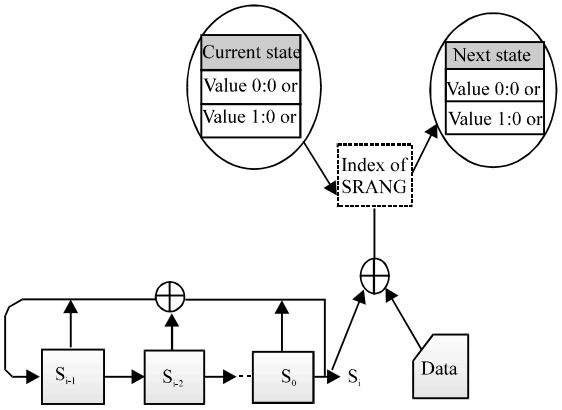

Fig. 6: Transition function of S-RADG

Step 1: $block_L$←decipher[C_L], L←|C|
Step 1: Search (q_{old}, Q)
Step 2: While (L≤ = 0)
Step 3: m_L←F (index, K_L)
Step 4: Add(P, m_L)
Step 5: reverse sequence (P_0, P_1,..., $P_{|t|}$)
Step 6: return P

RESULTS AND DISCUSSION

Security and performance analysis: The security and performance analysis of S-RADG clarified by implement example of 12 states with m = 4, n = 6, k = 2, all states in Fig. 6 have λ = 2. Where m = |R|, n = |Q|, k = |J|.

Security analysis

Confidentiality: Confidentiality prevent wrong people to access to information that mean, it equally to privacy of information (Stamp, 2011). Encryption and decryption can be used for confidentiality in the example below.

Example: The data to be encrypted is hi and the key is generated by LFSR algorithm which used for both encryption and decryption, so at first, we must compute the sequences of random key as illustrated in Table 3. Let suppose the primitive polynomial is $P(x) = x^4+x+1$. The stages of LFSR illustrated in Fig. 7. From Table 3, the key sequences are: 1010 1101 1110 1111.

Encryption data: First, we convert message into binary value. hi 0110 1000 0110 1001. It's important to see Table 4 and 5 together to understand how the encryption is done. A column block in Table 4, include 8 bits binary data, the left two bits are the index and state value and the others bits are address which is a state number after converted into binary values. Then, encrypted the blocks data by using DES algorithm to result a cipher data of S-RADG. DES algorithm encrypt 64 bits at time

Table 3: Sequences of LFSR

t	S_2	S_2	S_1	S_0
0	1	0	1	0
1	1	1	0	1
2	1	1	1	0
3	1	1	1	1

Table 4: S-RADG encryption

i	State number	Data	Key	Data⊕Key	Jump	Block	Cipher data
0	3	0	1	1	-	10000011	10011110
1	1	1	0	1	-	11000001	01100101
2	10	1	1	0	J	00001010	01001111
3	8	0	0	0	-	00001000	00101011
4	2	1	1	0	-	01000010	00101010
5	10	0	1	1	J	10001010	10001111
6	2	0	0	0	-	01000010	10100110
7	11	0	1	1	J	10001011	11111001
8	1	0	1	1	-	11000001	01010110
9	9	1	1	0	J	01001001	11100111
10	1	1	1	0	-	00000001	11011110
11	2	0	0	0	-	01000010	00101100
12	10	1	1	0	J	00001010	10001111
13	8	0	1	1	-	10001000	11101010
14	2	0	1	1	-	11000010	11011000
15	10	1	1	0	J	00001010	10001100

Table 5: S-RADG Decryption

I	Cipher data	Block	State number	State value	T^{-1}or search	Index	Key	Data
15	10001100	00001010	10	0	search	0	1	1
14	11011000	11000010	2	1	True	1	1	0
13	11101010	10001000	8	0	True	1	1	0
12	10001111	00001010	10	0	False then Search in R	0	1	1
11	00101100	01000010	2	1	True	0	0	0
10	11011110	00000001	1	0	True	0	1	1
9	11100111	01001001	9	1	False then Search in R	0	1	1
8	01010110	11000001	1	1	True	1	1	0
7	11111001	10001011	11	0	False then Search in R	1	1	0
6	10100110	01000010	2	1	True	0	0	0
5	10001111	10001010	10	0	False then Search in R	1	1	0
4	00101010	01000010	2	1	True	0	1	1
3	00101011	00001000	8	0	True	0	0	0
2	01001111	00001010	10	0	False then Search in R	0	1	1
1	01100101	11000001	1	1	True	1	0	1
0	10011110	10000011	3	0	True	1	1	0

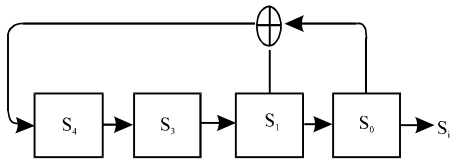

Fig. 7: LFSR length

with 56 bits of key (the cipher key is normally given as a 64-bits key in which 8 extra bits are the parity bits which are dropped before the actual key-generation process), so if, the data size to be encrypted is greater than 64 bits, the

Table 6: Values of R set

Number of states	First vlue	Second value
1	0	1
2	1	1
3	1	0
4	-	4
5	-	5
6	0	1
7	1	0
8	0	0

Table 7: Values of Q set

Number of states	First value	Second value
9	1	1
10	0	0
11	1	0
12	0	1

Table 8: The transition

Number of states	First value	Second value
1	2	4
2	5	5
3	7	1
4	-	-
5	-	-
6	2	8
7	4	3
8	2	2
9	1	6
10	8	2
11	3	1
12	7	8

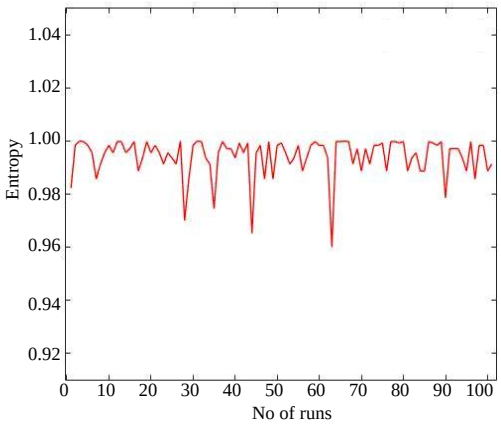

Fig. 8: Entropy for message length of 128 bits

DES will encrypt the first 8 blocks then return to encrypt other blocks, so if the other blocks is less than 8 blocks, needed to extended it by using padding DES. The DES encrypt the first 64 bits then repeat her research for other 64 bits by use same S-RADG key (Table 6 and 7).

Decryption data: A decryption process in a receiver side begin backward to result plain data. It start form cipher data to decrypt it by using DES algorithm from the 8 bits result, the receiver obtain state number. Then by XORded

index with the key to get original data that sent by a sender. Table 4-8 together to understand how exactly decryption process is done.

Integrity: Integrity means detect unauthorized user in writing (i.e., modification of data) (Stamp, 2011). In S-RADG algorithm, integrity is satisfy by notice that any modification on cipher data can detect easily. If the hacker modifies cipher block marked by unauthorized user, the decryption process of the cipher data will failed because the search process is failed where not found the value which must give the correct plain data and in this case cannot be continue to complete decryption process.

Performance analysis: Entropy is used to measured uncertainty in the system (Bat-Erdene *et al*., 2017). Using entropy to analysis performance of S-RADG method, S-RADG is run 100 times to show how it performs in terms of entropy for different cipher text values. Figure 8 show the entropy between individual cipher data for message length of 16 bits.

CONCLUSION

The proposed algorithm improved RADG implementation level, makes cryptography stronger, since, use stream cipher to generate a good keystream based on LFSR to generate random sequences of key. The LFSR satisfy security since add binary sequences to data, this add noise to information and makes it difficult to detect by unauthorized users. LFSR have highly long sequences may reach to 10^{50}, this makes S-RADG key strong and the hacker need long time to detect. The security analysis prove that S-RADG successfully provides confidentiality, Integrity which are aspects of security. The results obtained from performance analysis have shown that for the same plain data have the different cipher data, this proves the strong cryptography of S-RADG where the process of breaking the code within large systems requires a more effort compared to the RADG scheme that not require any key.

REFERENCES

Albermany, S.A. and G.A. Safda, 2014. Keyless security in wireless networks. Wirel. Pers. Commun., 79: 1713-1731.

Bat-Erdene, M., T. Kim, H. Park and H. Lee, 2017. Packer detection for multi-layer executables using entropy analysis. Entropy, 19: 125-125.

Klein, A., 2013. Stream Ciphers. Springer, London, England, ISBN:978-1-4471-5078-7, Pages: 399.

Paar, C. and J. Pelzl, 2010. Understanding Cryptography: A Textbook for Students and Practitioners. Springer, Berlin, Germany,.

Som, S. and S. Ghosh, 2012. A stream cipher cryptosystem based on linear feedback shift register. Intl. J. Math. Arch., 3: 362-372.

Stamp, M., 2011. Information Security: Principles and Practice. 2nd Edn., John Wiley & Sons, Hoboken, New Jersey, ISBN:978-0-470-62639-9,.