Modified the RC4 Stream Cipher Algorithm Based on Irreducible Polynomial

Dr. Abdul Monem Saleh Rahma
Computer Science Department / University of Technology/Baghdad
Email: monem.rahma@yahoo.com
Zainab Mohammed Hussein
Computer Science Department / University of Technology/Baghdad
Email: zainab_mohammed85@yahoo.com

Received on:24/9/2014 & Accepted on: 7/5/2015

ABSTRACT

The increase in the speed of computers and adoption on it as means of encryption (send and receive encrypted data), which led to the development of modern encryption techniques such as (stream cipher and block cipher). This increase in speed of the computers has increased the strength of the attacking techniques for stream cipher which is built according to two ways: (random key generation and XOR operation).

This paper proposes approach in modifying RC4 stream cipher algorithm based on irreducible polynomial which is used in modern encryption methods such as AES.

The proposed algorithm achieves best results, it provides high level of complexity, to decrypt an encrypted message is composed of 8-bit the attacker needs $30*2^8$ probability of keys at minimum, this mean the proposed algorithm will increase the complexity of the algorithm 30 times.

Keywords: Stream cipher, Rc4 algorithm, irreducible polynomial.

تعديل خوارزمية التشفير (RC4) بالاعتماد على متعددات الحدود الغير قابلة للاختزال

الخلاصة:

زيادة سرعة أجهزة الكمبيوتر واعتمادها كوسيلة للتشفير (إرسال واستقبال البيانات المشفرة)، أدى إلى تطوير تقنيات التشفير الحديثة مثل (التشفير الانسيابي والتشفير الكتلي). هذه الزيادة في سرعة أجهزة الكمبيوتر زاد من قوة تقنيات مهاجمة التشفير الانسيابي التي بنيت وفقا لطريقتين أساسيتين: (التوليد العشوائي للمفاتيح) وعملية (xor).

في هذا البحث تم اقتراح طريقة جديدة لتعديل خوارزمية التشفير الانسيابي (RC4) بالاعتماد على متعددات الحدود الغير قابلة للاختزال التي تستخدم في طرق التشفير الحديثة مثل خوارزمية (AES) الطريقة المقترحة حققت افضل النتائج وقد وفرت مستوى عال من التعقيد ، لفك شفرة متكونة من (8 bit)، المهاجم يحتاج الى ($30*2^8$) احتمالات من المفاتيح لفك هذه الشفرة هذا يعني ان الخوارزمية المقترحة زادت من تعقيد الخوارزمية الاصلية ب 30 مرة.

INTRODUCTION

In a stream cipher approach, encryption a digital data stream one bit or one byte at a time. It breaks the message M into successive characters or bits. M1,M2,……,and encipher each Mi with its element Ki of a key stream K= K1,K2,K3 …… That is EK (M) = EK1(M1) EK2(M2).Basic idea of stream cipher comes from One-Time-Pad cipher using XOR operator on the plain text and the key to generate the ciphertext [1]. A stream cipher treats the message as a stream of bits and performs mathematical functions on each bit individually. When using a stream cipher, a plaintext bit will be transformed into a different ciphertext bit each time it is encrypted. Stream ciphers use keystream generators, which produce a stream of bits that is XORed with the plaintext bits to produce ciphertext [2].Stream ciphers are faster and smaller to implement than block ciphers, however, they have an important security gap. If the same key stream is used, certain types of attacks may cause the information to be exposed [1].

RC4 is the one of popular stream cipher algorithm, such as (Vigenere cipher and Vernem cipher). Rc4 algorithm depends on classical operation (XOR).

This paper modifies the Rc4 algorithm based on the multiplication of irreducible polynomial mathematics.

RC4 Stream Cipher algorithm, historical review.

Ron Rivest [1], one of the inventors of RSA introduced the Rc4 algorithm in 1987. Rc4 is acronym for ”Rivest Cipher 4”, it is also known “Ron’s Code 4”.

The algorithm in RC4 is optimized for software implementation.RC4 produces a keystream byte at each step.[3]

Algorithm 1 (Rc4 Stream Cipher Encryption and Decryption)

Input [plaintext] and [key]

Output [cipher text]

Step 1: /Initialization /

for i = 0 to 255

S[i] = i;

T[i] = K[i mod key];

Next i;

Step 2: / Initial Permutation of S /

Set j = 0;

For i = 0 to 255

j = (j + S[i] + T[i]) mod 256;

Swap (S[i], S[j]);

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];

Step 4:/The process/

Step 4.1: encryption C = P k $\oplus$

Step 4.1: decryption P = C k $\oplus$

Step 5:/ End.

Irreducible Polynomial Mathematics

An (irreducible polynomial**)** is a polynomial f(x) over afield F if and only if f(x) cannot be expressed as a product of two polynomials both over F, and both of degree lower than that of f(x).A field is a commutative ring in which all nonzero elements have a multiplicative inverse. For a given prime, p, the finite field of order p, GF (p) is defined as the set Z_p of integers {0, 1, . . . , p − 1}, together with the arithmetic operations modulo p .The finite field of order p^n is generally written GF (p^n). GF is acronym for Galois field .There are a limited number of elements in the finite field; all operations performed in the finite field result in an element within that field. A polynomial f(x) in GF (2^n) is represented as:

$$f(x) = (a_{n-1} x^{n-1} + a_{n-2} x^{n-2} +\ldots+a_1x+a_0) \quad 1 \ [1].$$

Can be exclusively represented by its n binary coefficients (a_{n-1} , $a_{n-2}\ldots a_0$). Therefore, every polynomial in GF (2^n) can be represented by an n-bit number.

This paper is concerned with the finite field GF (2^8).

Each element of GF (2^8) is a polynomial of degree 7 with coefficients in GF (2). Thus, the coefficients of each term of the polynomial can take the value (0) or (1). Given that there are 8 terms in an element of GF (2^8), an element can be represented by bit string of length 8, where each bit represents a coefficient. The least significant bit is used to represent the constant of the polynomial, and going from Right to left, represents the coefficient of xi by the bit bi where bi is i bits to the left of the least significant bit. For example, the bit string (10101011) represents ($x^7+x^5+x^3$ + x+1). For convenience, a term xi is found in the expression if the corresponding coefficient is 1. The term is omitted from the expression if the coefficient is 0.

Addition of two elements in GF (2^8) is simply accomplished using eight *XOR* gates to add corresponding bits.

Multiplication in GF (2^8) can be accomplished by first multiplying each term of the first polynomial with all of the terms of the second polynomial. Each of these products should be added together. If the degree of the new polynomial is greater than 7, then it must be reduced modulo using one of the irreducible polynomials which are represented in Table (1).

The extended Euclidean algorithm can be adapted to find the multiplicative inverse of a polynomial and (gcd) of two polynomials. Suppose two polynomials a(x), b(x), and m(x) is an irreducible polynomial, a(x) and b(x) are mutual inverses If [a(x)*b(x) mod m(x) = [1].

Multiplication Table

To construct the multiplication finite field GF (2^8) requires choosing irreducible polynomial of degree 8 illustrated in table (1). Each element of the finite field set other than 0 has a multiplicative inverse [1].

Table (1) Some of Irreducible Polynomials

1	$x^8 + x^4 + x^3 + x + 1$	**16**	$x^8 + x^6 + x^5 + x^4 + x^3 + x + 1$
2	$x^8 + x^4 + x^3 + x^2 + 1$	**17**	$x^8 + x^7 + x^2 + x + 1$
3	$x^8 + x^5 + x^3 + x + 1$	**18**	$x^8 + x^7 + x^3 + x + 1$
4	$x^8 + x^5 + x^3 + x^2 + 1$	**19**	$x^8 + x^7 + x^3 + x^2 + 1$
5	$x^8 + x^5 + x^4 + x^3 + 1$	**20**	$x^8 + x^7 + x^4 + x^3 + x^2 + x + 1$
6	$x^8 + x^5 + x^4 + x^3 + x^2 + x + 1$	**21**	$x^8 + x^7 + x^5 + x + 1$
7	$x^8 + x^6 + x^3 + x^2 + 1$	**22**	$x^8 + x^7 + x^5 + x^3 + 1$
8	$x^8 + x^6 + x^4 + x^3 + x^2 + x + 1$	**23**	$x^8 + x^7 + x^5 + x^4 + 1$
9	$x^8 + x^6 + x^5 + x + 1$	**24**	$x^8 + x^7 + x^5 + x^4 + x^3 + x^2 + 1$
10	$x^8 + x^6 + x^5 + x^2 + 1$	**25**	$x^8 + x^7 + x^6 + x + 1$
11	$x^8 + x^6 + x^5 + x^3 + 1$	**26**	$x^8 + x^7 + x^6 + x^3 + x^2 + x + 1$
12	$x^8 + x^6 + x^5 + x^4 + 1$	**27**	$x^8 + x^7 + x^6 + x^4 + x^2 + x + 1$
13	$x^8 + x^4 + x^3 + x + 1$	**28**	$x^8 + x^7 + x^6 + x^4 + x^3 + x^2 + 1$
14	$x^8 + x^4 + x^3 + x^2 + 1$	**29**	$x^8 + x^7 + x^6 + x^5 + x^2 + x + 1$
15	$x^8 + x^6 + x^5 + x^4 + x^2 + x + 1$	**30**	$x^8 + x^7 + x^6 + x^5 + x^4 + x + 1$

The Finite field multiplication is achieved by multiplying the two polynomials for the two elements, if multiplication result in a polynomial of degree greater than n-1, then the polynomial is reduced modulo some irreducible polynomial m(x) of degree n. That is, divided by m(x) and keep the remainder. The definition of irreducible polynomial is a polynomial f(x) over a field F is called irreducible if and only if f(x) cannot be expressed as a product of two polynomials, both over F, and both of degree lower than that of f(x) [1].

Table (2) (multiplication in GF (2^8) with the irreducible Polynomial $m(x)(x^8 + x^4 + x^3 + x+1)$)

			00000001	00000010	00000011	00000100	…	11111100	11111101	11111110	11111111
Binary	**Poly.**	*	1	x	x+1	x^2	…	252	253	254	255
00000001	1	1	1	x	x+1	x^2	…	$x^7+x^6+x^5+x^4+x^3+x^2$	$x^7+x^6+x^5+x^4+x^3+x^2+1$	$x^7+x^6+x^5+x^4+x^3+x^2+x$	$x^7+x^6+x^5+x^4+x^3+x^2+x+1$
00000010	x	2	x	x^2	x^2+x	x^3	…	$x^7+x^6+x^5+x+1$	$x^7+x^6+x^5+1$	$x^7+x^6+x^5+x^2+x+1$	$x^7+x^6+x^5+x^2+1$
00000011	x+1	3	x+1	x^2+x	x^2+1	x^3+x^2	…	$x^4+x^3+x^2+x+1$	$x^4+x^3+x^2$	x^4+x^3+1	x^4+x^3+x
00000100	x^2	4	x^2	x^3	x^3+x^2	x^4	…	$x^7+x^6+x^4+x^3+x^2+1$	$x^7+x^6+x^4+x^3+1$	$x^7+x^6+x^4+x^2+1$	$x^7+x^6+x^4+1$
….	…….	….	….	…	…	…	…	…	…	…	…
….	…….	….	….	…	…	…	…	…	…	…	…
….	…….	….	….	…	…	…	…	…	…	…	…
….	…….	….	….	…	…	…	…	…	…	…	…
….	…….	….	….	…	…	…	…	…	…	…	…
11111100	$x^7+x^6+x^5+x^4+x^3+x^2$	252	$x^7+x^6+x^5+x^4+x^3+x^2$	$x^7+x^6+x^5+x+1$	$x^4+x^3+x^2+x+1$	$x^7+x^6+x^4+x^3+x^2+1$	…	x^4+x^2+1	$x^7+x^6+x^5+x^3+x$	$x^7+x^6+x^5+x^4+x^2+1$	x^3+1
11111101	$x^7+x^6+x^5+x^4+x^3+x^2+1$	253	$x^7+x^6+x^5+x^4+x^3+x^2+1$	$x^7+x^6+x^5+1$	$x^4+x^3+x^2$	$x^7+x^6+x^4+x^3+1$	…	$x^7+x^6+x^5+x^3+x$	x^4+x^2+x+1	x^3+x+1	$x^7+x^6+x^5+x^4+x^2+x$
11111110	$x^7+x^6+x^5+x^4+x^3+x^2+x$	254	$x^7+x^6+x^5+x^4+x^3+x^2+x$	$x^7+x^6+x^5+x^2+x+1$	x^4+x^3+1	$x^7+x^6+x^4+x^2+1$	…	$x^7+x^6+x^5+x^4+x^2+1$	x^3+x+1	x^4+x	$x^7+x^6+x^5+x^3+x^2$
11111111	$x^7+x^6+x^5+x^4+x^3+x^2+x+1$	255	$x^7+x^6+x^5+x^4+x^3+x^2+x+1$	$x^7+x^6+x^5+x^2+1$	x^4+x^3+x	$x^7+x^6+x^4+1$	…	x^3+1	$x^7+x^6+x^5+x^4+x^2+x$	$x^7+x^6+x^5+x^3+x^2$	x^4+x+1

Table (3) multiplication inverse in GF (2^8) with irreducible polynomial $m(x) = x^8 + x^4 + x^3 + x + 1$

k	k^{-1}
1	1
x	$x^7+x^3+x^2+1$
x+1	$x^7+x^6+x^5+x^4+x^2+x$
x^2	$x^7+x^6+x^3+x+1$
….	….
….	….
….	….
….	….
$x^7+x^6+x^5+x^4+x^3+x^2$	$x^7+x^6+x^3+x^2+1$
$x^7+x^6+x^5+x^4+x^3+x^2+1$	x^4+x^3+x
$x^7+x^6+x^5+x^4+x^3+x^2+x$	x^6+1
$x^7+x^6+x^5+x^4+x^3+x^2+x+1$	$x^4+x^3+x^2$

Modified Rc4 stream cipher algorithm based on Irreducible Polynomial

In this work, the Encryption operation is replaced XOR of RC4 algorithm with the equation

C = P * k mod irreducible polynomial …. (2)

Where

C = Cipher text

P = Plaintext
k= key
The decryption operation is
P = C *k^{-1} mod irreducible polynomial ….(3)
k^{-1} = key inverse.
The mathematical operation addition & multiplication of irreducible polynomial is based on mathematical theory of GF (2^8).

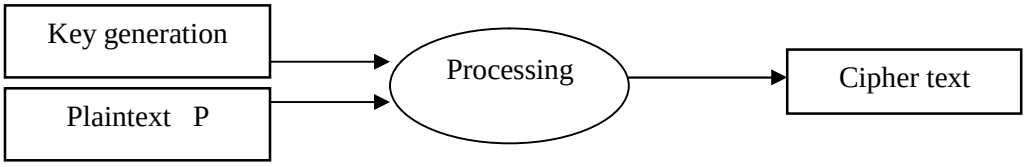

Figure (1) Block diagram of encryption system

Where
Processing = Encryption & Decryptio

The Modified Rc4:

Algorithm 2 (Modified Rc4 Stream Cipher Encryption and Decryption)
Input [plaintext] and [key]
Output [cipher text]
Step 1: /Initialization /
for i = 0 to 255
S[i] = i;
T[i] = K[i mod key];
Next i;
Step 2: / Initial Permutation of S /
Set j = 0;
For i = 0 to 255
j = (j + S[i] + T[i]) mod 256;
Swap (S[i], S[j]);
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];
Step 4:/The process/

1. Convert the value of plaintext and key to binary then to polynomial equation.
2. Determine (k & k^{-1}) from table 3.
3. To encrypt use equation 2.
4. To decrypt use equation 3.

Step 5:/End/

The following example illustrates our technique in encryption & decryption part:
Input: Plaintext [ali] and Key [vpn]
The irreducible polynomial is [$x^8+x^4+x^3+x+1$] randomly.

1. Find the ascii of [a,l,i]=[97,108,105] respectively.
2. Find the ascii of [v,p,n]=[118,112,110] respectively.
3. After perform **initialization Permutation** and **Key Scheduling Algorithm on algorithm 1**, the keys that was obtained after these two processes are [59,32,35].
4. Convert plaintext to binary then to polynomial

4.1) a=97= [01100001] = [x^6+x^5+1].
4.2) l=108= [01101100] =[$x^6+x^5+x^3+x^2$].
4.3) i=105= [01101001] = [$x^6+x^5+x^3+1$].

5. Convert Key to binary then to polynomial.

5.1) 59= [00111011] = [$x^5+x^4+x^3+x+1$].
5.2) 32=[00100000] = [x^5]
5.3) 35=[00100011] = [x^5+x+1]

6. **To encrypt** (C =P*k) mod irreducible polynomial.

C1=(x^6+x^5+1)*($x^5+x^4+x^3+x+1$) mod($x^8+x^4+x^3+x+1$) The result is = ($x^{11}+x^{10}+x^9+x^7+x^6+x^{10}+x^9+x^8+x^6+x^5+x^5+x^4+x^3+x+1$) = ($x^{11}+x^8+x^7+x^4+x^3+x+1$)which has a degree(11>7) so, The largest element appeared after multiplication process is (x^7) because the results of the encryption operation are calculated using GF (2^8) operations where, each element of GF (2^8) is a polynomial of degree 7 with coefficients in GF (2). Thus, if the result of multiplication with degree larger than 7, then the resulted polynomial should be reduced through dividing it by one of the irreducible polynomial in table (1) to get the remainder which will be used as a resulted polynomial.

$$
\begin{array}{r|l}
 & x^3+1 \\
\hline
x^8+x^4+x^3+x+1 & x^{11}+x^8+x^7+x^4+x^3+x+1 \\
 & x^{11}+x^7+x^6+x^4+x^3 \\
\hline
 & x^8+x^4+x^3+x+1 \\
 & x^8+x^6+x+1 \\
\hline
 & C1=x^6+x^4+x^3
\end{array}
$$

C2=P_2*k_2 = (x^5)*($x^6+x^5+x^3+x^2$) mod($x^8+x^4+x^3+x+1$) =($x^5+x^3+x^2+x+1$)
C3=P_3*k_3 = (x^5+x+1)*($x^6+x^5+x^3+x$) mod($x^8+x^4+x^3+x+1$) =($x^5+x^4+x^2$)

7. **To decrypt** (P =C* k^{-1}) mod irreducible polynomial.

P_1 =C_1* k_1^{-1} =($x^6+x^4+x^3$)*($x^6+x^5+x^3+x^2+x+1$) mod ($x^8+x^4+x^3+x+1$) =(x^6+x^5+1) which is equivalent to (01100001)= *97=**a***
P_2 =C_2* k_2^{-1} =($x^5+x^3+x^2+x+1$)*($x5+x^4+x^3+x$) mod($x^8+x^4+x^3+x+1$) = ($x^6+x^5+x^3+x^2$) which is equivalent to (01101100) *=108=**l***

$P_3=C_3 * k_3^{-1} = (x^5+x^4+ x^2)*(x^7+x^6+x^5+x^4+1) \bmod (x^8 +x^4 +x^3 +x+1) = (x^6+x^5+ x^3+ 1)$ which is equivalent to (0110101) =*105*= ***i***

Discussion:

Security

The number of possibility of keys to decrypt RC4 algorithm is: 2^8 possibilities.
The following example illustrates number of possibilities of keys that the attacker needs to decrypt the modified algorithm:
To construct GF (2^8), there are 30 irreducible polynomials of degree 8.
The encryption equation is: [C= P * k mod (irreducible polynomial)], k is key, the size is 8bit, and the number of possible keys is (2^8).
For example. To decrypt an encrypted message is composed of 8-bit the attacker needs $30*2^8$ probability of keys at minimum = 7680 probability.

To decode an encrypted message is composed of 700-byte the attacker needs (5376000) probability of keys.

The security has been increased, the most important secure key is the utilized irreducible polynomial in the modified scheme since we have 30 irreducible polynomials, and one can use any of those as explained in (table 1). This is hard to be attacked by the crypto-analysis.

Table (4): Run time Comparison using 256 bit key

Systems	Encryption Time (S:ms)	Decryption Time (S:ms)
RC4	56.36	3.06
Propose RC4	2.38	0.52

Randomness Test [6]

Table (4.2): Encrypted Data Test.

No.	Tests	Condition	Result Value	Result
Rc4	Frequency	<= 0.01	0.9124	PASS
	Frequency within block	<= 0.01	1.000012	PASS
	Runs	<= 0.01	0.8819	PASS
	Longest Run	<= 0.01	1.0022	PASS
Modified RC4	Frequency	<= 0.01	0.946	PASS
	Frequency within block	<= 0.01	1.000012	PASS
	Runs	<= 0.01	0.49589	PASS
	Longest Run	<= 0.01	1.0022	PASS

CONCLUSION

Because of Increase in the speed and the evolution of computers, the strength of the crypto-analysis has increased , the need arises to the increase the complexity of algorithms by using tiny block instead of using one bit and using alternative processes to (XOR).

In this paper it is better to use GF (2^8) instead of use GF (2) because using (modular arithmetic) with prime number has exceptions in this algorithm which works on 8 bits data at a time, and is executed division with 8 bits, integers can be represented in the range 0 through 255. However, (256,255,254,253,252) is not a prime numbers, so that if arithmetic is performed in (Z_{256} Z_{255} Z_{254} Z_{253}) (arithmetic modulo 256,255,253,252), this set of integers will not be a field. The closest prime number less than this numbers is 251.Thus, the set Z_{251}, using arithmetic modulo 251, is a field. However, in this case the 8-bit patterns representing the integers 251 through 255 would not be used.

The modified algorithm still stream cipher because encryption a digital data stream one bit or one byte at a time.

REFERENCES

[1]. Stallings W., "Cryptography and Network Security Principles and practices, Fifth Edition", Pearson Education, Inc. Pearson Prentice Hall ,USA,2011.

[2]. Harris S.," CISSP Exam Guides, Sixth Edition", McGraw-Hill Companies,2013

[3]. Stamp M. "Information Security Principles And Practice " John Wiley & Sons,2006.

[4]. Mao w.," Modern Cryptography: Theory and Practice", Prentice Hall PTR, 2003.

[5]. Henk C.A., van Tilborg, "Encyclopedia Of Cryptography And Security" Eindhoven University of Technology, The Netherlands.

[6]. Bassham L. E., " A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications", National Institute of Standards and Technology, Information Technology Laboratory, Computer Security Division, USA,2008.