A Modification on Rivest Cipher (RC4) Algorithm Against FMS Wired Equivalent Privacy (WEP) Attack

Mohammed A. Noaman* May Sabri Mohammed * Dina Harith Shakir*

Abstract

The modern communication systems consider the secure information transmission as an important condition to judge the successful of them. Therefore, the nowadays research focused on the development the cipher algorithms to overcome the cleared drawbacks. Rivest cipher (RC4) algorithm, adopted by wired equivalent piracy (WEP) in IEEE802.11 standard as a cryptographic algorithm, is one of the ciphering methods that suffers from numerous weaknesses. These weaknesses include in the algorithm design itself as well as the problem of attaching the WEP by hackers. As a result, these weaknesses encourage the hackers to attach the transmission information. In this paper, a modified cipher algorithm is presented, which combines the RC4 and developed version of Hill cipher algorithms. The objective of the proposed algorithm is to encrypt the plain textin two levels. Firstly, this text is encrypted utilizing the well-known RC4 and secondly the resulting ciphered text is encrypted using the modified Hill algorithm. The output text is the results of the introduced algorithm of this paper with high security and resiliency against the attaching actions. The proposed algorithm is simulated using C++ environment and the achieved results have been compared to the classic RC4. A superior performance has been recorded for the proposed algorithm in comparison with RC4.

Keywords: *Cryptography, RC4, WEP, Hill Cipher, Stream Cipher*

*University of technology

1. Introduction.

Rivest Cipher (RC4) is a stream cipher method designed in 1987 by Ron Rivestbased on Ron Rivest, Adi Shamir, and Len Adleman, (RSA) Security. The ideology og this method is established on a variable key-size stream cipher with byte-oriented operations. The generation of the utilized keys is based on random permutation process. RC4 is used in the Secure Sockets Layer/Transport Layer Security standards (SSL/TLS), defined for communication between web browsers in client-side and servers in the server-side. It is also used in the Wired Equivalent Privacy (WEP) protocol and the modern(WiFi) Protected Access (WPA) protocol. It is important to note that these protocols are considered by the IEEE 802.11 wireless Local Area Network (LAN) standard. Furthermore, RC4 was kept as a trade secret by RSA Security [1][2].

As mentioned earlier, the RC4 algorithm suffers for different weaknesses, descripted later in this paper. These weaknesses encourage numerous researchers to present research papers aimed toovercome these drawbacks by introducing various modifications on the algorithm itself. The modifications are proposed in order to reproduce a new rigid algorithm with high reliability. On the other hand,there are some weak points on WEP, particularly in the start point, used to find a way for capture the transferred data and crack it to produce the original message. Detailed explanations to the work steps of WEP have been considered to clarifyand tackle the weak points.

This paper proposed a modified ciphering algorithm based on combining the RC4 and modified Hill algorithms. The presented algorithm provides a new encrypting procedure to increase the step complexity, which reflects positively over whole security system.

2. WEP Cryptographic Operations.

Communication security aims to achieve three major objectives, in whichthey should be guaranteed for any protocol that attempts to secure the transmission [3]. These objectives include:

a)*Confidentiality,* whichis the term used to describe the protected data against interception by unauthorized parties.

b) *Integrity*thatmeans that the data has not been modified.

c) *Authentication,* which underpins any security strategy because part of the reliability of data is based on its origin. Users must ensure that data comes from the authorized source. Systems should use authentication to protect data appropriately.

On the other hand, WEP provides operations that attempt to meet the mentioned objectives. Frame body encryption supports confidentiality. In addition, an integrity check sequence protects data in transit and allows receivers to validate that the received data was not altered in transit. WEP also enables stronger shared-key authentication of stations for access points. In practice, WEP falls short in all of these areas. Confidentiality is compromised by flaws in the RC4 cipher; the integrity check was poorly designed; and authentication is of users' MAC addresses, not users themselves[4].

3. WEP Encryption Process and Frame.

Every data frame sent by a station in a WEP protected network is encrypted integrity protected. Figure (1) illustrates the process of encrypting a wireless packet in WEP. When a station sends a packet, the following steps are executed[5][6].

a) The station picks a 24 bit value called initialization vector IV. The IEEE 802.11 standard does not specify how to choose this value. Beside some minor modifications, most vendors implemented one of the following two methods[3][7]:

1) The IV is chosen by a pseudo random number generator PRNG independently from all other packets send by this station.

2) The station always remembers the last IV used. When a new IV needs to be chosen, the station interprets the last IV used as a number and adds 1 to this number. When the highest possible number is reached, the station starts again with 0. On startup, the IV counter either takes a fixed value or a random number is assigned to it.

b). The IV is concatenated to the secret key which is either a 40-bit code for “64-bit encryption” or a 104-bit code for “128-bit encryption” or 128-bit code for" 152-bit encryption" and form the per packet key K = IV|| Sk.

c). A CRC32 checksum of the payload is produced and appended to the payload. This checksum is called Integrity Check Value (ICV).

d)Meanwhile an initialization vector is randomly generated and appended to the “secret key” required for decryption. This resulting stream of data is processed through the RC4 Pseudo Random Generation Algorithm (PRGA) to form a “keystream” of equal length to the plaintext/CRC combination.

e) The plaintext/CRC combination is XOR’ed with the encoding keystream to result in an encrypted message and before transmitting this ciphertext, the initialization vector (IV) is prepended in the clear onto the ciphertext.

When a node receives the encrypted packet, it extracts the unencrypted IV and appends it with the preprogrammed secret key and decrypts the message by XOR’ing this keystream with the encrypted portion of the packet.

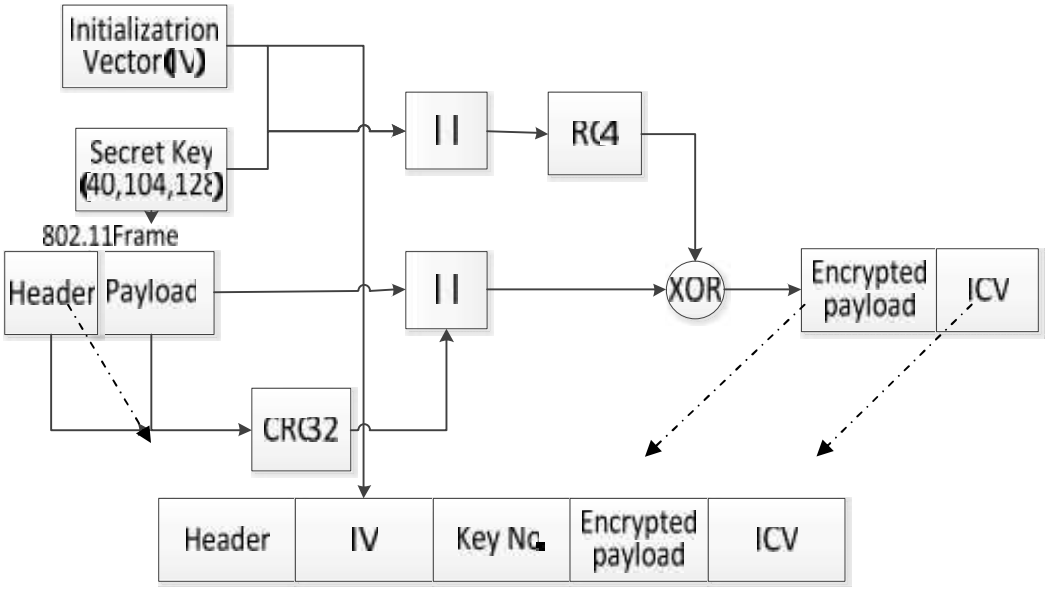

Figure 1. WEP encapsulation block diagram

Figure 2 Shows a simplified version of an 802.11 frame[10,6].

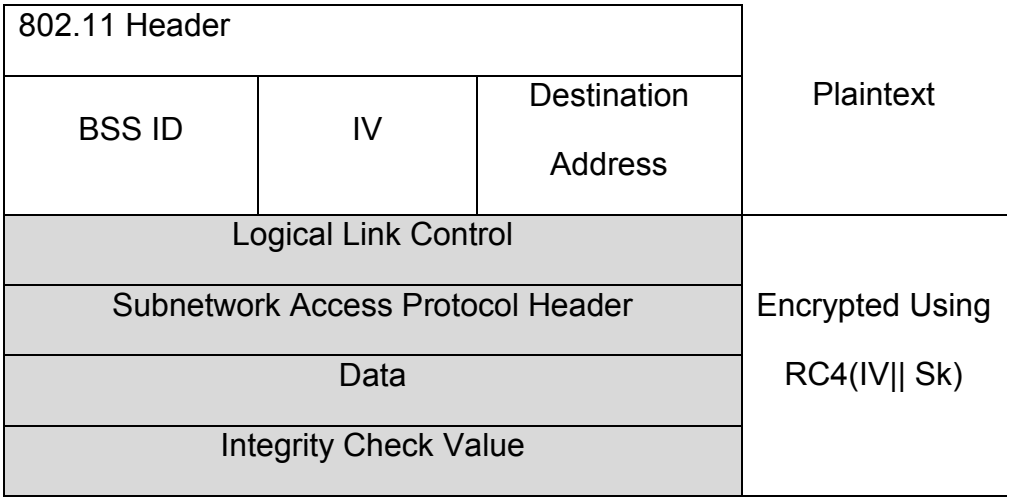

Figure 2. 802.11 Frame Encrypted Using WEP

4. Encapsulation of Higher-Layer Protocols within 802.11.

Like all other 802 link layers, 802.11 can transport any network-layer protocol. Unlike Ethernet, 802.11 relies on 802.2 logical-link control (LLC) encapsulation to carry higher-level protocols. Figure 3 shows how 802.2 LLC encapsulation is used to carry an IP packet. In the figure, the "MAC headers" for 802.11 might be the 12 bytes of source and destination MAC address information on Ethernet or the long 802.11 MAC headers from the previous section [4].

An Ethernet frame is shown in the top line of Figure 3. It has a MAC header composed of source and destination MAC addressesa type code, the embedded packet, and a frame check field. In the IP world, the Type code is either 0x0800 (2048 decimal) for IP itself, or 0x0806 (2054 decimal) for the Address Resolution Protocol (ARP).

802.11 are derivatives of 802.2's *sub-network access protocol* (SNAP). The MAC addresses are copied into the beginning of the encapsulation frame, and then a SNAP header is inserted. SNAP headers begin with a *destination service access point* (DSAP) and a *source service access point* (SSAP). After the addresses,SNAP includes a Control header. Like high-level data link control (HDLC) and its progeny, the Control field is set to 0x03 to denote unnumbered information (UI), a category that maps well to the best-effort delivery of IP datagrams. The last field inserted by SNAP is an organizationally unique identifier (OUI). Initially, the IEEE hoped that the 1-byte service access points would be adequate to handle the number of network protocols, but this proved to be an overly optimistic assessment of the state of the world. As a result, SNAP copies the type code from the original Ethernet frame.[4]

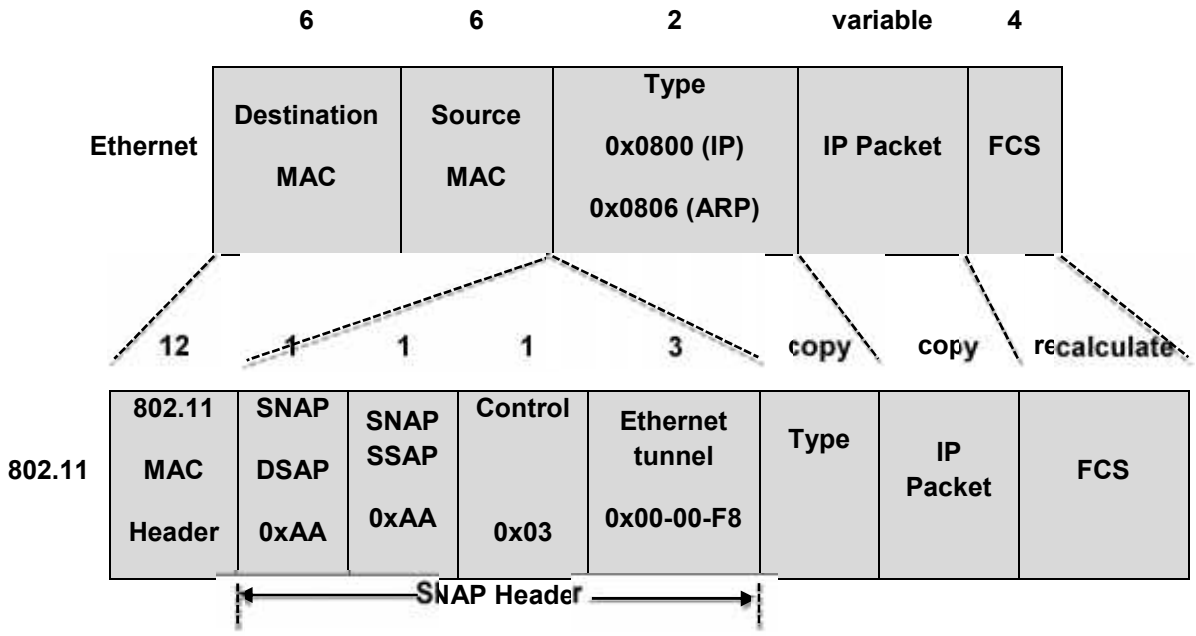

Figure 3. IP encapsulation in 802.11

5.Rivest Cipher (RC4) Algorithm.

The RC4 algorithm is remarkably simple and quite easy to explain. A variable-length key from 1 to 256 bytes (8 to 2048 bits) is used to initialize a 256-byte state vector S, with elements S [0], S [1],..., S [255]. At all times, S contains a permutation of all 8-bit numbers from 0through 255. For encryption and decryption, a byte *k* is generated from S by selecting one of the 255 entries in a systematic fashion. As each value of *k* is generated, the entries in S are once again permuted [1][2]:

Initialization of S:

To begin, the entries of S are set equal to the values from 0 through 255 in ascending order; that is; S[0] = 0, S[1] = 1,..., S[255] = 255. A temporary vector, T, is also created. If the length of the key K is 256 bytes, then K is transferred to T. Otherwise, for a key of length keylen bytes, the first keylen elements of T are copied from K and then K is repeated as many times as necessary to fill out T. These preliminary operations can be summarized as follows:

```
for i = 0 to 255 do
        S[i] = i;
        T[i] = K [i mod keylen];
end
/* Initial Permutation of S */
j = 0;
for i = 0 to 255 do
        j = (j + S[i] + T[i]) mod 256;
Swap (S[i], S[j]);
```

Because the only operation on S is a swap, the only effect is a permutation. S still contains all the numbers from 0 through 255.

Stream Generation:

Once the S vector is initialized, the input key is no longer used. Stream generation involves cycling through all the elements of S[i], and, for each S[i], swapping S[i] with another byte in S according to a scheme dictated by the current configuration of S. After S [255] is reached, the process continues, starting over again at S [0]:

```
/* Stream Generation */
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];
```

To encrypt, XOR the value k with the next byte of plaintext.To decrypt, XOR the value k with the next byte of cipher text.

6. Obtaining sufficient amounts of key stream.

The Internet Protocol (IP) is the most widely deployed network protocol. For this attack to work, assume that version 4 (IPv4) of this protocol is used on the wireless networks for attack.

If host A wants to send an IP datagram to host B, A needs the physical address of host B or the gateway through which B can be reached. To resolve IP addresses of hosts to their physical address, the Address Resolution Protocol (ARP) is used. This works as follows: Host A sends an ARP request to the link layer broadcast address. This request announces that A is looking for the physical address of host B. Host B responds with an ARP reply containing his own physical address to host A. Since the Address Resolution Protocol is a link layer protocol it is typically not restricted by any kind of packet filters or rate limiting rules[8].

ARP requests and ARP replies are of fixed size. Because the size of a packet is not masked by WEP, they can usually be easily distinguished from other traffic.

The first 16 bytes of cleartext of an ARP packet are made up of a 8 byte long 802.11 Logical Link Control (LLC) header followed by the first 8 bytes of the ARP packet itself. The LLC header is fixed for every ARP packet (AA AA 03 00 00 00 08 06). The first 8 bytes of an ARP request are also fixed. Their value is 00 01 08 00 06 04 00 01. For an ARP response, the last byte changes to 02, the rest of the bytes are identical to an ARP request. An ARP request is always sent to the broadcast address, while an ARP response is sent to a unicast address. Because the physical addresses are not encrypted by WEP, it is easy to distinguish between an encrypted ARP request and response[8].

By XORing a captured ARP packet with these fixed patterns, the first 16 bytes of the key stream can be recovered. The corresponding IV is transmitted in clear with the packet.

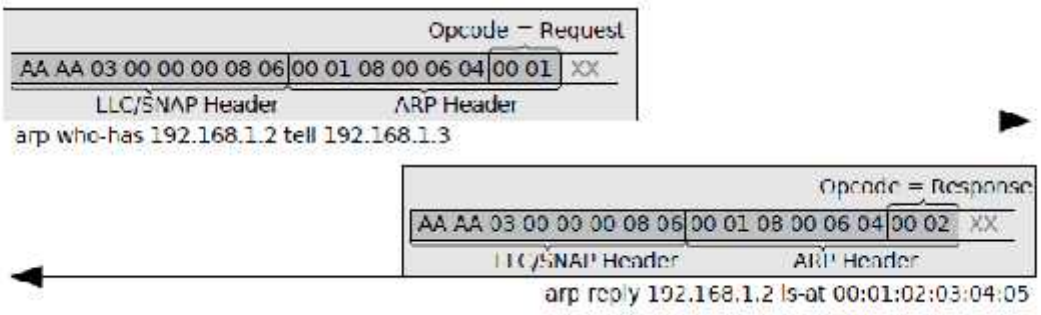

Figure 6. Clear text of ARP Request and Response Packets

7. The (Fluhrer-Mantin-Shamir) FMS Attack.

The Fluhrer, Mantin and Shamir (FMS) attack, published in a 2001 paper titled *"Weaknesses in the Key Scheduling Algorithm of RC4"* takes advantage of a weaknesses in the management of IVs in particular the RC4 key scheduling algorithm to reconstruct the key from a number of collected encrypted messages on WEP encrypted wireless networks [9].

The main problem is that each IV is concatenated with the root key, which is always the same. From some IV values knowing the first byte of keystream. All that is assumed is the ability to recover the first byte of the encrypted payload. Unfortunately, 802.11 uses LLC encapsulation, and the cleartext value of the first byte is known to be 0xAA (the first byte of the SNAP header). Because the first cleartext byte is known, the first byte of the keystream can be easily deduced from a trivial XOR operation with the first encrypted byte, as mentioned before the attacker can knows the first 16 bytes [10][4].

To start, the attacker utilizes the IV as the first 3 elements in K[ ]. He fills the S-box S[ ] with sequential values from 0 to n as RC4 does when initializing the S-box from a known K[ ]. This leads to j3 and S3, the versions of j and S after the 3rd round.For this to hold it is necessary that the values at the positions, S[1], S[S[1]] and S[3] are not swapped around in the remaining process of the key scheduling algorithm. Since the attacker has no knowledge of the rest of the Key K, he cannot know for sure that the relevant positions in S remain unchanged. In the next step j4 will be set to j3 +K[3]+S3[3] and S3[3] and S3[j4] will be swapped. If the attacker could gain knowledge of S4[3], he could recover K[3], since the following holds[9]:

$$K[3] = S_3^{-1}[S_4[3]] - j_3 - S_3[3]$$

At this point, the attacker does not yet have the fourth byte of the key. This algorithm does not regenerate the next byte of the key; it generates a possible value of the key. By collecting multiple messages, for example WEP packets and repeating these steps, the attacker will generate a number of different possible values. The correct value appears significantly more frequently than any other; the attacker can determine the value of the key by recognizing this value and selecting it as the next byte. At this point, he can start the attack over again on the fifth byte of the key[10][9].

8. Hill Cipher Algorithm.

It's interesting multiletter cipher developed by the mathematician Lester Hill in 1929. The encryption algorithm takes (m) successive plaintext letters and substitutes

for them (m)ciphertext letters. The substitution is determined by m linear equations in which each character is assigned a numerical value (a = 0, b = 1 ... z = 25)[1][11]. For m = 3, the system can be described as follows:

For encryption.

$$C_1 = (k_{11}P_1 + k_{12}P_2 + k_{13}P_3) \bmod 26.$$
$$C_2 = (k_{21}P_1 + k_{22}P_2 + k_{23}P_3) \bmod 26.$$
$$C_3 = (k_{31}P_1 + k_{32}P_2 + k_{33}P_3) \bmod 26.$$

This can be expressed in term of column vectors and matrices:

$$\begin{bmatrix} C1 \\ C2 \\ C2 \end{bmatrix} = \begin{bmatrix} K11 & K12 & K13 \\ K21 & K22 & K23 \\ K31 & K32 & K33 \end{bmatrix} \begin{bmatrix} P1 \\ P2 \\ P3 \end{bmatrix} \bmod 26 \quad or \quad C = KP \bmod 26$$

And for decryption

$$\begin{bmatrix} P1 \\ P2 \\ P2 \end{bmatrix} = \begin{bmatrix} K11 & K12 & K13 \\ K21 & K22 & K23 \\ K31 & K32 & K33 \end{bmatrix}^{-1} \begin{bmatrix} C1 \\ C2 \\ C3 \end{bmatrix} \bmod 26 \quad or \quad P = K^{-1} C$$

Where C and P are column vectors of length 3, representing the plaintext and ciphertext, K is a 3 x 3 matrix, representing the encryption key andK^{-1} is a 3 x 3 matrix, representing the decryption inverse key, the inverse K^{-1} of a matrix K is defined by the equation $K\,K^{-1} = K^{-1}\,K = I$, where I is the matrix that is all zeros except for ones along the main diagonal from upper left to lower right. The inverse of a matrix does not always exist, but when it does, it satisfies the preceding equation [1].

It's easy to calculate the matrix inverse by using linear congruence theorem and Extended Greatest Common Divisor algorithm (EGCD), where determinate of K not equal to zero.

Operations are performed modulus26; the proposed Hill cipher algorithm in this paper used the same algorithm, with modulus 256 to achieve the same characters used in RC4 algorithm.

The randomized key generated by RC4 increase the randomization of key used by Hill Cipher; this will increase the Confusion and Diffusion, this is very important point should be satisfied on Hill Cipher.

9. Proposed Algorithm.

In the proposed algorithm, RC4 and Hill Cipher have been combined the following fashion:

Step 1: **Generate k using RC4.**

Step 2: **J = random(0,255).**

Step 3: **C = Hill (CRC4, K).**

The key matrix will be in the form

$$\begin{bmatrix} K_1 & K_2 & K_3 \\ (K_1+J) \bmod 256 & (K_2+J) \bmod 256 & (K_3+J) \bmod 256 \\ (K_1+255-J) \bmod 256 & (K_2+255-J) \bmod 256 & (K_3+255-J) \bmod 256 \end{bmatrix}$$

The new algorithm is more secure, since, it proofed depending on the theory of the Data Encryption Standard (DES) that make it more secure by using (3DES) or by repeating the same algorithm 3 times, so the same technique have been used by mergingRC4 with Hill and dividing the key with the cipher to make a new strong algorithm that take long time to be decrypted than the previous one which is RC4, random generation of key increase the diffusion and confusion in the key selection of Hill Cipher, instead of a normal key selection , the random generation of key support the security of algorithm that make is more secure against the attackers.

The decryption algorithm works in the following fashion:

Step 1: **Generate k using RC4.**

Step 2: **J = random (0,255).**

Step 3: **CRC4 = Hill (C, K^{-1}).**

Step 4:**RC4, K = Plaintext**

10. Results.

The result found in this paper has been classified into two approaches, the time required for the Encryption/Decryption Algorithms and the time required for crack algorithms.

The simulated program done by using Borland C++ program (Version 5.02) on the Personal Computer with specification, Windows 7 32-bits Operating System, Intel core i3 processor and 2 G Byte Main Memory.

Time elapsed for Encryption/Decryption algorithms shown in the table **(1)**

Table 1

File size (Bytes)	RC4 Algorithm (Seconds)	Proposed Algorithm (Seconds)
256	0.42	0.44
1 K	1.12	1.23
10 K	7.55	8.24
20 K	14.29	16.32

Time elapsed for 5 secret key cracked algorithms shown in the table (2**)**

Table 2

Secret key size (Bytes)	RC4 Cracked Time (Seconds)	Proposed Algorithm Cracked Time (Seconds)≅
5	429	11008
13	1115	28620
16	1372	35225

11. Conclusions.

In this paper, a modified cryptography algorithm has been proposed. The proposed algorithm represents a combination of RC4 and modified Hill algorithms. This combination produces an algorithm that can tackle the weaknesses of RC4 method. The presented algorithm is designed to work in WEP technology to increase the security of information transmission from the servers to different clients and in opposite way. The simulation results show that the proposed algorithm outperform the conventional RC4, which provide a positive successful indicator.

References.

1. William Stallings," **Cryptography and Network Security Principles and Practices**", Fourth Edition, Prentice Hall, 2005.

2. Alan G. Konheim," **Computer Security and Cryptography** ", John Wiley & Sons, Inc. 2007.

3. S.M.K.M. Abbas Ahmad, E.G. Rajan and A. Govardhan," **Attack Robustness and Security Enhancement with Improved Wired Equivalent Protocol ",** ACEEE Int. J. on Network Security, Vol. 03, No. 02, 2012.

4. Matthew Gast," **802.11 Wireless Networks: The Definitive Guide** ", O'Reilly Pub Date, 2002.

5. Manuel Mogollon," **Cryptography and Security Services: Mechanisms and Applications",**CyberTech Publishing, USA, 2007

6. Alan Holt and Chi-Yu Huang," **802.11 Wireless Networks Security and Analysis**", Springer-Verlag London Limited 2010.

7. Javvin**," Network Protocols Handbook 4th Edition**," Copyright Javvin Technologies, Inc., 2007.

8. Erik Tews, Ralf-Philipp Weinmann, and Andrei Pyshkin ," **Breaking 104 bit WEP in less than 60 seconds**", TU Darmstadt, FB InformatikHochschulstrasse 10, 64289 Darmstadt, Germany, 2008.

9. MatthieuCaneill and Jean-Loup Gilis**," Attacks against the WiFi protocols WEP and WPA** ", 2010.

10. Wenbo Mao," **Modern Cryptography: Theory and Practice**", Prentice Hall PTR, 2003.

تعديل في خوارزمية تشفير رايفست(RC4)

محمد عبدالله نعمان (باحث) * مي صبري محمد (باحثة) * دينا حارث شاكر (باحثة)*

المستخلص

تعتبر أنظمة الاتصال الحديثة ارسال المعلومات الامينة كشرط في نجاحها . لذا،ركزت البحوث في الوقت الحاضرعلى تطويرالخوارزميات للتغلب على اي عوائق . خوارزمية تشفير رايفست (آرسي 4) ،المتبنية بالقرصنةِ المكافئةِ السلكية (WEP) في معيار (IEEE802.11)كخوارزمية تشفير،إحدى طرق التَشْفيرالتي تعاني العديد من نقاط الضعف. يَتضمّنُ هذه الضعفِ في تصميم الخوارزمية نفسها بالإضافة إلى مشكلةِ الهجوم على (WEP)مِن قِبل لصوص الحاسوب. كنتيجة،تشجع نقاط الضعف هذه لصوصَ الحواسيب لمهاجمة معلوماتِ الإرسالَ. في هذه الورقةِ , قدمت خوارزمية تشفير معدّلة،التي تَدْمجُ(RC4)ونسخة مطورة مِنْ خوارزمية تشفير الهل (Hill Cipher). إنّهدفَ الخوارزميةِ المُقتَرَحةِ أنْيُشفّرَالنصّ العادي في مستويين. الأول، النص الاصلي يشفر باستعمال(RC4) والثاني النَصّ المُشَفَّرالناتج يشفربإستعمال خوارزميةِ تشفير (هل) المعدلة. إنّ نَصَّ نَتائِجُ الخوارزميةِ المُقَدَّمةِ في هذه الورقةِ كانت بامنية عالية ضدّ أعمال المهاجمة . إنّ الخوارزمية المُقتَرَحة مثلت باستخدام بيئة (C++) والنَتائِج المُنجَزة قورنت مع خوارزمية (RC4) الكلاسيكية.َللخوارزميةِ المُقتَرَحةِ اداء متفوّق بالمقارنة مَع (RC4).

الكلمات المفتاحية :التشفير،RC4، WEP،تشفيرهل،التشفيرالتدفقي

* الجامعة التكنولوجية