A Novel Luby-Rackoff Based Cipher in a New Feistel-Network Based LPRKES for Smart Cards

Ehab Mahmoud Mohamed ehab@mobcom.is.kyushu-u.ac.jp
Faculty of Engineering/
Advanced Information Technology Dept/
Wireless Communication Section/Kyushu University
Motooka 744, Nishi-ku, Fukuoka-city 819-0395, Japan
Phone +81-92-802-3573, Fax +81-92-802-3572,

Yassin Mahmoud Yassin Hasan ymyhasan@aun.edu.eg
Faculty of Engineering /Electrical Engineering Dept/
Electronics and Communication Section
Assuit University
Assuit, Egypt.

Hiroshi Furukawa furuhiro@is.kyushu-u.ac.jp
Faculty of Engineering/
Advanced Information Technology Dept/
Wireless Communication Section/Kyushu University
Motooka 744, Nishi-ku, Fukuoka-city 819-0395, Japan
Phone +81-92-802-3573, Fax +81-92-802-3572,

Abstract

The RKES (Remotely Keyed Encryption Schemes) are greatly useful in solving the vital problem of how to do bulk encryption and decryption for high-bandwidth applications (like multimedia and video encryption) in a way that takes advantage of both the superior power of the host and the superior security of the smart card. In this way, we propose a novel length preserving (LP) RKES by using a proposed general view of Feistel-Network (FN) in which we use only two rounds in an efficient way. The proposed LPRKES needs a strong pseudorandom permutation (PRP) as its basic building block, so we introduce a new symmetric-key block cipher, with variable block and key lengths, referred to as NLMSFC (Nonlinear Matrix Structure Based Feistel Cipher), appropriate for hardware and software implementations. NLMSFC is a 3-round Luby-Rackoff construction. In this structure, robust pseudorandom functions (PF) are used to obtain a pseudorandom permutation (PRP). NLMSFC makes use of a novel PR keyed-subfunction in a matrix like structure. Extensive statistical tests are conducted upon NLMSFC and its round function in order to demonstrate their competitive diffusion, confusion and pseudorandomness characteristics. In addition NLMSFC is provably secure. At the end of this paper, we show how we can apply NLMSFC as a strong PRP in the suggested LPKES to be used for cryptographic smart cards.

Keywords: pseudorandom function (PF), pseudorandom permutation (PRP), Luby-Rackoff ciphers, Feistel Network (FN), LPRKES.

1. INTRODUCTION

Smart cards provide an effective tool for portable safe hardware storage of secret keys critically needed in many recent multimedia applications such as real time access control, software license management, e-technology, e-commerce and e-services [1]. Smart cards are mainly reliable because of their distinctive features of tamper-resistant packaging, loose coupling to the host and low cost [2]. However, with their computationally limited resources, smart cards cannot process large data blocks as fast as the host may need.

The Remotely Keyed Encryption Protocol RKEP), first introduced by Blaze, addressed how to do bulk encryption/decryption taking advantage of both the superior computational power, speed and resources of the (high bandwidth) host (trusted with plaintexts/ciphertexts) and the superior security of the slow (low bandwidth) smart-card (trusted with the key) [2]. Although of the interesting approach of Blaze, it suffers from some drawbacks. Its drawbacks basically result from the low security of the protocol. Lucks gave three attacks on the blaze's RKEP, namely a chosen plaintext attack, a two sided attack and a forgery attack (working on the decrypt only smart-card) [3]. In addition, Lucks specified three conditions, that Blaze's RKEP does not satisfy any of them, to make a secure RKE scheme (RKES). Moreover, Lucks suggested the RaMaRK "Random Mapping based RKES" which is based on the Luby-Rackoff construction. Although RaMaRK is based upon Lucks' criteria, a critical weakness was found in RaMaRK [4]. Consequently, Blaze, Feigenbaum and Naor suggested two general RKESs, classified based on the relative length of the ciphertext compared to the plaintext as: a length-preserving (LP) RKES and a length increasing (LI) RKES (with self validation), referred to as BFN-LPRKES and BFN-LIRKES, respectively [4]. To achieve self-validation in the BFN-LIRKES, a signature of the whole ciphertext is appended to the output ciphertext which cannot be computed by an adversary without running the encryption protocol. So any adversary cannot forge the scheme.

In this research, both the fact that, in order to produce a 2n-bit PRP (with entropy of 2n) from n-bit PRF (with entropy of n), it theoretically needs at least two rounds of n-bit PRFs and the fact that the main reason recalling for excess rounds in the Luby-Rackoff construction (and FN ciphers in general) is the rounds joining XOR function, motivated us to construct such a 2-round (only) network excluding the XOR and use a PRP instead. So, in this paper, we develop a new LPRKES employing only a 2-round network based on a general view of an unbalanced Luby-Rackoff construction. The proposed LPRKES is forgery secure, inversion secure and strong pseudorandom. The proposed LPRKES is more secure than the Blaze's RKEP and RaMaRK, more efficient than RaMaRK and the BFN-LPRKES from the card computations and key storage point of views, and requires less number of interactions between the host and the card than the BFN-LPRKES. In addition, the authors proposed an efficient and secure LIRKES [5].

Because of the requirement for a strong PRP in the proposed LPRKES, we introduce NLMSFC: Nonlinear Matrix Structure based Feistel Cipher as variable block-size symmetric-key block cipher. Block cipher is a PRP that maps a block of bits called plaintext into another block called ciphertext using the key bits. Pseudorandomness implies being not distinguishable form truly random permutation (TRP). In a well designed block cipher, a plaintext bit change should change each bit of the output ciphertext with a probability of 0.5. Also, there should be no plaintext/ciphertext-to-ciphertext correlations. Thus, secure block ciphers should essentially exhibit high degree of pseudorandomness, diffusion, and confusion [6]. In addition, a block cipher is most practically qualified as secure if it has survived after being extensively exposed to proficient cryptanalysis. The structure of a block cipher may be a substitution-permutation network (SPN) or Fesitel network (FN). The Advanced Encryption Standard AES-Rijndael is currently the most famous SPN cipher [7]. Alternatively, the FN structure, which is a universal method for converting a round function into a permutation, is adopted in several ciphers such as the DES, DESX, DEAL, FEAL, GOST, Khufu and Khafre, LOKI, CAST, and Blowfish [6], [7]. Rather than the use of many rounds, such as 16 in the DES, Luby and Rackoff introduced a 3-

round FN construction used in designing a provably secure PRP from pseudorandom functions (PRF) [8]. Further analysis and several block ciphers are designed based on the Luby-Rackoff construction [9]–[13]. NLMSFC is a Luby-Rackoff block cipher in which we make use of a new keyed PRF consisting of keyed PR subfunctions in a matrix like structure; the size of this matrix is a data dependent which gives NLMSFC a data dependent structure which significantly strengthens its security. Extensive confusion, diffusion and pseudorandomness tests based on the NIST statistical tests of NLMSFC and its underlying PRF consistently demonstrated their effectiveness. Furthermore, NLMSFC is not practically vulnerable to known attacks. Also it is suitable for both hardware and software implementations.
Although NLMSFC is introduced to be used in the proposed LPRKES, it can be used to strengthen wireless mesh networks clients security by applying it as a candidate with a good pseudorandom and security properties in the well known WPA2 protocol used in IEEE 802.11i standard [14], [15]. In addition, we can exploit the whole scheme (NLMSFC and the LPRKES) to build a smart card based wireless mesh network to enhance its authentication and security in general [16].

The rest of the paper is organized as follows. Section 2 describes the Luby-Rackoff construction in more details, section 3 introduces NLMSFC and its experimental work, section 4 gives the suggested LPRKES with its cryptanalysis, section 5 shows how we can apply NLMSFC in the LPRKES, and section 6 gives the conclusions and future work.

2. PRELIMINARIES

Let "$\oplus$"denote the bit-wise XOR operation and $f_1, f_3 : \{0,1\}^r \to \{0,1\}^l$ and $f_2 : \{0,1\}^l \to \{0,1\}^r$ be a keyed PRFs. Given a k-bit key $K \in \{0,1\}^k$, a plaintext message $P = (L,R) \in \{0,1\}^{l+r}$ is divided into an l-bit (left) block L and r-bit (right) block R. Let $C = (U,T) \in \{0,1\}^{l+r}$ be its corresponding ciphertext. In case of l=r (balanced structure), Luby and Rackoff described how to construct a secure (against known / chosen plaintext attacks) PRP $\psi(f_1, f_2, f_3)(L,R) = (U,T)$ over $\{0,1\}^{l+r}$, from r-bit PRF's using a 3-round balanced Feistel network, rather than the use of 16 rounds as in the DES algorithm[8], with U and T computed as follows Fig.1: $S = L \oplus f_1(K_1, R), T = R \oplus f_2(K_2, S)$ and $U = S \oplus f_3(K_3, T)$ where $S, U \in \{0,1\}^l$ and $T \in \{0,1\}^r$. Likewise, $\psi(f_3, f_2, f_1)$ yields the inverse PRP.
Note that because the entropy of the required permutation is (l+r)-bit, at least two rounds of PRFs are needed. But, using two rounds only, the attacker can distinguish the outputs from truly random permutation, if he simply chooses two different inputs with the same R. Luby and Rackoff even suggested the use of 4 rounds to prevent adaptive hosen plaintext-ciphertext attacks. Also unbalanced Luby-Rackoff construction $l \neq r$ is presented [9].

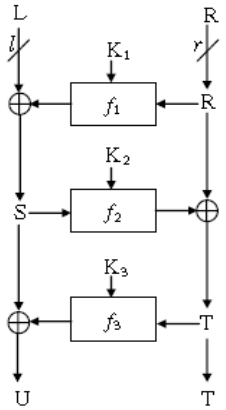

FIGURE 1: Luby-Rackoff cipher construction

3. The Proposed NLMSFC Cipher

As we mentioned, NLMSFC is a balanced 3-round FN cipher (*l*=*r*) like Luby-Rackoff construction. In addition, the (same) nonlinear matrix structure-based pseudorandom function *f* is employed in each of the three rounds as shown in Fig. 1. The motivation of using this matrix structure cipher is its proven highly diffusion, confusion and security properties [11], [17]. The input to the cipher algorithm is an arbitrary length plaintext that is multiple of 64-bit and an arbitrary length user key UserKey. If the input plaintext length isn't multiple of 64-bit padding will take place to get it multiple of 64-bit before the encryption process.

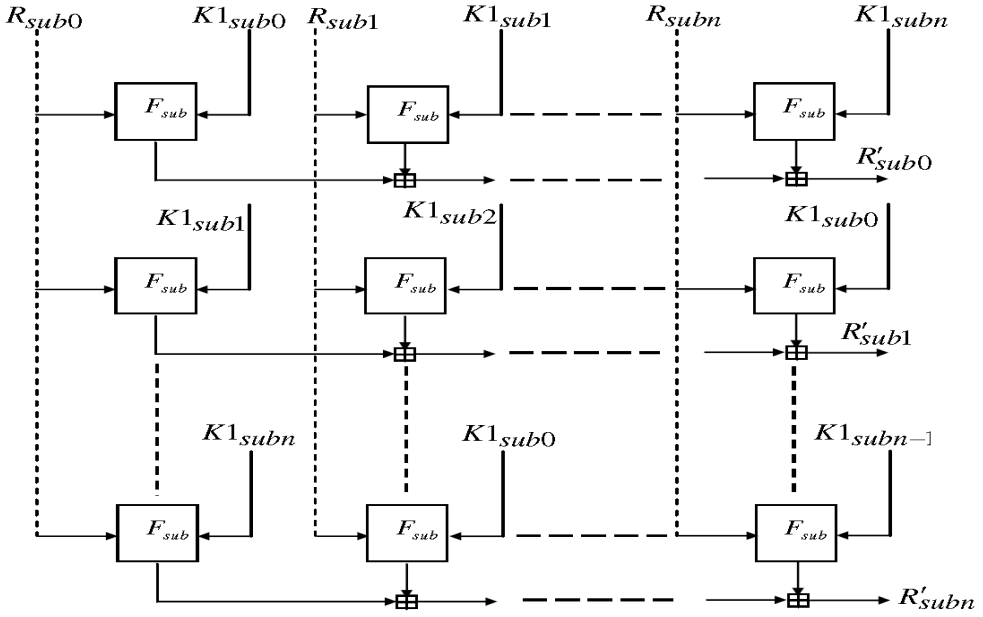

FIGURE 2: The Proposed NLMSFC PR round function F

3.1 The Proposed NLMSFC PR Round Function (F)

The PR round function F uses the PR F_{sub} as its basic building block. The inputs to the round function F are a data block R of length r bits and r-bit round key. First, R is equally divided into n-word [R_{sub0}, R_{sub1}…R_{subn}], each of length 32-bit, also the round key say K_1 is also equally divided into n-word [K_{1sub0}, K_{1sub1}…K_{1subn}]. Then the subfunction F_{sub} will be applied on each these words in a matrix like structure of a data dependent size $n \times n$, as shown in Fig.2, where ⊞ denotes addition mod 2^{32}.

3.2 The Proposed NLMSFC PR Subfunction (Fsub)

F_{sub} is the basic building block in constructing the PR round function F. F_{sub} is an iterated block cipher in which, we perform an invertible keyed operations number of times (rounds) on the input data. F_{sub} is responsible for making the confusion and the diffusion processes required from the PR round function F. Inputs to F_{sub} are a data subblock R_{subi}, $(0 \le i \le n)$ each of length 32-bit and a key subblock K_{subi}, $(0 \le i \le n)$ also each of length 32-bit. F_{sub} performs simple keyed byte operations (addition mod 256, XOR and circular shift left) on the inputs, and it outputs a 32-bit data block.

In designing F_{sub} we take into account that this subfunction must satisfy diffusion and confusion in a minimal number of rounds (4-round).The significant highly nonlinear operation used in F_{sub} is the keyed dependent circular shift used in the 4-round.
Figure 3 shows the PR F_{sub} construction.

The following notations are used in Fig.3.

⊞ Addition mod 2^8

⊕ Bitwise XOR

↵ Circular shift left

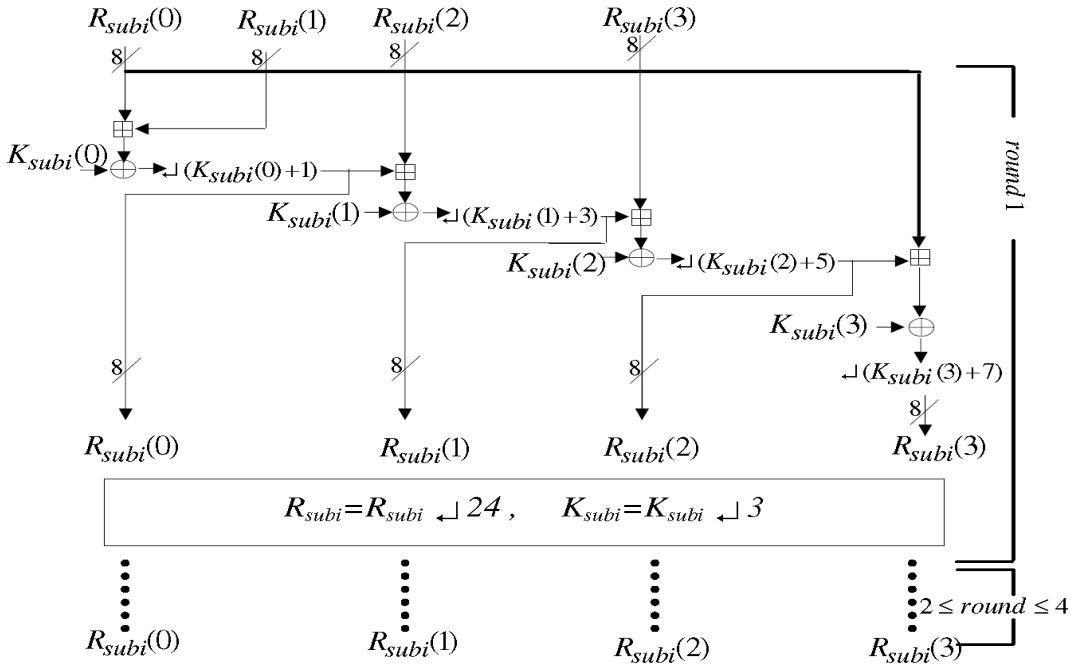

FIGURE 3: The Proposed NLMSFC PR F_{sub}

3.3 The Proposed NLMSFC Key Scheduling Algorithm

The key-scheduling algorithm is used to generate the 3-round keys K_1, K_2 and K_3 each of length r-bit, where the input to the key generation algorithm is the user input key (UserKey), and the output is the UserKey after modifications UserKey =[K_1,K_2,K_3] with length 3r-bit, where 3 indicates that the modified user key UserKey will be equally divided into 3-round keys.
There are 3 cases the key scheduling algorithm handles:
Case $UserKeyLen \geq 3.r$ -bit. In this case the algorithm truncates UserKey to length UserKeyLen = 3r-bit, and then equally divides it into three keys K_1, K_2 and K_3 each of length *r*-bit.
Case $UserKeyLen \geq 64 \& < 3.r$ -bit. In this case the algorithm makes expansion to the input UserKey until UserKeyLen=3r-bit, then equally divides it into three keys K_1, K_2 and K_3 each of length *r*-bit.
Case $UserKeyLen < 64$ -bit. In this case padding with $(64 - UserKeyLen)$ zeros will take place to the right of UserKey, and then the algorithm makes expansion to UserKey until UserKeyLen=3r-bit, and then equally divides it into the 3-round keys K_1, K_2 and K_3.

Expansion process: The following pseudo code shows the expansion process used in the key scheduling algorithm. In this pseudo code we use the following notations:

$\boxplus$ Addition mod 2^{32}
$\oplus$ Bitwise XOR
$\hookleftarrow$ Circular shift left
| Concatenation

```
1-Index=1
2-Indexx=1
3- While UserKeyLen=3r
        UserKey=UserKey ↵n
        If Indexx = odd then
            T=UserKey(Index:index+31) ⊞ UserKey(index+32:index+63)
       else
            T=UserKey(Index: index+31) ⊕ UserKey(index+32: index+63)
        End if
        UserKey=UserKey | T
        Indexx=Indexx+1
        Index=Index+32
   End while
4-truncate UserKey to length UserKeyLen=3r–bit
```

3.4 NLMSFC Cryptanalysis

In this section, we consider the performance of NLMSFC under several attacks types.

1- Exhaustive key search attack (brut search attack):
In this attack, the attacker has many plaintext-ciphertext pairs encrypted under the same key and his job is to search all possible keys to find the key used in the encryption process. But, NLMSFC prevents such type of attacks through using arbitrary key length, so the attacker cannot practically make such search. In addition, and if we assume that the attacker knows the operating NLMSFC block length B, $B \in \{64,128,192,.....\}$ he must search in $2^{\frac{3B}{2}}$ possible keys (without using the key-scheduling algorithm). So we recommend using large operating block lengths to get such attack computationally infeasible.

2- Dictionary attack: In this attack, the attacker makes a look up table (LUT) containing all possible plaintexts/ciphertexts pairs encrypted under all possible keys. In the case of NLMSFC, we allow the user to encrypt the whole message at once or divide it into blocks of sizes that are multiple of 64 bits (64,128,196,256…). Moreover, we allow the user to use any key length. Then, the attacker neither knows the block size nor the key length used. So he finds no way to make such a dictionary.

3- Linear and Differential Cryptanalysis: In linear and differential attacks [6], the attacker wants to know multiple distinct plaintexts-ciphertexts pairs encrypted under the same key, to know some of the key bits. So in order to prevent these attacks, we can encrypt the whole message at once using a different key each time or simply keep the employed NLMSFC running and use its successive output to encrypt the successive input blocks. However, more analysis needs to be done in this field.

4- Adaptive chosen plaintext/ciphertext attack: The 3-round Luby-Rackoff ciphers may not prevent the adaptive chosen plaintext/ciphertext (two-sided) attack, which is the strongest attack against any symmetric key block cipher (despite being of little practical availability where the

attacker can reach both the encryption and decryption engines). So, as suggested by Luby and Rackoff [8], a 4-round NLMSFC successfully prevents such type of attack.

Plaintext	Key	Ciphertext
0000000000000000	0000000000000001	5746958952AD3C9C
0000000000000001	0000000000000000	4B9E731C8A395EB2
0000000000000000	FFFFFFFFFFFFFFFF	AE4E811BB7B07217
FFFFFFFFFFFFFFFF	FFFFFFFFFFFFFFFF	351A6572A06FF9C6
0000000000000001	FFFFFFFFFFFFFFFF	1FB6F2FF51D31232
FFFFFFFFFFFFFFFE	FFFFFFFFFFFFFFFF	BE84D2178229B3FA
FFFFFFFFFFFFFFFF	FFFFFFFFFFFFFFFE	9DA076943DAF1157
FFFFFFFFFFFFFFFF	0000000000000000	E092484DCCB58153

TABLE 1: Examples of 64-bit test vectors (in Hex) for NLMSFC

3.5 NLMSFC Experimental Work

We fully software implemented NLMSFC as a variable block-size variable key-length cipher with a simple effective key scheduling scheme. Table.1 presents examples of plaintext-key-ciphertext NLMSFC test vectors, especially including low and high density and correlated plaintext and key patterns, assuming 64-bit plaintext/key that shows NLMSFC excellent diffusion and confusion properties.

As in all Luby-Rackoff ciphers, security and pseudorandomness of the cipher is based upon the PR of the employed keyed round PRF f_K. The diffusion and confusion properties as well as pseudorandomness of the proposed PRF and the overall NLMSFC have been verified using extensive statistical diffusion and confusion as well as NIST tests [18].

Diffusion Test: 100 64-bit (32-bit for testing the round function) PR plaintexts P_i, i=1,2,.. ..,100 and 100 64-bit key K_i, i=1,2,.. ..,100, are generated using the SEAL algorithm. For each P_i, 64 1-perturbed-bit plaintexts $\{P_{i,j}, j=1,2,.. ..,64\}$, with the *jth* bit inverted, are generated. Then, the histogram, mean value and variance of the 6400 hamming distances $d_{i,j}=\sum(E_{Ki}(P_i) \oplus E_{Ki}(P_{i,j}))$ are computed, where $E_{Ki}(P_i)$ means the encryption of plaintext P_i using the K_i key.

Confusion Test: For the $P_{i,j}$'s mentioned above, the histogram, mean value and variance of the 6400 plaintext-ciphertext correlation coefficients $\rho_{i,j}= \text{corr}(P_{i,j},E_{Ki}(P_{i,j}))$ are computed. Also, for the P_i's and $P_{i,j}$'s the histogram, mean value and variance of the 6400 ciphertext-ciphertext (of correlated plaintexts) correlation coefficients $\rho_{ij} = \text{corr}(E_{Ki(Pi,)},E_{Ki}(P_{i,j}))$ are computed.

The results of the confusion and diffusion tests (summarized in Table.2 and Fig.4, 5 and 6) illustrate competitive performance compared with the DES and IDEA ciphers [6] as the correlations are almost zero and the percentage of the changing bits due to 1-bit perturbations is almost 50%.

NIST Pseudorandomness tests: The NIST Test Suite is a statistical package composed of 16 tests, basically developed to test the randomness of PRNG sequences. To use the NIST tests for testing the pseudorandomness (and implicitly the diffusion and confusion) of a block cipher, 7 data types are generated, following the procedure suggested in [19]. Of each data type, 100 4096-bit binary sequences were analyzed. These data types include: Plaintext-Avalanche, Key-Avalanche, Plaintext-Ciphertext Correlation, Low-Density Plaintext, Low-Density Key, High-Density Plaintext and High-Density Key data types.

The following 13 tests, with 32 p-values, of the 16 NIST tests were applied, namely the frequency (monobit), frequency within a Block (using a 128-bit block length), runs, longest run-of-1's in a block (using a 128-bit block length), binary matrix rank (with a 3×3 size), discrete Fourier transform, overlapping template matching (using a template of 9 1's, with a block length of 512-bit), Maurer's "universal statistical" (with 4-bit per block with 60 blocks for the initialization

sequence), linear complexity (with a 20-bit block length), serial (with a 3-bit block length), approximate entropy (with a 2-bit block length), cumulative sums (Cusums), and random excursions variant tests.

Cipher Alog	Diffusion block length=64	Confusion tests block length=64	
		plain /cipher texts Corr.	Ciphertexts Corr.
	mean/64, var/64	Mean, var	Mean, var
NLMSFC	0.50, 0.24	-4.16e-5, 9.57e-4	-6.25e-4, 9.46e-4
DES	0.50, 0.24	-1.05e-5, 9.46e-4	-2.93e-4, 9.67e-4
IDEA	0.50, 0.25	-4.43e-4, 9.65e-4	-6.17e-4, 9.78e-4

TABLE 2: Comparison between the NLMSFC, DES, and IDEA.

Significance level of 0.01 indicates that one would expect 1 sequence out of 100 sequences to be rejected. A p-value ≥ 0.01 means that the sequence can be considered as random with a confidence of 99%. For each p-value, either success or failure evaluation was made based on being either above or below the pre-specified significance level of α=0.01 [18]. For each 100 sequences, two quantities were determined: the proportion of binary sequences passing the statistical test and an extra uniformity p-value based on a chi $\chi 2$ test (with 9 degree of freedom) applied to the p-values of the 100 sequences. A sample (of 100 sequences) was considered to be passed a statistical test if its proportion of success exceeded

$(1-\alpha)-3\sqrt{\frac{\alpha(1-\alpha)}{m}} = .99-3\sqrt{\frac{0.99\times 0.01}{100}} \approx 0.94$, i.e., 94%, and the uniformity test P-value exceeds 0.0001 [18]. The obtained results of the 32 p-values of the NIST tests successfully verified the pseudorandomness, diffusion and confusion properties of the proposed PRF and the overall NLMSFC with more than 94% proportion of succeeded sequences. Figure.7-9 illustrate samples of the obtained results, specifically the proportion of succeeded sequences for the 32 NIST tests applied to NLMSFC with Plaintext-Avalanche, Key-Avalanche, and Plaintext-Ciphertext Correlation generated data types.

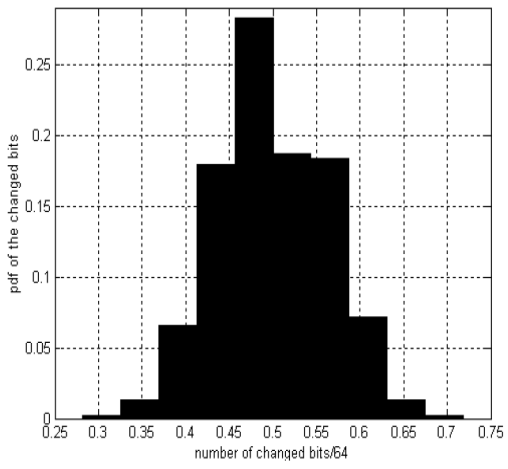

FIGURE 4: Diffusion test: NLMSFC

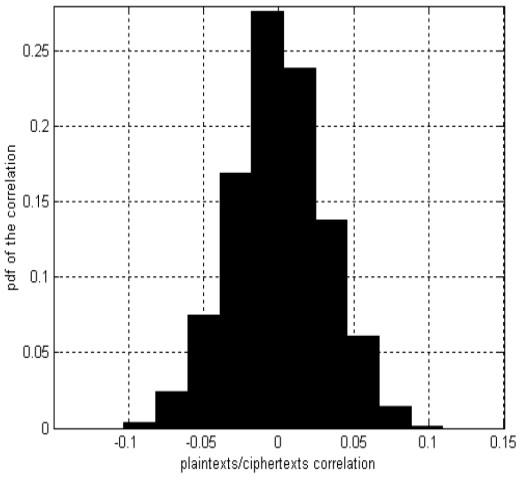

FIGURE 5: Confusion test: NLMSFC plaintext-ciphertexts Correlations histogram

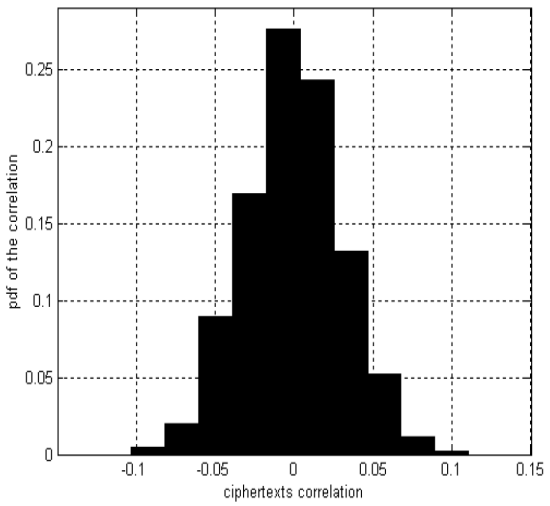

FIGURE 6: Confusion test: NLMSFC ciphertexts Correlations histogram

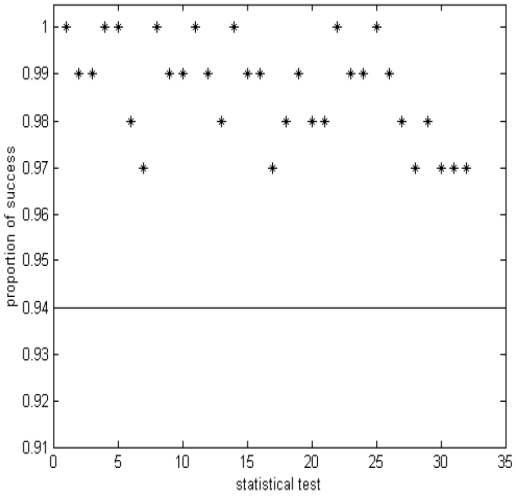

FIGURE 7: NIST tests using Plaintext-Avalanche data: Proportion of succeeded sequences for NLMSFC

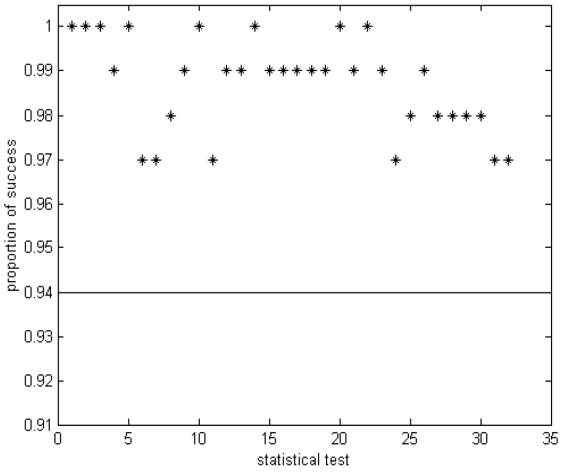

FIGURE 8: NIST tests using Plaintext- Ciphertext correlation: Proportion of succeeded sequences for NLMSFC

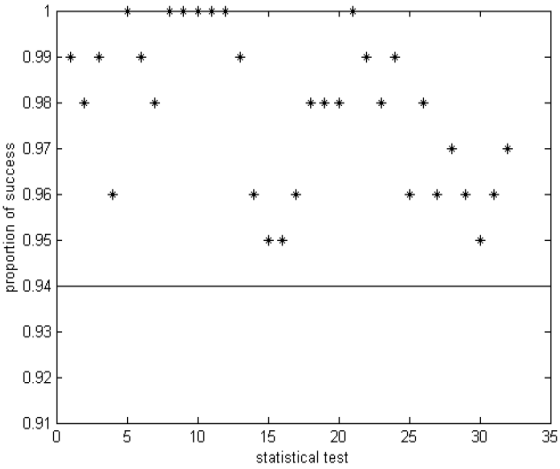

FIGURE 9: NIST tests using key-Avalanche data: Proportion of succeeded sequences for NLMSFC

4. A Novel LPRKES Based upon 2-Round Generalized FN for Smart Cards

4.1 Proposed Generalized 2-Round FN

Since, most of attacks on Luby-Rackoff and multiple rounds FN (e.g., DES) are based upon the linearity properties of the XOR function joining the rounds, we suggest the use of a keyed invertible encryption function $E_K(.):\{0,1\}^k \times \{0,1\}^l \rightarrow \{0,1\}^l$ instead of the XOR to propose a generalized 2-round FN which will be used as a new LPRKES. The E_K/D_K (encryption/decryption) function used in this scheme Fig. 10 should be a strong PRP functions like DES, AES, or the proposed NLMSFC.

In this network, the plaintext P and its ciphertext C is divided into m blocks, i.e. $P=(P_1, P_2,\ldots,P_m)$ and $C=(C_1, C_2,\ldots C_m)$, $L=P_1$, $R=(P_2,\ldots,P_m)$, $U=C_1$, $T=(C_2,\ldots,C_m)$ and *E* denotes the PRP encryption function. $H:\{0,1\}^* \rightarrow \{0,1\}^a$ denotes a collision resistant one way hash function, such

as SHA-1 [6], and $F:\{0,1\}^k \times \{0,1\}^a \rightarrow \{0,1\}^k$ is a keyed mapping function (ex , simply XOR). Also, the second round keyed hash function is simply interleaving or concatenating the input with the key, i.e, *H(U|K2), H(K2|U)* or *H(K2|U|K2)*.

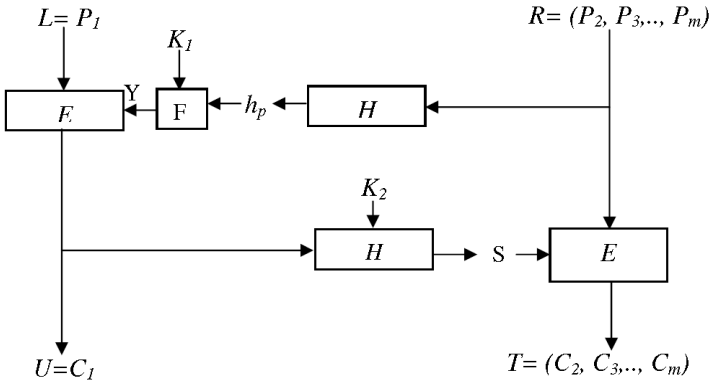

FIGURE 10: The proposed generalized 2-round FN

4.2 A Novel LPRKES Based upon the Proposed Generalized FN

We make use of the above 2-round FN in making a strong and highly secure LPRKES with a one interaction between the host and the smart card which is greatly important from the security point of view.

Proposed LPRKES Encryption Protocol:
Input P= (P_1, P_2… P_m) and output C=(C_1, C_2….C_m).

1. Host: $h_p \leftarrow H(P_2, P_3 ... P_m)$.
2. Host $\rightarrow$ Card: P_1, h_p.
3. Card: $C_1 \leftarrow E_{F(h_p, K_1)}(P_1)$.
4. Card: $S \leftarrow H(C_1, K_2)$.
5. Card $\rightarrow$ Host: C_1, S.
6. Host: $C_i \leftarrow E_S^i(P_2, P_3 .... P_m)$, $i \in \{2,3,....m\}$.

Proposed LPRKES Decryption protocol:
Input C=(C_1, C_2….C_m) and output P=(P_1, P_2… P_m).

1. Host $\rightarrow$ Card: C_1.
2. Card: $S \leftarrow H(C_1 | K_2)$.
3. Card $\rightarrow$ Host: S.
4. Host: $P_i \leftarrow D_S^i(C_2, C_3 ... C_m), i \in \{2,3....m\}$.
5. Host: $h_p \leftarrow H(P_2, P_3 ... P_m)$.
6. Host $\rightarrow$ Card : h_p.
7. Card: $P_1 \leftarrow D_{F(h_p, K_1)}(C_1)$.
8. Card $\rightarrow$ Host: P_1.

4.3 Security Analysis of the proposed LPRKES

We first prove that the proposed LPRKES satisfies LUCKs' postulates [3]:

Theorem 1: the proposed LPRKEs is forgery secure with a probability of $\frac{q^2}{2^l+1}+\varepsilon$, where forgery secure means that if an attacker can execute q encryptions/decryptions with arbitrarily plaintexts/ciphertexts, he can know no more than q valid plaintexts-ciphertexts pairs.

Proof: consider the following two cases with messages M_1= {L_1, R_1} and M_2 = {L_2, R_2}:

Case1: Consider the encryption protocol Fig.10 and assume that $R_1 = R_2$, $L_1 \neq L_2$ and let $E:\{0,1\}^k \times \{0,1\}^l \rightarrow \{0,1\}^l$ be a strong invertible pseudorandom permutation PRP (ex. DES, NLMSFC, AES). Then, the probability Pr (U_1=U_2 for $L_1 \neq L_2$) =0 or almost zero. Consequently, with $U_1 \neq U_2$ and the collision resistance hash function H (i.e., it is infeasible to find $t_1 \neq t_2$ with $H(t_1) = H(t_2)$) [3]. So, Pr ($S_1 = S_2$) will be negligible. Thus, in this case, with a probability near one, the ciphertexts $C_1=(U_1,T_1)$ and $C_2=(U_2,T_2)$ are independently chosen random values. So the attacker has no gain when encrypting many plaintexts with equal right halves. The same analysis will be applied with $L_1 = L_2$, $R_1 \neq R_2$, and for the decryption protocol.

Case 2: Let $L_1 \neq L_2$, $R_1 \neq R_2$. Consequently, $h_{p1} \neq h_{p2}$ and $Y_1 \neq Y_2$ (Fig.10), Also E is a strong PRP which means: Pr (U_1 ($E_{Y1}(L_1)$ = U_2 ($E_{Y2}(L_2)$) ≤ $\frac{1}{2^l}$ +ε where ε is a small number depending on the pseudorandomness of E (if E is truly random then ε = 0). In consequence Pr ($S_1 = S_2$) ≤ $\frac{1}{2^l}$ +ε.

If the attacker makes q encryptions, then there are q(q-1)/2 different messages pairs. Thus, the probability of $U_i = U_j$, i ≠ j satisfies {Pr ($U_i = U_j$)} ≤ $\frac{q(q-1)/2}{2^l}$ +ε ≈ $\frac{q^2}{2^{l+1}}$ +ε . The same discussion applies for the decryption protocol. Then, we can conclude that, by observing the encryption/decryption process for q plaintexts/ciphertexts, any attacker can distinguish the encryption/decryption permutation from a truly random permutation with a probability not more than $\frac{q(q-1)/2}{2^l}$ + ε.

Theorem 2: The proposed LPRKES is inversion secure.

Proof: inversion secure means the RKES must prevent chosen plaintext/ciphertext attacks. Such attacks can be done on the proposed scheme with a high probability only if the attacker can simulate the card's part of the RKES. From Theorem 1, the attacker can do this if he is able to encrypt/decrypt about $2^{l/2}$ different plaintexts/ciphertexts using the smart card which is impossible for large *l*. So the proposed LPRKES is inversion secure.

Theorem 3: The proposed LPRKES is pseudorandom.

Proof: The proof is included in proving Theorem 1.

Thus, based on the above analysis and compared to recent RKESs [2-4], the proposed LPRKES has the following advantages:

1. The proposed LPRKES is more secure than Blaze's RKEP [2] because it's shown in [3] that Blaze's RKEP is forgery insecure, inversion insecure, and non-pseudorandom.
2. The proposed LPRKES is more efficient than RaMaRK, because, in RaMaRK, Lucks [3] uses the first two plaintexts blocks in order to define an encryption key for the rest of the message. So any adversary that controls the host during the encryption or decryption of one file of a set of files that start with the same two blocks can subsequently decrypt the encryption of any file in the set. In contrast, the proposed scheme uses the rest of the message to define the key used to encrypt the first plaintext block, and then uses the encryption output of the first block to define the encryption key for the rest of the message. So, the keys used to encrypt two messages will be equal only if the two messages are equal (or after the attacker makes nearly $2^{l/2}$ different encryptions of $2^{l/2}$ different messages).

3. The proposed scheme is more computationally efficient than RaMaRK and BFNLPRKES [4] from the card point of view. In RaMaRK, it is required from the card to evaluate six different PRFs. So it is inadequate for inexpensive smart-cards with limited bandwidth, memory, and processor speed. This also happens in the BFN-LPRKES, in which, it's required from the card to evaluate three encryption functions and three mapping functions. However, the proposed scheme needs from the smart card to evaluate only three functions: encryption, hash and mapping (hash) functions.
4. The proposed scheme is more efficient than the BFN-LPRKES from the host-card communication point of view. The BFN-LPRKES requires two rounds of interaction between the host and the card, but the proposed scheme requires only one round which enhances the security of the scheme.

5. The Application of NLMSFC in the Proposed LPRKES.

Figure.11 shows how we can apply NLMSFC as the PRP in the suggested LPRKES.

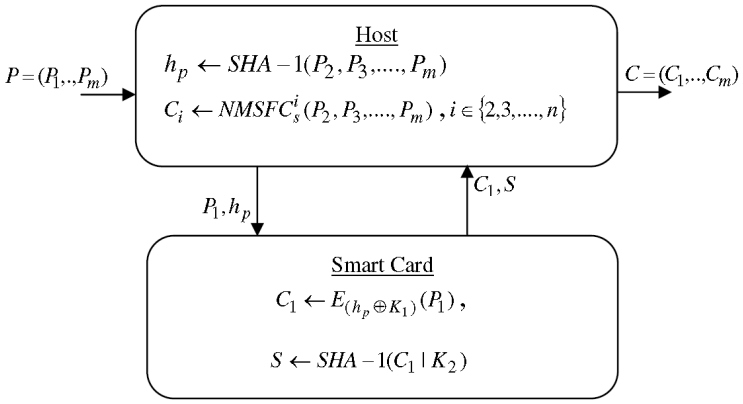

FIGURE 11: The proposed LPRKS using NLMSFC

6. CONSLUSION & FUTURE WORK

This paper deals with cryptographic smart cards protocols which are used to organize the bulk encryption process between the host and the smart card. In an attempt to solve this important issue, we introduce a 2-round network structure, based on a general view of an unbalanced reduced form FN. By exploiting this scheme, we develop smart-card based LPRKES. In addition we analyze this scheme from security and smart card efficiency point of views.
Because the suggested LPRKES is highly depending upon a strong PRP, we also present NLMSFC: A novel Luby-Rackoff construction-based variable block and key lengths symmetric-key block cipher. Its core function is a new pseudorandom function that consists of nonlinear matrix structure with a sub PR function as its elements. Extensive simulations, diffusion, confusion, and NIST pseudorandomess test proof that
NLMSFC and its round function are good PRP and PR function respectively. However, NLMSFC needs a complexity analysis beside the security analysis. But we believe that NLMSFC is less complex.
Also, we show how NLMSFC can be applied as a PRP in the suggested LPRKES. For future development, we will try to apply our cipher and LPRKES in enhancing the security and authentication of the wireless mesh networks especially the wireless backhaul system.

7. REFERENCES

[1] S. Yuan and J. Liu, "Proceedings of the IEEE international conference on e-tech, e-commerce and e-services," pp.91–110, 2004.

[2] M. Blaze, "High-bandwidth encryption with low-bandwidth smartcards," Lecture Notes in Computer Science, vol.1039, pp.33–40, 1996.

[3] S. Lucks, "On the security of remotely keyed encryption," Proceedings of the Fast Software Encryption Workshop, pp.219–229, Springer, 1997.

[4] M. Blaze, J. Feigenbaum, and M. Naor, "A formal treatment of remotely keyed encryption," Lecture Notes in Computer Science, vol.1403, pp.251–265, 1998.

[5] Y. Hasan, "Key-Joined Block Ciphers with Input-Output Pseudorandom Shuffling Applied to Remotely Keyed Authenticated Encryption," IEEE International Symposium on Signal Processing and Information Technology, pp.74–79, 2007.

[6] A. Menezes, P. Van Oorschot, and S. Vanstone, Handbook of Applied Cryptography, CRC press, 2001.

[7] A. Biryukov, "Block ciphers and stream ciphers: The state of the art," Lecture Notes in Computer Science, Proc. COSIC Summer Course, 2003.

[8] M. Luby and C. Rackoff, "How to construct pseudorandom permutations from pseudorandom functions," SIAM Journal on Computing, vol.17, no.2, pp.373–386, 1988.

[9] M. Naor, "On the Construction of Pseudorandom Permutations: LubyRackoff Revisited," Journal of Cryptology, vol.12, no.1, pp.29–66, 1999.

[10] R. Anderson and E. Biham, "Two practical and provably secure block ciphers: BEAR and LION," Lecture Notes in Computer Science, pp.113–120, 1996.

[11] Y. Hasan and E. Mohammed, "PATFC: novel pseudorandom affine transformation-Based Feistel-network cipher," Signal Processing and Information Technology, 2005. Proceedings of the Fifth IEEE International Symposium on, pp.811–816, 2005.

[12] P. Morin, "A critique of BEAR and LION," Manuscript, citeseer. nj. nec. Com/124166. html.

[13] Y. Hasan, "YC: A Luby-Rackoff ciphers family driven by pseudorandom vector/matrix transformations," Signal Processing and Its Applications, 2007. ISSPA 2007. 9th International Symposium on, pp.1–4, 2007.

[14] S. Frankel, B. Eydt, L. Owens, and K. Kent, "Guide to ieee 802.11 i: Establishing robust security networks," Technical Report 800-97, National Institute of Standards and Technology Administration US Department of Commerce, Computer Security Division Information Technology Laboratory National Institute of Standards and Technology Gaithersburg, MD 20899-8930, 2006.

[15] F. Martignon, S. Paris, and A. Capone, "MobiSEC: a novel security architecture for wireless mesh networks," Proceedings of the 4th ACM symposium on QoS and security for wireless and mobile networks, pp.35–42, ACM New York, NY, USA, 2008.

[16] M. Siddiqui and C. Hong, "Security issues in wireless mesh networks," IEEE intl. conf. on multimedia and ubiquitous engineering, 2007.

[17] Y. Hasan, "From stream to provably secure block ciphers based on pseudorandom matrix transformations," Communication Systems Software and Middleware and Workshops, 2008. COMSWARE 2008. 3rd International Conference on, pp.260–265, 2008.

[18] A. Rukhin, J. Soto, J. Nechvatal, M. Smid, and E. Barker, "A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications," , 2001.

[19] J. Soto and L. Bassham, "Randomness Testing of the Advanced Encryption Standard Finalist Candidates. National Institute of Standards and Technology (NIST)," Computer Security Division, 2000.