Cryptanalysis of a Multistage Encryption System¶

Chengqing Li*, Xinxiao Li†, Shujun Li‡§ and Guanrong Chen‡
*Department of Mathematics, Zhejiang University, Hangzhou, Zhejiang 310027, China
†Software Engineering Center, Toshiba, Japan
‡ Department of Electronic Engineering, City University of Hong Kong, Kowloon, HKSAR, China
§The corresponding author, personal web site: http://www.hooklee.com.

***Abstract*— This paper analyzes the security of a new multistage encryption system (MES) recently proposed in ISCAS'2004. It is found that MES is insecure against a differential chosen-plaintext/ciphertext attack. Experiments are given to support the proposed attack. It is also pointed out that the security of MES against brute-force attacks is not sufficiently high.**

I. Introduction

Since the 1990s, the use of chaotic systems in cryptography attracts more and more attention as a new source for designing secure communication systems and encryption schemes [1]. In recent years, Yen et al. proposed a series of encryption schemes based on the 1-D chaotic Logistic map, but some of them have been successfully broken [2, Sec. 4.4.3].

In [3], a multistage encryption system (MES) was proposed by Yen et al. as a new solution to provide a higher security level than their previous schemes. The present paper focuses on cryptanalysis of this new encryption scheme. MES is designed by combining the basic encryption techniques used in previous schemes, such as BRIE [4], RSES/RCES [5], [6] and TDCEA [7], [8], which have been cryptanalyzed in [9]–[11], respectively. Although the combination makes MES securer against the known-plaintext attacks, which can break previous schemes, this paper points out that MES is still insecure against a differential chosen-plaintext/ciphertext attack. Only three chosen plaintexts/ciphertexts are enough to construct some specific differentials to totally break MES. It is also noticed that the security of MES against brute-force attacks is not sufficiently strong.

The rest of this paper is organized as follows. Section II gives a brief introduction to MES. The proposed differential attack is discussed in detail in Sec. III, where some experimental results are given to support the theoretical analysis. A brief analysis on the security against brute-force attacks is then given in Sec. IV. The last section concludes the paper.

II. The Multistage Encryption System (MES)

MES encrypts the plaintext block by block, where each block contains 7 plain-bytes. Each 7-byte plain-block is firstly expanded to an 8-byte block by adding a secret pseudo-random byte, and then is encrypted by three different operations: byte permutation, value masking, and bit recirculation, which are all controlled by a secret pseudo-random bit sequence (PRBS) generated from the chaotic Logistic map [12]: $x(k+1) = \mu \cdot x(k) \cdot (1 - x(k))$.

To facilitate the description of MES, without loss of generality, assume that the plaintext is $f = \{f(i)\}_{i=0}^{N-1}$, where $f(i)$ denotes the i-th plain-byte and N can be exactly divided by 7. In this case, the plaintext has $N/7$ blocks: $f = \{f^{(7)}(k)\}_{k=0}^{N/7-1}$, where $f^{(7)}(k) = \{f^{(7)}(k,j)\}_{j=0}^{6} = \{f(7k+j)\}_{j=0}^{6}$. Similarly, assume that the ciphertext is $f' = \{f'(i)\}_{i=0}^{N-1} = \{f'^{(8)}(k)\}_{k=0}^{N/7-1}$, where $f'^{(8)}(k) = \{f'^{(8)}(k,j)\}_{j=0}^{7} = \{f'(8k+j)\}_{j=0}^{7}$ denotes the expanded cipher-block with 8 bytes. With the above notations, MES can be described as follows.

1) The secret key: three integers α, β, $Open$, the control parameter μ and the initial condition $x(0)$ of the chaotic Logistic map, where $\alpha > 0$, $\beta > 0$, $\alpha + \beta < 8$ and $Open \in \{0, \cdots, 255\}$.

2) The initialization procedure: a) in 33-bit fixed-point finite precision, run the Logistic map from $x(0)$ to generate a chaotic sequence, $\{x(k)\}_{k=0}^{N/7-1}$, and then extract the 33 bits of $x(k) = 0.b_{33k+0} \cdots b_{33k+32}$ to yield a chaotic PRBS, $\{b(i)\}_{i=0}^{33N/7-1}$; b) set $temp = Open$.

3) The encryption procedure of each plain-block $f^{(7)}(k)$ is composed of the following four steps:

a) Data expansion: get an 8-byte block, $f^{(8)}(k) = \{f^{(8)}(k,j)\}_{j=0}^{7} = \{temp, f^{(7)}(k,0), \cdots, f^{(7)}(k,6)\}$, and then set $temp = f^{(8)}(k, l(k))$, where $l(k) = \sum_{i=0}^{2} b(33k+i) \cdot 2^i$.

b) Byte permutation: do the random swapping operation, $Swap_{b(33k+l)}\left(f^{(8)}(k,i), f^{(8)}(k,j)\right)$, for 12 times with the following parameters in order: $(i,j,l) = (0,4,3)$, $(1,5,4)$, $(2,6,5)$, $(3,7,6)$, $(0,2,7)$, $(1,3,8)$, $(4,6,9)$, $(5,7,10)$, $(0,1,11)$, $(2,3,12)$, $(4,5,13)$, $(6,7,14)$, where $Swap_w(a,b)$ outputs (b,a) when $w = 1$ and (a,b) when $w = 0$. Denote the permuted 8-byte block by $f^{*(8)}(k)$.

c) Random masking: determine two pseudo-random bytes, $Seed1(k) = \sum_{i=0}^{7} b(33k+i) \cdot 2^{7-i}$ and $Seed2(k) = \sum_{i=0}^{7} b(33k+8+i) \cdot 2^{7-i}$, and then do the following masking operations for $j = 0 \sim 7$:

$$f^{**(8)}(k,j) = f^{*(8)}(k,j) \oplus Seed(k,j), \tag{1}$$

¶This paper has been published in *Proceedings of the 2005 IEEE International Symposium on Circuits and Systems* (ISCAS 2005, May, 2005, Kobe, Japan), pp. 880-883, 2005.

where $\oplus$ denotes the bitwise XOR operation,

$$Seed(k,j) = \begin{cases} Seed1(k), & B(k,j) = 3, \\ \overline{Seed1(k)}, & B(k,j) = 2, \\ Seed2(k), & B(k,j) = 1, \\ \overline{Seed2(k)}, & B(k,j) = 0, \end{cases} \quad (2)$$

and $B(k,j) = 2 \cdot b(33k+16+j) + b(33k+17+j)$.

d) Bit recirculation: for $j = 0 \sim 7$, do

$$f'(8k+j) = f'^{(8)}(k,j) = ROLR_{p(k,j)}^{q(k,j)}\left(f^{**(8)}(k,j)\right), \quad (3)$$

where $p(k,j) = b(33k+24+j)$, $q(k,j) = \alpha + \beta \cdot b(33k+25+j)$, and $ROLR_p^q$ denotes the q-bit cyclical shift operation whose direction is controlled by p as follows:

$$ROLR_p^q(a) = \begin{cases} a \ggg q, & p = 0, \\ a \lll q, & p = 1, \end{cases} \quad (4)$$

where "$\lll$" denotes the cyclical left-shift operation and "$\ggg$" denotes the cyclical right-shift operation.

4) The decryption procedure is the simple inverse of the above encryption procedure: for the k-th 8-byte cipher-block $f'^{(8)}(k)$, Step *d)* is first performed by replacing $p(k,j)$ with its complement $\overline{p(k,j)} = 1 - p(k,j)$, then Step *c)* is performed, and then Step *b)* is performed in the reversed order, finally the first byte is discarded to recover the plain-block $f^{(7)}(k)$.

III. The Proposed Differential Attack

A. Three properties of MES

Define the XOR-differential ("differential" in short) of two signals f_0 and f_1 as $f_{0\oplus1} = f_0 \oplus f_1$. Then, it is easy to prove the following three properties of MES, which will be the basis of the proposed attack.

Property 1: The random masking in Step *c)* cannot change the differential value, i.e., $\forall\, k, j,\ f_{0\oplus1}^{**(8)}(k,j) \equiv f_{0\oplus1}^{*(8)}(k,j)$.

Proof: From Eq. (1), $f_{0\oplus1}^{**(8)}(k,j) = f_0^{**(8)}(k,j) \oplus f_1^{**(8)}(k,j) = (f_0^{*(8)}(k,j) \oplus Seed(k,j)) \oplus (f_1^{*(8)}(k,j) \oplus Seed(k,j)) = f_0^{*(8)}(k,j) \oplus f_1^{*(8)}(k,j) = f_{0\oplus1}^{*(8)}(k,j)$. ■

Property 2: If the plaintext and the chaotic bit sequence are fixed, all differential bytes in $f_{0\oplus1}^{(8)}(k)$ are fixed, i.e., $f_{0\oplus1}^{(8)}(k)$ are independent of the value of $Open$.

Proof: For the first byte of each 8-byte block, if $temp = Open$, $f_{0\oplus1}^{(8)}(k,0) = 0$; otherwise, $temp$ is one plain-byte occurring before $f^{(7)}(k)$, which means $f_{0\oplus1}^{(8)}(k,0)$ is one of the differential bytes occurring before $f_{0\oplus1}^{(7)}(k)$. Apparently, the differential value $f_{0\oplus1}^{(8)}(k,0)$ is independent of the value of $Open$, but uniquely determined by the plaintext and the secret chaotic sequence. Since the other 7 bytes in $f_{0\oplus1}^{(8)}(k)$ are also independent of $Open$, this property is thus proved. ■

Properties 1 and 2 mean that MES is reduced to be a three-stage cipher with $Open = 0$ (thus becomes a modification of TDCEA [7], [8]), from the differential point of view.

Property 3: The byte permutation in Step *b)* cannot change each differential value, but its position in the 8-byte block.

Proof: This property is obviously true since the byte permutation only change the position of each byte. ■

A natural result of the above property is: if $f_{0\oplus1}^{(8)}(k,0) = \cdots = f_{0\oplus1}^{(8)}(k,7)$, then it is true that $f_{0\oplus1}^{(8)}(k) = f_{0\oplus1}^{**(8)}(k)$. This means that MES is further reduced to be a two-stage cipher (and be a data-expansion modification of BRIE [4]), for differential blocks with 8 identical bytes.

B. The differential attack

Utilizing Property 3 and the cryptanalysis on BRIE given in [9], one can easily break the secret bit recirculations in Step *d)*. Then, the secret byte permutations in Step *b)* and the secret data expansion in Step *a)* can be further broken by using Properties 1 and 2. Finally, the secret masking operations in Step *c)* will be recovered immediately. After all secret operations in the four steps are revealed, most secret chaotic bits can be broken to derive the secret key with a sufficiently small complexity, which leads to the complete breaking of MES. Note that the proposed differential attack can be carried out by choosing either plaintexts or ciphertexts.

1) Breaking the secret $ROLR$ operation in Step d): Choose two plaintexts to obtain the following differential signal $f_{0\oplus1}$: $\forall\ i = 0 \sim N-1,\ f_{0\oplus1}(i) \equiv a$. From the generation rule of $f^{(8)}(k,0)$, there exists a threshold integer, $k_0 \geq 1$, such that $f_{0\oplus1}^{(8)}(k,0) \equiv 0$ when $k \leq k_0$ and $f_{0\oplus1}^{(8)}(k,0) \equiv a$ when $k > k_0$. Assuming that $a \neq 0$ and each chaotic bit distribute uniformly over $\{0,1\}$, one can deduce that $Prob[k_0 = n] = Prob[l(0) = \cdots = l(n-1) = 0 \text{ and } l(n) \neq 0] = 7/8^{n+1}$. This means that $f_{0\oplus1}^{(8)}(k,0) \equiv a$ is almost true when k is sufficiently large. In this case, $f_{0\oplus1}^{(8)}(k,0) = \cdots = f_{0\oplus1}^{(8)}(k,7)$ is true, so from Property 3 one can see that only Step *d)* is left for MES, i.e., $\forall\ j = 0 \sim 7,\ f_{0\oplus1}'^{(8)}(k,j) = ROLR_{p(k,j)}^{q(k,j)}\left(f_{0\oplus1}^{(8)}(k,j)\right) = ROLR_{p(k,j)}^{q(k,j)}(a)$. Now, MES is reduced to be BRIE, and the secret $ROLR$ operations can be broken by setting $a = 1$ as discussed in [9]:

$$ROLR_{p(k,j)}^{q(k,j)} = ROLR_0^{8-\hat{q}(k,j)} = ROLR_1^{\hat{q}(k,j)}, \quad (5)$$

where $\hat{q}(k,j) = \log_2\left(f_{0\oplus1}'^{(8)}(k,j)\right)$, which is the new position of the only 1-bit of $a = 1$ after the $ROLR$ operation.

2) Breaking the secret byte permutation in Step b): Since the secret $ROLR$ operations in Step *d)* has been recovered, from the differential point of view, MES becomes a permutation-only cipher with data expansion. As we analyzed in [13], all permutation-only ciphers are not secure enough against chosen-plaintext attacks. If two plaintexts are chosen to ensure that any two elements in each 8-byte differential block are different, one can uniquely determine the secret permutations by comparing $f_{0\oplus1}^{(8)}(k,1) \sim f_{0\oplus1}^{(8)}(k,7)$ and $f_{0\oplus1}^{*(8)}(k,0) \sim f_{0\oplus1}^{*(8)}(k,7)$. It is easy to do so in chosen-ciphertext attacks, by choosing 8 different cipher-bytes for each $f_{0\oplus1}^{*(8)}(k)$. In chosen-plaintext attacks, since $f_{0\oplus1}^{*(8)}(k,0)$ cannot be freely chosen, the condition is a little more complicated. Let us choose two plaintexts to get the following differential signal $f_{0\oplus1}$: $\forall\ i = 0 \sim N,\ f_{0\oplus1}(i) = (i+1) \bmod 256$. In this case,

assuming that each chaotic bit distributes uniformly, one can calculate $P_c = Prob[f_{0\oplus1}(k,0) \in \{f_{0\oplus1}(k,j)\}_{j=1}^{7}]$, as:

- $P_c = 0$ when $0 \le k \le \lfloor 255/7 \rfloor - 1 = 35$;
- $P_c \le 1/8^{35} = 1/2^{105}$ when $k \ge 36$.

It is obvious that P_c is negligible in all cases, and it is almost true that $\forall\ i,j \in \{0,\cdots,7\}$ and $i \ne j$, $f_{0\oplus1}^{*(8)}(k,i) \ne f_{0\oplus1}^{*(8)}(k,j)$. As a result, the secret permutation of the k-th block can be uniquely determined as a bijective index-mapping $F(k,i) = i'$, where $i,i' \in \{0,\cdots,7\}$. If some bytes in $f_{0\oplus1}(k)$ happen to be identical, one can choose one more pair of plaintexts to try to recover the secret permutations.

3) Breaking the secret data expansion in Step a): Once the secret permutations of two consecutive blocks, $f_{0\oplus1}^{(8)}(k)$ and $f_{0\oplus1}^{(8)}(k+1)$, are broken, one can immediately get the value of $l(k)$ by finding the position of $f_{0\oplus1}^{(8)}(k+1,0)$ in the 8 bytes of $f_{0\oplus1}^{(8)}(k)$.

4) Breaking the secret masking parameters in Step c): After Steps *a)*, *b)* and *d)* are broken, the two intermediate blocks, $f_0^{*(8)}(k)$ and $f_0^{**(8)}(k)$ can be derived from f_0 and f_0', respectively. Then, the masking parameters can be calculated as follows: $\forall\ k,j,\ Seed(k,j) = f_0^{*(8)}(k,j) \oplus f_0^{**(8)}(k,j)$.

5) Breaking the secret chaotic bits and the secret key: Though the recovered secret operations in the above procedure can be used as the equivalent of the secret key to decrypt the ciphertexts, one can still further derive the secret chaotic bits, and then try to derive the values of α, β, μ and $x(0)$. Since the knowledge of $Open$ does not influence the decryption, it is excluded from the secret key.

In Step *a)*, the three involved chaotic bits, $b(33k+0) \sim b(33k+2)$ can be directly derived from the value of $l(k)$.

In Step *c)*, 25 chaotic bits are involved: $b(33k+0) \sim b(33k+7)$ and $b(33k+8) \sim b(33k+15)$ determine $Seed1(k)$ and $Seed2(k)$, respectively, and $b(33k+16) \sim b(33k+24)$ determine $B(k,0) \sim B(k,7)$. To derive the unknown bits, one has to search for the values of $Seed1(k)$ and $Seed2(k)$ in the set $\{Seed(k,0),\cdots,Seed(k,7)\} \subseteq \{Seed1(k), \overline{Seed1(k)}, Seed2(k), \overline{Seed2(k)}\}$. Apparently, the maximal number of possible combinations of $Seed1(k)$ and $Seed2(k)$ is 8. The three known bits $b(33k+0) \sim b(33k+2)$ can be used to eliminate some invalid combinations. Also, note that $B(k,j)$ and $B(k,j+1)$ $(j = 0 \sim 6)$ have a common bit, $b(33k+17+j)$, which can be used as a second constraint to eliminate invalid combinations of $Seed1(k)$ and $Seed2(k)$. In most cases, the values of $Seed1(k)$ and $Seed2(k)$ can be uniquely determined, and then all the 25 chaotic bits can be derived (see the experimental result in the next subsection).

In Step *d)*, 9 chaotic bits, $b(33k+24) \sim b(33k+32)$, are used to determine the values of $p(k,j)$ and $q(k,j)$, together with α and β. Observing the bit-recirculation procedure and Eq. (5), one can see that $\hat{q}(k,j) \in \boldsymbol{Q} = \{\alpha, \alpha+\beta, 8-\alpha, 8-(\alpha+\beta)\}$ holds. So, by exhaustively searching for all $1+\cdots+6 = 21$ possible combinations of α and β, one can determine the 9 chaotic bits with the following equations: $\forall\ j = 0 \sim 7$,

$$b(33k{+}25{+}j) = \begin{cases} 0, & \hat{q}(k,j) \in \{\alpha, 8-\alpha\}, \\ 1, & \hat{q}(k,j) \in \{\alpha+\beta, 8-(\alpha+\beta)\}, \end{cases} \tag{6}$$

$$b(33k{+}24{+}j) = \begin{cases} 0, & \hat{q}(k,j) \in \{\alpha, \alpha+\beta\}, \\ 1, & \hat{q}(k,j) \in \{8-\alpha, 8-(\alpha+\beta)\}. \end{cases} \tag{7}$$

Note Eq. (6) is invalid when $\alpha = 8-(\alpha+\beta)$, i.e., $2\alpha+\beta = 8$, and Eq. (7) is invalid when $\alpha = 4$, $\alpha+\beta = 4$ or $2\alpha+\beta = 8$. According to how the two equations can be used to determine the 9 chaotic bits from $\{\hat{q}(k,j)\}_{j=0}^{7}$, all possible values of (α,β) can be divided into the following three classes.

- *C1)* $\alpha \ne 4$, $\alpha+\beta \ne 4$ *and* $2\alpha+\beta \ne 8$: both Eqs. (6) and (7) are valid, so all the 9 chaotic bits, $b(33k+24) \sim b(33k+32)$, can be uniquely determined. There are 12 *C1*-values, all of which satisfy $\#(\boldsymbol{Q}) = 4$.
- *C2)* $4 \in \{\alpha, \alpha+\beta\}$ *(which ensures* $2\alpha+\beta \ne 8$*)*: Eq. (6) is valid and the 8 chaotic bits, $b(33k{+}25) \sim b(33k{+}32)$, can be uniquely determined. When $\alpha = 4$ and $b(33k+25) = 1$, or $\alpha \ne 4$ and $\widetilde{b}(33k+25) = 0$, one can also determine $b(33k{+}24)$ by Eq. (7). There are 6 *C2*-values, all of which satisfy $\#(\boldsymbol{Q}) = 3$.
- *C3)* $2\alpha+\beta = 8$: Eqs. (6) and (7) are not valid, so all the 9 chaotic bits have to be exhaustively guessed. There are 3 *C3*-values, which satisfy $\#(\boldsymbol{Q}) = 2$.

* For C2/C3-classes, note that $b(33k{+}24)$ can be recovered in Step *c)* in a high probability.

Since the above three classes correspond to different values of $\#(\boldsymbol{Q})$, one need not search for all 21 values of (α,β), but those corresponding to $\#(\boldsymbol{Q})$, which can reduce the search complexity to some extent. The value of $\#(\boldsymbol{Q})$ can be estimated from the cardinality of $\boldsymbol{Q'} = \{\hat{q}(k,j)\}_{k=0,j=0}^{N/7-1,7}$, or one of its subset. It is obvious that $\#(\boldsymbol{Q'}) = \#(\boldsymbol{Q})$ almost true when N is sufficiently large.

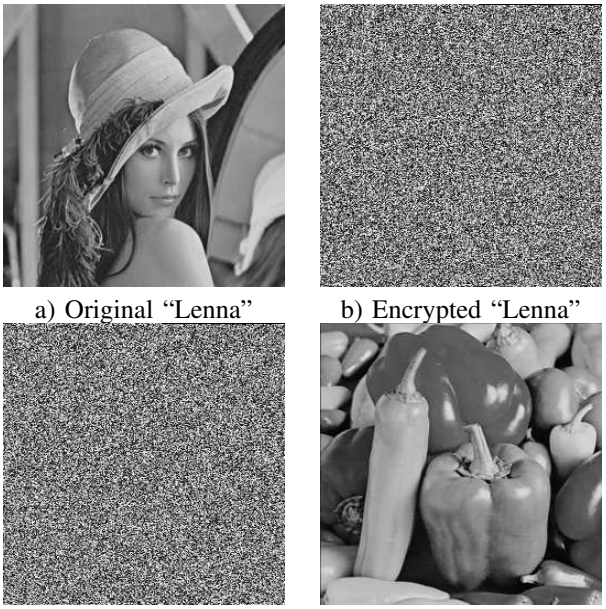
To verify which guessed value of (α,β) is the real one, the following procedure is useful by estimating the values of two consecutive chaotic states, $x(k)$ and $x(k+1)$, and the value of μ. Assuming all the 33 chaotic bits, $b(33k+0) \sim b(33k+32)$ have been successfully recovered (or guessed) with the above procedure, one can immediately get the value of $x(k) = 0.b(33k{+}0) \sim b(33k{+}32)$. After getting $x(k{+}1)$ in a similar way, one can calculate an estimation of μ as follows: $\widetilde{\mu} = \frac{x(k+1)}{x(k)\cdot(1-x(k))}$. Due to the quantization errors introduced in the finite-precision arithmetic, generally $\widetilde{\mu} \ne \mu$. Following the error analysis given in [14, Sec. 4.2], when $x(k+1) \ge 2^{-n}$, $|\widetilde{\mu}-\mu| < 2^{n+3}\cdot 2^{-33}$. Specially, when $x(k{+}1) \ge 2^{-1} = 0.5$, one can exhaustively search for the $2^4 = 16$ values in the neighborhood of $\widetilde{\mu}$ to find the right value of μ.

By iterating the Logistic map from $x(k{+}1)$ until $x(N/7{-}1)$ and then checking the coincidence between these chaotic states and the corresponding bits that can be uniquely derived, one can detect wrong values of (α,β) and μ and distinguish the real ones. To minimize the complexity, one can check only a number of chaotic states, sufficiently far from $x(k+1)$, to

eliminate most wrong values, and verify the few left ones by checking all chaotic states from $x(k+2)$ to $x(N/7-1)$.

C. Experiments and the attack complexity

As discussed above, to carry out the differential chosen-plaintext attack, only three plaintexts are enough to construct two plaintext differentials as follows: 1) $\forall\ i = 0 \sim N-1$, $f_{0\oplus 1}(i) \equiv 1$; 2) $\forall\ i = 0 \sim N-1$, $f_{0\oplus 1}(i) = (i+1) \bmod 256$. When one plaintext is chosen as an image "Lenna", the performance of the proposed attack has been tested, and the results are shown in Fig. 1. The two differentials are used to break the secret operations and then try to break some chaotic bits. It is found that for 8084 blocks in total 9363 ones, all the 33 involved chaotic bits can be uniquely determined. Two chaotic states, $x(1)$ and $x(2)$, are used to estimate μ, and then to find the secret key for recovering the ciphertext of another image "Peppers" (see Figs. 1c and d).



a) Original "Lenna" b) Encrypted "Lenna"

c) Encrypted "Peppers" d) Recovered "Peppers"

Fig. 1. The differential chosen-plaintext attack to MES

Finally, we briefly discuss the attack complexity. It can be easily verified that the complexity of breaking all secret operations is proportional to N. The complexity of breaking the secret key depends on the value of (α, β). When (α, β) belongs to C1 and C2 classes, the attack complexity is also proportional to N; when (α, β) belongs to C3 class, the attack complexity is $2^8 \cdot 2^8 = 2^{16}$ times of the complexity of that in C1/C2 cases, which is still practically small.

IV. Security against brute-force attacks

Another obvious problem of MES is that the key space is not cryptographically large. The secret key for decrypting MES includes $(\mu, x(0))$, which is represented by $2 \cdot 33 = 66$ secret bits, and (α, β), which has 21 possible values. Thus, one can see that the key space of MES is only $21 \cdot 2^{66}$, which is not sufficiently large from the cryptographical point of view [15]. What's worse, since the Logistic map is not chaotic for all values of μ far less than 4, the key space is even smaller than $21 \cdot 2^{66}$. To make MES practically secure in today's digital world, the key space should be not less than $O\left(2^{128}\right)$. One simple method to enlarge the key space is to realize the chaotic Logistic map with a higher finite precision, i.e., to increase the number of secret bits for representing μ and $x(0)$.

V. Conclusion

In this paper, the security of a recently-proposed encryption system called MES [3] has been studied in detail. A differential chosen-plaintext/ciphertext attack has been presented to separate four encryption steps of this product cipher and then break them one by one, with a *divide-and-conquer* (DAC) strategy. After breaking all the four steps, it becomes possible to break the secret key with a cryptographically small complexity. It is also noticed that the security of MES against brute-force attacks is not cryptographically high. It seems not easy to efficiently improve MES without changing the basic encryption structure and the employed encryption techniques.

Acknowledgements

The authors would like to thank Mr. Xiaoyu Ruan from the North Dakota State University for his assistance to this paper. This work was partially supported by the Applied R&D Centres of the City University of Hong Kong under grants no. 9410011 and no. 9620004, and by the National Natural Science Foundation of China under grant no. 60202002.

References

[1] S. Li, "Analyses and new designs of digital chaotic ciphers," Ph.D. dissertation, School of Electronics and Information Engineering, Xi'an Jiaotong University, Xi'an, China, June 2003, available online at http://www.hooklee.com/pub.html.

[2] S. Li, G. Chen, and X. Zheng, "Chaos-based encryption for digital images and videos," in *Multimedia Security Handbook*, B. Furht and D. Kirovski, Eds. CRC Press, December 2004, ch. 4, preprint is available online at http://www.hooklee.com/pub.html.

[3] J.-C. Yen, H.-C. Chen, and S.-S. Jou, "A new cryptographic system and its VLSI implementation," in *Proc. IEEE Int. Symposium on Circuits and Systems (ISCAS'2004)*, vol. 2, 2004, pp. 221–224.

[4] J.-C. Yen and J.-I. Guo, "A new image encryption algorithm and its VLSI architecture," in *Proc. IEEE Workshop on Signal Processing Systems (SiPS'99)*, 1999, pp. 430–437.

[5] H.-C. Chen, J.-C. Yen, and J.-I. Guo, "Design of a new cryptography system," in *Advances in Multimedia Information Processing - PCM 2002: Third IEEE Pacific Rim Conference on Multimedia Proc.*, ser. Lecture Notes in Computer Science, vol. 2532, 2002, pp. 1041–1048.

[6] H.-C. Chen and J.-C. Yen, "A new cryptography system and its VLSI realization," *J. Systems Architechture*, vol. 49, no. 7-9, pp. 355–367, 2003.

[7] J.-C. Yen and J.-I. Guo, "Design of a new signal security system," in *Proc. IEEE Int. Symposium on Circuits and Systems (ISCAS'2002)*, vol. 4, 2002, pp. 121–124.

[8] H.-C. Chen, J.-I. Guo, L.-C. Huang, and J.-C. Yen, "Design and realization of a new signal security system for multimedia data transmission," *EURASIP J. Applied Signal Processing*, vol. 2003, no. 13, pp. 1291–1305, 2003.

[9] S. Li and X. Zheng, "On the security of an image encryption method," in *Proc. IEEE Int. Conference on Image Processing (ICIP'2002)*, vol. 2, 2002, pp. 925–928.

[10] S. Li, C. Li, G. Chen, and X. Mou, "Cryptanalysis of the RCES/RSES image encryption scheme," Cryptology ePrint Archive: Report 2004/376, available online at http://eprint.iacr.org/2004/376, 2004.

[11] C. Li, S. Li, G. Chen, G. Chen, and L. Hu, "Cryptanalysis of a new signal security system for multimedia data transformation," accepted by *EURASIP J. Applied Signal Processing*, 2005, (in press).

[12] Hao Bai-Lin, *Starting with Parabolas: An Introduction to Chaotic Dynamics*. Shanghai, China: Shanghai Scientific and Technological Education Publishing House, 1993, (in Chinese).

[13] S. Li, C. Li, G. Chen, D. Zhang, and N. G. Bourbakis, "A general cryptanalysis of permutation-only multimedia encryption algorithms," Cryptology ePrint Archive: Report 2004/374, available online at http://eprint.iacr.org/2004/374, 2004.

[14] C. Li, S. Li, D. Zhang, and G. Chen, "Cryptanalysis of a chaotic neural network based multimedia encryption scheme," in *Advances in Multimedia Information Processing - PCM 2004 Proc., Part III*, ser. Lecture Notes in Computer Science, vol. 3333. Springer-Verlag, 2004, pp. 418–425.

[15] B. Schneier, *Applied Cryptography – Protocols, algorithms, and souce code in C*, 2nd ed. New York: John Wiley & Sons, Inc., 1996.