2012 Eighth International Conference on Computational Intelligence and Security

Improved impossible differential cryptanalysis of SMS4*

Tao Shi, Wei Wang
School of computer science and technology,
Shandong University
Jinan, China
st6529@163.com,weiwangsdu@sdu.edu.cn

Qiuliang Xu**
School of Computer Science and Technology
Shandong University
Jinan, 250101, China
xql@sdu.edu.cn

***Abstract*—The SMS4 is the first commercial block cipher published by Chinese government.It's a 32-round block cipher encrypted by 128-bit keys.By analyzing the changes of the difference between input and output pairs in each round, this paper presents a new impossible differential path of the 14-round SMS4. Using this path, a new method is submitted to cryptanalyze an 18-round SMS4.The time complexity of the attack is $2^{117.06}$ partial encryptions.**

***Keywords*-block cipher;SMS4;impossible differential attack;**

I. INTRODUCTION

Dating back to 2000, Biham and Keller firstly proposed impossible differential cryptanalysis for AES block cipher. Impossible differential attack is a chosen-plaintext attack, which makes use of the diffusion layer and active bytes to analyze.

Security problem has been the most important problem in the application of wireless LAN. In recent years, the security technology of wireless LAN has rapidly developed, but the security issues has still restricted the further promotion of wireless LAN technology. SMS4 [1] is a Generalized Feistel Network (GFN) cipher, appointed in the Wireless Authentication and Privacy Infrastructure (WAPI), which is official in wireless networks in China. The Chinese Standards Association (SAC) presented WAPI to ISO for recommendation as an international standard, almost the same time as the IEEE 802.11i standard. In consequence, SMS4 was the focus of a huge international controversy since its publication. However, up until recently only some cryptanalysis of SMS4 were submitted. The previously presented cryptanalytic results are the differential fault analysis presented in [11], the integral attack on 13 rounds [8], the rectangle attack on 14 rounds and the impossible differential attack on 16 rounds [9], the rectangle attack on 16 rounds and the differential attack on 21 rounds [12], and finally the rectangle attack on 18 rounds, the differential attack and linear attack on 22 rounds [6].SMS4 is the first commercial block cipher given by Chinese government in 2006. So how to evaluate the security of SMS4 block cipher in a more efficient way is a research hotspot in this field.

The cryptanalytic results on SMS4, on which we pay attention are those of [9].The impossible differential attack on 16-round SMS4 from [9] exploits 2^{105} chosen plaintexts and its time complexity is speculated to be 2^{107} 16-round SMS4 computations.

This paper is organized as follows: In Section 2, we give a brief description of the SMS4 cipher and its properties. In Section 3, we give a new 14-round impossible differential path of SMS4.In Section 4, we present our results on attack 18-round SMS4 of the impossible differential cryptanalysis. Finally, we conclude this paper and summarize our findings in Section 5.

II. DESCRIPTION OF SMS4

A. Notation

Throughout this paper, we will use the notation: Each 128-bit block is made up of four 32-bit words (X_0, X_1, X_2, X_3).Note that the blocks and words are in a Chinese-endian order(i.e., the most significant bit is the leftmost bit numbered 0, and the least significant bit is bit 31 for a 32-bit word).Similarly, the most significant byte of a word is the leftmost byte numbered 0, and least significant byte is numbered 3. We denote the bit rotation of the word *w* by *r* positions to the left by $w \lll r$.

B. The SMS4 block cipher

SMS4 [1] receives a 128-bit plaintext P= (P_0, P_1, P_2, P_3) and a 128-bit user key as inputs, and is constituted of 32 rounds. In each round, the least significant three bytes of the state are XORed with the round key, and then the output passes the *S* transformation. The *S* transformation utilizes an 8-bit to 8-bit bijective Sbox four times in parallel to generate each byte, and then the conjunctive bytes are generated using a linear transformation *L*.

Round Function:

Let $Xi = (X_{i,0}, X_{i,1}, X_{i,2}, X_{i,3})$ and $X_{i+1} = (X_{i+1,0}, X_{i+1,1}, X_{i+1,2}, X_{i+1,3})$ denote the 128-bit input and output to the *i*-th round, separately. Then the round function can be formally expressed as the following equations:

*This work is supported by the National Natural Science Foundation of China (Grant No. 61173139), National Natural Science Foundation of Shandong Province (Grant No. ZR2011FZ005), Project supported by the Specialized Research Fund for the Doctoral Program of Higher Education of China (Grant No.20110131110027).

**Corresponding Author

$$X_(i+1,0) = X_(i,1)$$
$$X_(i+1,1) = X_(i,2)$$
$$X_(i+1,2) = X_(i,3)$$
$$X_{i+1,3} = X_{i,0} \bigoplus L(S(X_{i,1} \bigoplus X_{i,2} \bigoplus X_{i,3} \bigoplus RK_i))$$

where the S transformation uses the Sbox provided in [1] and L is the linear transformation:

$L(x) = x \bigoplus (x \lll 2)(x \lll 10)(x \lll 18)(x \lll 24)\ where x \in Z_2^{32}$

The transformation $L \circ S$ makes up T in the formal document. RK_i is the 32-bit round subkey of the *i*-th round, computed through the key schedule. Decryption is corresponding to the encryption besides the order of the subkeys is used in the reverse order. One round function is presented in Figure 1.

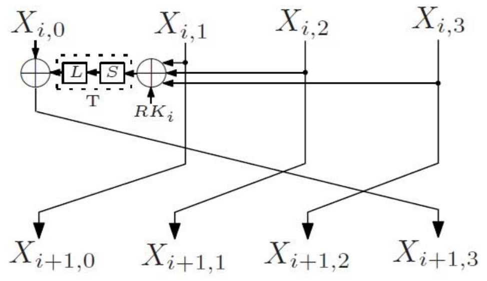

Figure 1: Round Function

Key Schedule:

The key schedule is just like the encryption function. The only difference is that it utilizes L' instead of L as the first transformation:

$L'(x) = x \bigoplus (x \ll 13) \bigoplus (x \ll 23)\ where x \in Z_2^{32}$

Furthermore, the user provided key K is xored with a system parameter, FK. The *j*-th round subkey RK_j is generated as follows:

$FK = (0xA3B1BAC6, 0x56AA3350, 0x677D9197, 0xB27022DC)$
$k = (k_0, k_1, k_2, k_3) = K \bigoplus FK$
$RK_j = k_{j+4} = k_j \bigoplus L'(S(k_{j+1} \bigoplus k_{j+2} \bigoplus k_{j+3} \bigoplus CK_j))$
$where\ CK_j = (ck_{j,0}, ck_{j,1}, ck_{j,2}, ck_{j,3})$
$and\ ck_{j,k} = 28j + 7k(mod 256)$

C. Properties and Definitions

For SMS4 uses a bijective SBox, thus, $S(\Delta x) = 0$ if and only if $\Delta x = 0$. The difference distribution table (DDT) of the SBox includes precisely 127 nonzero output differences for a known nonzero input difference. Only one of all these values exist probability of 2^{-6} while the other 126 remaining nonzero values have probability 2^{-7}.

The following definitions are used for inspecting the transmission of any nonzero input difference to the other rounds. In [9], it is not clearly explained what these sets refer to, and the expressions includes typos. Thus, readers may find the primal terminology confusing. Consequently, we rewrite the equations defining these sets, using the same names for the sets (with a clearer representation).

Suppose the input difference $(0, e_a, e_a, e_a)$to the *n*-th round, where a is a random but nonempty subset of $0, 1, , 31$, the set $G(e_a)$ is made up of all the 32-bit differences that an input difference ea to the T function can lead to:

$G(e_a) = \{x|x = L(\Delta d_1), Pr[S(e_a)\Delta d_1] > 0$
$for x, e_a \in Z_2^{32}\}$

Now, the input difference to the *(n+1)*-th round is (e_a, e_a, e_a, X) where $X \in G(e_a)$. The $H(e_a, X)$ is the set of all 32-bit differences, that an input difference *X* to the *T* function may lead to after a xor with e_a.

$H(e_a, X) = \{y|y = L(\Delta d_2), Pr[S(X) \to \Delta d_2] > 0$
$for X \in G(e_a), y, e_a \in Z_2^{32}\}$

Analogously, the input difference to the *(n+2)*-th round is (e_a, e_a, X, Y) where $X \in G(e_a)$ and $Y \in H(e_a, X)$. The corresponding 32-bit output differences, led by an input difference $e_a \bigoplus X \bigoplus Y$ to T are represented by the set $I(e_a, X, Y)$.

$I(e_a, X, Y) =$
$\{z|z = L(\Delta d_3)\Delta e_a, Pr[S(e_a \bigoplus X \bigoplus\ Y) \to \Delta d_3] > 0$
$for\ z, e_a \in Z_2^{32}\}$

Finally, the input difference to the *(n+3)*-th round is (e_a, X, Y, Z) where $X \in G(e_a)$,$Y \in H(e_a, X)$ and $Z \in I(e_a, X, Y)$. The set of 32-bit differences after the XOR operation in the *(n+3)*-th round with an input difference $X \bigoplus Y \bigoplus Z$ to T is denoted by the set $J(X, Y, Z)$.

$J(e_a, X, Y, Z) =$
$\{w|w = L(\Delta d_4) \bigoplus e_a, Pr[S(X \bigoplus Y \bigoplus Z) \to d_4] > 0$
$for\ w \in Z_2^{32}\}$

III. NEW 14 ROUND IMPOSSIBLE DIFFERENTIAL PATH OF SMS4

In SMS4, only one byte (32bit) of one round will be recomputed, leading that only a new difference will be generated. The new 14-round impossible differentials path we proposed are $(e_a, e_a, e_a, 0) \xrightarrow{pr=0} (0, e_a, e_a, e_a)$, where *a* is a random but nonempty subset of the set $\{0, 1, , 15\}$. These 14-round impossible differentials are based on a miss-in-the-middle manner [3]: an 8-round differential with probability *1* is concatenated with a 6-round differential with probability *1*, however, the middle differences of these two differentials contradict mutually. See Table 1.

Firstly, The 8-round differential with probability 1 is $(e_a, e_a, e_a, 0) \to (e_c, ?, ?, ?)$. The input difference $(e_a, e_a, e_a, 0)$ to Round 0 generates with probability 1 to the difference $(e_a, e_a, 0, e_a)$ after one round, which generates with a probability 1 to the difference $(0, e_a, e_a, e_a)$ after the next two rounds. Secondly, the difference $(0, e_a, e_a, e_a)$ generates to a difference belonging to the set $\{(e_a, e_a, e_a, e_b|e_b \in G(e_a))\}$ after Round 3 exactly, which ultimately transmits with probability 1 to a difference belonging to $\{e_a, x_1, y_1, z_1|x_1 \in I(e_a, e_b, e_c), y_1 \in J(e_b, e_c, x_1), z_1 \in K(e_c, x_1, y_1)\}$ after Rounds 4,5,6 and

7. On the other hand, when we trace back the output difference $(0, e_a, e_a, e_a)$ of the 8-14 rounds differential with the three successive rounds from Rounds 11 to 13 in the opposite direction, we will obtain the difference $(e_a, e_a, e_a, 0)$ just before Round 11 with probability 1. Afterwards, when we trace back the difference $(e_a, e_a, e_a, 0)$ through Round 10, we will exactly obtain a difference belonging to the set $\{(x_2, e_a, e_a, e_a)|x_2 \in G(e_a)\}$. Finally, when we go on to move back for two more rounds, we can exactly obtain a difference belonging to the set $\{(z_2, y_2, x_2, e_a)|x_2 \in G(e_a), y_2 \in H(e_a, e_b), z_2 \in I(e_a, e_b, e_c)\}$just before Round 8. Therefore, a contradiction arises, since we never have the one-round output difference $\{(y_2, x_2, e_a, e_a)|x_2 \in G(e_a), e_c, y_2 \in H(e_a, e_b)\}$ generated from an input difference belonging to $\{(e_a, x_1, y_1, z_1)|x_1 \in I(e_a, e_b, e_c), y_1 \in J(e_b, e_c, x_1), z_1 \in K(e_c, x_1, y_1)\}$.More specifically, to get a one-round output difference belonging to $\{(y_2, x_2, e_a, e_a)|x_2 \in G(e_a), y_2 \in H(e_a, e_b)\}$,the input difference of the 6-round differential should satisfy the set $\{(z_2, y_2, x_2, e_a)|x_2 \in G(e_a), y_2 \in H(e_a, e_b), z_2 \in I(e_a, e_b, e_c)\}$, however, note that the output difference of the 8-round differential is $\{(e_a, x_1, y_1, z_1)|x_1 \in I(e_a, e_b, e_c), y_1 \in (e_b, e_c, x_1), z_1 \in K(e_c, x_1, y_1)\}$, so it is obligatory that the following five conditions should hold for some sextuple $(x_1, y_1, z_1, x_2, y_2, z_2)$, where $x_2 \in G(e_a), y_2 \in H(e_a, e_b), x_1, z2 \in I(e_a, e_b, e_c), y_1 \in J(e_b, e_c, x_1), z_1 \in K(e_c, x_1, y_1)$:

$$e_c = z_2 \tag{1}$$

$$y_2 = x_1 \tag{2}$$

$$x_2 = y_1 \tag{3}$$

$$z_1 = e_a \tag{4}$$

$$L(S(x_2 \bigoplus y_2 \bigoplus e_a)) \bigoplus e_a = e_a \tag{5}$$

By given properties of SMS4 before we can learn that Eq.(5) is equivalent to the following equation:

$$x_1 \bigoplus y_1 \bigoplus e_a = 0 \tag{6}$$

We program a computer search over all the possibilities that may satisfy Eqs. (1)- (4) and (6), but discover that such a qualified sextuple $(x_1, y_1, z_1, x_2, y_2, z_2)$ for any nonempty subset a of the set $\{0, 1, , 15\}$ doesnt exist. Thus, these given 14-round impossible differentials are impossible.

Round $(i)\downarrow$	$(\Delta X_{i,0}, \Delta X_{i,1}, \Delta X_{i,2}, \Delta X_{i,3})$	*Round* $(i)\uparrow$	$(\Delta X_{i,0}, \Delta X_{i,1}, \Delta X_{i,2}, \Delta X_{i,3})$
0	$(e_a, e_a, e_a, 0)$	8	(z_2, y_2, x_2, e_a)
1	$(e_a, e_a, 0, e_a)$	9	(y_2, x_2, e_a, e_a)
2	$(e_a, 0, e_a, e_a)$	10	(x_2, e_a, e_a, e_a)
3	$(0, e_a, e_a, e_a)$	11	$(e_a, e_a, e_a, 0)$
4	(e_a, e_a, e_a, e_b)	12	$(e_a, e_a, 0, e_a)$
5	(e_a, e_a, e_b, e_c)	13	$(e_a, 0, e_a, ea)$
6	(e_a, e_b, e_c, x_1)	output	$(0, e_a, e_a, ea)$
7	(e_b, e_c, x_1, y_1)		
output	(e_c, x_1, y_1, z_1)		

Table I: THE TWO DIFFERENTIALS IN THE 14-ROUND IMPOSSIBLE DIFFERENTIALS

In table 1:

$$e_b, x_2 \in G(e_a), e_c \in H(e_a, e_b),$$

$$x1 \in I(e_a, e_b, e_c), y_1 \in J(e_a, e_b, e_c, x_1),$$

$$z1 \in J(e_b, e_c, x_1, y_1), y2 \in H(e_a, x_2), z_2 \in I(e_a, x_2, y_2)$$

IV. ATTACK 18 ROUND SMS4 BLOCK CIPHER

We can use the 14-round impossible differential to implement an impossible differential attack on SMS4 reduced to 18 rounds, by utilizing the early abort technique introduced in [13]. We assume that the attacked 18 rounds are from Rounds 0 to 17. To reduce the data and time complexities of the attack, we choose a = 0, 1,, 15. We use the 14-round impossible differential from Rounds 2 to 15. Suppose the output difference $(e_{0,1,15}, e_{0,1,15}, e_{0,1,15}, 0)$ of Round 1, thus, there are 127^2 possible input differences to Round 1, and at most 127^6 possible input differences to Round 0; we express them by the set $\Sigma 1$. Given the input difference $(0, e_{0,1,15}, e_{0,1,15}, e_{0,1,15})$ to Round 16, there are at most 127^2 possible output differences just after Round 16, and at most 127^6 possible output differences just after Round17;we express them by the set $\Sigma 2$.

However, in the impossible differential attack of [9], the time complexity analysis is calculated only for candidates of right pairs after preliminary elimination (the pairs which enter Step 2), and it does not include the time complexity of the first elimination itself. Moreover, the data complexity suggested in [9] is too low. Therefore, we use a new method to generate 18 round attacks.

Solve the data complexity problem: Each structure is constituted of 2^{96} plaintexts of the form $(*, *, (f, b_i), (c, d_i))$ where * denotes all possible values and f, c denote the chosen invariable of the structure, i.e., each structure contains $(2^{96})^2/2 = 2^{191}$ pairs. So as to get the desired input difference $(*, *, e_a, e_a)$, we should make $b_i \bigoplus b_j = d_i \bigoplus d_j = (\overline{e_a})$ where $(\overline{e_a})$ is the least significant two bytes of e_a for each a.

For each (b_i, b_j, d_i, d_j) in a structure, there are $2^{64} 2^{64} = 2^{128}$possible pairs of plaintexts. There are $2^{16} 2^{16}/2 = 2^{31}$ possible (b_i, b_j) pairs, and for each pair there exists 2^{16} possible d_is. Given (b_i, b_j) and d_i, there is a sole d_j value meeting the above condition. Therefore, only $2^{128}2^{31}2^{16} = 2^{175}$ of the 2^{191} pairs meet the desired input difference.

$\Sigma_1(a)$ is made up of $127^6 2^{42}$ possible input differences for each a. Therefore, the probability of a pair to have $P_1 \bigoplus P_2 \in \Sigma_1(a)$ is $2^{42}/2^{64} = 2^{-22}$, and $2^{175} \times 2^{-22} = 2^{153}$ pairs satisfy this step. Attention that once the plaintext pair is set, a is also set, so is $\Sigma_1(a)$ and $\Sigma_2(a)$. Just like $\Sigma_1(a)$, $\Sigma_2(a)$ is made up of $127^6 2^{42}$ possible output differences for each a. Therefore, the probability of a pair to have $C_1 \bigoplus C_2 \in \Sigma_2(a)$ is $2^{42}/2^{128} = 2^{-84}$ and the number of pairs for a known structure satisfying the Step 1 of the algorithm is $2^{153}2^{-84} = 2^{69}$.

Suppose that there exist N such structures, the number of plaintext pairs satisfying the preliminary elimination is $N?2^{69}$. The probability that a given subkey is dropped by a given structure is, $2^{69} \times (2^{-7})^{12} = 2^{-15}$, and that it is not dropped by all N structures is $(1 - 2^{-15})^N$. So as to drop all wrong subkeys, we need to guarantee that the probability of a wrong key to remain is about 2^{-96}, i.e., $2^{-96} = 1 - (1 - 2^{-15})^N$. So for $N = 2^9$, it is not possible to drop most of the subkey guesses.

The number of needful structures can be computed as follows: There are 2^{96} possible subkeys, and $N?2^{-15}$ pairs are desired for each subkey. In order to get all wrong subkeys with one pair (i.e., suggested by some pair and thus distinguished as wrong ones), the probability of a wrong key to have no pairs must be less than 2^{-96}. The probability of having no pairs is $e^{-N?2^{-15}}$. Solving this math problem, we get that $N = 2^{21.06}$ structures are needed for our attack.

Detection of candidate pairs procedure: Define a plaintext as $P_i = (P_{i,0}, P_{i,1}, (f_i, b_i), (c_i, d_i))$, a ciphertext as $C_i = ((w_i, x_i), (y_i, z_i), C_{i,2}, C_{i,3})$.

1) Input every plaintext-ciphertext pair (P_i, C_i) of each structure, indexed by the least significant 2 bytes of the rightmost two words of the plaintext and the most significant two words of the corresponding ciphertext (i.e., $b_i \parallel d_i \parallel w_i \parallel x_i \parallel y_i \parallel z_i$) into a hash table.
2) For each $\overline{e_a}$:
   For every non-empty bin satisfying $b_i < b_j$:
   - move to the corresponding bin:
     $b_i \parallel d_i \parallel w_i \parallel x_i \parallel y_i \parallel z_i \bigoplus \overline{e_a} \parallel \overline{e_a} \parallel e_a \parallel e_a = b_j \parallel d_j \parallel w_j \parallel x_j \parallel y_j \parallel z_j$(i.e.,$w_i = w_j$ and $y_i = y_j$)
   - For all possible input values, choose the plaintext pairs for which:
     A $\quad P_{i,1} \bigoplus P_{j,1} \in G(e_a)$
     B $\quad C_{i,2} \bigoplus C_{j,2} \in G(e_a)$
     C $\quad P_{i,0} \bigoplus P_{j,0} \in H(e_a, P_{i,1} \bigoplus P_{j,1})$
     D $\quad C_{i,3} \bigoplus C_{j,3} \in H(e_a, C_{i,2} \bigoplus C_{j,2})$
     is satisfied. (If one of them fails, do not run the remaining conditions.)
3) If any pair satisfying (A)-(D) is found, utilize it in Steps 2-6 of the attack latter.

The following is the time complexity of the preliminary elimination: In Step 1, we have 2^{96} memory accesses for each structure. There are 2^{96} plaintext-ciphertext pairs in a structure, thus, the expected number of inputs in each of the 2^{96} bins is 1. The resulting number of required memory accesses for Step 2 is $2^{16} * 2^{96}/2 = 2^{111}$ for a given structure. So the total number of memory accesses of our procedure is $N * 2^{111} = 2^{132.06}$. The attack procedure of our work is as follows:

1) Choose structures according to the procedure given above.
2) For all the remaining ciphertext pairs (C_i, C_j):
   a Calculate the 4-byte difference just before the L transformation in round 17, and denote it by $\Delta_{i,j}^{17} (i.e., \Delta_{i,j}^{17} = L^{-1}(C_{i,3} \bigoplus C_{j,3}))$.
   b For l=0 to 3:
   - Guess the l-th byte of the subkey RK_{17} and partially decrypt (C_i, C_j) to get the l-th byte of the difference just after the S transformation in round 17, denote them by $(T_{i,l}, T_{j,l})$.
   - Check if $T_{i,l} \bigoplus T_{j,l} = \Delta_{i,j,l}^{17}$ and keep the pairs that satisfy the equality.
3) For all the remaining pairs (T_i, T_j):
   a Calculate the 4-byte difference just before the L transformation in round 16 and denote it by $\Delta_{i,j}^{16}$.
   b For l=0 to 1:
   - Guess the l-th byte of the subkey RK16 and partially decrypt (T_i, T_j) to get the l-th byte of their intermediate values just after the S transformation in round 16, denote them by $(Q_{i,l}, Q_{j,l})$.
   - Check if $Q_{i,l} \bigoplus Q_{j,l} = \Delta_{i,j}^{16}$ and keep the pairs that satisfy the equality.
4) For all plaintext pairs (P_i, P_j) corresponding to the remaining ciphertexts after Step 3:
   a Calculate the 4-byte difference just before the L transformation in round 0 and denote by it $\Delta_{i,j}^{0}$.
   b For l=0 to 3:
   - Guess the l-th byte of the subkey RK_0 and partially encrypt (P_i, P_j) to get the l-th byte of their intermediate values just after the S transformation in round 0, denote them by $(R_{i,l}, R_{j,l})$
   - Check if $R_{i,l} \bigoplus R_{j,l} = \Delta_{i,j,l}^{0}$ and keep the pairs that satisfy the equality.
5) For all the remaining pairs (R_i, R_j):
   a Calculate the 4-byte difference just before the L transformation in round 1 and denote by $\Delta_{i,j}^{1}$.

b For l=0 to 1:

- Guess the l-th byte of the subkey RK_1 and partially encrypt (R_i, R_j) to get the l-th byte of their intermediate values just after the S transformation in round 1. Denote them by $(S_{i,l}, S_{j,l})$.
- Check if $S_{i,l} \bigoplus S_{j,l} = \Delta^1_{i,j,l}$. If there exists a qualified pair then discard the guess of 96 subkey bits and try another, otherwise proceed to the next step.

6) Guess the user key from the known subkey values, and perform a trial encryption. If a key is suggested then output it. Otherwise, continue with a new guess of RK_{17} (i.e., go to Step 2).

As referred to earlier in data complexity issues, the number of pairs satisfying the preliminary elimination is 2^{69} per structure. Hence, the number of plaintext pairs satisfying the preliminary elimination is $2^{21.06} * 2^{69} = 2^{90.06}$, starting with $S = 2^{21.06}$ structures, which results in $2^{117.06}$ chosen plaintexts. The time complexity of our 18 round encryptions/decryptions in Steps 2 b),3 b),4 b) and 5 b) of the algorithm is:

$$\sum_{i=1}^{14}(2^{91.06} * \frac{1}{127^{i-1}} * 2^8) * \frac{1}{18} = 2^{96}$$

However, the time complexity of the attack is dominated by the $2^{117.06}$ partial encryptions required to get the ciphertext pairs in Step 1, and by the $2^{132.06}$ memory accesses produced by the preliminary elimination.

V. Conclusion

In this paper, we proposed a new 14-round SMS4 impossible differential path by reviewing 16-round SMS4 cryptanalysis in [9].Also, we use a new method in the attack algorithms to lower the time and data complexity.

We also found some flaws in the previous impossible differential attack in [9].Throughout this paper, we know that more data is needed for the analysis, and we use a new algorithm for the preliminary elimination.

As to normal differential attacks, impossible differential attack is more effective in some block ciphers such as AES. Application in SMS4 from our work is also a new effective way in attacking other than differential attack.

References

[1] Beijing Data Security Technology Co. Ltd, Specification of SMS4 (2006)(inChinese),http://www.oscca.gov.cn/UpFile/200621016423197990.pdf

[2] Biham, E., Dunkelman, O., Keller, N.: The Rectangle Attack Rectangling the Serpent. In: Pfitzmann, B. (ed.) EUROCRYPT 2001. LNCS, vol. 2045, pp. 340C357. Springer, Heidelberg (2001).

[3] Biham, E., Birjukov, A., Shamir, A.: Miss in the Middle Attacks on IDEA and Khufu. In: Knudsen, L.R. (ed.) FSE 1999. LNCS, vol. 1636, pp. 124C138. Springer, Heidelberg (1999)

[4] Kim, T., Kim, J., Hong, S., Sung, J.: Linear and Differential Cryptanalysis of Reduced SMS4 Block Cipher, Cryptology ePrint Archive: Report 2008/281 (2008).

[5] Knudsen, L.R., Wagner, D.: Integral Cryptanalysis. In: Daemen, J., Rijmen, V. (eds.) FSE 2002. LNCS, vol. 2365, pp. 112C127. Springer, Heidelberg (2002).

[6] Liu, F., et al.: Analysis of the SMS4 Block Cipher. In: Pieprzyk, J., Ghodosi, H., Dawson, E. (eds.) ACISP 2007. LNCS, vol. 4586, pp. 158C170. Springer, Heidelberg (2007):

[7] Lu, J.: Attacking Reduced-Round Versions of the SMS4 Block Cipher in the Chinese WAPI Standart. In: Qing, S., Imai, H., Wang, G. (eds.) ICICS 2007. LNCS, vol. 4861, pp. 306C318. Springer, Heidelberg (2007).

[8] Lu, J., Kim, J., Keller, N., Dunkelman, O.: Differential and rectangle attacks on reduced-round SHACAL-1. In: Barua, R., Lange, T. (eds.) INDOCRYPT 2006. LNCS, vol. 4329, pp. 17-31. Springer, Heidelberg (2006)

[9] Lu, J., Kim, J., Keller, N., Dunkelman, O.: Improving the efficiency of impossible differential cryptanalysis of reduced Camellia and MISTY1, Archive available at: http://jiqiang.googlepages.com

[10] D. Toz, O. Dunkelman, Analysis of two attacks on reduced-round versions of the SMS4, ICICS 2008, Lecture Notesin Computer Science,Springer-Verlag,Vol.5308,pp.141