Computers and Electrical Engineering 36 (2010) 180–189

Contents lists available at ScienceDirect

Computers and Electrical Engineering

journal homepage: www.elsevier.com/locate/compeleceng

A new impossible differential attack on SAFER ciphers

Shihui Zheng, Licheng Wang *, Yixian Yang

Information Security Center, State Key Laboratory of Networking and Switching Technology, Key Laboratory of Network and Information Attack & Defence Technology of MOE, Beijing University of Posts and Telecommunications, Beijing 100876, PR China

ARTICLE INFO

Article history:
Received 19 January 2009
Accepted 26 August 2009
Available online 17 October 2009

Keywords:
Impossible differential
SAFER-SK
SAFER+
SAFER++

ABSTRACT

This paper presents an improved impossible differential cryptanalysis of SAFER ciphers, which uses the miss-in-the-middle technique developed by Biham et al. We analyze 3.75-round SAFER SK-64,[1] using 2^{45} chosen plaintexts, 2^{38} bytes memory and 2^{42} half round computations. Furthermore, the new impossible differential attack on 3.75-round SAFER+/128 uses 2^{78} chosen plaintexts, 2^{75} half round computations and 2^{68} bytes memory. And attack on 3.75-round SAFER++/128 uses 2^{78} data, 2^{66} time, and 2^{62} memory.

1. Introduction

Impossible differential (ID) attack is first used by Knudsen [1]. Later, Biham et al. [2] had a presentation about it and gave the name "impossible differential" at the rump session of CRYPTO'98. Impossible differential attack exploits differential holding with probability 0 over a number of rounds of the cipher. Then with a chosen text pair, the attacker can point out some wrong subkeys, which may cause the impossible differential, and discard them. Finally, the right subkey comes out, after selecting lots of particular text pairs and repeating the filtering algorithm.

The main technique to construct an impossible differential is the "miss-in-the-middle" method. In other words, the ID can be divided into two differentials. Both of them hold with probability 1, but their combination leads to a contradiction. Generally, some bytes of one differential's difference do not equal zero, but the corresponding bytes of the other must be zero. To our knowledge, the longest IDs for SAFER ciphers are 1.75-round [11] so far, and the authors proposed a 3.75-round attack on SAFER SK-64 and 2.75-round attack to SAFER+ and SAFER++.

We inspired by the special properties of the linear transformation of SAFER ciphers and found that no matter forward or reverse, for a type of input difference, the parity of some bytes of the 2-round output difference can be computed. Therefore, we conceived to design the IDs of SAFER ciphers by the conflict of the byte parity. However, we failed to construct a usable ID with the aforementioned type of input differences, because the corresponding attack complexity is more than that of the exhaustive search. However, with some other tricks, a 3-round differential is constructed for SAFER SK, SAFER+ and SAFER++ separately. And the contradiction of these IDs are described as between the odd byte and the even byte, though they are still between the zero byte and non-zero byte.

These 3-round IDs can be used to attack 3.75-round SAFER SK-64, SAFER+/128 and SAFER++/128. A summary of new ID attack results are listed in Table 1, taking the results in [11] for comparison.

Several theoretical attacks have been published on the ciphers of the SAFER family: differential cryptanalysis by Massey [7,10]. Strong evidence is given that SAFER is secure against differential cryptanalysis after five rounds. Truncated

* Corresponding author.
E-mail address: wanglc.cn@gmail.com (L. Wang).

[1] (In this paper, the number in the name of the ciphers indicates the key length).

0045-7906/$ - see front matter © 2009 Elsevier Ltd. All rights reserved.
doi:10.1016/j.compeleceng.2009.08.004

Table 1
Summary of ID attacks on SAFER ciphers.

Name	Rounds	From the first round	Recovered key bits	Data[a]	Comp.[b]	Memory (Bytes)
SAFER SK-64 [11]	3.75	No	64	2^{32}	2^{65}	2^{37}
SAFER SK-128 [11]	2.75	Yes	64	2^{39}	2^{49}	2^{64}
SAFER+/128 [11]	2.75	Yes	128	2^{64}	2^{60}	2^{104}
SAFER++/128 [11]	2.75	Yes	128	2^{64}	2^{60}	2^{104}
SAFER SK-64	3.75	Yes	64	2^{45}	2^{42}	2^{38}
SAFER SK-128	3.75	Yes	40	2^{45}	2^{42}	2^{38}
SAFER+/128	3.75	Yes	128	2^{78}	2^{75}	2^{68}
SAFER+/256	3.75	Yes	72	2^{78}	2^{75}	2^{68}
SAFER++/128	3.75	Yes	128	2^{78}	2^{66}	2^{62}
SAFER++/256	3.75	Yes	72	2^{78}	2^{74}	2^{70}

[a] Expressed in number of blocks.
[b] Expressed in number of half round computation.

differentials by Knudsen and Berson [4,18], later improved by Wu et al. [19], some algebraic properties of SAFER ciphers are investigated by Murphy [14], key schedule attacks by Knudsen [20] and by Kelsey et al. [16,21], and observations on the PHT design by Vaudenay [23], and Brincat et al. [22] and Massey [15]. Linear cryptanalysis of SAFER K-64 has been considered by Harpes et al. [24], where SAFER K-64/SK-64 is shown to be secure against linear cryptanalysis after two rounds. Then, Nakahara et al. proposed the linear cryptanalysis of all SAFER ciphers [13,12]. Piret and Quisquater cryptanalyzed SAFER++ with integral cryptanalysis [17]. Yeom et al. optimized the results and claimed that they can attack up 4.25 rounds of SAFER++/128 and 4.75 rounds of SAFER++/256. Biryukov et al. [3] gave multiset attack and boomerang attack of SAFER++. For each type of attack method, the best, i.e., longest rounds, attack results on SAFER Ciphers are listed in Table 2.

The paper is organized as follows: a brief review of SAFER ciphers and some important properties, which will be used in the following impossible differential attacks, are presented in Section 2. Then the detail analysis process and results to SAFER SK-64, SAFER+/128 and SAFER++/128 are given in Section 3. The analysis results of SAFER SK-128, SAFER+/256 and SAFER++/256, which only recover partial key bits, are included in this section also. Finally, Section 4 concludes the paper.

2. The SAFER family and properties of round function

The SAFER family comprises SAFER K [6,7], SAFER SK [8], SAFER+ [9], and SAFER++ [10]. The block size of the first two ciphers is 64 bits and SAFER SK is proposed to strengthen the key schedule of SAFER K. Thus, we just analyze SAFER SK-64 and SAFER SK-128 in this paper.

Table 2
Summary of attacks on SAFER ciphers.

Version[a]	Name	Rounds	Type[b]	Fraction of keys	Data[c]	Comp.[d]	Mem. Bytes
K-64/SK-64	TD[e] [19]	6	CP	All	2^{53}	2^{73}	2^{59}
K-64/SK-64	TD [19]	6	CP	All	2^{53}	2^{73}	2^{59}
K-64/SK-64	TD [19]	6	CP	All	2^{53}	2^{73}	2^{59}
SK-128	LNH[f] [12]	4.75	KP	All	2^{64}	$2^{99.5}$	2^{70}
SK-128	LNH [12]	2.75	CO	All	2^{29}	$2^{92.5}$	2^{70}
+/128	LNH [12]	3.25	KP	All	2^{101}	$2^{143.5}$	2^{108}
++/128	LNH [12]	3	KP	2^{-13}	2^{81}	2^{107}	2^{88}
++/256	LNH [12]	3	KP	2^{-13}	2^{81}	2^{182}	2^{88}
++/128	Integral [17]	4	CP	All	2^{64}	2^{128}	2^{23}
++/128	Integral [17]	4	CP	All	2^{64}	2^{120}	2^{71}
++/256	Integral [17]	4	CP	All	2^{64}	2^{152}	2^{71}
++/128	Integral [25][g]	4.25	CP	–	–	–	–
++/256	Integral [25]	4.75	CP	–	–	–	–
++/128	Multiset [3]	4.5	CP	128	2^{48}	2^{103}	2^{55}
++/128	Boomerang [3]	5.5	CP/ACC	128	2^{108}	2^{119}	2^{55}

[a] K-SAFER K, SK-SAFER SK, + – SAFER+, ++ – SAFER++.
[b] KP, known plaintext; CP, chosen plaintext; ACP, adaptive chosen plaintext; CO, ciphertext only; CC, chosen ciphertext; ACC, adaptive chosen ciphertext.
[c] Expressed in number of blocks.
[d] Expressed in number of half round computation.
[e] Abbr. of truncated differential.
[f] Abbr. of linear(non-homomorphic).
[g] As it published in Japan, we did not see the original paper.

SAFER+ was designed as a candidate for Advanced Encryption Standard. It is a 128-bit block cipher with a variable key size of 128, 192 or 256 bits, denoted by SAFER+/128, SAFER+/192 and SAFER+/256, respectively. Since some weaknesses to the key schedules of SAFER+/192 and SAFER+/256 are discovered, Massey et al. change the key schedule algorithm later. Here, we use the remedied key schedule algorithm as that of in [11]. SAFER++ was submitted to NESSIE project and was among the primitives selected for the second phase of this project. Its block size is also 128 bits and contains two versions SAFER++/128 and SAFER++/256.

2.1. Description of round function and key schedule of SAFER ciphers

The SAFER ciphers are byte-oriented cipher and have substitution–permutation network. Every round consists of an upper-key-mixing (UKM) layer, a S-BOX layer, a lower-key-mixing (LKM) layer and a linear transformation (LT) layer. After the final round there is an output transformation that is similar to the upper key layer.

The operations of the first three layers are the same in all SAFER members. In the upper-key-mixing layer, one round-key is combined with data by byte via exclusive-or and modulo addition in alternate fashion (see Figs. 1–3). And in the lower-key-mixing layer, another round-key alternately add and XOR with the data by byte. (see Figs. 1–3). In the S-BOX layer, two kinds of S-BOXes, denoted by **X** and **L**, are applied alternately too (see Figs. 1–3). They are defined as

$$\mathbf{X}(a) = (45^a \bmod 257) \bmod 256,$$
$$\mathbf{L}(a) = \log_{45}(a) \bmod 257$$

with the special case that $\mathbf{L}(0) = 128$.

Let X be the input of the LT layer and Y be the output, the LT can be represented by $Y = X \cdot M$. Here, M is pseudo Hadamard transformation (PHT) based matrix, which is distinct among SAFER SK, SAFER+ and SAFER++.

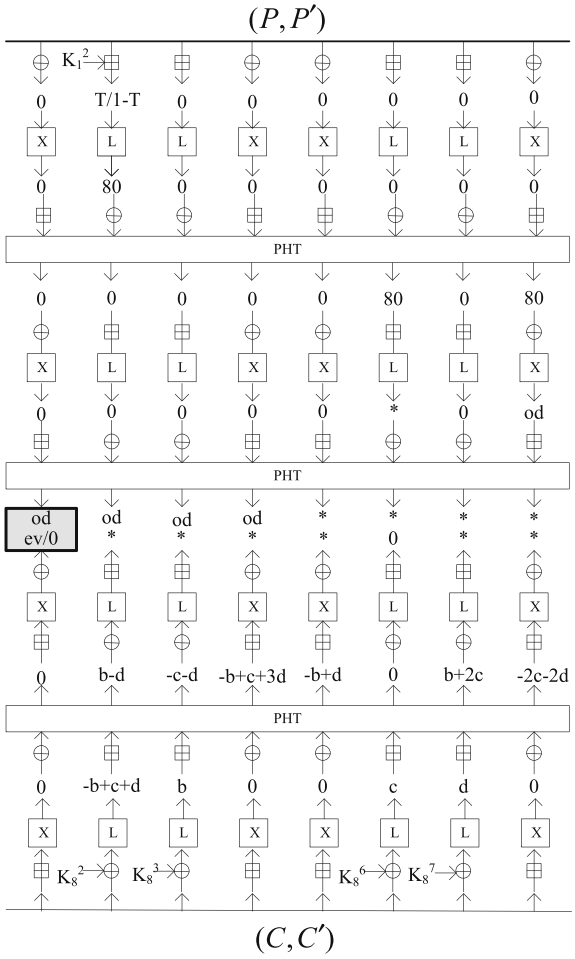

Fig. 1. Impossible differential attack on 3.75 rounds SAFER SK-64.

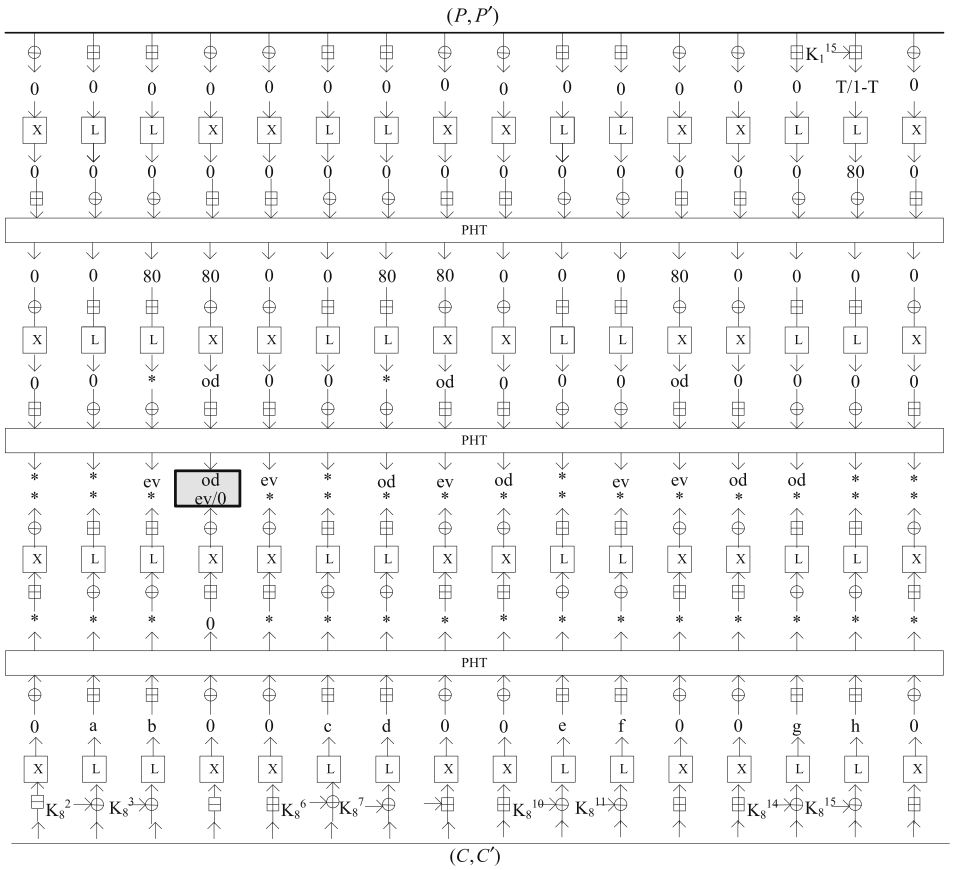

Fig. 2. Impossible differential attack on 3.75 rounds SAFER+/128.

The key schedule algorithms differ between the different versions of the ciphers. However, the basic processes to construct a round-key are similar. Firstly, the key is parted into bytes. Then, one additional key byte, denoted by $sp1$, is computed via XORing all the bytes. When the key size is more than the block size, two additional key bytes are computes, denoted by $sp1$, $sp2$. Finally, to generate a round-key, it picks 8 bytes (or 16 bytes for SAFER+ and SAFER++) according to some rules, rotates these bytes separately and adds some constant bytes to them. While the first round-key (or the first two round-keys when the key size is more than the block size) keep unchanged.

We take the key schedule of SAFER SK-64 as an example. The key length is 64-bit, which can be parted into 8 bytes, denoted by $K = (k_0^1, k_0^2, k_0^3, k_0^4, k_0^5, k_0^6, k_0^7, k_0^8)$. Only one additional byte $k_0^9 = sp1 = \oplus_{i=1}^{8} k_0^i$ is computed here. The first round-key is $K_1 = K$ and the ith round-key is $K_i = (ROL_{3(i-1)}(k_0^{i-1}), ROL_{3(i-1)}(k_0^i), \ldots, ROL_{3(i-1)}(k_0^{(i+7) \bmod 8})) + (b_i^1, \ldots, b_i^8)$. Here, b_i^j $(j = 1, \ldots, 8)$ is a constant byte, which can be pre-computed and stored in the memory, and $ROL_{3(i-1)}(k_0^*)$ means that the byte k_0^* is rotated $3(i-1)$ bits.

In Tables 3–6, we give some tables to show how key bytes are used in the first ten round-keys of SAFER ciphers. Because the key schedule of SAFER++ is same as the one of SAFER+, we just list four tables for SAFER SK-64, SAFER SK-128, SAFER+/128 and SAFER+/256. In each table, the numbers in the first column represent the number of the round and the remaining columns show the key bytes included in every round-key.

2.2. Some notations used in differentials

In the remainder of this paper, addition modulo 256 will be denoted by $\boxplus$ and subtraction modulo 256 by $\boxminus$.

P is a plaintext block and (P, P') is a plaintext pair. C is a ciphertext block and (C, C') is a ciphertext pair. And p denotes an input byte and k denotes a key byte.

Suppose (p, p') is a byte pair, then the difference of this pair is defined as:

$$\Delta p = (p - p') \bmod 256.$$

Δp is called a byte difference and the following symbols represent the byte differences with special meaning.

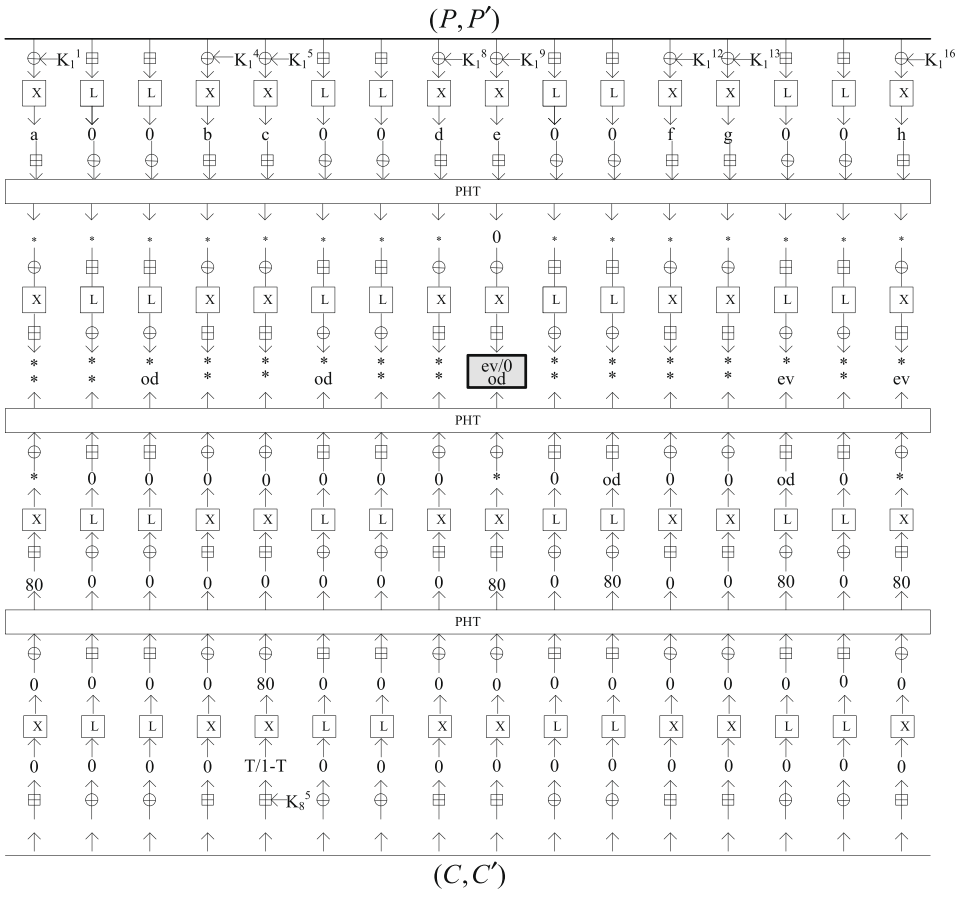

Fig. 3. Impossible differential attack on four rounds SAFER++/128.

0: $\Delta p = 0$.
a, b, c, d, e, f, g, h: each of them may be any given values in $\mathbb{Z}_{256}\backslash\{0\}$, but they together satisfy some linear relations.
80: 80 is hexadecimal, it means $\Delta p = 0x80$.

ev: $\Delta p \equiv 0 \pmod 2$.
od: $\Delta p \equiv 1 \pmod 2$.
$*$: the difference Δp may be any 8-bit value.
In addition, the notation $T/1-T$ means that the pair p, p' satisfies the relation $p + p' \equiv 1 \bmod 256$.

2.3. Important properties of round function

Proposition 1 (see [3,5]). *For every $p \in \mathbb{Z}_{256}$, the following equations hold.*

$$\mathbf{X}[\mathbf{L}(p)] = \mathbf{L}[\mathbf{X}(p)] = p,$$
$$\mathbf{X}(p \oplus 0x80) + \mathbf{X}(p) \equiv 1 \quad (mod\ 256),$$
$$\mathbf{L}(1-p) - \mathbf{L}(p) \equiv 0x80 \quad (mod\ 256).$$

Proposition 2 (see [7]). *For any byte pair (p, p'), if $(p - p') \equiv 0x80 \pmod{256}$, then the output difference $\mathbf{X}(p) \boxminus \mathbf{X}(p')$ is always odd.*

Proposition 3 (see [5]). *For any byte pair $(p, p'), p \oplus p' = 0x80$ always means $(p - p') \equiv 0x80 \pmod{256}$, and vice versa.*

In addition, we develop the following two propositions.

Proposition 4. *For any given byte pair (p, p'), if $p \oplus p'$ is odd, then $(p \boxplus k) \oplus (p' \boxplus k)$ is odd. Also, if $p \boxplus p'$ is odd, $(p \oplus k) \boxminus (p' \oplus k)$ is odd. Here, k can take any value in $\mathbf{Z}_{256}$.*

Proof. If $p \oplus p'$ and $p \boxplus p'$ are odd, then $p \bmod 2 \neq p' \bmod 2$.

Suppose $p \bmod 2 = 1$ and $p' \bmod 2 = 0$.

If $k \bmod 2 = 1$, we have

$$(p \boxplus k) \bmod 2 = 0, \quad (p' \boxplus k) \bmod 2 = 1,$$
$$(p \oplus k) \bmod 2 = 0, \quad (p' \oplus k)) \bmod 2 = 1.$$

Obviously, the above claims are correct.

If $k \bmod 2 = 0$, we have

$$(p \boxplus k) \bmod 2 = 1, \quad (p' \boxplus k) \bmod 2 = 0,$$
$$(p \oplus k) \bmod 2 = 1, \quad (p' \oplus k)) \bmod 2 = 0.$$

The above claims are correct also. □

Proposition 5. *For any given byte pair (p, p'), if $p \boxplus p'$ ($p \boxminus p'$) is odd, then there are two solutions $k_1, k_2 \in \mathbf{Z}_{256}$ to the following equation*

$$(p \boxplus k) + (p' \boxplus k) \equiv 1 \pmod{256}.$$

They are $k_1 = [257 - (p \boxplus p')]/2$ and $k_2 = [513 - (p \boxplus p')]/2$. If $p \boxplus p'$ is even, the above equation has no solution.

Proof. Let $p \boxplus p' = t$, then we have

$$((p \boxplus k) + (p' \boxplus k)) \bmod 256 = (p + 2k + p') \bmod 256 = (t + 2k) \bmod 256.$$

$t + 2k = 1 \bmod 256$ *means* $2k = 1 \boxminus t$. So, if $2 \nmid (1 \boxminus t)$ (i.e. t is even), there is no solution. However, if $2 | (1 \boxminus t)$, there are infinite solutions. Furthermore, k is a byte, thus there are only two solutions: $2k = 257 - t$ and $2k = 513 - t$. □

3. Impossible differential attack on SAFER ciphers

3.1. Impossible differential attack on SAFER SK

The ID cryptanalysis on SAFER SK-64 uses the following impossible differential:

$$(0, T/1 - T, 0, 0, 0, 0, 0, 0) \leftrightarrow (0, a, b, 0, 0, c, d, 0).$$

Here, (a, b, c, d) meets one of the following conditions:

$$\begin{aligned} &a \equiv -b + c + d \pmod{256}, \\ &a \equiv -2b + c + 2d \pmod{256}, \\ &a \equiv -b + 2c + 2d \pmod{256}, \\ &a \equiv -2b + 2c + 4d \pmod{256}. \end{aligned} \tag{1}$$

As mentioned above, the ID can be divided into two differentials. The first one (in the encryption direction) starts at the input of the S-BOX layer in the first round, with difference (0,$T/1 - T$,0,0,0,0,0,0). And it causes with probability one the difference $(od, od, od, od, *, *, *, *)$ after the LT layer of the second round.

$$\begin{aligned} &(0, T/1 - T, 0, 0, 0, 0, 0, 0) \xrightarrow[\text{Proposition 1}]{\text{S-BOX layer}} (0, 80, 0, 0, 0, 0, 0, 0) \\ &\xrightarrow[\text{Proposition 3}]{\text{LKM layer}} (0, 80, 0, 0, 0, 0, 0, 0) \xrightarrow{\text{LT layer}} (0, 0, 0, 0, 0, 80, 0, 80) \\ &\xrightarrow[\text{Proposition 3}]{\text{UKM layer}} (0, 0, 0, 0, 0, 80, 0, 80) \xrightarrow[\text{Proposition 2}]{\text{S-BOX layer}} (0, 0, 0, 0, 0, *, 0, od) \\ &\xrightarrow[\text{Proposition 4}]{\text{LKM layer}} (0, 0, 0, 0, 0, *, 0, od) \xrightarrow{\text{LT layer}} (od, od, od, od, *, *, *, *). \end{aligned}$$

On the other hand, the second one (in the decryption direction) has difference $(0, a, b, 0, 0, c, d, 0)$ at the output of the UKM layer of the fourth round. With different conditions among a, b, c, d, the resulted differences are different. For example, provided $a \equiv -b + c + d \pmod{256}$, the difference at the input of the UKM layer in round 3 is $(ev, *, *, *, *, 0, *, *)$. So the contradiction arises in the first byte at the input of the UKM layer in round 3. It will always be odd according to the first one, but should be even (i.e. 0) according to the second one (Fig. 1).

Other conditions in (1) separately lead to difference $(*, 0, *, *, *, *, *, *)$ or $(*, *, 0, *, *, *, *, 0)$ or $(*, *, *, 0, *, *, *, *)$ at the input of the UKM layer in round 3. And the contradiction will occur at the 2nd byte, the 3rd byte or the 4th byte of the input of the UKM layer.

Table 3
How key bytes are used in first five rounds of SAFER SK-64.

Round	Key bytes							
01	01	02	03	04	05	06	07	08
	02	03	04	05	06	07	08	sp1
02	03	04	05	06	07	08	sp1	01
	04	05	06	07	08	sp1	01	02
03	05	06	07	08	sp1	01	02	03
	06	07	08	sp1	01	02	03	04
04	07	08	sp1	01	02	03	04	05
	08	sp1	01	02	03	04	05	06
05	sp1	01	02	03	04	05	06	07
	01	02	03	04	05	06	07	08

Table 4
How key bytes are used in first five rounds of SAFER SK-128.

Round	Key bytes							
01	09	10	11	12	13	14	15	16
	02	03	04	05	06	07	08	sp1
02	11	12	13	14	15	16	sp2	09
	04	05	06	07	08	sp1	01	02
03	13	14	15	16	sp2	09	10	11
	06	07	08	sp1	01	02	03	04
04	15	16	sp2	09	10	11	12	13
	08	sp1	01	02	03	04	05	06
05	sp2	09	10	11	12	13	14	15
	01	02	03	04	05	06	07	08

Since the 3-round ID is constructed, partial key bytes of the first round-key and the 8th round-key can be recovered. The procedure of ID attack on 3.75-round SAFER SK-64 is as follows:

- Make a joint list of 2^{40} possible key for key bytes $(K_8^2, K_8^3, K_8^6, K_8^7, K_1^2)$, i.e. the 1st, 2nd, 4th, 5th and sp1 bytes of the extended key (see Table 3). It needs 2^{40} bits memory.
- Choose about 2^{37} structures. Each structure has 2^8 plaintexts, in which the 2nd byte takes all possible values and other bytes are identical.
- For 2^{37} structures, there are about $2^{37}(2^8 * (2^8 - 1)/2)/2 = 2^{51}$ plaintext pairs (P, P') that can produce the difference $(0,T/1-T,0,0,0,0,0,0)$ (see Proposition 5).
- From the 2^{51} plaintext pairs, collect about $2^{51}/2^{64-4*8} = 2^{19}$ plaintext/ciphertext pairs $(P, C; P', C')$ with the 1st, 4th, 5th, 8th bytes of the ciphertext difference being zero.
- For such a pair, compute two candidates for K_1^2, which lead to the difference $(0,T/1-T,0,0,0,0,0,0)$ at the input of S-BOX in round 1 (see Proposition 5). The computation complexity and memory complexity can be neglected.
- For each such pair, partially decrypt the 2nd, 3rd, 6th and 7th bytes of C and C' across the LKM layer and S-BOX layer using 2^{32} possible subkeys $(K_8^2, K_8^3, K_8^6, K_8^7)$. Collect about 2^{26} possible 32-bit subkeys, which cause the differences $(0, a, b, 0, 0, c, d, 0)$ at the output of the UKM layer in round 4, where (a, b, c, d) satisfy one of the conditions in (1).
- One such pair may suggest about 2^{27} incorrect 40-bit subkeys. These subkeys cannot be the correct values because they lead a text pair to the impossible differential. So delete them from the list of key candidates.This may take about $4 * (2^9) + (2^{27}/8)/2 < 2^{23}$ half round decryption and memory complexity is $4 * 2^8$ bytes.
- Using 2^{19} right plaintext pairs, the expected number of the remaining wrong subkeys is:

$$2^{40}(1 - 2^{27}/2^{40})^{2^{19}} = 2^{40}(1 - 2^{-13})^{2^{19}} = 2^{40} * e^{-2^6} < 1.$$

Thus, the correct subkey can be uniquely identified.

- The 2^{19} pairs correspond to 2^{45} chosen plaintexts (2^{37} structures). The memory complexity is $2^{29} + 2^{40}/8 < 2^{38}$ bytes for all the steps. And the time complexity is equivalent to $2^{23} * 2^{19} = 2^{42}$ half round SAFER SK computations.
- The remaining $64 - 40 = 24$ bits can be recovered by exhaustive search.

In the case of the attack on SAFER SK -128, five key bytes $(1, 4, 5, sp1, 10)$ (see Table 4) can be recovered using the above ID, and the memory complexity and the computation complexity are same as that of the above attack.

3.2. Impossible differential attack on SAFER+/128

The ID cryptanalysis on SAFER+ uses the impossible differential:

$$(0,0,0,0,0,0,0,0,0,0,0,0,0,0,T/1-T,0) \leftrightarrow (0,a,b,0,0,c,d,0,0,e,f,0,0,g,h,0).$$

Here, the vector (a,b,c,d,e,f,g,h) meets one of the following five conditions:

$$\begin{aligned}
&4a-b+4c-d+2e-f+4g-8h \equiv 0 \pmod{256},\\
&-8a+b-c+d-2e+2f-2g+h \equiv 0 \pmod{256},\\
&-2a+b-2c+2d-8e+f-g+h \equiv 0 \pmod{256},\\
&-a+2b-8c+d-2e+f-2g+h \equiv 0 \pmod{256},\\
&2a-2b+16c-d+4e-f+4g-h \equiv 0 \pmod{256}.
\end{aligned} \tag{2}$$

The above part (in the encryption direction) starts at the input of the S-BOX layer in the first round and outputs the difference $(*,*,ev,od,ev,*,od,ev,od,*,ev,ev,od,od,*,*)$ after round 2. The nether part (in the decryption direction) inputs difference $(0,a,b,0,0,c,d,0,0,e,f,0,0,g,h,0)$ at the output of the UKM layer. Similarly, different conditions among a,b,c,d,e,f,g,h result in different output differences. In Fig. 2, we suppose the first condition in (2) holds. Then, the resulted difference at the input of UKM layer in round 3 is $(*,*,*,ev,*,*,*,*,*,*,*,*,*,*,*,*)$. So contradiction occurs at the 4th byte, i.e. according to the above part, the 4th byte below round 2 should be odd, but the corresponding byte of the nether part is even.

Corresponding to other relations among (a,b,c,d,e,f,g,h), the contradiction separately arises at the 7th, 9th, 13th or 14th byte of the difference after the second round.

Using the 3-round impossible differential, we can analyze 3.75-round cipher. Firstly, one byte of the first round-key and 4 bytes of the 8th round-key, i.e. the 1st, 4th, 5th, 9th, 10th, 13th, 14th, 15th and sp1 bytes of the extended key (see Table 5), can be recovered. Then the remaining 56-bit subkey can be recovered by exhaustive search. The detail attack steps are same as those on 3.75-round SAFER SK-64. The attack needs 2^{78} chosen plaintexts (2^{70} structures) and the memory complexity is $2^{11} * 2^{20} + 2^{72}/8 < 2^{68}$ bytes for all the steps and the list of possible keys. The time complexity is equivalent to $2^{55} * 2^{20} = 2^{75}$ half round SAFER+/128 computations.

For the attack on SAFER+/256, nine key bytes $(15,17,20,21,25,26,29,30,sp2)$ (see Table 6) can be recovered using the same ID, meanwhile the memory complexity and the computation complexity are same to that of the attack on SAFER+/128.

Table 5
How key bytes are used in first five rounds of SAFER+/128.

Round	Key bytes															
01	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16
	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	sp1
02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	sp1	01
	04	05	06	07	08	09	10	11	12	13	14	15	16	sp1	01	02
03	05	06	07	08	09	10	11	12	13	14	15	16	sp1	01	02	03
	06	07	08	09	10	11	12	13	14	15	16	sp1	01	02	03	04
04	07	08	09	10	11	12	13	14	15	16	sp1	01	02	03	04	05
	08	09	10	11	12	13	14	15	16	sp1	01	02	03	04	05	06
05	09	10	11	12	13	14	15	16	sp1	01	02	03	04	05	06	07
	10	11	12	13	14	15	16	sp1	01	02	03	04	05	06	07	08

Table 6
How key bytes are used in first five rounds of SAFER+/256.

Round	Key bytes															
01	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16
	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	sp2
02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	sp1	01
	20	21	22	23	24	25	26	27	28	29	30	31	32	sp2	17	18
03	05	06	07	08	09	10	11	12	13	14	15	16	sp1	01	02	03
	22	23	24	25	26	27	28	29	30	31	32	sp2	17	18	19	20
04	07	08	09	10	11	12	13	14	15	16	sp1	01	02	03	04	05
	24	25	26	27	28	29	30	31	32	sp2	17	18	19	20	21	22
05	09	10	11	12	13	14	15	16	sp1	01	02	03	04	05	06	07
	26	27	28	29	30	31	32	sp2	17	18	19	20	21	22	23	24

3.3. Impossible differential attack on SAFER++/128

The ID cryptanalysis of SAFER++ uses the following ID:

$$(a,0,0,b,c,0,0,d,e,0,0,f,g,0,0,h)\leftrightarrow(0,0,0,0,T/1-T,0,0,0,0,0,0,0,0,0,0,0).$$

In the encryption direction, the difference $(a,0,0,b,c,0,0,d,e,0,0,f,g,0,0,h)$ begins at the input of the LKM layer in the first round, where the vector (a,b,c,d,e,f,g,h) meets one of the following three conditions:

$$\begin{aligned} &a+b+2c+d+2e+f+g+2h\equiv 0 \pmod{256},\\ &a+2b+c+d+2e+f+2g+h\equiv 0 \pmod{256},\\ &4a+b+c+2d+e+f+g+h\equiv 0 \pmod{256}. \end{aligned} \tag{3}$$

With different condition, the difference below the LKM layer of round 2 conflicts at the 3rd, 6th or 9th byte separately. For instance, with the assumption $4a+b+c+2d+e+f+g+h=0 \bmod 256$ (see Fig. 3), the 9th byte at the output difference of the LKM layer in round 2 is even in the encryption direction, but it must be an odd number in the decryption direction.

The 3-round ID can be used to recover partial key bytes of the first round-key and the 8th round-key. The procedure of ID attack on 3.75-round SAFER++/128 is as follows:

- Make a list of all possible 64-bit subkeys $(K_1^1,K_1^4,K_1^5,K_1^8,K_1^9,K_1^{12}/K_8^5,K_1^{13},K_1^{16})$ (see Table 5). It needs 2^{64} bits memory.
- Choose about 2^{14} structures. Each structure has 2^{64} plaintexts, in which the 1st, 4th, 5th, 8th, 9th, 12th, 13th and 16th bytes take all possible values and other bytes are identical. For each structure, there are about $2^{64}(2^{64}-1)/2\approx 2^{127}$ plaintext pairs.
- For each structure, about $(2^{127}/2^{15*8})/2=2^6$ plaintext/ciphertext pairs (see Proposition 5) can cause difference $(0,0,0,0,T/1-T,0,0,0,0,0,0,0,0,0,0,0)$. Moreover, for each such pair, there are only two choice for K_8^5 to generate difference $T/1-T$, and they can be computed immediately.
- Try all $2^{56}*2$ (K_8^5 is the transformation of K_1^{12}, so there are only two possible values) possible subkeys $(K_1^1,K_1^4,K_1^5,K_1^8,K_1^9,K_1^{12},K_1^{13},K_1^{16})$ to encrypt the related byte pairs of plaintext pair (P,P') across the UKM layer. Collect about 2^{50} possible 64-bit subkeys that cause the difference $(a,0,0,b,c,0,0,d,e,0,0,f,g,0,0,h)$. Here, the vector (a,b,c,d,e,f,g,h) satisfies one of the three conditions in (3).
- Each right pair suggests a list of about 2^{50} incorrect 64-bit subkeys.This may take about $7*(2^9)+(2^{50}/16)/2<2^{46}$ half round computation and memory complexity is 2^8*7 bytes.
- Using 2^{20} such pairs, the number of the remaining wrong subkeys is:

$$2^{64}(1-2^{50}/2^{64})^{2^{20}}=2^{64}(1-2^{-14})^{2^{20}}=2^{64}e^{-2^6}<1.$$

  So, the correct subkey can be uniquely identified.
- The 2^{20} pairs correspond to $2^{14}*2^{64}=2^{78}$ chosen plaintexts. The memory complexity is $2^{11}*2^{20}+2^{64}/8<2^{62}$ bytes and the time complexity is equivalent to $2^{46}*2^{20}=2^{66}$ half round SAFER++/128 computation.
- The remaining $128-64=64$ bits can be recovered by exhaustive search.

In the case of the attack on SAFER++/256, nine key bytes (1, 4, 5, 8, 9, 12, 13, 16, 28) (see Table 6) can be recovered using the same ID. The memory complexity is 2^{70} bytes and the computation complexity is 2^{74} half round SAFER++/256 computation.

4. Conclusions

In this paper, we construct new impossible differentials of SAFER ciphers. In the middle of these IDs, some output bytes of the above differential (in the encryption direction) and the corresponding output bytes of the nether differential (in the decryption direction) have different parity. With these IDs, the result of the impossible differential attacks on SAFER ciphers are improved.

Acknowledgments

This work is supported by National Basic research Program of China (973 program) (No. 2007CB310704, and No. 2007CB807902), National Natural Science Foundation of China (No. 90718001, No. 60821001, and No. 60973159), and National 863 (No. 2009AA012439).

References

[1] Knudsen L. DEAL – a 128-bit block cipher. NIST AES proposal. Technical report #151, February 21; 1998 [retrieved 27.02.07].
[2] Biham E, Biryukov A, Shamir A. Cryptanalysis of skipjack reduced to 31 rounds using impossible differentials. In: Stern J, editor. Advances in cryptology – Eurocrypt'99. LNCS, vol. 1592. Berlin, Heidelberg: Springer-Verlag; 1999. p. 12–23.

[3] Biryukov A, De Canniere C, Dellkrantz G. Cryptanalysis of SAFER++. In: Boneh D, editor. Advances in cryptology – CRYPTO'03. LNCS, vol. 2729. Berlin, Heidelberg: Springer-Verlag; 2003. p. 195–211.
[4] Knudsen L. Truncated and higher order differentials. In: Preneel B, editor. Proceedings of fast software encryption – FSE'94. LNCS, vol. 1008. Berlin, Heidelberg: Springer-Verlag; 1995. p. 196–211.
[5] Knudsen L. A detailed analysis of SAFER K. Journal of Cryptology 2000;13(4):417–36.
[6] Massey JL. SAFER K-64: a Byte-oriented block-ciphering algorithm. In: Anderson R, editor. First fast software encryption workshop. LNCS, vol. 809. Berlin, Heidelberg: Springer-Verlag; 1994. p. 1–17.
[7] Massey JL. SAFER K-64: one year later. In: Preneel B, editor. Fast software encryption, second international workshop. LNCS, vol. 1008. Berlin, Heidelberg: Springer-Verlag; 1995. p. 212–41.
[8] Massey JL. Strengthened key schedule for the cipher SAFER, posted to the USENET newsgroup sci.crypt; September 1995.
[9] Massey JL, Khachatrian GH, Kuregian MK. In: 1st AES conference on nomination of SAFER+ as candidate algorithm for the advanced encryption standard. California, USA; June 1998, <http://csrc.nist.gov/encryption/aes/>.
[10] Massey JL, Khachatrian GH, Kuregian MK. In: 1st NESSIE workshop on the SAFER++ block encryption algorithm. Heverlee, Belgium; November 2000, http://cryptonessie.org.
[11] Nakahara J, Preneel B, Vandewalle J. Impossible differential attacks on reduced-round SAFER ciphers. COSIC technical report; November 2003.
[12] Nakahara J. Cryptanalysis and design of block ciphers. PhD thesis. Katholieke University, Leuven; June 2003.
[13] Nakahara J, Preneel B, Vandewalle J. Linear cryptanalysis of reduced round versions of the SAFER block cipher family. In: Schneier B, editor. Proceedings of fast software encryption – FSE'00. LNCS, vol. 1978. Berlin, Heidelberg: Springer-Verlag; 1998. p. 244–61.
[14] Murphy S. An analysis of SAFER. J Cryptol 2000;11(4):235–51.
[15] Massey JL. On the optimality of SAFER+ diffusion. In: Proceedings of the second AES candidate conference, NIST; March 1999.
[16] Kelsey J, Schneier B, Wagner D. Key-schedule cryptanalysis of 3-WAY, IDEA, G-DES, RC4, SAFER, and Triple-DES. In: Koblitz N, editor. Advances in cryptology-CRYPTO'96. LNCS, vol. 1109. Berlin, Heidelberg: Springer-Verlag; 1996. p. 237–51.
[17] Piret G, Quisquater J. Integral cryptanalysis on reduced-round SAFER++ – a way to extend the attack? <http://eprint.iacr.org/2003/033.pdf>.
[18] Knudsen L, Berson T. Truncated differentials of SAFER. In: Gollmann D, editor. Proceedings of fast software encryption – FSE'00. LNCS, vol. 1039. Berlin, Heidelberg: Springer-Verlag; 1996. p. 15–26.
[19] Wu H, Bao F, Deng RH, Ye QZ. Improved truncated differential attacks on SAFER. In: Ohta K, Pei D, editors. Advances in cryptology – Asiacrypt'98. LNCS, vol. 1514. Berlin, Heidelberg: Springer-Verlag; 1998. p. 133–47.
[20] Knudsen L. A key-schedule weakness in SAFER K-64. In: Coppersmith D, editor. Advances in cryptology – CRYPTO'96. LNCS, vol. 963. Berlin, Heidelberg: Springer-Verlag; 1995. p. 274–86.
[21] Kelsey J, Schneier B, Wagner D. Key schedule weaknesses in SAFER+. In: Proceedings 2nd advanced encryption standard candidate conference; March 1999.
[22] Brincat K, Meijer A. On the SAFER cryptosystem. In: Darnell M, editor. Cryptography and coding, proceedings of 6th IMA conference. LNCS, vol. 1355. Berlin, Heidelberg: Springer-Verlag; 1995. p. 59–68.
[23] Vaudenay S. On the need for multipermutations: cryptanalysis of MD4 and SAFER. In: Gollmann D, editor. Proceedings of fast software encryption – FSE'00. LNCS, vol. 1039. Berlin,Heidelberg: Springer-Verlag; 1996. p. 286–97.
[24] Harpes C, Kramer G, Massey J. A generalization of linear cryptanalysis and the applicability of Matsui's piling-up lemma. In: Guillou L, Quisquater J, editors. Advances in cryptology – Eurocrypt'95. LNCS, vol. 1592. Berlin, Heidelberg: Springer-Verlag; 1995. p. 24–38.
[25] Yemo Y, Park I. Optimization of integral cryptanalysis on reduced-round SAFER++. Joho Shori Gakkai Shinpojiumu Ronbunshu (published in Japan). 2003(15):223–7;2003.

Shihui Zheng received his Ph.D degree in Information Security from Shandong University, Ji'nan, PR China in 2006. She is now a lecturer in the School of Computer Science, Beijing University of Posts and Telecommunications, Beijing, PR China. Her recent research interests include designing and analysis of cryptographic schemes.

Licheng Wang received his Ph.D degree in Information Security from Shanghai Jiaotong University, Shanghai, PR China in 2007. He is now a lecturer in the School of Computer Science, Beijing University of Posts and Telecommunications, Beijing, PR China. His recent research interests include designing and analysis of cryptographic schemes.

Yixian Yang received his Ph.D degree in Communication system from Beijing University of Posts and Telecommunications, Beijing, PR China in 1988. He is now a professor in the School of Computer Science, Beijing University of Posts and Telecommunications, Beijing, PR China. His recent research interests include cryptography and computer security.