A New Related-Key Boomerang Distinguishing Attack of Reduced-Round Threefish-256

Shusheng Liu, Libin Wang and Zheng Gong
School of Computer Science,
South China Normal University, Guangzhou 510631, China
{*shushengliu914,lbwang,cis.gong*}*@gmail.com*

Abstract. On Nov 2007, NIST announced the SHA-3 competition to select a new hash standard as a replacement of SHA-2. On Dec 2010, five submissions have been selected as the final round candidates, including Skein, which have components based on ARX. In this paper, a new related-key boomerang distinguishing attack is proposed on 31-round Threefish-256 with a time complexity of about 2^{234}. Our improved attack is based on the efficient algorithms for calculating differentials of modular addition.

Keywords: Skein, Differential analysis, Related key, Boomerang attack.

1 Introduction

In cryptology, hash functions are designed to protect data integrity by producing an fixed-length digest from an arbitrary-length message. Based on Wang *et al.*'s breakthrough in hash cryptanalysis, the widely-used hash functions (MD5, SHA-1, etc.) have been seriously attacked [13, 14, 15, 16]. If SHA-256 and SHA-512 were to be broken, the industry do not have any generally-accepted hash functions. As a response to this undesirable consequence, a public competition was hold by the National Institute of Standards and Technology (NIST) to collect the new designs for a secure and applicable hash function. After two-round competitions, five algorithms [9] has been selected as the final round candidates. One of the five proposals will be chosen as the SHA-3 standard in 2012.

The hash function Skein [5], which was designed by Ferguson *et al.*, has been selected as the one of the five final-round candidates for the SHA-3 competition. The design rationale of Skein combines speed, security and simplicity. Its conservative design provides a large security margin for the resistance of cryptanalysis. In the Skein proposal, the compression function of Skein is constructed from a family of tweakable block ciphers which is called *Threefish*. The family supports three different variants called Threefish-256, 512 and 1024, which implies to Skein-256, 512 and 1024 respectively. The algorithms within Threefish are fully based on addition, exclusive-or (XOR) and constant rotation (which are called AXR operations) on 64-bit words.

In the literature [1, 11, 17, 6], many cryptanalyses have been proposed on the compression function of Skein and the underlying block cipher Threefish. Table 1 summarizes the published results on reduced-round variants of Skein-256 (or Threefish-256). Aumasson *et al.* presented a related-key boomerang distinguishing attack on

34-round Threefish-512 with the old rotation constant [1]. Su *et al.* [11] proposed a 24-round near-collision of Skein-256/512 compression functions by using linear-differential analysis. Yu *et al.* [17] presented a semi-free start near-collision attack on 32-round Skein-256 compression functions based on the rebound attack. While Khovratovich *et al.* proposed a new distinguishing attack of 53-round skein-256 and 57-round Skein-512 by using the rotational rebound attack [6].

Table 1: Summary of the known results on Skein-256/Threefish-256. Where UBI-256 denotes the compression function of Skein-256.

Cipher	Rounds	Probability	Method	Attack	Reference
Threefish-256	24	?	Related-key differential	distinguishing Key recovery	[5]
UBI-256	24	2^{-60}	Linear differential	Near-collision	[11]
UBI-256	53	2^{-244}	Rotational rebound	Rotational collision	[6]
UBI-256	32	2^{-105}	Rebound attack	Near-collision	[17]
Threefish-256	31	2^{-234}	Related-key boomerang	distinguishing	this paper

In this paper, we present an improved related-key boomerang attack on 31-round Threefish-256. Our cryptanalysis is different from the cryptanalysis of [17], which is only valid for the compression function of Skein-256 that Near-collision is obtained by using Rebound attack. In particular, our attack analyze the related-key boomerang property of the block cipher Threefish-256. The strategy behind our attack is to extend local collisions to more round by using related-key and differential of addition. In order to avoid fast increasing complexity of attack, we deal it with the boomerang attack. Based on an efficient algorithm for computing differential of modular addition, we obtain a related-key boomerang distinguishing attack of 31-round Threefish with a time complexity of 2^{234}.

The remainder of this paper is organized as follows. Section 2 describes preliminaries for out attack. Section 3 first exploits two short related-key differentials, then our improved related-key boomerang distinguishing attack of 31-round Threefish-256 is presented. Finally a conclusion is given in Section 4.

2 Preliminaries

In this section, we first define the notations used throughout this paper, then we briefly describe the related-key boomerang attack, the algorithm for computing differential of addition. Finally we recall the specification of Threefish-256.

2.1 Notations

The notations used in our cryptanalysis are described as follows.

- $+$: addition modulo 2^{64}.
- $\lll$ and $\ggg$: cyclic left and right rotations respectively.
- $\ll$ and $\gg$: shift to left and right respectively.
- $\oplus$, $\vee$, $\wedge$ and $\neg$: "XOR", "OR", "AND" and "NOT", respectively.
- K: the key of Threefish, while K_i is the i-th word of K. sk_i denotes the i-th round subkey. Moreover, $sk_{i,j}$ is the j-th word of sk_i.

- T: the tweak of Threefish, while T_i is the i-th word of tweak T.
- R_i: the i-th round of Threefish.
- Δx: the XOR difference of x and x', while ΔK_i denotes the XOR difference of the i-th word of K and $K^{'}$. $\Delta sk_{i,j}$ represents the XOR difference of the j-th word of sk_i.

2.2 Related-key boomerang attack

The related-key attack was first introduced by Biham in [2]. The attack allows the accesses to encrypt plaintexts and decrypt ciphertexts under multiple unknown keys, but the relation between the unknown key is known to (or even chosen by) the adversary. The boomerang attack was introduced by Wagner in [12]. By extending the boomerang attack in the related-key model [3], Biham *et al.* proposed the related-key boomerang attack. As shown in Figure 1, the related-key boomerang attack views a cipher E as a decomposition into two sub-ciphers, such that $E = E_\alpha \circ E_\beta$. In each of two sub-ciphers, there exists a high probability related-key differential for constructing a boomerang attack.

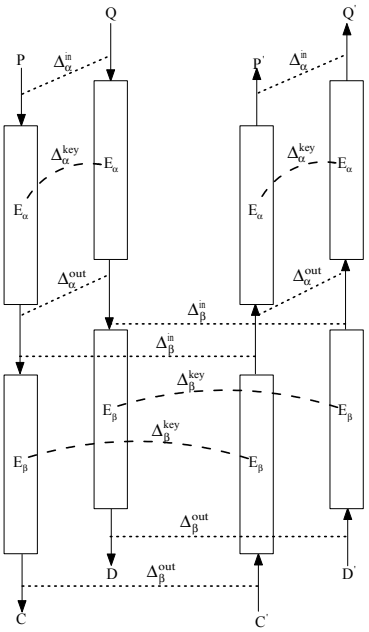

Figure 1: A schematic of related-key boomerang attack

If the probability of the E_α differential $(\Delta_\alpha^{in}, \Delta_\alpha^{out}, \Delta_\alpha^{key})$ is p and the probability of the E_β differential $(\Delta_\beta^{in}, \Delta_\beta^{out}, \Delta_\beta^{key})$ is q, it was proven that the probability of the corresponding related-key boomerang attack is close to $(p \cdot q)^2$.

2.3 The algorithm for computing differential of addition

For modular additions, efficient algorithms for computing the probability of any differential and finding optimal trails were analyzed in [7]. For S-function, a general

framework was presented in [8], which is used to calculate the probability that given input differences lead to given output differences, as well as to count the number of output differences with non-zero probability. Since the results of algorithm in [7] and algorithm in [8] for computing the probability of integer addition are identical, the algorithms in [7] are used to search the optimal differential trails of modular addition for Threefish-256. The algorithms are described as follows.

Without losing the generality, the differential of addition modulo 2^n often denotes as a triplet of two input and one output differences such that $(\alpha, \beta \mapsto \gamma), \alpha, \beta, \gamma \in \{0,1\}^n$. The differential probability of modular addition is defined as follows.

$$DP^{+}(\alpha, \beta \mapsto \gamma) \coloneqq \Pr[(x + y) \oplus ((x \oplus \alpha) + (y \oplus \beta)) = \gamma \mid x, y \in \{0,1\}^n]$$

the maximum differential probability of modular addition is defined in the following equation.

$$DP^{+}_{max}(\alpha, \beta) \coloneqq \max_{\gamma}(DP^{+}(\alpha, \beta \mapsto \gamma))$$

Algorithm 2 was introduced by Lipmaa and Moriai in [7], which calculates $DP^{+}(\alpha, \beta \mapsto \gamma)$ in a log-time. Algorithm 3 was described in [7], which finds all output differences γ that satisfies $DP^{+}(\alpha, \beta \mapsto \gamma)$ is equal to $DP^{+}_{max}(\alpha, \beta)$.

Lipmaa and Moriai also presented the definition of several functions in [7], which are used in Algorithm 2 and Algorithm 3. For any x, y and z , $eq(x, y, z) = (\neg x \oplus y) \wedge (\neg x \oplus z)$ and $xor(x, y, z) = x \oplus y \oplus z$. For any n, let $mask(n) = 2^n - 1$. The Hamming weight function $w_h(x) = \sum_{i=0}^{n-1} x_i$. The all-one parity of an n-bit number x is another n-bit number $y = aop(x)$, and $aop(x)$ is calculated by Algorithm 1. The common alternation parity of two n-bit numbers x and y is a function $C(x, y)$, such that

$$C(x, y) \coloneqq aop(\neg(x \oplus y) \wedge (\neg(x \oplus y) \gg 1) \wedge (x \oplus (x \gg 1))).$$

Algorithm 1 Log-time algorithm for $aop(x)$

Input: $x \in \{0,1\}^n$;
Output: $aop(x)$;
1: $x[1] = x \wedge (x \gg 1)$;
2: for $i \leftarrow 2$ to $log_2 n - 1$ do $x[i] \leftarrow x[i-1] \wedge (x[i-1] \gg 2^{i-1})$;
3: $y[1] \leftarrow x \wedge \neg x[1]$;
4: for $i \leftarrow 2$ to $log_2 n$ do $y[i] \leftarrow y[i-1] \vee ((y[i-1] \gg 2^{i-1}) \wedge x[i-1])$;
5: return $y[log_2 n]$;

Algorithm 2 Log-time algorithm for DP^{+}

Input: $(\alpha, \beta \mapsto \gamma)$;
Output: $DP^{+}(\alpha, \beta \mapsto \gamma)$;
1: if $eq(\alpha \ll 1, \beta \ll 1, \gamma \ll 1) \wedge ((xor(\alpha, \beta, \gamma) \oplus (\beta \ll 1))) \neq 0$ then return 0;
2: return $2^{-w_h(\neg eq(\alpha,\beta,\gamma) \wedge mask(n-1))}$;

2.4 A brief description of Threefish-256

Threefish-256 works on 64-bit words using exclusive-OR, addition modulo 2^{64} and cyclic shift. A 256-bit plaintext is parsed as four words $v_{0,0}, \ldots, v_{0,3}$, and encrypted through $N_r = 72$ rounds. Round d is from 1 up to 72, the encryption procedure of Threefish-256 operates as follows :

Algorithm 3 Algorithm that finds all γ, $DP^{+}(\alpha,\beta,\mapsto\gamma) = DP^{+}_{max}(\alpha,\beta)$

Input: (α,β);
Output: All (α,β)-optimal output differences γ;
1: $\gamma_0 \leftarrow \alpha_0 \oplus \beta_0$;
2: $p \leftarrow C(\alpha,\beta)$;
3: for $i \leftarrow 1$ to $n-1$ do
if $\alpha_{i-1} = \beta_{i-1} = \gamma_{i-1}$ then $\gamma_i \leftarrow \alpha_i \oplus \beta_i \oplus \alpha_{i-1}$
else if $i = n-1$ or $\alpha_i \neq \beta_i$ or $p_i = 1$ then $\gamma_i \leftarrow \{0,1\}$
else $\gamma_i \leftarrow \alpha_i$;
4: return γ

1. If $d \equiv 1 \ mod\ 4$, add a subkey by setting $e_{d,i} = v_{d-1,i} + k_{d/4,i}$, $i = 0,\ldots,3$. else, $e_{d,i} = v_{d-1,i}$, $i = 0,\ldots,3$.

2. $(f_{d,2i}, f_{d,2i+1}) \leftarrow MIX_{d,i}(e_{d,2i}, e_{d,2i+1})$, for $i = 0, 1$. Mix function is defined by

$$MIX_{d,i}(x,y) := (x+y, (x+y) \oplus (y \ll RC_{(d-1)\ mod\ 8,i}))$$

where $RC_{(d-1)\ mod\ 8,i}$ is a rotation constant in row $(d-1)\ mod\ 8$ column i of the rotation constant table that can be found in [5].

3. Permute the state words :

$$v_{d,0} = f_{d,0},\ v_{d,1} = f_{d,3},\ v_{d,2} = f_{d,2},\ v_{d,3} = f_{d,1}$$

After 72 rounds, the ciphertext is

$$(v_{72,0} + k_{19,0},\ \ldots,\ v_{72,3} + k_{19,3})$$

The set of subkeys is derived from the master key $K = (K_0, K_1, K_2, K_3)$ and tweak $T = (T_0, T_1)$ as follows.

$$\begin{aligned} sk_{s,0} &= K_{(s+0)\ mod\ 5} \\ sk_{s,1} &= K_{(s+1)\ mod\ 5} + T_{s\ mod\ 3} \\ sk_{s,2} &= K_{(s+2)\ mod\ 5} + T_{(s+1)\ mod\ 3} \\ sk_{s,3} &= K_{(s+3)\ mod\ 5} + s \end{aligned}$$

where $K_4 = \lfloor 2^{64}/3 \rfloor \oplus K_0 \oplus K_1 \oplus K_2 \oplus K_3$, $T_2 = T_0 \oplus T_1$ and s is the value of the s-th round.

3 The proposed attack

In this section, we describe how to build a related-key boomerang of Threefish-256 . E_α is viewed as the sub-cipher of the first 17 rounds of Threefish-256, and E_β is viewed as the sub-cipher of the following 14 rounds (18 to 31) of Threefish-256. We obtain a related-key differential of E_α with a probability 2^{-99} and a related-key differential of E_β with a probability 2^{-18}. Thus the boomerang distinguishing attack that makes use of E_α and E_β has a probability 2^{-234}. The details of the attack will be depicted in the following subsections.

3.1 The subkeys differential

Following the key schedule of Threefish-256, one can get all subkeys from an encryption key. Table 2 illustrates an overview of eight subkeys, which will be used in the first 32-round Threefish-256. The number i denotes the round constant. The subkeys differential of E_α and E_β are searched for the related-keys boomerang distinguishing attack.

Table 2: The first eight subkeys of the Threefish-256 key schedule

	word 0	word 1	word 2	word 3
sk_0	K_0	$K_1 + T_0$	$K_2 + T_1$	$K_3 + 0$
sk_1	K_1	$K_2 + T_1$	$K_3 + T_2$	$K_4 + 1$
sk_2	K_2	$K_3 + T_2$	$K_4 + T_0$	$K_0 + 2$
sk_3	K_3	$K_4 + T_0$	$K_0 + T_1$	$K_1 + 3$
sk_4	K_4	$K_0 + T_1$	$K_1 + T_2$	$K_2 + 4$
sk_5	K_0	$K_1 + T_2$	$K_2 + T_0$	$K_3 + 5$
sk_6	K_1	$K_2 + T_0$	$K_3 + T_1$	$K_4 + 6$
sk_7	K_2	$K_3 + T_1$	$K_4 + T_2$	$K_0 + 7$
sk_8	K_3	$K_4 + T_2$	$K_0 + T_0$	$K_1 + 8$

- **Subkeys differential of E_α.** For a pair of key and tweak, their difference patterns are chosen for the related-key differential of E_α as follows.

$$((K_0, K_1, K_2, K_3), (T_0, T_1)) \neq ((K_0^{'}, K_1^{'}, K_2^{'}, K_3^{'}), (T_0^{'}, T_1^{'})).$$

Hence the difference in the sk_2 is eliminated, which implies

$$K_2 = K_2^{'},\ K_3 + T_2 = K_3^{'} + T_2^{'},\ K_4 + T_0 = K_4^{'} + T_0^{'},\ K_0 + 2 = K_0^{'} + 2.$$

Let the difference $\delta = 0x8000000000000000$, where the most significant bit is isolated. One set difference of key/tweak pair can be represented as follows.

$$K_2 \oplus K_2^{'} = 0,\ K_0 \oplus K_0^{'} = 0,\ K_1 \oplus K_1^{'} = 0,$$
$$K_3 \oplus K_3^{'} = \delta,\ T_0 \oplus T_0^{'} = \delta, T_1 \oplus T_1^{'} = 0.$$

Under condition $\Delta_\alpha^{key} = ((\Delta K_0\ \Delta K_1\ \Delta K_2\ \Delta K_3), (\Delta T_0\ \Delta T_1)) = ((0\ 0\ 0\ \delta), (\delta\ 0))$, $\Delta sk_1 = (0, 0, 0, \delta)$. The difference of the i-th subkeys($0 \leq i \leq 4$) are shown in Table 3.

Table 3: Subkey's differences of the Threefish-256 key

	subkeys differential of E_α
Δsk_0	$(0, \delta, 0, \delta)$
Δsk_1	$(0, 0, 0, \delta)$
Δsk_2	$(0, 0, 0, 0)$
Δsk_3	$(\delta, 0, 0, 0)$
Δsk_4	$(\delta, 0, \delta, 0)$
	subkeys differential of E_β
Δsk_5	$(0, 0, 0, \delta)$
Δsk_6	$(0, 0, 0, 0)$
Δsk_7	$(\delta, 0, 0, 0)$
Δsk_8	$(\delta, 0, \delta, 0)$

- **Subkeys differential of E_β.** A difference of key/tweak pair is chosen for the related-key differential of E_β such that $\Delta sk_6 = (0, 0, 0, 0)$. this implies

$$K_1 = K_1^{'},\ K_2 + T_0 = K_2^{'} + T_0^{'},\ K_3 + T_1 = K_3^{'} + T_1^{'},\ K_4 + 6 = K_4^{'} + 6.$$

  One can set $\Delta_\beta^{key} = ((\Delta K_0\ \Delta K_1\ \Delta K_2\ \Delta K_3), (\Delta T_0\ \Delta T_1)) = ((0\ 0\ \delta\ \delta), (\delta\ \delta))$ to obtain $\Delta sk_6 = (0, 0, 0, 0)$. In this case, $\Delta sk_5 = (0, 0, 0, \delta)$. The difference of the i-th subkeys($5 \le i \le 8$) are given in Table 3.

3.2 The E_α differential

In this section, we search the related-key differential of E_α. Firstly, we assign the output difference of R_4 to $\Delta sk_1 = (0, 0, 0, \delta)$, then we compute the backward related-key differential from R_4 to R_1. Secondly, we investigate the forward related-key differential from R_{13} to R_{17}.

- **Trail of R_1 to R_4.** In order to reach the output difference of R_4 $(0, 0, 0, \delta)$, we need to reverse the difference of R_4 to the input difference sk_0. In Figure 2, the numbers connected with the arrows are the Hamming weight of the differences. Δsk_1 is assigned to the output difference of R_4. The input difference of sk_0 is calculated from Δsk_1 in backward. The input difference of round 4 is described as follows.

$$\begin{aligned}
\Delta v_{4,1} &= (\Delta sk_{1,0} \oplus \Delta sk_{1,3}) \ggg 5 \\
\Delta v_{4,0} &= \Delta v_{4,1} \oplus \Delta sk_{1,0} \\
\Delta v_{4,3} &= (\Delta sk_{1,2} \oplus \Delta sk_{1,1}) \ggg 37 \\
\Delta v_{4,2} &= \Delta v_{4,3} \oplus \Delta sk_{1,2}.
\end{aligned}$$

  $\Delta v_{i,j}$ is the input difference of the j-th word of the i-th round, while the corresponding rotation constants (5 and 37) are used in the R_4 of Threefish-256. The input difference of E_α $(\Delta v_{0,0}, \Delta v_{0,1}, \Delta v_{0,2}, \Delta v_{0,3})$ with the effect of sk_0 is computed in the following equations.

$$\begin{aligned}
\Delta v_{0,0} &= \Delta sk_{0,0} \oplus \Delta v_{1,0} \\
\Delta v_{0,1} &= \Delta sk_{0,1} \oplus \Delta v_{1,1} \\
\Delta v_{0,2} &= \Delta sk_{0,2} \oplus \Delta v_{1,2} \\
\Delta v_{0,3} &= \Delta sk_{0,3} \oplus \Delta v_{1,3}
\end{aligned}$$

  The first 4-round related-key differential of E_α is listed in Table 4.

- **Trail of R_5 to R_{12}.** The sk_1 adds difference $(0, 0, 0, \delta)$ to the output difference of R_4 so that its difference is vanished. The state of difference remains $(0, 0, 0, 0)$ until the sk_3 is added. After the effect of sk_3, the value of the output difference is $(\delta, 0, 0, 0)$. Figure 3 illustrates the trail.

- **Trail of R_{13} to R_{17}.** In this step, the related-key differential from R_{13} to R_{17} is calculated by equation 1 ($13 \le i \le 17$) when the input difference of R_{13} is $(\delta, 0, 0, 0)$.

$$\begin{aligned}
\Delta v_{i+1,0} &= \mathit{Algorithm}\ 3(\Delta v_{i,0}, \Delta v_{i,1}) \\
\Delta v_{i+1,2} &= \mathit{Algorithm}\ 3(\Delta v_{i,2}, \Delta v_{i,3}) \\
\Delta v_{i+1,1} &= \Delta v_{i+1,2} \oplus (\Delta v_{i,3} \lll RC_{(i-1)\ mod\ 8,1}) \\
\Delta v_{i+1,3} &= \Delta v_{i+1,0} \oplus (\Delta v_{i,1} \lll RC_{(i-1)\ mod\ 8,0})
\end{aligned} \tag{1}$$

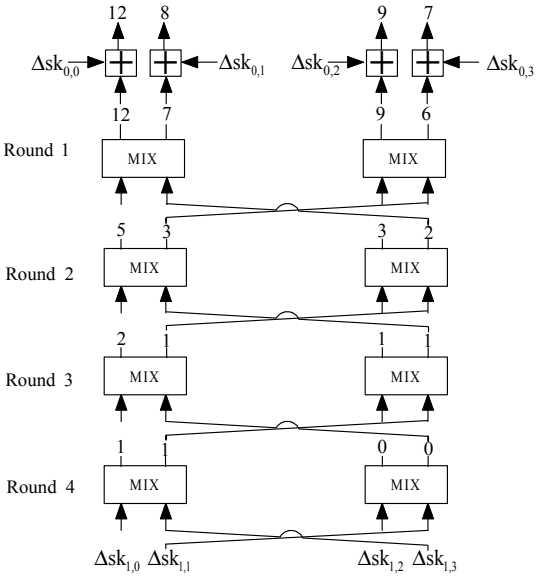

Figure 2: Backward differential from R_4 to R_1. The numerals are the number of bit 1 of corresponding difference.

Table 4: Differential trail of E_α (Differences are described in hexadecimal basis)

round	input difference	Pr
sk_0	0500900A50210840 8100100210210800 0040040086044204 8040000084004204	2^{-34}
1	0500900A50210840 0100100210210800 0040040086044204 0040000084004204	2^{-21}
2	0400800840000040 0000800040000040 0000040002040000 0000040002000000	2^{-8}
3	0400000800000000 0000000800000000 0000000000040000 0000000000040000	2^{-3}
4	0400000000000000 0400000000000000 0000000000000000 0000000000000000	2^{-1}
sk_1	0000000000000000 0000000000000000 0000000000000000 8000000000000000	1
5-12	0000000000000000 0000000000000000 0000000000000000 0000000000000000	1
13	8000000000000000 0000000000000000 0000000000000000 0000000000000000	1
14	8000000000000000 0000000000000000 0000000000000000 8000000000000000	1
15	8000000000000000 8000000000000800 8000000000000000 8000000000000000	2^{-1}
16	0000000000000800 0000000000200000 0000000000000000 0200000000000820	2^{-5}
sk_4	0000000000200800 0200082002000820 0200000000000820 0020000000200800	2^{-14}
17	8000000000200800 0200082002000820 8200000000000820 0020000000200800	2^{-12}
-	8200082002200020 8220002008200000 8220000000200020 800808A0002800A0	-

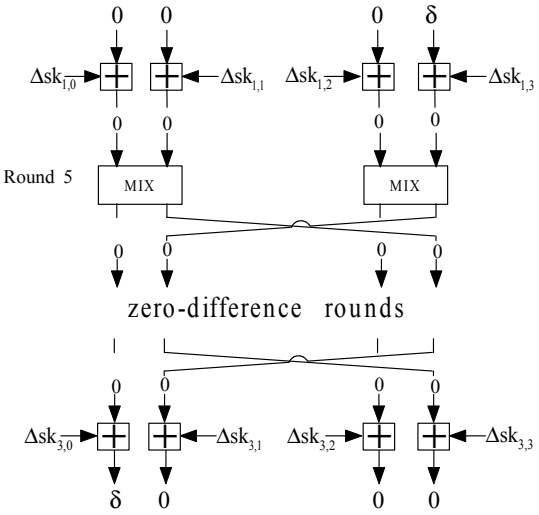

Figure 3: Forward differential from R_5 to R_{12}(including sk_1 to sk_3). The numerals are the number of bit 1 of corresponding difference.

Table 5: Differential trail of E_β (Differences are described in hexadecimal basis.

round	input difference	Pr
18	0400800840000040 0000800040000040 0000040002040000 0000040002000000	2^{-8}
19	0400000800000000 0000000800000000 0000000000040000 0000000000040000	2^{-3}
20	0400000000000000 0400000000000000 0000000000000000 0000000000000000	2^{-1}
sk_5	0000000000000000 0000000000000000 0000000000000000 8000000000000000	1
21-28	0000000000000000 0000000000000000 0000000000000000 0000000000000000	1
29	8000000000000000 0000000000000000 0000000000000000 0000000000000000	1
30	8000000000000000 0000000000000000 0000000000000000 8000000000000000	1
31	8000000000000000 8000000000000800 8000000000000000 8000000000000000	2^{-1}
sk_8	0000000000000800 0000000000200000 0000000000000000 0200000000000820	2^{-5}
-	8000000000000800 0000000000200000 8000000000000000 0200000000000820	-

Figure 4 illustrates the trail, while its patterns are also given in Table 4.

3.3 The E_β Differential

Similar to the method of searching the differential of E_α, the differential of E_β is computed as follows.

- **Trail of R_{18} to R_{20}.** In order to reach difference $(0, 0, 0, \delta)$ in the output of R_{20}, the input difference of R_{18} is computed in backward. The 3-round differential is shown in Table 5.
- **Trail of R_{21} to R_{28}.** Since sk_5 adds the output difference of R_{20} $(0, 0, 0, \delta)$, the difference will be vanished until the sk_7 is added. The difference becomes $(\delta, 0, 0, 0)$ because the effect of sk_7.
- **Trail of R_{29} to R_{31}.** In this step, we use the equation 1 ($29 \leq i \leq 31$) to calculate

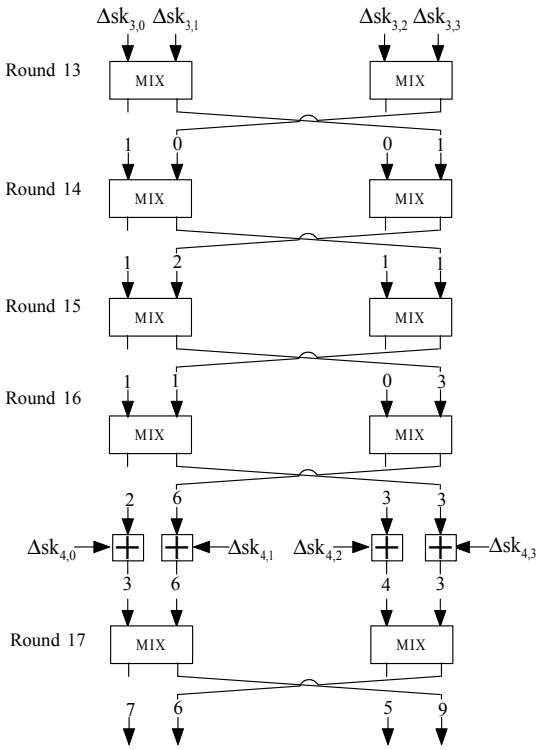

Figure 4: Forward differential from R_{13} to R_{17}. The numerals are the number of bit 1 of corresponding difference.

the differential from R_{29} to R_{32}. The input difference of R_{29} is $(\delta, 0, 0, 0)$. Also the patterns of the trail is shown in Table 5.

Based on the related-key differential of E_α and E_β, the value of there differences in Figure 1 are derived as follows.

$$\begin{aligned}
\Delta_\alpha^{in} &= 0500900A502108408100100210210800004004008604420480400000084004204\\
\Delta_\alpha^{out} &= 8200082002200020822000200820000082200000002000208008080A0002800A0\\
\Delta_\beta^{in} &= 0400800840000040000080004000004000000400020400000000040002000000\\
\Delta_\beta^{out} &= 8000000000008000000000000200000800000000000000000200000000000820
\end{aligned}$$

Therefore we obtain a differential $(\Delta_\alpha^{in}, \Delta_\alpha^{out}, \Delta_\alpha^{key})$ of E_α with the probability 2^{-99} and a differential $(\Delta_\beta^{in}, \Delta_\beta^{out}, \Delta_\beta^{key})$ of E_β with the probability 2^{-18}.

3.4 The related-key boomerang distinguishing attack and the Complexity of Computation

The related-key boomerang distinguishing attack of 31-round Threefish-256 that exploits $(\Delta_\alpha^{in}, \Delta_\alpha^{out}, \Delta_\alpha^{key})$ of E_α and $(\Delta_\beta^{in}, \Delta_\beta^{out}, \Delta_\beta^{key})$ of E_β has a probability 2^{-234}. The distinguisher works as follows:

1. Chooses a random message P and calculates $Q = P \oplus \Delta_\alpha^{in}$.
2. Encrypts P and Q, obtain $C = E_k(P)$ and $D = E_{k \oplus \Delta_\alpha^{key}}(Q)$.
3. Sets $C' = C \oplus \Delta_\beta^{out}$ and $D' = D \oplus \Delta_\beta^{out}$.
4. Decrypts C' and D', obtains $P' = E^{-1}_{k \oplus \Delta_\beta^{key}}(C')$ and $Q' = E^{-1}_{k \oplus \Delta_\alpha^{key} \oplus \Delta_\beta^{key}}(D')$.
5. Checks if $P' \oplus Q' = \Delta_\alpha^{in}$.

For an ideal cipher, the probability of $P' \oplus Q' = \Delta_\alpha^{in}$ is expected to be 2^{-256}. On the other hand, the final equation is expected to hold with probability $(2^{-99} \times 2^{-18})^2 = 2^{-234}$ in the related-key boomerang distinguisher, which is apparently lower than exhaustive search. Therefore, an adversary can distinguish between 31-round Threefish-256 and an ideal cipher by implementing our improved boomerang attack.

4 Conclusion

In this paper, we have proposed a new related-key boomerang distinguishing attack on a reduced-round variant of Threefish-256. By combining two short differentials that we have found, our boomerang attack can be used to distinguish 31-round Threefish with the time complexity of 2^{234}. Since Threefish is the primitive of the Skein, our analysis will be useful to further cryptanalyses of Skein for the SHA-3 competition.

References

[1] Jean-Philippe Aumasson, Çagdas Çalik, Willi Meier, Onur Özen, Raphael C.-W. Phan, and Kerem Varici. Improved cryptanalysis of skein. In Mitsuru Matsui, editor, *ASIACRYPT*, volume 5912 of *Lecture Notes in Computer Science*, pages 542–559. Springer, 2009.

[2] Eli Biham. New types of cryptanalytic attacks using related keys. *J. Cryptology*, 7(4):229–246, 1994.

[3] Eli Biham, Orr Dunkelman, and Nathan Keller. Related-key boomerang and rectangle attacks. In Cramer [4], pages 507–525.

[4] Ronald Cramer, editor. *Advances in Cryptology - EUROCRYPT 2005, 24th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Aarhus, Denmark, May 22-26, 2005, Proceedings*, volume 3494 of *Lecture Notes in Computer Science*. Springer, 2005.

[5] Niels Ferguson, Stefan Lucks, Bruce Schneier, Doug Whiting, Mihir Bellare, Tadayoshi Kohno, Jon Callas, and Jesse Walker. The skein hash function family.

[6] Dmitry Khovratovich, Ivica Nikolic, and Christian Rechberger. Rotational rebound attacks on reduced skein. In Masayuki Abe, editor, *ASIACRYPT*, volume 6477 of *Lecture Notes in Computer Science*, pages 1–19. Springer, 2010.

[7] Helger Lipmaa and Shiho Moriai. Efficient algorithms for computing differential properties of addition. In Mitsuru Matsui, editor, *FSE*, volume 2355 of *Lecture Notes in Computer Science*, pages 336–350. Springer, 2001.

[8] Nicky Mouha, Vesselin Velichkov, Christophe De Cannière, and Bart Preneel. The differential analysis of s-functions. In Alex Biryukov, Guang Gong, and Douglas R. Stinson, editors, *Selected Areas in Cryptography*, volume 6544 of *Lecture Notes in Computer Science*, pages 36–56. Springer, 2010.

[9] National Institute of Standards and Technology. Announcing request for candidate algorithm nominations for a new cryptographic hash algorithm (sha-3) family. 2 Nov 2007,.

[10] Victor Shoup, editor. *Advances in Cryptology - CRYPTO 2005: 25th Annual International Cryptology Conference, Santa Barbara, California, USA, August 14-18, 2005, Proceedings*, volume 3621 of *Lecture Notes in Computer Science*. Springer, 2005.

[11] Bozhan Su, Wenling Wu, Shuang Wu, and Le Dong. Near-collisions on the reduced-round compression functions of skein and blake. In Swee-Huay Heng, Rebecca N. Wright, and Bok-Min Goi, editors, *CANS*, volume 6467 of *Lecture Notes in Computer Science*, pages 124–139. Springer, 2010.

[12] David Wagner. The boomerang attack. In Lars R. Knudsen, editor, *FSE*, volume 1636 of *Lecture Notes in Computer Science*, pages 156–170. Springer, 1999.

[13] Xiaoyun Wang, Xuejia Lai, Dengguo Feng, Hui Chen, and Xiuyuan Yu. Cryptanalysis of the hash functions md4 and ripemd. In Cramer [4], pages 1–18.

[14] Xiaoyun Wang, Yiqun Lisa Yin, and Hongbo Yu. Finding collisions in the full sha-1. In Shoup [10], pages 17–36.

[15] Xiaoyun Wang and Hongbo Yu. How to Break MD5 and Other Hash Functions. In Cramer [4], pages 19–35.

[16] Xiaoyun Wang, Hongbo Yu, and Yiqun Lisa Yin. Efficient collision search attacks on sha-0. In Shoup [10], pages 1–16.

[17] Hongbo Yu, Jiazhe Chen, Ketingjia, and Xiaoyun Wang. Near-collision attack on the step-reduced compression function of skein-256. Cryptology ePrint Archive, Report 2011/148, 2011.