Preimage and Pseudo-Collision Attacks on Step-Reduced SM3 Hash Function

Gaoli Wang[1] and Yanzhao Shen[1]

[1]School of Computer Science and Technology, Donghua University,
Shanghai 201620, China
wanggaoli@dhu.edu.cn, yanzhao_shen@yahoo.com.cn

Abstract. SM3 [11] is the Chinese cryptographic hash standard which was announced in 2010 and designed by Wang *et al.*. It is based on the Merkle-Damgård design and its compression function can be seen as a block cipher used in Davies-Meyer mode. It uses message block of length 512 bits and outputs hash value of length 256 bits.

This paper studies the security of SM3 hash function against preimage attack and pseudo-collision attack. We propose preimage attacks on 29-step and 30-step S-M3, and pseudo-preimage attacks on 31-step and 32-step SM3 out of 64 steps. The complexities of these attacks are 2^{245} 29-step operations, $2^{251.1}$ 30-step operations, 2^{245} 31-step operations and $2^{251.1}$ 32-step operations, respectively. These (pseudo) preimage attacks are all from the first step of the reduced SM3. Meanwhile, these (pseudo) preimage attacks can be converted into pseudo-collision attacks on SM3 reduced to 29 steps, 30 steps, 31 steps and 32 steps with complexities of 2^{122}, $2^{125.1}$, 2^{122} and $2^{125.1}$ respectively. As far as we know, the previously best known preimage attacks on SM3 cover 28 steps (from the first step) and 30 steps (from the 7-th step), and there is no publicly published result on (pseudo) collision attack on SM3.

Keywords: Preimage Attack, Collision attack, Differential meet-in-the-middle, SM3, Hash function.

1 Introduction

Hash functions are an important cryptographic primitive and play a very important role in modern cryptology. They are supposed to satisfy collision resistance, preimage resistance and second preimage resistance. There is a breakthrough in the collision attack on hash functions in 2005 [12, 13, 2]. With the collision attacks on a series of standard hash functions, preimage attack has drawn a great amount of attention from many researchers (see [1, 3, 5, 8, 14] for example). Up to now, the meet-in-the-middle technique and many improved techniques such as initial structure, splice-and-cut, biclique and so on have been widely used in the preimage attack. Recently, a differential view on the meet-in-the-middle technique [5] was proved very useful for the preimage attack on hash functions with linear message expansion and weak diffusion properties.

SM3 [11] hash function is the Chinese cryptographic hash standard which was designed by Wang *et al.* and announced in 2010. It has the similar structure as SHA-256.

However, it has a more complex step function and stronger message dependency than SHA-256. Few attacks were published on SM3 hash function. The work in [15] presented preimage attacks on 28-step (from the 1-st step) and 30-step (from the 7-th step) SM3 with complexities of 2^{249} and $2^{241.5}$ respectively. Recently, a boomerang attack on SM3 reduced to 35 steps [4] was proposed with a complexity of $2^{117.1}$.

In this paper, we focus on the security evaluation of the preimage resistance and collision resistance of SM3 hash function. Based on the differential meet-in-the-middle technique etc., we successfully present (pseudo) preimage attacks and pseudo-collision attacks on 29-step, 30-step, 31-step and 32-step reduced SM3 hash function. All of these attacks start from the first step of SM3. This result provides a better understanding concerning the message expansion and diffusion properties of SM3 hash function. The previous results and the summary of our results are given in Table 1.

The rest of this paper is organized as follows. Section 2 introduces the techniques used throughout the paper. Section 3 gives a brief description of SM3, some notations used in this paper. Section 4 presents preimage and pseudo-collision attacks on step-reduced SM3. Section 5 concludes this paper.

Table 1. Summary of the attacks on SM3 compression function (CF) and hash function (HF)

Attack	CF/HF	Steps	Time	Source
Preimage attack	HF	28	$2^{241.5}$	[15]
Preimage attack	HF	30*	2^{249}	[15]
Boomerang attack	CF	32	$2^{14.4}$	[4]
Boomerang attack	CF	33	$2^{32.4}$	[4]
Boomerang attack	CF	34	$2^{53.1}$	[4]
Boomerang attack	CF	35	$2^{117.1}$	[4]
Preimage attack	HF	29	2^{245}	Sect. 4
Preimage attack	HF	30	$2^{251.1}$	Sect. 4
Pseudo-preimage attack	HF	31	2^{245}	Sect. 4
Pseudo-preimage attack	HF	32	$2^{251.1}$	Sect. 4
Pseudo-collision attack	HF	29	2^{122}	Sect. 4
Pseudo-collision attack	HF	30	$2^{125.1}$	Sect. 4
Pseudo-collision attack	HF	31	2^{122}	Sect. 4
Pseudo-collision attack	HF	32	$2^{125.1}$	Sect. 4

* The attack starts from the 7-th step.

2 Techniques for Preimage Attack and Pseudo-Collision Attack

In this section, we will introduce the related techniques used throughout the paper.

2.1 The Meet-in-the-Middle Preimage Attack

The general idea of the meet-in-the-middle preimage attack can be described as follows(See Fig. 1). It is a type of birthday attack and makes use of a space-time tradeoff. Split the compression function into two subparts (E_f and E_b). E_f computes forward from the splitting point and obtain a set of values at the matching point. Then E_b computes backward and gets another set of results. The two sets of results are compared to search an intersection. The two computation procedures must be independent on each other so that the birthday attack rule can be applied.

The meet-in-the-middle technique can be combined with many techniques such as initial structure technique, splice-and-cut technique, biclique technique, etc. to improve the preimage attack.

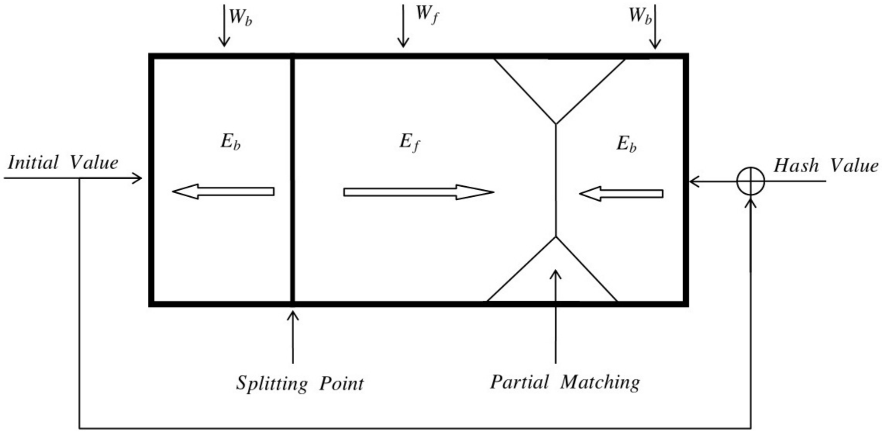

Fig. 1. Schematic view of the meet-in-the-middle attack

2.2 The Differential Meet-in-the-Middle Technique

We review the differential meet-in-the-middle preimage attack [5] which uses the truncated differential [7] in the following.

For a truncation mask vector $T \in \{0,1\}^n$, the equation $A =_T B$ denotes $T \wedge (A \oplus B) = 0$, where $\wedge$ is bitwise AND. The compression functions of SM3 can be seen as $CF = E(M, IV) \oplus IV$, where $E : \{0,1\}^k \times \{0,1\}^n \rightarrow \{0,1\}^n$ is a block cipher with block length n and key length k ($k > n$). Divide E into two parts, $E = E_2 \cdot E_1$, and find two linear spaces $D_1, D_2 \subset \{0,1\}^k$ which satisfy the following three conditions. Firstly, $D_1 \cap D_2 = \{0\}$. Secondly, for an uniformly chosen message M, for each $\delta_1 \in D_1$, there is a $\varDelta_1 \in \{0,1\}^n$ such that $p_1 = Pr[\varDelta_1 =_T E_1(M, IV) \oplus E_1(M \oplus \delta_1, IV)]$, that is, $(\delta_1, 0) \rightarrow \varDelta_1$ is a related-key differential for E_1 with probability p_1. Thirdly, for an uniformly chosen message M, for each $\delta_2 \in D_2$, there is a $\varDelta_2 \in \{0,1\}^n$ such that

$p_2 = Pr[\varDelta_2 =_T E_2^{-1}(M, H \oplus IV) \oplus E_2^{-1}(M \oplus \delta_2, H \oplus IV)]$, that is, $(\delta_2, 0) \rightarrow \varDelta_2$ is a related-key differential for E_2^{-1} with probability p_2.

A candidate preimage can be searched using Algorithm 1. The above second and third conditions make sure that Algorithm 1 answers correctly with probability $p_1 \cdot p_2$. The error of Algorithm 1 is defined as follows: $M \oplus \delta_1 \oplus \delta_2$ is a preimage, but Algorithm 1 rejects it. The error probability is about $1 - p_1 p_2$, which can be obtained by experiment. For the detailed description, we refer to [5]. We can also know that $L_1[\delta_2] =_T L_2[\delta_1]$ (in the last loop of Algorithm 1) is equivalent to $E_1(M \oplus \delta_1 \oplus \delta_2, IV) =_T E_2^{-1}(M \oplus \delta_1 \oplus \delta_2, H \oplus IV)$ which is true if $M \oplus \delta_1 \oplus \delta_2$ is a candidate preimage.

Algorithm 1

Testing $M \oplus \delta_1 \oplus \delta_2$ for a candidate preimage

```
Input: $D_1, D_2 \subset \{0,1\}^k$, $T \in \{0,1\}^n$, $M \subset \{0,1\}^k$
Output: A candidate preimage of the hash value
        $H$ if one is contained in $M \oplus \delta_1 \oplus \delta_2$
Algorithm:
for all $\delta_2 \in D_2$, do
    $L_1[\delta_2] = E_1(M \oplus \delta_2, IV) \oplus \varDelta_2$
end for
for all $\delta_1 \in D_1$, do
    $L_2[\delta_1] = E_2^{-1}(M \oplus \delta_1, H \oplus IV) \oplus \varDelta_1$
end for
for all $(\delta_1 \times \delta_2) \in D_1 \times D_2$, do
    if $L_1[\delta_2] =_T L_2[\delta_1]$, then
        return $M \oplus \delta_1 \oplus \delta_2$
    end if
end for
return No candidate preimage in $M \oplus \delta_1 \oplus \delta_2$
```

If D_1 and D_2 both have dimension d, for a random M, the set $M \oplus D_1 \oplus D_2$ contains $2^{2d} = 2^d \times 2^d$ different messages. Using Algorithm 1, we can observe that a preimage can be obtained with a complexity of $(2^{n-d}\varGamma + 2^{n-t}\varGamma_{re})/(p_1 \cdot p_2)$, where $\varGamma$ is the cost of one compression function operation, $\varGamma_{re}$ is the cost of retesting a candidate preimage and t is the hamming weight of T.

2.3 Converting Pseudo-Preimage Attack into Pseudo-Collision Attack

[10] proposed a technique to convert pseudo-preimage attack into pseudo-collision attack. Assume we can get a t-bit partial target preimage M with matching point in the last step with complexity 2^k, then by finding $2^{(n-t)/2}$ different t-bit partial target preimage Ms, we can get a pseudo-collision with high probability. The total complexity to get a pseudo-collision is $2^{(n-t)/2} \times 2^k$.

If the t-bit partial target preimages are constructed by the meet-in-the-middle technique, then we can evaluate the complexity as follows. For example, in the case of $t = 6$

and $d = 5(> t/2)$, we can find $2^4(= 2^{5+5}/2^6)$ 6-bit partial target preimage with complexity 2^5, which means that a 6-bit partial target preimage is found with the complexity of $2(= 2^5/2^4)$. Then a pseudo-collision can be found with the complexity of $2^{(n-6)/2} \times 2$.

3 Description of SM3 and Notations

In this section, we will give a brief description of SM3 and some notations used throughout the paper.

3.1 Description of SM3

The SM3 hash function compresses any message no more than $2^{64}-1$ bits into a 256-bit hash value. The algorithm first pads any given message into n 512-bit message blocks. The hash function consists of the following two parts: the message expansion and the state update transformation. For the detailed description of SM3, we refer to [11].
Message Expansion. The message expansion of SM3 splits the 512-bit message block M into 16 words w_i $(0 \le i \le 15)$, and expands them into 68 expanded message words w_i $(0 \le i \le 67)$ and 64 expanded message words $w_i{}'$ $(0 \le i \le 63)$ as follows:

$$w_i = P_1(w_{i-16} \oplus w_{i-9} \oplus (w_{i-3} \lll 15)) \oplus (w_{i-13} \lll 7) \oplus w_{i-6}, 16 \le i \le 67,$$

$w_i{}' = w_i \oplus w_{i+4}, 0 \le i \le 63$, where $P_1(X) = X \oplus (X \lll 15) \oplus (X \lll 23)$.
State Update Transformation. The state update transformation starts from an initial value $(A_0, B_0, C_0, D_0, E_0, F_0, G_0, H_0) = IV$ of eight 32-bit words and updates them in 64 steps. In step $i+1(0 \le i \le 63)$ the 32-bit words w_i and $w_i{}'$ are used to update the state variables $A_i, B_i, C_i, D_i, E_i, F_i, G_i, H_i$ as follows:

$$\begin{aligned}
SS1_i &= ((A_i \lll 12) + E_i + (T_i \lll i)) \lll 7,\\
SS2_i &= SS1_i \oplus (A_i \lll 12),\\
TT1_i &= FF_i(A_i, B_i, C_i) + D_i + SS2_i + w_i{}',\\
TT2_i &= GG_i(E_i, F_i, G_i) + H_i + SS1_i + w_i,\\
A_{i+1} &= TT1_i, B_{i+1} = A_i, C_{i+1} = (B_i \lll 9), D_{i+1} = C_i,\\
E_{i+1} &= P_0(TT2_i), F_{i+1} = E_i, G_{i+1} = (F_i \lll 19), H_{i+1} = G_i.
\end{aligned}$$

where $P_0(X) = X \oplus (X \lll 9) \oplus (X \lll 17)$.

The bitwise boolean functions $FF_i(X_i, Y_i, Z_i)$ and $GG_i(X_i, Y_i, Z_i)$ are defined as follows.

$$FF_i(X_i, Y_i, Z_i) = \begin{cases} X_i \oplus Y_i \oplus Z_i, & 0 \le i \le 15,\\ (X_i \wedge Y_i) \vee (X_i \wedge Z_i) \vee (Y_i \wedge Z_i), & 16 \le i \le 63, \end{cases}$$
$$GG_i(X_i, Y_i, Z_i) = \begin{cases} X_i \oplus Y_i \oplus Z_i, & 0 \le i \le 15,\\ (X_i \wedge Y_i) \vee (\neg X_i \wedge Z_i), & 16 \le i \le 63. \end{cases}$$

If M is the last block, then $(A_{64} \oplus A_0, B_{64} \oplus B_0, C_{64} \oplus C_0, D_{64} \oplus D_0, E_{64} \oplus E_0, F_{64} \oplus F_0, G_{64} \oplus G_0, H_{64} \oplus H_0)$ is the hash value. Otherwise it is part of the input of the next message block.

3.2 Notations

- $\varDelta v$: the difference corresponding to the variable v.
- $\varDelta x = [\alpha \sim \beta]$: the bits from the α-th bit to the β-th bit of $\varDelta x$ ($x = w_i, w_i'$) take all possible values, and the other bits of $\varDelta x$ are zero.
- $\varDelta y = \langle\alpha \sim \beta\rangle$: the bits from the α-th bit to the β-th bit of $\varDelta y$ ($y = A_i, B_i, \ldots, H_i$) are known, and the other bits of $\varDelta y$ are unknown.
- $\varDelta y = \langle\alpha \sim \beta, \gamma \sim \delta\rangle$: the bits from the α-th bit to the β-th bit and from the γ-th bit to the δ-th bit of $\varDelta y$ are known, and the other bits of $\varDelta y$ are unknown.
- $\varDelta y = \langle\alpha\rangle$: the α-th bit of $\varDelta y$ is known, and the other bits of $\varDelta y$ are unknown.
- $\varDelta z =?$: $\varDelta z$ is unknown.

4 Preimage and Pseudo-Collision Attacks on Step-Reduced SM3 Hash Function

In this section, we present the attacks on 29-step and 30-step SM3 hash function and expand the 30-step attacks to 32-step attacks using the biclique technique. Before describing the attacks, we give a property of the integer modular addition.

Property 1. Let x, y be two n-bit words and $z = x + y$. Denote $x = x_{n-1}x_{n-2}...x_1x_0$, $y = y_{n-1}y_{n-2}...y_1y_0$ and $z = z_{n-1}z_{n-2}...z_1z_0$. x and $y_i(i = s, ..., n-1)$ are known, and $y_i(i = 0, ..., s-1)$ are unknown. Let p represent the probability that $z_i(i = n-t, ..., n-1)$ are known, then $p \geq 2^{t+s-n}$, where $n - t \geq s$.

Proof. Denote $\bar{x} = x_{n-1}...x_s0...0$, $\bar{y} = y_{n-1}...y_s0...0$ and $\bar{z} = \bar{x} + \bar{y}$. If there is no carry from the $(s-1)$-th bit to the s-th bit, then $z_i = \bar{z}_i(i = n-t, ..., n-1)$ are always known. Assume there is a bit carry from the $(s-1)$-th bit to the s-th bit, then there is a bit carry from the $(n-t-1)$-th bit to the $(n-t)$-th bit if and only if $\bar{z}_i = 1(i = s, ..., n-t-1)$, which holds with a probability of $2^{-(n-t-1-s+1)} = 2^{t+s-n}$.

4.1 Preimage and Pseudo-Collision Attacks on 29-Step SM3

The forward subpart is from the 1-st step to the 15-th step and the backward subpart is from the 29-th step to the 16-th step. The linear spaces D_1, D_2 and the truncation mask vector T_{29f}, T_{29b} are chosen as follows: $D_1 = \{x_0\|\ldots\|x_{15}\big| x_i = 0, x_{15} \in [0 \sim 5], 0 \leq i \leq 14\}$, $D_2 = \{x_{17}\|\ldots\|x_{32}\big| x_{17} \in [26 \sim 31], x_i = 0, 18 \leq i \leq 32\}$, $T_{29f} = \{0,0,0,0,0,0,0, ffffffff\}$, and $T_{29b} = \{0,0,0,0,0,0,0,7f\}$. The differential characteristics for steps 1-15 and 29-16 are presented in Table 2 and Table 3 respectively.

In the forward subpart, for all $\delta_1 \in D_1$ and uniformly chosen message M, the equation $Pr[0 =_{T_{29f}} E_1(M, IV) \oplus E_1(M \oplus \delta_1, IV)] = 1$ always holds, which means $\varDelta H_{15} = 0$ always holds. In the backward subpart, the equation $Pr[0 =_{T_{29b}} E_2^{-1}(M, H \oplus IV) \oplus E_2^{-1}(M \oplus \delta_2, H \oplus IV)] = 1$ always holds, which means $\varDelta H_{15} = \langle 0 \sim 6\rangle$ always holds. Therefore, we can choose $\{0,0,0,0,0,0,0,7f\}$ as the truncation mask vector. In this case, the complexity of the preimage attack on 29-step SM3 is $2^{256-6} + 2^{256-7} \approx 2^{250.58}$.

Table 2. Differential characteristic for steps 1-15(29 steps, 6 bits)

Differences	Step						
	1	$\cdots$	11	12	13	14	15
Δw	0	$\cdots$	0	0	0	0	0
$\Delta w'$	0	$\cdots$	0	$[0 \sim 5]$	0	0	?
ΔA	0	$\cdots$	0	?	?	?	?
ΔB	0	$\cdots$	0	0	?	?	?
ΔC	0	$\cdots$	0	0	0	?	?
ΔD	0	$\cdots$	0	0	0	0	?
ΔE	0	$\cdots$	0	0	?	?	?
ΔF	0	$\cdots$	0	0	0	?	?
ΔG	0	$\cdots$	0	0	0	0	?
ΔH	0	$\cdots$	0	0	0	0	0
Probability	1	$\cdots$	1	1	1	1	1

The preimage attack on 29-step SM3 can also be carried out in the following implementations. We choose $D_1 = \{x_0\|\ldots\|x_{15}\big| x_i = 0, x_{15} = [3 \sim 14], 0 \leq i \leq 14\}$ and $D_2 = \{x_{17}\|\ldots\|x_{32}\big| x_{17} = [20 \sim 31], x_i = 0, 18 \leq i \leq 32\}$. The differential characteristics for each subpart are presented in Table 5 and Table 6.

For the forward subpart, according to the algorithm of SM3 and Property 1, we can get $\Delta A_{12} = \langle 0 \sim 2, 23 \sim 31\rangle$ holds with probability $1 - 2^{-8}$. Thus $\Delta D_{15} = \Delta C_{14} = \Delta B_{13} \lll 9 = \Delta A_{12} \lll 9 = (\langle 0 \sim 2, 23 \sim 31\rangle) \lll 9 = \langle 0, 11\rangle$ holds with probability $1 - 2^{-8}$. We can easily get $\Delta H_{15} = 0$ always holds. We can get $\Delta D_{15} = \langle 0, 10\rangle$ and $\Delta H_{15} = \langle 0\rangle$ in the backward subpart. Therefore, we choose $\{0, 0, 0, 7ff, 0, 0, 0, 1\}$ as the truncation mask vector. Hence, we can obtain a one-block preimage attack on 29-step SM3 with no padding. The complexity of the attack is $(2^{244} + 2^{244})/(1 - 2^{-8}) \approx 2^{245}$.

We can use the splice-and-cut technique to move the matching point to the end of the compression function, combined with the technique described in Section 2.3, a one-block pseudo-collision attack on 29-step SM3 can be obtained with the complexity of $2^{(256-12)/2}/(1 - 2^{-8}) \approx 2^{122}$.

4.2 Preimage and Pseudo-Collision Attacks on 30-Step SM3

The forward subpart is from the 1-st step to the 16-th step and the backward subpart is from the 30-th step to the 17-th step. We choose the linear spaces $D_1 = \{x_0\|\ldots\|x_{15}\big| x_i = 0, x_{15} = [0 \sim 5], 0 \leq i \leq 14\}$ and $D_2 = \{x_{18}\|\ldots\|x_{33}\big| x_{18} = [26 \sim 31], x_i = 0, 19 \leq i \leq 33\}$.

For the forward subpart, $Pr[0 =_{\{0,0,0,0,0,0,0,3f\}} E_1\ (M, IV) \oplus E_1(M \oplus \delta_1, IV)]$ holds with the probability of $0.932 (= 1 - 0.035 - 0.017 - 0.009 - 0.004 - 0.002 - 0.001)$ on average by experiment (See Table 4). In the backward subpart, $\Delta H_{16} = \langle 0 \sim 5\rangle$ holds

Table 3. Differential characteristic for steps 29-16(29 steps, 6 bits)

Differences	Step					
	29	$\cdots$	19	18	17	16
Δw	0	$\cdots$	0	$[26 \sim 31]$	0	0
$\Delta w'$	0	$\cdots$	0	$[26 \sim 31]$	0	0
ΔA	0	$\cdots$	0	0	0	0
ΔB	0	$\cdots$	0	0	0	?
ΔC	0	$\cdots$	0	0	?	?
ΔD	0	$\cdots$	0	?	?	?
ΔE	0	$\cdots$	0	0	0	0
ΔF	0	$\cdots$	0	0	0	$\langle 0 \sim 6 \rangle$
ΔG	0	$\cdots$	0	0	$\langle 0 \sim 25 \rangle$	$\langle 0 \sim 25 \rangle$
ΔH	0	$\cdots$	0	$\langle 0 \sim 25 \rangle$	$\langle 0 \sim 25 \rangle$	$\langle 0 \sim 6 \rangle$
Probability	1	$\cdots$	1	1	1	1

with probability 1. We choose the truncation mask vector as $\{0, 0, 0, 0, 0, 0, 0, 3f\}$, then the probability of $\Delta H_{16} = \langle 0 \sim 5 \rangle$ holds with the probability of 0.932. So we can get a preimage with no padding of 30-step SM3 with a complexity of $(2^{250} + 2^{250})/0.932 \approx 2^{251.1}$.

Table 4. Test the probability for the forward subpart (30-step with dimension 6)*

Test Number	Error Number					
	H_0	H_1	H_2	H_3	H_4	H_5
2^{20}	43367	21711	10820	5558	2622	1371
2^{24}	350340	175413	87739	44166	21672	10889
2^{26}	2804897	1402582	700876	352625	174518	87331

* $H_{16,i}$ denotes the *i-th* bit of H_{16} used in the forward subpart.

By moving the matching point to the end of the compression function using the splice-and-cut technique, combined with the algorithm described in section 2.3, we can get a pseudo-collision attack on 30-step SM3 with the complexity of $2^{((256-6))/2}/0.932 \approx 2^{125.1}$.

4.3 Pseudo-Preimage and Pseudo-Collision Attacks on 32-Step SM3 using bicliques

Based on the preimage attack on 30-step SM3, we present a preimage attack on 32-step SM3 by adding 2-step biclique. The forward subpart is from the 17-th step to the 32-nd step, and the corresponding compression function is denoted as F_1. The backward subpart is from the 14-th step to the 1-st step, and the corresponding compression function is denoted as F_2^{-1}. The biclique covers the steps 15 and 16, and the corresponding compression function is denoted as F_{bic}. We choose the linear spaces $D_1 = \{x_{16}\|\dots\|x_{31}\big| x_i = 0, x_{31} = [0 \sim 5], 16 \leq i \leq 30\}$ and $D_2 = \{x_2\|\dots\|x_{17}\big| x_i = 0, x_2 = [26 \sim 31], 3 \leq i \leq 17\}$. From the preimage attack on 30-step SM3, we know that $\varDelta H_{32} = \langle 0 \sim 5\rangle$ holds with with probability 0.932 in the forward subpart and $\varDelta H_0 = \langle 0 \sim 5\rangle$ holds with with probability 1 in the backward subpart.

So we focus on how to construct the bicliques. The biclique technique [6] is formalized from the initial structure technique [9]. A biclique for F_{bic} is a tuple $\{M, D_1, D_2, P, Q\}$ where M is a message, D_1 and D_2 are linear difference spaces of dimension d, and P is a list of 2^d states $P[\delta_{1i}]$ for $\delta_{1i} \in D_1$, Q is a list of 2^d states $Q[\delta_{2j}]$ for $\delta_{2j} \in D_2$, such that for all $(\delta_{1i}, \delta_{2j}) \in D_1 \times D_2$, $Q[\delta_{2j}] = F_{bic}(M \oplus \delta_{1i} \oplus \delta_{2j}, P[\delta_{1i}])$ holds. With such a biclique, the set $M \oplus D_1 \oplus D_2$ can be searched for candidate pseudo-preimage by testing $F_1(M \oplus \delta_{2j}, Q[\delta_{2j}]) \oplus \varDelta_2 =_T F_2^{-1}(M \oplus \delta_{1i}, P[\delta_{1i}]) \oplus \varDelta_1 \oplus H$, where H is the hash value.

The details of the 2-step biclique can be explained as follows. From the linear space D_1, we know that $\varDelta w_i = 0$ $(16 \leq i \leq 30)$, $\varDelta w_{31} \neq 0$ in the forward subpart, which means $\varDelta w_{14} = 0$, $\varDelta w'_{14} = 0$, $\varDelta w_{15} \neq 0$ and $\varDelta w'_{15} \neq 0$. Similarly, from the linear space D_2, we can get $\varDelta w_{14} = 0$, $\varDelta w'_{14} \neq 0$, $\varDelta w_{15} = 0$ and $\varDelta w'_{15} = 0$. For any randomly chosen M and $P_0 = (A_{14}, B_{14}, C_{14}, D_{14}, E_{14}, F_{14}, G_{14}, H_{14})$, denote the output of the i-th step as $(A_i, B_i, C_i, D_i, E_i, F_i, G_i, H_i)$ $(i = 15, 16)$. We can obtain that for the message $M \oplus \delta_{2j}$ $(\delta_{2j} \neq 0)$ and $P_0 = (A_{14}, B_{14}, C_{14}, D_{14}, E_{14}, F_{14}, G_{14}, H_{14})$, the output of the 16-th step is $Q[\delta_{2j}] = (A_{16}^{\delta_{2j}}, A_{15}^{\delta_{2j}}, C_{16}, D_{16}, E_{16}^{\delta_{2j}}, F_{16}, G_{16}, H_{16})$ where $A_{16}^{\delta_{2j}} \neq A_{16}$, $A_{15}^{\delta_{2j}} \neq A_{15}$ and $E_{16}^{\delta_{2j}} \neq E_{16}$. For the message $M \oplus \delta_{1i}$ $(\delta_{1i} \neq 0)$ and $Q_0 = (A_{16}, B_{16}, C_{16}, D_{16}, E_{16}, F_{16}, G_{16}, H_{16})$, we can get that the input of the 15-th step is $P[\delta_{1i}] = (A_{14}, B_{14}, D_{15}^{\delta_{1i}}, D_{14}^{\delta_{1i}}, E_{14}, F_{14}, H_{15}^{\delta_{1i}}, H_{14}^{\delta_{1i}})$ where $D_{15}^{\delta_{1i}} \neq C_{14}$, $D_{14}^{\delta_{1i}} \neq D_{14}$, $H_{15}^{\delta_{1i}} \neq G_{14}$ and $H_{14}^{\delta_{1i}} \neq H_{14}$. According to the algorithm of SM3, we get get that for $M \oplus \delta_{1i} \oplus \delta_{2j}$ and $P[\delta_{1i}]$, the output of the 16-th step is $Q[\delta_{2j}]$. Therefore, the 2-step biclique covering steps 15 and 16 can be constructed and the cost of constructing the bicliques is negligible. Thus, the pseudo-preimage attack on 32-step SM3 hash function with no padding can be obtained with a complexity of about $2^{251.1}$ and the pseudo-collision attack on 32-step SM3 with no padding can be obtained with a complexity of about $2^{125.1}$.

By the way, by adding 2-step biclique covering steps 15 and 16 to the preimage attack on 29-step SM3, we can obtain a pseudo-preimage attack on 31-step SM3 with the same complexity as the preimage attack on 29-step SM3. Furthermore, the pseudo-preimage attack on 31-step SM3 can be converted into the pseudo-collision attack on 31-step SM3 with the same complexity as the pseudo-collision attack on 29-step SM3. 3-step biclique cannot be constructed because of the characteristic of the state update transformation of SM3.

5 Conclusions

In this paper we have presented some new results on preimage attacks and pseudo-collision attacks on SM3. We first construct several high probability truncated differen-

Algorithm 2 Construction of the bicliques

```
Input: $M, D_1, D_2 \subset \{0,1\}^k$
Output: A biclique
Algorithm:
For uniformly chosen $P_0$, do $P_0 \rightarrow Q_0$
for all $\delta_{2j} \in D_2$, do $P_0 \rightarrow Q[\delta_{2j}]$
end for
for all $\delta_{1i} \in D_1$, do $P[\delta_{1i}] \leftarrow Q_0$
end for
return $P[\delta_{1i}] \rightarrow Q[\delta_{2j}]$
```

tials by taking advantage of some particular weakness of the state update transformation and linear message expansion of SM3. On the basis of them, we have been able to give the best preimage attack known on SM3 by applying the differential meet-in-the-middle technique. Our preimage attack works up to 32 steps (from the first step), while the analysis for the biggest number of steps in the previous result worked on 28 steps (from the first step) and 30 steps (from the 7-th step). Furthermore, we convert the preimage attacks to the pseudo-collision attacks on SM3. The time complexities of the (pseudo) preimage attacks on 29-step, 30-step, 31-step and 32-step SM3 are 2^{245}, $2^{251.1}$, 2^{245} and $2^{251.1}$ respectively. The time complexities of the pseudo-collision attacks on 29-step, 30-step, 31-step and 32-step SM3 are 2^{122}, $2^{125.1}$, 2^{122} and $2^{125.1}$ respectively.

References

1. Kazumaro Aoki, Yu Sasaki, Preimage Attacks on One-Block MD4, 63-Step MD5 and More, in: Selected Areas in Cryptography - SAC 2008, in: LECTURE NOTES IN COMPUTER SCIENCE, vol. 5381, Springer, 2008, pp. 103-119.
2. Eli Biham, Rafi Chen, Antoine Joux, Patrick Carribault, Christophe Lemuet, William Jalby, Collisions of SHA-0 and Reduced SHA-1, in: Advances in Cryptology - EUROCRYPT 2005, in: LECTURE NOTES IN COMPUTER SCIENCE, vol. 3494, Springer, 2005, pp. 36-57.
3. Jian Guo, San Ling, Christian Rechberger, Huaxiong Wang, Advanced Meet-in-the-Middle Preimage Attacks: First Results on Full Tiger, and Improved Results on MD4 and SHA-2, in: Advances in Cryptology - ASIACRYPT 2010, in: LECTURE NOTES IN COMPUTER SCIENCE, vol. 6477, Springer,
4. Aleksandar Kircanski, Yanzhao Shen, Gaoli Wang, Amr M. Youssef, Boomerang and Slide-Rotational Analysis of SM3 hash function, in: Selected Areas in Cryptography - SAC 2012, in: LECTURE NOTES IN COMPUTER SCIENCE, vol. 7707, Springer, 2012, pp. 305-321.
5. Simon Knellwolf1, Dmitry Khovratovich, New Preimage Attacks against Reduced SHA-1, in: Advances in Cryptology-CRYPTO 2012, in: LECTURE NOTES IN COMPUTER SCIENCE, vol. 7417, Springer, 2012, pp. 367-383.
6. Dmitry Khovratovich, Christian Rechberger, Alexandra Savelieva, Bicliques for Preimages: Attacks on Skein-512 and the SHA-2 family, in: Fast Software Encryption - FSE 2012, in: LECTURE NOTES IN COMPUTER SCIENCE, vol. 7549, Springer, 2012, pp. 244-263.
7. Lars R. Knudsen, Truncated and Higher Order Differentials, in: Fast Software Encryption - FSE 1994, in: LECTURE NOTES IN COMPUTER SCIENCE, vol. 1008, Springer, 1994, pp. 196-211.

8. Gaëtan Leurent, MD4 is Not One-Way, in: Fast Software Encryption - FSE 2008, in: Lecture Notes in Computer Science, vol. 5086, Springer, 2008, pp. 412-428.
9. Yu Sasaki, Kazumaro Aoki, Finding Preimages in Full MD5 Faster Than Exhaustive Search, in: Advances in Cryptology - EUROCRYPT 2009, in: Lecture Notes in Computer Science, vol. 5479, Springer, 2009, pp. 134-152.
10. Ji Li, Takanori Isobe, Kyoji Shibutani, Converting meet-in-the-middle preimage attack into pseudo collision attack: Application to SHA-2, in: Fast Software Encryption - FSE 2012, in: LECTURE NOTES IN COMPUTER SCIENCE, vol. 7549, Springer, 2012, pp. 264-286 .
11. *Specification of SM3 cryptographic hash function (In Chinese), http://www.oscca.gov.cn/UpFile /20101222141857786.pdf.*
12. Xiaoyun Wang, Hongbo Yu, How to Break MD5 and Other Hash Functions, in: Advances in Cryptology - EUROCRYPT 2005, in: Lecture Notes in Computer Science, vol. 3494, Springer, 2005, pp. 19-35.
13. Xiaoyun Wang, Yiqun Lisa Yin, Hongbo Yu, Finding Collisions in the Full SHA-1, in: Advances in Cryptology - CRYPTO 2005, in: Lecture Notes in Computer Science, vol. 3621, Springer, 2005, pp. 17-36.
14. Hongbo Yu, GaoliWang, Guoyan Zhang, XiaoyunWang, The second-preimage attack on MD4, in Cryptology and Network Security - CANS 2005, in: LECTURE NOTES IN COMPUTER SCIENCE, vol. 3810, Springer, 2005, pp. 1-12.
15. Jian Zou, Wenling Wu, Shuang Wu, Bozhan Su, Le Dong, Preimage Attacks on Step-Reduced SM3 Hash Function, in: International Conference on Information Security and Cryptology - ICISC 2011, in: Lecture Notes in Computer Science, vol. 7259, Springer, 2011, pp. 375-390.

Table 5. Differential characteristic for steps 1-15(29 steps, 12 bits)

Differences	Step						
	1	$\cdots$	11	12	13	14	15
Δw	0	$\cdots$	0	0	0	0	0
$\Delta w'$	0	$\cdots$	0	$[3 \sim 14]$	0	0	?
ΔA	0	$\cdots$	0	$\langle 0 \sim 2, 23 \sim 31\rangle$	?	?	?
ΔB	0	$\cdots$	0	0	$\langle 0 \sim 2, 23 \sim 31\rangle$	?	?
ΔC	0	$\cdots$	0	0	0	$\langle 0 \sim 11\rangle$	?
ΔD	0	$\cdots$	0	0	0	0	$\langle 0 \sim 11\rangle$
ΔE	0	$\cdots$	0	0	?	?	?
ΔF	0	$\cdots$	0	0	0	?	?
ΔG	0	$\cdots$	0	0	0	0	?
ΔH	0	$\cdots$	0	0	0	0	0
Probability	1	$\cdots$	1	$1-2^{-8}$	1	1	1

Table 6. Differential characteristic for steps 29-16(29 steps, 12 bits)

Differences	Step					
	29	$\cdots$	19	18	17	16
Δw	0	$\cdots$	0	$[20 \sim 31]$	0	0
$\Delta w'$	0	$\cdots$	0	$[20 \sim 31]$	0	0
ΔA	0	$\cdots$	0	0	0	0
ΔB	0	$\cdots$	0	0	0	$\langle 0 \sim 10 \rangle$
ΔC	0	$\cdots$	0	0	$\langle 0 \sim 19 \rangle$	$\langle 0 \sim 19 \rangle$
ΔD	0	$\cdots$	0	$\langle 0 \sim 19 \rangle$	$\langle 0 \sim 19 \rangle$	$\langle 0 \sim 10 \rangle$
ΔE	0	$\cdots$	0	0	0	0
ΔF	0	$\cdots$	0	0	0	$\langle 0 \rangle$
ΔG	0	$\cdots$	0	0	$\langle 0 \sim 19 \rangle$	$\langle 0 \sim 19 \rangle$
ΔH	0	$\cdots$	0	$\langle 0 \sim 19 \rangle$	$\langle 0 \sim 19 \rangle$	$\langle 0 \rangle$
Probability	1	$\cdots$	1	1	1	1