Multi-differential Cryptanalysis on Reduced DM-PRESENT-80: Collisions and Other Differential Properties

Takuma Koyama[1], Yu Sasaki[2], and Noboru Kunihiro[1]

[1] The University of Tokyo
5-1-5 Kashiwanoha, Kashiwa-shi, Chiba 277-8561 Japan
{t-koyama@it.,kunihiro@}k.u-tokyo.ac.jp
[2] NTT Secure Platform Laboratories, NTT Corporation
3-9-11 Midori-cho, Musashino-shi, Tokyo 180-8585 Japan
sasaki.yu@lab.ntt.co.jp

Abstract. The current paper studies differential properties of the compression function of reduced-round DM-PRESENT-80, which was proposed at CHES 2008 as a lightweight hash function with 64-bit digests. Our main result is a collision attack on 12 rounds with a complexity of $2^{29.18}$ 12-round DM-PRESENT computations. Then, the attack is extended to an 18-round distinguisher and an 12-round second preimage attack. In our analysis, the differential characteristic is satisfied by the start-from-the-middle approach. Our success lies in the detailed analysis of the data transition, where the internal state and message values are carefully chosen so that a differential characteristic for 5 rounds can be satisfied with complexity 1 on average. In order to reduce the attack complexity, we consider as many techniques as possible; multi-inbound technique, early aborting technique, precomputation of look-up tables, multi-differential characteristics.

Keywords: DM-PRESENT-80, Collision, Second preimage, Multi-differential cryptanalysis, Rebound attack.

1 Introduction

Recently, demand on the secure communication in a resource constraint environment has been increased, e.g., sensor network with RFID tags. From this background, block-ciphers and hash functions suitable for a resource constraint environment are actively discussed. They are called lightweight block-ciphers and hash functions. One of the remarkable designs for lightweight block-ciphers is PRESENT, which was proposed by Bogdanov *et al.* at CHES 2007 [4]. The block size of PRESENT is 64 bits, and it supports 80- and 128-bit keys. It adopts an SPN structure and consists of 31 rounds for both key sizes. Recently, PRESENT has been adopted by ISO as one of the international standards in lightweight cryptography [12]. Several cryptanalytic results were published against reduced-round PRESENT [2, 7, 8, 11, 14, 21–23, 25]. The current best key recovery attack

T. Kwon, M.-K. Lee, and D. Kwon (Eds.): ICISC 2012, LNCS 7839, pp. 352–367, 2013.

is up to 26 rounds with the assumption that the full codebook is available to the attacker. Without the full codebook, the best attack is up to 25 rounds.

Hash functions are usually constructed by using a block-cipher or permutation as a building block. Hence, it is natural to design lightweight hash functions based on lightweight block-ciphers or a permutation inside lightweight block-ciphers. In fact, there are several lightweight hash functions based on PRESENT or the permutation inside PRESENT. DM-PRESENT was proposed by Bogdanov *et al.* at CHES 2008 [5], where the compression function is simply constructed by using the PRESENT block-cipher in the Davies-Meyer mode [20, Algorithm 9.42]. Another compression function called H-PRESENT was proposed in [5], which consists of a double-block-length mode-of-operation instantiating the PRESENT block-cipher. SPONGENT was proposed by Bogdanov *et al.* at CHES 2011 [3], which adopts the sponge construction [1], and its internal permutation is based on PRESENT.

Attack scenarios for block-ciphers and hash functions are very different. Intuitively, the complexity for the attack on block-ciphers is bounded by the key size, while, for collision attacks agains hash function, the attack complexity is bounded by only a half of the digest size. Thus, attacks on block-ciphers cannot be converted to the attack on hash functions directly. Hence the security of PRESENT-based hash functions must be evaluated independently of the attacks on the PRESENT block-cipher.

In this paper, we study the security of DM-PRESENT. As far as we know, no result is published about it. Regarding H-PRESENT, two results have been announced. One is by Ferguson at the rump session of CRYPTO 2011 [9]. It reported a weakness of the mode-of-operation of H-PRESENT-128 leading to the pseudo-preimage attack whose complexity is a half of the brute force attack. The other is by Kobayashi and Hirose at SCIS 2012 [13], which reports differential attacks for H-PRESENT-80 reduced to 10 rounds.

Our Contributions. This paper studies differential properties of the compression function of reduced-round DM-PRESENT-80. Our main result is a collision attack on 12 rounds with a complexity of $2^{29.18}$ 12-round DM-PRESENT computations. Attacks on block-ciphers and hash functions are different. Therefore, we need to construct a differential characteristic from scratch with considering the following properties; (1) for attacks on hash functions, the attacker can choose internal state and message values so that differential characteristics for several rounds can be satisfied with low complexity. Hence, characteristics must be chosen to take into account such impact. (2) To generate collisions, the differential form of the plaintext and ciphertext must be identical so that they can be canceled each other with the feed-forward operation.

As a result, we construct a 12-round differential characteristic that produces a collision of the compression function with probability of 2^{-70} for a randomly chosen message and chaining variable. We then search for paired values satisfying the characteristic much more efficiently with the rebound attack [19]. The characteristic is divided into inbound part (Round 3 to 7) and outbound part

(Round 0 to 2 and Round 8 to 11). With several techniques such as multi-inbound technique [15, 16] and precomputation of a look-up table, the inbound part is satisfied with a very low complexity, which reduces the attack complexity from 2^{70} to 2^{37}. Moreover, with several techniques such as early aborting technique [6, 26] and multi-differential characteristics [18], the outbound part is satisfied with the complexity of $2^{29.18}$ 12-round DM-PRESENT computations. Finally, the attack becomes faster than the birthday attack.

The 12-round differential characteristic can be extended for other attack scenarios. With respect to a distinguisher, the differential form of the plaintext and ciphertext can be different and the attacker may be allowed to spend more than 2^{32} computations. By extending the 12-round characteristic in forward and backward, we can construct an 18-round distinguisher. Furthermore, with the approach by Yu *et al.* [27], this can be used to mount a second preimage attack for 12 rounds of the compression function. The attack results are summarized in Table 1.

Table 1. Summary of our attacks

Attacks	♯Rounds	Time	Memory
Collision	12	$2^{29.18}$	2^{12}
2nd Preimage	12	$2^{61.91}$	Negl.
Distinguisher	18	$2^{57.18}$	2^{12}

Paper Outline. The organization of this paper is as follows. Sect. 2 summarizes related work. Sect. 3 describes a new collision attack against 12 rounds. In Sect. 4, we extend the attack to several different scenarios. Finally, we conclude the paper in Sect. 5. We postpone the specification of DM-PRESENT-80 in App. A.

2 Previous Work

2.1 Iterative Linear Characteristic of Key Recovery Attack

Linear and multi-linear analyses are the best approach for the key recovery attack on PRESENT. They use linear characteristics of an iterative form. The base of the iterative characteristic is as follows. The linear form of `0x02` can be transformed into `0x06` during the S-box transformation, and the opposite also can be transformed. The idea is also useful to construct an iterative differential characteristic in our attack.

2.2 Rebound Attack

Rebound attack, which was proposed by Mendel *et al.* at FSE 2009, is an approach to satisfy a truncated differential characteristic when the key value is known to the attacker [19]. The technique is useful to analyze hash functions.

Suppose that the round function adopts the SPN structure, where S-layer adopts the S-box transformations, and P-layer introduces a diffusion. In truncated differential cryptanalysis, the only probabilistic part is the transformation in the P-layer. The basic rebound attack can satisfy the differential characteristic with two P-layers $\sharp\mathrm{IN} \rightarrow \mathrm{P} \rightarrow \mathrm{S} \rightarrow \mathrm{P} \rightarrow \sharp\mathrm{OUT}$. The attacker chooses the input difference Δ^{IN} and compute $\mathrm{P}(\Delta^{\mathrm{IN}})$. This can be computed without determining actual values. Similarly, the attacker chooses the output difference Δ^{OUT} and compute $\mathrm{P}^{-1}(\Delta^{\mathrm{OUT}})$. Finally, paired values are determined so that the differential transformation through the middle S-layer is satisfied. Several improved techniques have been proposed after the publication of [19]. In this paper, we particularly use the start-from-the-middle technique [17] and the multi-inbound technique [15, 16]. We stress that our attack is the differential attack, not the truncated one. Thus, these techniques cannot be applied straight-forward, but the ideas of determining internal state values and bypassing several rounds are also useful for our attack.

2.3 Second Preimage Attack on MD4

In 2005, Yu *et al.* proposed a second preimage attack on MD4 [27]. In the second preimage attack, the attacker is given a message M and its digest $H(M)$. For a random oracle, the probability that $H(M) = H(M')$ is satisfied for $M \neq M'$ is 2^{-n}, where n is the size of the hash value. Therefore, finding a way to choose M' satisfying the above equation with a higher probability than 2^{-n} can be regarded as the second preimage attack. MD4 generates 128-bit hash values. Yu *et al.*, against full MD4, found the message difference ΔM that would result in $H(M) = H(M \oplus \Delta M)$ with a probability of 2^{-61}. In a later section, we propose the second preimage attack on the reduced-round compression function of DM-PRESENT-80 with a similar idea.

3 12-Round Collision Attack on Compression Function

This section shows a collision attack against the 12-round DM-PRESENT-80 compression function with a complexity of $2^{29.18}$. We choose the differential approach to find a collision. At first, we give the detailed analysis of the differential propagation within 1 round. Then, the differential characteristic for 12 rounds is introduced.

3.1 Analysis of Differential Properties of S-box

The S-box used in PRESENT is a 4-bit to 4-bit S-box $S(\cdot)$. The following table gives the detailed specification of the S-box in hexadecimal notation.

x	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
$S(x)$	C	5	6	B	9	0	A	D	3	E	F	8	4	7	1	2

We search for pairs of input/output differences of the S-box $(\Delta x, \Delta y)$, where $\Delta x, \Delta y \in \mathbb{F}_2^4$, satisfying the following two conditions.

$$\mathcal{HW}(\Delta x) + \mathcal{HW}(\Delta y) = 3, \tag{1}$$

$$\exists x, y \in \mathbb{F}_2^4 : \Big(S(x) \oplus S(x \oplus \Delta x) = \Delta y\Big) \wedge \Big(S(y) \oplus S(y \oplus \Delta y) = \Delta x\Big). \tag{2}$$

Note that $\mathcal{HW}(x)$ indicates a Hamming weight of x. Let $\Pr(\Delta a, \Delta b)$ be the probability that the input difference Δa is transformed into Δb with an S-box transformation. More strictly, $\Pr(\Delta a, \Delta b)$ is defined as $\sharp\{a | S(a) \oplus S(a \oplus \Delta a) = \Delta b\}/2^4$. Then, we identify $(\Delta x, \Delta y)$ which achieves the maximum value for the following probability;

$$\Pr(\Delta x, \Delta y) \times \Pr(\Delta y, \Delta x). \tag{3}$$

- (1): Slower differential propagations lead to longer differential characteristics. Thus, we need to minimize the number of bits with differences. In the S-box of PRESENT, any input difference with a single bit always produces output differences with at least two bits. Thus, the minimum number of (1) is 3.
- (3): For a fixed $(\Delta x, \Delta y)$, we get an input/output pair with the probability $\Pr(\Delta x, \Delta y)$. The $\Pr(\Delta \cdot, \Delta \cdot)$ is either 2^{-2}, 2^{-3}, or zero.

These can be verified by enumerating through all $2^4 \times 2^4$ input/output pairs. It is remarkable that there is no differential pairs satisfied with the total probability of $2^{-2-2} = 2^{-4}$ in the condition (3). In other words, the maximum value is 2^{-2-3} or 2^{-3-2}, which is 2^{-5}. As a result, we found only three pairs $(\Delta\mathtt{0x4}, \Delta\mathtt{0x9})$, $(\Delta\mathtt{0x4}, \Delta\mathtt{0x5})$, and $(\Delta\mathtt{0x1}, \Delta\mathtt{0x3})$ that satisfy the all conditions. A detailed description of the pairs is given in App. B.

3.2 Entire Differential Characteristic

We construct a 12-round differential characteristic by using the good 1-round characteristics observed in Sect. 3.1. Each three pair can be used to construct a 6-round iterative characteristic with the same probability. However, using $(\Delta\mathtt{0x4}, \Delta\mathtt{0x9})$ and $(\Delta\mathtt{0x4}, \Delta\mathtt{0x5})$ is more advantageous than using $(\Delta\mathtt{0x1}, \Delta\mathtt{0x3})$ because multi-differential characteristics can be constructed. Hereafter, we mainly use the pair $(\Delta\mathtt{0x4}, \Delta\mathtt{0x9})$ to construct 12-round characteristic, and use $(\Delta\mathtt{0x4}, \Delta\mathtt{0x5})$ for the multi-differential characteristics.

Fig. 1 shows our 12-round differential characteristic. Hereafter, we call the bits with differences *active* or *active bits*. In Fig. 1, the blue and black bits represent active bits for the characteristic for $(\Delta\mathtt{0x4}, \Delta\mathtt{0x9})$, and the red and black bits in the first and last 3 rounds represent the ones for $(\Delta\mathtt{0x4}, \Delta\mathtt{0x5})$. We describe the 12-round characteristic for $(\Delta\mathtt{0x4}, \Delta\mathtt{0x9})$ right now, and mention the characteristic for $(\Delta\mathtt{0x4}, \Delta\mathtt{0x5})$ later in Sect. 3.3. Note that $\Pr(\Delta\mathtt{0x4}, \Delta\mathtt{0x9})$ and $\Pr(\Delta\mathtt{0x9}, \Delta\mathtt{0x4})$ are 2^{-3} and 2^{-2}, respectively. First of all, by using the pair $(\Delta\mathtt{0x4}, \Delta\mathtt{0x9})$, we construct a 6-round iterative differential characteristic where the number of active bits transits $8 \to 4 \to 2 \to 1 \to 2 \to 4 \to 8$ with a probability of $2^{-8-4-2-3-6-12} = 2^{-35}$. Second, we repeat the iterative

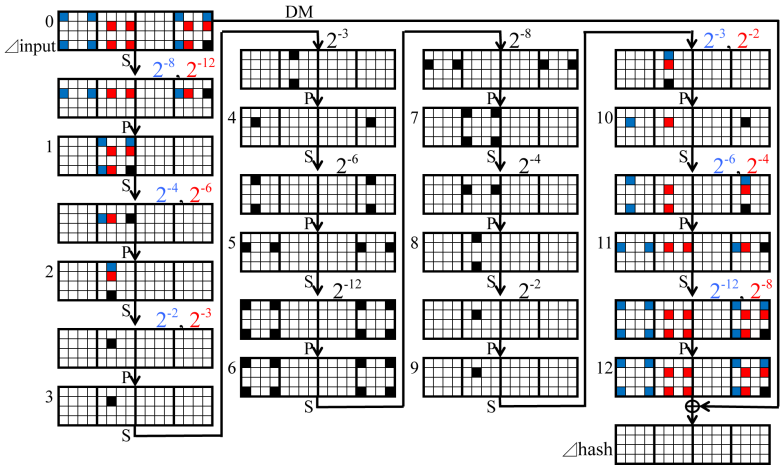

Fig. 1. Differential characteristics on 12-round DM-PRESENT. Black bits denote active bits. White bits denote zero difference. Red and blue bits represent two variants.

characteristic twice and construct the 12-round differential characteristic. The total probability that a randomly chosen input/output pair satisfies the 12-round differential characteristic is $2^{-35-35} = 2^{-70}$.

3.3 Multi-differential Characteristics for Collisions

As later discussed, we use the rebound approach to satisfy the characteristic. For this purpose, we fix the differential characteristic for the middle 5 rounds (Round 3 to Round 8) and choose internal state and message values so that the characteristic can be satisfied. Then, the first 3 and last 4 rounds belong to the outbound phase, where the characteristic is satisfied probabilistically.

We consider reducing the complexity of our collision attack by introducing multi-differential characteristics for the outbound phase. Because both of $(\Delta\texttt{0x4}, \Delta\texttt{0x9})$ and $(\Delta\texttt{0x4}, \Delta\texttt{0x5})$ contain $\Delta\texttt{0x4}$, from the fixed middle 6-round characteristic, we can construct two differential characteristics which have the same active-bit patterns at the plaintext and ciphertext. In Fig. 1, two characteristics are denoted by blue and red. $\Pr(\Delta\texttt{0x4}, \Delta\texttt{0x5})$ and $\Pr(\Delta\texttt{0x5}, \Delta\texttt{0x4})$ are 2^{-2} and 2^{-3}, respectively. Hence, the probability of satisfying the red characteristic is the same as the one for the blue characteristic. As a result, we can obtain a collision pair that follows either two differential characteristics with 2^{-69}, which is double of the single-characteristic case.

3.4 Attack Overview

This section gives an overview of our collision attack procedure. The procedure mainly consists of an inbound phase and an outbound phase. These names derive from the rebound attack described in Section 2.2. We start searching for

a colliding pair from a middle round. In the inbound phase, we aim to obtain many of internal-state values and round-message values that satisfy the differential characteristic for the middle five rounds; state ♯3 to state ♯8 of Fig. 2. The paired values satisfying the differential characteristic for the inbound phase are called *start points*. We need to generate many start points so that the differential propagation of the outbound phase can be satisfied. The inbound phase is further divided into five 1-round procedures. In each procedure, several bits of internal states are fixed to satisfy the differential characteristic. We independently perform the procedures, and then choose several bits of round-message values that connect the results of procedures without any contradiction. After we choose several bits of the internal states and round messages that satisfy the middle five rounds, 63 bits of a round message remain unfixed. We use those bits as the available degrees of freedom for the outbound phase. Therefore, we can prepare enough start points with a very low complexity.

In the outbound phase, we compute each start point in outward until plaintext and ciphertext with checking whether or not the differential propagation conforms one of the multi-differential characteristics. Because the differential propagation is probabilistic, we need to generate enough start points. Due to the DM-mode, the output of the compression function is derived from the exclusive-or of plaintext and ciphertext, and they have the identical differential form. Therefore, the plaintext and ciphertext differences cancel each other surely. In the following part, we describe the procedure of inbound phase and outbound phase in more details.

The Inbound Phase. This phase consists of five 1-round inbound procedures. In each 1-round inbound procedure, the goal is finding paired values satisfying the differential characteristic between the state just before the sBoxlayer and immediately after the pLayer. Note that our inbound phase is the (single) differential attack, not truncated differential attack. Therefore, the differential characteristic is already fixed uniquely. In the inbound phase of i-th round, we firstly fix active column values of state $\sharp i.5$ to satisfy the differential characteristic. For instance, if 1 active column transits from $\Delta 4$ to $\Delta 9$ through the S-box, the output of the S-box must be either `0x7` or `0xE` due to the S-box characteristic. For more details of possible output values of the active column, please refer to App. B. After we fix the paired values of active columns either (`0x7`, `0xE`) or (`0xE`, `0x7`), we compute these values in backward through one S^{-1} function and in forward by one pLayer. The inbound procedure is applied to another adjacent round independently. Then, we merge two inbound procedures by fixing several bits of a round-message value. By iterating the above, we can merge five inbound procedures by fixing several bits of round messages. Choosing several bits of round-message values in different rounds sometimes causes the contradiction in the key schedule function. Using the precomputed look-up table, we can merge five inbound procedures. That is to say, a start point for the outbound phase is generated with a low complexity.

The Outbound Phase. The first and last three rounds belong to the outbound phase, where the differential transition of each S-box is satisfied probabilistically. The probability that one start point satisfies the differential characteristic of seven outbound rounds is $2^{-2-4-8-2-3-6-12} = 2^{-37}$. By using the multi-differential characteristics described in Section 3.3, the probability increases to 2^{-36}. In other words, one of the 2^{36} start points can yield the collision. We check the differential propagation of each of 2^{36} start points round by round with the early aborting technique [6, 26]. Namely, we first check whether or not the start points satisfy the differential transition from round 8 to round 9. If it is not satisfied, we stop the computation of this start point, and choose different one. If it is satisfied, we continue the computation for the next round. Due to this effort, the complexity for examining 2^{36} start points can be reduced into about $\frac{1}{12\times16} \times 2^{36}$, which is faster than the birthday attack on 64-bit values, 2^{32}.

3.5 Attack Procedure

Our attack procedure against the 12-round compression function is as follows.

1. In the precomputation step, we generate a lookup table that are used in Step 11 of inbound phase. We compute $2^{4+4+4=12}$ tuples of $\{(x, y, m)|x, y, m \in \mathbb{F}_2^4; y = S(x \oplus m)\}$ and store them in the look-up table. x and y indicate the input/output of a single S-box, and m is 4 bits of a round message.
2. Inbound phase consists of 11 steps as follows. In this phase, we look for a pair of internal state values and one message value that satisfy the differential transition from state ♯3 to ♯8. We fix the internal states and round messages bit by bit. In Fig. 2, the colored bits in the inbound phase are classified into two types, simple-colored bits and shaded-colored bits. The simple-colored bits of internal states are fixed solely by the S-box characteristic. The shaded-colored bits are fixed after all bits of round message are determined.

Step 1. fix the values of the four columns indicated by black and red (active columns) in state ♯5.5 so that the differential transition from Δ0x4 to Δ0x9 is satisfied through the sBoxlayer in round 5. The probability of this transition is 2^{-3} per column, which means that only one pair of values can satisfy the transition. 0x7 and 0xE are the values of the S-box output satisfying the transition. For the eight black bits, we have 2^4 choices (2 choices per column). We pick any 1 from these 2^4 patterns. We then compute these 4 columns in forward until state ♯6.

Step 2. fix similarly the black and blue active columns in state ♯4.5. We need to fix the values of each S-box output only either 0x7 or 0xE for the same reason as described in Step 1. We pick any 1 from these 2^2 patterns, and then compute these 2 columns in backward until state ♯5. The 8 bits of the state just before the sBoxlayer of the fourth round are also uniquely computed.

Step 3. merge the fixed values in Step 1 and 2 to satisfy the transition from state ♯5 to ♯5.5 by fixing the four red bits of M_6. Depending on un-fixed

bits of M_6, there are several possibilities for the shaded-red bits of state $\sharp 5$ in Fig. 2. The following Steps from 4 to 9 are similar to Step 2 and 3. Only positions of the fixed bits and the number of choices are different from Step 2 and 3. Step 8 and 9 are, however, reverse of Step 2 and 3.

Step 4. fix the black and yellow active columns in state $\sharp 3.5$. For the four black bits, we have 2 choices. After we pick one choice for two black bits, we compute the fixed bits in forward until state $\sharp 4$ and in backward until state $\sharp 3$.

Step 5. merge the fixed values in Step 2 and 4 to satisfy the transition from state $\sharp 4$ to $\sharp 4.5$. We can fix two blue bis of M_5.

Step 6. fix the black and green active columns in state $\sharp 6.5$. In fact, we can fix the each column either `0x0`, `0x4`, `0xB`, or `0xF` due to the differential characteristic of S-box. For the eight black bits, we have 2^8 choices (4 choices per column).

Step 7. merge the fixed values in Step 1 and 6 to satisfy the transition from state $\sharp 6$ to $\sharp 6.5$. We can fix eight green bis of M_7.

Step 8. fix two orange bis of M_8 to merge the fixed values in Step 6 and 9 to satisfy the transition. Due to the message schedule, the values of two bits of fixed M_7 and two bits of M_8 are overlapped. Thus we cannot fix both adjacent inbound procedures independently. Because of the characteristic of S-box, we can still merge them by reducing the choices from 4 to 2 per column.

Step 9. fix the two black and orange active columns in state $\sharp 7.5$. We already fixed some bits of M_8, thus we have 2^2 choices (2 choices per column, not 4 choices) for the two black bits.

Step 10. fix all remaining 63 bits of round messages randomly.

Step 11. merge whole inbound procedures. After Step 10, we can compute the shaded-colored bits in Fig. 2. In $\sharp 5$, each of 12 columns including fixed yellow or fixed blue bit must transits compatibly to $\sharp 5.5$. Thus, we fix the white bits of the internal states to satisfy the transition referring the look-up table. For each of 12 columns, we have two choices of values on averages. In other words, 2^{12} start points for one round message can be constructed with a low complexity. And we can construct more $2^{4+2+1+8+2+63+12} = 2^{92}$ start points because of the forementioned freedom degree.

3. Outbound phase consists of two steps as follows.

Step 1. compute start points in both forward and backward. Then, the total probability of the outbound phase is 2^{-37}. We check the differential propagation round by round by using the early aborting technique.

Step 2. link the input and output values of the internal cipher by exclusive-or of the DM-mode. The input and output differences match with probability of exactly 1, because both of these differences are Δ`0x`9009000000009009[1].

[1] Note that our attack is a differential attack, not a truncated differential attack. Hence, the probability of the match is 1, not 2^{-8}.

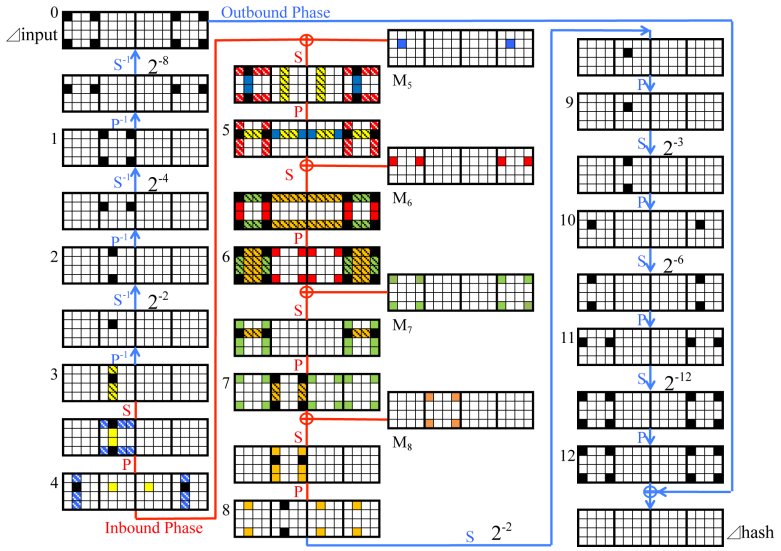

Fig. 2. Differential characteristic focused the inbound and outbound phases. Black denotes active bits. The rest of colored bits is fixed in the inbound phase.

3.6 Complexity Evaluation

We can generate 2^{12} start points for the outbound phase that satisfies the inbound phase with the complexity of about 1 on average and the 2^{12} bits memory requirement. The probability that a pair satisfies the whole outbound phase is 2^{-37}. Utilizing the multi-differential characteristics, the probability that a pair satisfies outbound phase is 2^{-36}. Then, we have to generate 2^{36} start points to find a collision pair. Remember that we can generate enough start points for the outbound phase because of the freedom degrees of internal state and message values. At a glance, a rough evaluation of our attack complexity to find a collision is 2^{36} 12-round DM-PRESENT-80 computations. However, considering the early aborting technique [6, 26], our attack complexity to find a collision is in fact much smaller. Let the complexity of 1-round function is $\frac{1}{12}$ of the 12-round DM-PRESENT-80 function, and the complexity of a column is $\frac{1}{16}$ of 1-round function. We examine all 2^{36} start points for round 8. It is only necessary to compute a column whether or not the start points satisfy the differential characteristics. Hence, the complexity for round 8 is $\frac{1}{12} \times \frac{1}{16} \times 2^{36}$. After round 8, only $2^{36-2} = 2^{34}$ pairs follow the characteristics in Fig. 1. Similarly, we examine 2^{34} start points for round 3. And then, $2^{34-3} = 2^{31}$ pairs follow the red characteristic, and $2^{34} \times 2^{-2} = 2^{32}$ pairs follow the blue one. Hence, we examine $2^{31} + 2^{32}$ pairs for round 9, and thus the complexity for round 9 is $\frac{1}{12} \times \frac{4}{16} \times (2^{31} + 2^{32})$. After round 9, $2^{36-2-3-2} = 2^{29}$ pairs satisfy the red characteristic and $2^{36-2-2-3} = 2^{29}$ pairs satisfy the blue one. After all, the attacker computes $\frac{1}{12\times16} \times (2^{36} + 2^{34}) + \frac{4}{12\times16} \times (2^{31} + 2^{32} + 2^{29} + 2^{29}) + \frac{1}{12} \times (2^{23} + 2^{25} + 2^{11} + 2^{17} + 2^{7} + 2^{11}) \approx 2^{29.18}$

12-round DM-PRESENT-80 computations. Finally, we can find collisions of the 12-round DM-PRESENT-80 compression function faster than the birthday attack, which requires 2^{32} computations.

4 Application for other Attacks

The differential characteristic discussed in the previous section can be used to construct other kinds of attacks. In this section, we discuss an 18-round distinguisher and 12-round second preimage attack on the compression function.

4.1 18-Round Distinguisher

We construct an 18-round differential distinguisher that finds a pair of messages with specific input and output differences. We show that the attack on 18-round DM-PRESENT-80 is faster than the attack on an ideal compression function.

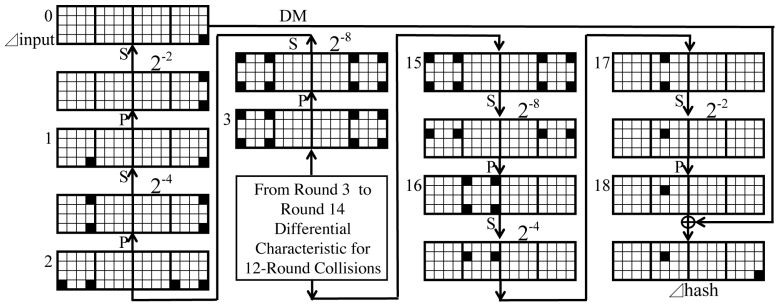

Fig. 3. Differential characteristic for 18-round distinguisher

For this attack, we extend the differential characteristic by 3 rounds in backward and 3 rounds in forward. The procedure of the multi-inbound phase is the same as the one in Sect 3, where middle 5 rounds can be satisfied with average complexity 1. Hence, only the outbound phase is extended and satisfying the entire differential characteristic becomes harder for these extended outbound phase. Note that we do not have to match the differential forms of the plaintext and ciphertext. This enables the attacker to use the differential propagation with probability 2^{-2} for each S-box transformation in both directions, i.e., $\Delta 1 \rightarrow \Delta 9$ with probability of 2^{-2} instead of $\Delta 4 \rightarrow \Delta 9$ with probability of 2^{-3}.

The differential propagations for the first and last 3 rounds are shown in Fig. 3. The input value has 1-bit difference in the chaining variable and no difference in the message. The output values has 1-bit difference. As discussed in Sect. 3, satisfying the middle 12 rounds (round 3 to round 14) requires the complexity of $2^{29.18}$. Then, extending the characteristic from Round 3 to 2, 2 to 1, and 1 to 0 requires the complexity of $2^{2\times 4} = 2^8$, $2^{2\times 2} = 2^4$, and 2^2, respectively. Similarly extending the characteristic from Round 14 to 17 requires $2^{8+4+2} + 2^{14}$. Finally, the

entire characteristic is satisfied with the complexity of $2^{2+4+8+29.18+14} = 2^{57.18}$ 18-round DM-PRESENT computations. We then show that finding such pairs in an ideal compression function requires more complexity. In the truncated differential analysis, this complexity is evaluated by the limited birthday attack proposed Gilbert and Peyrin [10]. However, our attack only allows a 1-bit difference on the input. Therefore, the *structure* technique (constructing 2^{2x-1} pairs with 2^x queries) cannot be applied. The best way is randomly generating (h_{i-1}, M) and check whether or not $\mathrm{CF}(h_{i-1}, M) \oplus \mathrm{CF}(h_{i-1} \oplus \Delta^{\mathrm{IN}}, M) = \Delta^{\mathrm{OUT}}$. The relation holds with a probability of 2^{-64}. Thus, our attack is faster than the ideal case.

4.2 12-Round Second Preimage Attack on Compression Function

Our differential attack can be converted into a second preimage attack on the compression function by using the conversion proposed at CANS 2005 by Yu *et al.* [27]. In the second preimage attack on the compression function, the attacker is given a message M, input chaining variable h, and the output of the compression function $\mathrm{CF}(h, M)$. For an ideal compression function, the probability that $\mathrm{CF}(h, M) = \mathrm{CF}(h', M')$ is satisfied for $(h, M) \neq (h', M')$ is 2^{-n}. Therefore, finding a way to choose (h', M') satisfying the above equation with a higher probability than 2^{-n} can be regarded as the second preimage attack. Note that the second preimage attack on the compression function is also discussed by Rechberger [24]. Finally, if we can find the differences $(\Delta h, \Delta M)$ where $\Pr[\mathrm{CF}(h, M) = \mathrm{CF}(h \oplus \Delta h, M \oplus \Delta M)] > 2^{-n}$, for a randomly chosen h, M, we can succeed in constructing the second preimage attack. The differential characteristic for 12-round collisions in Sect. 3 satisfies the above equation with probability 2^{-70}. At a glance, we need 2^{70} 12-round DM-PRESENT-80 computations to the second preimage attack. However, considering the early aborting technique again, our attack complexity can be much smaller. In fact, we can construct the second preimage attack on the compression function with a complexity of $2^{61.91}$ 12-round DM-PRESENT-80 computations. Due to the regulation of pages, we omit the detail of the complexity evaluation.

5 Concluding Remarks

In this paper, we presented the first third-party security analysis of reduced-round DM-PRESENT-80. The main result is a collision attack on the 12-round compression function. We constructed a differential characteristic suitable for collisions, and efficiently found paired values with the multi-inbound technique. Based on this attack, we also presented the 18-round distinguisher and 12-round second preimage attack on the compression function. Because PRESENT is one of the most successful designs for the lightweight cryptography, we believe that our results contribute to deeper understanding of lightweight designs.

Acknowledgments. The authors would like to thank anonymous reviewers for their helpful comments to improve the quality of this paper.

References

1. Bertoni, G., Daemen, J., Peeters, M., Van Assche, G.: On the indifferentiability of the sponge construction. In: Smart, N.P. (ed.) EUROCRYPT 2008. LNCS, vol. 4965, pp. 181–197. Springer, Heidelberg (2008)
2. Blondeau, C., Gérard, B.: Multiple differential cryptanalysis: Theory and practice. In: Joux, A. (ed.) FSE 2011. LNCS, vol. 6733, pp. 35–54. Springer, Heidelberg (2011)
3. Bogdanov, A., Knežević, M., Leander, G., Toz, D., Varıcı, K., Verbauwhede, I.: SPONGENT: A lightweight hash function. In: Preneel, B., Takagi, T. (eds.) CHES 2011. LNCS, vol. 6917, pp. 312–325. Springer, Heidelberg (2011)
4. Bogdanov, A., Knudsen, L.R., Leander, G., Paar, C., Poschmann, A., Robshaw, M.J.B., Seurin, Y., Vikkelsoe, C.: PRESENT: An ultra-lightweight block cipher. In: Paillier, P., Verbauwhede, I. (eds.) CHES 2007. LNCS, vol. 4727, pp. 450–466. Springer, Heidelberg (2007)
5. Bogdanov, A., Leander, G., Paar, C., Poschmann, A., Robshaw, M.J.B., Seurin, Y.: Hash functions and RFID tags: Mind the gap. In: Oswald, E., Rohatgi, P. (eds.) CHES 2008. LNCS, vol. 5154, pp. 283–299. Springer, Heidelberg (2008)
6. De Cannière, C., Rechberger, C.: Finding SHA-1 characteristics: General results and applications. In: Lai, X., Chen, K. (eds.) ASIACRYPT 2006. LNCS, vol. 4284, pp. 1–20. Springer, Heidelberg (2006)
7. Cho, J.Y.: Linear cryptanalysis of reduced-round PRESENT. In: Pieprzyk, J. (ed.) CT-RSA 2010. LNCS, vol. 5985, pp. 302–317. Springer, Heidelberg (2010)
8. Dai, Z., Wang, M., Sun, Y.: Effect of the dependent paths in linear hull. Cryptology ePrint Archive: Report 2010/325 (2010)
9. Ferguson, N.: Observations on H-PRESENT-128. Rump Session of CRYPTO 2011 (2011)
10. Gilbert, H., Peyrin, T.: Super-Sbox cryptanalysis: Improved attacks for AES-like permutations. In: Hong, S., Iwata, T. (eds.) FSE 2010. LNCS, vol. 6147, pp. 365–383. Springer, Heidelberg (2010)
11. Hermelin, M., Nyberg, K.: Linear cryptanalysis using multiple linear approximations. Cryptology ePrint Archive: Report 2011/093 (2011)
12. ISO/IEC 29192-2:2011: Information technology–Security techniques–Lightweight cryptography–Part 2: Block ciphers (2011)
13. Kobayashi, T., Hirose, S.: Collision attack on double-block length compression function using round-reduced PRESENT. In: SCIS 2012 (2012) (in Japanese)
14. Kumar, M., Yadav, P., Kumari, M.: Flaws in differential cryptanalysis of reduced round PRESENT. Cryptology ePrint Archive: Report 2010/407 (2010)
15. Lamberger, M., Mendel, F., Rechberger, C., Rijmen, V., Schläffer, M.: Rebound distinguishers: Results on the full whirlpool compression function. In: Matsui, M. (ed.) ASIACRYPT 2009. LNCS, vol. 5912, pp. 126–143. Springer, Heidelberg (2009)
16. Matusiewicz, K., Naya-Plasencia, M., Nikolić, I., Sasaki, Y., Schläffer, M.: Rebound attack on the full LANE compression function. In: Matsui, M. (ed.) ASIACRYPT 2009. LNCS, vol. 5912, pp. 106–125. Springer, Heidelberg (2009)
17. Mendel, F., Peyrin, T., Rechberger, C., Schläffer, M.: Improved cryptanalysis of the reduced Grøstl compression function, ECHO permutation and AES block cipher. In: Jacobson Jr., M.J., Rijmen, V., Safavi-Naini, R. (eds.) SAC 2009. LNCS, vol. 5867, pp. 16–35. Springer, Heidelberg (2009)
18. Mendel, F., Pramstaller, N., Rechberger, C., Rijmen, V.: The impact of carries on the complexity of collision attacks on SHA-1. In: Robshaw, M.J.B. (ed.) FSE 2006. LNCS, vol. 4047, pp. 278–292. Springer, Heidelberg (2006)

19. Mendel, F., Rechberger, C., Schläffer, M., Thomsen, S.S.: The Rebound Attack: Cryptanalysis of Reduced Whirlpool and Grøstl. In: Dunkelman, O. (ed.) FSE 2009. LNCS, vol. 5665, pp. 260–276. Springer, Heidelberg (2009)
20. Menezes, A.J., van Oorschot, P.C., Vanstone, S.A.: Handbook of applied cryptography. CRC Press (1997)
21. Nakahara Jr., J., Sepehrdad, P., Zhang, B., Wang, M.: Linear (hull) and algebraic cryptanalysis of the block cipher PRESENT. In: Garay, J.A., Miyaji, A., Otsuka, A. (eds.) CANS 2009. LNCS, vol. 5888, pp. 58–75. Springer, Heidelberg (2009)
22. Ohkuma, K.: Weak keys of reduced-round PRESENT for linear cryptanalysis. In: Jacobson Jr., M.J., Rijmen, V., Safavi-Naini, R. (eds.) SAC 2009. LNCS, vol. 5867, pp. 249–265. Springer, Heidelberg (2009)
23. Özen, O., Varıcı, K., Tezcan, C., Kocair, Ç.: Lightweight block ciphers revisited: Cryptanalysis of reduced round PRESENT and HIGHT. In: Boyd, C., González Nieto, J. (eds.) ACISP 2009. LNCS, vol. 5594, pp. 90–107. Springer, Heidelberg (2009)
24. Rechberger, C.: Second-preimage analysis of reduced SHA-1. In: Steinfeld, R., Hawkes, P. (eds.) ACISP 2010. LNCS, vol. 6168, pp. 104–116. Springer, Heidelberg (2010)
25. Wang, M.: Differential cryptanalysis of reduced-round PRESENT. In: Vaudenay, S. (ed.) AFRICACRYPT 2008. LNCS, vol. 5023, pp. 40–49. Springer, Heidelberg (2008)
26. Wang, X.: Cryptanalysis of hash functions and potential dangers. Invited Talk at CT-RSA 2006 (2006)
27. Yu, H., Wang, G., Zhang, G., Wang, X.: The second-preimage attack on MD4. In: Desmedt, Y.G., Wang, H., Mu, Y., Li, Y. (eds.) CANS 2005. LNCS, vol. 3810, pp. 1–12. Springer, Heidelberg (2005)

A Specification of DM-PRESENT-80

The lightweight block cipher PRESENT was proposed by Bogdanov *et al.* in CHES 2007 [4]. PRESENT has a 31-round SPN (Substitution and Permutation Network) construction. The block length is 64 bits and two key lengths of 80 and 128 bits are supported. The lightweight hash functions DM-PRESENT-80/-128 were also proposed by Bogdanov *et al.* at CHES 2008 [5]. The compression function of DM-PRESENT is constructed by PRESENT with the Davies-Meyer (DM) mode [20, Algorithm 9.42]. Thus, the j-th 64-bit chaining variable H_j of DM-PRESENT is updated using a 80 or 128 bits message M_j to $H_{j+1} = E(H_j, M) \oplus H_j$. $E(\cdots, K)$ indicates the encryption operation of PRESENT under a secret key K. The proposers of PRESENT recommended the 80-bit key version for applications in resource constraint environments. So we show attacks on DM-PRESENT-80 in this paper, and the details of PRESENT-80 in this section. We denote a state of 64 bits data block $X = (x_{63}, x_{62}, ..., x_0)$ by 4-by-16 matrix as;

$$\mathrm{X} = \begin{pmatrix} x_{63}, x_{59}, x_{55}, x_{51}, x_{47}, x_{43}, x_{39}, x_{35}, x_{31}, x_{27}, x_{23}, x_{19}, x_{15}, x_{11}, x_7, x_3 \\ x_{62}, x_{58}, x_{54}, x_{50}, x_{46}, x_{42}, x_{38}, x_{34}, x_{30}, x_{26}, x_{22}, x_{18}, x_{14}, x_{10}, x_6, x_2 \\ x_{61}, x_{57}, x_{53}, x_{49}, x_{45}, x_{41}, x_{37}, x_{33}, x_{29}, x_{25}, x_{21}, x_{17}, x_{13}, x_9, x_5, x_1 \\ x_{60}, x_{56}, x_{52}, x_{48}, x_{44}, x_{40}, x_{36}, x_{32}, x_{28}, x_{24}, x_{20}, x_{16}, x_{12}, x_8, x_4, x_0 \end{pmatrix}. \tag{4}$$

The round transformations of PRESENT are as follows.

- addRoundkey (AK) adds the the 64 bits round key.
- sBoxlayer (S) is a 4-to-4 bits S-box of PRESENT and applies to each vertical 4 bits. The transition by sBoxlayer is illustrated at the right side of Fig. 4.
- pLayer (P) permutes the horizontal 4 bits to the vertical 4 bits. The left side of Fig. 4 illustrates where every 4 bits group is permuted by pLayer.

Fig. 4 illustrates the transition by S and P. Then, the i-th round function F of PRESENT can be denoted by

$$X_{i+1} = F(X_i) \equiv P \circ S \circ AK(X_i). \tag{5}$$

The round keys are generated as follows. The 80 bits secret key is stored in a key register K and represented as $k_{79}k_{78}...k_0$. The i-th round key K_i $(1 \leq i \leq 32)$ consists of leftmost 64-bit of the actual content of register K. Thus the first round key K_1 is $K_1 = k_{79}k_{78}...k_{16}$. To generate next round key, the key resister K is updated as follows.

- 61 bits rotation: $[k_{79}k_{78}...k_0] = [k_{18}k_{17}...k_0k_{79}...k_{19}]$
- partial sBoxlayer: $[k_{79}k_{78}k_{77}k_{76}] = sBoxlayer[k_{79}k_{78}k_{77}k_{76}]$
- addRound_counter: $[[k_{19}k_{18}k_{17}k_{16}k_{15}] = [k_{19}k_{18}k_{17}k_{16}k_{15}] \oplus round\_counter(i)$

The $round\_counter(i)$ is the 5-bit binary representation of i. K_{32} is used for post-whitening. It is similar to the round keys that we call M_i as i-th round message from the message M.

We use the following notation to the internal states to describe our attacks: $\sharp x.y$ denotes the number of intermediate states. See Fig. 1, state $\sharp 0$, for example, indicates an input differential value, and state $\sharp 5.5$ indicates the internal state after the S transformation of the fifth round and before the P transformation of the fifth round, and state $\sharp 5.5$, for example, indicates the internal state after the S transformation of the fifth round and before the P transformation of the fifth round.

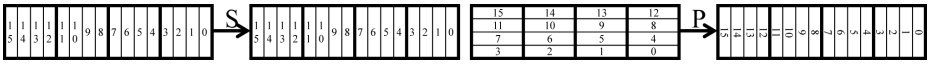

Fig. 4. The sBoxlayer and pLayer of PRESENT. Each rectangle contains 4 bits. Both transformations are operated per 4 bits.

B Differential Characteristics of S-Box

This section shows the input/output pairs of S-box that are used in our differential characteristic in detail. We note that following values of the transition are represented in hexadecimal notation. For a fixed differences $(\Delta x, \Delta y)$, where $\Delta x = x \oplus x'$ and $\Delta y = S(x) \oplus S(x')$, we searched for the input pairs (x, x') that satisfy $(\Delta x, \Delta y)$. These pairs can be searched by computing all 16×16 pairs

of (x, x') and $(S(x), S(x'))$. As a result the number of input/output pairs that satisfy one $(\Delta x, \Delta y)$ is either two, four or zero. Obviously, the number of input pairs (x, x') that satisfy $\Delta x = x \oplus x'$ is 16. Table 2 indicates all the possible input pairs (x, x') that satisfy either $(\Delta 4, \Delta 9)$, $(\Delta 9, \Delta 4)$, $(\Delta 4, \Delta 5)$, or $(\Delta 5, \Delta 4)$. The input pair $(9, D)$, for instance, satisfies the differential transition of $(\Delta 4, \Delta 9)$. The order of x and x' can be exchanged against the S-box operation. Then the symmetric input pairs $(9, D)$ and $(D, 9)$ satisfy $(\Delta x, \Delta y)$. Hence, there are two pairs that satisfy $(\Delta 4, \Delta 9)$. And there are four pairs that satisfy $(\Delta 9, \Delta 4)$ similarly. The same analysis is applied to $(\Delta 4, \Delta 5)$ and $(\Delta 5, \Delta 4)$.

Table 2. The input pairs (x, x') of S-box used in our differential characteristic

(x, x')	$\Delta input$	$\Delta output$
(9, D)	9⊕D = 4	S(9)⊕S(D) = E⊕7 = 9
(3, A)	3⊕A = 9	S(3)⊕S(A) = B⊕F = 4
(5, C)	5⊕C = 9	S(5)⊕S(C) = 0⊕4 = 4
(0, 4)	0⊕4 = 4	S(0)⊕S(4) = C⊕9 = 5
(1, 5)	1⊕5 = 4	S(1)⊕S(5) = 5⊕0 = 5
(8, D)	8⊕D = 5	S(8)⊕S(D) = 3⊕7 = 4