Open access • Book Chapter • DOI:10.1007/978-3-642-36095-4_11

Collisions for the WIDEA-8 compression function — Source link

Florian Mendel, Vincent Rijmen, Deniz Toz, Kerem Varici

Institutions: Graz University of Technology, Katholieke Universiteit Leuven

Published on: 25 Feb 2013 - The Cryptographers' Track at the RSA Conference

Topics: Key schedule, Block size, Residual block termination, CBC-MAC and Weak key

Collisions for the WIDEA-8 Compression Function

Florian Mendel[1], Vincent Rijmen[2], Deniz Toz[2], and Kerem Varıcı[2]

[1] Graz University of Technology, IAIK, Austria
[2] KU Leuven, ESAT/COSIC and iMinds, Belgium

Abstract. WIDEA is a family of block ciphers inspired by the IDEA block cipher. The design uses n-parallel instances of IDEA with an improved key schedule to obtain block ciphers with larger block sizes. Moreover, the given design is suggested as the compression function for Davies-Meyer mode. In this paper, we discuss the security of the block cipher when used as a compression function. Inspired by the weak key attacks on IDEA, we take the advantage of slow diffusion mechanism of the key schedule and present free-start collisions for WIDEA-8 which is the specified version by designers. Our results are practical and we are able to obtain free-start collisions with a complexity of $2^{13.53}$.

Keywords: hash functions, cryptanalysis, WIDEA-8

1 Introduction

Block ciphers are key components of cryptography. In the last two decades, parallel to the improvement in the technology, algorithms have evolved and more efficient and secure designs have been proposed. However, some algorithms managed to survive despite the extensive cryptanalysis. The block cipher IDEA [20], designed in 1990 by Lai and Massey, is a nice example of such an algorithm. There were various attacks on reduced round version of IDEA [4–7, 10, 13, 17], but there was no known attack for full IDEA except the discovered weak key classes [9, 11, 16]. Recently, in EUROCRYPT'12, an attack which is better than exhaustive search with a factor of four was presented [19] for the full number of rounds.

WIDEA-n [18] is a family of block ciphers which aims to extend the block size of IDEA from 64-bit to $n \times 64$-bit by improving the performance results. In addition, the key schedule of IDEA is patched to make the design more secure against existing attacks and a non-linear shift register is used rather than rotations in the subkey generation.

Related work To the best of our knowledge, no external analysis of WIDEA-n has been published so far. But as a related work, security of

the single-length and double-length hashing modes by using the IDEA as compression function is analyzed in [22]. The main idea of the analysis is to use the weak keys defined previously in [11] as an iterative characteristic for the compression function. Free-start collisions and semi-free-start collisions are obtained for the various schemes with practical complexity.

Our contribution. In this paper, we study the security of WIDEA-n block cipher when it is used as a compression function in Davies-Meyer mode. We first use an approach similar to the one described above. However, due to the changes in key schedule, we only obtain free-start collisions up to seven rounds. Then, we modify the attack according to the new key schedule. We find new iterative characteristics with high probabilities such that the basic attack strategy is still applicable. At the end, we get free-start collisions for the full (8.5 round) WIDEA-8. Furthermore, we show that two free-start collisions can be combined to get a second order differential collision. These attacks are based on the utilization of weak keys and our results are given in Table 1.

Table 1. Results for WIDEA-8

target	rounds	time	attack type	sect.
comp. function	7	1	free-start collision	§4.2
comp. function	8.5 (full)	1	free-start near-collision	§4.3
comp. function	8.5 (full)	$2^{13.53}$	free-start collision	§4.3

Outline. This paper is organized as follows. In Section 2, we give a brief description of the WIDEA-n block cipher. In Section 3, we give an overview of the weak keys and describe the previous attacks on IDEA. Then, we present our observations and describe our attack procedure in Section 4. Finally, we conclude and summarize our results in Section 5.

2 Description of WIDEA

WIDEA-n [18] is a family of block ciphers, designed by Junod and Macchetti, presented at FSE 2009. The design uses n parallel applications of the IDEA [20] round function, strengthened with a mixing layer based on an MDS matrix.

In this paper, we focus on the version with $n = 8$ since it is introduced in the original paper with full specification. WIDEA-8 accepts a 512-bit plaintext $\mathbb{X} = X_0||X_1||X_2||X_3$ and a 1024-bit user key K which can be seen as an array of eight 128-bit words as inputs, and is composed of 8.5 rounds. Throughout this paper we will use the following notation.

$\odot$	Modular multiplication in $\mathbb{Z}^*_{2^{16}+1}$
$\boxplus$	Modular addition in $\mathbb{Z}_{2^{16}}$
$\oplus$	XOR
$\lll n$	left rotation of n positions
$X^{(i)}$	The input of the i-th round

Let $X^{(i)} = X_0^{(i)}||X_1^{(i)}||X_2^{(i)}||X_3^{(i)}$ with $X_j^{(i)} = x_{j,1}^{(i)}||x_{j,2}^{(i)}||\dots||x_{j,n}^{(i)}$ and $x_{j,k}^{(i)} \in \mathbb{Z}_{2^{16}}$. Then, the round function of WIDEA-8 is given in Figure 1.

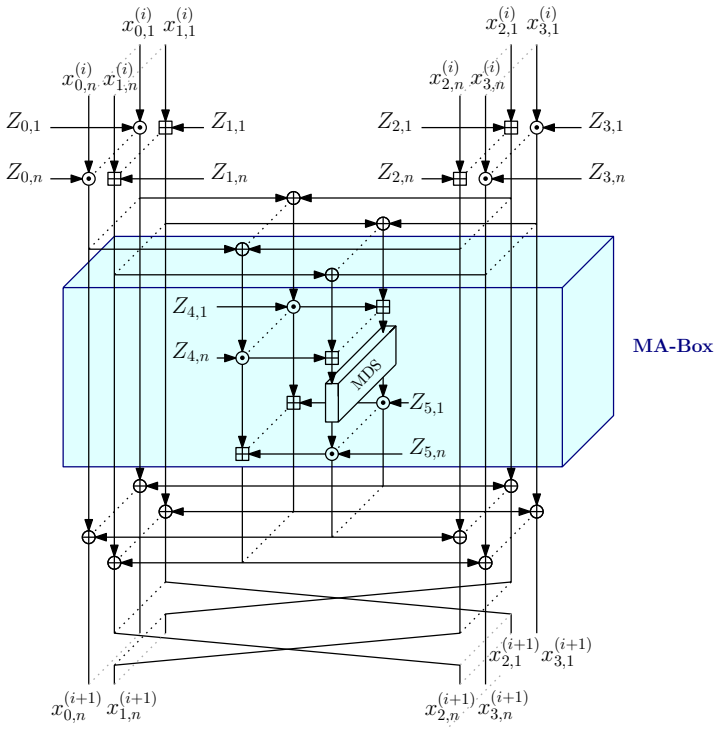

Fig. 1. The round function for WIDEA.

In the design paper, WIDEA-8 is proposed as the compression function of Davies-Meyer mode [12] and the software performance is as good as the SHA-2 family [1] and the SHA-3 finalists [2, 3, 14, 15, 23].

2.1 Key Schedule

In the IDEA block cipher, the subkeys are generated by rotating the master key which causes some weaknesses in the design. Therefore, in the key schedule of WIDEA a non-linear feedback shift register is used to generate the subkeys. Let K_i, $0 \leq i \leq 7$ be the master key; C_i, $0 \leq i \leq 6$ be the chosen constants and Z_i, $0 \leq i \leq 51$ denote the subkeys that are used in the 8.5 rounds of WIDEA-n. Then, the key schedule is given as follows:

$$
\begin{aligned}
Z_i &= K_i && 0 \leq i \leq 7 \\
Z_i &= ((((Z_{i-1} \oplus Z_{i-8}) \overset{16}{\boxplus} Z_{i-5}) \overset{16}{\lll} 5) \lll 24) \oplus C_{\frac{i}{8}-1} && 8 \leq i \leq 51, \quad 8 | i \\
Z_i &= ((((Z_{i-1} \oplus Z_{i-8}) \overset{16}{\boxplus} Z_{i-5}) \overset{16}{\lll} 5) \lll 24) && 8 \leq i \leq 51, \quad 8 \nmid i
\end{aligned}
$$

Here, each subkey Z_i has a size of $n \times 16$ bits and it can be split into the n 16-bit slices (i.e., $Z_i = z_{i,1}, \ldots, z_{i,n}$). Note that rotation by 5 bit positions is independently carried out on each slice $z_{i,j}$ for $1 \leq j \leq n$ and rotation by 24 bit positions is carried out globally for Z_i. For more detail, we refer to the specification of WIDEA [18].

3 Weak Keys for IDEA

If a key results in nonrandom behavior of the cipher it is called a weak key. For most of the ciphers, the weak keys are only a small fraction of the possible key space and hence the attacker tries to find the large set of weak classes to mount an attack. However, when a hash function is constructed from a block cipher, as in Davies-Meyer construction, the message takes the role of the key and the attacker has full control over it. Note that there exists various analysis on the weak keys for IDEA [9, 11, 16]. The ones related with our analysis are described below.

3.1 Weak Key Classes

Daemen et al. studied the classes of weak keys yielding characteristics with probability one in [11]. The nonlinear operations in the round function of

IDEA are the modulo addition in $\mathbb{Z}_{2^{16}}$ and the modular multiplication in $\mathbb{Z}^*_{2^{16}+1}$ which provide good diffusion. Therefore, the basic idea is using a pair of inputs that differ only in the most significant bit (for each 16-bit word) and finding keys that will preserve this difference after modular addition and modular multiplication. To be more precise, let $\Delta = \texttt{0x8000}$, if the key value entering the modular operation, say Z_i, equals to ± 1 in $\mathbb{Z}_{2^{16}+1}$ then the difference after the modular operation again equals to Δ. This observation puts conditions on the subkey values used in the multiplication operation. But the rest of the subkeys can be chosen freely. Using this idea, the authors presented all possible characteristics for the round function of the IDEA block cipher with the conditions on the corresponding subkeys resulting in weak key classes of size up to 2^{35} out of 2^{128}.

3.2 Application to Hashing Modes

Recently, Wei et al. studied the security of the IDEA block cipher when it is used in various single-length or double-length hashing modes [22]. They were able to generate free-start collisions, semi-free-start collisions, pseudo-preimages or even hash collisions in practical complexity for most of these modes. Their attacks are based on the weak key classes mentioned above.

The simplest collision attack in the paper uses the null key (all zeros) for the encryption and each 16-bit plaintext word has the difference $(\Delta, \Delta, \Delta, \Delta)$. As stated in [11], these differences will behave linearly and lead to the same difference in the ciphertext with probability one. This difference is then canceled with the feed-forward operation resulting in a collision for the compression or hash function depending on the hashing mode.

4 Collision Attack on WIDEA-8

In this section, we study the security of the WIDEA-8 block cipher when it is used in Davies-Meyer construction. We first describe our basic attack strategy and show how the attacks on IDEA in hashing mode can be modified to attack WIDEA-8. We then present our results and give sample collisions.

4.1 Basic Attack Strategy

Although the round function of the WIDEA-8 block cipher is more complex than that of the IDEA block cipher due to the MDS operation, the

previous attack strategy is still applicable if one is able to find an iterative characteristic that holds with high probability.

Observation 1 *The parallel instances of IDEA are only connected by the MDS matrix in the MA-box and hence if we can find a characteristic for one slice where the MA-box (see Figure 1) is never active, the attack is reduced to attacking only one slice instead of all eight.*

Based on this observation, if we have the input difference $(\Delta, \Delta, \Delta, \Delta)$ in one of the slices, then these differences in the 16-bit words cancel each other before the MDS operation and the input difference does not affect the other slices. As a result, we are able to obtain an iterative characteristic. A sample characteristic when there is a difference in the n-th slice is given in Figure 2.

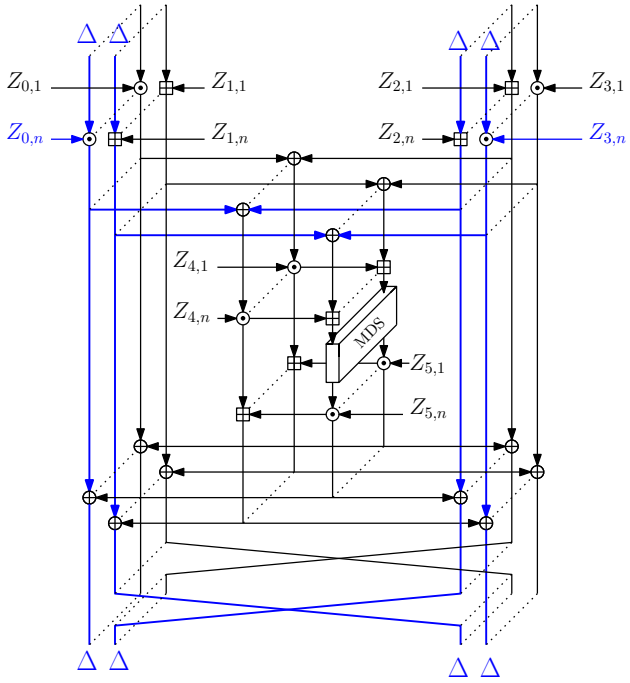

Fig. 2. A sample iterative characteristic for WIDEA-n

When an iterative characteristic is found, then due to the feed-forward operation in the Davies-Meyer mode, the output difference cancels out

with the initial difference resulting in a collision for the compression function.

Unfortunately, it is not possible to use directly the null key as in the previous attacks. Whereas the key schedule of the IDEA block cipher uses only rotations and hence is linear, as described in Section 2.1 the key schedule of the WIDEA-8 block cipher consists of addition with a constant, modular addition, rotation and xor. Therefore, even though a null key is chosen after some rounds the subkeys will have nonzero values.

4.2 Collision Attack on 7 rounds

In order to perform an attack, we want to find the longest iterative characteristic. Now, being familiar with the basic attack strategy, the existing challenge can be summarized as follows. To minimize the diffusion of the input differences, we need not only that the message words (which are used as keys) entering the multiplication have no difference but also the message words have to be zero. However, if we start with a null master key, the after three rounds all slices have nonzero values. Therefore, our aim is to find the maximum number of rounds such that all subkeys are zero at least in one of the slices. For this purpose we make use of the following observation.

Observation 2 *Given any eight consecutive subkeys* $\{Z_{i+1}, Z_{i+2}, \ldots, Z_{i+8}\}$, *it is possible to construct the whole set of subkeys.*

This allows us to start from the middle by setting the intermediate subkey values to zero and calculate forwards and backwards using the inverse key-schedule.

$$\begin{aligned} Z_i &= [([(Z_{i+8} \oplus C_{\frac{i+1}{8}-1}) \ggg 24] \overset{16}{\ggg} 5) \overset{16}{\boxminus} Z_{i+3}] \oplus Z_{i+7}, \quad 0 \le i \le 51, \quad 8|i \\ Z_i &= [([Z_{i+8} \ggg 24] \overset{16}{\ggg} 5) \overset{16}{\boxminus} Z_{i+3}] \oplus Z_{i+7}, \qquad\qquad 0 \le i \le 51, \quad 8 \nmid i \end{aligned}$$

The best results we found are obtained by setting the subkeys $Z_{25} = Z_{26} = \ldots = Z_{32} = 0$. As it can be seen from Table 2, the subkey values entering the multiplication for WIDEA-8 equals to zero for the first slice up to Z_{42}. We want to note that we focus only on the subkey values $z_{i,j} = z_{(i+3),j}$ with $i = 0, 6, 12, ..., 48$, since if this values equal to zero then the multiplication operation behaves linear and the characteristic will hold.

Table 2. Subkeys for WIDEA-8 when $Z_{25} = Z_{26} = \ldots = Z_{32} = 0$.

i	$z_{i,1}$	$z_{i,2}$	$z_{i,3}$	$z_{i,4}$	$z_{i,5}$	$z_{i,6}$	$z_{i,7}$	$z_{i,8}$
0	0000	E7FD	1444	6810	8B79	2822	47C8	0200
3	0000	E7FE	06F8	0000	0000	0000	0000	0000
6	0000	0001	F2E9	AFF7	0600	0000	0000	0000
9	0000	E7FF	FC58	0000	0000	0000	0000	0000
12	0000	F001	0520	0000	0000	0000	0000	0000
15	0000	0FFF	FAE0	0000	0000	0000	0000	0000
18	0000	F001	0520	0000	0000	0000	0000	0000
21	0000	0FFF	FAE0	0000	0000	0000	0000	0000
24	0000	F001	0520	0000	0000	0000	0000	0000
27	0000	0000	0000	0000	0000	0000	0000	0000
30	0000	0000	0000	0000	0000	0000	0000	0000
33	0000	0000	0000	0000	0000	0000	0000	0000
36	0000	0000	0000	0000	0000	0000	0000	0000
39	0000	0000	0000	0000	0000	0000	0000	0000
42	0000	0000	0000	0000	0015	E080	0B00	0000
45	4891	8264	0000	0000	0000	0000	00AF	5C00

As a result, we are able to pass seven rounds with probability one when there is the chosen difference $(\Delta, \Delta, \Delta, \Delta)$ where $\Delta = \texttt{0x8000}$ in the first slice. A collision example is given in Table 3.

4.3 Extending the attack to full WIDEA-8

In this section, we discuss how the attack on 7 rounds can be extended to full WIDEA-8. By ignoring the conditions on the message words in the first round we could find an input where most other subkeys are zero as required for the attack. In more detail, we set the subkeys $Z_{33} = Z_{34} = \ldots = Z_{40} = 0$ and find all other subkeys by computing forwards and backwards. The results are given in Table 4.

Using this as message input and the same iterative characteristic as for the attack on 7 rounds with $\Delta = \texttt{0x8000}$ we could find a free-start near-collision for full WIDEA-8. Note that due to the fact that $z_{0,8} = \texttt{0x42B4}$ another difference Δ' is needed in the chaining value $x_{0,8}$ to get the difference $\Delta = \texttt{0x8000}$ after the multiplication operation in the first round. The result is a free-start near-collision for full WIDEA-8 with complexity of 1 and only a difference in one 16-bit word at the output of the compression function after the application of the feed-forward. Moreover, we want to note that two free-start near-collisions can be combined to get a zero-sum (second-order differential collision [8, 21]) for full WIDEA-8 with a complexity of only 2 compression function evaluations.

Table 3. A collision example for seven round of WIDEA-8 in hexadecimal.

State							
5750	C1C3	1603	ADC5	2A12	DE9C	3547	1F24
5D9F	6856	D5A3	0188	808B	6D14	B0F4	58A9
6143	365B	BFDA	89DA	551B	F732	225A	FE0C
9BA8	C55E	AA2E	B4E1	3417	720D	22CF	8A28
State'							
D750	C1C3	1603	ADC5	2A12	DE9C	3547	1F24
DD9F	6856	D5A3	0188	808B	6D14	B0F4	58A9
E143	365B	BFDA	89DA	551B	F732	225A	FE0C
1BA8	C55E	AA2E	B4E1	3417	720D	22CF	8A28
$M_1 = M_2$							
0000	E7FD	1444	6810	8B79	2822	47C8	0200
0000	C7FF	67D5	2FE1	4839	0840	0000	0000
0000	D7FF	3F97	0009	931F	A917	0000	0000
0000	E7FE	06F8	0000	0000	0000	0000	0000
0000	E800	96CD	C81A	F500	0000	0000	0000
0000	D000	FC31	5803	3414	2F78	0000	0000
0000	0001	F2E9	AFF7	0600	0000	0000	0000
0000	E7FF	EC1A	5FEE	0C00	0000	0000	0000
WIDEA-8(State , M_1) $\oplus$ M_1 = WIDEA-8(State', M_2) $\oplus$ M_2							
D029	603E	368F	998F	7585	021C	492B	7DF0
BCB0	B142	15B0	B273	B503	1A6A	F410	9E4D
8F7F	BA4D	460E	8C9D	D2AD	0036	104B	43E6
E306	6246	6D73	3CDF	FD52	B205	267E	0720

However, by using differences other than $\Delta = \texttt{0x8000}$ in the chaining input we can turn the free-start near-collision into a free-start collision for the compression function. Note that this will effect the probability of the attack due to the modular additions in WIDEA-8. Remember, we have $z_{0,8} = \texttt{0x42B4}$, $z_{2,8} = \texttt{0x7e49}$, $z_{8,8} = \texttt{0x2600}$ and $z_{49,8} = \texttt{0x5E00}$.

$$(x_{2,8}^{(1)} \overset{16}{\boxplus} z_{2,8}) \oplus ((x_{2,8}^{(1)} \oplus \Delta) \overset{16}{\boxplus} z_{2,8}) = \Delta \tag{1}$$

$$(x_{8,8}^{(2)} \overset{16}{\boxplus} z_{5,8}) \oplus ((x_{8,8}^{(2)} \oplus \Delta) \overset{16}{\boxplus} z_{8,8}) = \Delta \tag{2}$$

$$(x_{49,8}^{(8)} \overset{16}{\boxplus} z_{49,8}) \oplus ((x_{49,8}^{(8)} \oplus \Delta) \overset{16}{\boxplus} z_{49,8}) = \Delta \tag{3}$$

Table 4. Subkeys for WIDEA-8 when $Z_{33} = Z_{34} = \ldots = Z_{40} = 0$.

i	$z_{i,1}$	$z_{i,2}$	$z_{i,3}$	$z_{i,4}$	$z_{i,5}$	$z_{i,6}$	$z_{i,7}$	$z_{i,8}$
0	3209	680D	AB9C	470D	6357	300A	C7C8	42B4
3	0000	BFFB	22E2	E13A	8FBC	B209	0800	0000
6	0000	2806	E120	46FD	F980	0000	0000	0000
9	0000	67FF	9C35	7EB3	3108	31C0	0400	0000
12	0000	F001	5517	790A	1080	0000	0000	0000
15	0000	27FA	E93A	9F2E	F600	0000	0000	0000
18	0000	D806	CECC	48A1	0B80	0000	0000	0000
21	0000	0FFF	72E5	CF97	FB00	0000	0000	0000
24	0000	F001	4521	1838	0680	0000	0000	0000
27	0000	0000	0000	0000	0000	0000	0000	0000
30	0000	0000	0000	0000	0000	0000	0000	0000
33	0000	0000	0000	0000	0000	0000	0000	0000
36	0000	0000	0000	0000	0000	0000	0000	0000
39	0000	0000	0000	0000	0000	0000	0000	0000
42	0000	0000	0000	0000	0000	0000	0000	0000
45	0000	0000	0000	0000	0000	0000	0000	0000
48	F7BA	0000	0000	0000	0000	0000	0000	0000
51	0000	0000	0003	C018	1C60	0100	0000	0000

We aim for differences with low Hamming weight. Moreover, we need a difference for which we can find a chaining input such that

$$(x_{0,8}^{(1)} \odot \texttt{0x42B4}) \oplus ((x_{0,8}^{(1)} \oplus \Delta) \odot \texttt{0x42B4}) = \Delta \tag{4}$$

has a solution for some $x_{0,8}^{(1)} \in \mathtt{Z}_2^{16}$, and

$$(x_{i,8}^{(j)} \odot \texttt{0x0000}) \oplus ((x_{i,8}^{(j)} \oplus \Delta) \odot \texttt{0x0000}) = \Delta \tag{5}$$

occurs with a high probability for all j where $i \geq 3$ and $i|3$.

Attack Procedure. We performed a search over all possible Δ values and found the best one (satisfying the conditions above) as $\Delta = \texttt{0x5820}$. We generate two input values $\mathbb{X}$ and $\mathbb{X}'$ as follows.

- For $\mathbb{X}$, restrict $x_{0,8}^{(1)}$ to the values that satisfy Equation (4) and choose random values for the remaining 496 bits.
- Assign $\mathbb{X}' = \mathbb{X} \oplus (0^{112}||\Delta||0^{112}||\Delta||0^{112}||\Delta||0^{112}||\Delta)$

We then compute the output of full WIDEA-8 used in Davies-Meyer mode and check whether a collision occurs or not.

Complexity of the Attack. Equations (1)−(3) each have a success probability of 2^{-4} and Equation (5) is satisfied with probability $2^{-0.09}$. Therefore the complexity of the attack can be approximated as $(2^{-0.09})^{17} \cdot (2^{-4})^3 = 2^{-13.53}$ when xor is used in the feed-forward.
As a result, after generating 2^{14} initial values, one can find a free-start collisions for WIDEA-8 with full number of rounds. In practice, we found a collision after 2^8 trials which is better than our estimated complexity. The example is given in Table 5.

Table 5. A collision example for full WIDEA-8 in hexadecimal.

State							
2C7A	0866	9F38	C148	3FB1	7BDA	0232	9054
E56C	8780	3E0D	96F3	6D1D	F028	907A	CA77
DDB6	AC09	77E4	D4C5	6715	E3CA	165A	3396
A835	DACB	CA5D	CC01	5270	F382	D7D7	7873
State'							
2C7A	0866	9F38	C148	3FB1	7BDA	0232	C874
E56C	8780	3E0D	96F3	6D1D	F028	907A	9257
DDB6	AC09	77E4	D4C5	6715	E3CA	165A	6BB6
A835	DACB	CA5D	CC01	5270	F382	D7D7	2053
$M_1 = M_2$							
3209	680D	AB9C	470D	6357	300A	C7C8	42B4
0000	5801	97F4	D0DA	0371	04E1	F400	0000
0000	7FF8	5C75	B946	131E	6335	CCF1	7E49
0000	BFFB	22E2	E13A	8FBC	B209	0800	0000
0000	4FF7	753E	2805	3E23	80E2	0C00	0000
0000	E7F9	7FC3	1818	DE12	EF37	C8F4	C1FF
0000	2806	E120	46FD	F980	0000	0000	0000
0000	4008	8FE9	8005	8A98	FF6E	F800	0000
WIDEA-8(State , M_1) $\oplus$ M_1 = WIDEA-8(State', M_2) $\oplus$ M_2							
2C06	6743	87F8	775D	8AB8	5957	226C	4F0F
626F	934B	949F	7195	333A	997A	0D1E	9A32
3D2C	3435	3861	E7CB	2198	8074	94DA	2C26
2544	AD24	4881	E8DC	2344	015F	B015	6D81

5 Conclusion and Discussion

We have implemented the attacks and found free-start collisions for Davies-Meyer mode when it is initiated with WIDEA-8 as compression function.

Since this single-length hashing mode is assumed to be secure in the ideal cipher model, it is not a good choice to use WIDEA-8 in this mode with the initially defined parameters. The easiest solution to fix this weakness seems like choosing the constant values more randomly. But still, the best way might be to use a new key schedule whose diffusion is better in the both forward and backward direction.

Acknowledgments. The work presented in this paper was done while Florian Mendel was with KU Leuven. The work has been supported in part by the IAP Programme P6/26 BCRYPT of the Belgian State (Belgian Science Policy), by the European Commission through the ICT programme under contract ICT-2007-216676 ECRYPT II, and by the Research Fund KU Leuven, OT/08/027.

References

1. Secure Hash Standard. Federal Information Processing Standard 180-4. National Institute of Standards and Technology (2012), http://csrc.nist.gov/publications/fips/
2. Aumasson, J.P., Henzen, L., Meier, W., Phan, R.C.W.: SHA-3 proposal BLAKE. Submission to NIST (Round 3) (2010), http://131002.net/blake/blake.pdf
3. Bertoni, G., Daemen, J., Peeters, M., Assche, G.V.: The Keccak SHA-3 submission. Submission to NIST (Round 3) (2011), http://keccak.noekeon.org/Keccak-submission-3.pdf
4. Biham, E., Biryukov, A., Shamir, A.: Miss in the Middle Attacks on IDEA and Khufu. In: Knudsen, L.R. (ed.) FSE. Lecture Notes in Computer Science, vol. 1636, pp. 124–138. Springer (1999)
5. Biham, E., Dunkelman, O., Keller, N.: New Cryptanalytic Results on IDEA. In: Lai, X., Chen, K. (eds.) ASIACRYPT. Lecture Notes in Computer Science, vol. 4284, pp. 412–427. Springer (2006)
6. Biham, E., Dunkelman, O., Keller, N.: A New Attack on 6-Round IDEA. In: Biryukov, A. (ed.) FSE. Lecture Notes in Computer Science, vol. 4593, pp. 211–224. Springer (2007)
7. Biham, E., Dunkelman, O., Keller, N., Shamir, A.: New Data-Efficient Attacks on Reduced-Round IDEA. IACR Cryptology ePrint Archive 2011, 417 (2011)
8. Biryukov, A., Lamberger, M., Mendel, F., Nikolic, I.: Second-Order Differential Collisions for Reduced SHA-256. In: Lee, D.H., Wang, X. (eds.) ASIACRYPT. Lecture Notes in Computer Science, vol. 7073, pp. 270–287. Springer (2011)
9. Biryukov, A., Nakahara Jr., J., Preneel, B., Vandewalle, J.: New Weak-Key Classes of IDEA. In: Deng, R.H., Qing, S., Bao, F., Zhou, J. (eds.) ICICS. Lecture Notes in Computer Science, vol. 2513, pp. 315–326. Springer (2002)
10. Borst, J., Knudsen, L.R., Rijmen, V.: Two Attacks on Reduced IDEA. In: Fumy, W. (ed.) EUROCRYPT. Lecture Notes in Computer Science, vol. 1233, pp. 1–13. Springer (1997)
11. Daemen, J., Govaerts, R., Vandewalle, J.: Weak Keys for IDEA. In: Stinson, D.R. (ed.) CRYPTO. Lecture Notes in Computer Science, vol. 773, pp. 224–231. Springer (1993)

12. Davies, D., Price, W.: Digital signatures, an update. In: 5th International Conference on Computer Communication. pp. 845–849 (1994)
13. Demirci, H.: Square-like Attacks on Reduced Rounds of IDEA. In: Nyberg, K., Heys, H.M. (eds.) Selected Areas in Cryptography. Lecture Notes in Computer Science, vol. 2595, pp. 147–159. Springer (2002)
14. Ferguson, N., Lucks, S., Schneier, B., Whiting, D., Bellare, M., Kohno, T., Callas, J., Walker, J.: The Skein Hash Function Family. Submission to NIST (Round 3) (2010), http://www.skein-hash.info/sites/default/files/skein1.3.pdf
15. Gauravaram, P., Knudsen, L.R., Matusiewicz, K., Mendel, F., Rechberger, C., Schlffer, M., Thomsen, S.S.: Grøstl – a SHA-3 candidate. Submission to NIST (Round 3) (2011), http://www.groestl.info/Groestl.pdf
16. Hawkes, P.: Differential-Linear Weak Key Classes of IDEA. In: Nyberg, K. (ed.) EUROCRYPT. Lecture Notes in Computer Science, vol. 1403, pp. 112–126. Springer (1998)
17. Junod, P.: New Attacks Against Reduced-Round Versions of IDEA. In: Gilbert, H., Handschuh, H. (eds.) FSE. Lecture Notes in Computer Science, vol. 3557, pp. 384–397. Springer (2005)
18. Junod, P., Macchetti, M.: Revisiting the IDEA Philosophy. In: Dunkelman, O. (ed.) FSE. Lecture Notes in Computer Science, vol. 5665, pp. 277–295. Springer (2009)
19. Khovratovich, D., Leurent, G., Rechberger, C.: Narrow-Bicliques: Cryptanalysis of Full IDEA. In: Pointcheval, D., Johansson, T. (eds.) EUROCRYPT. Lecture Notes in Computer Science, vol. 7237, pp. 392–410. Springer (2012)
20. Lai, X., Massey, J.L.: A Proposal for a New Block Encryption Standard. In: Damgård, I. (ed.) EUROCRYPT. Lecture Notes in Computer Science, vol. 473, pp. 389–404. Springer (1990)
21. Lamberger, M., Mendel, F.: Higher-Order Differential Attack on Reduced SHA-256. Cryptology ePrint Archive, Report 2011/037 (2011), http://eprint.iacr.org/
22. Wei, L., Peyrin, T., Sokolowski, P., Ling, S., Pieprzyk, J., Wang, H.: On the (In)Security of IDEA in Various Hashing Modes. In: Canteaut, A. (ed.) FSE. Lecture Notes in Computer Science, vol. 7549, pp. 163–179. Springer (2012)
23. Wu, H.: The Hash Function JH. Submission to NIST (round 3) (2011), http://www3.ntu.edu.sg/home/wuhj/research/jh/jh_round3.pdf