Security of Iterated Hash Functions Based on Block Ciphers

Walter Hohl Xuejia Lai Thomas Meier Christian Waldvogel

Signal and Information Processing Laboratory
Swiss Federal Institute of Technology
CH-8092 Zurich, Switzerland

Abstract. Cryptographic hash functions obtained by iterating a round function constructed from a block cipher and for which the hash-code length is twice the block length m of the underlying block cipher are considered. The computational security of such hash functions against two particular attacks, namely, the free-start target and free-start collision attacks, is investigated; these two attacks differentiate themselves from the "usual" target and collision attacks by not specifying the initial value of the iterations. The motivation is that computationally secure iterated hash functions against these two particular attacks implies computationally secure iterated hash functions against the "usual" target and collision attacks. For a general class of such $2m$-bit iterated hash functions, tighter upper bounds than the one yet published in the literature on the complexity of free-start target and free-start collision attacks are derived. A proposal for a $2m$-bit iterated hash function achieving these upper bounds is made; this new proposal is shown to be computationally more secure against free-start target and free-start collision attacks than some of the already proposed schemes falling into this general class. It is also shown that our proposal is better than the present proposal for an ISO standard in the sense that both schemes achieve these upper bounds but one encryption is required in our proposal for hashing one m-bit message block as opposed to two encryptions in the ISO proposal. Finally, two new attacks on the LOKI Double-Block-Hash function are presented with lower complexities than the known ones.

1 Introduction

A *hash function* is an easily computable mapping from the set of all binary sequences of some minimum length or greater to the set of binary sequences of some fixed length. In cryptography, hash functions are used to provide data integrity and to produce short digital signatures.

One well-known method to obtain hash function is the use of iteration. Let H_0 denote an m-bit initial value ($m > 0$) and let M denote a binary message whose length is a positive multiple of m, i.e., $M = (M_1, M_2, \ldots, M_n)$ for some positive n with M_i representing an m-bit block. [Note that one can extend the message length to a multiple of m by applying deterministic "padding"

D.R. Stinson (Ed.): Advances in Cryptology - CRYPTO '93, LNCS 773, pp. 379-390, 1994.

techniques, cf. [ISO 91, Merkle 90].] Then, we write $hash(\cdot\ ,\ \cdot)$ to denote the *m-bit iterated hash function* which, given H_0 and M, computes the hash value $hash(H_0, M) = H_n$ according to the recursive equation

$$H_i \ = \ round(H_{i-1}, M_i) \qquad\qquad i = 1, 2, \ldots, n \qquad (1)$$

where $round(\cdot\ ,\ \cdot)$ is an easily computable function from two m-bit inputs to an m-bit output. The function $round(\cdot\ ,\ \cdot)$ will be called the *m-bit round function.*

Let H_0 and $\hat{H}_0$ be two m-bit initial values and, let $M = (M_1, \ldots, M_n)$ and $\hat{M} = (\hat{M}_1, \ldots, \hat{M}_{\hat{n}})$ be two binary messages with M_i $(1 \leq i \leq n)$ and $\hat{M}_i$ $(1 \leq i \leq \hat{n})$ denoting m-bit blocks. For the iterated hash function $hash(\cdot\ ,\ \cdot)$ one can distinguish between the following five attacks, cf. [Lai 92]:

- **target attack:** Given H_0 and M, find $\hat{M}$ such that $\hat{M} \neq M$ but $hash(H_0, \hat{M}) = hash(H_0, M)$;
- **free-start target attack:** Given H_0 and M, find $\hat{H}_0$ and $\hat{M}$ such that $(\hat{H}_0, \hat{M}) \neq (H_0, M)$ but $hash(\hat{H}_0, \hat{M}) = hash(H_0, M)$;
- **collision attack:** Given H_0, find M and $\hat{M}$ such that $\hat{M} \neq M$ but $hash(H_0, \hat{M}) = hash(H_0, M)$;
- **semi-free-start collision attack:** Find H_0, M and $\hat{M}$ such that $\hat{M} \neq M$ but $hash(H_0, \hat{M}) = hash(H_0, M)$;
- **free-start collision attack:** Find H_0, $\hat{H}_0$, M and $\hat{M}$ such that $(\hat{H}_0, \hat{M}) \neq (H_0, M)$ but $hash(\hat{H}_0, \hat{M}) = hash(H_0, M)$.

When the messages M and $\hat{M}$ contain only one block, i.e., $n = \hat{n} = 1$, we have $hash(H_0, M) = round(H_0, M)$, from which it follows that each above attack reduces to an attack of the same type on the m-bit round function. For example, a target attack reads: given H_0 and M, find $\hat{M}$ such that $\hat{M} \neq M$ but $round(H_0, \hat{M}) = round(H_0, M)$.

We will consider iterated hash functions based on (m, k) block ciphers, where an (m, k) *block cipher* defines, for each k-bit key, a reversible mapping from the set of all m-bit plaintexts onto the set of all m-bit ciphertexts. Given an (m, k) block cipher, we write $\mathbf{E}_Z(X)$ to denote the encryption of the m-bit plaintext X under the k-bit key Z, and $\mathbf{D}_Z(Y)$ to denote the decryption of the m-bit ciphertext Y under the k-bit key Z. In our discussion, we will always assume that the (m, k) block cipher has no known weaknesses. We define the *rate* of such an iterated hash function (or equivalently, of an round function) as the number of m-bit message blocks processed per encryption or decryption.

Given that the m-bit round function is based on an (m, k) block cipher, we define the *complexity* of an attack as the total number of encryptions or decryptions of the (m, k) block cipher required for this attack, e.g., an attack requiring 2^s encryptions or decryptions is said to have complexity 2^s. Because an attack on the m-bit round function implies an attack of the same type on the corresponding m-bit iterated hash function with roughly the same complexity, the design of

computationally secure round functions is a necessary (but not sufficient) condition for the design of computationally secure iterated hash functions. Moreover, under certain conditions (cf. [Merkle 90, Damgaard 90, Naor 89, Lai 92]), a computationally secure round function implies a computationally secure iterated hash function. We will therefore concentrate our attention to the design of computationally secure round functions.

In Section 2 we will consider a general class of $2m$-bit iterated hash functions of rate 1 based on an (m, m) block cipher. Several previously proposed schemes [Preneel 89, Quisquater 89, Brown 90] are shown to be in this class. For this class, we derive upper bounds on the complexities of free-start target and free-start collision attacks, by describing attacks that are better than the brute-force attacks. In Section 3, we propose a $2m$-bit iterated hash function which will be proven, under plausible assumptions, to achieve these upper bounds. Section 4 contains a new free-start collision attack using two encryptions on the LOKI Double-Block-Hash scheme [Brown 90] and a new semi-free-start collision attack requiring about $2^{m/2}$ encryptions. In Section 5 we investigate a class of $2m$-bit iterated hash functions with rate $1/2$. It is shown that the upper bounds derived in Section 2 also hold for this class of rate $1/2$ hash functions. It then follows that both our proposal and the Meyer-Schilling scheme [Meyer 88] (which is presently under consideration for an ISO standard [ISO 91]) achieve the same computational security against free-start attacks; however, our proposal is more efficient in the sense that one encryption is required for hashing one m-bit message block as opposed to two encryptions in the Meyer-Schilling scheme.

2 $2m$-bit round functions with rate 1

In this section, we consider $2m$-bit round functions with rate 1 based on (m, m) block ciphers, i.e., block ciphers with m-bit ciphertext-plaintext and m-bit keys. Let the $2m$-bit hash values H_i be written as the concatenation (denoted by the symbol :) of two m-bit vectors H_i^1 and H_i^2 such that $H_i = H_i^1 : H_i^2$; similarly, let the $2m$-bit message block M_i be written as $M_i = M_i^1 : M_i^2$ with M_i^1 and M_i^2 denoting two m-bit vectors. Using this new notation, we can rewrite (1) as

$$H_i^1 : H_i^2 \quad = \quad round\left(H_{i-1}^1 : H_{i-1}^2 \ , \ M_i^1 : M_i^2\right) \quad . \tag{2}$$

One of the proposals of $2m$-bit iterated hash functions based on the $2m$-bit round function with rate 1 defined in (2) is the following:

LOKI Double Block Hash (DBH) scheme: For the $2m$-bit iterated hash function proposed in [Brown 90], the $2m$-bit round function is given by

$$\begin{cases} H_i^1 &= \mathbf{E}_{H_{i-1}^1 \oplus M_i^1}(H_{i-1}^1 \oplus M_i^2) \quad \oplus \quad H_{i-1}^1 \oplus H_{i-1}^2 \oplus M_i^2 \\ H_i^2 &= \mathbf{E}_{H_{i-1}^2 \oplus M_i^2}(H_{i-1}^1 \oplus M_i^1 \oplus H_i^1) \quad \oplus \quad H_{i-1}^1 \oplus H_{i-1}^2 \oplus M_i^1 \end{cases} \tag{3}$$

with the symbol $\oplus$ denoting bitwise modulo-2 addition.

Other similar proposals are the Preneel-Bosselaers-Govaerts-Vandewalle (PBGV) scheme proposed in [Preneel 89] and the Quisquater-Girault (QG) scheme proposed in [Quisquater 89]. The purpose of these schemes was to obtain secure $2m$-bit round functions by modifying the apparently secure m-bit round function proposed by Davies and by Meyer [Davies 85, Matyas 85, Winternitz 84]. However, several recent works (cf. [Quisquater 89, Miyaguchi 91, Lai 92, Preneel 93]) and the attacks on the LOKI-DBH scheme that will be presented in this paper show that these proposals of $2m$-bit round functions are in fact weaker than the underlying m-bit round function against free-start attacks. In order to give a systematic solution to this problem, we will consider the following general form of such $2m$-bit round functions:

General form of the $2m$-bit round function with rate 1:

$$\begin{cases} H_i^1 &= \mathbf{E}_A(B) \oplus C \\ H_i^2 &= \mathbf{E}_R(S) \oplus T \end{cases} \tag{4}$$

where A, B and C are binary linear combinations of the m-bit vectors H_{i-1}^1, H_{i-1}^2, M_i^1 and M_i^2, and where R, S and T are some (not necessarily binary linear) combinations of the vectors H_{i-1}^1, H_{i-1}^2, M_i^1, M_i^2 and H_i^1. We can therefore write A, B and C in matrix-form as

$$\begin{bmatrix} A \\ B \\ C \end{bmatrix} = \begin{bmatrix} a_1 & a_2 & a_3 & a_4 \\ b_1 & b_2 & b_3 & b_4 \\ c_1 & c_2 & c_3 & c_4 \end{bmatrix} \begin{bmatrix} H_{i-1}^1 \\ H_{i-1}^2 \\ M_i^1 \\ M_i^2 \end{bmatrix} \tag{5}$$

for some binary values a_i, b_i and c_i $(1 \leq i \leq 4)$.

One can easily see that for the LOKI-DBH scheme we have

$$\begin{bmatrix} A \\ B \\ C \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 1 \end{bmatrix} \begin{bmatrix} H_{i-1}^1 \\ H_{i-1}^2 \\ M_i^1 \\ M_i^2 \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} R \\ S \\ T \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} H_{i-1}^1 \\ H_{i-1}^2 \\ M_i^1 \\ M_i^2 \\ H_i^1 \end{bmatrix}.$$

The PBGV and QG schemes can also be represented in a similar way.

We now show upper bounds on the complexity of a free-start target and free-start collision attacks on $2m$-bit iterated hash functions whose round function is of type (4).

Proposition 1: For the $2m$-bit iterated hash function with rate 1 whose $2m$-bit round function is of type (4), the complexity of a free-start target attack is upper-bounded by about 2^m encryptions, and the complexity of a free-start collision attack is upper-bounded by about $2^{m/2}$.

Proof: Since there are 2^{4m} possible values for the $4m$-bit vector $(H_{i-1}^1, H_{i-1}^2, M_i^1, M_i^2)$ and 2^{3m} possible values for the $3m$-bit vector (A, B, C), it follows that,

for each possible value of (A, B, C), there are at least $\frac{2^{4m}}{2^{3m}} = 2^m$ values of $(H_{i-1}^1, H_{i-1}^2, M_i^1, M_i^2)$ satisfying (5). The problem of finding these 2^m values is computationally negligible, because from (5) we see that this problem is equivalent to solving a system of three linear equations with four unknowns.

Free-start target attack: For a given value of $(H_{i-1}^1, H_{i-1}^2, M_i^1, M_i^2)$, we will find a different value of $(H_{i-1}^1, H_{i-1}^2, M_i^1, M_i^2)$ yielding the same value of (H_i^1, H_i^2) according to (4). We proceed as followed: for the given value of $(H_{i-1}^1, H_{i-1}^2, M_i^1, M_i^2)$ we compute the value of (A, B, C) according to (5) and, using the above argument we produce 2^m different values of $(H_{i-1}^1, H_{i-1}^2, M_i^1, M_i^2)$ yielding the same value for (A, B, C) as the given value. We then compute the value of H_i^2 for the given value and for each of the 2^m newly produced values of $(H_{i-1}^1, H_{i-1}^2, M_i^1, M_i^2)$. Because there are 2^m possible values of the m-bit block H_i^2, it follows that one must compute H_i^2 for about 2^m different values of $(H_{i-1}^1, H_{i-1}^2, M_i^1, M_i^2)$ to have an 0.63 probability of finding a value of $(H_{i-1}^1, H_{i-1}^2, M_i^1, M_i^2)$ yielding the same value for H_i^2 as the given value of $(H_{i-1}^1, H_{i-1}^2, M_i^1, M_i^2)$. Such an attack requires therefore about 2^m encryptions, which gives an upper bound of about 2^m for the complexity of a free-start target attack on the $2m$-bit iterated hash function.

Free-start collision attack: We will find two different values for $(H_{i-1}^1, H_{i-1}^2, M_i^1, M_i^2)$ yielding the same value of (H_i^1, H_i^2) according to (4). We proceed as follows: we first produce $2^{m/2}$ values for $(H_{i-1}^1, H_{i-1}^2, M_i^1, M_i^2)$ yielding the same value for H_i^1. Because there are 2^m possible values for the m-bit block H_i^2, it follows from the usual "birthday argument" that one must compute H_i^2 for about $2^{m/2}$ values of $(H_{i-1}^1, H_{i-1}^2, M_i^1, M_i^2)$ to have an 0.63 probability of finding two values of $(H_{i-1}^1, H_{i-1}^2, M_i^1, M_i^2)$ yielding the same hash value H_i^2. Thus, such an attack requires about $2^{m/2}$ encryptions, which gives an upper bound of about $2^{m/2}$ for the complexity of a free-start collision attack on the $2m$-bit iterated hash function. □

Remark. The basic idea behind the attacks in Proposition 1 is to attack the two equations in (4) separately. If one can find many values for $(H_{i-1}^1, H_{i-1}^2, M_i^1, M_i^2)$ yielding the same value for (A, B, C), then the attack on the $2m$-bit round function of type (4) is reduced to an attack on one m-bit round function. Thus, similar attacks as the ones described in the proof of Proposition 1 will also work, even if the mapping from $(H_{i-1}^1, H_{i-1}^2, M_i^1, M_i^2)$ to (A, B, C) in (4) is not a binary linear combination. Therefore, Proposition 1 also holds if it is easy to find 2^m different values of $(H_{i-1}^1, H_{i-1}^2, M_i^1, M_i^2)$ for a given value of (A, B, C).

Given the $2m$-bit iterated hash function $hash(\cdot\ ,\ \cdot)$ whose $2m$-bit round function is of type (4), we say that $hash(\cdot\ ,\ \cdot)$ is *optimum against a free-start target attack* when the best possible free-start target attack has complexity about 2^m; similarly, $hash(\cdot\ ,\ \cdot)$ is said to be *optimum against a free-start collision attack* when the best possible free-start collision attack has complexity about $2^{m/2}$.

3 Proposal for a $2m$-bit hash function with rate 1

In this section, we propose a new $2m$-bit iterated hash function whose $2m$-bit round function is of type (4). We will prove that this new proposal is optimum against a free-start target and free-start collision attacks.

Before introducing our proposal, we describe the m-bit iterated hash function which was proposed independently by Davies and Meyer, cf. [Davies 85, Matyas 85, Winternitz 84].

Davies-Meyer (DM) scheme: This scheme consists of an m-bit iterated hash function as defined in (1) whose m-bit round function is based on an (m,k) block cipher. For our purpose, we will assume that $k = m$, i.e., the plaintext-ciphertext length and the key length are the same. Letting H_{i-1} and M_i denote two m-bit blocks, the m-bit output H_i of the DM round function for the input pair (H_{i-1}, M_i) is given by

$$H_i = \mathbf{E}_{M_i}(H_{i-1}) \oplus H_{i-1} \quad . \tag{6}$$

The DM scheme is generally considered to be secure, i.e., a free-start target and free-start collision attacks on (6) need about 2^m and $2^{m/2}$ encryptions, respectively. Our proposal is based on the DM scheme:

Parallel Davies-Meyer (Parallel-DM) scheme: For the $2m$-bit iterated hash function defined in (1), we propose the following $2m$-bit round function with rate 1 based on an $(m, m,)$ block cipher:

$$\begin{cases} H_i^1 = \mathbf{E}_{M_i^1 \oplus M_i^2}(H_{i-1}^1 \oplus M_i^1) \oplus H_{i-1}^1 \oplus M_i^1 \\ H_i^2 = \mathbf{E}_{M_i^1}(H_{i-1}^2 \oplus M_i^2) \oplus H_{i-1}^2 \oplus M_i^2 \end{cases} \quad . \tag{7}$$

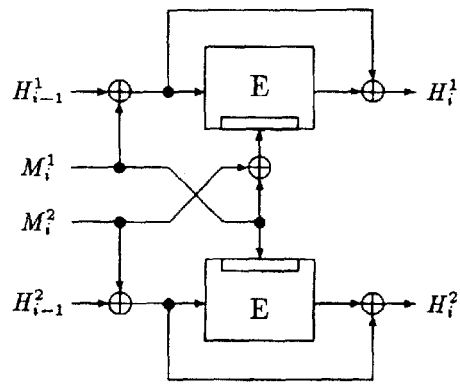

Fig. 1. *The $2m$-bit round function of the proposed Parallel-DM scheme (the small boxes indicate the key input of the block cipher).*

In order to avoid trivial attacks based on the fact that a falsified message can have a different length from that of the genuine message, we defined the following strengthening on the iterated hash functions, which was proposed independently by Merkle and Damgaard, cf. [Merkle 90, Damgaard 90]:

Merkle-Damgaard (MD) Strengthening: For the iterated hash function defined by (1), the MD-strengthening consists of specifying that the last block M_n of the binary message $M = (M_1, M_2, \ldots, M_n)$ to be hashed must represent the length of M (in binary form), i.e., the total length of $(M_1, M_2, \ldots, M_{n-1})$.

Proposition 2: Assuming that the DM scheme is secure and that the $2m$-bit iterated hash function is used with MD-strengthening, it follows that the Parallel-DM scheme is optimum against free-start target and free-start collision attacks.

Proof: By applying the invertible transformation
$(H^1_{i-1}, H^2_{i-1}, M^1_i, M^2_i) \rightarrow (h^1_{i-1}, h^2_{i-1}, m^1_i, m^2_i)$
such that

$$\begin{array}{lllll} h^1_{i-1} & = & H^1_{i-1} \oplus M^1_i, & h^2_{i-1} & = & H^2_{i-1} \oplus M^2_i, \\ m^1_i & = & M^1_i, & m^2_i & = & M^1_i \oplus M^2_i \end{array}$$

to the Parallel-DM round function defined in (7), we obtain the new $2m$-bit round function

$$\begin{cases} h^1_i & = & \mathbf{E}_{m^2_i}(h^1_{i-1}) & \oplus & h^1_{i-1} \\ h^2_i & = & \mathbf{E}_{m^1_i}(h^2_{i-1}) & \oplus & h^2_{i-1} \end{cases} . \tag{8}$$

From (8) we see that h^1_i depends only on m^2_i and h^1_{i-1} and, h^2_i depends only on m^1_i and h^2_{i-1}, which implies that h^1_i and h^2_i can be attacked separately. Moreover, we see from (8) that the equations for h^1_i and h^2_i correspond each to the m-bit round function of the DM scheme defined in (6). By the assumption that the DM scheme is secure, it follows that the best possible free-start target and collision attacks on (8) require about 2^m and $2^{m/2}$ encryptions, respectively. From the *Transformation principle*, viz. applying any simple (in both directions) invertible transformation to the input and to the output of the round function produces a new round function with the same computational security as the original one against free start attacks (cf. [Lai 92]), it follows that the best possible free-start target and collision attacks on the Parallel-DM round function defined by (7) require about 2^m and $2^{m/2}$ encryptions, respectively. Because a free-start attack on an iterated hash function with MD-strengthening is roughly as hard as an attack of the same type on its round function (cf.[Merkle 90, Damgaard 90, Naor 89, Lai 92]), we have completed the proof. □

4 New attacks on the LOKI-DBH scheme

We describe here a new free-start collision attack on the LOKI-DBH scheme which requires two encryptions and a semi-free-start collision attack

which uses about $2^{m/2}$ encryptions. These attacks can be applied to the LOKI-DBH scheme with any underlying (m, m) block cipher. Such low attacking complexities on the LOKI-DBH scheme have not yet being reported in the literature. Moreover, the low complexity of this new free-start collision shows that the LOKI-DBH scheme is *not* optimum against a free-start collision attack.

By applying the invertible transformation
$(H^1_{i-1}, H^2_{i-1}, M^1_i, M^2_i) \rightarrow (h^1_{i-1}, h^2_{i-1}, m^1_i, m^2_i)$
such that

$$\begin{array}{lllll} h^1_{i-1} & = & H^1_{i-1}, & h^2_{i-1} & = & H^2_{i-1}, \\ m^1_i & = & M^1_i \oplus H^1_{i-1}, & m^2_i & = & M^2_i \oplus H^2_{i-1} \end{array} \tag{9}$$

to the LOKI-DBH round function defined in (3), we obtain the new $2m$-bit round function

$$\begin{cases} h^1_i & = & \mathbf{E}_{m^1_i}(h^1_{i-1} \oplus h^2_{i-1} \oplus m^2_i) & \oplus & h^1_{i-1} \oplus m^2_i \\ h^2_i & = & \mathbf{E}_{m^2_i}(m^1_i \oplus h^1_i) & \oplus & h^2_{i-1} \oplus m^1_i \end{cases} . \tag{10}$$

Free-start collision attack: We will find two different values for $(H^1_{i-1}, H^2_{i-1}, M^1_i, M^2_i)$ yielding the same value for (H^1_i, H^2_i) according to (3). We proceed as followed:

Step 1: We randomly choose an m-bit value x and two distinct m-bit values y and $\hat{y}$.
Step 2: Computing z and $\hat{z}$ such that

$$z = \mathbf{E}_y(x \oplus y) \oplus x \oplus y \tag{11}$$

$$\hat{z} = \mathbf{E}_{\hat{y}}(x \oplus \hat{y}) \oplus x \oplus \hat{y} \tag{12}$$

we obtain two different values $(z, x \oplus z, y, y)$ and $(\hat{z}, x \oplus \hat{z}, \hat{y}, \hat{y})$ for $(h^1_{i-1}, h^2_{i-1}, m^1_i, m^2_i)$.
Step 3: By applying the inverse transformation of (9), we obtain two different values $(z, x \oplus z, y \oplus z, x \oplus y \oplus z)$ and $(\hat{z}, x \oplus \hat{z}, \hat{y} \oplus \hat{z}, x \oplus \hat{y} \oplus \hat{z})$ for $(H^1_{i-1}, H^2_{i-1}, M^1_i, M^2_i)$.

Note that substituting $(z, x \oplus z, y, y)$ for $(H^1_{i-1}, H^2_{i-1}, M^1_i, M^2_i)$ in (3) gives

$$\begin{cases} H^1_i & = & \mathbf{E}_y(x \oplus y) & \oplus & y \oplus z \\ H^2_i & = & \mathbf{E}_y(y \oplus H^1_i) & \oplus & x \oplus y \oplus z \end{cases}$$

which, using the expression for z defined in (11), gives

$$\begin{cases} H^1_i & = & x \\ H^2_i & = & \mathbf{E}_y(H^1_i \oplus y) \oplus \mathbf{E}_y(x \oplus y) \end{cases} .$$

Replacing H^1_i by x in the right side of the second equation gives $(H^1_i, H^2_i) = (x, 0)$. In exactly the same manner, it can be shown that the substitution of $(\hat{z}, x \oplus \hat{z}, \hat{y} \oplus \hat{z}, x \oplus \hat{y} \oplus \hat{z})$ for $(H^1_{i-1}, H^2_{i-1}, M^1_i, M^2_i)$ in (3) also yields $(H^1_i, H^2_i) = (x, 0)$, which proves the correctness of our attack. Because an attack on the round function implies an attack of the same type on the iterated hash function

with the same complexity, we conclude that the above attack implies a free-start collision attack on the LOKI-DBH iterated hash function requiring two encryptions.

Semi-free-start collision attack: We will find a value for (H_{i-1}^1, H_{i-1}^2) and two different values for (M_i^1, M_i^2) yielding the same value (H_i^1, H_i^2) according to (3). We proceed as followed:

Step 1: We randomly choose an m-bit value x.
Step 2: Let z and $\hat{z}$ be defined by (11) and (12), respectively. Given x, we randomly choose a pair $(y, \hat{y})$ of two distinct m-bit values until we find a pair yielding matching z and $\hat{z}$, i.e., $z = \hat{z}$. By the usual "birthday argument", it takes about $2^{m/2}$ encryptions to have an 0.63 probability of finding such a pair $(y, \hat{y})$.
Step 3: By applying the inverse transformation of (9) we obtain a value $(z, x \oplus z)$ for (H_{i-1}^1, H_{i-1}^2) and two different values $(y \oplus z, x \oplus y \oplus z)$ and $(\hat{y} \oplus z, x \oplus \hat{y} \oplus z)$ for (M_i^1, M_i^2).

By applying similar substitutions as for the free-start collision attack, one can easily prove the correctness of this attack. We then conclude that the just described attack implies a semi-free-start collision attack on the LOKI-DBH $2m$-bit iterated hash function using about $2^{m/2}$.

5 $2m$-bit round functions with rate 1/2

In this section, we consider $2m$-bit round functions with rate 1/2 based on (m, m) block ciphers. Letting the $2m$-bit hash values H_i be written as the concatenation (denoted by the symbol :) of two m-bit vectors H_i^1 and H_i^2 such that $H_i = H_i^1 : H_i^2$, we can rewrite (1) as

$$H_i^1 : H_i^2 \quad = \quad round\left(H_{i-1}^1 : H_{i-1}^2 \ , \ M_i\right) \quad . \tag{13}$$

where M_i denotes an m-bit message block.

General form of the $2m$-bit round function with rate 1/2:

$$\begin{cases} H_i^1 & = & \mathbf{E}_A(B) \ \oplus \ C \\ H_i^2 & = & \mathbf{E}_R(S) \ \oplus \ T \end{cases} \tag{14}$$

where A, B and C are binary linear combinations of the m-bit vectors H_{i-1}^1, H_{i-1}^2 and M_i, and where R, S and T are some (not necessarily binary linear) combinations of the vectors H_{i-1}^1, H_{i-1}^2, M_i and H_i^1. We can therefore write A, B and C in matrix-form as

$$\begin{bmatrix} A \\ B \\ C \end{bmatrix} \quad = \quad \prod \quad \begin{bmatrix} H_{i-1}^1 \\ H_{i-1}^2 \\ M_i \end{bmatrix} \tag{15}$$

where $\prod$ denotes a 3×3 matrix whose entries are 0's and 1's.

From Section 2, we know that the complexity of a $2m$-bit iterated hash function with *rate* 1 whose round function is of type (4) is upper-bounded by about 2^m for a free-start target attack and by about $2^{m/2}$ for a free-start collision attack. We now show that the same upper-bounds hold for $2m$-bit iterated hash function with *rate* 1/2 whose round function is of type (14).

Proposition 3: For the $2m$-bit iterated hash function with rate 1/2 whose $2m$-bit round function is of type (14), the complexity of a free-start target attack is upper-bounded by about 2^m and the complexity of a free-start collision attack is upper-bounded by about $2^{m/2}$.

Proof: We first consider the **free-start target attack**, i.e., for a given value of $(H_{i-1}^1, H_{i-1}^2, M_i)$, we will find a different value for $(H_{i-1}^1, H_{i-1}^2, M_i)$ yielding the same value for (H_i^1, H_i^2) according to (14). When the matrix $\prod$ defined in (15) is non-singular, let D be the value of H_i^1 for the given value of $(H_{i-1}^1, H_{i-1}^2, M_i)$. We then generate 2^m different values of $(H_{i-1}^1, H_{i-1}^2, M_i)$ yielding the same value D by first computing $C = D \oplus \mathbf{E}_A(B)$ for 2^m randomly chosen values of (A, B) and then, for each value of (A, B, C), by computing $(H_{i-1}^1, H_{i-1}^2, M_i)$ according to

$$\begin{bmatrix} H_{i-1}^1 \\ H_{i-1}^2 \\ M_i \end{bmatrix} = \prod^{-1} \begin{bmatrix} A \\ B \\ C \end{bmatrix} ,$$

where $\prod^{-1}$ denotes the inverse of the non-singular matrix $\prod$. When the matrix $\prod$ is singular, there exist, for the value of (A, B, C) obtained from the given value of $(H_{i-1}^1, H_{i-1}^2, M_i)$, at least 2^m different values of $(H_{i-1}^1, H_{i-1}^2, M_i)$ yielding the same value for (A, B, C), i.e., the same value for H_i^1. For the given and the 2^m newly generated values of $(H_{i-1}^1, H_{i-1}^2, M_i)$, we compute the value of H_i^2 according to (14). Because there are 2^m possible values of the m-bit block H_i^2, it follows that one must compute H_i^2 for about 2^m different values of $(H_{i-1}^1, H_{i-1}^2, M_i)$ to have an 0.63 probability of finding a value of $(H_{i-1}^1, H_{i-1}^2, M_i)$ yielding the same value for H_i^2 as the given value of $(H_{i-1}^1, H_{i-1}^2, M_i)$. Such an attack requires therefore about 2^m encryptions.

We now consider the **free-start collision attack**, i.e. we will find two different values of $(H_{i-1}^1, H_{i-1}^2, M_i)$ yielding the same value for (H_i^1, H_i^2) according to (14). This attack is similar to the free-start target attack just described, except that here, one only generates $2^{m/2}$ values of $(H_{i-1}^1, H_{i-1}^2, M_i)$ yielding the same value of H_i^1. This follows from the usual "birthday paradox" which says that one only needs to try $2^{m/2}$ randomly chosen values of $(H_{i-1}^1, H_{i-1}^2, M_i)$ to have an 0.63 probability of finding two values of $(H_{i-1}^1, H_{i-1}^2, M_i)$ yielding the same value for H_i^2. □

Example: Meyer-Schilling scheme (modified)

In [Meyer 88], Meyer and Schilling proposed a $2m$-bit hash function based on the (m=64, k=56) block cipher DES, which was later named as MDC-2 in [Matyas 91] and which is presently under consideration as an ISO standard [ISO 91]. The Meyer-Schilling scheme, after some minor modifications, namely,

using an (m, m) block cipher instead of DES and not considering the additional "cut-and-paste" and swapping operations, can be written as

$$H_i^1 = \mathbf{E}_{H_{i-1}^1}(M_i) \oplus M_i$$
$$H_i^2 = \mathbf{E}_{H_{i-1}^2}(M_i) \oplus M_i$$

or, under its general form (14), as

$$\begin{bmatrix} A \\ B \\ C \end{bmatrix} = \begin{bmatrix} 1\,0\,0 \\ 0\,0\,1 \\ 0\,0\,1 \end{bmatrix} \begin{bmatrix} H_{i-1}^1 \\ H_{i-1}^2 \\ M_i \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} R \\ S \\ T \end{bmatrix} = \begin{bmatrix} 0\,1\,0 \\ 0\,0\,1 \\ 0\,0\,1 \end{bmatrix} \begin{bmatrix} H_{i-1}^1 \\ H_{i-1}^2 \\ M_i \end{bmatrix}.$$

Thus, the upper bounds of Proposition 3 also hold for the Meyer-Schilling scheme. By applying the similar approach as in the proof of Proposition 2, we can show that the Meyer-Schilling scheme indeed achieves these upper bound if the underlying cipher has no weaknesses.

6 Conclusion

In this paper, we have derived upper bounds for the complexities of free-start target and free-start collision attacks on a large class of $2m$-bit iterated hash functions based on an (m, m) block cipher. Eventhough these free-start target and free-start collision attacks are "non-real" attacks (because the initial value of the iterated hash function is usually fixed), their complexities give a lower bound for the complexities of "real" target and collision attacks, respectively.

We have also proposed a $2m$-bit iterated hash function with rate 1 which, under the assumption that the DM scheme is secure and that MD-strengthening is applied, was proven to be optimum against free-start target and free-start collision attacks. Moreover, our new free-start collision attack on the LOKI-DBH scheme shows that this scheme is not optimum against a free-start target collision attack. Finally, even though the Meyer-Schilling $2m$-bit iterated hash function is also optimum with respect to these two attacks, it only achieves a rate of $1/2$, while our proposal achieves a rate of 1, i.e., two encryptions are needed in the Meyer-Schilling scheme to hash one m-bit message block as opposed to one encryption for our proposal.

Since the m-bit Davies-Meyer scheme appears to be secure, it has been an open question [Preneel 93, Lai 92] whether one can modify the DM scheme to construct a $2m$-bit iterated hash function that is more secure than the original m-bit DM scheme. This problem is partly solved by the results of Propositions 1 and 3, which show that, given a $2m$-bit round function which hashes at least one m-bit message block by using twice an m-bit block cipher with an m-bit key, there always exist free-start attacks that are better than the brute-force attacks. That is, by using any (m, m) block cipher twice plus some "simple operations", it is impossible to obtain a $2m$-bit round function of rate $1/2$ or greater that is more secure than the m-bit DM round function against free-start attacks. Thus,

to obtain such a secure $2m$-bit round function, one has to apply an (m, m) block cipher at least three times in each round, or, to use twice an m-bit block cipher with key length greater than m. In fact, it appears that [Lai 92] secure $2m$-bit round functions of rate 1/2 can be constructed by using twice an m-bit block cipher with a $2m$-bit key.

References

[Brown 90] L. Brown, J. Pieprzyk and J. Seberry, *"LOKI - A Cryptographic Primitive for Authentication and Secrecy Applications"*, Advances in Cryptology - AUSCRYPT'90 Proceedings, pp. 229-236, Springer-Verlag, 1990.

[Damgaard 90] I.B. Damgaard, *"A Design Principle for Hash Functions"*, Advances in Cryptology - CRYPT0'89 Proceedings, pp. 416-427, Springer-Verlag, 1990.

[Davies 85] R.W. Davies and W.L. Price, *"Digital Signature - an Update"*, Proc. International Conference on Computer Communications, Sydney, Oct. 1984, Elsevier, North Holland, pp. 843-847, 1985.

[ISO 91] ISO/IEC CD 10118, *Information technology - Security techniques - Hash-functions*, I.S.O., 1991.

[Lai 92] X. Lai and J.L. Massey, *"Hash Functions Based on Block Ciphers"*, Advances in Cryptology - EUROCRYPT'92 Proceedings, pp. 55-70, LNCS 658, Springer-Verlag, 1993.

[Matyas 85] S.M. Matyas, C.H. Meyer and J. Oseas, *"Generating Strong One-Way Functions with Cryptographic Algorithm"*, IBM Technical Disclosure Bulletin, Vol. 27, No. 10A, pp. 5658-5659, March 1985.

[Matyas 91] S.M. Matyas, *"Key Processing with Control Vectors"*, Journal of Cryptology, Vol.3, No.2, pp. 113-136, 1991.

[Merkle 90] R.C. Merkle, *"One-Way Hash Functions and DES"*, Advances in Cryptology - CRYPTO'89 Proceedings, pp. 428-446, Springer-Verlag, 1990.

[Meyer 88] C. H. Meyer and M. Schilling, *"Secure Program Code with Modification Detection Code"*, Proceedings of SECURICOM 88, pp. 111-130, SEDEP.8, Rue de la Michodies, 75002, Paris, France.

[Miyaguchi 91] S. Miyaguchi, K. Ohta and M. Iwata, *Confirmation that Some Hash Functions Are Not Collision Free*, Advances in Cryptology-EUROCRYPT '90, Proceedings, LNCS 473, pp. 326-343, Springer Verlag, Berlin, 1991.

[Naor 89] M. Naor and M. Yung, *"Universal One-way Hash Functions and Their Cryptographic Applications"*, Proc. 21 Annual ACM Symposium on Theory of Computing, Seattle, Washington, May 15-17, 1989, pp. 33-43.

[Preneel 89] B. Preneel, A. Bosselaers, R. Govaerts and J. Vandewalle, *"Collision-Free Hashfunctions Based on Blockcipher Algorithm"*, Proceedings of 1989 International Carnahan Conference on Security Technology, pp.203-210, 1989.

[Preneel 93] B. Preneel, *Analysis and Design of Cryptographic Hash Hashfunctions* , Ph.D thesis, Katholieke Universiteit Leuven, Belgium, January 1993.

[Quisquater 89] J.J. Quisquater and M. Girault, *"2n-bit Hash Functions Using n-bit Symmetric Block Cipher Algorithms"*, Abstracts of EUROCRYPT'89.

[Winternitz 84] R.S. Winternitz, *"Producing One-Way Hash Function from DES"*, Advances in Cryptology - CRYPTO'83 Proceedings, pp. 203-207, Plenum Press, New York, 1984.