A Block-Cipher-Based Hash Function Using an MMO-Type Double-Block Compression Function

Shoichi Hirose[1] and Hidenori Kuwakado[2]

[1] Graduate School of Engineering, University of Fukui, Japan
[2] Faculty of Informatics, Kansai University, Japan

Abstract. Methods to construct a hash function using an existing block cipher recently attract some interests as an approach to implement a hash function on constrained devices. It is often required to construct a hash function whose output length is larger than that of the underlying block cipher to provide sufficient level of collision resistance with the use of an existing block cipher. This article presents a new mode of double-block compression function, which is based on the mode proposed by Jonsson and Robshaw at PKC 2005. The mode can be instantiated with a block cipher whose key-length is larger than its block-length such as AES-192/256, PRESENT-128, etc. This article also provides provable security analyses to an iterated hash function using the proposed mode and the MDP domain extension. The security properties discussed are collision resistance, preimage resistance, pseudorandom-function property of the keyed-via-IV mode, and the indifferentiability from a random oracle.

1 Introduction

Background. A cryptographic hash function transforms strings of arbitrary length to strings of fixed length. It usually consists of a compression function and domain extension. A compression function is a function from strings of fixed length to strings of fixed smaller length. Domain extension specifies how to process input strings of arbitrary length using a given compression function. A cryptographic hash function of this type is called an iterated hash function.

Most of the iterated hash functions are classified into two types according to their compression-function construction: block-cipher-based and permutation-based. The methods to construct block-cipher-based compression functions are further classified into dedicated and using existing block ciphers. The former includes most of the widely deployed or well-known hash functions such as MDx [25,26], SHA-x [8], Whirlpool [24] and so on. On the other hand, the latter attracts some interests as an approach to implement a hash function on constrained devices [4,27]. This is the topic of this article.

The collision resistance of a hash function producing n-bit digests is at most $O(2^{n/2})$ due to the birthday attack. To provide sufficient level of collision resistance with the use of existing block ciphers, it is necessary to construct a compression function whose output length is larger than that of the underlying block ciphers. There have been several proposals for modes to construct double-block

S.S.M. Chow et al. (Eds.): ProvSec 2014, LNCS 8782, pp. 71–86, 2014.

compression functions [6,11,13,15]. One line of research is to present a general model and discuss security properties in a unified way [22]. We are interested in another line of research: identifying modes of practical interest.

Our Contribution. We first present a mode of compression function based on the mode proposed by Jonsson and Robshaw [13]. Then, we provide provable security analyses to an iterated hash function using the proposed compression function and the MDP domain extension [12] in terms of collision resistance (CR), preimage resistance (PR), pseudorandomness as a function (PRF), and indifferentiability from a random oracle (IRO).

CR, PR and IRO are discussed in the ideal cipher model, and PRF is discussed in the standard model. Birthday-type lower bounds are given to its CR and IRO. These bounds are optimal up to some constant factors for this kind of iterated hash functions. A lower bound optimal up to a constant factor is also given to its PR. The keyed-via-IV (KIV) mode is shown to be a PRF if the underlying block cipher is a pseudorandom permutation (PRP) under rather mild related-key attacks.

The proposed mode requires an underlying block cipher with its key length larger than its block length, which is similar to that of abreast-/tandem-DM [15] and Hirose mode [11]. The advantage of the proposed mode over them is that the key input of the underlying block cipher only receives the chaining value. It prevents attackers from manipulating the key inputs directly. It also enables the reduction of the PRF property of the hash function to the PRP property of the underlying block cipher. The advantage of the proposed mode over MDC-2/4 [6] is that the security reductions are settled and, in particular, optimal security levels (up to some constants) are achieved for CR, PR and IRO.

Related Work. Security properties such as collision resistance and preimage resistance of existing double-block modes have also been analysed in the ideal cipher model. Steinberger gave a lower bound on CR of MDC-2 [28], which is quite lower than the birthday bound. Optimal birthday-type lower bounds were obtained on CR of abreast-DM and Hirose modes [9,11,16]. A nearly optimal lower bound was obtained on CR for tandem-DM [19]. Optimal lower bounds on PR were obtained for abreast-DM, tandem-DM and Hirose modes [1].

Özen and Stam [22] presented a general model of double-block modes using one or two calls to a $2n$-bit-key and n-bit-block block cipher, and discussed CR and PR of the modes in this model. Strictly, our analysis of CR is not covered by theirs since our analysis accepts a block cipher with variable-length key. Furthermore, they discussed neither IRO nor PRF.

There are some proposals to construct double-block iterated hash functions using a block cipher. Naito [21] proposed a scheme using a $2n$-bit-key and n-bit-block block cipher. He also presented a birthday-type lower bound on IRO of the hash functions in the ideal cipher model. Kuwakado and Hirose [14] proposed a scheme suitable for lightweight block ciphers. They discussed the preimage resistance of the hash function and the PRF property of its keyed mode in the standard model. Lee and Stam [18] recently showed that the iterated hash function using the double-block compression function called MJH [17] has asymptotically optimal collision resistance in the ideal cipher model.

Organization. Section 2 gives some notations and definitions of security properties used and discussed in the paper. The proposed double-block mode is presented in Sect. 3. The iterated hash function composed of the compression function with the MDP domain extension is also presented in this section. Collision resistance and preimage resistance are discussed in Sect. 4. Pseudorandomness of the KIV mode is discussed in Sect. 5. IRO is discussed in Sect. 6.

2 Preliminaries

2.1 Notations

Let $\boldsymbol{F}(\mathcal{X}, \mathcal{Y})$ be the set of all functions with domain $\mathcal{X}$ and range $\mathcal{Y}$. Let $\boldsymbol{P}(\mathcal{X})$ be the set of all permutations on $\mathcal{X}$. Let $\mathcal{BC}(n, \kappa)$ be the set of all (n, κ) block ciphers, where n and κ represent their block size and key size, respectively.

Let $\Sigma = \{0, 1\}$. Let $\Sigma^* = \bigcup_{i=0}^{\infty} \Sigma^i$, $(\Sigma^n)^+ = \bigcup_{i=1}^{\infty} \Sigma^{ni}$, and $(\Sigma^n)^{\leq i} = \bigcup_{j=0}^{i} \Sigma^{nj}$.

For binary strings x and y, let $x \| y$ be their concatenation. For simplicity, for $M_1, M_2, \ldots, M_l \in \Sigma^n$, $M_1 \| M_2 \| \cdots \| M_l$ will be denoted by $M_{[1,l]}$ or $M_1 M_2 \cdots M_l$.

Let ϕ be the permutation on Σ^k defined by $\phi(x_{\mathrm{L}} \| x_{\mathrm{R}}) = x_{\mathrm{R}} \| x_{\mathrm{L}}$ for every x_{L} and x_{R} in $\Sigma^{k/2}$.

2.2 Collision Resistance and Preimage Resistance

Let H^E be a hash function using a block cipher E. The collision resistance and preimage resistance of a block-cipher-based hash function are often discussed in the ideal cipher model [3]. We follow this convention.

In the ideal cipher model, the underlying block cipher E is assumed to be uniformly distributed over $\mathcal{BC}(n, \kappa)$. An encryption/decryption operation is an encryption/decryption query to the oracle E. Without loss of generality, it is assumed that an adversary does not make any query to which it already knows the answer.

Let A be an adversary trying to find a collision for H^E, that is, a pair of distinct inputs mapped to the same output by H^E. The col-advantage of A against H^E is given by

$$\mathrm{Adv}_{H^E}^{\mathrm{col}}(A) = \Pr[A^E = (M, M') \wedge H^E(M) = H^E(M') \wedge M \neq M'] \ ,$$

where E is uniformly distributed over $\mathcal{BC}(n, \kappa)$. It is assumed that A makes all the queries necessary to compute $H^E(M)$ and $H^E(M')$. Let $\mathrm{Adv}_{H^E}^{\mathrm{col}}(q)$ be the maximum col-advantage over all adversaries asking at most q queries.

Let A be an adversary trying to find a preimage of a given output v for H^E. The pre-advantage of A against H^E is given by

$$\mathrm{Adv}^{\mathrm{pre}}_{H^E}(A) = \Pr[A^E(v) = M \wedge H^E(M) = v] \ ,$$

where E is uniformly distributed over $\mathcal{BC}(n, \kappa)$. It is assumed that A makes all the queries necessary to compute $H^E(M)$. Let $\mathrm{Adv}^{\mathrm{pre}}_{H^E}(q)$ be the maximum pre-advantage over all adversaries asking at most q queries.

2.3 Pseudorandom Function and Permutation (PRF & PRP)

Let $f \in \boldsymbol{F}(\mathcal{K} \times \mathcal{X}, \mathcal{Y})$ be a keyed function from $\mathcal{X}$ to $\mathcal{Y}$ with key space $\mathcal{K}$. Let A be an adversary which has oracle access to a function from $\mathcal{X}$ to $\mathcal{Y}$ and outputs 0 or 1. The prf-advantage of A against f is given by

$$\mathrm{Adv}^{\mathrm{prf}}_{f}(A) = \left|\Pr[A^{f_K} = 1] - \Pr[A^{\rho} = 1]\right| \ ,$$

where K is uniformly distributed over $\mathcal{K}$ and ρ is uniformly distributed over $\boldsymbol{F}(\mathcal{X}, \mathcal{Y})$.

Let $f \in \boldsymbol{F}(\mathcal{K} \times \mathcal{X}, \mathcal{X})$ be a keyed function. Then, the prp-advantage of A against f is given by

$$\mathrm{Adv}^{\mathrm{prp}}_{f}(A) = \left|\Pr[A^{f_K} = 1] - \Pr[A^{\rho} = 1]\right| \ ,$$

where K is uniformly distributed over $\mathcal{K}$ and ρ is uniformly distributed over $\boldsymbol{P}(\mathcal{X})$.

2.4 PRF & PRP under Related-Key Attacks

The PRF and PRP under related-key attacks are formalized by Bellare and Kohno [2]. Let $\Phi \subset \boldsymbol{F}(\mathcal{K}, \mathcal{K})$. Let A be an adversary which has oracle access to $g(\mathsf{key}(\cdot, K), \cdot)$, where $g \in \boldsymbol{F}(\mathcal{K} \times \mathcal{X}, \mathcal{Y})$, $K \in \mathcal{K}$ and $\mathsf{key} \in \boldsymbol{F}(\Phi \times \mathcal{K}, \mathcal{K})$ such that $\mathsf{key}(\varphi, K) = \varphi(K)$. A asks a pair of $\varphi \in \Phi$ and $x \in \mathcal{X}$ as a query, and obtains $g(\varphi(K), x)$. For simplicity, $g(\mathsf{key}(\cdot, K), \cdot)$ is denoted by (g, K). The prf-rka-advantage of A against $f \in \boldsymbol{F}(\mathcal{K} \times \mathcal{X}, \mathcal{Y})$ restricted by Φ is given by

$$\mathrm{Adv}^{\mathrm{prf\text{-}rka}}_{\Phi, f}(A) = \left|\Pr[A^{(f,K)} = 1] - \Pr[A^{(\rho,K)} = 1]\right| \ ,$$

where K is uniformly distributed over $\mathcal{K}$ and ρ is uniformly distributed over $\boldsymbol{F}(\mathcal{K} \times \mathcal{X}, \mathcal{Y})$.

Let $\boldsymbol{P}(\mathcal{K} \times \mathcal{X}, \mathcal{X})$ be the set of all keyed permutations on $\mathcal{X}$ with key space $\mathcal{K}$. The prp-rka-advantage of A against $f \in \boldsymbol{F}(\mathcal{K} \times \mathcal{X}, \mathcal{X})$ restricted by Φ is given by

$$\mathrm{Adv}^{\mathrm{prp\text{-}rka}}_{\Phi, f}(A) = \left|\Pr[A^{(f,K)} = 1] - \Pr[A^{(\rho,K)} = 1]\right| \ ,$$

where K is uniformly distributed over $\mathcal{K}$ and ρ is uniformly distributed over $\boldsymbol{P}(\mathcal{K} \times \mathcal{X}, \mathcal{X})$.

2.5 Indifferentiability from Random Oracle

The notion of indifferentiability is introduced by Maurer et al. [20] as a generalized notion of indistinguishability. It is tailored to security analysis of hash functions by Coron et al. [7].

Let C be an algorithm with oracle access to an ideal primitive $\mathcal{F}$. In the setting of this article, C is an algorithm to construct a hash function using $\mathcal{F}$ with fixed input length. Let $\mathcal{H}$ be a variable-input-length (VIL) random oracle and S be a simulator which has oracle access to $\mathcal{H}$. $S^{\mathcal{H}}$ tries to behave like $\mathcal{F}$ in order to convince an adversary that $\mathcal{H}$ is $C^{\mathcal{F}}$. Let A be an adversary with access to two oracles. The indiff-advantage of A against C with respect to S is given by

$$\mathrm{Adv}^{\mathrm{indiff}}_{C,S}(A) = \left|\Pr[A^{C^{\mathcal{F}},\mathcal{F}} = 1] - \Pr[A^{\mathcal{H},S^{\mathcal{H}}} = 1]\right| .$$

3 Construction

Let $E \in \mathcal{BC}(n,k)$, where k is an even integer such that $n \leq k \leq 2n$. We consider constructions of an iterated hash function with the following compression function $F : \Sigma^k \times \Sigma^n \to \Sigma^k$ based on E:

$$F(h_i, M_i) = \mathsf{tr}_{k/2}(E_{h_i}(M_i) \oplus M_i) \| \mathsf{tr}_{k/2}(E_{h_i}(\sigma(M_i)) \oplus \sigma(M_i)) .$$

$\sigma : \Sigma^n \to \Sigma^n$ is an involution with no fixed points, that is, $\sigma = \sigma^{-1}$ and $\sigma(M_i) \neq M_i$ for any $M_i \in \Sigma^n$. $\mathsf{tr}_{k/2} : \Sigma^n \to \Sigma^{k/2}$ outputs $k/2$ least significant bits of the input. F is depicted in Fig. 1. It is based on the mode proposed by Jonsson and Robshaw [13], and its upper or right half has the structure of the Matyas-Meyer-Oseas (MMO) mode. It can be instantiated with AES with 256-bit or 192-bit key.

MDP [12] is adopted for domain extension. Let π be a permutation on Σ^k with at most few fixed points. For $1 \leq i \leq N$, let $M_i \in \Sigma^n$. $F^{\circ}_{\pi} : \Sigma^k \times (\Sigma^n)^+ \to \Sigma^k$ is an iterated hash function such that $F^{\circ}_{\pi}(IV, M_1 \| \cdots \| M_N) = h_N$, where $h_0 = IV$ is a fixed initial value, $h_i = F(h_{i-1}, M_i)$ for $1 \leq i \leq N-1$, and $h_N = F(\pi(h_{N-1}), M_N)$. Notice that $h_1 = F(\pi(IV), M_1)$ if $N = 1$. For $M \in \Sigma^*$, an unambiguous padding function $\mathsf{pad} : \Sigma^* \to (\Sigma^n)^+$ is necessary to apply F°_{π} to M. F°_{π} is illustrated in Fig. 2.

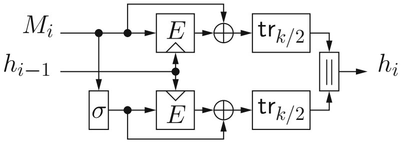

Fig. 1. Compression function F

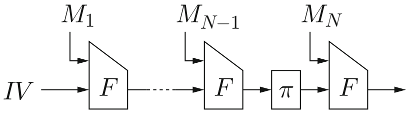

Fig. 2. Hash function F°_{π}

4 Collision Resistance and Preimage Resistance

In this section, the collision resistance and preimage resistance of F_π° is evaluated in the ideal cipher model. The followings are assumed here:

- When adversary A makes an encryption query (K, X), A receives Y such that $E_K(X) = Y$ and also gets for free $Y' = E_K(\sigma(X))$.
- When A makes a decryption query (K, Y), A receives X such that $E_K(X) = Y$ and also gets for free $Y' = E_K(\sigma(X))$.

4.1 Collision Resistance

The theorem given below implies that the collision resistance of F_π° is optimal up to some constant factor.

Theorem 1. *For* $1 \leq q < 2^{n-1}$,

$$\mathrm{Adv}_{F_\pi^\circ}^{\mathrm{col}}(q) \leq \frac{q}{2^{k/2}(1 - q/2^{n-1})} + \frac{q^2 + 2q}{2^k(1 - q/2^{n-1})^2} \ .$$

Example 1. The upper bound of Theorem 1 is 0.5 if $q = 2^{125.7}$ for $(n, k) = (128, 256)$ and if $q = 2^{94.5}$ for $(n, k) = (128, 192)$.

It is easy to see that $\mathrm{Adv}_{F_\pi^\circ}^{\mathrm{col}}(q) \leq \mathrm{Adv}_F^{\mathrm{col}}(q) + \mathrm{Adv}_F^{\mathrm{pre}}(q)$. Upper bounds on $\mathrm{Adv}_F^{\mathrm{col}}(q)$ and $\mathrm{Adv}_F^{\mathrm{pre}}(q)$ are given in Lemmas 1 and 2, respectively. The upper bound on $\mathrm{Adv}_F^{\mathrm{pre}}(q)$ is not so tight but suffices for our purpose.

Lemma 1. *For* $1 \leq q < 2^{n-1}$,

$$\mathrm{Adv}_F^{\mathrm{col}}(q) \leq \frac{q}{2^{k/2}(1 - q/2^{n-1})} + \left(\frac{q}{2^{k/2}(1 - q/2^{n-1})}\right)^2 \ .$$

Proof. Let A be any collision-finding adversary against F asking at most q queries to E. For $1 \leq i \leq q$, making the i-th query, adversary A obtains some (K_i, X_i, Y_i) and $(K_i, \sigma(X_i), Y_i')$ such that $E_{K_i}(X_i) = Y_i$ and $E_{K_i}(\sigma(X_i)) = Y_i'$. Let $W_i = \mathsf{tr}_{k/2}(Y_i \oplus X_i) \| \mathsf{tr}_{k/2}(Y_i' \oplus \sigma(X_i)))$.

Let $\mathsf{Col}_{1,i}$ be the event that $W_i = \phi(W_i)$. Let $\mathsf{Col}_{2,i}$ be the event that $W_i \in \bigcup_{j=1}^{i-1}\{W_j, \phi(W_j)\}$. If A succeeds in finding a collision for F just after the i-th query, then either $\mathsf{Col}_{1,i}$ or $\mathsf{Col}_{2,i}$ occurs. For the two events,

$$\Pr[\mathsf{Col}_{1,i}] \leq \frac{2^{n-k/2}}{2^n - (2i-1)} \text{ and } \Pr[\mathsf{Col}_{2,i}] \leq \frac{(2^{n-k/2})^2 2(i-1)}{(2^n - (2i-2))(2^n - (2i-1))} \ .$$

The probability that A finds a collision for F is bounded above by

$$\begin{aligned}\sum_{i=1}^{q}(\Pr[\mathsf{Col}_{1,i}] + \Pr[\mathsf{Col}_{2,i}]) &\leq \frac{2^{n-k/2}q}{2^n - (2q-1)} + \frac{(2^{n-k/2})^2 q(q-1)}{(2^n - (2q-2))(2^n - (2q-1))} \\ &\leq \frac{2^{n-k/2}q}{2^n - 2q} + \left(\frac{2^{n-k/2}q}{2^n - 2q}\right)^2 \ .\end{aligned}$$

□

Lemma 2. *For* $1 \le q < 2^{n-1}$,

$$\mathrm{Adv}_F^{\mathrm{pre}}(q) \le \frac{2q}{2^k(1-q/2^{n-1})^2} \ .$$

Proof. Let A be any preimage-finding adversary against F asking at most q queries to E. For $1 \le i \le q$, making the i-th query, adversary A obtains some (K_i, X_i, Y_i) and $(K_i, \sigma(X_i), Y_i')$ such that $E_{K_i}(X_i) = Y_i$ and $E_{K_i}(\sigma(X_i)) = Y_i'$. Let $W_i = \mathsf{tr}_{k/2}(Y_i \oplus X_i) \| \mathsf{tr}_{k/2}(Y_i' \oplus \sigma(X_i)))$.

Let T be the given digest. Let Pre_i be the event that $W_i = T$ or $\phi(W_i) = T$. Then,

$$\Pr[\mathsf{Pre}_i] \le \frac{(2^{n-k/2})^2 \cdot 2}{(2^n - (2i-2))(2^n - (2i-1))} \ .$$

The probability that A finds a preimage of T for F is bounded above by

$$\sum_{i=1}^{q} \Pr[\mathsf{Pre}_i] \le \frac{(2^{n-k/2})^2 \cdot 2q}{(2^n - (2q-2))(2^n - (2q-1))} \le \frac{(2^{n-k/2})^2 \cdot 2q}{(2^n - 2q)^2} \ .$$

□

4.2 Preimage Resistance

With the technique of "super query" introduced by [19], it can also be proved that the preimage resistance of F_π° is optimal up to a constant factor in the ideal cipher model.

Theorem 2.

$$\mathrm{Adv}_{F_\pi^\circ}^{\mathrm{pre}}(q) \le \frac{q}{2^{k-4}(1-2^{1-n})} \ .$$

Proof. Let A be any preimage-finding adversary against F asking at most q queries to E. Here, we call the queries normal queries. It is assumed that, if A makes 2^{n-2} normal queries with respect to a key, then it is given for free the remaining 2^{n-1} pairs of plaintexts and ciphertexts with respect to the same key. This event is called a super query.

Let PreN be the event that a preimage is obtained by some normal query. Let PreS be the event that a preimage is obtained by some super query. Then,

$$\mathrm{Adv}_{F_\pi^\circ}^{\mathrm{pre}}(q) \le \Pr[\mathsf{PreN}] + \Pr[\mathsf{PreS}] \ .$$

For PreN, the probability that a preimage is obtained by a normal query is at most $(2^{n-k/2}/2^{n-1})^2 \cdot 2 = 1/2^{k-3}$. Since A makes at most q normal queries, $\Pr[\mathsf{PreN}] \le q/2^{k-3}$.

On the other hand, for PreS, the probability that a preimage is obtained by a super query is at most

$$\frac{2^{n-k/2}}{2^{n-1}} \cdot \frac{2^{n-k/2}}{2^{n-1}-1} \cdot 2 \cdot 2^{n-2} \le \frac{2^{n+1}}{2^k(1-2^{1-n})} \ .$$

Since A makes at most $q/2^{n-2}$ super queries, $\Pr[\mathsf{PreS}] \le q/(2^{k-3}(1-2^{1-n}))$. □

5 Keyed Hashing Mode

We consider a keyed hashing mode of F°_{π}: keyed-via-IV (KIV) mode. It is obtained simply by replacing the initial value IV with secret key K, that is, $F^{\circ}_{\pi}(K,\cdot)$, where $K \in \Sigma^k$.

For this mode, it is assumed that the inputs satisfy the following property. Let $\mathcal{M} \subset (\Sigma^n)^+$ be the domain of the KIV mode of F°_{π}. For any positive integer l, for any $M_{[1,l]}$ and $M'_{[1,l]}$ in $\mathcal{M} \cap \Sigma^{nl}$, $M_l \neq \sigma(M'_l)$ if $M_{[1,l-1]} = M'_{[1,l-1]}$. Let us call this property σ-free. It is easy to see that the KIV mode of F°_{π} cannot be a PRF if its domain is not σ-free.

The following theorem implies that the KIV mode of F°_{π} is a PRF if E is a PRP under related-key attacks with respect to $\mathsf{Rel} = \{id, \phi, \pi, \pi \circ \phi\}$, where id is the identity permutation on Σ^k. Let $P_{\pi,\phi} = \{x \in \Sigma^k \mid \pi(x) = x \vee \pi(x) = \phi(x)\}$.

Theorem 3. *Let A be a prf-adversary against the KIV mode of F°_{π}. Suppose that the domain of the KIV mode of F°_{π} is σ-free. Suppose that A runs in time at most τ, and makes at most q queries, and each query has at most ℓ message blocks. Suppose that $q \leq \lambda 2^n/\mathrm{e}$ for some positive constant $\lambda < 1$, where e is the base of the natural logarithm. Then, there exists a prp-rka-adversary B against E such that*

$$\mathrm{Adv}^{\mathrm{prf}}_{F^{\circ}_{\pi}}(A) \leq \ell q \cdot \mathrm{Adv}^{\mathrm{prp\text{-}rka}}_{\mathsf{Rel},E}(B) + \ell\, q \left(\frac{|P_{\pi,\phi}|}{2^k} + \frac{1}{2^{k/2}} \right) + \frac{\ell\, 2^{k/2}}{1-\lambda} \left(\frac{\mathrm{e}\, q}{2^n} \right)^{2^{n-k/2}+1} .$$

B makes at most q queries restricted by Rel and runs in time at most $\tau + O(\ell q T_E)$, where T_E represents the time required to compute E.

It is easy to make $P_{\pi,\phi}$ small. For example, $P_{\pi,\phi}$ is empty if $\pi(x_{\mathrm{L}} \| x_{\mathrm{R}}) = (x_{\mathrm{L}} \oplus c_{\mathrm{L}}) \| (x_{\mathrm{R}} \oplus c_{\mathrm{R}})$, where $x_{\mathrm{L}}, x_{\mathrm{R}}, c_{\mathrm{L}}, c_{\mathrm{R}} \in \Sigma^{k/2}$ and c_{L} and c_{R} are distinct constants.

The last term of the upper bound in Theorem 3 is $\Omega(1)$ for $\sqrt{\ell}\, q = \Omega(2^{n/2})$ if $k = 2n$. If $k = 2n - 2c$ for some constant c, then it is $\Omega(1)$ for $\ell^{1/(2^c+1)} q = \Omega(2^{n/(1+2^{-c})})$.

Theorem 3 directly follows from the succeeding three lemmas.

Let A be an adversary with access to m oracles (u_1, K_1), (u_2, K_2), $\ldots$, (u_m, K_m), where $u_i \in \boldsymbol{F}(\mathcal{K} \times \mathcal{X}, \mathcal{Y})$ and $K_i \in \mathcal{K}$ for $1 \leq i \leq m$. Each query by A is directed to just one of the m oracles. Let us define the following notation: $\langle (u_j, K_j) \rangle_{j=1}^{m} = (u_1, K_1), (u_2, K_2), \ldots, (u_m, K_m)$. The m-prf-rka-advantage of A against h under Φ-related-key attacks is defined as follows:

$$\mathrm{Adv}^{m\text{-prf-rka}}_{\Phi,h}(A) = \left| \Pr[A^{\langle (h,K_j) \rangle_{j=1}^{m}} = 1] - \Pr[A^{\langle (\rho_j,K_j) \rangle_{j=1}^{m}} = 1] \right| ,$$

where K_j's are independent random variables uniformly distributed over $\mathcal{K}$, and ρ_j's are independent random keyed functions uniformly distributed over $\boldsymbol{F}(\mathcal{K} \times \mathcal{X}, \mathcal{Y})$.

Lemma 3. *Suppose that there are q balls and t bins. Each ball is placed in a bin chosen independently and uniformly at random. Let m be a positive integer* *and λ be a real such that $0 < \frac{\mathrm{e}\,q}{m\,t} \leq \lambda < 1$. Then, some bin contains m or more balls with probability at most*

$$\frac{t}{1-\lambda}\left(\frac{\mathrm{e}\,q}{m\,t}\right)^{m} \quad .$$

Proof. Omitted due to the page limit. □

Lemma 4. *Let $f(K,x) = \mathsf{tr}_{k/2}(E_K(x) \oplus x)$. Let A be a prf-adversary against the KIV mode of F°_{π}. Suppose that the domain of the KIV mode of F°_{π} is σ-free. Suppose that A runs in time at most τ, and makes at most q queries, and each query has at most ℓ message blocks. Then, there exists a prf-rka-adversary B against f with access to q oracles such that*

$$\mathrm{Adv}^{\mathrm{prf}}_{F^{\circ}_{\pi}}(A) \leq \ell \cdot \mathrm{Adv}^{q\text{-}\mathrm{prf}\text{-}\mathrm{rka}}_{\mathsf{Rel},f}(B) + \ell\,q\left(\frac{|P_{\pi,\phi}|}{2^k} + \frac{1}{2^{k/2}}\right) \quad .$$

B makes at most q queries restricted by Rel *and runs in time at most $\tau + O(\ell q T_E)$, where T_E represents the time required to compute E.*

Proof. For $i \in \{0,1,\ldots,\ell\}$ ($\ell \geq 1$), let $I_i : (\Sigma^n)^{\leq \ell} \to \Sigma^k$ be a random function such that

$$I_i(M_{[1,l]}) = \begin{cases} \alpha_0(M_{[1,l]}) & \text{if } 1 \leq l \leq i, \\ F^{\circ}_{\pi}(\alpha_1(M_{[1,i]}), M_{[i+1,l]}) & \text{if } i+1 \leq l \leq \ell \end{cases} ,$$

where α_0 and α_1 are independent and random functions; α_0 is uniformly distributed over $\boldsymbol{F}((\Sigma^n)^{\leq i}, \Sigma^k)$, and α_1 is uniformly distributed over

$$\{\alpha \,|\, \alpha \in \boldsymbol{F}((\Sigma^n)^i, \Sigma^k) \text{ and } \alpha(M_{[1,i-1]} \| \sigma(M_i)) = \phi(\alpha(M_{[1,i]}))\} \quad .$$

Notice that α_0 and α_1 are independent and random elements uniformly distributed over Σ^k if $i = 0$. Then,

$$\mathrm{Adv}^{\mathrm{prf}}_{F^{\circ}_{\pi}}(A) = \left|\Pr[A^{I_0} = 1] - \Pr[A^{I_\ell} = 1]\right| \quad .$$

A prf-rka-adversary B with q oracles $\langle (u_j, K_j) \rangle_{j=1}^{q}$ is constructed using A as a subroutine. B first selects $i \in \{1,\ldots,\ell\}$ uniformly at random. Then, B runs A. B simulates a random function β uniformly distributed over $\boldsymbol{F}((\Sigma^n)^{\leq i-1}, \Sigma^k)$ via lazy sampling. B answers to the t-th query of A, $M^{(t)} = M^{(t)}_{[1,l]}$, as follows:

1. If $1 \leq l \leq i-1$, then B returns $\beta(M^{(t)})$.
2. Suppose that $i \leq l \leq \ell$. Let

$$p = \min\left\{t' \,|\, t' < t \wedge \left(M^{(t')}_{[1,i-1]} = M^{(t)}_{[1,i-1]} \vee M^{(t')}_{[1,i-1]} = M^{(t)}_{[1,i-2]} \| \sigma(M^{(t)}_{i-1})\right)\right\} \quad .$$

   (a) Suppose that $l = i$. If $p \neq \bot$, then B returns
      - $u_p(\pi(K_p), M^{(t)}_i) \| u_p(\pi(K_p), \sigma(M^{(t)}_i))$ if $M^{(p)}_{[1,i-1]} = M^{(t)}_{[1,i-1]}$, and

- $u_p(\pi(\phi(K_p)), M_i^{(t)}) \| u_p(\pi(\phi(K_p)), \sigma(M_i^{(t)}))$ if $M_{[1,i-1]}^{(p)} = M_{[1,i-2]}^{(t)} \| \sigma(M_{i-1}^{(t)})$.

  Otherwise, B returns $u_t(\pi(K_t), M_i^{(t)}) \| u_t(\pi(K_t), \sigma(M_i^{(t)}))$.

(b) Suppose that $i+1 \le l \le \ell$. If $p \ne \perp$, then B returns
- $F_\pi^\circ(u_p(K_p, M_i^{(t)}) \| u_p(K_p, \sigma(M_i^{(t)})), M_{[i+1,l]}^{(t)})$ if $M_{[1,i-1]}^{(p)} = M_{[1,i-1]}^{(t)}$, and
- $F_\pi^\circ(u_p(\phi(K_p), M_i^{(t)}) \| u_p(\phi(K_p), \sigma(M_i^{(t)})), M_{[i+1,l]}^{(t)})$ if $M_{[1,i-1]}^{(p)} = M_{[1,i-2]}^{(t)} \| \sigma(M_{i-1}^{(t)})$.

  Otherwise, B returns $F_\pi^\circ(u_t(K_t, M_i^{(t)}) \| u_t(K_t, \sigma(M_i^{(t)})), M_{[i+1,l]}^{(t)})$.

Now, suppose that B is given oracles $\langle (f, K_j) \rangle_{j=1}^q$, where K_j's are independent random variables uniformly distributed over Σ^k. Then,

$$u_p(\pi(K_p), M_i^{(t)}) \| u_p(\pi(K_p), \sigma(M_i^{(t)})) = F_\pi^\circ(K_p, M_i^{(t)})$$
$$u_p(\pi(\phi(K_p)), M_i^{(t)}) \| u_p(\pi(\phi(K_p)), \sigma(M_i^{(t)})) = F_\pi^\circ(\phi(K_p), M_i^{(t)})$$

and

$$F_\pi^\circ(u_p(K_p, M_i^{(t)}) \| u_p(K_p, \sigma(M_i^{(t)})), M_{[i+1,l]}^{(t)}) = F_\pi^\circ(K_p, M_{[i,l]}^{(t)})$$
$$F_\pi^\circ(u_p(\phi(K_p), M_i^{(t)}) \| u_p(\phi(K_p), \sigma(M_i^{(t)})), M_{[i+1,l]}^{(t)}) = F_\pi^\circ(\phi(K_p), M_{[i,l]}^{(t)}) \ .$$

Therefore, we can say that A has oracle access to I_{i-1}, and

$$\Pr\left[B^{\langle (f,K_j) \rangle_{j=1}^q} = 1\right] = \frac{1}{\ell} \sum_{i=1}^{\ell} \Pr[A^{I_{i-1}} = 1] \ .$$

Next, suppose that B has oracle access to $\langle (\rho_j, K_j) \rangle_{j=1}^q$, where ρ_j's are independent random functions uniformly distributed over $\boldsymbol{F}(\Sigma^k \times \Sigma^n, \Sigma^{k/2})$, and K_j's are independent random variables uniformly distributed over Σ^k. Since the domain of F_π° is σ-free, B can successfully simulate I_i to A if $\phi(K_j) \ne K_j$ and $\{\pi(K_j), \pi(\phi(K_j))\} \cap \{K_j, \phi(K_j)\}$ is empty for every $1 \le j \le q$. Let Bad be the event that $\phi(K_j) = K_j$ or $\{\pi(K_j), \pi(\phi(K_j))\} \cap \{K_j, \phi(K_j)\}$ is not empty for some j. Then,

$$\begin{aligned}
&\Pr\left[B^{\langle (\rho_j,K_j) \rangle_{j=1}^q} = 1\right] \\
&= \Pr[\neg\mathsf{Bad}] \Pr\left[B^{\langle (\rho_j,K_j) \rangle_{j=1}^q} = 1 \,\middle|\, \neg\mathsf{Bad}\right] + \Pr\left[\mathsf{Bad} \wedge B^{\langle (\rho_j,K_j) \rangle_{j=1}^q} = 1\right] \\
&= \frac{\Pr[\neg\mathsf{Bad}]}{\ell} \sum_{i=1}^{\ell} \Pr[A^{I_i} = 1] + \Pr\left[\mathsf{Bad} \wedge B^{\langle (\rho_j,K_j) \rangle_{j=1}^q} = 1\right] \\
&= \frac{1}{\ell} \sum_{i=1}^{\ell} \Pr[A^{I_i} = 1] - \frac{\Pr[\mathsf{Bad}]}{\ell} \sum_{i=1}^{\ell} \Pr[A^{I_i} = 1] + \Pr\left[\mathsf{Bad} \wedge B^{\langle (\rho_j,K_j) \rangle_{j=1}^q} = 1\right] \ .
\end{aligned}$$

From the discussions above,

$$\begin{aligned}\mathrm{Adv}^{q\text{-}\mathrm{prf}\text{-}\mathrm{rka}}_{\mathsf{Rel},f}(B) &= \left|\Pr\left[B^{\langle(f,K_j)\rangle_{j=1}^{q}}=1\right]-\Pr\left[B^{\langle(\rho_j,K_j)\rangle_{j=1}^{q}}=1\right]\right| \\ &\geq \frac{1}{\ell}\left|\Pr[A^{I_0}=1]-\Pr[A^{I_\ell}=1]\right| - \Pr[\mathsf{Bad}] \\ &= \frac{1}{\ell}\,\mathrm{Adv}^{\mathrm{prf}}_{F^{\circ}_{\pi}}(A) - \Pr[\mathsf{Bad}]\ .\end{aligned}$$

Thus,

$$\begin{aligned}\mathrm{Adv}^{\mathrm{prf}}_{F^{\circ}_{\pi}}(A) &\leq \ell\cdot\mathrm{Adv}^{q\text{-}\mathrm{prf}\text{-}\mathrm{rka}}_{\mathsf{Rel},f}(B) + \ell\cdot\Pr[\mathsf{Bad}] \\ &\leq \ell\cdot\mathrm{Adv}^{q\text{-}\mathrm{prf}\text{-}\mathrm{rka}}_{\mathsf{Rel},f}(B) + \ell\, q\left(\frac{|P_{\pi,\phi}|}{2^k}+\frac{1}{2^{k/2}}\right)\ .\end{aligned}$$

B makes at most q queries and runs in time at most $\tau + O(\ell q T_E)$. □

Lemma 5. *Let $f(K,x) = \mathsf{tr}_{k/2}(E_K(x)\oplus x)$. Let A be a prf-rka-adversary against f with m oracles. Suppose that A runs in time at most τ and makes at most q queries restricted by* Rel. *Suppose that $q \leq \lambda 2^n/\mathrm{e}$ for some positive constant $\lambda < 1$. Then, there exists a prp-rka-adversary B against E such that*

$$\mathrm{Adv}^{m\text{-}\mathrm{prf}\text{-}\mathrm{rka}}_{\mathsf{Rel},f}(A) \leq m\cdot\mathrm{Adv}^{\mathrm{prp}\text{-}\mathrm{rka}}_{\mathsf{Rel},E}(B) + \frac{2^{k/2}}{1-\lambda}\left(\frac{\mathrm{e}\,q}{2^n}\right)^{2^{n-k/2}+1}\ .$$

B makes at most q queries restricted by Rel *and runs in time at most $\tau + O(q\,T_E)$, where T_E represents the time required to compute E.*

Proof. Let $K_1,\ldots,K_m$ be independent random variables uniformly distributed over Σ^k. Let $\rho_1,\ldots,\rho_m$ be independent and random keyed functions uniformly distributed over $\boldsymbol{F}(\Sigma^k\times\Sigma^n,\Sigma^{k/2})$. Let $\varpi_1,\ldots,\varpi_m$ be independent random keyed permutations uniformly distributed over $\boldsymbol{P}(\Sigma^k\times\Sigma^n,\Sigma^n)$, and let $\tilde{\varpi}_j(\cdot,x) = \mathsf{tr}_{k/2}(\varpi_j(\cdot,x)\oplus x)$ for $1\leq j\leq m$. Then,

$$\begin{aligned}\mathrm{Adv}^{m\text{-}\mathrm{prf}\text{-}\mathrm{rka}}_{\mathsf{Rel},f}(A) \leq{}& \left|\Pr\left[A^{\langle(f,K_j)\rangle_{j=1}^{m}}=1\right]-\Pr\left[A^{\langle(\tilde{\varpi}_j,K_j)\rangle_{j=1}^{m}}=1\right]\right| + \\ & \left|\Pr\left[A^{\langle(\tilde{\varpi}_j,K_j)\rangle_{j=1}^{m}}=1\right]-\Pr\left[A^{\langle(\rho_j,K_j)\rangle_{j=1}^{m}}=1\right]\right|\ .\end{aligned}$$

Let $\mathcal{O}_i$ be m oracles such that $(f,K_1),\ldots,(f,K_i),(\tilde{\varpi}_{i+1},K_{i+1}),\ldots,(\tilde{\varpi}_m,K_m)$ for $0\leq i\leq m$. Notice that $\mathcal{O}_0 = \langle(\tilde{\varpi}_j,K_j)\rangle_{j=1}^{m}$ and $\mathcal{O}_m = \langle(f,K_j)\rangle_{j=1}^{m}$.

A prp-rka-adversary B is constructed using A as a subroutine. The algorithm of B with an oracle (u,K) is given below, where u is either E or ϖ. ϖ is a random keyed permutation uniformly distributed over $\boldsymbol{P}(\Sigma^k\times\Sigma^n,\Sigma^n)$, and K is a random variable uniformly distributed over Σ^k.

1. selects i from $\{1,2,\ldots,m\}$ uniformly at random.
2. runs A with oracles $(f,K_1),\ldots,(f,K_{i-1})$, $(\tilde{u},K)$, $(\tilde{\varpi}_{i+1},K_{i+1}),\ldots,(\tilde{\varpi}_m,K_m)$ by simulating $(f,K_1),\ldots,(f,K_{i-1})$, and $(\tilde{\varpi}_{i+1},K_{i+1}),\ldots,(\tilde{\varpi}_m,K_m)$, where $\tilde{u}(\cdot,x) = \mathsf{tr}_{k/2}(u(\cdot,x)\oplus x)$.

3. outputs A's output.

Then,

$$\Pr\left[B^{(E,K)}=1\right]=\frac{1}{m}\sum_{i=1}^{m}\Pr\left[A^{\mathcal{O}_i}=1\right]$$

and

$$\Pr\left[B^{(\varpi,K)}=1\right]=\frac{1}{m}\sum_{i=0}^{m-1}\Pr\left[A^{\mathcal{O}_i}=1\right] \ .$$

Thus,

$$\mathrm{Adv}^{\mathrm{prp\text{-}rka}}_{\mathsf{Rel},E}(B)=\frac{1}{m}\left|\Pr\left[A^{\mathcal{O}_m}=1\right]-\Pr\left[A^{\mathcal{O}_0}=1\right]\right| \ .$$

B makes at most q queries and runs in time at most $\tau+O(q\,T_E)$.

It is possible to distinguish $\tilde{\varpi}_j$ and ρ_j only by the fact that there may be $(2^{n-k/2}+1)$-collision for $\rho_j(\cdot,x)\oplus x$. Thus, since A makes at most q queries,

$$\left|\Pr\left[A^{\langle(\tilde{\varpi}_j,K_j)\rangle_{j=1}^m}=1\right]-\Pr\left[A^{\langle(\rho_j,K_j)\rangle_{j=1}^m}=1\right]\right|\le\frac{2^{k/2}}{1-\lambda}\left(\frac{\mathrm{e}\,q}{2^n}\right)^{2^{n-k/2}+1} \ ,$$

which follows from Lemma 3. □

5.1 An Example of Padding for σ-Free Inputs

In this subsection, σ is assumed to be a permutation on Σ^n such that $\sigma(x)=x\oplus c$ for some non-zero constant c. The permutation is denoted by σ_c.

Let pad be a padding function such that

$$\mathsf{pad}(M)=M\|10^{d+n/2}\|\mathsf{len}_{n/2}(M) \ ,$$

where d is a minimum non-zero integer such that $|M|+d\equiv n-1 \pmod{n}$, and $\mathsf{len}_{n/2}(M)$ is the $n/2$-bit binary representation of $|M|$. It is easy to see that pad is σ_c-free if, for example, $c=1^{n/2}\|0^{n/2}$.

6 Indifferentiability from Random Oracle

We show that F°_{π} is indifferentiable from a VIL random oracle in the ideal cipher model with pad and σ_c given in the previous section.

Theorem 4. *Let $E\in\mathcal{BC}(n,k)$. Let P_π be the set of fixed points of π. Let A be an adversary that asks at most q_{V} queries to the VIL oracle, q_{e} queries to the encryption oracle and q_{d} queries to the decryption oracle. Let ℓ be the maximum number of message blocks in a VIL query. Suppose that $q=\ell q_{\mathrm{V}}+q_{\mathrm{e}}+q_{\mathrm{d}}<2^{n-1}/3$. Then, in the ideal cipher model, $\mathrm{Adv}^{\mathrm{indiff}}_{F^{\circ}_{\pi},S}(A)$ is bounded from above by*

$$\frac{q}{2^{k/2}(1-3q/2^{n-1})}+\frac{9q^2+2(|P_\pi|-1)q}{2^k(1-3q/2^{n-1})^2}+\frac{q^2}{4(2^k-2^{k/2}-6q-|P_\pi|+4)}+\frac{q}{2^{n-1}} \ ,$$

where the simulator S is given in Figure 3. S makes at most $2(q_{\mathrm{e}}+q_{\mathrm{d}})$ queries and runs in time $O((q_{\mathrm{e}}+q_{\mathrm{d}})^2)$.

Proof. Omitted due to the page limit. □

Theorem 4 implies that the query complexity to differentiate F_π° from a VIL random oracle is $\Omega(\min\{2^{k/2}, 2^n\})$, which is optimal up to a constant factor.

Example 2. The upper bound of Theorem 4 is 0.5 if $q = 2^{124.3}$ for $k = 256$ and if $q = 2^{93.5}$ for $k = 192$. Though IRO implies CR, Theorem 1 gives a slightly better bound for CR than Theorem 4.

The simulator S given in Figure 3 simulates the ideal cipher by lazy evaluation. $\mathcal{P}(s)$ ($\mathcal{C}(s)$) is the set of plaintexts (ciphertexts) which are available for the reply to the current query with the key s. $\mathtt{E}_s(x)$ and $\mathtt{D}_s(x)$ are $\perp$ for any $s \in \Sigma^k$ and $x \in \Sigma^n$ initially. They get defined by the queries of the adversary and the corresponding oracle replies. $\mathcal{V}$ is the set of the keys in the queries so far.

The simulator keeps a tree, which initially consists of the root IV. $\mathcal{T}$ is the set of the nodes in the tree so far. During the simulation, for example, new nodes $F(s, x) = t_0 \| t_1$ and $F(s, \sigma(x)) = t_1 \| t_0$ are created by an encryption query (s, x) if $s \in \mathcal{T}$, and they augment the tree together with the edges $s \xrightarrow{x} t_0 \| t_1$ and $s \xrightarrow{\sigma(x)} t_1 \| t_0$.

The procedure $\mathsf{extend}(s)$ uses the VIL random oracle H and evaluates $F_\pi^\circ(IV, \cdot)$ for the message, if any, corresponding to the path in the tree from the root IV to s such that s is the chaining value fed into final F through π. Owing to the padding $\mathtt{pad}$, the message is unique if it exists. The procedure $\mathsf{path}(s)$ returns the message. $\mathtt{lb}(\tilde{M})$ is the last block of $\mathtt{pad}(\tilde{M})$. $\mathtt{fhalf}$ and $\mathtt{shalf}$ give the first half and the second half of the input string, respectively.

7 Implementation

We implemented the proposed compression function by instantiating the ideal cipher E with AES-192 or AES-256. The involution σ was defined with the bitwise complement of the first byte of M_i. The throughput of the compression function was measured on the Intel Core i7-2600S, the Intel Core i7-2600, and the Intel Core i7-2720QM, which support the AES instruction set (AES-NI). The GNU Compiler Collection version 4.4.5 or 4.4.6 was used for code compilation. The result is shown in Table 1. In the *serial* implementation, after the topside encryption is finished, the downside encryption is performed. In the *pipelined* implementation, each round of two encryption functions is interleaved. In both of implementations, the key schedule is performed only once. The throughput of our hash function will approach asymptotically to these values for sufficiently large data.

The result showed that the pipelined implementation was better. The Intel manual [10] recommends to process 4 or 8 blocks in parallel for optimized throughput since the hardware that supports the four AES round instructions is pipelined. Bos et al. [5] pointed out that constructions such as the DM construction gave an advantage on exploiting such a hardware feature. Our hash function can also gain the benefit of the hardware feature by interleaving each round of two encryption functions.

```
Initialize:
$\mathcal{V} \leftarrow \emptyset$; $\mathcal{T} \leftarrow \{IV\}$; $\mathcal{P}(s) \leftarrow \Sigma^n$; $\mathcal{C}(s) \leftarrow \Sigma^n$;

Interface $\mathcal{E}(s, x)$:
if $\mathtt{E}_s(x) = \perp$ then
    if $s \in \mathcal{T}$ then
        $\mathtt{E}_s(x) \xleftarrow{\$} \mathcal{C}(s)$; $\mathtt{E}_s(\sigma(x)) \xleftarrow{\$} \mathcal{C}(s) \setminus \{\mathtt{E}_s(x)\}$;
        $t_0 \leftarrow \mathsf{tr}_{k/2}(\mathtt{E}_s(x) \oplus x)$; $t_1 \leftarrow \mathsf{tr}_{k/2}(\mathtt{E}_s(\sigma(x)) \oplus \sigma(x))$;
        if $t_0 = t_1 \vee \{t_0 \| t_1, t_1 \| t_0\} \cap \boldsymbol{B} \neq \emptyset$ then abort;
        $\mathcal{T} \leftarrow \mathcal{T} \cup \{t_0 \| t_1, t_1 \| t_0\}$;
        $\mathsf{extend}(t_0 \| t_1)$; $\mathsf{extend}(t_1 \| t_0)$;
    else
        $\mathtt{E}_s(x) \xleftarrow{\$} \mathcal{C}(s)$; $\mathtt{E}_s(\sigma(x)) \xleftarrow{\$} \mathcal{C}(s) \setminus \{\mathtt{E}_s(x)\}$;
    $\mathcal{V} \leftarrow \mathcal{V} \cup \{s\}$; $\mathcal{P}(s) \leftarrow \mathcal{P}(s) \setminus \{x, \sigma(x)\}$; $\mathcal{C}(s) \leftarrow \mathcal{C}(s) \setminus \{\mathtt{E}_s(x), \mathtt{E}_s(\sigma(x))\}$;
return $\mathtt{E}_s(x)$;

Interface $\mathcal{D}(s, x)$:
if $\mathtt{D}_s(x) = \perp$ then
    if $s \in \mathcal{T}$ then
        $\mathtt{D}_s(x) \xleftarrow{\$} \mathcal{P}(s)$; $\mathtt{E}_s(\sigma(\mathtt{D}_s(x))) \xleftarrow{\$} \mathcal{C}(s) \setminus \{x\}$;
        $t_0 \leftarrow \mathsf{tr}_{k/2}(\mathtt{D}_s(x) \oplus x)$; $t_1 \leftarrow \mathsf{tr}_{k/2}(\sigma(\mathtt{D}_s(x)) \oplus \mathtt{E}_s(\sigma(\mathtt{D}_s(x))))$;
        if $t_0 = t_1 \vee \{t_0 \| t_1, t_1 \| t_0\} \cap \boldsymbol{B} \neq \emptyset$ then abort;
        $\mathcal{T} \leftarrow \mathcal{T} \cup \{t_0 \| t_1, t_1 \| t_0\}$;
        $\mathsf{extend}(t_0 \| t_1)$; $\mathsf{extend}(t_1 \| t_0)$;
    else
        $\mathtt{D}_s(x) \xleftarrow{\$} \mathcal{P}(s)$; $\mathtt{E}_s(\sigma(\mathtt{D}_s(x))) \xleftarrow{\$} \mathcal{C}(s) \setminus \{x\}$;
    $\mathcal{V} \leftarrow \mathcal{V} \cup \{s\}$; $\mathcal{P}(s) \leftarrow \mathcal{P}(s) \setminus \{\mathtt{D}_s(x), \sigma(\mathtt{D}_s(x))\}$; $\mathcal{C}(s) \leftarrow \mathcal{C}(s) \setminus \{x, \mathtt{E}_s(\sigma(\mathtt{D}_s(x)))\}$;
return $\mathtt{D}_s(x)$;

Subroutine $\mathsf{extend}(s)$:
$\tilde{s} \leftarrow \pi(s)$; $\tilde{M} \leftarrow \mathsf{path}(\tilde{s})$; $x \leftarrow \mathtt{lb}(\tilde{M})$;
if $x \neq \perp \wedge \mathtt{E}_{\tilde{s}}(x) = \perp$ then                    ▷ if $\tilde{M}$ exists
    $t'_0 \xleftarrow{\$} \Sigma^{n-k/2}$; $t'_1 \xleftarrow{\$} \Sigma^{n-k/2}$;
    $t_0 \leftarrow t'_0 \| \mathtt{fhalf}(\mathsf{H}(\tilde{M}))$; $t_1 \leftarrow t'_1 \| \mathtt{shalf}(\mathsf{H}(\tilde{M}))$;
    $\mathtt{E}_{\tilde{s}}(x) \leftarrow t_0 \oplus x$; $\mathtt{E}_{\tilde{s}}(\sigma(x)) \leftarrow t_1 \oplus \sigma(x)$;
    if $\mathtt{E}_{\tilde{s}}(x) = \mathtt{E}_{\tilde{s}}(\sigma(x)) \vee \{\mathtt{E}_{\tilde{s}}(x), \mathtt{E}_{\tilde{s}}(\sigma(x))\} \not\subset \mathcal{C}(\tilde{s})$ then abort;
    $\mathcal{V} \leftarrow \mathcal{V} \cup \{\tilde{s}\}$; $\mathcal{P}(\tilde{s}) \leftarrow \mathcal{P}(\tilde{s}) \setminus \{x, \sigma(x)\}$; $\mathcal{C}(\tilde{s}) \leftarrow \mathcal{C}(\tilde{s}) \setminus \{\mathtt{E}_{\tilde{s}}(x), \mathtt{E}_{\tilde{s}}(\sigma(x))\}$;
```

Fig. 3. Pseudocode for the simulator S. $\boldsymbol{B} = \mathcal{V} \cup \mathcal{T} \cup \pi^{-1}(\mathcal{V} \cup \mathcal{T}) \cup \pi(\mathcal{T}) \cup P_\pi$.

Table 1. Throughput [cycles/byte]

k	192			256		
Core i7	2600S	2600	2720QM	2600S	2600	2720QM
serial	7.07	8.43	6.44	9.07	11.09	8.21
pipelined	6.44	8.06	5.84	8.00	9.80	7.26

References

1. Armknecht, F., Fleischmann, E., Krause, M., Lee, J., Stam, M., Steinberger, J.: The preimage security of double-block-length compression functions. In: Lee, D.H., Wang, X. (eds.) ASIACRYPT 2011. LNCS, vol. 7073, pp. 233–251. Springer, Heidelberg (2011)
2. Bellare, M., Kohno, T.: A theoretical treatment of related-key attacks: RKA-PRPs, RKA-PRFs, and applications. In: Biham, E. (ed.) EUROCRYPT 2003. LNCS, vol. 2656, pp. 491–506. Springer, Heidelberg (2003)
3. Black, J., Rogaway, P., Shrimpton, T., Stam, M.: An analysis of the blockcipher-based hash functions from PGV. Journal of Cryptology 23(4), 519–545 (2010)
4. 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)
5. Bos, J.W., Özen, O., Stam, M.: Efficient hashing using the AES instruction set. In: Preneel, B., Takagi, T. (eds.) CHES 2011. LNCS, vol. 6917, pp. 507–522. Springer, Heidelberg (2011)
6. Brachtl, B.O., Coppersmith, D., Hyden, M.M., Matyas Jr., S.M., Meyer, C.H.W., Oseas, J., Pilpel, S., Schilling, M.: Data authentication using modification detection codes based on a public one-way encryption function. U. S. Patent # 4,908,861 (March 1990)
7. Coron, J.-S., Dodis, Y., Malinaud, C., Puniya, P.: Merkle-Damgård revisited: How to construct a hash function. In: Shoup, V. (ed.) CRYPTO 2005. LNCS, vol. 3621, pp. 430–448. Springer, Heidelberg (2005)
8. FIPS PUB 180-4. Secure hash standard (SHS) (March 2012)
9. Fleischmann, E., Gorski, M., Lucks, S.: Security of cyclic double block length hash functions. In: Parker (ed.) [23], pp. 153–175
10. Gueron, S.: Intel advanced encryption standard (AES) instructions set (2010), `http://software.intel.com/en-us/articles/intel-advanced-encryption-standard-aes-instructions-set/`
11. Hirose, S.: Some plausible constructions of double-block-length hash functions. In: Robshaw, M.J.B. (ed.) FSE 2006. LNCS, vol. 4047, pp. 210–225. Springer, Heidelberg (2006)
12. Hirose, S., Park, J.H., Yun, A.: A simple variant of the Merkle-Damgård scheme with a permutation. In: Kurosawa, K. (ed.) ASIACRYPT 2007. LNCS, vol. 4833, pp. 113–129. Springer, Heidelberg (2007)
13. Jonsson, J., Robshaw, M.J.B.: Securing RSA-KEM via the AES. In: Vaudenay, S. (ed.) PKC 2005. LNCS, vol. 3386, pp. 29–46. Springer, Heidelberg (2005)
14. Kuwakado, H., Hirose, S.: Hashing mode using a lightweight blockcipher. In: Stam, M. (ed.) IMACC 2013. LNCS, vol. 8308, pp. 213–231. Springer, Heidelberg (2013)
15. Lai, X., Massey, J.L.: Hash functions based on block ciphers. In: Rueppel, R.A. (ed.) EUROCRYPT 1992. LNCS, vol. 658, pp. 55–70. Springer, Heidelberg (1993)

16. Lee, J., Kwon, D.: The security of Abreast-DM in the ideal cipher model. IEICE Transactions 94-A(1), 104–109 (2011)
17. Lee, J., Stam, M.: MJH: A faster alternative to MDC-2. In: Kiayias, A. (ed.) CT-RSA 2011. LNCS, vol. 6558, pp. 213–236. Springer, Heidelberg (2011)
18. Lee, J., Stam, M.: MJH: A faster alternative to MDC-2. Cryptology ePrint Archive, Report 2014/108 (2014), http://eprint.iacr.org/
19. Lee, J., Stam, M., Steinberger, J.: The collision security of Tandem-DM in the ideal cipher model. In: Rogaway, P. (ed.) CRYPTO 2011. LNCS, vol. 6841, pp. 561–577. Springer, Heidelberg (2011)
20. Maurer, U.M., Renner, R., Holenstein, C.: Indifferentiability, impossibility results on reductions, and applications to the random oracle methodology. In: Naor, M. (ed.) TCC 2004. LNCS, vol. 2951, pp. 21–39. Springer, Heidelberg (2004)
21. Naito, Y.: Blockcipher-based double-length hash functions for pseudorandom oracles. In: Miri, A., Vaudenay, S. (eds.) SAC 2011. LNCS, vol. 7118, pp. 338–355. Springer, Heidelberg (2012)
22. Özen, O., Stam, M.: Another glance at double-length hashing. In: Parker (ed.) [23], pp. 176–201
23. Parker, M.G. (ed.): Cryptography and Coding 2009. LNCS, vol. 5921. Springer, Heidelberg (2009)
24. Rijmen, V., Barreto, P.S.L.M.: The Whirlpool hash function (2000), http://www.larc.usp.br/~pbarreto/WhirlpoolPage.html
25. Rivest, R.: The MD4 message-digest algorithm. Request for Comments 1320 (RFC 1320), The Internet Engineering Task Force (1992)
26. Rivest, R.: The MD5 message-digest algorithm. Request for Comments 1321 (RFC 1321), The Internet Engineering Task Force (1992)
27. Rohde, S., Eisenbarth, T., Dahmen, E., Buchmann, J., Paar, C.: Fast hash-based signatures on constrained devices. In: Grimaud, G., Standaert, F.-X. (eds.) CARDIS 2008. LNCS, vol. 5189, pp. 104–117. Springer, Heidelberg (2008)
28. Steinberger, J.P.: The collision intractability of MDC-2 in the ideal-cipher model. In: Naor, M. (ed.) EUROCRYPT 2007. LNCS, vol. 4515, pp. 34–51. Springer, Heidelberg (2007)