Vector-Input Hashing Modes for Collision-Resistant Pseudorandom Function*

Shoichi Hirose[1], Tetsu Iwata[2], and Hidenori Kuwakado[3]

[1] University of Fukui, Fukui, Japan
hrs_shch@u-fukui.ac.jp
[2] Nagoya University, Nagoya, Japan
tetsu.iwata@nagoya-u.jp
[3] Kansai University, Takatsuki, Japan
kuwakado@kansai-u.ac.jp

Abstract. This paper presents vector-input keyed hashing modes that construct collision-resistant pseudorandom functions (CR PRFs) using a keyed hash function, where a vector refers to a sequence of variable-length strings. The proposed vector-input keyed hashing modes, VIM1 and VIM2, originate from the intuition that a string-input keyed hashing mode using a compression function results in a vector-input keyed hashing mode by replacing the compression function with a hash function. Combined with the recently proposed string-input keyed hashing modes KHC1 or KHC2, VIM1 and VIM2 are shown to yield CR PRFs from a compression function satisfying extended collision resistance and being a secure PRF under related-key attacks. Extended collision resistance means that it is intractable to find a distinct input pair whose output difference falls within a small set. This paper also introduces a keyed hashing mode, PVIM, which allows parallel processing of strings in a vector. However, it requires more calls to the underlying keyed hash function than VIM1 and VIM2 do. To the best of our knowledge, this is the first proposal of dedicated vector-input CR PRFs.

Keywords: Keyed hashing mode · Vector input · Collision resistance · Pseudorandom function · Compression function

1 Introduction

Background. Cryptographic hash functions are useful for various applications such as message digest, message authentication, and pseudorandom generation. It is well known that a keyed hash function H that satisfies both collision-resistance and pseudorandomness instantiates computationally hiding and computationally binding string commitment. For message M, one chooses key K uniformly at random and computes commitment $\sigma \leftarrow H_K(M)$. Recently, some interesting cryptographic schemes were constructed with such collision-resistant

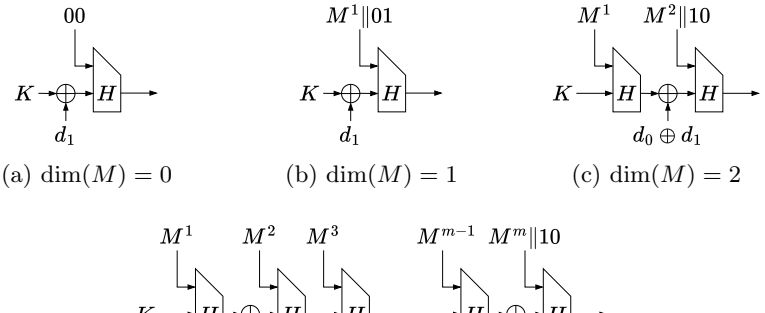
* This is the full version of a paper that appeared in the 28th International Conference on Information Security and Cryptology (ICISC 2025).

pseudorandom functions (CR PRFs): compactly committing authenticated encryption with associated data (ccAEAD) [11,16] and hash-based post-quantum EPID signatures [9].

Conventional CR PRFs such as HMAC [3,12] and KMAC [22] are defined over strings. However, data to be processed have specific formats or structures in many practical situations. A recognized method to process such data is to encode a given datum into a string and feed it to a conventional string-input CR PRF. An example of such encoding is included in the specification of TupleHash [22] for a sequence of variable-length strings, which is often called a vector. Although this method always works, the encoding process introduces an efficiency loss.

Our Contribution. Our goal is to construct a vector-input CR PRF using a compression function. Our idea is to nest compression-function-based keyed hashing modes constructing string-input CR PRFs, that is, to replace the compression function of a keyed hashing mode with a string-input CR PRF using a compression function. To the best of our knowledge, this is the first proposal of dedicated vector-input CR PRFs.

For the proposed construction, we adopt two string-input keyed hashing modes KHC1 and KHC2 [19], which are cascades of a compression function. Precisely, simple nesting is incomplete since it cannot handle the 0-dimensional vector (the vector with no string). Thus, we first present two vector-input keyed hashing modes VIM1 and VIM2 based on KHC1 and KHC2, respectively. They construct vector-input keyed hash functions using a string-input keyed hash function. VIM1 and VIM2 using a keyed hash function $H : \{0,1\}^n \times \{0,1\}^* \to \{0,1\}^n$ are depicted in Figs. 1 and 2, respectively. In these figures, $K \in \{0,1\}^n$ is a secret key, d_0, d_1, d_2, d_3 are constants in $\{0,1\}^n$, and ε is the empty string.

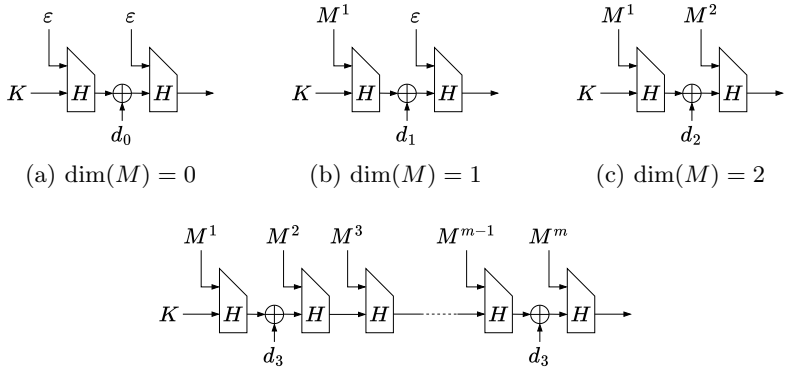


(a) $\dim(M) = 0$ (b) $\dim(M) = 1$ (c) $\dim(M) = 2$

(d) $\dim(M) \geq 3$

Fig. 1: Proposed vector-input keyed hashing mode VIM1

VIM1 and VIM2 are shown to be collision-resistant if the underlying keyed hash function satisfies extended collision resistance, that is, it is intractable to



(a) $\dim(M) = 0$ (b) $\dim(M) = 1$ (c) $\dim(M) = 2$

(d) $\dim(M) \geq 3$

Fig. 2: Proposed vector-input keyed hashing mode VIM2

find a pair of distinct inputs to the underlying keyed hash function such that the difference of their corresponding outputs is in a small set defined by the constants specifying VIM1 and VIM2. VIM1 and VIM2 are also shown to be a secure PRF, using the hybrid argument for cascade construction of a string-input PRF [4], if the underlying keyed hash function is a secure PRF under related-key attacks [5]. Their key derivation functions are defined by the constants specifying VIM1 and VIM2.

KHC1 and KHC2 were shown to construct string-input CR PRFs [19]. However, it is not sufficient for constructing vector-input CR PRFs by using them as a building block of VIM1 and VIM2. Thus, in this paper, KHC1 and KHC2 are shown to satisfy extended collision resistance if the underlying compression function satisfies it. KHC1 and KHC2 are also shown to be a secure PRF under related-key attacks if the underlying compression function is a secure PRF under related-key attacks.

Finally, we present a parallelizable vector-input keyed hashing mode, which constructs CR PRFs by using KHC1 or KHC2. While it enables parallel processing when the strings in a vector are simultaneously available, it requires more calls to the underlying compression function than VIM1 and VIM2 with KHC1 or KHC2.

Related Work. An iterated hash function consisting of a compression function is called a Merkle-Damgård hash function [10,25]. Pseudorandom functions were first introduced by Goldreich, Goldwasser, and Micali [15].

KHC1 and KHC2 [19] are inspired by the suffix-free-prefix-free (SFPF) hash function construction [1] and KMDP$^+$ [17]. KMDP$^+$, which improves Keyed-MDP [21], is the most efficient string-input keyed hashing mode of a compression function to construct a CR PRF. KMDP$^+$ cannot be utilized for VIM1 and VIM2 since it only accepts secret keys shorter than the outputs. Both KHC1 and KHC2

are competitive with KMDP$^+$ in terms of the number of calls to the underlying compression function.

There are many proposals of string-input keyed hashing constructions based on the Merkle-Damgård construction and satisfying collision resistance and pseudorandomness. They are classified by their underlying primitives: HMAC [3], Sandwich construction [28], H^2-MAC [29], and AMAC [2] use a cryptographic hash function, while NMAC [3], EMD [6], Keyed MDP [21], Prefix MDP [21], and KMDP$^+$ [17] use a compression function.

There are several proposals of constructions for a vector-input PRF, which do not address collision resistance. Rogaway and Shrimpton [27] presented the first vector-input PRF called S2V as a component of their SIV construction for deterministic authenticated encryption. Minematsu [26] proposed a universal hash function based on bit rotation and used it to construct a vector-input PRF. Hirose, Kuwakado, and Yoshida [20] constructed a vector-input PRF based on the lightweight hash function Lesamnta-LW [18] and MDP [21], employing the protected counter sum construction [7]. Furuya et al. [13] adopted PMAC2x [24] using a tweakable block cipher [23] as the baseline scheme and constructed the vector-input PRF called PMACrx that achieved the beyond-the-birthday-bound security. Furuya and Minematsu [14] presented a vector-input PRF based on PMAC [8], which attains optimal efficiency with respect to the number of calls to the underlying block cipher.

Organization. Section 2 gives notations and definitions. It also introduces KHC1 and KHC2. Section 3 proposes vector-input keyed hashing modes VIM1 and VIM2. Section 4 shows that both KHC1 and KHC2 construct keyed hash functions satisfying extended collision resistance and being a secure PRF under related-key attacks. Section 5 confirms that both VIM1 and VIM2 construct CR PRFs by using KHC1 or KHC2. Section 6 presents a parallelizable vector-input keyed hashing mode for CR PRFs. Section 7 gives a concluding remark.

2 Preliminaries

2.1 Notations

For set $\mathcal{S}$, let $\mathcal{S}^* \coloneqq \bigcup_{i=0}^{\infty} \mathcal{S}^i$ and $\mathcal{S}^+ \coloneqq \bigcup_{i=1}^{\infty} \mathcal{S}^i$. Let $s \twoheadleftarrow \mathcal{S}$ denote that s is assigned an element chosen uniformly at random from set $\mathcal{S}$.

Let $\Sigma \coloneqq \{0,1\}$. $\Sigma^0 = \{\varepsilon\}$, where ε is the empty sequence.

For binary sequence $u \in \Sigma^*$, let $|u|$ be the length of u in bits. For binary sequences $z_i, z_{i+1}, \ldots, z_{i+j} \in \Sigma^*$, let $z_i \| z_{i+1} \| \cdots \| z_{i+j}$ be their concatenation, which is also denoted by $z_{[i,i+j]}$ for simplicity.

For integer κ, the identity permutation over Σ^κ is denoted by id, and the permutation $x \mapsto x \oplus c$ over Σ^κ is denoted by x_c, where $c \in \Sigma^\kappa$. For $\Delta \subseteq \Sigma^\kappa$ and $c \in \Sigma^\kappa$, let $\Delta \oplus c \coloneqq \{\delta \oplus c \,|\, \delta \in \Delta\}$.

Let $\Sigma^{**} \coloneqq (\Sigma^*)^*$ be the set of vectors. For $V \in \Sigma^{**}$, let $\dim(V)$ be the dimension of V. $(\Sigma^*)^0 = \{\lambda\}$, where $\lambda \coloneqq ()$ is the zero-dimensional vector.

For integers a, b, and d, $a \equiv b \pmod{d}$ is denoted by $a \equiv_d b$.

A random function $\rho : \mathcal{D} \to \mathcal{R}$ is called a random oracle if for any $x \in \mathcal{D}$, $\rho(x)$ is assigned an element chosen uniformly at random from $\mathcal{R}$.

2.2 Collision Resistance and Its Extension

Let $H^F : \mathcal{X} \to \Sigma^n$ be a hash function that uses compression function F. An input pair X and X' for H^F is called colliding if $X \neq X'$ and $H^F(X) = H^F(X')$. H^F is called collision-resistant if it is intractable to find a colliding input pair for H^F.

We extend the notion of collision resistance as follows. For $\Delta \subset \Sigma^n$, an input pair X and X' for H^F is called Δ-colliding if $X \neq X'$ and $H^F(X) \oplus H^F(X') \in \Delta$. H^F is called Δ-collision-resistant if it is intractable to find a Δ-colliding input pair for H^F. $\{0^n\}$-collision resistance refers to collision resistance.

2.3 Pseudorandom Function

Let $f : \mathcal{K} \times \mathcal{X} \to \mathcal{Y}$ be a keyed function with key space $\mathcal{K}$ and $f_K(\cdot) \coloneqq f(K, \cdot)$. Let $\mathbf{A}$ be an adversary against f. The goal of $\mathbf{A}$ is to distinguish between f_K and a random oracle $\rho : \mathcal{X} \to \mathcal{Y}$, where $K \twoheadleftarrow \mathcal{K}$. $\mathbf{A}$ is given either f_K or ρ as an oracle and makes adaptive queries in $\mathcal{X}$. $\mathbf{A}$ outputs 0 or 1. The prf advantage of $\mathbf{A}$ against f is defined as

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

where $\mathbf{A}$ is regarded as a random variable that takes values in $\{0, 1\}$. f is called a secure pseudorandom function (PRF) if no efficient adversary $\mathbf{A}$ has any significant prf advantage.

2.4 PRF under Related-Key Attack

Let $\mathbf{A}$ be an adversary against $f : \mathcal{K} \times \mathcal{X} \to \mathcal{Y}$. Let Φ be a set of functions from $\mathcal{K}$ to $\mathcal{K}$. $\mathbf{A}$ makes a Φ-related-key attack (Φ-RKA) [5]: $\mathbf{A}$ is given $g\langle K\rangle : \Phi \times \mathcal{X} \to \mathcal{Y}$ such that $g\langle K\rangle(\varphi, X) \coloneqq g(\varphi(K), X)$ as an oracle, where g is either f or a random oracle $\rho : \mathcal{K} \times \mathcal{X} \to \mathcal{Y}$ and $K \twoheadleftarrow \mathcal{K}$. $\varphi \in \Phi$ is called a related-key-derivation function. $\mathbf{A}$ makes adaptive queries to the oracle and outputs 0 or 1. The prf-rka advantage of $\mathbf{A}$ making a Φ-RKA on f is given by

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

f is called a secure PRF under Φ-RKAs if no efficient adversary $\mathbf{A}$ has any significant prf-rka advantage.

The prf-rka advantage of $\mathbf{A}$ with access to p oracles is defined as

$$\mathrm{Adv}_{f,\Phi}^{p\text{-prf-rka}}(\mathbf{A}) \coloneqq \left|\Pr[\mathbf{A}^{f\langle K_1\rangle,\ldots,f\langle K_p\rangle} = 1] - \Pr[\mathbf{A}^{\rho_1\langle K_1\rangle,\ldots,\rho_p\langle K_p\rangle} = 1]\right|,$$

where $(K_1, \ldots, K_p) \twoheadleftarrow \mathcal{K}^p$ and $\rho_1, \ldots, \rho_p$ are independent random oracles.

In the subsequent sections, $\mathcal{K} := \Sigma^\kappa$ for some integer κ, and we are only interested in bitwise XOR for related-key-derivation functions. For $\Delta \subset \Sigma^\kappa$, $\{x \oplus \delta \,|\, \delta \in \Delta\}$-RKA is called Δ-XOR related-key attack (Δ-XRKA), and $\mathrm{Adv}^{\text{prf-rka}}_{f,\{x\oplus\delta \,|\, \delta\in\Delta\}}$ is simply denoted by $\mathrm{Adv}^{\text{prf-rka}}_{f,\Delta}$.

2.5 Keyed Hashing Modes: KHC1 and KHC2

KHC1 and KHC2 [19] are keyed hashing modes using a compression function $F : \Sigma^n \times \Sigma^w \to \Sigma^n$, where $n \geq 1$ and $w \geq 2$ are integers. F is treated as a keyed function with key space Σ^n.

KHC1 constructs a keyed hash function $\mathsf{KHC1}^F : \Sigma^n \times \Sigma^* \to \Sigma^n$. It is specified in Algorithm 1 and depicted in Fig. 3. It uses two distinct non-zero constants $c_0, c_1 \in \Sigma^n$. It uses a padding function $\mathtt{pad1} : \Sigma^* \to (\Sigma^w)^+$ such that

$$\mathtt{pad1}(M) := \begin{cases} M\|10^d\|0 & \text{if } |M| \leq w-2, \\ M\|10^d\|1 & \text{otherwise,} \end{cases}$$

where d is the smallest non-negative integer that satisfies $|M| + 2 + d \equiv_w 0$.

Algorithm 1: $\mathsf{KHC1}^F : \Sigma^n \times \Sigma^* \to \Sigma^n$

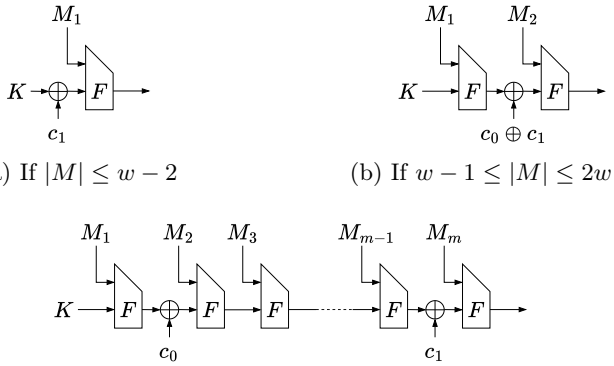
input : (K, M)
output: $\mathsf{KHC1}^F(K, M)$
$M_1\|M_2\|\cdots\|M_m \leftarrow \mathtt{pad1}(M)$; `/*` $|M_i| = w$ `for` $1 \leq i \leq m$ `*/`
if $m = 1$ **then** $V_1 \leftarrow F(K \oplus c_1, M_1)$;
else
  $V_1 \leftarrow F(K, M_1) \oplus c_0$;
  for $i = 2$ **to** $m-1$ **do** $V_i \leftarrow F(V_{i-1}, M_i)$;
  $V_m \leftarrow F(V_{m-1} \oplus c_1, M_m)$;
return V_m;

KHC2 constructs a keyed hash function $\mathsf{KHC2}^F : \Sigma^n \times \Sigma^* \to \Sigma^n$. It is specified in Algorithm 2 and depicted in Fig. 4. It uses three distinct non-zero constants $c_0, c_{1,0}, c_{1,1} \in \Sigma^n$, satisfying $c_0 \oplus c_{1,0} \oplus c_{1,1} \neq 0^n$. It uses a padding function $\mathtt{pad2} : \Sigma^* \to (\Sigma^w)^+$ such that

$$\mathtt{pad2}(M) := \begin{cases} M & \text{if } |M| \geq 2w \text{ and } |M| \equiv_w 0, \\ M\|10^d & \text{otherwise,} \end{cases}$$

where d is the smallest non-negative integer that satisfies $|\mathtt{pad2}(M)| \geq 2w$ and $|\mathtt{pad2}(M)| \equiv_w 0$. Notice that, for any M such that $0 \leq |M| \leq 2w-1$, $\mathtt{pad2}(M) = M\|10^{2w-1-|M|}$.

The following theorems are obtained for collision resistance and pseudorandomness of $\mathsf{KHC1}^F$ and $\mathsf{KHC2}^F$ [19]:



(a) If $|M| \leq w - 2$

(b) If $w - 1 \leq |M| \leq 2w - 2$

(c) If $|M| \geq 2w - 1$

Fig. 3: Keyed hashing mode KHC1

Algorithm 2: $\mathsf{KHC2}^F : \Sigma^n \times \Sigma^* \to \Sigma^n$

input : (K, M)
output: $\mathsf{KHC2}^F(K, M)$
$M_1 \| M_2 \| \cdots \| M_m \leftarrow \mathtt{pad2}(M)$; `/*` $|M_i| = w$ `for` $1 \leq i \leq m$ `*/`
$V_1 \leftarrow F(K, M_1) \oplus c_0$;
for $i = 2$ **to** $m - 1$ **do** $V_i \leftarrow F(V_{i-1}, M_i)$;
if $|M| \geq 2w \wedge |M| \equiv_w 0$ **then** $V_m \leftarrow F(V_{m-1} \oplus c_{1,0}, M_m)$;
else $V_m \leftarrow F(V_{m-1} \oplus c_{1,1}, M_m)$;
return V_m;

Theorem 1 (Collision resistance of KHC1). *If a colliding input pair is found for* $\mathsf{KHC1}^F$*, then a* $\{0^n, c_0\}$*-colliding input pair is found for* F*.*

Theorem 2 (Collision resistance of KHC2). *If a colliding input pair is found for* $\mathsf{KHC2}^F$*, then a* $\{0^n, c_0, c_{1,0} \oplus c_{1,1}, c_0 \oplus c_{1,0} \oplus c_{1,1}\}$*-colliding input pair is found for* F*.*

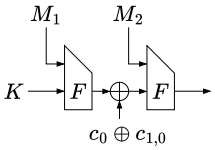
Theorem 3 (Pseudorandomness of KHC1). *For any adversary* $\mathbf{A}$ *taking at most* t *time and making at most* q *queries each of which has at most* ℓ_{m} *blocks after padding, there exist some adversaries* $\dot{\mathbf{A}}$ *and* $\ddot{\mathbf{A}}$ *such that*

$$\mathrm{Adv}^{\mathrm{prf}}_{\mathsf{KHC1}^F}(\mathbf{A}) \leq \mathrm{Adv}^{\text{prf-rka}}_{F,\{0^n,c_1\}}(\dot{\mathbf{A}}) + (\ell_{\mathrm{m}} - 1) \cdot \mathrm{Adv}^{q\text{-prf-rka}}_{F,\{0^n,c_1\}}(\ddot{\mathbf{A}}).$$

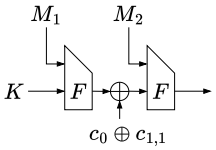
$\dot{\mathbf{A}}$ *and* $\ddot{\mathbf{A}}$ *take at most about* $t + O(q\ell_{\mathrm{m}}\tau_F)$ *time and make at most* q *queries, where* τ_F *is time required to compute* F*.*

Theorem 4 (Pseudorandomness of KHC2). *For any adversary* $\mathbf{A}$ *taking at most* t *time and making at most* q *queries each of which has at most* ℓ_{m} *blocks after padding, there exist some adversaries* $\dot{\mathbf{A}}$ *and* $\ddot{\mathbf{A}}$ *such that*

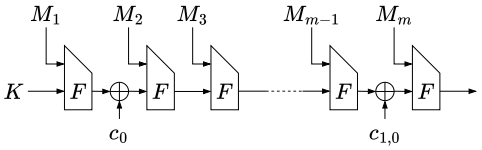
$$\mathrm{Adv}^{\mathrm{prf}}_{\mathsf{KHC2}^F}(\mathbf{A}) \leq \mathrm{Adv}^{\mathrm{prf}}_{F}(\dot{\mathbf{A}}) + (\ell_{\mathrm{m}} - 1) \cdot \mathrm{Adv}^{q\text{-prf-rka}}_{F,\{0^n,c_{1,0},c_{1,1}\}}(\ddot{\mathbf{A}}).$$



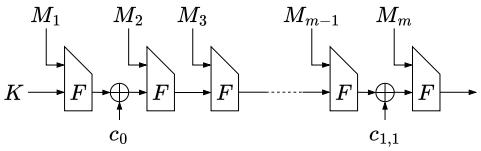
(a) If $|M| = 2w$



(b) If $0 \leq |M| \leq 2w - 1$



(c) If $|M| \geq 2w + 1$ and $|M| \equiv_w 0$



(d) If $|M| \geq 2w + 1$ and $|M| \not\equiv_w 0$.

Fig. 4: Keyed hashing mode KHC2

$\dot{\mathbf{A}}$ *and* **$\ddot{\mathbf{A}}$** *take at most about* $t + O(q\ell_{\mathrm{m}}\tau_F)$ *time and make at most* q *queries, where* τ_F *is time required to compute* F*.*

3 Vector-Input Keyed Hashing Modes

We propose two vector-input keyed hashing modes VIM1 and VIM2 using a hash function $H : \Sigma^n \times \Sigma^* \to \Sigma^n$, where $n \geq 1$ is an integer. H is treated as a keyed hash function with key space Σ^n. VIM1 and VIM2 are based on KHC1 and KHC2, respectively.

3.1 Description

VIM1 constructs a vector-input keyed hash function $\mathsf{VIM1}^H : \Sigma^n \times \Sigma^{**} \to \Sigma^n$. It is specified in Algorithm 3 and depicted in Fig. 1. It uses two distinct non-zero constants $d_0, d_1 \in \Sigma^n$.

VIM2 constructs a vector-input keyed hash function $\mathsf{VIM2}^H : \Sigma^n \times \Sigma^{**} \to \Sigma^n$. It is specified in Algorithm 4 and depicted in Fig. 2. It uses four distinct non-zero constants $d_0, d_1, d_2, d_3 \in \Sigma^n$.

3.2 Collision Resistance

Collision resistance of $\mathsf{VIM1}^H$ is implied by extended collision resistance of H:

Algorithm 3: $\mathsf{VIM1}^H : \Sigma^n \times \Sigma^{**} \to \Sigma^n$

input : (K, M)
output: $\mathsf{VIM1}^H(K, M)$
$m \leftarrow \dim(M)$; /* $M \coloneqq (M^1, \ldots, M^m)$ for $m \geq 1$ */
if $m = 0$ **then** $V_0 \leftarrow H(K \oplus d_1, 00)$;
else if $m = 1$ **then** $V_1 \leftarrow H(K \oplus d_1, M^1 \| 01)$;
else
  $V_1 \leftarrow H(K, M^1) \oplus d_0$;
  for $i = 2$ **to** $m - 1$ **do** $V_i \leftarrow H(V_{i-1}, M^i)$;
  $V_m \leftarrow H(V_{m-1} \oplus d_1, M^m \| 10)$;
return V_m;

Algorithm 4: $\mathsf{VIM2}^H : \Sigma^n \times \Sigma^{**} \to \Sigma^n$

input : (K, M)
output: $\mathsf{VIM2}^H(K, M)$
$m \leftarrow \dim(M)$; /* $M \coloneqq (M^1, \ldots, M^m)$ for $m \geq 1$ */
if $m = 0$ **then** $V_0 \leftarrow H(H(K, \varepsilon) \oplus d_0, \varepsilon)$;
else if $m = 1$ **then** $V_1 \leftarrow H(H(K, M^1) \oplus d_1, \varepsilon)$;
else if $m = 2$ **then** $V_2 \leftarrow H(H(K, M^1) \oplus d_2, M^2)$;
else
  $V_1 \leftarrow H(K, M^1) \oplus d_3$;
  for $i = 2$ **to** $m - 1$ **do** $V_i \leftarrow H(V_{i-1}, M^i)$;
  $V_m \leftarrow H(V_{m-1} \oplus d_3, M^m)$;
return V_m;

Theorem 5. *If a colliding input pair is found for* $\mathsf{VIM1}^H$*, then a* $\{0^n, d_0\}$*-colliding input pair is found for* H*.*

Proof. Suppose that a colliding input pair, (K, M) and $(\tilde{K}, \tilde{M})$, is found for $\mathsf{VIM1}^H$. Namely, $(K, M) \neq (\tilde{K}, \tilde{M})$ and $\mathsf{VIM1}^H(K, M) = \mathsf{VIM1}^H(\tilde{K}, \tilde{M})$. It is assumed that $\dim(M) \leq \dim(\tilde{M})$ without loss of generality.

Suppose that $\dim(M) = 0$. Then, $\dim(\tilde{M}) = 0$ and $K \neq \tilde{K}$, or $\dim(\tilde{M}) \geq 1$. Thus, a colliding input pair is found for H.

Suppose that $\dim(M) = 1$. Then, $\dim(\tilde{M}) \geq 1$. Thus, a colliding input pair is found for H.

Suppose that $\dim(M) = 2$. If $\dim(\tilde{M}) = 2$, then a colliding input pair is found for H. If $\dim(\tilde{M}) \geq 3$, then a colliding or $\{d_0\}$-colliding input pair is found for H.

Suppose that $\dim(M) \geq 3$. Then, $\dim(\tilde{M}) \geq 3$. If $\dim(M) < \dim(\tilde{M})$, then a colliding or $\{d_0\}$-colliding input pair is found for H. If $\dim(M) = \dim(\tilde{M})$, then a colliding input pair is found for H. □

Collision resistance of $\mathsf{VIM2}^H$ is also implied by extended collision resistance of H:

Theorem 6. *If a colliding input pair is found for* $\mathsf{VIM2}^H$*, then a* $(\{0^n, d_3\} \cup \{d_i \oplus d_j\}_{0 \le i < j \le 3})$*-colliding input pair is found for* H*.*

The proof is similar to that of Theorem 5 and is given in Appendix A.

3.3 Pseudorandomness

The following theorem states that $\mathsf{VIM1}^H$ is a secure vector-input PRF if H is a secure PRF under $\{0^n, d_1\}$-XRKAs:

Theorem 7. *For any adversary* $\mathbf{A}$ *taking at most* t *time and making at most* q *queries each of which is at most* ℓ_{v}*-dimensional, there exist some adversaries* $\dot{\mathbf{A}}$ *and* $\ddot{\mathbf{A}}$ *such that*

$$\mathrm{Adv}^{\mathrm{prf}}_{\mathsf{VIM1}^H}(\mathbf{A}) \le \mathrm{Adv}^{\text{prf-rka}}_{H,\{0^n,d_1\}}(\dot{\mathbf{A}}) + (\ell_{\mathrm{v}} - 1) \cdot \mathrm{Adv}^{q\text{-prf-rka}}_{H,\{0^n,d_1\}}(\ddot{\mathbf{A}}).$$

$\dot{\mathbf{A}}$ *and* $\ddot{\mathbf{A}}$ *take at most about* $t + O(q\,\ell_{\mathrm{v}}\,\tau_H)$ *time and make at most* q *queries, where* τ_H *is the time required to compute* H*.*

Proof. For $G : \Sigma^n \times \mathcal{S} \to \Sigma^n$ and $c \in \Sigma^n$, let $\mathsf{I}^{G,c} : \Sigma^n \times \mathcal{S}^+ \to \Sigma^n$ be a keyed function with key space Σ^n specified in Algorithm 5. For integer $k \ge 0$ and functions $\mu : \Sigma^{**} \to \Sigma^n$ and $\nu : \Sigma^{**} \to \Sigma^n$, let $\mathsf{Hy}[k] : \Sigma^{**} \to \Sigma^n$ be a function specified as follows: For $M \in \Sigma^{**}$ such that $\dim(M) = m$,

$$\mathsf{Hy}[0](M) \coloneqq \begin{cases} H(\nu(\lambda) \oplus d_1, 00) & \text{if } m = 0, \\ H(\nu(\lambda) \oplus d_1, M^1 \| 01) & \text{if } m = 1, \\ \mathsf{I}^{H,d_1}(H(\nu(\lambda), M^1) \oplus d_0, (M^2, \ldots, M^{m-1}, M^m \| 10)) & \text{if } m \ge 2, \end{cases}$$

$$\mathsf{Hy}[1](M) \coloneqq \begin{cases} \mu(M) & \text{if } m \le 1, \\ \mathsf{I}^{H,d_1}(\nu(M^1) \oplus d_0, (M^2, \ldots, M^{m-1}, M^m \| 10)) & \text{if } m \ge 2, \end{cases}$$

and, for $k \ge 2$,

$$\mathsf{Hy}[k](M) \coloneqq \begin{cases} \mu(M) & \text{if } m \le k, \\ \mathsf{I}^{H,d_1}(\nu(M^1, \ldots, M^k), (M^{k+1}, \ldots, M^{m-1}, M^m \| 10)) & \text{otherwise.} \end{cases}$$

Algorithm 5: $\mathsf{I}^{G,c} : \Sigma^n \times \mathcal{S}^+ \to \Sigma^n$

input : $(U, (X_1, X_2, \ldots, X_x))$
output: $\mathsf{I}^{G,c}(U, (X_1, X_2, \ldots, X_x))$
/* $U \in \Sigma^n$ and $X_i \in \mathcal{S}$ for $1 \le i \le x$ */
$V_0 \leftarrow U$;
for $i = 1$ **to** $x - 1$ **do** $V_i \leftarrow G(V_{i-1}, X_i)$;
$V_x \leftarrow G(V_{x-1} \oplus c, X_x)$;
return V_x;

Suppose that μ and ν are random oracles. Then, $\mathsf{Hy}[0]$ is equivalent to $\mathsf{VIM1}^H$, and $\mathsf{Hy}[\ell_{\mathrm{v}}]$ works as a random oracle for any $M \in \Sigma^{**}$ such that $\dim(M) \leq \ell_{\mathrm{v}}$. Since the dimension of every query made by $\mathbf{A}$ is assumed to be at most ℓ_{v},

$$\begin{aligned}\mathrm{Adv}^{\mathrm{prf}}_{\mathsf{VIM1}^H}(\mathbf{A}) &= \left|\Pr[\mathbf{A}^{\mathsf{Hy}[0]} = 1] - \Pr[\mathbf{A}^{\mathsf{Hy}[\ell_{\mathrm{v}}]} = 1]\right| \\ &\leq \sum_{i=0}^{\ell_{\mathrm{v}}-1} \left|\Pr[\mathbf{A}^{\mathsf{Hy}[i]} = 1] - \Pr[\mathbf{A}^{\mathsf{Hy}[i+1]} = 1]\right|. \qquad (1)\end{aligned}$$

For $\left|\Pr[\mathbf{A}^{\mathsf{Hy}[0]} = 1] - \Pr[\mathbf{A}^{\mathsf{Hy}[1]} = 1]\right|$, let $\mathbf{D}_0$ be an adversary against H. $\mathbf{D}_0$ has an oracle $\mathcal{G}\langle K\rangle$, which is either $H\langle K\rangle$ or $\rho\langle K\rangle$, where $K \twoheadleftarrow \Sigma^n$ and $\rho : \Sigma^n \times \Sigma^* \to \Sigma^n$ is a random oracle. $\mathbf{D}_0$ runs $\mathbf{A}$ and simulates the oracle of $\mathbf{A}$ using its oracle. $\mathbf{D}_0$ outputs the output of $\mathbf{A}$. For each query M made by $\mathbf{A}$ such that $M = \lambda$ if $m = 0$ and $M = (M^1, \ldots, M^m)$ if $m \geq 1$, $\mathbf{D}_0$ acts as follows:

- Suppose that $m = 0$. $\mathbf{D}_0$ asks $(\mathsf{x}_{d_1}, 00)$ to $\mathcal{G}\langle K\rangle$ and transfers the reply $\mathcal{G}(K \oplus d_1, 00)$ to $\mathbf{A}$.
- Suppose that $m = 1$. $\mathbf{D}_0$ asks $(\mathsf{x}_{d_1}, M^1\|01)$ to $\mathcal{G}\langle K\rangle$ and transfers the reply $\mathcal{G}(K \oplus d_1, M_1\|01)$ to $\mathbf{A}$.
- Suppose that $m \geq 2$. $\mathbf{D}_0$ asks (id, M^1) to $\mathcal{G}\langle K\rangle$, gets the reply $\mathcal{G}(K, M_1)$ and returns $\mathsf{I}^{H,d_1}(\mathcal{G}(K, M^1) \oplus d_0, (M^2, \ldots, M^{m-1}, M^m\|10))$ to $\mathbf{A}$.

$\mathbf{D}_0$ implements $\mathsf{Hy}[0]$ as the oracle of $\mathbf{A}$ if its oracle is $H\langle K\rangle$. It implements $\mathsf{Hy}[1]$ if its oracle is $\rho\langle K\rangle$. Thus,

$$\begin{aligned}\left|\Pr[\mathbf{A}^{\mathsf{Hy}[0]} = 1] - \Pr[\mathbf{A}^{\mathsf{Hy}[1]} = 1]\right| &= \left|\Pr[\mathbf{D}_0^{H\langle K\rangle} = 1] - \Pr[\mathbf{D}_0^{\rho\langle K\rangle} = 1]\right| \\ &= \mathrm{Adv}^{\text{prf-rka}}_{H,\{0^n,d_1\}}(\mathbf{D}_0). \qquad (2)\end{aligned}$$

$\mathbf{D}_0$ takes at most about $t + O(q\ell_{\mathrm{v}}\tau_H)$ time and makes at most q queries.

For $\left|\Pr[\mathbf{A}^{\mathsf{Hy}[1]} = 1] - \Pr[\mathbf{A}^{\mathsf{Hy}[2]} = 1]\right|$, let $\mathbf{D}_1$ be an adversary against H. $\mathbf{D}_1$ has oracles $\mathcal{G}_1\langle K_1 \oplus d_0\rangle, \ldots, \mathcal{G}_q\langle K_q \oplus d_0\rangle$, which are either $H\langle K_1 \oplus d_0\rangle, \ldots, H\langle K_q \oplus d_0\rangle$ or $\rho_1\langle K_1 \oplus d_0\rangle, \ldots, \rho_q\langle K_q \oplus d_0\rangle$, where $K_i \twoheadleftarrow \Sigma^n$ and $\rho_i : \Sigma^n \times \Sigma^* \to \Sigma^n$ is a random oracle for $1 \leq i \leq q$. $\mathbf{D}_1$ runs $\mathbf{A}$ and simulates the oracle of $\mathbf{A}$ using its oracle. $\mathbf{D}_1$ outputs the output of $\mathbf{A}$. For the j-th query M made by $\mathbf{A}$, where $M = \lambda$ if $m = 0$ and $M = (M^1, \ldots, M^m)$ if $m \geq 1$, $\mathbf{D}_1$ acts as follows:

- Suppose that $m \leq 1$. $\mathbf{D}_1$ simulates μ and returns $\mu(M)$ to $\mathbf{A}$.
- Suppose that $m \geq 2$. For $\tilde{\jmath} \leq j$, let $\tilde{M}$ be the $\tilde{\jmath}$-th query made by $\mathbf{A}$, where $\tilde{M} = M$ if $\tilde{\jmath} = j$. Let $j^* \coloneqq \min\{\tilde{\jmath} \,|\, (\tilde{\jmath} \leq j) \wedge (\dim(\tilde{M}) \geq 2) \wedge (\tilde{M}^1 = M^1)\}$.
  - If $m = 2$, then $\mathbf{D}_1$ asks $(\mathsf{x}_{d_1}, M^2\|10)$ to $\mathcal{G}_{j^*}\langle K_{j^*} \oplus d_0\rangle$ and transfers the reply $\mathcal{G}_{j^*}(K_{j^*} \oplus d_0 \oplus d_1, M^2\|10)$ to $\mathbf{A}$.
  - If $m \geq 3$, then $\mathbf{D}_1$ asks (id, M^2) to $\mathcal{G}_{j^*}\langle K_{j^*} \oplus d_0\rangle$, gets the reply $\mathcal{G}_{j^*}(K_{j^*} \oplus d_0, M^2)$, and returns $\mathsf{I}^{H,d_1}(\mathcal{G}_{j^*}(K_{j^*} \oplus d_0, M^2), (M^3, \ldots, M^{m-1}, M^m\|10))$ to $\mathbf{A}$.

$\mathbf{D}_1$ implements $\mathsf{Hy}[1]$ as the oracle of $\mathbf{A}$ if its oracles are $H\langle K_1 \oplus d_0\rangle, \ldots, H\langle K_q \oplus d_0\rangle$. $\mathbf{D}_1$ implements $\mathsf{Hy}[2]$ if its oracles are $\rho_1\langle K_1 \oplus d_0\rangle, \ldots, \rho_q\langle K_q \oplus d_0\rangle$. Thus,

$$\left|\Pr[\mathbf{A}^{\mathsf{Hy}[1]} = 1] - \Pr[\mathbf{A}^{\mathsf{Hy}[2]} = 1]\right| = \mathrm{Adv}^{q\text{-prf-rka}}_{H,\{0^n,d_1\}}(\mathbf{D}_1). \tag{3}$$

$\mathbf{D}_1$ takes at most about $t + O(q\,\ell_{\mathrm{v}}\,\tau_H)$ time and makes at most q queries.

For $2 \le k \le \ell_{\mathrm{v}} - 1$, let $\mathbf{D}_k$ be an adversary against H. $\mathbf{D}_k$ has oracles $\mathcal{G}_1\langle K_1\rangle, \ldots, \mathcal{G}_q\langle K_q\rangle$, which are either $H\langle K_1\rangle, \ldots, H\langle K_q\rangle$ or $\rho_1\langle K_1\rangle, \ldots, \rho_q\langle K_q\rangle$, where $K_i \twoheadleftarrow \Sigma^n$ and $\rho_i : \Sigma^n \times \Sigma^* \to \Sigma^n$ is a random oracle for $1 \le i \le q$. $\mathbf{D}_k$ runs $\mathbf{A}$ and simulates the oracle of $\mathbf{A}$ using its oracle. $\mathbf{D}_k$ outputs the output of $\mathbf{A}$. For the j-th query M made by $\mathbf{A}$, where $M = \lambda$ if $m = 0$ and $M = (M^1, \ldots, M^m)$ if $m \ge 1$, $\mathbf{D}_k$ acts as follows:

- Suppose that $m \le k$. $\mathbf{D}_k$ simulates μ and returns $\mu(M)$ to $\mathbf{A}$.
- Suppose that $m \ge k+1$. For $\tilde{j} \le j$, let $\tilde{M}$ be the $\tilde{j}$-th query made by $\mathbf{A}$, where $\tilde{M} = M$ if $\tilde{j} = j$. Let $j^* \coloneqq \min\{\tilde{j} \,|\, (\tilde{j} \le j) \wedge (\dim(\tilde{M}) \ge k+1) \wedge ((\tilde{M}^1, \ldots, \tilde{M}^k) = (M^1, \ldots, M^k))\}$.
  - If $m = k+1$, then $\mathbf{D}_k$ asks $(\mathsf{x}_{d_1}, M^{k+1})$ to $\mathcal{G}_{j^*}\langle K_{j^*}\rangle$ and transfers the reply $\mathcal{G}_{j^*}(K_{j^*} \oplus d_1, M^{k+1}\|10)$ to $\mathbf{A}$.
  - If $m \ge k+2$, then $\mathbf{D}_k$ asks (id, M^{k+1}) to $\mathcal{G}_{j^*}\langle K_{j^*}\rangle$, gets $\mathcal{G}_{j^*}(K_{j^*}, M^{k+1})$ and returns $\mathsf{I}^{H,d_1}(\mathcal{G}_{j^*}(K_{j^*}, M^{k+1}), (M^{k+2}, \ldots, M^{m-1}, M^m\|10))$ to $\mathbf{A}$.

$\mathbf{D}_k$ implements $\mathsf{Hy}[k]$ as the oracle of $\mathbf{A}$ if its oracles are $H_1\langle K_1\rangle, \ldots, H_q\langle K_q\rangle$. $\mathbf{D}_k$ implements $\mathsf{Hy}[k+1]$ if its oracles are $\rho_1\langle K_1\rangle, \ldots, \rho_q\langle K_q\rangle$. Thus,

$$\left|\Pr[\mathbf{A}^{\mathsf{Hy}[k]} = 1] - \Pr[\mathbf{A}^{\mathsf{Hy}[k+1]} = 1]\right| = \mathrm{Adv}^{q\text{-prf-rka}}_{H,\{0^n,d_1\}}(\mathbf{D}_k). \tag{4}$$

$\mathbf{D}_k$ takes at most about $t + O(q\,\ell_{\mathrm{v}}\,\tau_H)$ time and makes at most q queries.

From Inequality (1) and Equalities (2), (3), and (4), there exist adversaries $\dot{\mathbf{A}}$ and $\ddot{\mathbf{A}}$ such that

$$\mathrm{Adv}^{\mathrm{prf}}_{\mathsf{VIM1}^H}(\mathbf{A}) \le \mathrm{Adv}^{\mathrm{prf\text{-}rka}}_{H,\{0^n,d_1\}}(\dot{\mathbf{A}}) + (\ell_{\mathrm{v}} - 1) \cdot \mathrm{Adv}^{q\text{-prf-rka}}_{H,\{0^n,d_1\}}(\ddot{\mathbf{A}}).$$

$\dot{\mathbf{A}}$ and $\ddot{\mathbf{A}}$ take at most about $t + O(q\,\ell_{\mathrm{v}}\,\tau_H)$ time and make at most q queries. □

The following theorem states that $\mathsf{VIM2}^H$ is a secure vector-input PRF if H is a secure PRF under $\{0^n, d_0, d_1, d_2, d_3\}$-XRKAs:

Theorem 8. *For any adversary $\mathbf{A}$ taking at most t time and making at most q queries each of which is at most ℓ_{v}-dimensional, there exist adversaries $\dot{\mathbf{A}}$, $\ddot{\mathbf{A}}$, and $\dddot{\mathbf{A}}$ such that*

$$\mathrm{Adv}^{\mathrm{prf}}_{\mathsf{VIM2}^H}(\mathbf{A}) \le \mathrm{Adv}^{\mathrm{prf}}_{H}(\dot{\mathbf{A}}) + \mathrm{Adv}^{q\text{-prf-rka}}_{H,\{0^n,d_0,d_1,d_2,d_3\}}(\ddot{\mathbf{A}}) + (\ell_{\mathrm{v}} - 2) \cdot \mathrm{Adv}^{q\text{-prf-rka}}_{H,\{0^n,d_3\}}(\dddot{\mathbf{A}}).$$

$\dot{\mathbf{A}}$, $\ddot{\mathbf{A}}$, and $\dddot{\mathbf{A}}$ take at most about $t + O(q\,\ell_{\mathrm{v}}\,\tau_H)$ time and make at most q queries, where τ_H is time required to compute H.

The proof is similar to that of Theorem 7 and is given in Appendix B.

4 Further Security Analyses of KHC1 and KHC2

Our goal is to construct vector-input CR PRFs by using $\mathsf{KHC1}^F$ or $\mathsf{KHC2}^F$ for keyed hash function H of $\mathsf{VIM1}^H$ or $\mathsf{VIM2}^H$. Thus, in this section, we will see that the security requirements of $\mathsf{VIM1}^H$ and $\mathsf{VIM2}^H$ for H, that is, extended collision resistance and PRF under related-key attacks, can be satisfied by $\mathsf{KHC1}^F$ and $\mathsf{KHC2}^F$.

4.1 Extended Collision Resistance

The following theorems are obtained from Theorems 1 and 2 and from the fact that, for any $\delta \in \Sigma^n \setminus \{0^n\}$, a $\{\delta\}$-colliding input pair for $\mathsf{KHC1}^F$ or $\mathsf{KHC2}^F$ yields a $\{\delta\}$-colliding input pair for F:

Theorem 9. *For any $\Delta \subseteq \Sigma^n$ such that $0^n \in \Delta$, a Δ-colliding input pair for $\mathsf{KHC1}^F$ yields a $(\Delta \cup \{c_0\})$-colliding input pair for F.*

Theorem 10. *For any $\Delta \subseteq \Sigma^n$ such that $0^n \in \Delta$, a Δ-colliding input pair for $\mathsf{KHC2}^F$ yields a $(\Delta \cup \{c_0, c_{1,0} \oplus c_{1,1}, c_0 \oplus c_{1,0} \oplus c_{1,1}\})$-colliding input pair for F.*

4.2 PRF under Related-Key Attacks

The following theorem states that $\mathsf{KHC1}^F$ is a secure PRF under Δ-XRKAs if Δ and $\Delta \oplus c_1$ are mutually disjoint, and F is a secure PRF under $(\Delta \cup (\Delta \oplus c_1))$-XRKAs:

Theorem 11. *Let $\Delta \subseteq \Sigma^n$ be any set such that $0^n \in \Delta$. Suppose that $\Delta \cap (\Delta \oplus c_1) = \emptyset$. Then, for any adversary $\mathbf{A}$ taking at most t time and making at most q queries each of which has at most ℓ_{m} blocks after padding, there exist some adversaries $\dot{\mathbf{A}}$ and $\ddot{\mathbf{A}}$ such that*

$$\mathrm{Adv}^{p\text{-prf-rka}}_{\mathsf{KHC1}^F,\Delta}(\mathbf{A}) \le \mathrm{Adv}^{p\text{-prf-rka}}_{F,\Delta\cup(\Delta\oplus c_1)}(\dot{\mathbf{A}}) + (\ell_{\mathrm{m}} - 1) \cdot \mathrm{Adv}^{q\text{-prf-rka}}_{F,\{0^n,c_1\}}(\ddot{\mathbf{A}}).$$

$\dot{\mathbf{A}}$ and $\ddot{\mathbf{A}}$ take at most about $t + O(q\ell_{\mathrm{m}}\tau_F)$ time and make at most q queries, where τ_F is time required to compute F.

Proof. For $G : \Sigma^n \times \mathcal{S} \to \Sigma^n$ and $c \in \Sigma^n$, let $\mathsf{I}^{G,c} : \Sigma^n \times \mathcal{S}^+ \to \Sigma^n$ be a keyed function specified in Algorithm 5. For integer $k \ge 0$ and functions $\mu : \Sigma^n \times \Sigma^* \to \Sigma^n$ and $\nu : \Sigma^n \times (\Sigma^w)^+ \to \Sigma^n$, let $\mathsf{Hy}[k] : \Sigma^n \times \Sigma^* \to \Sigma^n$ be a function specified as follows: For $K \in \Sigma^n$ and $M \in \Sigma^*$ such that $\mathtt{pad1}(M) = M_1 \| \cdots \| M_m$,

$$\mathsf{Hy}[0](K, M) \coloneqq \begin{cases} F(K \oplus c_1, M_1) & \text{if } m = 1, \\ \mathsf{I}^{F,c_1}(F(K, M_1) \oplus c_0, M_{[2,m]}) & \text{otherwise,} \end{cases}$$

$$\mathsf{Hy}[1](K, M) \coloneqq \begin{cases} \mu(K, M) & \text{if } m \le 1, \\ \mathsf{I}^{F,c_1}(\nu(K, M_1) \oplus c_0, M_{[2,m]}) & \text{if } m \ge 2, \end{cases}$$

and, for $k \geq 2$,

$$\mathsf{Hy}[k](K, M) \coloneqq \begin{cases} \mu(K, M) & \text{if } m \leq k, \\ \mathsf{I}^{F,c_1}(\nu(K, M_{[1,k]}), M_{[k+1,m]}) & \text{if } m \geq k+1. \end{cases}$$

Suppose that μ and ν are random oracles. Then, $\mathsf{Hy}[0]$ is equivalent to $\mathsf{KHC1}^F$, and $\mathsf{Hy}[\ell_{\mathrm{m}}]$ works as a random oracle for any $M \in \Sigma^*$ such that $\mathtt{pad1}(M)$ consists of at most ℓ_{m} blocks. Since every query made by $\mathbf{A}$ is assumed to consist of at most ℓ_{m} blocks after padding,

$$\begin{aligned} &\mathrm{Adv}^{p\text{-prf-rka}}_{\mathsf{KHC1}^F,\Delta}(\mathbf{A}) \\ &\quad = \left|\Pr[\mathbf{A}^{\mathsf{Hy}_1[0]\langle K_1\rangle,\ldots,\mathsf{Hy}_p[0]\langle K_p\rangle} = 1] - \Pr[\mathbf{A}^{\mathsf{Hy}_1[\ell_{\mathrm{m}}]\langle K_1\rangle,\ldots,\mathsf{Hy}_p[\ell_{\mathrm{m}}]\langle K_p\rangle} = 1]\right| \\ &\quad \leq \sum_{i=0}^{\ell_{\mathrm{m}}-1} \left|\Pr[\mathbf{A}^{\mathsf{Hy}_1[i]\langle K_1\rangle,\ldots,\mathsf{Hy}_p[i]\langle K_p\rangle} = 1] - \right. \\ &\qquad\qquad \left. \Pr[\mathbf{A}^{\mathsf{Hy}_1[i+1]\langle K_1\rangle,\ldots,\mathsf{Hy}_p[i+1]\langle K_p\rangle} = 1]\right|, \end{aligned} \tag{5}$$

where $\mathsf{Hy}_j[i]\langle K_j\rangle$ uses random oracles μ_j and ν_j. $\mu_1, \ldots, \mu_p$ and $\nu_1, \ldots, \nu_p$ are independent as well as $K_1, \ldots, K_p$.

For $\left|\Pr[\mathbf{A}^{\mathsf{Hy}_1[0]\langle K_1\rangle,\ldots,\mathsf{Hy}_p[0]\langle K_p\rangle} = 1] - \Pr[\mathbf{A}^{\mathsf{Hy}_1[1]\langle K_1\rangle,\ldots,\mathsf{Hy}_p[1]\langle K_p\rangle} = 1]\right|$, let $\mathbf{D}_0$ be an adversary against F. $\mathbf{D}_0$ has oracles $\mathcal{G}_1\langle K'_1\rangle, \ldots, \mathcal{G}_p\langle K'_p\rangle$, which are either $F\langle K'_1\rangle, \ldots, F\langle K'_p\rangle$ or $\rho_1\langle K'_1\rangle, \ldots, \rho_p\langle K'_p\rangle$, where $K'_i \twoheadleftarrow \Sigma^n$, and $\rho_i : \Sigma^n \times \Sigma^w \to \Sigma^n$ is a random oracle for $1 \leq i \leq p$. $\mathbf{D}_0$ runs $\mathbf{A}$ and simulates the oracle of $\mathbf{A}$ using its oracles. $\mathbf{D}_0$ outputs the output of $\mathbf{A}$. For each query (x_d, M) made by $\mathbf{A}$ to the u-th oracle such that $\mathtt{pad1}(M) = M_1 \| \cdots \| M_m$, $\mathbf{D}_0$ acts as follows:

- Suppose that $m = 1$. $\mathbf{D}_0$ asks $(\mathsf{x}_{d \oplus c_1}, M_1)$ to $\mathcal{G}_u\langle K'_u\rangle$ and transfers the reply $\mathcal{G}_u(K'_u \oplus d \oplus c_1, M_1)$ to $\mathbf{A}$.
- Suppose that $m \geq 2$. $\mathbf{D}_0$ asks (x_d, M_1) to $\mathcal{G}_u\langle K'_u\rangle$, gets $\mathcal{G}_u(K'_u \oplus d, M_1)$ and returns $\mathsf{I}^{F,c_1}(\mathcal{G}_u(K'_u \oplus d, M_1) \oplus c_0, M_{[2,m]})$ to $\mathbf{A}$.

$\mathbf{D}_0$ implements $\mathsf{Hy}_1[0]\langle K_1\rangle, \ldots, \mathsf{Hy}_p[0]\langle K_p\rangle$ as the oracles of $\mathbf{A}$ if its oracles are $F\langle K'_1\rangle, \ldots, F\langle K'_p\rangle$. It implements $\mathsf{Hy}_1[1]\langle K_1\rangle, \ldots, \mathsf{Hy}_p[1]\langle K_p\rangle$ if its oracles are $\rho_1\langle K'_1\rangle, \ldots, \rho_p\langle K'_p\rangle$ since $\Delta \cap (\Delta \oplus c_1) = \emptyset$. Thus,

$$\begin{aligned} &\left|\Pr[\mathbf{A}^{\mathsf{Hy}_1[0]\langle K_1\rangle,\ldots,\mathsf{Hy}_p[0]\langle K_p\rangle} = 1] - \Pr[\mathbf{A}^{\mathsf{Hy}_1[1]\langle K_1\rangle,\ldots,\mathsf{Hy}_p[1]\langle K_p\rangle} = 1]\right| \\ &\quad = \left|\Pr[\mathbf{D}_0^{F\langle K'_1\rangle,\ldots,F\langle K'_p\rangle} = 1] - \Pr[\mathbf{D}_0^{\rho_1\langle K'_1\rangle,\ldots,\rho_p\langle K'_p\rangle} = 1]\right| \\ &\quad = \mathrm{Adv}^{p\text{-prf-rka}}_{F,\Delta\cup(\Delta\oplus c_1)}(\mathbf{D}_0). \end{aligned} \tag{6}$$

$\mathbf{D}_0$ takes at most about $t + O(q\,\ell_{\mathrm{m}}\,\tau_F)$ time and makes at most q queries.

For $\left|\Pr[\mathbf{A}^{\mathsf{Hy}_1[1]\langle K_1\rangle,\ldots,\mathsf{Hy}_p[1]\langle K_p\rangle} = 1] - \Pr[\mathbf{A}^{\mathsf{Hy}_1[2]\langle K_1\rangle,\ldots,\mathsf{Hy}_p[2]\langle K_p\rangle} = 1]\right|$, let $\mathbf{D}_1$ be an adversary against F. $\mathbf{D}_1$ has oracles $\mathcal{G}_1\langle K'_1 \oplus c_0\rangle, \ldots, \mathcal{G}_q\langle K'_q \oplus c_0\rangle$, which are either $F\langle K'_1 \oplus c_0\rangle, \ldots, F\langle K'_q \oplus c_0\rangle$ or $\rho_1\langle K'_1 \oplus c_0\rangle, \ldots, \rho_q\langle K'_q \oplus c_0\rangle$, where $K'_i \twoheadleftarrow \Sigma^n$ and $\rho_i : \Sigma^n \times \Sigma^w \to \Sigma^n$ is a random oracle for $1 \leq i \leq q$. $\mathbf{D}_1$ runs $\mathbf{A}$ and simulates the oracles of $\mathbf{A}$ using its oracles. $\mathbf{D}_1$ outputs the output of $\mathbf{A}$. For the j-th query (x_d, M) made by $\mathbf{A}$ to the u-th oracle such that $\mathtt{pad1}(M) = M_1 \| \cdots \| M_m$, $\mathbf{D}_1$ acts as follows:

– Suppose that $m = 1$. Then, $\mathbf{D}_1$ simulates μ_u and returns $\mu_u(K_u \oplus d, M)$ to $\mathbf{A}$.
– Suppose that $m \geq 2$. For $\tilde{\jmath} \leq j$, let $(\mathsf{x}_{\tilde{d}}, \tilde{M})$ be the $\tilde{\jmath}$-th query made by $\mathbf{A}$ to the $\tilde{u}$-th oracle, and $\mathtt{pad1}(\tilde{M}) = \tilde{M}_1 \| \cdots \| \tilde{M}_{\tilde{m}}$. Let

$$j^* \coloneqq \min\{\tilde{\jmath} \,|\, (\tilde{\jmath} \leq j) \wedge (\tilde{u} = u) \wedge (\tilde{d} = d) \wedge (\tilde{m} \geq 2) \wedge (\tilde{M}_1 = M_1)\}.$$

  - If $m = 2$, then $\mathbf{D}_1$ asks (x_{c_1}, M_2) to $\mathcal{G}_{j^*}\langle K'_{j^*} \oplus c_0\rangle$ and transfers the reply $\mathcal{G}_{j^*}(K'_{j^*} \oplus c_0 \oplus c_1, M_2)$ to $\mathbf{A}$.
  - If $m \geq 3$, then $\mathbf{D}_1$ asks (id, M_2) to $\mathcal{G}_{j^*}\langle K'_{j^*} \oplus c_0\rangle$, gets $\mathcal{G}_{j^*}(K'_{j^*} \oplus c_0, M_2)$ and returns $\mathsf{I}^{F,c_1}(\mathcal{G}_{j^*}(K'_{j^*} \oplus c_0, M_2), M_{[3,m]})$ to $\mathbf{A}$.

$\mathbf{D}_1$ implements $\mathsf{Hy}_1[1]\langle K_1\rangle, \ldots, \mathsf{Hy}_p[1]\langle K_p\rangle$ as the oracles of $\mathbf{A}$ if its oracles are $F\langle K'_1 \oplus c_0\rangle, \ldots, F\langle K'_q \oplus c_0\rangle$. It implements $\mathsf{Hy}_1[2]\langle K_1\rangle, \ldots, \mathsf{Hy}_p[2]\langle K_p\rangle$ if its oracles are $\rho_1\langle K'_1 \oplus c_0\rangle, \ldots, \rho_q\langle K'_q \oplus c_0\rangle$. Thus,

$$\begin{aligned}&\left|\Pr[\mathbf{A}^{\mathsf{Hy}_1[1]\langle K_1\rangle, \ldots, \mathsf{Hy}_p[1]\langle K_p\rangle} = 1] - \Pr[\mathbf{A}^{\mathsf{Hy}_1[2]\langle K_1\rangle, \ldots, \mathsf{Hy}_p[2]\langle K_p\rangle} = 1]\right| \\ &\quad = \mathrm{Adv}^{q\text{-prf-rka}}_{F,\{0^n,c_1\}}(\mathbf{D}_1). \end{aligned} \tag{7}$$

$\mathbf{D}_1$ takes at most about $t + O(q\,\ell_{\mathrm{m}}\,\tau_F)$ time and makes at most q queries.

For $2 \leq k \leq \ell_{\mathrm{m}} - 1$, let $\mathbf{D}_k$ be an adversary against F. $\mathbf{D}_k$ has oracles $\mathcal{G}_1\langle K'_1\rangle, \ldots, \mathcal{G}_q\langle K'_q\rangle$, which are either $F\langle K'_1\rangle, \ldots, F\langle K'_q\rangle$ or $\rho_1\langle K'_1\rangle, \ldots, \rho_q\langle K'_q\rangle$, where $K'_i \twoheadleftarrow \Sigma^n$ and $\rho_i : \Sigma^n \times \Sigma^w \to \Sigma^n$ is a random oracle for $1 \leq i \leq q$. $\mathbf{D}_k$ runs $\mathbf{A}$ and simulates the oracles of $\mathbf{A}$ using its oracles. $\mathbf{D}_k$ outputs the output of $\mathbf{A}$. For the j-th query (x_d, M) made by $\mathbf{A}$ to the u-th oracle such that $\mathtt{pad1}(M) = M_1 \| \cdots \| M_m$, $\mathbf{D}_k$ acts as follows:

– Suppose that $m \leq k$. Then, $\mathbf{D}_k$ simulates μ_u and returns $\mu_u(K_u \oplus d, M)$ to $\mathbf{A}$.
– Suppose that $m \geq k + 1$. For $\tilde{\jmath} \leq j$, let $(\mathsf{x}_{\tilde{d}}, \tilde{M})$ be the $\tilde{\jmath}$-th query made by $\mathbf{A}$ to the $\tilde{u}$-th oracle, and $\mathtt{pad1}(\tilde{M}) = \tilde{M}_1 \| \cdots \| \tilde{M}_{\tilde{m}}$. Let

$$j^* \coloneqq \min\{\tilde{\jmath} \,|\, (\tilde{\jmath} \leq j) \wedge (\tilde{u} = u) \wedge (\tilde{d} = d) \wedge (\tilde{m} \geq k+1) \wedge (\tilde{M}_{[1,k]} = M_{[1,k]})\}.$$

  - If $m = k + 1$, then $\mathbf{D}_k$ asks $(\mathsf{x}_{c_1}, M_{k+1})$ to $\mathcal{G}_{j^*}\langle K'_{j^*}\rangle$ and transfers the reply $\mathcal{G}_{j^*}(K'_{j^*} \oplus c_1, M_{k+1})$ to $\mathbf{A}$.
  - If $m \geq k+2$, then $\mathbf{D}_k$ asks (id, M_{k+1}) to $\mathcal{G}_{j^*}\langle K'_{j^*}\rangle$, gets $\mathcal{G}_{j^*}(K'_{j^*}, M_{k+1})$ returns $\mathsf{I}^{F,c_1}(\mathcal{G}_{j^*}(K'_{j^*}, M_{k+1}), M_{[k+2,m]})$ to $\mathbf{A}$.

$\mathbf{D}_k$ implements $\mathsf{Hy}_1[k]\langle K_1\rangle, \ldots, \mathsf{Hy}_p[k]\langle K_p\rangle$ as the oracle of $\mathbf{A}$ if its oracles are $F\langle K'_1\rangle, \ldots, F\langle K'_q\rangle$. $\mathbf{D}_k$ implements $\mathsf{Hy}_1[k+1]\langle K_1\rangle, \ldots, \mathsf{Hy}_p[k+1]\langle K_p\rangle$ if its oracles are $\rho_1\langle K'_1\rangle, \ldots, \rho_q\langle K'_q\rangle$. Thus,

$$\begin{aligned}&\left|\Pr[\mathbf{A}^{\mathsf{Hy}_1[k]\langle K_1\rangle, \ldots, \mathsf{Hy}_p[k]\langle K_p\rangle} = 1] - \Pr[\mathbf{A}^{\mathsf{Hy}_1[k+1]\langle K_1\rangle, \ldots, \mathsf{Hy}_p[k+1]\langle K_p\rangle} = 1]\right| \\ &\quad = \mathrm{Adv}^{q\text{-prf-rka}}_{F,\{0^n,c_1\}}(\mathbf{D}_1). \end{aligned} \tag{8}$$

$\mathbf{D}_k$ takes at most about $t + O(q\,\ell_{\mathrm{m}}\,\tau_F)$ time and makes at most q queries.

From Inequality (5) and Equalities (6), (7), and (8), there exist adversaries $\dot{\mathbf{A}}$ and $\ddot{\mathbf{A}}$ such that

$$\mathrm{Adv}^{p\text{-prf-rka}}_{\mathsf{KHC1}^F,\Delta}(\mathbf{A}) \le \mathrm{Adv}^{p\text{-prf-rka}}_{F,\Delta\cup(\Delta\oplus c_1)}(\dot{\mathbf{A}}) + (\ell_{\mathrm{m}} - 1) \cdot \mathrm{Adv}^{q\text{-prf-rka}}_{F,\{0^n,c_1\}}(\ddot{\mathbf{A}}).$$

$\dot{\mathbf{A}}$ and $\ddot{\mathbf{A}}$ take at most about $t + O(q\,\ell_{\mathrm{m}}\,\tau_F)$ time and make at most q queries. □

The following theorem states that $\mathsf{KHC2}^F$ is a secure PRF under Δ-XRKAs if F is a secure PRF under Δ-XRKAs and $\{0^n, c_{1,0}, c_{1,1}\}$-XRKAs:

Theorem 12. *Let $\Delta \subseteq \Sigma^n$ be any set such that $0^n \in \Delta$. Then, for any adversary $\mathbf{A}$ taking at most t time and making at most q queries each of which has at most ℓ_{m} blocks after padding, there exist adversaries $\dot{\mathbf{A}}$ and $\ddot{\mathbf{A}}$ such that*

$$\mathrm{Adv}^{p\text{-prf-rka}}_{\mathsf{KHC2}^F,\Delta}(\mathbf{A}) \le \mathrm{Adv}^{p\text{-prf-rka}}_{F,\Delta}(\dot{\mathbf{A}}) + (\ell_{\mathrm{m}} - 1) \cdot \mathrm{Adv}^{q\text{-prf-rka}}_{F,\{0^n,c_{1,0},c_{1,1}\}}(\ddot{\mathbf{A}}).$$

$\dot{\mathbf{A}}$ and $\ddot{\mathbf{A}}$ take at most about $t + O(\ell_{\mathrm{m}} \cdot q \cdot \tau_F)$ time and make at most q queries, where τ_F is time required to compute F.

The proof is similar to that of Theorem 11 and is given in Appendix C.

5 Vector-Input CR PRF

This section summarizes the results on reduction of security of vector-input keyed hash functions $\mathsf{VIM1}^{\mathsf{KHC1}^F}$, $\mathsf{VIM1}^{\mathsf{KHC2}^F}$, $\mathsf{VIM2}^{\mathsf{KHC1}^F}$, and $\mathsf{VIM2}^{\mathsf{KHC2}^F}$ to that of the underlying compression function F. All the proofs are straightforward and omitted.

5.1 Collision Resistance

Collision resistance of $\mathsf{VIM1}^{\mathsf{KHC1}^F}$, $\mathsf{VIM1}^{\mathsf{KHC2}^F}$, $\mathsf{VIM2}^{\mathsf{KHC1}^F}$, and $\mathsf{VIM2}^{\mathsf{KHC2}^F}$ is reduced to extended collision resistance of F:

Corollary 1. *A colliding input pair for $\mathsf{VIM1}^{\mathsf{KHC1}^F}$ yields a $\{0^n, c_0, d_0\}$-colliding input pair for F.*

Corollary 2. *A colliding input pair for $\mathsf{VIM1}^{\mathsf{KHC2}^F}$ yields a $\{0^n, c_0, c_{1,0} \oplus c_{1,1}, c_0 \oplus c_{1,0} \oplus c_{1,1}, d_0\}$-colliding input pair for F.*

Corollary 3. *A colliding input pair for $\mathsf{VIM2}^{\mathsf{KHC1}^F}$ yields a $(\{0^n, c_0, d_3\} \cup \{d_i \oplus d_j\}_{0 \le i < j \le 3})$-colliding input pair for F.*

Corollary 4. *A colliding input pair for $\mathsf{VIM2}^{\mathsf{KHC2}^F}$ yields a $(\{0^n, c_0, c_{1,0} \oplus c_{1,1}, c_0 \oplus c_{1,0} \oplus c_{1,1}, d_3\} \cup \{d_i \oplus d_j\}_{0 \le i < j \le 3})$-colliding input pair for F.*

The corollaries above suggest that any adversary treating F as a black box needs $\Omega(2^{n/2})$ queries to find a colliding input pair for $\mathsf{VIM1}^{\mathsf{KHC1}^F}$, $\mathsf{VIM1}^{\mathsf{KHC2}^F}$, $\mathsf{VIM2}^{\mathsf{KHC1}^F}$, or $\mathsf{VIM2}^{\mathsf{KHC2}^F}$.

5.2 PRF Property

Pseudorandomness of $\mathsf{VIM1}^{\mathsf{KHC1}^F}$, $\mathsf{VIM1}^{\mathsf{KHC2}^F}$, $\mathsf{VIM2}^{\mathsf{KHC1}^F}$, and $\mathsf{VIM2}^{\mathsf{KHC2}^F}$ is reduced to pseudorandomness of F under XOR related-key attacks:

Corollary 5. *Suppose that $c_1 \neq d_1$. Let $\mathbf{A}$ be any adversary against $\mathsf{VIM1}^{\mathsf{KHC1}^F}$ taking at most t time and making at most q queries. Suppose that for each query $M = (M^1, \ldots, M^m)$ made by $\mathbf{A}$, $m \leq \ell_{\mathrm{v}}$ and padded M^i results in at most ℓ_{m} blocks for $1 \leq i \leq m$. Then, there exist adversaries $\dot{\mathbf{A}}$ and $\ddot{\mathbf{A}}$ such that*

$$\mathrm{Adv}^{\mathrm{prf}}_{\mathsf{VIM1}^{\mathsf{KHC1}^F}}(\mathbf{A}) \leq \ell_{\mathrm{v}} \cdot \mathrm{Adv}^{q\text{-prf-rka}}_{F,\{0^n,c_1,d_1,c_1\oplus d_1)}(\dot{\mathbf{A}}) + \ell_{\mathrm{v}} \cdot \ell_{\mathrm{m}} \cdot \mathrm{Adv}^{q\text{-prf-rka}}_{F,\{0^n,c_1\}}(\ddot{\mathbf{A}}).$$

$\dot{\mathbf{A}}$ and $\ddot{\mathbf{A}}$ take at most about $t + O(q\,\ell_{\mathrm{v}}\,\ell_{\mathrm{m}}\,\tau_F)$ time and make at most q queries, where τ_F is the time required to compute F.

Corollary 6. *Let $\mathbf{A}$ be any adversary against $\mathsf{VIM1}^{\mathsf{KHC2}^F}$ taking at most t time and making at most q queries. Suppose that for each query $M = (M^1, \ldots, M^m)$ made by $\mathbf{A}$, $m \leq \ell_{\mathrm{v}}$ and padded M^i results in at most ℓ_{m} blocks for $1 \leq i \leq m$. Then, there exist adversaries $\dot{\mathbf{A}}$ and $\ddot{\mathbf{A}}$ such that*

$$\mathrm{Adv}^{\mathrm{prf}}_{\mathsf{VIM1}^{\mathsf{KHC2}^F}}(\mathbf{A}) \leq \ell_{\mathrm{v}} \cdot \mathrm{Adv}^{q\text{-prf-rka}}_{F,\{0^n,d_1\}}(\dot{\mathbf{A}}) + \ell_{\mathrm{v}} \cdot \ell_{\mathrm{m}} \cdot \mathrm{Adv}^{q\text{-prf-rka}}_{F,\{0^n,c_{1,0},c_{1,1}\}}(\ddot{\mathbf{A}}).$$

$\dot{\mathbf{A}}$ and $\ddot{\mathbf{A}}$ take at most about $t + O(q\,\ell_{\mathrm{v}}\,\ell_{\mathrm{m}}\,\tau_F)$ time and make at most q queries, where τ_F is the time required to compute F.

Corollary 7. *Suppose that $c_1 \notin \{d_0, d_1, d_2, d_3\} \cup \{d_i \oplus d_j\}_{0\leq i<j\leq 3}$. Let $\mathbf{A}$ be any adversary against $\mathsf{VIM2}^{\mathsf{KHC1}^F}$ taking at most t time and making at most q queries. Suppose that for each query $M = (M^1, \ldots, M^m)$ made by $\mathbf{A}$, $m \leq \ell_{\mathrm{v}}$ and padded M^i results in at most ℓ_{m} blocks for $1 \leq i \leq m$. Then, there exist adversaries $\dot{\mathbf{A}}$, $\ddot{\mathbf{A}}$, and $\dddot{\mathbf{A}}$ such that*

$$\begin{aligned}\mathrm{Adv}^{\mathrm{prf}}_{\mathsf{VIM2}^{\mathsf{KHC1}^F}}(\mathbf{A}) \leq\ & \mathrm{Adv}^{q\text{-prf-rka}}_{F,\{0^n,c_1\}\cup\{d_i,c_1\oplus d_i\}_{0\leq i\leq 3}}(\dot{\mathbf{A}}) \\ & + \ell_{\mathrm{v}} \cdot \mathrm{Adv}^{q\text{-prf-rka}}_{F,\{0^n,c_1,d_3,c_1\oplus d_3\}}(\ddot{\mathbf{A}}) + \ell_{\mathrm{v}} \cdot \ell_{\mathrm{m}} \cdot \mathrm{Adv}^{q\text{-prf-rka}}_{F,\{0^n,c_1\}}(\dddot{\mathbf{A}}).\end{aligned}$$

$\dot{\mathbf{A}}$, $\ddot{\mathbf{A}}$, and $\dddot{\mathbf{A}}$ take at most about $t + O(q\,\ell_{\mathrm{v}}\,\ell_{\mathrm{m}}\,\tau_F)$ time and make at most q queries, where τ_F is time required to compute F.

Corollary 8. *Let $\mathbf{A}$ be any adversary against $\mathsf{VIM2}^{\mathsf{KHC2}^F}$ taking at most t time and making at most q queries. Suppose that for each query $M = (M^1, \ldots, M^m)$ made by $\mathbf{A}$, $m \leq \ell_{\mathrm{v}}$ and padded M^i results in at most ℓ_{m} blocks for $1 \leq i \leq m$. Then, there exist adversaries $\dot{\mathbf{A}}$, $\ddot{\mathbf{A}}$, and $\dddot{\mathbf{A}}$ such that*

$$\begin{aligned}\mathrm{Adv}^{\mathrm{prf}}_{\mathsf{VIM2}^{\mathsf{KHC2}^F}}(\mathbf{A}) \leq\ & \mathrm{Adv}^{q\text{-prf-rka}}_{F,\{0^n,d_0,d_1,d_2,d_3\}}(\dot{\mathbf{A}}) \\ & + \ell_{\mathrm{v}} \cdot \mathrm{Adv}^{q\text{-prf-rka}}_{F,\{0^n,d_3\}}(\ddot{\mathbf{A}}) + \ell_{\mathrm{v}} \cdot \ell_{\mathrm{m}} \cdot \mathrm{Adv}^{q\text{-prf-rka}}_{F,\{0^n,c_{1,0},c_{1,1}\}}(\dddot{\mathbf{A}}).\end{aligned}$$

$\dot{\mathbf{A}}$, $\ddot{\mathbf{A}}$, and $\dddot{\mathbf{A}}$ take at most about $t + O(q\,\ell_{\mathrm{v}}\,\ell_{\mathrm{m}}\,\tau_F)$ time and make at most q queries, where τ_F is time required to compute F.

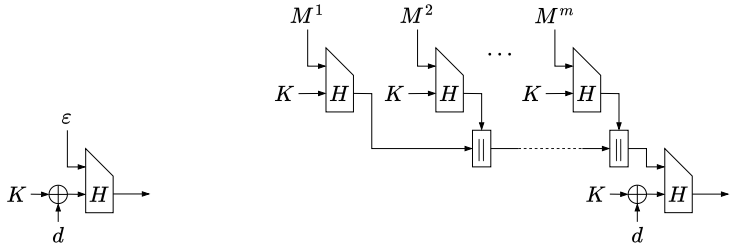
In Corollaries 5, 6, 7, and 8, the number of message blocks in each query is at most $\ell_{\mathrm{v}}\,\ell_{\mathrm{m}}$. Thus, their upper bounds are as tight as the upper bound shown by Bellare et al. [4] for the cascade construction of a string-input PRF using a compression function.

6 Parallelizable Vector-Input Keyed Hashing Mode

We present a parallelizable vector-input keyed hashing mode PVIM. It constructs a vector-input keyed hash function $\mathsf{PVIM}^H : \Sigma^n \times \Sigma^{**} \to \Sigma^n$, where $H : \Sigma^n \times \Sigma^* \to \Sigma^n$ is a keyed hash function. It is specified in Algorithm 6 and depicted in Fig. 5, where $d \in \Sigma^n$ is a non-zero constant.

Algorithm 6: $\mathsf{PVIM}^H : \Sigma^n \times \Sigma^{**} \to \Sigma^n$

input : (K, M)
output: $\mathsf{PVIM}^H(K, M)$
$m \leftarrow \dim(M)$; `/*` $M \coloneqq (M^1, \ldots, M^m)$ `for` $m \geq 1$ `*/`
if $m = 0$ **then** $V \leftarrow H(K \oplus d, \varepsilon)$;
else $V \leftarrow H(K \oplus d, H(K, M^1) \| H(K, M^2) \| \cdots \| H(K, M^m))$;
return V;



(a) $\dim(M) = 0$ (b) $\dim(M) \geq 1$

Fig. 5: Proposed parallelizable vector-input keyed hashing mode PVIM

Collision resistance of PVIM^H is implied by collision resistance of H:

Theorem 13. *If a colliding pair of inputs is found for* PVIM^H*, then a colliding pair of inputs is found for* H*.*

The proof of Theorem 13 is straightforward and omitted.

PVIM^H is a secure PRF if H is a secure PRF under $\{0^n, d\}$-XRKAs:

Theorem 14. *Let* $\mathbf{A}$ *be any adversary against* PVIM^H *taking at most* t *time and making at most* q *queries. Suppose that the total number of strings in the*

queries (the sum of dimensions of the queries) made by $\mathbf{A}$ *is at most* σ*. Then, there exists an adversary* $\dot{\mathbf{A}}$ *such that*

$$\mathrm{Adv}^{\mathrm{prf}}_{\mathsf{PVIM}^H}(\mathbf{A}) \le \mathrm{Adv}^{\text{prf-rka}}_{H,\{0^n,d\}}(\dot{\mathbf{A}}) + \frac{\sigma^2}{2^{n+1}}.$$

$\dot{\mathbf{A}}$ *takes at most about* t *time and makes at most* $(q+\sigma)$ *queries.*

Proof. Let $\rho_{\mathrm{v}} : \Sigma^{**} \to \Sigma^n$ be a random oracle. Let $\dot{\mathbf{A}}$ be an adversary against H making a $\{0^n, d\}$-XRKA. The oracle of $\dot{\mathbf{A}}$ is either $H\langle K\rangle$ or $\rho\langle K\rangle$, where $K \twoheadleftarrow \Sigma^n$, and $\rho : \Sigma^n \times \Sigma^* \to \Sigma^n$ is a random oracle. $\dot{\mathbf{A}}$ runs $\mathbf{A}$ and simulates the oracle of $\mathbf{A}$ using its oracle. Finally, $\dot{\mathbf{A}}$ outputs the output of $\mathbf{A}$. Then,

$$\begin{aligned}
\mathrm{Adv}^{\mathrm{prf}}_{\mathsf{PVIM}^H}(\mathbf{A}) &= \left|\Pr[\mathbf{A}^{\mathsf{PVIM}^H(K,\cdot)} = 1] - \Pr[\mathbf{A}^{\rho_{\mathrm{v}}} = 1]\right| \\
&\le \left|\Pr[\mathbf{A}^{\mathsf{PVIM}^H(K,\cdot)} = 1] - \Pr[\mathbf{A}^{\mathsf{PVIM}^\rho(K,\cdot)} = 1]\right| \\
&\qquad + \left|\Pr[\mathbf{A}^{\mathsf{PVIM}^\rho(K,\cdot)} = 1] - \Pr[\mathbf{A}^{\rho_{\mathrm{v}}} = 1]\right| \\
&\le \left|\Pr[\dot{\mathbf{A}}^{H\langle K\rangle} = 1] - \Pr[\dot{\mathbf{A}}^{\rho\langle K\rangle} = 1]\right| + \left|\Pr[\mathbf{A}^{\mathsf{PVIM}^\rho(K,\cdot)} = 1] - \Pr[\mathbf{A}^{\rho_{\mathrm{v}}} = 1]\right| \\
&\le \mathrm{Adv}^{\text{prf-rka}}_{H,\{0^n,d\}}(\dot{\mathbf{A}}) + \left|\Pr[\mathbf{A}^{\mathsf{PVIM}^\rho(K,\cdot)} = 1] - \Pr[\mathbf{A}^{\rho_{\mathrm{v}}} = 1]\right|.
\end{aligned}$$

$\dot{\mathbf{A}}$ takes at most about t time and makes at most $(q+\sigma)$ queries.

$\mathbf{A}$ can distinguish $\mathsf{PVIM}^\rho(K,\cdot)$ and ρ_{v} only if a collision is found among the inputs to the outer (final) ρ for $\mathsf{PVIM}^\rho(K,\cdot)$, which implies that a colliding pair of inputs is obtained for inner ρ during the execution of $\mathbf{A}$. Thus,

$$\left|\Pr[\mathbf{A}^{\mathsf{PVIM}^\rho(K,\cdot)} = 1] - \Pr[\mathbf{A}^{\rho_{\mathrm{v}}} = 1]\right| \le \frac{\sigma^2}{2^{n+1}}.$$

This completes the proof. □

When $\mathsf{KHC1}^F$ or $\mathsf{KHC2}^F$ is used for H, PVIM^H generally needs more calls to F than $\mathsf{VIM1}^H$ and $\mathsf{VIM2}^H$ though it is parallelizable.

7 Conclusion

We have proposed vector-input keyed hashing modes, VIM1, VIM2, and PVIM using a string-input keyed hash function. The sequential modes VIM1 and VIM2 extend conventional string-input keyed hashing modes KHC1 and KHC2 using a compression function. We have combined VIM1, VIM2, or PVIM with KHC1 or KHC2 and constructed vector-input CR PRFs by using a compression function satisfying extended collision resistance and pseudorandomness under related-key attacks. Though PVIM allows parallel processing, it requires more calls to the underlying compression function than VIM1 and VIM2.

Acknowledgments. The first and third authors were supported by JSPS KAKENHI Grant Number 24K14944.

A Proof of Theorem 6

Suppose that a colliding input pair, (K, M) and $(\tilde{K}, \tilde{M})$, is found for $\mathsf{VIM2}^H$. Namely, $(K, M) \neq (\tilde{K}, \tilde{M})$ and $\mathsf{VIM2}^H(K, M) = \mathsf{VIM2}^H(\tilde{K}, \tilde{M})$. It is assumed that $\dim(M) \leq \dim(\tilde{M})$ without loss of generality.

Suppose that $\dim(M) = 0$. Then, $\dim(\tilde{M}) = 0$ and $K \neq \tilde{K}$, or $\dim(\tilde{M}) \geq 1$. Thus, a colliding or $\{d_0 \oplus d_1, d_0 \oplus d_2, d_0 \oplus d_3\}$-colliding input pair is found for H.

Suppose that $\dim(M) = 1$. Then, $\dim(\tilde{M}) \geq 1$. Thus, a colliding or $\{d_1 \oplus d_2, d_1 \oplus d_3\}$-colliding input pair is found for H.

Suppose that $\dim(M) = 2$. If $\dim(\tilde{M}) = 2$, then a colliding input pair is found for H. If $\dim(\tilde{M}) \geq 3$, then a colliding or $\{d_2 \oplus d_3\}$-colliding input pair is found for H.

Suppose that $\dim(M) \geq 3$. Then, $\dim(\tilde{M}) \geq 3$. If $\dim(M) < \dim(\tilde{M})$, then a colliding or $\{d_3\}$-colliding input pair is found for H. If $\dim(M) = \dim(\tilde{M})$, then a colliding input pair is found for H.

B Proof of Theorem 8

Let $\mathsf{I}^{G,c} : \Sigma^n \times \mathcal{S}^+ \to \Sigma^n$ be a keyed function specified in Algorithm 5. For an integer $k \geq 0$ and functions $\mu : \Sigma^{**} \to \Sigma^n$ and $\nu : \Sigma^{**} \to \Sigma^n$, let $\mathsf{Hy}[k] : \Sigma^{**} \to \Sigma^n$ be a function specified as follows: For $M \in \Sigma^{**}$ such that $\dim(M) = m$,

$$\mathsf{Hy}[0](M) \coloneqq \begin{cases} H(H(\nu(\lambda), \varepsilon) \oplus d_0, \varepsilon) & \text{if } m = 0, \\ H(H(\nu(\lambda), M^1) \oplus d_1, \varepsilon) & \text{if } m = 1, \\ H(H(\nu(\lambda), M^1) \oplus d_2, M^2) & \text{if } m = 2, \\ \mathsf{I}^{H,d_3}(H(\nu(\lambda), M^1) \oplus d_3, (M^2, \ldots, M^m)) & \text{if } m \geq 3, \end{cases}$$

$$\mathsf{Hy}[1](M) \coloneqq \begin{cases} H(\nu((\varepsilon)) \oplus d_0, \varepsilon) & \text{if } m = 0, \\ H(\nu((M^1)) \oplus d_1, \varepsilon) & \text{if } m = 1, \\ H(\nu((M^1)) \oplus d_2, M^2) & \text{if } m = 2, \\ \mathsf{I}^{H,d_3}(\nu((M^1)) \oplus d_3, (M^2, \ldots, M^m)) & \text{if } m \geq 3, \end{cases}$$

and for $k \geq 2$,

$$\mathsf{Hy}[k](M) \coloneqq \begin{cases} \mu(M) & \text{if } m \leq k, \\ \mathsf{I}^{H,d_3}(\nu((M^1, \ldots, M^k)), (M^{k+1}, \ldots, M^m)) & \text{if } m \geq k+1. \end{cases}$$

Suppose that μ and ν are random oracles. Then, $\mathsf{Hy}[0]$ is equivalent to $\mathsf{VIM2}^H$, and $\mathsf{Hy}[\ell_{\mathrm{v}}]$ works as a random oracle for any $M \in \Sigma^{**}$ such that $\dim(M) \leq \ell_{\mathrm{v}}$. Since the dimension of every query made by $\mathbf{A}$ is assumed to be at most ℓ_{v},

$$\begin{aligned} \mathrm{Adv}^{\mathrm{prf}}_{\mathsf{VIM2}^H}(\mathbf{A}) &= \left|\Pr[\mathbf{A}^{\mathsf{Hy}[0]} = 1] - \Pr[\mathbf{A}^{\mathsf{Hy}[\ell_{\mathrm{v}}]} = 1]\right| \\ &\leq \sum_{i=0}^{\ell_{\mathrm{v}}-1} \left|\Pr[\mathbf{A}^{\mathsf{Hy}[i]} = 1] - \Pr[\mathbf{A}^{\mathsf{Hy}[i+1]} = 1]\right|. \end{aligned} \tag{9}$$

For $\left|\Pr[\mathbf{A}^{\mathsf{Hy}[0]} = 1] - \Pr[\mathbf{A}^{\mathsf{Hy}[1]} = 1]\right|$, let $\mathbf{D}_0$ be an adversary against H. $\mathcal{G}$ has an oracle $\mathbf{D}_0$, which is either H or ρ, where $\rho : \Sigma^* \to \Sigma^n$ is a random oracle. $\mathbf{D}_0$ runs $\mathbf{A}$ and simulates the oracle of $\mathbf{A}$ using its oracle. $\mathbf{D}_0$ outputs the output of $\mathbf{A}$. For each query M made by $\mathbf{A}$ such that $M = \lambda$ if $m = 0$ and $M = (M^1, \ldots, M^m)$ if $m \geq 1$, $\mathbf{D}_0$ acts as follows:

- Suppose that $m = 0$. $\mathbf{D}_0$ asks ε to $\mathcal{G}$, gets $\mathcal{G}(\varepsilon)$, and returns $H(\mathcal{G}(\varepsilon) \oplus d_0, \varepsilon)$ to $\mathbf{A}$.
- Suppose that $m = 1$. $\mathbf{D}_0$ asks M^1 to $\mathcal{G}$, gets $\mathcal{G}(M^1)$, and returns $H(\mathcal{G}(M^1) \oplus d_1, \varepsilon)$ to $\mathbf{A}$.
- Suppose that $m = 2$. $\mathbf{D}_0$ asks M^1 to $\mathcal{G}$, gets $\mathcal{G}(M^1)$, and returns $H(\mathcal{G}(M^1) \oplus d_2, M^2)$ to $\mathbf{A}$.
- Suppose that $m \geq 3$. $\mathbf{D}_0$ asks M^1 to $\mathcal{G}$, gets $\mathcal{G}(M^1)$, and returns $\mathsf{I}^{H,d_3}(\mathcal{G}(M^1) \oplus d_3, (M^2, \ldots, M^m))$ to $\mathbf{A}$.

$\mathbf{D}_0$ implements $\mathsf{Hy}[0]$ as the oracle of $\mathbf{A}$ if its oracle is H. It implements $\mathsf{Hy}[1]$ if its oracle is ρ. Thus,

$$\begin{aligned}\left|\Pr[\mathbf{A}^{\mathsf{Hy}[0]} = 1] - \Pr[\mathbf{A}^{\mathsf{Hy}[1]} = 1]\right| &= \left|\Pr[\mathbf{D}_0^H = 1] - \Pr[\mathbf{D}_0^\rho = 1]\right| \\ &= \mathrm{Adv}^{\mathrm{prf}}_H(\mathbf{D}_0). \end{aligned} \tag{10}$$

$\mathbf{D}_0$ takes at most about $t + O(q\,\ell_{\mathrm{v}}\,\tau_H)$ time and makes at most q queries.

For $\left|\Pr[\mathbf{A}^{\mathsf{Hy}[1]} = 1] - \Pr[\mathbf{A}^{\mathsf{Hy}[2]} = 1]\right|$, let $\mathbf{D}_1$ be an adversary against H. $\mathbf{D}_1$ hash oracles $\mathcal{G}_1\langle K_1\rangle, \ldots, \mathcal{G}_q\langle K_q\rangle$, which are either $H\langle K_1\rangle, \ldots, H\langle K_q\rangle$ or $\rho_1\langle K_1\rangle, \ldots, \rho_q\langle K_q\rangle$, where $K_i \xleftarrow{} \Sigma^n$ and $\rho_i : \Sigma^n \times \Sigma^* \to \Sigma^n$ is a random oracle for $1 \leq i \leq q$. $\mathbf{D}_1$ runs $\mathbf{A}$ and simulates the oracle of $\mathbf{A}$ using its oracles. $\mathbf{D}_1$ outputs the output of $\mathbf{A}$. Let M be the j-th query made by $\mathbf{A}$, where $M = \lambda$ if $m = 0$ and $M = (M^1, \ldots, M^m)$ if $m \geq 1$. For $\tilde{\jmath} \leq j$, let $\tilde{M} \coloneqq (\tilde{M}^1, \ldots, \tilde{M}^{\tilde{m}})$ be the $\tilde{\jmath}$-th query made by $\mathbf{A}$, where $\tilde{M} = \lambda$ if $\tilde{m} = 0$. For the j-th query M, $\mathbf{D}_1$ acts as follows:

- Suppose that $m = 0$. Let $j^* \coloneqq \min\{\tilde{\jmath} \,|\, (\tilde{\jmath} \leq j) \wedge (\tilde{M}^1 = \varepsilon)\}$. $\mathbf{D}_1$ asks $(\mathsf{x}_{d_0}, \varepsilon)$ to $\mathcal{G}_{j^*}\langle K_{j^*}\rangle$ and transfers the reply $\mathcal{G}_{j^*}(K_{j^*} \oplus d_0, \varepsilon)$ to $\mathbf{A}$.
- Suppose that $m \geq 1$. Let

  $$j^* \coloneqq \min\{\tilde{\jmath} \,|\, (\tilde{\jmath} \leq j) \wedge ((\tilde{M}^1 = M^1) \vee (M^1 = \varepsilon) \wedge (\tilde{M} = \lambda))\}.$$

  If $m = 1$, then $\mathbf{D}_1$ asks $(\mathsf{x}_{d_1}, \varepsilon)$ to $\mathcal{G}_{j^*}\langle K_{j^*}\rangle$ and transfers the reply $\mathcal{G}_{j^*}(K_{j^*} \oplus d_1, \varepsilon)$ to $\mathbf{A}$. If $m = 2$, then $\mathbf{D}_1$ asks (x_{d_2}, M^2) to $\mathcal{G}_{j^*}\langle K_{j^*}\rangle$ and transfers the reply $\mathcal{G}_{j^*}(K_{j^*} \oplus d_2, M^2)$ to $\mathbf{A}$. If $m \geq 3$, then $\mathbf{D}_1$ asks (x_{d_3}, M^2) to $\mathcal{G}_{j^*}\langle K_{j^*}\rangle$, gets the reply $\mathcal{G}_{j^*}(K_{j^*} \oplus d_3, M^2)$ and returns $\mathsf{I}^{H,d_3}(\mathcal{G}_{j^*}(K_{j^*} \oplus d_3, M^2), (M^3, \ldots, M^m))$ to $\mathbf{A}$.

$\mathbf{D}_1$ implements $\mathsf{Hy}[1]$ as the oracle of $\mathbf{A}$ if its oracles are $H\langle K_1\rangle, \ldots, H\langle K_q\rangle$. It implements $\mathsf{Hy}[2]$ if its oracles are $\rho_1\langle K_1\rangle, \ldots, \rho_q\langle K_q\rangle$. Thus,

$$\left|\Pr[\mathbf{A}^{\mathsf{Hy}[1]} = 1] - \Pr[\mathbf{A}^{\mathsf{Hy}[2]} = 1]\right| = \mathrm{Adv}^{q\text{-prf-rka}}_{H,\{0^n,d_0,d_1,d_2,d_3\}}(\mathbf{D}_1). \tag{11}$$

$\mathbf{D}_1$ takes at most about $t + O(q\ell_{\mathrm{v}}\tau_H)$ time and makes at most q queries.

For $2 \le k \le \ell_{\mathrm{v}} - 1$, let $\mathbf{D}_k$ be a prf-rka-adversary against H. $\mathbf{D}_k$ has oracles $\mathcal{G}_1\langle K_1\rangle, \ldots, \mathcal{G}_q\langle K_q\rangle$, which are either $H\langle K_1\rangle, \ldots, H\langle K_q\rangle$ or $\rho_1\langle K_1\rangle, \ldots, \rho_q\langle K_q\rangle$, where $K_i \twoheadleftarrow \Sigma^n$, and $\rho_i : \Sigma^n \times \Sigma^* \to \Sigma^n$ is a random oracle for $1 \le i \le q$. $\mathbf{D}_k$ runs $\mathbf{A}$ and simulates the oracle of $\mathbf{A}$ using its oracles. $\mathbf{D}_k$ outputs the output of $\mathbf{A}$. For the j-th query M made by $\mathbf{A}$, $\mathbf{D}_k$ acts as follows:

- Suppose that $m \le k$. Then, $\mathbf{D}_k$ simulates μ and returns $\mu(M)$ to $\mathbf{A}$.
- Suppose that $m \ge k+1$. Let $j^* := \min\{\tilde{j} \,|\, (\tilde{j} \le j) \wedge (\dim(\tilde{M}) \ge k+1) \wedge ((\tilde{M}^1, \ldots, \tilde{M}^k) = (M^1, \ldots, M^k))\}$. If $m = k+1$, then $\mathbf{D}_k$ asks $(\mathsf{x}_{d_3}, M^{k+1})$ to $\mathcal{G}_{j^*}\langle K_{j^*}\rangle$ and transfers the reply $\mathcal{G}_{j^*}(K_{j^*} \oplus d_3, M^{k+1})$ to $\mathbf{A}$. If $m \ge k+2$, then $\mathbf{D}_k$ asks (id, M^{k+1}) to $\mathcal{G}_{j^*}\langle K_{j^*}\rangle$, gets $\mathcal{G}_{j^*}(K_{j^*}, M^{k+1})$ and returns $\mathsf{I}^{H,d_3}(\mathcal{G}_{j^*}(K_{j^*}, M^{k+1}), (M^{k+2}, \ldots, M^m))$ to $\mathbf{A}$.

$\mathbf{D}_k$ implements $\mathsf{Hy}[k]$ as the oracle of $\mathbf{A}$ if its oracles are $H\langle K_1\rangle, \ldots, H\langle K_q\rangle$. $\mathbf{D}_k$ implements $\mathsf{Hy}[k+1]$ if its oracles are $\rho_1\langle K_1\rangle, \ldots, \rho_q\langle K_q\rangle$. Thus,

$$\left|\Pr[\mathbf{A}^{\mathsf{Hy}[k]} = 1] - \Pr[\mathbf{A}^{\mathsf{Hy}[k+1]} = 1]\right| = \mathrm{Adv}^{q\text{-prf-rka}}_{F,\{0^n,d_3\}}(\mathbf{D}_k). \tag{12}$$

$\mathbf{D}_k$ takes at most about $t + O(q\ell_{\mathrm{v}}\tau_H)$ time and makes at most q queries.

From Inequality (9), Equalities (10), (11), and (12), there exists an adversary $\dot{\mathbf{A}}$ such that

$$\mathrm{Adv}^{\mathrm{prf}}_{\mathsf{VIM2}^H}(\mathbf{A}) \le \mathrm{Adv}^{\mathrm{prf}}_{H}(\dot{\mathbf{A}}) + \mathrm{Adv}^{q\text{-prf-rka}}_{H,\{0^n,d_0,d_1,d_2,d_3\}}(\ddot{\mathbf{A}}) + (\ell_{\mathrm{v}}-2)\cdot\mathrm{Adv}^{q\text{-prf-rka}}_{H,\{0^n,d_3\}}(\dddot{\mathbf{A}}).$$

$\dot{\mathbf{A}}$, $\ddot{\mathbf{A}}$ and $\dddot{\mathbf{A}}$ take at most about $t + O(q\ell_{\mathrm{v}}\tau_H)$ time and make at most q queries.

C Proof of Theorem 12

For $G : \Sigma^n \times \mathcal{S} \to \Sigma^n$ and $c \in \Sigma^n$, let $\mathsf{I}^{G,c} : \Sigma^n \times \mathcal{S}^+ \to \Sigma^n$ be a keyed function specified in Algorithm 5. For an integer $k \ge 0$ and functions $\mu : \Sigma^n \times \Sigma^* \to \Sigma^n$ and $\nu : \Sigma^n \times (\Sigma^w)^+ \to \Sigma^n$, let $\mathsf{Hy}[k] : \Sigma^n \times \Sigma^* \to \Sigma^n$ be a function specified as follows: For $K \in \Sigma^n$ and $M \in \Sigma^*$ such that $\mathtt{pad2}(M) = M_1 \| \cdots \| M_m$,

$$\mathsf{Hy}[0](K, M) := \begin{cases} \mathsf{I}^{F,c_{1,0}}(F(K, M_1) \oplus c_0, M_{[2,m]}) & \text{if } |M| \ge 2w \wedge |M| \equiv_w 0, \\ \mathsf{I}^{F,c_{1,1}}(F(K, M_1) \oplus c_0, M_{[2,m]}) & \text{otherwise,} \end{cases}$$

$$\mathsf{Hy}[1](K, M) := \begin{cases} \mathsf{I}^{F,c_{1,0}}(\nu(K, M_1) \oplus c_0, M_{[2,m]}) & \text{if } |M| \ge 2w \wedge |M| \equiv_w 0, \\ \mathsf{I}^{F,c_{1,1}}(\nu(K, M_1) \oplus c_0, M_{[2,m]}) & \text{otherwise,} \end{cases}$$

and for $k \ge 2$,

$$\mathsf{Hy}[k](K, M) := \begin{cases} \mu(K, M) & \text{if } m \le k, \\ \mathsf{I}^{F,c_{1,0}}(\nu(K, M_{[1,k]}), M_{[k+1,m]}) & \text{if } m \ge k+1 \wedge |M| \equiv_w 0, \\ \mathsf{I}^{F,c_{1,1}}(\nu(K, M_{[1,k]}), M_{[k+1,m]}) & \text{if } m \ge k+1 \wedge |M| \not\equiv_w 0. \end{cases}$$

Suppose that μ and ν are random oracles. Then, $\mathsf{Hy}[0]$ is equivalent to $\mathsf{KHC2}^F$, and $\mathsf{Hy}[\ell_{\mathrm{m}}]$ works as a random oracle for any $M \in \Sigma^*$ such that $\mathtt{pad2}(M)$ consists of at most ℓ_{m} blocks. Since every query made by $\mathbf{A}$ is assumed to consist of at most ℓ_{m} blocks after padding,

$$\begin{aligned}
&\mathrm{Adv}^{p\text{-prf-rka}}_{\mathsf{KHC2}^F,\Delta}(\mathbf{A}) \\
&\quad = \left|\Pr[\mathbf{A}^{\mathsf{Hy}_1[0]\langle K_1\rangle,\ldots,\mathsf{Hy}_p[0]\langle K_p\rangle} = 1] - \Pr[\mathbf{A}^{\mathsf{Hy}_1[\ell_{\mathrm{m}}]\langle K_1\rangle,\ldots,\mathsf{Hy}_p[\ell_{\mathrm{m}}]\langle K_p\rangle} = 1]\right| \\
&\quad \le \sum_{i=0}^{\ell_{\mathrm{m}}-1} \left|\Pr[\mathbf{A}^{\mathsf{Hy}_1[i]\langle K_1\rangle,\ldots,\mathsf{Hy}_p[i]\langle K_p\rangle} = 1] - \right. \\
&\qquad\qquad \left. \Pr[\mathbf{A}^{\mathsf{Hy}_1[i+1]\langle K_1\rangle,\ldots,\mathsf{Hy}_p[i+1]\langle K_p\rangle} = 1]\right|, \qquad (13)
\end{aligned}$$

where $\mathsf{Hy}_j[i]\langle K_j\rangle$ uses random oracles μ_j and ν_j. $\mu_1, \ldots, \mu_p$ and $\nu_1, \ldots, \nu_p$ are independent as well as $K_1, \ldots, K_p$.

For $\left|\Pr[\mathbf{A}^{\mathsf{Hy}_1[0]\langle K_1\rangle,\ldots,\mathsf{Hy}_p[0]\langle K_p\rangle} = 1] - \Pr[\mathbf{A}^{\mathsf{Hy}_1[1]\langle K_1\rangle,\ldots,\mathsf{Hy}_p[1]\langle K_p\rangle} = 1]\right|$, let $\mathbf{D}_0$ be an adversary against F. $\mathbf{D}_0$ has oracles $\mathcal{G}_1\langle K'_1\rangle, \ldots, \mathcal{G}_p\langle K'_p\rangle$, which are either $F\langle K'_1\rangle, \ldots, F\langle K'_p\rangle$ or $\rho_1\langle K'_1\rangle, \ldots, \rho_p\langle K'_p\rangle$, where $K'_i \twoheadleftarrow \Sigma^n$, and $\rho_i : \Sigma^n \times \Sigma^w \to \Sigma^n$ is a random oracle for $1 \le i \le p$. $\mathbf{D}_0$ runs $\mathbf{A}$ and simulates the oracles of $\mathbf{A}$ using its oracles. $\mathbf{D}_0$ outputs the output of $\mathbf{A}$. For each query (x_d, M) made by $\mathbf{A}$ to the u-th oracle such that $\mathtt{pad2}(M) = M_1 \| \cdots \| M_m$, $\mathbf{D}_0$ acts as follows:

- If $|M| \ge 2w$ and $|M| \equiv_w 0$, then $\mathbf{D}_0$ asks (x_d, M_1) to $\mathcal{G}_u\langle K'_u\rangle$, gets $\mathcal{G}_u(K'_u \oplus d, M_1)$ and returns $\mathsf{I}^{F,c_{1,0}}(\mathcal{G}_u(K'_u \oplus d, M_1) \oplus c_0, M_{[2,m]})$ to $\mathbf{A}$.
- Otherwise, $\mathbf{D}_0$ asks (x_d, M_1) to $\mathcal{G}_u\langle K'_u\rangle$, gets $\mathcal{G}_u(K'_u \oplus d, M_1)$ and returns $\mathsf{I}^{F,c_{1,1}}(\mathcal{G}_u(K'_u \oplus d, M_1) \oplus c_0, M_{[2,m]})$ to $\mathbf{A}$.

$\mathbf{D}_0$ implements $\mathsf{Hy}_1[0]\langle K_1\rangle, \ldots, \mathsf{Hy}_p[0]\langle K_p\rangle$ as the oracles of $\mathbf{A}$ if its oracles are $F\langle K'_1\rangle, \ldots, F\langle K'_p\rangle$. It implements $\mathsf{Hy}_1[1]\langle K_1\rangle, \ldots, \mathsf{Hy}_p[1]\langle K_p\rangle$ if its oracles are $\rho_1\langle K'_1\rangle, \ldots, \rho_p\langle K'_p\rangle$. Thus,

$$\begin{aligned}
&\left|\Pr[\mathbf{A}^{\mathsf{Hy}_1[0]\langle K_1\rangle,\ldots,\mathsf{Hy}_p[0]\langle K_p\rangle} = 1] - \Pr[\mathbf{A}^{\mathsf{Hy}_1[1]\langle K_1\rangle,\ldots,\mathsf{Hy}_p[1]\langle K_p\rangle} = 1]\right| \\
&\quad = \left|\Pr[\mathbf{D}_0^{F\langle K'_1\rangle,\ldots,F\langle K'_p\rangle} = 1] - \Pr[\mathbf{D}_0^{\rho_1\langle K'_1\rangle,\ldots,\rho_p\langle K'_p\rangle} = 1]\right| \\
&\quad = \mathrm{Adv}^{p\text{-prf-rka}}_{F,\Delta}(\mathbf{D}_0). \qquad (14)
\end{aligned}$$

$\mathbf{D}_0$ takes at most about $t + O(q\,\ell_{\mathrm{m}}\,\tau_F)$ time and makes at most q queries.

For $\left|\Pr[\mathbf{A}^{\mathsf{Hy}_1[1]\langle K_1\rangle,\ldots,\mathsf{Hy}_p[1]\langle K_p\rangle} = 1] - \Pr[\mathbf{A}^{\mathsf{Hy}_1[2]\langle K_1\rangle,\ldots,\mathsf{Hy}_p[2]\langle K_p\rangle} = 1]\right|$, let $\mathbf{D}_1$ be an adversary against F. $\mathbf{D}_1$ has oracles $\mathcal{G}_1\langle K'_1 \oplus c_0\rangle, \ldots, \mathcal{G}_q\langle K'_q \oplus c_0\rangle$, which are either $F\langle K'_1 \oplus c_0\rangle, \ldots, F\langle K'_q \oplus c_0\rangle$ or $\rho_1\langle K'_1 \oplus c_0\rangle, \ldots, \rho_q\langle K'_q \oplus c_0\rangle$, where $K'_i \twoheadleftarrow \Sigma^n$ and $\rho_i : \Sigma^n \times \Sigma^w \to \Sigma^n$ is a random oracle for $1 \le i \le q$. $\mathbf{D}_1$ runs $\mathbf{A}$ and simulates the oracles of $\mathbf{A}$ using its oracles. $\mathbf{D}_1$ outputs the output of $\mathbf{A}$. For the j-th query (x_d, M) made by $\mathbf{A}$ to the u-th oracle, let $\mathtt{pad2}(M) = M_1 \| \cdots \| M_m$. For $\tilde{\jmath} \le j$, let $(\mathsf{x}_{\tilde{d}}, \tilde{M})$ be the $\tilde{\jmath}$-th query made by $\mathbf{A}$ to the $\tilde{u}$-th oracle, and $\mathtt{pad2}(\tilde{M}) = \tilde{M}_1 \| \cdots \| \tilde{M}_{\tilde{m}}$. Let

$$j^* := \min\{\tilde{\jmath} \,|\, (\tilde{\jmath} \le j) \wedge (\tilde{u} = u) \wedge (\tilde{d} = d) \wedge (\tilde{M}_1 = M_1)\}.$$

For the j-th query (x_d, M) made by $\mathbf{A}$, $\mathbf{D}_1$ acts as follows:

– Suppose that $m = 2$. If $|M| \geq 2w$ and $|M| \equiv_w 0$, then $\mathbf{D}_1$ asks $(\mathsf{x}_{c_{1,0}}, M_2)$ to $\mathcal{G}_{j^*}\langle K'_{j^*} \oplus c_0\rangle$ and transfers the reply $\mathcal{G}_{j^*}(K'_{j^*} \oplus c_0 \oplus c_{1,0}, M_2)$ to $\mathbf{A}$. Otherwise, $\mathbf{D}_1$ asks $(\mathsf{x}_{c_{1,1}}, M_2)$ to $\mathcal{G}_{j^*}\langle K'_{j^*} \oplus c_0\rangle$ and transfers the reply $\mathcal{G}_{j^*}(K'_{j^*} \oplus c_0 \oplus c_{1,1}, M_2)$ to $\mathbf{A}$.
– Suppose that $m \geq 3$. If $|M| \geq 2w$ and $|M| \equiv_w 0$, then $\mathbf{D}_1$ asks (id, M_2) to $\mathcal{G}_{j^*}\langle K'_{j^*} \oplus c_0\rangle$, gets $\mathcal{G}_{j^*}(K'_{j^*} \oplus c_0, M_2)$ and returns $\mathsf{I}^{F,c_{1,0}}(\mathcal{G}_{j^*}(K'_{j^*} \oplus c_0, M_2), M_{[3,m]})$ to $\mathbf{A}$. Otherwise, $\mathbf{D}_1$ asks (id, M_2) to $\mathcal{G}_{j^*}\langle K'_{j^*} \oplus c_0\rangle$, gets $\mathcal{G}_{j^*}(K'_{j^*} \oplus c_0, M_2)$ and returns $\mathsf{I}^{F,c_{1,1}}(\mathcal{G}_{j^*}(K'_{j^*} \oplus c_0, M_2), M_{[3,m]})$ to $\mathbf{A}$.

$\mathbf{D}_1$ implements $\mathsf{Hy}_1[1]\langle K_1\rangle, \ldots, \mathsf{Hy}_p[1]\langle K_p\rangle$ as the oracles of $\mathbf{A}$ if its oracles are $F\langle K'_1 \oplus c_0\rangle, \ldots, F\langle K'_q \oplus c_0\rangle$. It implements $\mathsf{Hy}_1[2]\langle K_1\rangle, \ldots, \mathsf{Hy}_p[2]\langle K_p\rangle$ if its oracles are $\rho_1\langle K'_1 \oplus c_0\rangle, \ldots, \rho_q\langle K'_q \oplus c_0\rangle$. Thus,

$$\begin{aligned}&\left|\Pr[\mathbf{A}^{\mathsf{Hy}_1[1]\langle K_1\rangle,\ldots,\mathsf{Hy}_p[1]\langle K_p\rangle} = 1] - \Pr[\mathbf{A}^{\mathsf{Hy}_1[2]\langle K_1\rangle,\ldots,\mathsf{Hy}_p[2]\langle K_p\rangle} = 1]\right| \\ &\quad = \mathrm{Adv}^{q\text{-prf-rka}}_{F,\{0^n,c_{1,0},c_{1,1}\}}(\mathbf{D}_1). \end{aligned} \tag{15}$$

$\mathbf{D}_1$ takes at most about $t + O(q\ell_{\mathrm{m}}\tau_F)$ time and makes at most q queries.

For $2 \leq k \leq \ell_{\mathrm{m}} - 1$, let $\mathbf{D}_k$ be an adversary against F. $\mathbf{D}_k$ has oracles $\mathcal{G}_1\langle K'_1\rangle, \ldots, \mathcal{G}_q\langle K'_q\rangle$, which are either $F\langle K'_1\rangle, \ldots, F\langle K'_q\rangle$ or $\rho_1\langle K'_1\rangle, \ldots, \rho_q\langle K'_q\rangle$, where $K'_i \xleftarrow{} \Sigma^n$ and $\rho_i : \Sigma^n \times \Sigma^w \to \Sigma^n$ is a random oracle for $1 \leq i \leq q$. $\mathbf{D}_k$ runs $\mathbf{A}$ and simulates the oracles of $\mathbf{A}$ using its oracles. $\mathbf{D}_k$ outputs the output of $\mathbf{A}$. For the j-th query (x_d, M) made by $\mathbf{A}$ to the u-th oracle such that $\mathtt{pad2}(M) = M_1\| \cdots \|M_m$, $\mathbf{D}_k$ acts as follows:

– Suppose that $m \leq k$. Then, $\mathbf{D}_k$ simulates μ_u and returns $\mu_u(K_u \oplus d, M)$ to $\mathbf{A}$.
– Suppose that $m \geq k+1$. For $\tilde{\jmath} \leq j$, let $(\mathsf{x}_{\tilde{d}}, \tilde{M})$ be the $\tilde{\jmath}$-th query made by $\mathbf{A}$ to the $\tilde{u}$-th oracle such that $\mathtt{pad2}(\tilde{M}) = \tilde{M}_1\| \cdots \|\tilde{M}_{\tilde{m}}$, where $(\mathsf{x}_{\tilde{d}}, \tilde{M}) = (\mathsf{x}_d, M)$ and $\tilde{u} = u$ if $\tilde{\jmath} = j$. Let

$$j^* \coloneqq \min\{\tilde{\jmath} \,|\, (\tilde{\jmath} \leq j) \wedge (\tilde{u} = u) \wedge (\tilde{d} = d) \wedge (\tilde{m} \geq k+1) \wedge (\tilde{M}_{[1,k]} = M_{[1,k]})\}.$$

  - Suppose that $m = k+1$. If $|M| \equiv_w 0$, then $\mathbf{D}_k$ asks $(\mathsf{x}_{c_{1,0}}, M_{k+1})$ to $\mathcal{G}_{j^*}\langle K'_{j^*}\rangle$ and transfers the reply $\mathcal{G}_{j^*}(K'_{j^*} \oplus c_{1,0}, M_{k+1})$ to $\mathbf{A}$. Otherwise, it asks $(\mathsf{x}_{c_{1,1}}, M_{k+1})$ to $\mathcal{G}_{j^*}\langle K'_{j^*}\rangle$ and transfers the reply $\mathcal{G}_{j^*}(K'_{j^*} \oplus c_{1,1}, M_{k+1})$ to $\mathbf{A}$.
  - Suppose that $m \geq k+2$. If $|M| \equiv_w 0$, then $\mathbf{D}_k$ asks (id, M_{k+1}) to $\mathcal{G}_{j^*}\langle K'_{j^*}\rangle$, gets $\mathcal{G}_{j^*}(K'_{j^*}, M_{k+1})$, and returns $\mathsf{I}^{F,c_{1,0}}(\mathcal{G}_{j^*}(K'_{j^*}, M_{k+1}), M_{[k+2,m]})$ to $\mathbf{A}$. Otherwise, it asks (id, M_{k+1}) to $\mathcal{G}_{j^*}\langle K'_{j^*}\rangle$, gets $\mathcal{G}_{j^*}(K'_{j^*}, M_{k+1})$, and returns $\mathsf{I}^{F,c_{1,1}}(\mathcal{G}_{j^*}(K'_{j^*}, M_{k+1}), M_{[k+2,m]})$ to $\mathbf{A}$.

$\mathbf{D}_k$ implements $\mathsf{Hy}_1[k]\langle K_1\rangle, \ldots, \mathsf{Hy}_p[k]\langle K_p\rangle$ as the oracles of $\mathbf{A}$ if its oracles are $F\langle K'_1\rangle, \ldots, F\langle K'_q\rangle$. It implements $\mathsf{Hy}_1[k+1]\langle K_1\rangle, \ldots, \mathsf{Hy}_p[k+1]\langle K_p\rangle$ if its oracles are $\rho_1\langle K'_1\rangle, \ldots, \rho_q\langle K'_q\rangle$. Thus,

$$\begin{aligned}&\left|\Pr[\mathbf{A}^{\mathsf{Hy}_1[k]\langle K_1\rangle,\ldots,\mathsf{Hy}_p[k]\langle K_p\rangle} = 1] - \Pr[\mathbf{A}^{\mathsf{Hy}_1[k+1]\langle K_1\rangle,\ldots,\mathsf{Hy}_p[k+1]\langle K_p\rangle} = 1]\right| \\ &\quad = \mathrm{Adv}^{q\text{-prf-rka}}_{F,\{0^n,c_{1,0},c_{1,1}\}}(\mathbf{D}_k). \end{aligned} \tag{16}$$

$\mathbf{D}_k$ takes at most about $t + O(q\,\ell_{\mathrm{m}}\,\tau_F)$ time and makes at most q queries.

From Inequality (13) and Equalities (14), (15), and (16), there exist adversaries $\dot{\mathbf{A}}$ and $\ddot{\mathbf{A}}$ such that

$$\mathrm{Adv}^{p\text{-prf-rka}}_{\mathsf{KHC2}^F,\Delta}(\mathbf{A}) \leq \mathrm{Adv}^{p\text{-prf-rka}}_{F,\Delta}(\dot{\mathbf{A}}) + (\ell_{\mathrm{m}} - 1) \cdot \mathrm{Adv}^{q\text{-prf-rka}}_{F,\{0^n,c_{1,0},c_{1,1}\}}(\ddot{\mathbf{A}}).$$

$\dot{\mathbf{A}}$ and $\ddot{\mathbf{A}}$ take at most about $t + O(q\,\ell_{\mathrm{m}}\,\tau_F)$ time and make at most q queries.

References

1. Bagheri, N., Gauravaram, P., Knudsen, L.R., Zenner, E.: The suffix-free-prefix-free hash function construction and its indifferentiability security analysis. International Journal of Information Security **11**(6), 419–434 (2012). https://doi.org/10.1007/s10207-012-0175-4 3
2. Bellare, M., Bernstein, D.J., Tessaro, S.: Hash-function based PRFs: AMAC and its multi-user security. In: Fischlin, M., Coron, J. (eds.) Advances in Cryptology - EUROCRYPT 2016 - 35th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Vienna, Austria, May 8-12, 2016, Proceedings, Part I. Lecture Notes in Computer Science, vol. 9665, pp. 566–595. Springer (2016). https://doi.org/10.1007/978-3-662-49890-3_22 4
3. Bellare, M., Canetti, R., Krawczyk, H.: Keying hash functions for message authentication. In: Koblitz, N. (ed.) Advances in Cryptology - CRYPTO '96, 16th Annual International Cryptology Conference, Santa Barbara, California, USA, August 18-22, 1996, Proceedings. Lecture Notes in Computer Science, vol. 1109, pp. 1–15. Springer (1996). https://doi.org/10.1007/3-540-68697-5_1 2, 4
4. Bellare, M., Canetti, R., Krawczyk, H.: Pseudorandom functions revisited: The cascade construction and its concrete security. In: Proceedings of the 37th IEEE Symposium on Foundations of Computer Science. pp. 514–523 (1996). https://doi.org/10.1109/SFCS.1996.548510 3, 18
5. Bellare, M., Kohno, T.: A theoretical treatment of related-key attacks: RKA-PRPs, RKA-PRFs, and applications. In: Biham, E. (ed.) Advances in Cryptology - EUROCRYPT 2003, International Conference on the Theory and Applications of Cryptographic Techniques, Warsaw, Poland, May 4-8, 2003, Proceedings. Lecture Notes in Computer Science, vol. 2656, pp. 491–506. Springer (2003). https://doi.org/10.1007/3-540-39200-9_31 3, 5
6. Bellare, M., Ristenpart, T.: Multi-property-preserving hash domain extension and the EMD transform. In: Lai, X., Chen, K. (eds.) Advances in Cryptology - ASIACRYPT 2006, 12th International Conference on the Theory and Application of Cryptology and Information Security, Shanghai, China, December 3-7, 2006, Proceedings. Lecture Notes in Computer Science, vol. 4284, pp. 299–314. Springer (2006). https://doi.org/10.1007/11935230_20 4
7. Bernstein, D.J.: How to stretch random functions: The security of protected counter sums. Journal of Cryptology **12**(3), 185–192 (1999). https://doi.org/10.1007/s001459900051 4
8. Black, J., Rogaway, P.: A Block-Cipher Mode of Operation for Parallelizable Message Authentication. In: Knudsen, L.R. (ed.) Advances in Cryptology - EUROCRYPT 2002, International Conference on the Theory and Applications of Cryptographic Techniques, Amsterdam, The Netherlands, April 28 - May 2, 2002, Proceedings. Lecture Notes in Computer Science, vol. 2332, pp. 384–397. Springer (2002). https://doi.org/10.1007/3-540-46035-7_25 4

9. Boneh, D., Eskandarian, S., Fisch, B.: Post-quantum EPID signatures from symmetric primitives. In: Matsui, M. (ed.) Topics in Cryptology - CT-RSA 2019 - The Cryptographers' Track at the RSA Conference 2019, San Francisco, CA, USA, March 4-8, 2019, Proceedings. Lecture Notes in Computer Science, vol. 11405, pp. 251–271. Springer (2019). https://doi.org/10.1007/978-3-030-12612-4_13 2
10. Damgård, I.: A design principle for hash functions. In: Brassard, G. (ed.) Advances in Cryptology - CRYPTO '89, 9th Annual International Cryptology Conference, Santa Barbara, California, USA, August 20-24, 1989, Proceedings. Lecture Notes in Computer Science, vol. 435, pp. 416–427. Springer (1989). https://doi.org/10.1007/0-387-34805-0_39 3
11. Dodis, Y., Grubbs, P., Ristenpart, T., Woodage, J.: Fast message franking: From invisible salamanders to encryptment. In: Shacham, H., Boldyreva, A. (eds.) Advances in Cryptology - CRYPTO 2018 - 38th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 19-23, 2018, Proceedings, Part I. Lecture Notes in Computer Science, vol. 10991, pp. 155–186. Springer (2018). https://doi.org/10.1007/978-3-319-96884-1_6 2
12. FIPS PUB 198-1: The keyed-hash message authentication code (HMAC) (2008). https://doi.org/10.6028/NIST.FIPS.198-1 2
13. Furuya, I., Kasahara, H., Inoue, A., Minematsu, K., Iwata, T.: PMAC*r*x: A vector-input MAC for high-dimensional vectors with BBB security. In: Shikata, J., Kuzuno, H. (eds.) Advances in Information and Computer Security - 18th International Workshop on Security, IWSEC 2023, Yokohama, Japan, August 29-31, 2023, Proceedings. Lecture Notes in Computer Science, vol. 14128, pp. 77–97. Springer (2023). https://doi.org/10.1007/978-3-031-41326-1_5 4
14. Furuya, I., Minematsu, K.: Revisiting vector-input MACs. IACR Transactions on Symmetric Cryptology **2025**(3), 261–288 (2025). https://doi.org/10.46586/tosc.v2025.i3.261-288 4
15. Goldreich, O., Goldwasser, S., Micali, S.: How to construct random functions. Journal of the ACM **33**(4), 792–807 (1986). https://doi.org/10.1145/6490.6503 3
16. Grubbs, P., Lu, J., Ristenpart, T.: Message franking via committing authenticated encryption. In: Katz, J., Shacham, H. (eds.) Advances in Cryptology - CRYPTO 2017 - 37th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 20-24, 2017, Proceedings, Part III. Lecture Notes in Computer Science, vol. 10403, pp. 66–97. Springer (2017). https://doi.org/10.1007/978-3-319-63697-9_3 2
17. Hirose, S.: Collision-resistant and pseudorandom function based on Merkle-Damgård hash function. In: Park, J.H., Seo, S. (eds.) Information Security and Cryptology - ICISC 2021 - 24th International Conference, Seoul, South Korea, December 1-3, 2021, Revised Selected Papers. Lecture Notes in Computer Science, vol. 13218, pp. 325–338. Springer (2021). https://doi.org/10.1007/978-3-031-08896-4_17 3, 4
18. Hirose, S., Ideguchi, K., Kuwakado, H., Owada, T., Preneel, B., Yoshida, H.: A lightweight 256-bit hash function for hardware and low-end devices: Lesamnta-LW. In: Rhee, K.H., Nyang, D. (eds.) Information Security and Cryptology - ICISC 2010 - 13th International Conference, Seoul, Korea, December 1-3, 2010, Revised Selected Papers. Lecture Notes in Computer Science, vol. 6829, pp. 151–168. Springer (2010). https://doi.org/10.1007/978-3-642-24209-0_10 4
19. Hirose, S., Kuwakado, H.: Collision-resistant and pseudorandom hashing modes of compression functions. IEICE Transactions on Fundamentals **E109-A**(3), 185–193 (2026). https://doi.org/10.1587/transfun.2025CIP0005 2, 3, 6

20. Hirose, S., Kuwakado, H., Yoshida, H.: A pseudorandom-function mode based on Lesamnta-LW and the MDP domain extension and its applications. IEICE Transactions on Fundamentals **101-A**(1), 110–118 (2018). https://doi.org/10.1587/transfun.E101.A.110 4
21. Hirose, S., Park, J.H., Yun, A.: A simple variant of the Merkle-Damgård scheme with a permutation. In: Kurosawa, K. (ed.) Advances in Cryptology - ASIACRYPT 2007, 13th International Conference on the Theory and Application of Cryptology and Information Security, Kuching, Malaysia, December 2-6, 2007, Proceedings. Lecture Notes in Computer Science, vol. 4833, pp. 113–129. Springer (2007). https://doi.org/10.1007/978-3-540-76900-2_7 3, 4
22. Kelsey, J., Chang, S., Perlner, R.: SHA-3 derived functions: cSHAKE, KMAC, TupleHash and ParallelHash. NIST Special Publication 800-135 (2016). https://doi.org/10.6028/NIST.SP.800-185 2
23. Liskov, M.D., Rivest, R.L., Wagner, D.A.: Tweakable block ciphers. In: Yung, M. (ed.) Advances in Cryptology - CRYPTO 2002, 22nd Annual International Cryptology Conference, Santa Barbara, California, USA, August 18-22, 2002, Proceedings. Lecture Notes in Computer Science, vol. 2442, pp. 31–46. Springer (2002). https://doi.org/10.1007/3-540-45708-9_3 4
24. List, E., Nandi, M.: Revisiting full-PRF-secure PMAC and using it for beyond-birthday authenticated encryption. In: Handschuh, H. (ed.) Topics in Cryptology - CT-RSA 2017 - The Cryptographers' Track at the RSA Conference 2017, San Francisco, CA, USA, February 14-17, 2017, Proceedings. Lecture Notes in Computer Science, vol. 10159, pp. 258–274. Springer (2017). https://doi.org/10.1007/978-3-319-52153-4_15 4
25. Merkle, R.C.: One way hash functions and DES. In: Brassard, G. (ed.) Advances in Cryptology - CRYPTO '89, 9th Annual International Cryptology Conference, Santa Barbara, California, USA, August 20-24, 1989, Proceedings. Lecture Notes in Computer Science, vol. 435, pp. 428–446. Springer (1989). https://doi.org/10.1007/0-387-34805-0_40 3
26. Minematsu, K.: A short universal hash function from bit rotation, and applications to blockcipher modes. In: Susilo, W., Reyhanitabar, R. (eds.) Provable Security - 7th International Conference, ProvSec 2013, Melaka, Malaysia, October 23-25, 2013. Proceedings. Lecture Notes in Computer Science, vol. 8209, pp. 221–238. Springer (2013). https://doi.org/10.1007/978-3-642-41227-1_13 4
27. Rogaway, P., Shrimpton, T.: A provable-security treatment of the key-wrap problem. In: Vaudenay, S. (ed.) Advances in Cryptology - EUROCRYPT 2006, 25th Annual International Conference on the Theory and Applications of Cryptographic Techniques, St. Petersburg, Russia, May 28 - June 1, 2006, Proceedings. Lecture Notes in Computer Science, vol. 4004, pp. 373–390. Springer (2006). https://doi.org/10.1007/11761679_23 4
28. Yasuda, K.: "Sandwich" is indeed secure: How to authenticate a message with just one hashing. In: Pieprzyk, J., Ghodosi, H., Dawson, E. (eds.) Information Security and Privacy, 12th Australasian Conference, ACISP 2007, Townsville, Australia, July 2-4, 2007, Proceedings. Lecture Notes in Computer Science, vol. 4586, pp. 355–369. Springer (2007). https://doi.org/10.1007/978-3-540-73458-1_26 4
29. Yasuda, K.: HMAC without the "second" key. In: Samarati, P., Yung, M., Martinelli, F., Ardagna, C.A. (eds.) Information Security, 12th International Conference, ISC 2009, Pisa, Italy, September 7-9, 2009. Proceedings. Lecture Notes in Computer Science, vol. 5735, pp. 443–458. Springer (2009). https://doi.org/10.1007/978-3-642-04474-8_35 4