IEICE TRANS. FUNDAMENTALS, VOL.E102–A, NO.1 JANUARY 2019

PAPER

The PRF Security of Compression-Function-Based MAC Functions in the Multi-User Setting

Shoichi HIROSE[†a)], ***Member***

SUMMARY A compression-function-based MAC function called FMAC was presented as well as a vector-input PRF called vFMAC in 2016. They were proven to be secure PRFs on the assumption that their compression function is a secure PRF against related-key attacks with respect to their non-cryptographic permutations in the single user setting. In this paper, it is shown that both FMAC and vFMAC are also secure PRFs in the multi-user setting on the same assumption as in the single user setting. These results imply that their security in the multi-user setting does not degrade with the number of the users and is as good as in the single user setting.

key words: *compression function, MAC, pseudorandom function, multi-user security, vector-input PRF*

1. Introduction

(1) Background.

Message authentication is an important role of cryptography. A secret-key cryptographic primitive called a MAC function is used for message authentication. MAC stands for message authentication code, which is a short sequence called a tag computed by a MAC function from a message to be authenticated and a secret key. A typical construction of a MAC function uses a block cipher or a cryptographic hash function as its building block. This paper deals with construction of a MAC function using a cryptographic hash function.

HMAC [3] is the most famous and widely deployed MAC function constructed from a cryptographic hash function. It was originally designed to be constructed from iterated hash functions such as SHA-1, SHA-256 and SHA-512 [10]. Due to their length extension property, the construction of a MAC function from them is not straightforward. Roughly, a hash function H is said to have the length extension property if, for sequences M and M', $H(M\|M')$ can be computed from $H(M)$ and M', where $M\|M'$ represents concatenation of M and M'. Thus, if H has the length extension property, then, for a secret key K, one can compute $H(K\|M\|M')$ from $H(K\|M)$ and M' without knowing K. To avoid the problem, HMAC has the following structure:

$$H((K \oplus \mathtt{opad})\|H((K \oplus \mathtt{ipad})\|M)), \quad (1)$$

where $\oplus$ represents bitwise XOR, `ipad` and `opad` are distinct constants. Since HMAC calls H twice, it is not efficient for short messages.

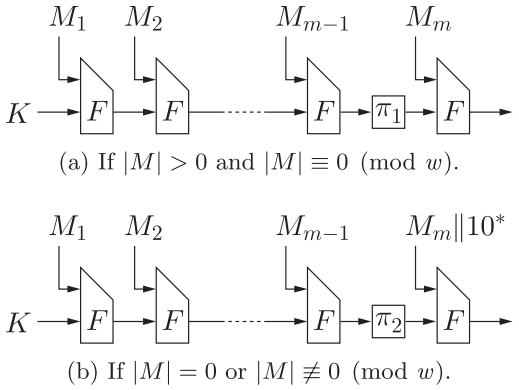


Fig. 1 FMAC. F is a compression function. $M = M_1\|M_2\|\cdots\|M_m$. $|M_i| = w$ for $1 \le i \le m-1$ and $0 \le |M_m| \le w$.

In addition to efficiency, the other matter to be considered is security in the multi-user setting. HMAC is shown to be a secure pseudorandom function (PRF) under reasonable assumptions [1], [3], [11]. As far as we know, however, the analyses are only in the single user setting, and the simple reduction [4] only guarantees the security level degrading with the number of the users.

FMAC [15] is a recently proposed simple MAC function composed with a compression function such as those of SHA-1, SHA-256 and SHA-512. It is depicted in Fig. 1. π_1 and π_2 are non-cryptographic permutations, and a candidate for them is addition of some constant. FMAC was shown to be a secure PRF if its compression function is a secure PRF against related-key attacks with respect to the permutations π_1 and π_2. A vector-input PRF, vFMAC, consisting of FMAC was also proposed and shown to be a secure PRF on the same assumption [15]. On the other hand, the PRF security of FMAC and vFMAC was not discussed in the multi-user setting.

(2) Contribution.

This paper shows that the PRF security of FMAC and vFMAC in the multi-user setting is essentially independent of the number of the users. The PRF security of FMAC and vFMAC in the multi-user setting can be proved on the same assumption as in the single user setting. The proofs heavily use the hybrid argument [13]. In particular, vFMAC is the first vector-input PRF (vPRF) that is shown to be as secure in the multi-user setting as in the single user setting.

Actually, the proof of PRF security of FMAC or vFMAC

Manuscript received June 17, 2018.
Manuscript revised August 21, 2018.

[†]The author is with Faculty of Engineering, University of Fukui, Fukui-shi, 910-8507 Japan.
a) E-mail: hrs_shch@u-fukui.ac.jp

DOI: 10.1587/transfun.E102.A.270

in the multi-user setting is almost the same as the proof in the single user setting. It is due to an essential property of the hybrid argument used in the proof: The hybrid argument is free from the number of the instances (secret keys), and only the number of the queries matters. It was first revealed by the proof of PRF security of AMAC in the multi-user setting [2].

(3) Related Work.

AMAC [2] is a MAC function using a hash function augmented with an unkeyed output function such as truncation and the mod function. AMAC is more efficient than HMAC for short messages as well as FMAC. AMAC is shown to be a secure PRF if its compression function is a secure PRF under leakage of the key by the output function. It is also shown to have as good PRF security in the multi-user setting as in the single user setting.

Suppose that AMAC and FMAC are instantiated with (a compression function of) an iterated hash function such as SHA-1, SHA-256 and SHA-512. The PRF security of AMAC requires its compression function to be a secure PRF with two keying strategies, that is, keyed both via initialization vector (IV) and via message. The PRF security of FMAC requires its compression function to be a secure PRF keyed only via IV. FMAC is slightly more efficient than AMAC since AMAC takes a secret key as a part of message input and involves Merkle-Damgård strengthening, while FMAC does not. On the other hand, AMAC is easier to be implemented since AMAC can be implemented with a hash function and FMAC with a compression function. The difference may be smaller than before in some situations, however, since Intel SHA extensions are now available.

FMAC is based on MDP [14], which was proposed as a multi-property preserving domain extension. The notion of multi-property preservation was introduced by Bellare and Ristenpart [7] together with the first multi-property preserving domain extension EMD.

HMAC is shown to be a secure PRF if its compression function is a secure PRF keyed both via IV and via message [1]. In addition, the compression function keyed via IV is required to be a secure PRF against related-key attacks with respect to `ipad` and `opad`.

A variant of HMAC called H^2-MAC was presented by Yasuda [23]. It is shown to be a secure PRF on the assumption that its compression function remains a secure PRF even if a piece of information on the secret key is disclosed.

Bellare et al. [4] showed that the plain Merkle-Damgård iteration keyed via IV is a secure PRF against attacks making prefix-free queries if its compression function is a secure PRF. They also introduced the notion of multi-user security.

Security of some other symmetric-key schemes are also analyzed in the multi-user setting: block cipher [16], [17] and authenticated encryption [9], [18].

Rogaway and Shrimpton introduced the notion of vPRF [22]. They also presented generic construction of a vPRF from a usual string-input PRF in the same paper. Minematsu presented a vPRF using his universal hash function based on bit rotation [20].

The CBC-MAC variants GCBC1 and GCBC2 [21] finalize their iteration with multiple non-cryptographic transformations for domain separation.

LightMAC [19] is a new MAC mode of operation for lightweight block ciphers, which has a similar structure to vFMAC.

(4) Organization.

Section 2 gives notations and definitions for the remaining parts of the paper. It is shown in Sect. 3 that the MDP domain extension produces multiple independent secure PRFs with multiple secret keys and permutations. Based on this result, the PRF security of FMAC and vFMAC in the multi-user setting is analyzed in Sect. 4 and in Sect. 5, respectively. Section 6 gives a brief concluding remark.

2. Preliminaries

2.1 Notations and Definitions

For integers i_1 and i_2 such that $i_1 \le i_2$, let $[i_1, i_2]$ represent the set of integers between i_1 and i_2 inclusive.

Let $\Sigma \triangleq \{0,1\}$. For a non-negative integer l, let Σ^l represent the set of all Σ-sequences of length l. Let ε be the Σ-sequence of length 0. For $l \ge 1$, let $(\Sigma^l)^* \triangleq \bigcup_{i\ge 0} \Sigma^{li}$ and $(\Sigma^l)^+ \triangleq (\Sigma^l)^* \setminus \{\varepsilon\}$. For $k_1 \le k_2$, let $(\Sigma^l)^{[k_1,k_2]} \triangleq \bigcup_{i=k_1}^{k_2} \Sigma^{li}$.

For $x \in \Sigma^*$, let $|x|$ be the length of x. For $x, y \in \Sigma^*$, let $x\|y$ be the concatenation of x and y.

Let $s \leftsquigarrow S$ represent that an element s is taken from a set S in uniform distribution.

Let $f : \mathcal{K} \times \mathcal{D} \to \mathcal{R}$ represent a keyed function from $\mathcal{D}$ to $\mathcal{R}$ with its key space $\mathcal{K}$. $f(K, \cdot)$ is often denoted by $f_K(\cdot)$.

Let $\mathcal{F}_{\mathcal{D},\mathcal{R}}$ or $\mathcal{F}(\mathcal{D},\mathcal{R})$ be the set of all functions from $\mathcal{D}$ to $\mathcal{R}$. Let $\mathcal{P}_{\mathcal{D}}$ be the set of all permutations on $\mathcal{D}$. Let id represent an identity permutation.

Security requirements of cryptographic primitives or schemes are usually formalized by their insecurity, that is, advantage of adversaries against them. An adversary is given one or more oracles. It makes queries to each of them and obtains the answers. Without loss of generality, it is assumed that all the queries made by the adversary to each oracle are distinct from each other.

2.2 Pseudorandom Functions

A pseudorandom function (PRF) [12] is a keyed function $f : \mathcal{K} \times \mathcal{D} \to \mathcal{R}$. The security requirement of a PRF is defined as follows [5], [8], [12]. An adversary $\mathbf{A}$ against f is given oracle access to f_K or ρ, where $K \leftsquigarrow \mathcal{K}$ and $\rho \leftsquigarrow \mathcal{F}_{\mathcal{D},\mathcal{R}}$, and makes adaptive queries in $\mathcal{D}$ and obtains the corresponding outputs. The prf-advantage of $\mathbf{A}$ against f is defined as

$$\mathrm{Adv}_f^{\mathrm{prf}}(\mathbf{A}) \triangleq \left|\Pr[\mathbf{A}^{f_K} = 1] - \Pr[\mathbf{A}^{\rho} = 1]\right|, \quad (2)$$

where $\mathbf{A}$ is regarded as a random variable.

Informally, f is called a secure PRF if any adversary with realistic computational resources can have only negligible prf-advantage against f.

The definition of the prf-advantage given above is said to be in the single user setting. The prf-advantage in the multi-user setting is defined with adversaries given multiple oracles as follows [4]:

$$\mathrm{Adv}_f^{m\text{-prf}}(\mathbf{A}) \triangleq \left|\Pr[\mathbf{A}^{F_{K_1},\ldots,F_{K_m}} = 1] - \Pr[\mathbf{A}^{\rho_1,\ldots,\rho_m} = 1]\right|, \quad (3)$$

where $K_i \xleftarrow{\$} \mathcal{K}$ and $\rho_i \xleftarrow{\$} \mathcal{F}_{\mathcal{D},\mathcal{R}}$ for every $i \in [1, m]$.

The following proposition relates the PRF security in the multi-user setting to the PRF security in the single user setting.

Proposition 1 (Lemma 3.3 in [4]) For any adversary $\mathbf{A}_{\mathrm{m}}$ against f with access to m oracles, there exists some adversary $\mathbf{A}_{\mathrm{s}}$ against f such that

$$\mathrm{Adv}_f^{m\text{-prf}}(\mathbf{A}_{\mathrm{m}}) \le m \cdot \mathrm{Adv}_f^{\mathrm{prf}}(\mathbf{A}_{\mathrm{s}}). \quad (4)$$

The run time of $\mathbf{A}_{\mathrm{s}}$ is approximately total of that of $\mathbf{A}_{\mathrm{m}}$ and the time to compute f for the queries made by $\mathbf{A}_{\mathrm{m}}$. The number of the queries made by $\mathbf{A}_{\mathrm{s}}$ is at most $\max\{q_i \mid i \in [1, m]\}$, where q_i is the number of the queries from $\mathbf{A}_{\mathrm{m}}$ to its i-th oracle.

Remark 1 In this paper, the PRF security in the multi-user setting is formalized with the multi-oracle families [4]. In this formalization, the number of the instances is fixed as a parameter m. In the formalization of [2], on the other hand, this is not the case: Adversaries are allowed to ask an oracle to create a new instance as it wishes. The two kinds of formalization are essentially the same in spite of their different appearances. This paper adopts the more classical and simpler formalization.

2.3 PRFs Under Related-Key Attacks

A PRF under related-key attacks is formalized by Bellare and Kohno [6]. For $\Phi \subset \mathcal{F}(\mathcal{K}, \mathcal{K})$, let $\mathsf{key} \in \mathcal{F}(\Phi \times \mathcal{K}, \mathcal{K})$ be a function such that $\mathsf{key}(\varphi, K) = \varphi(K)$. Let $\mathbf{A}$ be an adversary against $f \in \mathcal{F}(\mathcal{K} \times \mathcal{D}, \mathcal{R})$. $\mathbf{A}$ has oracle access to $g(\mathsf{key}(\cdot, K), \cdot)$, where g is either f or $\rho \xleftarrow{\$} \mathcal{F}(\mathcal{K} \times \mathcal{D}, \mathcal{R})$, and $K \xleftarrow{\$} \mathcal{K}$. $\mathbf{A}$ asks $(\varphi, x) \in \Phi \times \mathcal{D}$ as a query and gets $g(\varphi(K), x)$. Just for simplicity, $g[K] \triangleq g(\mathsf{key}(\cdot, K), \cdot)$. The prf-rka-advantage of $\mathbf{A}$ making a Φ-related-key attack (Φ-RKA) against f is given by

$$\mathrm{Adv}_{\Phi,f}^{\text{prf-rka}}(\mathbf{A}) \triangleq \left|\Pr[\mathbf{A}^{f[K]} = 1] - \Pr[\mathbf{A}^{\rho[K]} = 1]\right|. \quad (5)$$

The prf-rka-advantage of $\mathbf{A}$ making a Φ-RKA in the multi-user setting is defined as

$$\mathrm{Adv}_{\Phi,f}^{m\text{-prf-rka}}(\mathbf{A}) \triangleq |\Pr[\mathbf{A}^{f[K_1],\ldots,f[K_m]} = 1] - \Pr[\mathbf{A}^{\rho_1[K_1],\ldots,\rho_m[K_m]} = 1]|, \quad (6)$$

where $K_i \xleftarrow{\$} \mathcal{K}$ and $\rho_i \xleftarrow{\$} \mathcal{F}(\mathcal{K} \times \mathcal{D}, \mathcal{R})$ for every $i \in [1, m]$.

3. A PRF Based on MDP

3.1 Definition

The MDP domain extension [14] is a variant of the plain Merkle-Damgård domain extension. To simplify the notation, let $\mathcal{C} \triangleq \Sigma^n$ and $\mathcal{B} \triangleq \Sigma^w$. Let $F : \mathcal{C} \times \mathcal{B} \to \mathcal{C}$ be a compression function. A keyed function based on MDP, $\mathsf{J}^F : \mathcal{C} \times \mathcal{P}_{\mathcal{C}} \times \mathcal{B}^+ \to \mathcal{C}$ with its key space $\mathcal{C}$, is defined as follows: For $X_1, X_2, \ldots, X_x \in \mathcal{B}$ with $x \ge 1$,

$$\mathsf{J}^F(K, \pi, X_1 \| X_2 \| \cdots \| X_x) = Y_x, \quad (7)$$

where $Y_0 \leftarrow K$ and

$$Y_i \leftarrow \begin{cases} F(Y_{i-1}, X_i) & \text{if } 1 \le i \le x-1, \\ F(\pi(Y_{x-1}), X_x) & \text{if } i = x. \end{cases} \quad (8)$$

J^F is depicted in Fig. 2.

J^F makes it unnecessary to introduce the prfs-advantage in [15]. It is advantage of an adversary in distinguishing multiple keyed functions sharing a single key from multiple random functions, which is different from the m-prf-advantage.

3.2 Security Analysis

Let $\Pi \subset \mathcal{P}_{\mathcal{C}} \setminus \{id\}$. Let

$$p_{\Pi} \triangleq \Pr\left[\begin{array}{l}\text{There exist some distinct } \pi, \pi' \text{ in} \\ \Pi \cup \{id\} \text{ such that } \pi(X) = \pi'(X)\end{array}\right], \quad (9)$$

where X is a random variable with uniform distribution over $\mathcal{C}$.

The following theorem says that J^F is a secure PRF against adversaries making queries only on the permutations in Π in the multi-user setting if F is a secure PRF against $(\Pi \cup \{id\})$-related-key attacks in the single user setting.

Theorem 1 Let $\mathbf{A}$ be any adversary against J^F. Suppose that $\mathbf{A}$ runs in time at most t and makes at most q queries in $\Pi \times \mathcal{B}^{[1,\ell]}$ in total. Then, there exists some adversary $\mathbf{B}$ against F such that

$$\mathrm{Adv}_{\mathsf{J}^F}^{m\text{-prf}}(\mathbf{A}) \le \ell q \left(\mathrm{Adv}_{\Pi \cup \{id\},F}^{\text{prf-rka}}(\mathbf{B}) + p_{\Pi}\right). \quad (10)$$

$\mathbf{B}$ runs in time at most $t + O(\ell q T_F)$ and makes at most q

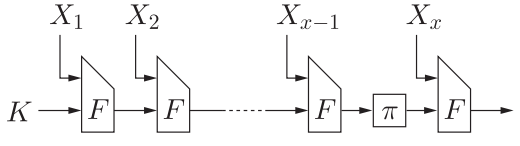


Fig. 2 $\mathsf{J}^F(K, \pi, X_1 \| X_2 \| \cdots \| X_x)$.

queries, where T_F is the time required to compute F.

The upper bound of the prf-advantage of $\mathbf{A}$ against J^F presented by Theorem 1 is essentially independent of the number of the users m. Actually, in the proof of Theorem 1, $\mathbf{B}$ is constructed with $\mathbf{A}$ as a subroutine and $\mathbf{B}$ should realize, for each query made by $\mathbf{A}$, which one of the m oracles of $\mathbf{A}$ receives it. In this sense, the prf-rka-advantage of $\mathbf{B}$ depends on m. However, it is not significant.

Theorem 1 is meaningful if the probability p_{Π} is sufficiently small, which is not a problem from the following remark.

Remark 2 ([15]) Let $c_1, c_2, \ldots, c_d$ be distinct nonzero constants in C.

- If $\Pi = \{\pi_j \mid \pi_j(x) = x \oplus c_j \text{ for every } j \in [1,d]\}$, then $p_{\Pi} = 0$.
- If $\Pi = \{\pi_j \mid \pi_j(x) = c_j \cdot x \text{ and } c_j \neq 1 \text{ for every } j \in [1,d]\}$, then $p_{\Pi} = 1/2^n$.

Theorem 1 follows from Lemma 1 and Lemma 2 presented in the remaining part. From Remark 2, it is assumed that the permutations in Π are much easier to be computed than F in the evaluation of time complexity of adversaries in Lemma 1 and Lemma 2.

Lemma 1 Let $\mathbf{A}$ be any adversary against J^F. Suppose that $\mathbf{A}$ runs in time at most t and makes at most q queries in $\Pi \times \mathcal{B}^{[1,\ell]}$ in total. Then, there exists some adversary $\mathbf{B}$ against F such that

$$\mathrm{Adv}^{m\text{-prf}}_{\mathsf{J}^F}(\mathbf{A}) \leq \ell \left(\mathrm{Adv}^{q\text{-prf-rka}}_{\Pi\cup\{id\},F}(\mathbf{B}) + qp_{\Pi}\right). \tag{11}$$

$\mathbf{B}$ runs in time at most $t + O(\ell q T_F)$ and makes at most q queries in total.

Proof For an integer $k \geq 0$ and two functions $\mu : \mathcal{P}_C \times \mathcal{B}^+ \to C$ and $\xi : \mathcal{B}^* \to C$, let $\mathsf{H}[k]^{\mu,\xi} : \mathcal{P}_C \times \mathcal{B}^+ \to C$ such that, for $X = X_1 \| X_2 \| \cdots \| X_l$ with $|X_i| = w$ for every $i \in [1,l]$,

$$\mathsf{H}[k]^{\mu,\xi}(\pi, X) \triangleq \begin{cases} \mu(\pi, X) & \text{if } l \leq k, \\ \mathsf{J}^F(\xi(X_{[1,k]}), \pi, X_{[k+1,l]}) & \text{otherwise,} \end{cases} \tag{12}$$

where $X_{[i_1,i_2]} \triangleq X_{i_1} \| X_{i_1+1} \| \cdots \| X_{i_2}$, $X_{[i_1,i_2]} = X_{i_1}$ if $i_1 = i_2$ and $X_{[i_1,i_2]} = \varepsilon$ if $i_1 > i_2$.

Let

$$P_k \triangleq \Pr\left[\mathbf{A}^{\mathsf{H}[k]^{\mu_1,\xi_1},\ldots,\mathsf{H}[k]^{\mu_m,\xi_m}} = 1\right], \tag{13}$$

where $\mu_i \leftsquigarrow \mathcal{F}_{\mathcal{P}_C \times \mathcal{B}^+, C}$ and $\xi_i \leftsquigarrow \mathcal{F}_{\mathcal{B}^*, C}$ for every $i \in [1,m]$. Then, the advantage of $\mathbf{A}$ is

$$\mathrm{Adv}^{m\text{-prf}}_{\mathsf{J}^F}(\mathbf{A}) = |P_0 - P_{\ell}|. \tag{14}$$

Here, notice that $\mathsf{H}[0]^{\mu_i,\xi_i}(\pi, X) = \mathsf{J}^F(\xi_i(\varepsilon), \pi, X)$ and $\mathsf{H}[\ell]^{\mu_i,\xi_i}(\pi, X) = \mu_i(\pi, X)$ for $i \in [1,m]$ since $l \in [1,\ell]$.

Let $\mathbf{B}$ be an adversary against F with q oracles, which works as follows. $\mathbf{B}$ first executes $r \leftsquigarrow [1,\ell]$. Then, $\mathbf{B}$ runs $\mathbf{A}$. Finally, $\mathbf{B}$ returns the output of $\mathbf{A}$. $\mathbf{A}$ makes at most q queries to its oracles. $\mathbf{B}$ responds to each query made by $\mathbf{A}$ to its oracle in the following way.

For $t \in [1,q]$, let (π, X) be the t-th query made by $\mathbf{A}$, where $X = X_1 \| X_2 \| \cdots \| X_l$ and $l \in [1,\ell]$. Suppose that (π, X) is given to the i^*-th oracle of $\mathbf{A}$, where $i^* \in [1,m]$. If $l \geq r$, then $\mathbf{B}$ makes a query to its $\mathtt{idx}(i^*, X_{[1,r-1]})$-th oracle, where $\mathtt{idx}(i^*, X_{[1,r-1]})$ equals the minimum $t' \in [1,t]$ such that

- the t'-th query (π', X') made by $\mathbf{A}$ is given to its i^*-th oracle, and
- $X'_{[1,r-1]} = X_{[1,r-1]}$.

The query made by $\mathbf{B}$ to its $\mathtt{idx}(i^*, X_{[1,r-1]})$-th oracle is (π, X_r) if $l = r$ and (id, X_r) if $l \geq r+1$. Notice that $\mathbf{B}$ makes a $(\Pi \cup \{id\})$-related-key attack.

Let $g_1, \ldots, g_q$ be the oracles given to $\mathbf{B}$. Then, in response to the query (π, X) made by $\mathbf{A}$, $\mathbf{B}$ returns

- $\mu_{i^*}(\pi, X)$ if $l \leq r-1$,
- $g_{\mathtt{idx}(i^*, X_{[1,r-1]})}(\pi, X_r)$ if $l = r$, and
- $\mathsf{J}^F(g_{\mathtt{idx}(i^*, X_{[1,r-1]})}(id, X_r), \pi, X_{[r+1,l]})$ if $l \geq r+1$.

$\mathbf{B}$ simulates μ_{i^*} with the lazy evaluation which selects an output uniformly at random from C for a new input.

Now, suppose that $g_i = F[K_i]$ with $K_i \leftsquigarrow C$ for every $i \in [1,q]$. Then,

$$\begin{aligned} g_{\mathtt{idx}(i^*, X_{[1,r-1]})}(\pi, X_r) &= F_{\pi(K_{\mathtt{idx}(i^*, X_{[1,r-1]})})}(X_r) && (15)\\ &= \mathsf{J}^F(K_{\mathtt{idx}(i^*, X_{[1,r-1]})}, \pi, X_r) && (16) \end{aligned}$$

and

$$\begin{aligned} &\mathsf{J}^F(g_{\mathtt{idx}(i^*, X_{[1,r-1]})}(id, X_r), \pi, X_{[r+1,l]}) \\ &\quad = \mathsf{J}^F(F_{K_{\mathtt{idx}(i^*, X_{[1,r-1]})}}(X_r), \pi, X_{[r+1,l]}) && (17)\\ &\quad = \mathsf{J}^F(K_{\mathtt{idx}(i^*, X_{[1,r-1]})}, \pi, X_{[r,l]}). && (18) \end{aligned}$$

$K_{\mathtt{idx}(i^*, X_{[1,r-1]})}$ implements $\xi_{i^*}(X_{[1,r-1]})$ since $K_i \leftsquigarrow C$ for every $i \in [1,q]$. Thus, in this case, $\mathbf{B}$ simulates $\mathsf{H}[r-1]^{\mu_1,\xi_1}, \ldots, \mathsf{H}[r-1]^{\mu_m,\xi_m}$ for $\mathbf{A}$. Thus,

$$\begin{aligned} &\Pr\left[\mathbf{B}^{F[K_1],\ldots,F[K_q]} = 1\right] \\ &= \sum_{k=1}^{\ell} \Pr\left[r = k \wedge \mathbf{B}^{F[K_1],\ldots,F[K_q]} = 1\right] && (19)\\ &= \frac{1}{\ell} \sum_{k=1}^{\ell} \Pr\left[\mathbf{B}^{F[K_1],\ldots,F[K_q]} = 1 \,\middle|\, r = k\right] && (20)\\ &= \frac{1}{\ell} \sum_{k=1}^{\ell} \Pr\left[\mathbf{A}^{\mathsf{H}[k-1]^{\mu_1,\xi_1},\ldots,\mathsf{H}[k-1]^{\mu_m,\xi_m}} = 1\right] && (21)\\ &= \frac{1}{\ell} \sum_{k=1}^{\ell} P_{k-1}. && (22) \end{aligned}$$

Suppose that $g_i = \tilde{\rho}_i$ with $\tilde{\rho}_i \leftsquigarrow \mathcal{F}_{\mathcal{P}_C \times \mathcal{B}, C}$ for every $i \in [1,q]$. Then,

$$g_{\mathtt{idx}(i^*,X_{[1,r-1]})}(\pi, X_r) = \tilde{\rho}_{\mathtt{idx}(i^*,X_{[1,r-1]})}(\pi, X_r) \quad (23)$$

and

$$\mathsf{J}^F(g_{\mathtt{idx}(i^*,X_{[1,r-1]})}(id, X_r), \pi, X_{[r+1,l]}) = \mathsf{J}^F(\tilde{\rho}_{\mathtt{idx}(i^*,X_{[1,r-1]})}(id, X_r), \pi, X_{[r+1,l]}). \quad (24)$$

$\tilde{\rho}_{\mathtt{idx}(i^*,X_{[1,r-1]})}(\pi, X_r)$ implements $\mu_{i^*}(\pi, X)$ for $l = r$, and $\tilde{\rho}_{\mathtt{idx}(i^*,X_{[1,r-1]})}(id, X_r)$ implements $\xi_{i^*}(X_{[1,r]})$. Thus, $\mathbf{B}$ simulates $\mathsf{H}[r]^{\mu_1,\xi_1}, \ldots, \mathsf{H}[r]^{\mu_m,\xi_m}$ for $\mathbf{A}$, and

$$\Pr[\mathbf{B}^{\tilde{\rho}_1,\ldots,\tilde{\rho}_q} = 1] = \frac{1}{\ell}\sum_{k=1}^{\ell} P_k. \quad (25)$$

Thus,

$$\left|\Pr[\mathbf{B}^{F[K_1],\ldots,F[K_q]} = 1] - \Pr[\mathbf{B}^{\tilde{\rho}_1,\ldots,\tilde{\rho}_q} = 1]\right| = \frac{1}{\ell}\mathrm{Adv}^{m\text{-prf}}_{\mathsf{J}^F}(\mathbf{A}). \quad (26)$$

Now, let $\rho_i \leftsquigarrow \mathcal{F}_{\mathcal{C}\times\mathcal{B},\mathcal{C}}$ for every $i \in [1, q]$. Then,

$$\begin{aligned}&\left|\Pr[\mathbf{B}^{F[K_1],\ldots,F[K_q]} = 1] - \Pr[\mathbf{B}^{\tilde{\rho}_1,\ldots,\tilde{\rho}_q} = 1]\right| \\ &\le \left|\Pr[\mathbf{B}^{F[K_1],\ldots,F[K_q]} = 1] - \Pr[\mathbf{B}^{\rho_1[K_1],\ldots,\rho_q[K_q]} = 1]\right| \\ &+ \left|\Pr[\mathbf{B}^{\rho_1[K_1],\ldots,\rho_q[K_q]} = 1] - \Pr[\mathbf{B}^{\tilde{\rho}_1,\ldots,\tilde{\rho}_q} = 1]\right| \quad (27) \\ &= \mathrm{Adv}^{q\text{-prf-rka}}_{\Pi\cup\{id\},F}(\mathbf{B}) + \\ &\quad \left|\Pr[\mathbf{B}^{\rho_1[K_1],\ldots,\rho_q[K_q]} = 1] - \Pr[\mathbf{B}^{\tilde{\rho}_1,\ldots,\tilde{\rho}_q} = 1]\right|. \quad (28)\end{aligned}$$

$\rho_i[K_i]$ and $\tilde{\rho}_i$ are identical to each other as long as $\pi(K_i) \neq \pi'(K_i)$ for any distinct $\pi, \pi' \in \Pi \cup \{id\}$. Thus,

$$\left|\Pr[\mathbf{B}^{\rho_1[K_1],\ldots,\rho_q[K_q]} = 1] - \Pr[\mathbf{B}^{\tilde{\rho}_1,\ldots,\tilde{\rho}_q} = 1]\right| \le qp_{\Pi}. \quad (29)$$

To answer to the queries made by $\mathbf{A}$, $\mathbf{B}$ may compute J^F or simulate μ_i's. It approximately costs at most ℓq evaluations of F. □

For Lemma 1, the single user setting is simply the case where $m = 1$. In the proof of Lemma 1, if $m = 1$, then $\mathbf{A}$ is given a single oracle $\mathsf{H}[k]^{\mu_1,\xi_1}$ and i^* always equals 1.

Lemma 2 relates the PRF security of F against related-key attacks in the multi-user setting with that in the single user setting. It can be proved in the same way as Proposition 1.

Lemma 2 ([15]) Let $\mathbf{A}$ be any adversary with m oracles against F running in time at most t, and making at most q queries. Then, there exists an adversary $\mathbf{B}$ against F such that

$$\mathrm{Adv}^{m\text{-prf-rka}}_{\Pi\cup\{id\},F}(\mathbf{A}) \le m \cdot \mathrm{Adv}^{\text{prf-rka}}_{\Pi\cup\{id\},F}(\mathbf{B}). \quad (30)$$

$\mathbf{B}$ runs in time at most $t + O(qT_F)$ and makes at most q queries, where T_F represents the time required to compute F.

4. FMAC in the Multi-User Setting

FMAC [15] is a MAC function defined with a compression function $F : \mathcal{C} \times \mathcal{B} \to \mathcal{C}$ and distinct permutations $\pi_1, \pi_2 \in \mathcal{P}_{\mathcal{C}} \setminus \{id\}$.

The padding function of FMAC is defined as follows: For any $M \in \Sigma^*$,

$$\mathsf{pad}(M) \triangleq \begin{cases} M & \text{if } |M| > 0 \text{ and } |M| \equiv 0 \pmod w, \\ M\|10^l & \text{if } |M| = 0 \text{ or } |M| \not\equiv 0 \pmod w, \end{cases} \quad (31)$$

where l is the minimum non-negative integer such that $|M| + 1 + l \equiv 0 \pmod w$.

FMAC is defined by $\mathsf{C}^{F,\{\pi_1,\pi_2\}} : \mathcal{C} \times \Sigma^* \to \mathcal{C}$ such that $\mathsf{C}^{F,\{\pi_1,\pi_2\}}(K, M) \triangleq \mathsf{J}^F(K, \pi, \mathsf{pad}(M))$, where

$$\pi = \begin{cases} \pi_1 & \text{if } |M| > 0 \text{ and } |M| \equiv 0 \pmod w, \\ \pi_2 & \text{if } |M| = 0 \text{ or } |M| \not\equiv 0 \pmod w. \end{cases} \quad (32)$$

The theorem shown below says that $\mathsf{C}^{F,\{\pi_1,\pi_2\}}$ is a secure PRF in the multi-user setting if F is a secure PRF against $\{id, \pi_1, \pi_2\}$-related-key attacks in the single user setting and $p_{\{\pi_1,\pi_2\}}$ is negligibly small. The upper bound of the prf-advantage of adversaries against FMAC is essentially independent of the number of the users.

Theorem 2 For any adversary $\mathbf{A}$ against $\mathsf{C}^{F,\{\pi_1,\pi_2\}}$ running in time at most t and making at most q queries in $\Sigma^{[0,\ell w]}$, there exists some adversary $\mathbf{B}$ against F such that

$$\mathrm{Adv}^{m\text{-prf}}_{\mathsf{C}^{F,\{\pi_1,\pi_2\}}}(\mathbf{A}) \le \ell q\left(\mathrm{Adv}^{\text{prf-rka}}_{\{id,\pi_1,\pi_2\},F}(\mathbf{B}) + p_{\{\pi_1,\pi_2\}}\right). \quad (33)$$

$\mathbf{B}$ runs in time at most $t + O(\ell q T_F)$ and makes at most q queries. T_F is the time required to compute F.

Theorem 2 is led from the simple lemma given below and Theorem 1.

Lemma 3 For any adversary $\mathbf{A}$ against $\mathsf{C}^{F,\{\pi_1,\pi_2\}}$ running in time at most t and making at most q queries in $\Sigma^{[0,\ell w]}$, there exists some adversary $\hat{\mathbf{A}}$ against J^F such that

$$\mathrm{Adv}^{m\text{-prf}}_{\mathsf{C}^{F,\{\pi_1,\pi_2\}}}(\mathbf{A}) = \mathrm{Adv}^{m\text{-prf}}_{\mathsf{J}^F}(\hat{\mathbf{A}}). \quad (34)$$

$\hat{\mathbf{A}}$ runs in time at most t and makes at most q queries in $\{\pi_1, \pi_2\} \times \mathcal{B}^{[1,\ell]}$ in total.

Proof $\hat{\mathbf{A}}$ has m oracles $g_1, \ldots, g_m$, which are either $\mathsf{J}^F_{K'_1}, \ldots, \mathsf{J}^F_{K'_m}$ or $\rho'_1, \ldots, \rho'_m$, where $K'_i \leftsquigarrow \mathcal{C}$ and $\rho'_i \leftsquigarrow \mathcal{F}(\mathcal{P}_{\mathcal{C}} \times \mathcal{B}^+, \mathcal{C})$ for every $i \in [1, m]$.

$\hat{\mathbf{A}}$ runs $\mathbf{A}$. For a query M made by $\mathbf{A}$ to its i^*-th oracle, $\hat{\mathbf{A}}$ makes the following query to its i^*-th oracle: $(\pi_1, \mathsf{pad}(M))$ if $|M| > 0$ and $|M| \equiv 0 \pmod w$ and $(\pi_2, \mathsf{pad}(M))$ otherwise.

$\hat{\mathbf{A}}$ transfers the reply from the oracle to $\mathbf{A}$. Finally, $\hat{\mathbf{A}}$ returns the output of $\mathbf{A}$.

Notice that

$$\Pr[\mathbf{A}^{\mathsf{C}^{F,\{\pi_1,\pi_2\}}_{K_1},\ldots,\mathsf{C}^{F,\{\pi_1,\pi_2\}}_{K_m}} = 1] = \Pr[\hat{\mathbf{A}}^{\mathsf{J}^F_{K'_1},\ldots,\mathsf{J}^F_{K'_m}} = 1], \tag{35}$$

where $K_i \leftsquigarrow C$ for every $i \in [1, m]$, and

$$\Pr[\mathbf{A}^{\rho_1,\ldots,\rho_m} = 1] = \Pr[\hat{\mathbf{A}}^{\rho'_1,\ldots,\rho'_m} = 1], \tag{36}$$

where $\rho_i \leftsquigarrow \mathcal{F}(\Sigma^*, C)$ for every $i \in [1, m]$. □

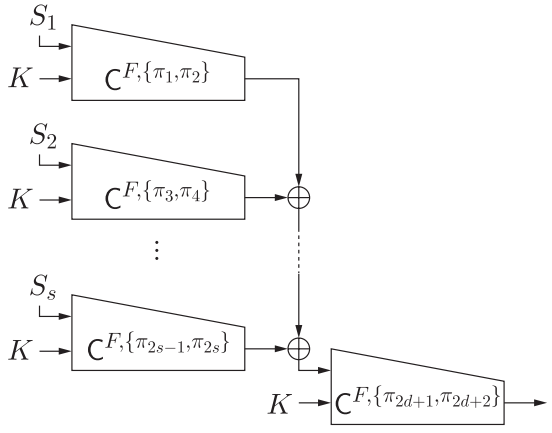
5. vFMAC in the Multi-User Setting

vFMAC [15] is a vector-input PRF (vPRF) using FMAC. Let $F : C \times \mathcal{B} \to C$. For a positive integer d, let $\Pi = \{\pi_1, \pi_2, \ldots, \pi_{2d+2}\} \subset \mathcal{P}_C \setminus \{id\}$. vFMAC is defined by $\mathsf{V}^{F,\Pi} : C \times (\Sigma^*)^{[0,d]} \to C$ such that, for an s-component vector $\boldsymbol{S} = (S_1, S_2, \ldots, S_s)$ with $s \in [0, d]$,

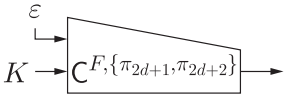
$$\mathsf{V}^{F,\Pi}(K, \boldsymbol{S}) \triangleq \begin{cases} \mathsf{C}^{F,\{\pi_{2d+1},\pi_{2d+2}\}}_K(\varepsilon) & \text{if } s = 0, \\ \mathsf{C}^{F,\{\pi_{2d+1},\pi_{2d+2}\}}_K\left(\bigoplus_{i=1}^{s} \mathsf{C}^{F,\{\pi_{2i-1},\pi_{2i}\}}_K(S_i)\right) & \text{if } s \geq 1, \end{cases} \tag{37}$$

which is also depicted in Fig. 3. vFMAC accepts vectors with at most d components as inputs, while a vPRF accepts vectors with any number of components as inputs in the original formalization [22].

The following theorem says that $\mathsf{V}^{F,\Pi}$ is a secure PRF in the multi-user setting if F is a secure PRF against $(\Pi \cup \{id\})$-related-key attacks in the single user setting and p_Π is negligible. The upper bound of the prf-advantage of adversaries against vFMAC is also essentially independent of the number of the users.



(a) $s \in [1, d]$.



(b) $s = 0$.

Fig. 3 vFMAC $\mathsf{V}^{F,\Pi}(K, \boldsymbol{S})$ for $\boldsymbol{S} = (S_1, S_2, \ldots, S_s)$, where $s \in [0, d]$.

Theorem 3 Let $\mathbf{A}$ be any adversary against $\mathsf{V}^{F,\Pi}$ running in time at most t and making at most q queries. Suppose that the length of each vector component in queries is at most ℓw and that the total number of the vector components in all of the queries is at most $\sigma(\geq q - 1)$. Then, there exists some adversary $\mathbf{B}$ against F such that

$$\mathrm{Adv}^{m\text{-prf}}_{\mathsf{V}^{F,\Pi}}(\mathbf{A}) \leq \ell(\sigma + q)\left(\mathrm{Adv}^{\text{prf-rka}}_{\Pi\cup\{id\},F}(\mathbf{B}) + p_\Pi\right) + \frac{q^2}{2^{n+1}}. \tag{38}$$

$\mathbf{B}$ runs in time at most $t + O(\ell\sigma T_F)$ and makes at most $(\sigma + q)$ queries. T_F is the time required to compute F.

Theorem 3 directly follows from Lemma 4 and Theorem 1.

Lemma 4 Let $\mathbf{A}$ be any adversary against $\mathsf{V}^{F,\Pi}$ running in time at most t and making at most q queries. Suppose that the length of each vector component in queries is at most ℓw and that the total number of the vector components in all of the queries is at most σ. Then, there exists some adversary $\hat{\mathbf{A}}$ against J^F such that

$$\mathrm{Adv}^{m\text{-prf}}_{\mathsf{V}^{F,\Pi}}(\mathbf{A}) \leq \mathrm{Adv}^{m\text{-prf}}_{\mathsf{J}^F}(\hat{\mathbf{A}}) + \frac{q^2}{2^{n+1}}. \tag{39}$$

$\hat{\mathbf{A}}$ runs in time at most t and makes at most $(\sigma + q)$ queries in $\Pi \times \mathcal{B}^{[0,\ell]}$ in total.

Proof Notice that $\mathrm{Adv}^{m\text{-prf}}_{\mathsf{V}^{F,\Pi}}(\mathbf{A})$ is

$$\left|\Pr[\mathbf{A}^{\mathsf{V}^{F,\Pi}_{K_1},\ldots,\mathsf{V}^{F,\Pi}_{K_m}} = 1] - \Pr[\mathbf{A}^{\rho_1,\ldots,\rho_m} = 1]\right|, \tag{40}$$

where $K_i \leftsquigarrow C$ and $\rho_i \leftsquigarrow \mathcal{F}((\Sigma^*)^{[0,d]}, C)$ for every $i \in [1, m]$.

Let $\hat{\mathsf{V}}^{\mathsf{J}^F_{K_i}}$ be an algorithm to compute $\mathsf{V}^{F,\Pi}_{K_i}$ by using $\mathsf{J}^F_{K_i}$. Then,

$$\begin{aligned}\mathrm{Adv}^{m\text{-prf}}_{\mathsf{V}^{F,\Pi}}(\mathbf{A}) \leq\ & \left|\Pr[\mathbf{A}^{\hat{\mathsf{V}}^{\mathsf{J}^F_{K_1}},\ldots,\hat{\mathsf{V}}^{\mathsf{J}^F_{K_m}}} = 1] - \Pr[\mathbf{A}^{\hat{\mathsf{V}}^{\mu_1},\ldots,\hat{\mathsf{V}}^{\mu_m}} = 1]\right| \\ &+ \left|\Pr[\mathbf{A}^{\hat{\mathsf{V}}^{\mu_1},\ldots,\hat{\mathsf{V}}^{\mu_m}} = 1] - \Pr[\mathbf{A}^{\rho_1,\ldots,\rho_m} = 1]\right|, \end{aligned} \tag{41}$$

where $\mu_i \leftsquigarrow \mathcal{F}(\mathcal{P}_C \times \mathcal{B}^+, C)$ and $\hat{\mathsf{V}}^{\mu_i}$ is obtained from $\hat{\mathsf{V}}^{\mathsf{J}^F_{K_i}}$ simply by replacing $\mathsf{J}^F_{K_i}$ with μ_i for every $i \in [1, m]$.

For the first term of the upper bound of Eq. (41), there exists some adversary $\hat{\mathbf{A}}$ such that

$$\mathrm{Adv}^{m\text{-prf}}_{\mathsf{J}^F}(\hat{\mathbf{A}}) = \left|\Pr[\mathbf{A}^{\hat{\mathsf{V}}^{\mathsf{J}^F_{K_1}},\ldots,\hat{\mathsf{V}}^{\mathsf{J}^F_{K_m}}} = 1] - \Pr[\mathbf{A}^{\hat{\mathsf{V}}^{\mu_1},\ldots,\hat{\mathsf{V}}^{\mu_m}} = 1]\right|, \tag{42}$$

and $\hat{\mathbf{A}}$ runs in time at most t and makes at most $(\sigma + q)$

queries in $\Pi \times \mathcal{B}^{[0,\ell]}$ in total.

For the second term of the upper bound of Eq. (41), let us consider an algorithm R which works as the m oracles of $\mathbf{A}$ as follows:

1. Prior to the interaction with $\mathbf{A}$,
   - $Y_{t,j} \leftarrow \perp$ for every $t \in [1,q]$ and $j \in [1,d]$,
   - $Z_t \leftsquigarrow C$ for every $t \in [1,q]$, and
   - $bad \leftarrow 0$.
2. During the interaction with $\mathbf{A}$, return Z_t in response to the t-th query made by $\mathbf{A}$.
3. For $t \in [1,q]$, let $\boldsymbol{S}_t = (S_{t,1}, S_{t,2}, \ldots, S_{t,s_t})$ be the t-th query made by $\mathbf{A}$, where $s_t \in [0,d]$. Let $\mathtt{i}(t)$ indicate the oracle receiving the t-th query. Namely, $\mathbf{A}$ asks $\boldsymbol{S}_t$ to its $\mathtt{i}(t)$-th oracle. For every $j \in [1,s_t]$,
   - $Y_{t,j} \leftsquigarrow C$ if $S_{t,j}$ is new, that is, $S_{t,j} \neq S_{t',j}$ for any $t' < t$ such that $\mathtt{i}(t') = \mathtt{i}(t)$, and
   - $Y_{t,j} \leftarrow Y_{t',j}$ if there exists some $t' < t$ such that $\mathtt{i}(t') = \mathtt{i}(t)$ and $S_{t,j} = S_{t',j}$.
4. $bad \leftarrow 1$ if, for some distinct t_1 and t_2 in $[1,q]$, $\mathtt{i}(t_1) = \mathtt{i}(t_2)$ and

$$\bigoplus_{j=1}^{s_{t_1}} Y_{t_1,j} = \bigoplus_{j=1}^{s_{t_2}} Y_{t_2,j}. \tag{43}$$

Since R is identical to $\rho_1, \ldots, \rho_m$, $\Pr[\mathbf{A}^{\mathsf{R}} = 1] = \Pr[\mathbf{A}^{\rho_1,\ldots,\rho_m} = 1]$. As long as $bad = 0$, R is also identical to $\hat{\mathrm{V}}^{\mu_1}, \ldots, \hat{\mathrm{V}}^{\mu_m}$. Notice that, for distinct t_1 and t_2 in $[1,q]$ such that $\mathtt{i}(t_1) = \mathtt{i}(t_2)$,

$$\Pr\left[\bigoplus_{j=1}^{s_{t_1}} Y_{t_1,j} = \bigoplus_{j=1}^{s_{t_2}} Y_{t_2,j}\right] \leq \frac{1}{2^n}. \tag{44}$$

Thus,

$$\left|\Pr[\mathbf{A}^{\hat{\mathrm{V}}^{\mu_1},\ldots,\hat{\mathrm{V}}^{\mu_m}} = 1] - \Pr[\mathbf{A}^{\rho_1,\ldots,\rho_m} = 1]\right| \leq \sum_{i=1}^{m} \frac{q_i(q_i-1)}{2^{n+1}} \leq \frac{q^2}{2^{n+1}}, \tag{45}$$

where q_i is the number of the queries to the i-th oracle and $q_1 + \cdots + q_m \leq q$. □

6. Conclusion

In this paper, the PRF security of FMAC and vFMAC in the multi-user setting is reduced to that of their compression function against related-key attacks with respect to their non-cryptographic permutations in the single user setting. This result shows that the PRF security of FMAC and vFMAC in the multi-user setting is as good as in the single user setting.

Future work is to evaluate the security of other PRFs and vPRFs in the multi-user setting.

Acknowledgments

We would like to thank the reviewers for their valuable comments. We would also like to thank Atsushi Yabumoto and Hibiki Hanai for their fruitful discussions. This work was supported in part by JSPS KAKENHI Grant Number JP16H02828.

References

[1] M. Bellare, "New proofs for NMAC and HMAC: Security without collision-resistance," CRYPTO 2006, Proceedings, C. Dwork, ed., Lecture Notes in Computer Science, vol.4117, pp.602–619, Springer, 2006.
[2] M. Bellare, D.J. Bernstein, and S. Tessaro, "Hash-function based PRFs: AMAC and its multi-user security," EUROCRYPT 2016, Proceedings, Part I, M. Fischlin and J. Coron, eds., Lecture Notes in Computer Science, vol.9665, pp.566–595, Springer, 2016.
[3] M. Bellare, R. Canetti, and H. Krawczyk, "Keying hash functions for message authentication," CRYPTO '96, Proceedings, N. Koblitz, ed., Lecture Notes in Computer Science, vol.1109, pp.1–15, Springer, 1996.
[4] M. Bellare, R. Canetti, and H. Krawczyk, "Pseudorandom functions revisited: The cascade construction and its concrete security," Proc. 37th IEEE Symposium on Foundations of Computer Science, pp.514–523, 1996.
[5] M. Bellare, J. Kilian, and P. Rogaway, "The security of cipher block chaining," CRYPTO'94, Proceedings, Y. Desmedt, ed., Lecture Notes in Computer Science, vol.839, pp.341–358, Springer, 1994.
[6] M. Bellare and T. Kohno, "A theoretical treatment of related-key attacks: RKA-PRPs, RKA-PRFs, and applications," EUROCRYPT 2003, Proceedings, E. Biham, ed., Lecture Notes in Computer Science, vol.2656, pp.491–506, Springer, 2003.
[7] M. Bellare and T. Ristenpart, "Multi-property-preserving hash domain extension and the EMD transform," ASIACRYPT 2006, Proceedings, X. Lai and K. Chen, eds., Lecture Notes in Computer Science, vol.4284, pp.299–314, Springer, 2006. The full version is "Cryptology ePrint Archive: Report 2006/399" at http://eprint.iacr.org/
[8] M. Bellare and P. Rogaway, "On the construction of variable-input-length ciphers," FSE'99, Proceedings, L.R. Knudsen, ed., Lecture Notes in Computer Science, vol.1636, pp.231–244, Springer, 1999.
[9] M. Bellare and B. Tackmann, "The multi-user security of authenticated encryption: AES-GCM in TLS 1.3," CRYPTO 2016, Proceedings, Part I, M. Robshaw and J. Katz, eds., Lecture Notes in Computer Science, vol.9814, pp.247–276, Springer, 2016.
[10] FIPS PUB 180-4, "Secure hash standard (SHS)," Aug. 2015.
[11] P. Gaži, K. Pietrzak, and M. Rybár, "The exact PRF-security of NMAC and HMAC," CRYPTO 2014, Proceedings, Part I, J.A. Garay and R. Gennaro, eds., Lecture Notes in Computer Science, vol.8616, pp.113–130, Springer, 2014.
[12] O. Goldreich, S. Goldwasser, and S. Micali, "How to construct random functions," J. ACM, vol.33, no.4, pp.792–807, 1986.
[13] S. Goldwasser and S. Micali, "Probabilistic encryption," J. Comput. Syst. Sci., vol.28, no.2, pp.270–299, 1984.
[14] S. Hirose, J.H. Park, and A. Yun, "A simple variant of the Merkle-Damgård scheme with a permutation," J. Cryptol., vol.25, no.2, pp.271–309, 2012.
[15] S. Hirose and A. Yabumoto, "A tweak for a PRF mode of a compression function and its applications," SECITC 2016, Revised Selected Papers, I. Bica and R. Reyhanitabar, eds., Lecture Notes in Computer Science, vol.10006, pp.103–114, 2016.

[16] V.T. Hoang and S. Tessaro, "Key-alternating ciphers and key-length extension: Exact bounds and multi-user security," CRYPTO 2016, Proceedings, Part I, M. Robshaw and J. Katz, eds., Lecture Notes in Computer Science, vol.9814, pp.3–32, Springer, 2016.
[17] V.T. Hoang and S. Tessaro, "The multi-user security of double encryption," EUROCRYPT 2017, Proceedings, Part II, J. Coron and J.B. Nielsen, eds., Lecture Notes in Computer Science, vol.10211, pp.381–411, 2017.
[18] A. Luykx, B. Mennink, and K.G. Paterson, "Analyzing multi-key security degradation," ASIACRYPT 2017, Proceedings, Part II, T. Takagi and T. Peyrin, eds., Lecture Notes in Computer Science, vol.10625, pp.575–605, Springer, 2017.
[19] A. Luykx, B. Preneel, E. Tischhauser, and K. Yasuda, "A MAC mode for lightweight block ciphers," FSE 2016, Revised Selected Papers, T. Peyrin, ed., Lecture Notes in Computer Science, vol.9783, pp.43–59, Springer, 2016.
[20] K. Minematsu, "A short universal hash function from bit rotation, and applications to blockcipher modes," ProvSec 2013, Proceedings, W. Susilo and R. Reyhanitabar, eds., Lecture Notes in Computer Science, vol.8209, pp.221–238, Springer, 2013.
[21] M. Nandi, "Fast and secure CBC-type MAC algorithms," FSE 2009, Revised Selected Papers, O. Dunkelman, ed., Lecture Notes in Computer Science, vol.5665, pp.375–393, Springer, 2009.
[22] P. Rogaway and T. Shrimpton, "A provable-security treatment of the key-wrap problem," EUROCRYPT 2006, Proceedings, S. Vaudenay, ed., Lecture Notes in Computer Science, vol.4004, pp.373–390, Springer, 2006.
[23] K. Yasuda, "HMAC without the "second" key," ISC 2009, Proceedings, P. Samarati, M. Yung, F. Martinelli, and C.A. Ardagna, eds., Lecture Notes in Computer Science, vol.5735, pp.443–458, Springer, 2009.

Shoichi Hirose received the B.E., M.E. and D.E. degrees in information science from Kyoto University, Kyoto, Japan, in 1988, 1990 and 1995, respectively. From 1990 to 1998, he was a research associate at Faculty of Engineering, Kyoto University. From 1998 to 2005, he was a lecturer at Graduate School of Informatics, Kyoto University. From 2005 to 2009, he was an associate professor at Faculty of Engineering, University of Fukui. From 2009, he is a professor at Graduate School of Engineering, University of Fukui. His current interests include cryptography and information security. He received Young Engineer Award from IEICE in 1997, and KDDI Foundation Research Award in 2008.