IEICE TRANSACTIONS

on Fundamentals of Electronics, Communications and Computer Sciences

DOI:10.1587/transfun.2024CIP0006

Publicized:2024/10/23

This advance publication article will be replaced by the finalized version after proofreading.

A PUBLICATION OF THE ENGINEERING SCIENCES SOCIETY

The Institute of Electronics, Information and Communication Engineers

Kikai-Shinko-Kaikan Bldg., 5-8, Shibakoen 3chome, Minato-ku, TOKYO, 105-0011 JAPAN

PAPER

Compactly Committing Authenticated Encryption Made Simpler

Shoichi HIROSE[†a)] ***and*** **Kazuhiko MINEMATSU**[††,†††]**, *Members***

SUMMARY In 2016, message franking was introduced by Facebook in end-to-end encrypted messaging. This feature enables recipients to report harmful content to their service provider in a verifiable manner. Grubbs et al. (CRYPTO 2017) formalized compactly committing authenticated encryption with associated data (ccAEAD) as a symmetric-key primitive that can be used for message franking and presented its generic constructions. Dodis et al. (CRYPTO 2018) proposed encryptment as a core component of ccAEAD and presented two transforms to build ccAEAD from encryptment. One transform builds randomized ccAEAD with one call to conventional AEAD, while the other builds nonce-based ccAEAD with two calls to a pseudorandom function (PRF). Hirose and Minematsu presented an improved transform that requires a tweakable block cipher instead of AEAD. This paper presents an even simplified transform to build randomized ccAEAD, which requires only one call to a PRF. The resulting ccAEAD is more efficient regarding bandwidth than Dodis et al. and has a smaller computation cost than Hirose and Minematsu. The presented transform can be extended to build nonce-based ccAEAD, which is also more efficient than the one presented by Dodis et al. regarding bandwidth, though it requires two calls to a PRF as well as their transform.

key words: *Authenticated encryption, Commitment, Pseudorandom function, Encryptment*

1. Introduction

1.1 Background

Many people enjoy end-to-end encrypted messaging services such as Facebook Messenger [1], Signal [2], and Whatsapp Messenger [3]. End-to-end messaging brings new security requirements apart from privacy and authenticity. One major concern is preventing malicious senders from sending harassing messages or harmful content. To achieve this goal, Facebook introduced message franking [4], a cryptographic protocol enabling users to report receiving abusive messages to Facebook in a verifiable manner.

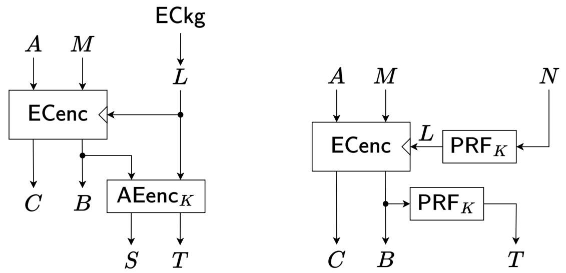
Grubbs et al. [5] initiated the formal study of message franking and introduced a new type of authenticated encryption with associated data (AEAD) [6], which they called compactly committing AEAD (ccAEAD). For ccAEAD, a small part of the ciphertext is used as a commitment value to the message and its associated data. Decryption returns an opening key together with a recovered message. Additionally, ccAEAD provides an algorithm that checks the recovered message against the commitment value using the opening key. Grubbs et al. also presented two generic constructions of ccAEAD: CtE (Commit-then-Encrypt), which combines a commitment scheme and an AEAD scheme; CEP (Committing Encrypt-and-PRF), which consists of a pseudorandom generator, a pseudorandom function (PRF), and a collision-resistant PRF.

Aiming to construct more efficient ccAEAD than CtE and CEP, Dodis et al. [7], [8] (DGRW18) abstracted a core component of ccAEAD, which they called encryptment. It is roughly one-time ccAEAD and simultaneously encrypts and commits to a given message. They constructed an encryptment scheme called HFC (hash function chaining) using a Merkle-Damgård hash function [9], [10] and presented two transforms to build ccAEAD from encryptment. One transform builds randomized ccAEAD with one call to conventional AEAD, and the other builds nonce-based ccAEAD with two calls to a PRF. The encryption algorithms of ccAEAD built by these transforms are depicted in Fig. 1. For the first transform, Hirose and Minematsu [11], [12] (HM23) demonstrated that AEAD can be replaced with a tweakable block cipher (TBC) [13], [14] as shown in Fig. 2. For Figures 1 and 2, ECkg and ECenc are key-generation and encryption algorithms of encryptment, respectively. AEenc is an encryption algorithm of AEAD. PRF is a PRF. E is a TBC. K is a secret key shared by a sender and a receiver. A is associated data, M is a message, and N is a nonce. L is a secret key for encryptment. C is a ciphertext, and B is a binding tag used as a commitment value for A and M. AEenc_K treats B and L as associated data and a message, respectively, and produces a ciphertext S and a tag T. E treats B as a tweak.

1.2 Our Contributions

We further simplify the transform to build ccAEAD from encryptment. The proposed transform needs one call to a PRF, which is depicted in Fig. 3a. In terms of the bandwidth of resultant ccAEAD, the transform of ours as well as that of HM23 is more efficient than that of Dodis et al. [7], [8]. From implementation perspective, ours generally has merits over HM23 as the PRF in our transform has a smaller input size than the TBC in HM23 (in terms of the total inputs, namely a tweak and a message block). Moreover, the TBC in HM23 needs both forward and backward circuits,

[†]The author is with Faculty of Engineering, University of Fukui, Fukui-shi, 910-8507 Japan.
[††]The author is with NEC Corporation, Kawasaki-shi, 211-8601 Japan.
[†††]The author is with Yokohama National University, Yokohama-shi, 240-8501 Japan.
a) E-mail: hrs_shch@u-fukui.ac.jp



(a) Randomized (using AEAD) (b) Nonce-based (using PRF)

Fig. 1: Encryption algorithms of ccAEAD built by transforms of Dodis et al. [7], [8]

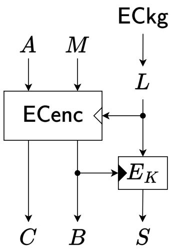


Fig. 2: Encryption algorithm of ccAEAD built by transform of Hirose and Minematsu [11], [12]

which even increases the footprint. The transform can be extended straightforwardly to nonce-based ccAEAD as shown in Fig. 3b. It needs two calls to a PRF as well as that of Dodis et al. [7], [8]. However, the former has a smaller bandwidth for the resultant ccAEAD.

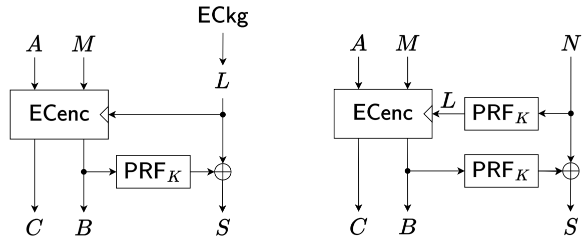
Table 1 summarizes the ccAEAD schemes built by the transforms from encryptment. Notice that all the transforms can use the identical encryptment scheme. For the randomized ccAEAD schemes, S and L have the same length. For the nonce-based ccAEAD schemes, S and N have the same length.

The security requirements of ccAEAD built by the proposed transforms are reduced to those of the underlying encryptment and PRF. For ciphertext integrity, the proposed transforms as well as that of HM23 require that the underlying encryptment satisfies targeted-ciphertext unforgeability, which is relevant to preimage resistance of a cryptographic hash function family. On the other hand, the transform of Dodis et al. [7], [8] requires that the underlying encryptment satisfies second-ciphertext unforgeability, which is relevant to second-preimage resistance of a cryptographic hash function family.

1.3 Related Work

Authenticated encryption is a symmetric-key primitive providing privacy and authenticity. It has been attracting interests among researchers for many years. Its formal treatments were initiated by Katz and Yung [15] and by Bellare and Namprempre [16].

Message franking schemes with additional features



(a) Randomized (b) Nonce-based

Fig. 3: Encryption algorithms of ccAEAD built by proposed transforms

were also discussed in the literature. Message franking schemes enabling recipients to report abusive messages by revealing only abusive parts were investigated independently by Leontiadis and Vaudenay [17], [18] and by Chen and Tang [19]. A secure bidirectional channel with message franking was formalized and instantiated by Huguenin-Dumittan and Leontiadis [20]. Yamamuro et al. [21], [22] formalized forward secure message franking and presented its generic constructions. Tyagi et al. [23] formalized asymmetric message franking and presented construction from signatures of knowledge [24] for designated verifier signatures [25].

Hirose [26] instantiated the transform of Dodis et al. [7], [8] to build nonce-based ccAEAD (Fig. 1b) only with a TBC. The instantiation does not reduce the bandwidth.

Farshim et al. [27], Albertini et al. [28], Len et al. [29], Bellare and Hoang [30], and Chan and Rogaway [31] investigated so-called committing authenticated encryption. While their definitions and security goals are not identical, their primary goal was basically to decrease the risk of error or misuse by application designers, and message franking was out of scope for the lack of opening key needed by ccAEAD.

1.4 Organization

Section 2 introduces notations and formalizes pseudorandom functions, ccAEAD, and encryptment. Section 3 describes the proposed transforms to build ccAEAD from encryptment and proves the security of resultant ccAEAD. Section 4 gives brief concluding remarks.

2. Preliminaries

Let $\mathbb{N}$ be the set of non-negative integers. Let $\Sigma := \{0, 1\}$. For any integer $l \geq 0$, let Σ^l be the set of all Σ-sequences of length l. Let $\Sigma^* := \bigcup_{i \geq 0} \Sigma^i$. The length of $x \in \Sigma^*$ is denoted by $|x|$. Concatenation of $x_1, x_2 \in \Sigma^*$ is denoted by $x_1 \| x_2$. A uniform random choice of an element s from a set $\mathcal{S}$ is denoted by $s \twoheadleftarrow \mathcal{S}$.

2.1 Pseudorandom Functions

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

Table 1: ccAEAD schemes built by the transforms from encryptment. 'Type' indicates randomized (R) or nonce-based (N). 'Primitive' indicates a required primitive. '# calls' indicates the number of calls to the primitive. 'Overall ciphertext' indicates components sent to a receiver.

Scheme	Type	Primitive	# calls	Overall ciphertext
DGRW18 (Fig. 1a)	R	AEAD	1	A, C, B, S, T
HM23 (Fig. 2)	R	TBC	1	A, C, B, S
Proposed (Fig. 3a)	R	PRF	1	A, C, B, S
DGRW18 (Fig. 1b)	N	PRF	2	N, A, C, B, T
Proposed (Fig. 3b)	N	PRF	2	A, C, B, S

adversary which has a function from $\mathcal{D}_f$ to $\mathcal{R}_f$ as an oracle and outputs 0 or 1. The advantage of $\mathbf{A}$ against f concerning a pseudorandom function (PRF) is given by

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

where $K \twoheadleftarrow \mathcal{K}_f$, and $\rho : \mathcal{D}_f \to \mathcal{R}_f$ is a uniform random function.

2.2 ccAEAD

2.2.1 Syntax

Following the convention [5], [7], we first formalize the syntax of randomized ccAEAD and then formalize that of nonce-based ccAEAD. We refer to randomized ccAEAD as ccAEAD.

A tuple of algorithms $\mathsf{CAE} := (\mathsf{CAEkg}, \mathsf{CAEenc}, \mathsf{CAEdec}, \mathsf{CAEver})$ specifies ccAEAD. CAEkg is a probabilistic algorithm for key generation. CAEenc is a probabilistic algorithm for encryption. CAEdec is a deterministic algorithm for decryption. CAEver is a deterministic algorithm for verification. ccAEAD is involved with the following subsets of Σ^*: a key space $\mathcal{K}_{\mathrm{CAE}}$, an associated-data space $\mathcal{A}_{\mathrm{CAE}}$, a message space $\mathcal{M}_{\mathrm{CAE}}$, a ciphertext space C_{CAE}, an opening-key space $\mathcal{L}_{\mathrm{CAE}}$, a binding-tag space $\mathcal{T}_{\mathrm{CAE}}$, and an attachment space $\mathcal{S}_{\mathrm{CAE}}$. For every $l \in \mathbb{N}$, $\Sigma^l \subseteq \mathcal{M}_{\mathrm{CAE}}$ or $\Sigma^l \cap \mathcal{M}_{\mathrm{CAE}} = \emptyset$. A targeted security level of ccAEAD determines the key length n, the opening-key length ℓ, the binding-tag length τ, and the attachment length σ. Thus, $\mathcal{K}_{\mathrm{CAE}} := \Sigma^n$, $\mathcal{L}_{\mathrm{CAE}} := \Sigma^\ell$, $\mathcal{T}_{\mathrm{CAE}} := \Sigma^\tau$, and $\mathcal{S}_{\mathrm{CAE}} := \Sigma^\sigma$. The compactly-committing property requires that τ is small.

- CAEkg returns a secret key $K \in \mathcal{K}_{\mathrm{CAE}}$ chosen uniformly at random.
- CAEenc takes $(K, A, M) \in \mathcal{K}_{\mathrm{CAE}} \times \mathcal{A}_{\mathrm{CAE}} \times \mathcal{M}_{\mathrm{CAE}}$ as input and returns $(C, B, S) \in C_{\mathrm{CAE}} \times \mathcal{T}_{\mathrm{CAE}} \times \mathcal{S}_{\mathrm{CAE}}$. $|C|$ depends only on $|M|$, and let $\mathsf{clen} : \mathbb{N} \to \mathbb{N}$ be a function such that $|C| = \mathsf{clen}(|M|)$.
- CAEdec takes $(K, A, C, B, S) \in \mathcal{K}_{\mathrm{CAE}} \times \mathcal{A}_{\mathrm{CAE}} \times C_{\mathrm{CAE}} \times \mathcal{T}_{\mathrm{CAE}} \times \mathcal{S}_{\mathrm{CAE}}$ as input and returns $(M, L) \in \mathcal{M}_{\mathrm{CAE}} \times \mathcal{L}_{\mathrm{CAE}}$ or $\perp \notin \mathcal{M}_{\mathrm{CAE}} \times \mathcal{L}_{\mathrm{CAE}}$.
- CAEver takes $(A, M, L, B) \in \mathcal{A}_{\mathrm{CAE}} \times \mathcal{M}_{\mathrm{CAE}} \times \mathcal{L}_{\mathrm{CAE}} \times \mathcal{T}_{\mathrm{CAE}}$ as input and returns $b \in \Sigma$.

It is common that CAE is assumed to satisfy correctness: For any $(K, A, M) \in \mathcal{K}_{\mathrm{CAE}} \times \mathcal{A}_{\mathrm{CAE}} \times \mathcal{M}_{\mathrm{CAE}}$, if $(C, B, S) \leftarrow \mathsf{CAEenc}(K, A, M)$, then there exists some $L \in \mathcal{L}_{\mathrm{CAE}}$ such that $\mathsf{CAEdec}(K, A, C, B, S) = (M, L)$ and $\mathsf{CAEver}(A, M, L, B) = 1$.

Remark 1 In the formalization by Grubbs et al. [5] and Dodis et al. [7], $(C, S) \in C_{\mathrm{CAE}} \times \mathcal{S}_{\mathrm{CAE}}$ is specified as a ciphertext.

Nonce-based ccAEAD is specified by a tuple of algorithms $\mathsf{nCAE} := (\mathsf{CAEkg}, \mathsf{nCAEenc}, \mathsf{CAEdec}, \mathsf{CAEver})$. The difference between CAE and nCAE is minor. They share CAEkg, CAEdec, and CAEver. $\mathsf{nCAEenc}$ is a deterministic algorithm for encryption. It takes as input $(K, N, A, M) \in \mathcal{K}_{\mathrm{CAE}} \times \mathcal{N}_{\mathrm{CAE}} \times \mathcal{A}_{\mathrm{CAE}} \times \mathcal{M}_{\mathrm{CAE}}$ and returns $(C, B, S) \in C_{\mathrm{CAE}} \times \mathcal{T}_{\mathrm{CAE}} \times \mathcal{S}_{\mathrm{CAE}}$, where $\mathcal{N}_{\mathrm{CAE}} \subseteq \Sigma^*$ is a nonce space.

2.2.2 Security Requirements

The security requirements of (nonce-based) ccAEAD are confidentiality, ciphertext integrity, and binding properties. Confidentiality and ciphertext integrity are inherited from AEAD and tailored to (nonce-based) ccAEAD. The binding properties are specific to (nonce-based) ccAEAD.

Hereafter, the security requirements are formalized only for ccAEAD. They are similarly formalized for nonce-based ccAEAD since the syntax of nonce-based ccAEAD is very similar to that of ccAEAD.

(1) Confidentiality

Confidentiality is formalized as real-or-random indistinguishability in the multi-opening setting. The advantage of an adversary $\mathbf{A}$ for confidentiality of CAE is

$$\mathrm{Adv}_{\mathsf{CAE}}^{\mathrm{mo\text{-}ror}}(\mathbf{A}) := \left|\Pr[\text{MO-REAL}_{\mathsf{CAE}}^{\mathbf{A}} = 1] - \Pr[\text{MO-RAND}_{\mathsf{CAE}}^{\mathbf{A}} = 1]\right|,$$

where the games $\text{MO-REAL}_{\mathsf{CAE}}^{\mathbf{A}}$ and $\text{MO-RAND}_{\mathsf{CAE}}^{\mathbf{A}}$ are shown in Fig. 4. $\mathbf{A}$ is allowed to access the oracles **Enc**, **Dec**, and **ChalEnc**. The same **Enc** and **Dec** oracles are given to $\mathbf{A}$ in both of the games. **Dec** returns (M, L) for any query (A, C, B, S) if it appears in the previous query-response pairs for **Enc** (multi-opening setting). Otherwise, **Dec** returns $\perp$. For each query, **ChalEnc** returns the output of CAEenc in MO-REAL and a uniform random sequence in MO-RAND.

(2) Ciphertext Integrity

Ciphertext integrity is formalized as existential unforgeabil-

```
K ← CAEkg(); 𝒴 ← ∅
b ← A^{Enc,Dec,ChalEnc}                    ▷ b ∈ Σ
return b

Enc(A, M)
(C, B, S) ← CAEenc(K, A, M)
𝒴 ← 𝒴 ∪ {(A, C, B, S)}
return (C, B, S)

Dec(A, C, B, S)
if (A, C, B, S) ∉ 𝒴 then
    return ⊥
end if
(M, L) ← CAEdec(K, A, C, B, S)
return (M, L)

ChalEnc(A, M)
(C, B, S) ← CAEenc(K, A, M)
return (C, B, S)
```

(a) MO-REAL$^{\mathbf{A}}_{\mathsf{CAE}}$

```
K ← CAEkg(); 𝒴 ← ∅
b ← A^{Enc,Dec,ChalEnc}                    ▷ b ∈ Σ
return b

Enc(A, M)
(C, B, S) ← CAEenc(K, A, M)
𝒴 ← 𝒴 ∪ {(A, C, B, S)}
return (C, B, S)

Dec(A, C, B, S)
if (A, C, B, S) ∉ 𝒴 then
    return ⊥
end if
(M, L) ← CAEdec(K, A, C, B, S)
return (M, L)

ChalEnc(A, M)
(C, B, S) ↞ Σ^{clen(|M|)} × Σ^τ × Σ^σ
return (C, B, S)
```

(b) MO-RAND$^{\mathbf{A}}_{\mathsf{CAE}}$

Fig. 4: Games for confidentiality of ccAEAD

```
K ← CAEkg(); 𝒴 ← ∅
win ← 0
A^{Enc,Dec,ChalDec}
return win                          ChalDec(A, C, B, S)
                                    if (A, C, B, S) ∈ 𝒴 then
Enc(A, M)                               return ⊥
(C, B, S) ← CAEenc(K, A, M)         end if
𝒴 ← 𝒴 ∪ {(A, C, B, S)}              if CAEdec(K, A, C, B, S) ≠ ⊥
return (C, B, S)                    then
                                        win ← 1
Dec(A, C, B, S)                     end if
return CAEdec(K, A, C, B, S)        return CAEdec(K, A, C, B, S)
```

Fig. 5: Game MO-CTXT$^{\mathbf{A}}_{\mathsf{CAE}}$ for ciphertext integrity of ccAEAD

ity in the multi-opening setting. The advantage of an adversary $\mathbf{A}$ for ciphertext integrity of CAE is

$$\mathrm{Adv}^{\text{mo-ctxt}}_{\mathsf{CAE}}(\mathbf{A}) := \Pr[\text{MO-CTXT}^{\mathbf{A}}_{\mathsf{CAE}} = 1],$$

where the game MO-CTXT$^{\mathbf{A}}_{\mathsf{CAE}}$ is shown in Fig. 5. $\mathbf{A}$ is allowed to access the oracles **Enc**, **Dec**, and **ChalDec**. The game outputs 1 if $\mathbf{A}$ asks a successful query to **ChalDec** which does not appear in the previous query-response pairs for **Enc**.

(3) Binding Properties

Binding properties are formalized with respect to a sender and a receiver. Receiver binding describes that a malicious receiver should not be able to blame an honest sender. The advantage of an adversary $\mathbf{A}$ for receiver binding of CAE is

$$\mathrm{Adv}^{\text{r-bind}}_{\mathsf{CAE}}(\mathbf{A}) := \Pr[((A, M, L), (A', M', L'), B) \leftarrow \mathbf{A} : (A, M) \neq (A', M') \wedge \mathsf{CAEver}(A, M, L, B) = \mathsf{CAEver}(A', M', L', B) = 1].$$

The advantage of $\mathbf{A}$ for strong receiver binding of CAE is

$$\mathrm{Adv}^{\text{sr-bind}}_{\mathsf{CAE}}(\mathbf{A}) := \Pr[((A, M, L), (A', M', L'), B) \leftarrow \mathbf{A} : (A, M, L) \neq (A', M', L') \wedge \mathsf{CAEver}(A, M, L, B) = \mathsf{CAEver}(A', M', L', B) = 1].$$

It is apparent that $\mathrm{Adv}^{\text{r-bind}}_{\mathsf{CAE}}(\mathbf{A}) \leq \mathrm{Adv}^{\text{sr-bind}}_{\mathsf{CAE}}(\mathbf{A})$.

Sender binding describes that a dishonest sender should be blamed. The advantage of $\mathbf{A}$ for sender binding of CAE is

$$\mathrm{Adv}^{\text{s-bind}}_{\mathsf{CAE}}(\mathbf{A}) := \Pr[(K, A, C, B, S) \leftarrow \mathbf{A} : \mathsf{CAEdec}(K, A, C, B, S) \neq \perp \wedge (M, L) \leftarrow \mathsf{CAEdec}(K, A, C, B, S) \wedge \mathsf{CAEver}(A, M, L, B) = 0].$$

Remark 2 (Message Franking Using ccAEAD) A service provider is responsible for relaying all communications among users. Users encrypt their communication using ccAEAD. When a sender sends a ciphertext, the service provider computes a tag using a MAC function to the binding tag in the ciphertext, and then sends the ciphertext and the tag to the receiver. If the receiver recovers an abusive message from the ciphertext, then they report it to the service provider along with the opening key, binding tag, and the tag attached by the service provider.

2.3 Encryptment

Encryptment is relatively a new primitive introduced and formalized by Dodis et al. [7]. It is roughly one-time ccAEAD, and its formal descriptions are similar to those of ccAEAD in many aspects.

2.3.1 Syntax

A tuple of algorithms $\mathsf{EC} = (\mathsf{ECkg}, \mathsf{ECenc}, \mathsf{ECdec}, \mathsf{ECver})$ specifies encryptment. ECkg is a probabilistic algorithm for key generation. ECenc is a deterministic algorithm for encryptment. ECdec is a deterministic algorithm for decryptment. ECver is a deterministic algorithm for verification.

Encryptment is involved with the following subsets of Σ^*: a key space $\mathcal{K}_{\mathrm{EC}}$, an associated-data space $\mathcal{A}_{\mathrm{EC}}$, a message space $\mathcal{M}_{\mathrm{EC}}$, a ciphertext space $\mathcal{C}_{\mathrm{EC}}$, and a binding-tag space $\mathcal{T}_{\mathrm{EC}}$. A targeted security level determines the key length ℓ and the binding-tag length τ. Thus, $\mathcal{K}_{\mathrm{EC}} := \Sigma^\ell$, and $\mathcal{T}_{\mathrm{EC}} := \Sigma^\tau$.

- ECkg returns a secret key $K_{\mathrm{ec}} \in \mathcal{K}_{\mathrm{EC}}$ chosen uniformly at random.
- ECenc takes $(K_{\mathrm{ec}}, A, M) \in \mathcal{K}_{\mathrm{EC}} \times \mathcal{A}_{\mathrm{EC}} \times \mathcal{M}_{\mathrm{EC}}$ as input and returns $(C, B) \in \mathcal{C}_{\mathrm{EC}} \times \mathcal{T}_{\mathrm{EC}}$. $|C|$ depends only on $|M|$.
- ECdec takes $(K_{\mathrm{ec}}, A, C, B) \in \mathcal{K}_{\mathrm{EC}} \times \mathcal{A}_{\mathrm{EC}} \times \mathcal{C}_{\mathrm{EC}} \times \mathcal{T}_{\mathrm{EC}}$ as input and returns $M \in \mathcal{M}_{\mathrm{EC}}$ or $\perp \notin \mathcal{M}_{\mathrm{EC}}$.
- ECver takes $(A, M, K_{\mathrm{ec}}, B) \in \mathcal{A}_{\mathrm{EC}} \times \mathcal{M}_{\mathrm{EC}} \times \mathcal{K}_{\mathrm{EC}} \times \mathcal{T}_{\mathrm{EC}}$ as input and returns $b \in \Sigma$.

EC is assumed to satisfy correctness as well as ccAEAD. Namely, for any $(K_{\mathrm{ec}}, A, M) \in \mathcal{K}_{\mathrm{EC}} \times \mathcal{A}_{\mathrm{EC}} \times \mathcal{M}_{\mathrm{EC}}$, if $(C, B) \leftarrow \mathsf{ECenc}(K_{\mathrm{ec}}, A, M)$, then $\mathsf{ECdec}(K_{\mathrm{ec}}, A, C, B) = M$ and $\mathsf{ECver}(A, M, K_{\mathrm{ec}}, B) = 1$. EC is said to satisfy strong correctness if, for any $(K_{\mathrm{ec}}, A, C, B) \in \mathcal{K}_{\mathrm{EC}} \times \mathcal{A}_{\mathrm{EC}} \times \mathcal{C}_{\mathrm{EC}} \times \mathcal{T}_{\mathrm{EC}}$, $\mathsf{ECdec}(K_{\mathrm{ec}}, A, C, B) \neq \perp$, then $\mathsf{ECenc}(K_{\mathrm{ec}}, A, \mathsf{ECdec}(K_{\mathrm{ec}}, A, C, B)) = (C, B)$.

2.3.2 Security Requirements

The security requirements of encryptment are confidentiality, second-ciphertext unforgeability, and binding properties [7], [8]. Targeted-ciphertext unforgeability is also introduced [11], [12].

(1) Confidentiality

Confidentiality is formalized as real-or-random indistinguishability. The advantage of an adversary $\mathbf{A}$ for confidentiality of EC is

$$\mathrm{Adv}^{\text{ot-ror}}_{\mathsf{EC}}(\mathbf{A}) := \left|\Pr[\mathrm{otREAL}^{\mathbf{A}}_{\mathsf{EC}} = 1] - \Pr[\mathrm{otRAND}^{\mathbf{A}}_{\mathsf{EC}} = 1]\right|,$$

where the games $\mathrm{otREAL}^{\mathbf{A}}_{\mathsf{EC}}$ and $\mathrm{otRAND}^{\mathbf{A}}_{\mathsf{EC}}$ are shown in Fig. 6. $\mathbf{A}$ is allowed to ask a single query to the **enc** oracle.

(a) $\mathrm{otREAL}^{\mathbf{A}}_{\mathsf{EC}}$	(b) $\mathrm{otRAND}^{\mathbf{A}}_{\mathsf{EC}}$
$K_{\mathrm{ec}} \leftarrow \mathsf{ECkg}()$	
$b \leftarrow \mathbf{A}^{\mathbf{enc}}$ $\quad \triangleright\, b \in \Sigma$	$b \leftarrow \mathbf{A}^{\mathbf{enc}}$ $\quad \triangleright\, b \in \Sigma$
return b	**return** b
$\mathbf{enc}(A, M)$	$\mathbf{enc}(A, M)$
$(C, B) \leftarrow \mathsf{ECenc}(K_{\mathrm{ec}}, A, M)$	$(C, B) \twoheadleftarrow \Sigma^{\mathsf{clen}(\lvert M\rvert)} \times \Sigma^\tau$
return (C, B)	**return** (C, B)

Fig. 6: Games for confidentiality of encryptment

(2) Second-Ciphertext Unforgeability

An adversary $\mathbf{A}$ first makes a query $(A, M) \in \mathcal{A}_{\mathrm{EC}} \times \mathcal{M}_{\mathrm{EC}}$ to $\mathsf{ECenc}(K_{\mathrm{ec}}, \cdot, \cdot)$ and gets (C, B) and K_{ec}, where $K_{\mathrm{ec}} \leftarrow \mathsf{ECkg}()$ and $(C, B) \leftarrow \mathsf{ECenc}_{K_{\mathrm{ec}}}(A, M)$. Then, $\mathbf{A}$ outputs $(A', C') \in \mathcal{A}_{\mathrm{EC}} \times \mathcal{C}_{\mathrm{EC}}$. The advantage of $\mathbf{A}$ for second-ciphertext unforgeability of EC is

$$\mathrm{Adv}^{\mathrm{scu}}_{\mathsf{EC}}(\mathbf{A}) := \Pr[(A, C) \neq (A', C') \wedge \mathsf{ECdec}_{K_{\mathrm{ec}}}(A', C', B) \neq \perp].$$

Second-ciphertext unforgeability recalls second-preimage resistance of a cryptographic hash function family.

(3) Targeted-Ciphertext Unforgeability

Targeted-ciphertext unforgeability [11], [12] recalls everywhere preimage resistance of a cryptographic hash function family [32]. Let $\mathbf{A} := (\mathbf{A}_1, \mathbf{A}_2)$ be a two-phase adversary. First, $\mathbf{A}_1$ takes no input and outputs (B, state), where $B \in \mathcal{T}_{\mathrm{EC}}$ and state is some state information. Then, $\mathbf{A}_2$ takes (B, state) and K_{ec} as input and outputs $(A, C) \in \mathcal{A}_{\mathrm{EC}} \times \mathcal{C}_{\mathrm{EC}}$, where $K_{\mathrm{ec}} \leftarrow \mathsf{ECkg}()$. The advantage of $\mathbf{A}$ for targeted-ciphertext unforgeability of EC is

$$\mathrm{Adv}^{\mathrm{tcu}}_{\mathsf{EC}}(\mathbf{A}) := \Pr[\mathsf{ECdec}(K_{\mathrm{ec}}, A, C, B) \neq \perp].$$

It is shown that the HFC (hash function chaining) encryptment scheme [7], [8] satisfies targeted-ciphertext unforgeability in the random oracle model [11], [12].

(4) Binding properties

The advantage of $\mathbf{A}$ for receiver binding of EC is

$$\begin{aligned}\mathrm{Adv}^{\text{r-bind}}_{\mathsf{EC}}(\mathbf{A}) :=\ & \Pr[((K_{\mathrm{ec}}, A, M), (K'_{\mathrm{ec}}, A', M'), B) \leftarrow \mathbf{A} : \\ & (A, M) \neq (A', M') \wedge \\ & \mathsf{ECver}(A, M, K_{\mathrm{ec}}, B) = \mathsf{ECver}(A', M', K'_{\mathrm{ec}}, B) = 1].\end{aligned}$$

The advantage of $\mathbf{A}$ for strong receiver binding of EC is

$$\begin{aligned}\mathrm{Adv}^{\text{sr-bind}}_{\mathsf{EC}}(\mathbf{A}) :=\ & \Pr[((K_{\mathrm{ec}}, A, M), (K'_{\mathrm{ec}}, A', M'), B) \leftarrow \mathbf{A} : \\ & (K_{\mathrm{ec}}, A, M) \neq (K'_{\mathrm{ec}}, A', M') \wedge \\ & \mathsf{ECver}(A, M, K_{\mathrm{ec}}, B) = \mathsf{ECver}(A', M', K'_{\mathrm{ec}}, B) = 1].\end{aligned}$$

The advantage of an adversary $\mathbf{A}$ for sender binding of EC is

$$\begin{aligned}\mathrm{Adv}^{\text{s-bind}}_{\mathsf{EC}}(\mathbf{A}) :=\ & \Pr[(K_{\mathrm{ec}}, A, C, B) \leftarrow \mathbf{A}, M \leftarrow \mathsf{ECdec}(K_{\mathrm{ec}}, A, C, B) : \\ & M \neq \perp \wedge \mathsf{ECver}(A, M, K_{\mathrm{ec}}, B) = 0].\end{aligned}$$

For strongly correct encryptment, receiver binding implies second-ciphertext unforgeability, while the converse does not hold [11], [12]:

Proposition 1 Let EC be a strongly correct encryptment scheme. Then, for any adversary $\mathbf{A}$ against EC for second-ciphertext unforgeability, there exists an adversary $\dot{\mathbf{A}}$ such that $\mathrm{Adv}^{\mathrm{scu}}_{\mathsf{EC}}(\mathbf{A}) \leq \mathrm{Adv}^{\text{r-bind}}_{\mathsf{EC}}(\dot{\mathbf{A}})$ and the run time of $\dot{\mathbf{A}}$ is at most about that of $\mathbf{A}$.

Strongly correct encryptment satisfies sender binding:

Proposition 2 Let EC be a strongly correct encryptment scheme. Then, for any adversary $\mathbf{A}$ against EC for sender binding, $\mathrm{Adv}^{\text{s-bind}}_{\mathsf{EC}}(\mathbf{A}) = 0$.

Proof For $(K_{\mathrm{ec}}, A, C, B) \in \mathcal{K}_{\mathrm{EC}} \times \mathcal{A}_{\mathrm{EC}} \times \mathcal{C}_{\mathrm{EC}} \times \mathcal{T}_{\mathrm{EC}}$, suppose that there exists some $M \in \mathcal{M}_{\mathrm{EC}}$ such that $\mathsf{ECdec}(K_{\mathrm{ec}}, A, C, B) = M$. Then, $\mathsf{ECenc}(K_{\mathrm{ec}}, A, M) = (C, B)$ since EC satisfies strong correctness. Thus, $\mathsf{ECver}(A, M, K_{\mathrm{ec}}, B) = 1$ follows from correctness of EC. □

3. ccAEAD Using Encryptment and PRF

3.1 Scheme

The proposed transforms to build ccAEAD from encryptment construct randomized ccAEAD ECP (EnCryptment-and-Prf) $\mathsf{ECP} := (\mathsf{KG}, \mathsf{ENC}, \mathsf{DEC}, \mathsf{VER})$ and nonce-based ccAEAD $\mathsf{nECP} := (\mathsf{KG}, \mathsf{nENC}, \mathsf{nDEC}, \mathsf{VER})$. They combine strongly correct encryptment $\mathsf{EC} := (\mathsf{ECkg}, \mathsf{ECenc}, \mathsf{ECdec}, \mathsf{ECver})$ and a PRF F.

ECP and nECP are involved with a key space $\mathcal{K} := \Sigma^n$, an associated-data space $\mathcal{A} := \mathcal{A}_{\mathrm{EC}}$, a message space $\mathcal{M} := \mathcal{M}_{\mathrm{EC}}$, a ciphertext space $\mathcal{C} := \mathcal{C}_{\mathrm{EC}}$, an opening-key space $\mathcal{L} := \Sigma^\ell (= \mathcal{K}_{\mathrm{EC}})$, a binding-tag space $\mathcal{T} := \mathcal{T}_{\mathrm{EC}}$, and an attachment space $\mathcal{S} = \mathcal{L}$. nECP is also involved with a nonce space $\mathcal{N} := \Sigma^\ell$. ECP and nECP share KG and VER. KG returns a secret key $K \twoheadleftarrow \Sigma^n$ for PRF, and VER simply runs ECver. ENC and DEC are shown in Fig. 7, and nENC and nDEC are shown in Fig. 8. ENC and nENC are also depicted in Fig. 3. For nECP, it is assumed that the nonce space and the binding-tag space are disjoint for domain separation of F.

```
ENC(K, A, M)                   DEC(K, A, C, B, S)
  L <<- Σ^ℓ                      L <- F_K(B) ⊕ S
  (C, B) <- ECenc(L, A, M)       M <- ECdec(L, A, C, B)
  S <- F_K(B) ⊕ L                if M = ⊥ then
  return (C, B, S)                   return ⊥
                                 end if
                                 return (M, L)
```

Fig. 7: The encryption and decryption algorithms of ECP

```
nENC(K, N, A, M)               nDEC(K, A, C, B, S)
  L <- F_K(N)                    N <- F_K(B) ⊕ S
  (C, B) <- ECenc(L, A, M)       L <- F_K(N)
  S <- F_K(B) ⊕ N                M <- ECdec(L, A, C, B)
  return (C, B, S)               if M = ⊥ then
                                     return ⊥
                                 end if
                                 return (M, L)
```

Fig. 8: The encryption and decryption algorithms of nECP

3.2 Security

3.2.1 Confidentiality

Confidentiality of ECP is reduced to confidentiality and strong receiver binding of EC and the PRF property of F:

Theorem 1 For any adversary $\mathbf{A}$ against ECP making at most q_{e} and q_{c} queries to **Enc** and **ChalEnc**, respectively, there exist adversaries $\dot{\mathbf{A}}$, $\ddot{\mathbf{A}}$, and $\dddot{\mathbf{A}}$ such that

$$\mathrm{Adv}^{\text{mo-ror}}_{\mathsf{ECP}}(\mathbf{A}) \le \mathrm{Adv}^{\text{sr-bind}}_{\mathsf{EC}}(\dot{\mathbf{A}}) + q_{\mathrm{c}} \cdot \mathrm{Adv}^{\text{ot-ror}}_{\mathsf{EC}}(\ddot{\mathbf{A}}) + 2 \cdot \mathrm{Adv}^{\text{prf}}_{\mathsf{F}}(\dddot{\mathbf{A}}) + (q_{\mathrm{e}} + q_{\mathrm{c}})^2/2^{\ell+1}.$$

The run time of $\dot{\mathbf{A}}$, $\ddot{\mathbf{A}}$, and $\dddot{\mathbf{A}}$ is at most about that of $\text{MO-REAL}^{\mathbf{A}}_{\mathsf{ECP}}$. $\dddot{\mathbf{A}}$ makes at most $(q_{\mathrm{e}} + q_{\mathrm{c}})$ queries to its oracle.

Proof The games $\text{MO-REAL}^{\mathbf{A}}_{\mathsf{ECP}}$ and $\text{MO-RAND}^{\mathbf{A}}_{\mathsf{ECP}}$ are shown in Fig. 9. In the games, R keeps (M, L) with index (A, C, B, S) for each query (A, M) to **Enc**. Then,

$$\mathrm{Adv}^{\text{mo-ror}}_{\mathsf{ECP}}(\mathbf{A}) = \left|\Pr[\text{MO-REAL}^{\mathbf{A}}_{\mathsf{ECP}} = 1] - \Pr[\text{MO-RAND}^{\mathbf{A}}_{\mathsf{ECP}} = 1]\right|.$$

The game $\text{MO-ROR-G}^{\mathbf{A}}_1$ shown in Fig. 10 is obtained from $\text{MO-REAL}^{\mathbf{A}}_{\mathsf{ECP}}$ by replacing F_K with a uniform random function ρ. Let $\mathbf{A}_1$ be an adversary against F. $\mathbf{A}_1$ is given either F_K or ρ as an oracle. $\mathbf{A}_1$ simulates $\text{MO-REAL}^{\mathbf{A}}_{\mathsf{ECP}}$ and $\text{MO-ROR-G}^{\mathbf{A}}_1$ by making use of F_K and ρ, respectively. Then,

$$\begin{aligned}\mathrm{Adv}^{\text{prf}}_{\mathsf{F}}(\mathbf{A}_1) &= \left|\Pr[\mathbf{A}_1^{\mathsf{F}_K} = 1] - \Pr[\mathbf{A}_1^{\rho} = 1]\right| \\ &= \left|\Pr[\text{MO-REAL}^{\mathbf{A}}_{\mathsf{ECP}} = 1] - \Pr[\text{MO-ROR-G}^{\mathbf{A}}_1 = 1]\right|.\end{aligned}$$

The run time of $\mathbf{A}_1$ is at most about that of $\text{MO-REAL}^{\mathbf{A}}_{\mathsf{ECP}}$. $\mathbf{A}_1$ makes at most $(q_{\mathrm{e}} + q_{\mathrm{c}})$ queries to its oracle.

The game $\text{MO-ROR-G}^{\mathbf{A}}_2$ shown in Fig. 11 is obtained from $\text{MO-ROR-G}^{\mathbf{A}}_1$ by replacing $S \leftarrow \rho(B) \oplus L$ with $S \twoheadleftarrow \Sigma^\ell$ in **ChalEnc**. In $\text{MO-ROR-G}^{\mathbf{A}}_2$, as long as no collision is found for L and for B, S is chosen uniformly at random and independently of (C, B) in **ChalEnc**. Let $\dot{\mathbf{A}}$ be an adversary against EC for strong receiver binding. $\dot{\mathbf{A}}$ runs $\text{MO-ROR-G}^{\mathbf{A}}_1$ and finally outputs $((L, A, M), (L', A', M'), B)$. Then,

$$\begin{aligned}&\left|\Pr[\text{MO-ROR-G}^{\mathbf{A}}_1 = 1] - \Pr[\text{MO-ROR-G}^{\mathbf{A}}_2 = 1]\right| \\ &\quad \le \mathrm{Adv}^{\text{sr-bind}}_{\mathsf{EC}}(\dot{\mathbf{A}}) + (q_{\mathrm{e}} + q_{\mathrm{c}})^2/2^{\ell+1}.\end{aligned}$$

The run time of $\dot{\mathbf{A}}$ is at most about that of $\text{MO-REAL}^{\mathbf{A}}_{\mathsf{ECP}}$.

The game $\text{MO-ROR-G}^{\mathbf{A}}_3$ shown in Fig. 12 is obtained from $\text{MO-ROR-G}^{\mathbf{A}}_2$ by replacing $(C, B) \leftarrow \mathsf{ECenc}(L, A, M)$ with $(C, B) \twoheadleftarrow \Sigma^{\mathsf{clen}(|M|)} \times \Sigma^\tau$ in **ChalEnc**. For transformation from $\text{MO-ROR-G}^{\mathbf{A}}_2$ to $\text{MO-ROR-G}^{\mathbf{A}}_3$, let us consider the game $\text{MO-HYB}^{\mathbf{A}}_k$ shown in Fig. 13, where $k \in [0, q_{\mathrm{c}}]$. $\text{MO-HYB}^{\mathbf{A}}_k$ is different from $\text{MO-ROR-G}^{\mathbf{A}}_2$ only for

ChalEnc. Then,

$$\begin{aligned}&\left|\Pr[\text{MO-ROR-G}_2^{\mathbf{A}} = 1] - \Pr[\text{MO-ROR-G}_3^{\mathbf{A}} = 1]\right| \\ &= \left|\Pr[\text{MO-HYB}_0^{\mathbf{A}} = 1] - \Pr[\text{MO-HYB}_{q_\mathrm{c}}^{\mathbf{A}} = 1]\right| \\ &\leq \sum_{l=1}^{q_\mathrm{c}} \left|\Pr[\text{MO-HYB}_{l-1}^{\mathbf{A}} = 1] - \Pr[\text{MO-HYB}_l^{\mathbf{A}} = 1]\right|.\end{aligned}$$

Let $\mathbf{A}'_l$ be an adversary against EC for confidentiality, where $l \in [1, q_\mathrm{c}]$. $\mathbf{A}'_l$ simulates $\text{MO-HYB}_l^{\mathbf{A}}$ except for the l-th query to **ChalEnc** made by $\mathbf{A}$. $\mathbf{A}'_l$ forwards it to its ECenc oracle and returns the reply to $\mathbf{A}$. Finally, $\mathbf{A}'_l$ produces the same output as $\mathbf{A}$. Then,

$$\begin{aligned}&\left|\Pr[\text{MO-HYB}_{l-1}^{\mathbf{A}} = 1] - \Pr[\text{MO-HYB}_l^{\mathbf{A}} = 1]\right| \\ &= \left|\Pr[\text{otREAL}_{\mathsf{EC}}^{\mathbf{A}'_l} = 1] - \Pr[\text{otRAND}_{\mathsf{EC}}^{\mathbf{A}'_l} = 1]\right| \\ &= \mathrm{Adv}_{\mathsf{EC}}^{\text{ot-ror}}(\mathbf{A}'_l).\end{aligned}$$

There exists some adversary $\ddot{\mathbf{A}}$ such that $\mathrm{Adv}_{\mathsf{EC}}^{\text{ot-ror}}(\mathbf{A}'_l) \leq \mathrm{Adv}_{\mathsf{EC}}^{\text{ot-ror}}(\ddot{\mathbf{A}})$ for every $l \in [1, q_\mathrm{c}]$ and its run time is at most about that of $\text{MO-REAL}_{\mathsf{ECP}}^{\mathbf{A}}$. Thus,

$$\begin{aligned}&\left|\Pr[\text{MO-ROR-G}_2^{\mathbf{A}} = 1] - \Pr[\text{MO-ROR-G}_3^{\mathbf{A}} = 1]\right| \\ &\leq q_\mathrm{c} \cdot \mathrm{Adv}_{\mathsf{EC}}^{\text{ot-ror}}(\ddot{\mathbf{A}}).\end{aligned}$$

For transformation from $\text{MO-ROR-G}_3^{\mathbf{A}}$ to $\text{MO-RAND}_{\mathsf{ECP}}^{\mathbf{A}}$, similarly to the transformation from $\text{MO-REAL}_{\mathsf{ECP}}^{\mathbf{A}}$ to $\text{MO-ROR-G}_1^{\mathbf{A}}$, there exists some adversary $\mathbf{A}_2$ such that

$$\begin{aligned}\mathrm{Adv}_{\mathsf{F}}^{\text{prf}}(\mathbf{A}_2) &= \left|\Pr[\mathbf{A}_2^{\mathsf{F}_K} = 1] - \Pr[\mathbf{A}_2^{\rho} = 1]\right| \\ &= \left|\Pr[\text{MO-RAND}_{\mathsf{ECP}}^{\mathbf{A}} = 1] - \Pr[\text{MO-ROR-G}_3^{\mathbf{A}} = 1]\right|.\end{aligned}$$

The run time of $\mathbf{A}_2$ is at most about that of $\text{MO-REAL}_{\mathsf{ECP}}^{\mathbf{A}}$. $\mathbf{A}_2$ makes at most q_e queries to its oracle. Thus, there exists some adversary $\dddot{\mathbf{A}}$ such that

$$\mathrm{Adv}_{\mathsf{F}}^{\text{prf}}(\dddot{\mathbf{A}}) \geq \max\{\mathrm{Adv}_{\mathsf{F}}^{\text{prf}}(\mathbf{A}_1), \mathrm{Adv}_{\mathsf{F}}^{\text{prf}}(\mathbf{A}_2)\}.$$

The run time of $\dddot{\mathbf{A}}$ is at most about that of $\text{MO-REAL}_{\mathsf{ECP}}^{\mathbf{A}}$. $\dddot{\mathbf{A}}$ makes at most $(q_\mathrm{e} + q_\mathrm{c})$ queries to its oracle. □

Confidentiality of nonce-based ECP is also reduced to confidentiality and strong receiver binding of EC and the PRF property of F. The proof is omitted since it is very similar to that of Theorem 1.

Corollary 1 For any adversary $\mathbf{A}$ against nECP making at most q_e and q_c queries to **Enc** and **ChalEnc**, respectively, there exist adversaries $\dot{\mathbf{A}}$, $\ddot{\mathbf{A}}$, and $\dddot{\mathbf{A}}$ such that

$$\begin{aligned}\mathrm{Adv}_{\mathsf{nECP}}^{\text{mo-ror}}(\mathbf{A}) \leq \mathrm{Adv}_{\mathsf{EC}}^{\text{sr-bind}}(\dot{\mathbf{A}}) + q_\mathrm{c} \cdot \mathrm{Adv}_{\mathsf{EC}}^{\text{ot-ror}}(\ddot{\mathbf{A}}) +& \\ 2 \cdot \mathrm{Adv}_{\mathsf{F}}^{\text{prf}}(\dddot{\mathbf{A}}) + (q_\mathrm{e} + q_\mathrm{c})^2/2^{\ell}&.\end{aligned}$$

The run time of $\dot{\mathbf{A}}$, $\ddot{\mathbf{A}}$, and $\dddot{\mathbf{A}}$ is at most about that of $\text{MO-REAL}_{\mathsf{nECP}}^{\mathbf{A}}$. $\dddot{\mathbf{A}}$ makes at most $2(q_\mathrm{e} + q_\mathrm{c})$ queries to its oracle.

```
K ↞ Σ^n; 𝒴 ← ∅
b ← A^{Enc,Dec,ChalEnc}
return b

Enc(A, M)
L ↞ Σ^ℓ
(C, B) ← ECenc(L, A, M)
S ← F_K(B) ⊕ L
𝒴 ← 𝒴 ∪ {(A, C, B, S)}
R[A, C, B, S] ← (M, L)
return (C, B, S)

Dec(A, C, B, S)
if (A, C, B, S) ∉ 𝒴 then
    return ⊥
end if
return R[A, C, B, S]

ChalEnc(A, M)
L ↞ Σ^ℓ
(C, B) ← ECenc(L, A, M)
S ← F_K(B) ⊕ L
return (C, B, S)
```

(a) $\text{MO-REAL}_{\mathsf{ECP}}^{\mathbf{A}}$

```
K ↞ Σ^n; 𝒴 ← ∅
b ← A^{Enc,Dec,ChalEnc}
return b

Enc(A, M)
L ↞ Σ^ℓ
(C, B) ← ECenc(L, A, M)
S ← F_K(B) ⊕ L
𝒴 ← 𝒴 ∪ {(A, C, B, S)}
R[A, C, B, S] ← (M, L)
return (C, B, S)

Dec(A, C, B, S)
if (A, C, B, S) ∉ 𝒴 then
    return ⊥
end if
return R[A, C, B, S]

ChalEnc(A, M)

(C, B) ↞ Σ^{clen(|M|)} × Σ^τ
S ↞ Σ^ℓ
return (C, B, S)
```

(b) $\text{MO-RAND}_{\mathsf{ECP}}^{\mathbf{A}}$

Fig. 9: Games for confidentiality of ECP

```
K ↞ Σ^n; 𝒴 ← ∅
b ← A^{Enc,Dec,ChalEnc}
return b

Enc(A, M)
L ↞ Σ^ℓ
(C, B) ← ECenc(L, A, M)
S ← ρ(B) ⊕ L
𝒴 ← 𝒴 ∪ {(A, C, B, S)}
R[A, C, B, S] ← (M, L)
return (C, B, S)

Dec(A, C, B)
if (A, C, B, S) ∉ 𝒴 then
    return ⊥
end if
return R[A, C, B, S]

ChalEnc(A, M)
L ↞ Σ^ℓ
(C, B) ← ECenc(L, A, M)
S ← ρ(B) ⊕ L
return (C, B, S)
```

Fig. 10: $\text{MO-ROR-G}_1^{\mathbf{A}}$

```
K ↞ Σ^n; 𝒴 ← ∅
b ← A^{Enc,Dec,ChalEnc}
return b

Enc(A, M)
L ↞ Σ^ℓ
(C, B) ← ECenc(L, A, M)
S ← ρ(B) ⊕ L
𝒴 ← 𝒴 ∪ {(A, C, B, S)}
R[A, C, B, S] ← (M, L)
return (C, B, S)

Dec(A, C, B)
if (A, C, B, S) ∉ 𝒴 then
    return ⊥
end if
return R[A, C, B, S]

ChalEnc(A, M)
L ↞ Σ^ℓ
(C, B) ← ECenc(L, A, M)
S ↞ Σ^ℓ
return (C, B, S)
```

Fig. 11: $\text{MO-ROR-G}_2^{\mathbf{A}}$

3.2.2 Ciphertext Integrity

Ciphertext integrity of ECP is reduced to targeted-ciphertext unforgeability and strong receiver binding of EC, and the PRF property of F:

Theorem 2 For any adversary $\mathbf{A}$ against ECP making at most q_e, q_d, and q_c queries to **Enc**, **Dec**, and **ChalDec**, respectively, there exist adversaries $\dot{\mathbf{A}}$, $\ddot{\mathbf{A}}$, and $\dddot{\mathbf{A}}$ such that

$$\mathrm{Adv}_{\mathsf{ECP}}^{\text{mo-ctxt}}(\mathbf{A}) \leq \mathrm{Adv}_{\mathsf{F}}^{\text{prf}}(\dot{\mathbf{A}}) + \mathrm{Adv}_{\mathsf{EC}}^{\text{sr-bind}}(\ddot{\mathbf{A}}) + q_\mathrm{c} \cdot \mathrm{Adv}_{\mathsf{EC}}^{\text{tcu}}(\dddot{\mathbf{A}}).$$

The run time of $\dot{\mathbf{A}}$, $\ddot{\mathbf{A}}$, and $\dddot{\mathbf{A}}$ is at most about that of $\text{MO-CTXT}_{\mathsf{ECP}}^{\mathbf{A}}$. $\dot{\mathbf{A}}$ makes at most $(q_\mathrm{e} + q_\mathrm{c})$ queries to its oracle.

Proof The game $\text{MO-CTXT}_{\mathsf{ECP}}^{\mathbf{A}}$ is shown in Fig. 14.

```
K ↞ Σ^n; 𝒴 ← ∅
b ← A^{Enc,Dec,ChalEnc}
return b

Enc(A, M)
L ↞ Σ^ℓ
(C, B) ← ECenc(L, A, M)
S ← ρ(B) ⊕ L
𝒴 ← 𝒴 ∪ {(A, C, B, S)}
R[A, C, B, S] ← (M, L)
return (C, B, S)

Dec(A, C, B)
if (A, C, B, S) ∉ 𝒴 then
    return ⊥
end if
return R[A, C, B, S]

ChalEnc(A, M)
(C, B) ↞ Σ^{clen(|M|)} × Σ^τ
S ↞ Σ^ℓ
return (C, B, S)
```

Fig. 12: MO-ROR-$G_3^{\mathbf{A}}$

```
K ↞ Σ^n; 𝒴 ← ∅; ctr ← 0
b ← A^{Enc,Dec,ChalEnc}
return b

Enc(A, M)
L ↞ Σ^ℓ
(C, B) ← ECenc(L, A, M)
S ← ρ(B) ⊕ L
𝒴 ← 𝒴 ∪ {(A, C, B, S)}
R[A, C, B, S] ← (M, L)
return (C, B, S)

Dec(A, C, B)
if (A, C, B, S) ∉ 𝒴 then
    return ⊥
end if
return R[A, C, B, S]

ChalEnc(A, M)
ctr ← ctr + 1
if ctr > k then
    L ↞ Σ^ℓ
    (C, B) ← ECenc(L, A, M)
else
    (C, B) ↞ Σ^{clen(|M|)} × Σ^τ
end if
S ↞ Σ^ℓ
return (C, B, S)
```

Fig. 13: MO-$\mathrm{HYB}_k^{\mathbf{A}}$

$$\mathrm{Adv}_{\mathsf{ECP}}^{\text{mo-ctxt}}(\mathbf{A}) = \Pr[\text{MO-CTXT}_{\mathsf{ECP}}^{\mathbf{A}} = 1].$$

The game MO-CTXT-$G_1^{\mathbf{A}}$ shown in Fig. 15 is obtained from MO-CTXT$_{\mathsf{ECP}}^{\mathbf{A}}$ by replacing F_K with a uniform random function ρ. Let $\dot{\mathbf{A}}$ be an adversary against F. $\dot{\mathbf{A}}$ is given either F_K or ρ as an oracle. $\dot{\mathbf{A}}$ simulates MO-CTXT$_{\mathsf{ECP}}^{\mathbf{A}}$ and MO-CTXT-$G_1^{\mathbf{A}}$ by making use of F_K and ρ, respectively. Then,

$$\begin{aligned}\mathrm{Adv}_{\mathsf{F}}^{\mathrm{prf}}(\dot{\mathbf{A}}) &= \left|\Pr[\dot{\mathbf{A}}^{\mathsf{F}_K} = 1] - \Pr[\dot{\mathbf{A}}^{\rho} = 1]\right| \\ &= \left|\Pr[\text{MO-CTXT}_{\mathsf{ECP}}^{\mathbf{A}} = 1] - \Pr[\text{MO-CTXT-G}_1^{\mathbf{A}} = 1]\right|.\end{aligned}$$

$\dot{\mathbf{A}}$ makes at most $(q_{\mathrm{e}} + q_{\mathrm{c}})$ queries to its oracle. The run time of $\dot{\mathbf{A}}$ is at most about that of MO-CTXT$_{\mathsf{ECP}}^{\mathbf{A}}$.

In MO-CTXT-$G_1^{\mathbf{A}}$, suppose that a query (A^*, C^*, B^*, S^*) sets *win* 1. Then, there are two cases: (1) There exists some $(A, C, B, S) \in \mathcal{Y}$ such that $B = B^*$ and $(A, C, S) \neq (A^*, C^*, S^*)$, and (2) there exists no $(A, C, B, S) \in \mathcal{Y}$ such that $B = B^*$.

For the first case, let $\ddot{\mathbf{A}}$ be an adversary against EC for strong receiver binding. $\ddot{\mathbf{A}}$ first simply runs MO-CTXT-$G_1^{\mathbf{A}}$. Let $(A', C', B', S') \in \mathcal{Y}$ be a tuple such that $B' = B^*$ and $(A', C', S') \neq (A^*, C^*, S^*)$. Let (M', L') be a tuple such that $\mathsf{ECenc}(L', A', M') = (C', B')$. Let (M^*, L^*) be returned by **ChalDec** in response to (A^*, C^*, B^*, S^*). Then, $\ddot{\mathbf{A}}$ terminates with the output $((L', A', M'), (L^*, A^*, M^*), B^*)$. Let us see that $\ddot{\mathbf{A}}$ is successful. Since EC is (strongly) correct, $\mathsf{ECenc}(L^*, A^*, M^*) = (C^*, B^*)$, and $\mathsf{ECver}(A^*, M^*, L^*, B^*) = 1$. It is easy to see that $(L', A', M') \neq (L^*, A^*, M^*)$.

For the second case, let $\dddot{\mathbf{A}} := (\dddot{\mathbf{A}}_1, \dddot{\mathbf{A}}_2)$ be an adversary against EC for targeted-ciphertext unforgeability. $\dddot{\mathbf{A}}_1$ first selects $r \in [1, q_{\mathrm{c}}]$ uniformly at random. Then, $\dddot{\mathbf{A}}_1$ simulates MO-CTXT-$G_1^{\mathbf{A}}$ except that $\dddot{\mathbf{A}}_1$ returns $\perp$ to the i-th query to **ChalDec** made by $\mathbf{A}$ for every $i < r$. For the r-th query (A'', C'', B'', S'') to **ChalDec** made by $\mathbf{A}$, $\dddot{\mathbf{A}}_1$ terminates the simulation and outputs (B'', state), where *state* is some state information including (A'', C''). Then, $\dddot{\mathbf{A}}_2$ takes (B'', state) and L'' as input, where $L'' \twoheadleftarrow \Sigma^\ell$, and outputs (A'', C''). $\dddot{\mathbf{A}}$ is successful if $(A'', C'', B'', S'') = (A^*, C^*, B^*, S^*)$. □

```
K ↞ Σ^n; 𝒴 ← ∅
win ← 0
A^{Enc,Dec,ChalDec}
return win

Enc(A, M)
L ↞ Σ^ℓ
(C, B) ← ECenc(L, A, M)
S ← F_K(B) ⊕ L
𝒴 ← 𝒴 ∪ {(A, C, B, S)}
R[A, C, B, S] ← (M, L)
return (C, B, S)

Dec(A, C, B, S)          ▹ (A, C, B, S) ∈ 𝒴
return R[A, C, B, S]

ChalDec(A, C, B, S)      ▹ (A, C, B, S) ∉ 𝒴
L ← F_K(B) ⊕ S
M ← ECdec(L, A, C, B)
if M = ⊥ then
    return ⊥
end if
win ← 1
return (M, L)
```

Fig. 14: Game MO-CTXT$_{\mathsf{ECP}}^{\mathbf{A}}$

```
K ↞ Σ^n; 𝒴 ← ∅
win ← 0
A^{Enc,Dec,ChalDec}
return win

Enc(A, M)
L ↞ Σ^ℓ
(C, B) ← ECenc(L, A, M)
S ← ρ(B) ⊕ L
𝒴 ← 𝒴 ∪ {(A, C, B, S)}
R[A, C, B, S] ← (M, L)
return (C, B, S)

Dec(A, C, B, S)          ▹ (A, C, B, S) ∈ 𝒴
return R[A, C, B, S]

ChalDec(A, C, B, S)      ▹ (A, C, B, S) ∉ 𝒴
L ← ρ(B) ⊕ S
M ← ECdec(L, A, C, B)
if M = ⊥ then
    return ⊥
end if
win ← 1
return (M, L)
```

Fig. 15: MO-CTXT-$G_1^{\mathbf{A}}$

Ciphertext integrity of nonce-based ECP is also reduced to targeted-ciphertext unforgeability and strong receiver binding of EC, and the PRF property of F. The proof is omitted since it is very similar to that of Theorem 2.

Corollary 2 For any adversary $\mathbf{A}$ against nECP making at most q_{e}, q_{d}, and q_{c} queries to **Enc**, **Dec**, and **ChalDec**, respectively, there exist adversaries $\dot{\mathbf{A}}$, $\ddot{\mathbf{A}}$, and $\dddot{\mathbf{A}}$ such that

$$\mathrm{Adv}_{\mathsf{nECP}}^{\text{mo-ctxt}}(\mathbf{A}) \leq \mathrm{Adv}_{\mathsf{F}}^{\mathrm{prf}}(\dot{\mathbf{A}}) + \mathrm{Adv}_{\mathsf{EC}}^{\text{sr-bind}}(\ddot{\mathbf{A}}) + q_{\mathrm{c}} \cdot \mathrm{Adv}_{\mathsf{EC}}^{\mathrm{tcu}}(\dddot{\mathbf{A}}).$$

The run time of $\dot{\mathbf{A}}$, $\ddot{\mathbf{A}}$, and $\dddot{\mathbf{A}}$ is at most about that of MO-CTXT$_{\mathsf{nECP}}^{\mathbf{A}}$. $\dot{\mathbf{A}}$ makes at most $2(q_{\mathrm{e}} + q_{\mathrm{c}})$ queries to its oracle.

3.2.3 Binding Properties

ECP and nECP inherit (strong) receiver binding from EC since VER simply runs ECver:

Theorem 3 For any adversary $\mathbf{A}$ against ECP for (strong) receiver binding, there exists an adversary $\dot{\mathbf{A}}$ such that

$\mathrm{Adv}^{\mathrm{(s)r\text{-}bind}}_{\mathsf{ECP}}(\mathbf{A}) \leq \mathrm{Adv}^{\mathrm{(s)r\text{-}bind}}_{\mathsf{EC}}(\dot{\mathbf{A}})$. The run time of $\dot{\mathbf{A}}$ is at most about that of $\mathbf{A}$.

Sender binding of ECP and nECP is implied by strong correctness of EC. Suppose that (K, A, C, B, S) satisfies $\mathsf{DEC}(K, A, C, B, S) = (M, L) \neq \perp$. Then, $\mathsf{ECdec}(L, A, C, B) = M$. Since EC is strongly correct, $\mathsf{ECenc}(L, A, M) = (C, B)$ and $\mathsf{ECver}(A, M, L, B) = 1 = \mathsf{VER}(A, M, L, B)$.

Theorem 4 Suppose that the underlying EC of ECP and nECP satisfies strong correctness. Then, for any adversary $\mathbf{A}$, $\mathrm{Adv}^{\mathrm{s\text{-}bind}}_{\mathsf{ECP}}(\mathbf{A}) = 0$ and $\mathrm{Adv}^{\mathrm{s\text{-}bind}}_{\mathsf{nECP}}(\mathbf{A}) = 0$.

4. Conclusion

We have presented a transform to build randomized ccAEAD from encryptment, requiring only a single call to a PRF. We have also extended it to build nonce-based ccAEAD, which requires two calls to a PRF. Then, we have reduced the security of the resultant ccAEAD to the security of underlying encryptment and a PRF.

Future work is to study generic ccAEAD construction simpler than CtE and CEP. It is also interesting to explore applications ccAEAD is useful for.

Acknowledgments

The authors thank Akiko Inoue for fruitful discussions. The first author was supported by JSPS KAKENHI Grant Number 21K11885 and 24K14944.

References

[1] Facebook, “Facebook messenger.” https://www.messenger.com. Accessed on 20/02/2024.
[2] Signal Foundation, “Signal.” https://signal.org/. Accessed on 20/02/2024.
[3] WhatsApp, “WhatsApp Messenger.” https://www.whatsapp.com. Accessed on 20/02/2024.
[4] Facebook, “Messenger secret conversations.” Technical Whitepaper, 2016. https://about.fb.com/wp-content/uploads/2016/07/messenger-secret-conversations-technical-whitepaper.pdf.
[5] P. Grubbs, J. Lu, and T. Ristenpart, “Message franking via committing authenticated encryption,” Advances in Cryptology - CRYPTO 2017 - 37th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 20-24, 2017, Proceedings, Part III, ed. J. Katz and H. Shacham, Lecture Notes in Computer Science, vol.10403, pp.66–97, Springer, 2017.
[6] P. Rogaway, “Authenticated-encryption with associated-data,” Proceedings of the 9th ACM Conference on Computer and Communications Security, CCS 2002, Washington, DC, USA, November 18-22, 2002, ed. V. Atluri, pp.98–107, ACM, 2002.
[7] Y. Dodis, P. Grubbs, T. Ristenpart, and J. Woodage, “Fast message franking: From invisible salamanders to encryptment,” Advances in Cryptology - CRYPTO 2018 - 38th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 19-23, 2018, Proceedings, Part I, ed. H. Shacham and A. Boldyreva, Lecture Notes in Computer Science, vol.10991, pp.155–186, Springer, 2018.
[8] Y. Dodis, P. Grubbs, T. Ristenpart, and J. Woodage, “Fast message franking: From invisible salamanders to encryptment.” Cryptology ePrint Archive, Paper 2019/016, 2019. https://eprint.iacr.org/2019/016.
[9] I. Damgård, “A design principle for hash functions,” Advances in Cryptology - CRYPTO ’89, 9th Annual International Cryptology Conference, Santa Barbara, California, USA, August 20-24, 1989, Proceedings, ed. G. Brassard, Lecture Notes in Computer Science, vol.435, pp.416–427, Springer, 1989.
[10] R.C. Merkle, “One way hash functions and DES,” Advances in Cryptology - CRYPTO ’89, 9th Annual International Cryptology Conference, Santa Barbara, California, USA, August 20-24, 1989, Proceedings, ed. G. Brassard, Lecture Notes in Computer Science, vol.435, pp.428–446, Springer, 1989.
[11] S. Hirose and K. Minematsu, “Compactly committing authenticated encryption using encryptment and tweakable block cipher.” Cryptology ePrint Archive, Paper 2022/1670, 2022. https://eprint.iacr.org/2022/1670.
[12] S. Hirose and K. Minematsu, “Compactly committing authenticated encryption using encryptment and tweakable block cipher,” Selected Areas in Cryptography - SAC 2023, 30th International Conference, Fredericton, Canada, August 14–18, 2023, Revised Selected Papers, ed. C. Carlet, K. Mandal, and V. Rijmen, Lecture Notes in Computer Science, vol.14201, pp.233–252, Springer, 2023.
[13] M.D. Liskov, R.L. Rivest, and D.A. Wagner, “Tweakable block ciphers,” Advances in Cryptology - CRYPTO 2002, 22nd Annual International Cryptology Conference, Santa Barbara, California, USA, August 18-22, 2002, Proceedings, ed. M. Yung, Lecture Notes in Computer Science, vol.2442, pp.31–46, Springer, 2002.
[14] M.D. Liskov, R.L. Rivest, and D.A. Wagner, “Tweakable block ciphers,” Journal of Cryptology, vol.24, no.3, pp.588–613, 2011.
[15] J. Katz and M. Yung, “Complete characterization of security notions for probabilistic private-key encryption,” Proceedings of the Thirty-Second Annual ACM Symposium on Theory of Computing, pp.245–254, 2000.
[16] M. Bellare and C. Namprempre, “Authenticated encryption: Relations among notions and analysis of the generic composition paradigm,” Advances in Cryptology - ASIACRYPT 2000, 6th International Conference on the Theory and Application of Cryptology and Information Security, Kyoto, Japan, December 3-7, 2000, Proceedings, ed. T. Okamoto, Lecture Notes in Computer Science, vol.1976, pp.531–545, Springer, 2000.
[17] I. Leontiadis and S. Vaudenay, “Private message franking with after opening privacy.” Cryptology ePrint Archive, Report 2018/938, 2018. https://eprint.iacr.org/2018/938.
[18] I. Leontiadis and S. Vaudenay, “Private message franking with after opening privacy,” Information and Communications Security - 25th International Conference, ICICS 2023, Tianjin, China, November 18-20, 2023, Proceedings, ed. D. Wang, M. Yung, Z. Liu, and X. Chen, Lecture Notes in Computer Science, vol.14252, pp.197–214, Springer, 2023.
[19] L. Chen and Q. Tang, “People who live in glass houses should not throw stones: Targeted opening message franking schemes.” Cryptology ePrint Archive, Report 2018/994, 2018. https://eprint.iacr.org/2018/994.
[20] L. Huguenin-Dumittan and I. Leontiadis, “A message franking channel,” Information Security and Cryptology - 17th International Conference, Inscrypt 2021, Virtual Event, August 12-14, 2021, Revised Selected Papers, ed. Y. Yu and M. Yung, Lecture Notes in Computer Science, vol.13007, pp.111–128, Springer, 2021.
[21] H. Yamamuro, K. Hara, M. Tezuka, Y. Yoshida, and K. Tanaka, “Forward secure message franking,” Information Security and Cryptology - ICISC 2021 - 24th International Conference, Seoul, South Korea, December 1-3, 2021, Revised Selected Papers, ed. J.H. Park and S. Seo, Lecture Notes in Computer Science, vol.13218, pp.339–358, Springer, 2021.
[22] H. Yamamuro, K. Hara, M. Tezuka, Y. Yoshida, and K. Tanaka, “Forward secure message franking with updatable reporting tags,” IEICE Trans. Fundam. Electron. Commun. Comput. Sci., vol.106,

no.9, pp.1164–1176, 2023.

[23] N. Tyagi, P. Grubbs, J. Len, I. Miers, and T. Ristenpart, "Asymmetric message franking: Content moderation for metadata-private end-to-end encryption," Advances in Cryptology - CRYPTO 2019 - 39th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 18-22, 2019, Proceedings, Part III, ed. A. Boldyreva and D. Micciancio, Lecture Notes in Computer Science, vol.11694, pp.222–250, Springer, 2019.

[24] Q. Huang, G. Yang, D.S. Wong, and W. Susilo, "Efficient strong designated verifier signature schemes without random oracle or with non-delegatability," International Journal of Information Security, vol.10, no.6, pp.373–385, 2011.

[25] M. Jakobsson, K. Sako, and R. Impagliazzo, "Designated verifier proofs and their applications," Advances in Cryptology - EUROCRYPT '96, International Conference on the Theory and Application of Cryptographic Techniques, Saragossa, Spain, May 12-16, 1996, Proceeding, ed. U.M. Maurer, Lecture Notes in Computer Science, vol.1070, pp.143–154, Springer, 1996.

[26] S. Hirose, "Compactly committing authenticated encryption using tweakable block cipher," Network and System Security - 14th International Conference, NSS 2020, Melbourne, VIC, Australia, November 25-27, 2020, Proceedings, ed. M. Kutylowski, J. Zhang, and C. Chen, Lecture Notes in Computer Science, vol.12570, pp.187–206, Springer, 2020.

[27] P. Farshim, C. Orlandi, and R. Rosie, "Security of symmetric primitives under incorrect usage of keys," IACR Transactions on Symmetric Cryptology, vol.2017, no.1, pp.449–473, 2017.

[28] A. Albertini, T. Duong, S. Gueron, S. Kölbl, A. Luykx, and S. Schmieg, "How to abuse and fix authenticated encryption without key commitment," 31st USENIX Security Symposium, USENIX Security 2022, Boston, MA, USA, August 10-12, 2022, ed. K.R.B. Butler and K. Thomas, pp.3291–3308, USENIX Association, 2022.

[29] J. Len, P. Grubbs, and T. Ristenpart, "Partitioning oracle attacks," 30th USENIX Security Symposium, USENIX Security 2021, August 11-13, 2021, ed. M. Bailey and R. Greenstadt, pp.195–212, USENIX Association, 2021.

[30] M. Bellare and V.T. Hoang, "Efficient schemes for committing authenticated encryption," Advances in Cryptology - EUROCRYPT 2022 - 41st Annual International Conference on the Theory and Applications of Cryptographic Techniques, Trondheim, Norway, May 30 - June 3, 2022, Proceedings, Part II, ed. O. Dunkelman and S. Dziembowski, Lecture Notes in Computer Science, vol.13276, pp.845–875, Springer, 2022.

[31] J. Chan and P. Rogaway, "On committing authenticated-encryption," Computer Security - ESORICS 2022 - 27th European Symposium on Research in Computer Security, Copenhagen, Denmark, September 26-30, 2022, Proceedings, Part II, ed. V. Atluri, R.D. Pietro, C.D. Jensen, and W. Meng, Lecture Notes in Computer Science, vol.13555, pp.275–294, Springer, 2022.

[32] P. Rogaway and T. Shrimpton, "Cryptographic hash-function basics: Definitions, implications, and separations for preimage resistance, second-preimage resistance, and collision resistance," Fast Software Encryption, 11th International Workshop, FSE 2004, Delhi, India, February 5-7, 2004, Revised Papers, ed. B.K. Roy and W. Meier, Lecture Notes in Computer Science, vol.3017, pp.371–388, Springer, 2004.

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 research interests include cryptography and information security. He received Young Engineer Award from IEICE in 1997, and KDDI Foundation Research Award in 2008.

Kazuhiko Minematsu received B.E., M.E., and Dr.S degrees from Waseda University in 1996, 1998, and 2008. He joined NEC since 1998 and is a research fellow since 2020. His research interests are design and analysis of symmetric-key ciphers and its application systems. Since 2019, he is also a visiting professor at Yokohama National University. He is a co-author of the best paper awards at FSE 2015 and CRYPTO 2019, and received SCAT award at 2020.