Improved XKX-based AEAD Scheme: Removing the Birthday Terms

Yusuke Naito

Mitsubishi Electric Corporation, Kanagawa, Japan
Naito.Yusuke@ce.MitsubishiElectric.co.jp

Abstract. Recently, Naito [ToSC 2017, Issue 2] proposed XKX, a tweakable blockcipher (TBC) based on a blockcipher (BC). It offers efficient authenticated encryption with associated data (AEAD) schemes with beyond-birthday-bound (BBB) security, by combining with efficient TBC-based AEAD schemes such as ΘCB3. In the resultant schemes, for each data block, a BC is called once. The security bound is roughly $\ell^2 q/2^n + \sigma_A^2/2^n + \sigma_{\mathcal{D}}^2/2^n$, where n is the block size of the BC in bits, ℓ is the number of BC calls by a query, q is the number of queries, σ_A is the number of BC calls handing associated data by encryption queries, and $\sigma_{\mathcal{D}}$ is the number of BC calls by decryption queries. Hence, assuming $\ell, \sigma_A, \sigma_{\mathcal{D}} \ll 2^{n/2}$, the AEAD schemes achieve BBB security. However, the birthday terms $\sigma_A^2/2^n$, $\sigma_{\mathcal{D}}^2/2^n$ might become dominant, for example, when n is small such as $n = 64$ and when DoS attacks are performed. The birthday terms are introduced due to the modular proof via the XKX's security proof.
In this paper, in order to remove the birthday terms, we slightly modify ΘCB3 called ΘCB3†, and directly prove the security of ΘCB3† with XKX. We show that the security bound becomes roughly $\ell^2 q/2^n$.

Keywords: Blockcipher, tweakable blockcipher, efficient authenticated encryption, beyond-birthday-bound security

1 Introduction

Background.[1] Confidentiality and authenticity of data are the most important properties to securely communicate over an insecure channel. In the symmetric-key setting, an authenticated encryption with associated data (AEAD) scheme ensures jointly these properties. AEAD schemes have been mainly designed from a blockcipher (BC). In AEAD research, designing an efficient AEAD scheme is a main theme. In efficient AEAD schemes such as OCB schemes [26, 24, 25, 13] and OTR [20], a BC is called once for each data block[2] (for associated data or a plaintext).

[1] Our result is an extension of the result in [21], and thus several parts of the background are reused from [21].
[2] The data block is equal to the block size of the underlying BC.

Efficient BC-based AEAD schemes have been designed by incorporating an efficient BC-based TBC into an efficient tweakable-BC(TBC)-based AEAD scheme: in efficient TBC-based AEAD schemes such as ΘCB3 [13] and $\mathbb{OTR}$ [20], a TBC is called once for each data block; in efficient BC-based TBCs such as `LRW2`-type TBCs [16, 25, 13], a BC is called once for each query. Since the efficient BC-based TBCs have birthday-bound security, i.e., security up to $2^{n/2}$ BC calls, so are the combined schemes, where n is the block size in bits. However, birthday-bound security sometimes becomes unreliable; for example, when a lightweight BC is used, when large amounts of data are processed, or when a large number of connections need to be kept secure. Hence, designing an AEAD scheme with *beyond-birthday-bound* (BBB) security is also important.

Landecker et al. [15] proposed a TBC called Chained `LRW2` (`CLRW2`) with security up to $2^{2n/3}$ BC calls, where `LRW2` is iterated twice. Lampe and Seurin [14] considered a more general scheme called r-`CLRW` with security up to $2^{rn/(r+2)}$ BC calls, where `LRW2` is iterated r times. Using the TBCs, BC-based AEAD schemes with BBB security can be obtained. Iwata [8] proposed an AEAD scheme with security up to $2^{2n/3}$ BC calls. In the default setting of the AEAD scheme, for each 4 data blocks, it requires 6 BC calls, and for each data block, it requires one multiplication. Iwata and Yasuda [11, 12] pointed out that a combination of the xor of BCs [17] and the Feistel network with six rounds [22] offers BBB-secure AEAD schemes. However, the resultant AEAD schemes require 6 BC calls for each data block. Iwata and Minematsu [10] proposed AEAD schemes with security up to $2^{rn/(r+1)}$ BC calls, where for each data block, a BC is called r times, and a tag is generated by using r almost XOR universal hash functions. These AEAD schemes have BBB security but are not efficient.

Recently, Naito [21] proposed `XKX`, a BC-based TBC that offers efficient nonce-based AEAD schemes with BBB security, by combining with ΘCB3 or $\mathbb{OTR}$. `XKX` is a combination of Minematsu's TBC `Min` [19] and `LRW2`, where a BC's key is defined by using a pseudorandom function (PRF) whose input is a nonce, and then a data block is encrypted by `LRW2` with the nonce dependent key.[3] In `XKX`-based ΘCB3 (or $\mathbb{OTR}$), for each query, after the nonce dependent key is defined, a BC is called once for each data block. The security bounds of the `XKX` based AEAD schemes are roughly $\ell^2 q/2^n + \sigma_A^2/2^n + \sigma_{\mathcal{D}}^2/2^n$, where ℓ is the number of BC calls by a query, q is the number of queries, σ_A is the number of BC calls handing associated data by encryption queries, and $\sigma_{\mathcal{D}}$ is the number of BC calls by decryption queries.[4] Hence, if $\ell, \sigma_A, \sigma_{\mathcal{D}} \ll 2^{n/2}$, the AEAD schemes have BBB security.

---

[3] He gave BC-based instantiations of the PRF; the XOR of BCs and the concatenation. The PRF advantage of the XOR is roughly $q/2^n$. The PRF advantage of the concatenation is roughly $q^2/2^n$. Using these instantiations, these terms are introduced in the security bounds of the `XKX`-based AEAD schemes.

[4] More precisely, (the PRF-security advantage) and $q\times$(the strong pseudo-random permutation advantage) are defined in the security bound. For simplicity, assume that these terms are negligible.

Motivation. The birthday terms $\sigma_A^2/2^n, \sigma_{\mathcal{D}}^2/2^n$ might become dominant, when n is small e.g., $n = 64$. Security bounds define a span of changing a key, and if the threshold is e.g., $1/2^{20}$ (a key is changed when a security bound reaches the threshold), the security bound reaches the threshold when $\sigma_A = 2^{22}$ or $\sigma_{\mathcal{D}} = 2^{22}$, which might cause frequent key updates due to DoS attacks.

The reason why the birthday terms are introduced is the modular proof, which is a combination of the security proofs of ΘCB3 (or $\mathbb{OTR}$) and of `XKX`. In the security bound of `XKX`, the term $\nu^2/2^n$ is defined, where ν is the number of BC calls with the same key. Hence, the birthday term $\sigma_A^2/2^n$ is introduced, since in the AEAD schemes, the same BC's key is used for every associated data block. The birthday term $\sigma_{\mathcal{D}}^2/2^n$ is introduced, since an adversary can make decryption queries with the same nonce (i.e, the corresponding BC's keys are the same).

Instead of the modular proof, the birthday terms might be removed by directly proving the security of the AEAD scheme. However, it might be be hard. In `XKX`-based ΘCB3, the checksum of plaintext blocks is encrypted, associated data is hashed, and the tag is defined by XOR-ing the encrypted checksum with the hash value. For this construction, an adversary can make decryption queries where the encrypted checksums are the same, and thus the randomnesses of the tags depend on the hash values. Since the BC's key to handle associated data (to define hash values) is fixed, the birthday term regarding associated data by decryption queries might remain in the security bound due to the PRF-PRP switch for the BC's outputs.

Our Result. In order to remove the birthday terms, we slightly modify `XKX`-based ΘCB3 called ΘCB3†, and then directly prove the security of ΘCB3†. In this modification, the hash value is XOR-ed with the checksum (instead of the encrypted checksum). Hence, one does not need to consider the randomnesses of hash values. We show that the birthday terms can be removed, that is, the security bound becomes roughly $\ell^2 q/2^n$. Note that in this modification, since one does not need to keep a hash value when generating a tag, the memory size can be reduced by the hash value.

Related Works. Mennink [18] proposed two TBCs with BBB security in the ideal cipher model (ICM). Wang et al. [27] generalized his TBCs and gave 32 TBCs with BBB security in the ICM, where some of the TBCs offer efficient AEAD schemes with BBB security in the ICM. Note that our target scheme is an efficient AEAD scheme with BBB security in the standard model.

Organization. In Section 2, we start by giving notations and security definitions. In Section 3, we give the previous result for `XKX`, where the specifications of `XKX` schemes and the security results are given. In Section 4, we give our result, where the specification of ΘCB3† with `XKX`, the security bounds, and the proofs are given. In Section 5, we give how to realize ΘCB3† with `XKX` from only a BC with respect to the PRF (in `Min`) and the almost XOR universal hash function (in `LRW2`). Finally, in Section 6, we give a conclusion of this paper.

2 Preliminaries

2.1 Notations

$\{0,1\}^*$ denotes the set of all bit strings, and λ denotes the empty string. For a natural integer n, $\{0,1\}^n$ denotes the set of n-bit strings, and 0^n denotes the bit string of n-bit zeroes. We write $[i] := \{1, 2, \ldots, i\}$ for a positive integer i. For a finite set $\mathcal{X}$, $x \xleftarrow{\$} \mathcal{X}$ means that an element is randomly drawn from $\mathcal{X}$ and is assigned to x. For a bit string x and a set $\mathcal{X}$, $|x|$ and $|\mathcal{X}|$ denote the bit length of x and the number of elements in $\mathcal{X}$, respectively. For a bit string x and an integer $i \le |x|$, $[x]^i$ denotes the first i-bit string of x. For a bit string M, $M_1, \ldots, M_m, M_* \xleftarrow{n} M$ means that M is partitioned into n-bit strings $M_1, \ldots, M_m$ and $(|M| - mn)$-bit string M_* such that $|M_*| < n$ and $M = M_1 \| \ldots \| M_m \| M_*$. Let $\mathsf{Perm}(\mathcal{B})$ be the set of all permutations over a non-empty set $\mathcal{B}$. A random permutation over $\mathcal{B}$ is defined as $P \xleftarrow{\$} \mathsf{Perm}(\mathcal{B})$. The inverse is denoted by P^{-1}. For an adversary $\mathbf{A}$ with oracle access to $\mathcal{O}$, its output is denoted by $\mathbf{A}^{\mathcal{O}}$. In this paper, an adversary is a computationally bounded algorithm and the resource is measured in terms of time and query complexities.

2.2 Definitions of (Tweakable) Blockciphers

Blockcipher (BC). A BC $E : \mathcal{K} \times \mathcal{B} \to \mathcal{B}$ is a family of permutations over the set of blocks $\mathcal{B}$ indexed by the set of keys $\mathcal{K}$. $E_K(\cdot)$ denotes the encryption function E having a key $K \in \mathcal{K}$. The decryption function is denoted by E^{-1}, and E_K^{-1} denotes E^{-1} having a key $K \in \mathcal{K}$, and becomes the inverse permutation of E_K. $\mathsf{BC}(\mathcal{K}, \mathcal{B})$ denotes the set of all encryptions of BCs.

We consider Strong-Pseudo-Random Permutation (SPRP) security. The advantage function of an sprp-adversary $\mathbf{A}$ that outputs a bit are defined as

$$\mathbf{Adv}_E^{\mathsf{sprp}}(\mathbf{A}) = \Pr[K \xleftarrow{\$} \mathcal{K}; \mathbf{A}^{E_K, E_K^{-1}} = 1] - \Pr[P \xleftarrow{\$} \mathsf{Perm}(\mathcal{B}); \mathbf{A}^{P, P^{-1}} = 1] ,$$

where the probabilities are taken over $\mathbf{A}$, K and P. We say $\mathbf{A}$ is a (q, t)-sprp-adversary if $\mathbf{A}$ makes q queries and runs in time t.

Tweakable Blockcipher (TBC). A TBC $\widetilde{E} : \mathcal{K} \times \mathcal{TW} \times \mathcal{B} \to \mathcal{B}$ is a family of permutations over the set of blocks $\mathcal{B}$ indexed by the set of keys $\mathcal{K}$ and the set of tweaks $\mathcal{TW}$. $\widetilde{E}_K(tw, \cdot)$ denotes the encryption of $\widetilde{E}$ having a key $K \in \mathcal{K}$ and a tweak $tw \in \mathcal{TW}$. The decryption function is denoted by $\widetilde{E}^{-1}$, and $\widetilde{E}_K^{-1}(tw, \cdot)$ is the inverse permutation of $\widetilde{E}_K(tw, \cdot)$.

We consider Tweakable-Strong-Pseudo-Random Permutation (TSPRP) security. Let $\widetilde{\mathsf{Perm}}(\mathcal{TW}, \mathcal{B})$ be the set of all tweakable permutations with the sets of tweaks $\mathcal{TW}$ and of blocks $\mathcal{B}$, where $\widetilde{P} \in \widetilde{\mathsf{Perm}}(\mathcal{TW}, \mathcal{B})$ is a family of permutations over $\mathcal{B}$ indexed by $\mathcal{TW}$, and a tweakable RP (TRP) is defined as

$\widetilde{P} \xleftarrow{\$} \widetilde{\mathsf{Perm}}(\mathcal{TW}, \mathcal{B})$. The inverse is denoted by $\widetilde{P}^{-1}$. The advantage function of a tsprp-adversary $\mathbf{A}$ that outputs a bit is defined as

$$\mathbf{Adv}^{\widetilde{\mathsf{sprp}}}_{\widetilde{E}}(\mathbf{A}) = \Pr\left[K \xleftarrow{\$} \mathcal{K}; \mathbf{A}^{\widetilde{E}_K, \widetilde{E}_K^{-1}} = 1\right] - \Pr\left[\widetilde{P} \xleftarrow{\$} \widetilde{\mathsf{Perm}}(\mathcal{TW}, \mathcal{B}); \mathbf{A}^{\widetilde{P}, \widetilde{P}^{-1}} = 1\right] ,$$

where the probabilities are taken over $\mathbf{A}$, K and $\widetilde{P}$. We say $\mathbf{A}$ is a (q,t)-tsprp-adversary if $\mathbf{A}$ makes at most q queries and runs in time t.

2.3 Definition of Pseudo-Random Function

Let $\mathsf{Func}(\mathcal{X}, \mathcal{Y})$ be the set of all functions from a set $\mathcal{X}$ to a set $\mathcal{Y}$. Let $\mathcal{F} \subseteq \mathsf{Func}(\mathcal{X}, \mathcal{Y})$ be a family of functions that maps $\mathcal{X}$ to $\mathcal{Y}$. We consider Pseudo-Random-Function (PRF) security of $\mathcal{F}$ that is indistinguishability from a random function (RF), where an RF is defined as $f \xleftarrow{\$} \mathsf{Func}(\mathcal{X}, \mathcal{Y})$. The advantage function of a prf-adversary $\mathbf{A}$ that outputs a bit is defined as

$$\mathbf{Adv}^{\mathsf{prf}}_{\mathcal{F}}(\mathbf{A}) = \Pr[F \xleftarrow{\$} \mathcal{F}; \mathbf{A}^F = 1] - \Pr[f \xleftarrow{\$} \mathsf{Func}(\mathcal{X}, \mathcal{Y}); \mathbf{A}^f = 1] ,$$

where the probabilities are taken over $\mathbf{A}$, F and f. We say $\mathbf{A}$ is a (q,t)-prf-adversary if $\mathbf{A}$ makes at most q queries and runs in time t.

2.4 Definition of Nonce-Based Authenticated Encryption with Associated Data

In this paper, we consider nonce-based authenticated encryption with associated data (nAEAD) schemes. The syntax and the definition of nAEAD schemes are given below.

An nAEAD scheme Π is a pair of encryption and decryption algorithms $\Pi = (\Pi.\mathtt{Enc}, \Pi.\mathtt{Dec})$. $\mathcal{K}, \mathcal{N}, \mathcal{M}, \mathcal{C}, \mathcal{A}$ and $\mathcal{T}$ are the sets of keys, nonces, messages, ciphertexts, associated data and tags of the nAEAD scheme. The encryption algorithm with a key $K \in \mathcal{K}$, $\Pi.\mathtt{Enc}_K$, takes a nonce $N \in \mathcal{N}$, associated data $A \in \mathcal{A}$, and a plaintext $M \in \mathcal{M}$. $\Pi.\mathtt{Enc}_K(N, A, M)$ returns, deterministically, a pair of a ciphertext $C \in \mathcal{C}$ and a tag $T \in \mathcal{T}$. The decryption algorithm with a key $K \in \mathcal{K}$, $\Pi.\mathtt{Dec}_K$, takes a tuple $(N, A, C, T) \in \mathcal{N} \times \mathcal{A} \times \mathcal{C} \times \mathcal{T}$. $\Pi.\mathtt{Dec}_K(N, A, C, T)$ returns, deterministically, either the distinguished invalid symbol $\perp$ or a plaintext $M \in \mathcal{M}$. We require $|\Pi.\mathtt{Enc}_K(N, A, M)| = |\Pi.\mathtt{Enc}_K(N, A, M')|$ when $|M| = |M'|$.

We follow the security definition in [1, 24] that considers privacy and authenticity of an nAEAD scheme Π. The privacy advantage of an adversary $\mathbf{A}$ that outputs a bit is defined as

$$\mathbf{Adv}^{\mathsf{priv}}_{\Pi}(\mathbf{A}) = \Pr[K \xleftarrow{\$} \mathcal{K}; \mathbf{A}^{\Pi.\mathtt{Enc}_K} = 1] - \Pr[\mathbf{A}^{\$} = 1] ,$$

where a random-bits oracle \$ has the same interface as $\Pi.\mathtt{Enc}_K$, and for query (N, A, M) returns a random bit string of length $|\Pi.\mathtt{Enc}_K(N, A, M)|$. The authenticity advantage of an adversary $\mathbf{A}$ is defined as

$$\mathbf{Adv}^{\mathsf{auth}}_{\Pi}(\mathbf{A}) = \Pr[K \xleftarrow{\$} \mathcal{K}; \mathbf{A}^{\Pi.\mathtt{Enc}_K, \Pi.\mathtt{Dec}_K} \text{ forges}] ,$$

where "$\mathbf{A}^{\Pi.\mathtt{Enc}_K,\Pi.\mathtt{Dec}_K}$ forges" means that $\mathbf{A}$ makes a query to $\Pi.\mathtt{Dec}_K$ whose response is not $\perp$. We call queries to $\Pi.\mathtt{Enc}_K$ "encryption queries," and those to $\Pi.\mathtt{Dec}_K$ "decryption queries." We demand that $\mathbf{A}$ is nonce-respecting, namely, never asks two encryption queries with the same nonce, that $\mathbf{A}$ never asks a decryption query (N, A, C, T) such that there is no prior encryption query with $(C, T) = \Pi.\mathtt{Enc}_K(N, A, M)$, and that $\mathbf{A}$ never repeats a query.

2.5 Definition of Almost XOR Universal Hash Function

We will need a class of non-cryptographic functions called universal hash functions [4] defined as follows.

Definition 1. *Let $\mathcal{H}$ be a family of functions from (some set) $\mathcal{TW}_{ctr}$ to $\{0,1\}^n$ indexed by the set of keys $\mathcal{K}$. $\mathcal{H}$ is said to be (ϵ, δ)-almost XOR universal ((ϵ, δ)-AXU) if for any $c \in \{0,1\}^n$ and $ctr, ctr' \in \mathcal{TW}_{ctr}$ with $ctr \neq ctr'$, $\Pr[H \xleftarrow{\$} \mathcal{H} : H(ctr) \oplus H(ctr') = c] \leq \epsilon$ and $\Pr[H \xleftarrow{\$} \mathcal{H} : H(ctr) = c] \leq \delta$.*

3 XK and XKX [21]

3.1 Specification

XK and XKX are a combination of Minematsu's TBC Min [19] and Liskov et al.'s TBC LRW2 [16]. Let n and k be positive integers, and $\mathcal{TW}_N$ and $\mathcal{TW}_{ctr}$ non-empty sets. Let $\mathcal{F} \subseteq \mathsf{Func}(\mathcal{TW}_N, \{0,1\}^k)$ and $\mathcal{H} \subseteq \mathsf{Func}(\mathcal{TW}_{ctr}, \{0,1\}^n)$ be families of functions used in XK and XKX. Let $E \in \mathsf{BC}(\{0,1\}^k, \{0,1\}^n)$, $F \in \mathcal{F}$ and $H \in \mathcal{H}$. For a tweak $tw \in \mathcal{TW}_N$ and a plaintext block $M \in \{0,1\}^n$, the encryption of Minematsu's TBC is defined as

$$\mathtt{Min}[E, F](N, M) = E_{K_N}(M) \text{ where } K_N = F(N) \ .$$

For tweaks $(N, ctr) \in \mathcal{TW}_N \times \mathcal{TW}_{ctr}$ and a plaintext $M \in \{0,1\}^n$, the encryption of XK is defined as

$$\mathtt{XK}[E, F, h]((N, ctr), M) := \mathtt{Min}[E, F](\Delta \oplus M) \text{ where } \Delta := H(ctr) \ ,$$

and the encryption of XKX is defined as

$$\mathtt{XKX}[E, F, h]((N, ctr), M) := \Delta \oplus \mathtt{Min}[E, F](\Delta \oplus M) \text{ where } \Delta := H(ctr) \ .$$

Hereafter, F is called a first tweak function, and H is called a second tweak function. N is called a first tweak, and ctr is called a second tweak. Note that using XK and XKX in a scheme, the second tweak spaces of XK and of XKX should not be overlapped with each other. The combination of XK and XKX is denoted by XKX*.

3.2 Security of $\mathtt{XKX}^*$

$\mathtt{XKX}^*$ is a secure TSPRP [21] as long as E is a secure SPRP, $\mathcal{F}$ is a secure PRF, $\mathcal{H}$ is AXU, an adversary does not make a decryption query to $\mathtt{XK}$ and does not make queries to $\mathtt{XKX}^*$ such that the second tweak spaces of $\mathtt{XK}$ and of $\mathtt{XKX}$ are not overlapped with each other. The security bound is given below.

Theorem 1 (TSPRP Security of $\mathtt{XKX}^*$ [21]). *Assume that $\mathcal{H}$ is (ϵ, δ)-AXU. Let $\mathbf{A}$ be a (σ, t)-tsprp-adversary that does not make a decryption query to* $\mathtt{XK}$. *Here, q is the number of distinct first tweaks, and ℓ_N is the number of queries with first tweak $N \in \mathcal{TW}_N$. Then, there exist a $(\sigma, t + O(\sigma))$-sprp-adversary $\mathbf{A}_E$ and $(q, t + O(\sigma))$-prf-adversary $\mathbf{A}_F$ such that*

$$\mathbf{Adv}^{\widetilde{\mathsf{sprp}}}_{\mathtt{XKX}^*}(\mathbf{A}) \leq q \cdot \mathbf{Adv}^{\mathsf{sprp}}_{E}(\mathbf{A}_E) + \mathbf{Adv}^{\mathsf{prf}}_{\mathcal{F}}(\mathbf{A}_F) + \sum_{N \in \mathcal{N}} \ell_N^2 \cdot \max\{\epsilon, \delta\} \ .$$

3.3 $\mathtt{XKX}^*$-based AEAD schemes

In [21], $\mathtt{XKX}^*$ is applied to TBC-based nAEAD schemes such as ΘCB3 [13] and $\mathbb{OTR}$ [20]. Consider ΘCB3 with $\mathtt{XKX}^*$. In ΘCB3, each plaintext block is encrypted by the TBC, where a nonce and a counter are inputted as a tweak, and then the checksum of the plaintext blocks are encrypted. Each associated data block is encrypted by the TBC, where a counter is inputted as a tweak, and then a hash value is defined as the xor of the encrypted values. Finally, a tag is defined as the xor of the encrypted checksum and the hash value. In [21], the security bounds of ΘCB3 with $\mathtt{XKX}^*$ are given by using Theorem 1. Here, we assume that an adversary makes $q_{\mathcal{E}}$ encryption queries and q queries such that the number of BC calls of handing associated data by encryption queries is σ_A and the number of BC calls by decryption queries is $\sigma_{\mathcal{D}}$. For simplicity, we fix ℓ the number of BC calls by a query, and use the optimal parameters for $\mathcal{H}$: $\epsilon = \delta = 1/2^n$. Regarding the privacy, for each query to ΘCB3 with $\mathtt{XKX}^*$, the BC's key to take plaintext blocks and the checksum is changed, whereas the BC's key to handle associated data is fixed. Hence, using Theorem 1, the privacy bound becomes roughly $\ell^2 q_{\mathcal{E}}/2^n + \sigma_A^2/2^n$. Regarding the authenticity, when an adversary can make decryption queries with the same nonce, the BC's keys to take ciphertext blocks and the checksums by decryption queries are the same. Hence, using Theorem 1, the term $\sigma_{\mathcal{D}}^2/2^n$ is introduced in addition to $\ell^2 q_{\mathcal{E}}/2^n + \sigma_A^2/2^n$, that is, the authenticity bound becomes roughly $\ell^2 q/2^n + (\sigma_A^2 + \sigma_{\mathcal{D}}^2)/2^n$. Note that we assume that the terms $q \cdot \mathbf{Adv}^{\mathsf{sprp}}_{E}(\mathbf{A}_E)$ and $\mathbf{Adv}^{\mathsf{prf}}_{\mathcal{F}}(\mathbf{A}_F)$ are negligible compared with other terms.

4 Our Result: Improved Security Bound of $\mathtt{XKX}^*$-based nAEAD scheme

In stead of the modular proof via $\mathtt{XKX}$'s result (Theorem 1), the birthday terms $\sigma_A^2/2^n$ and $\sigma_{\mathcal{D}}^2/2^n$ might be removed by directly proving the security of the

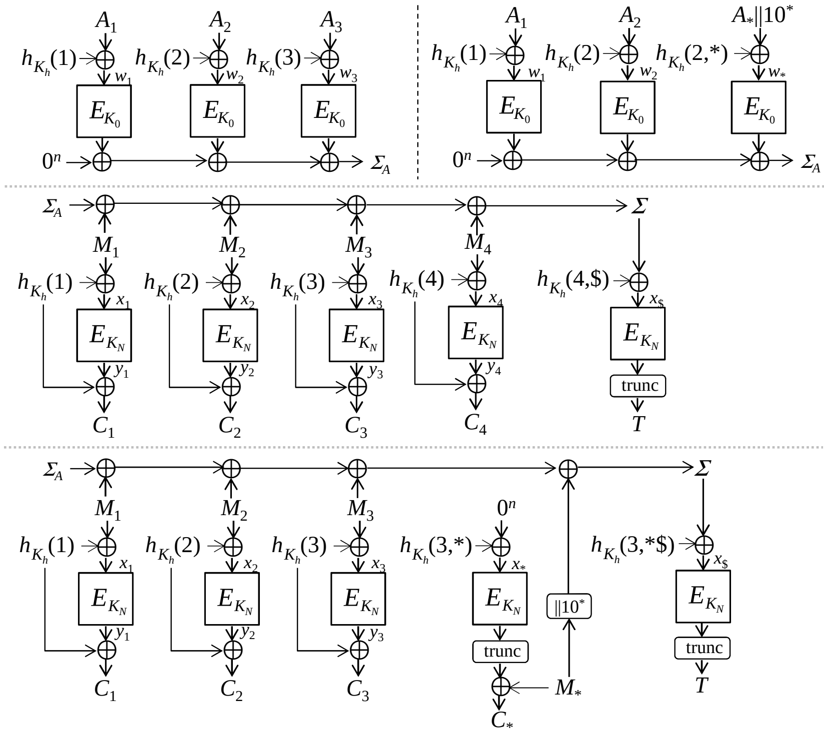

Fig. 1. ΘCB3†.`Enc` where $K_0 \leftarrow F(0)$ and $K_N \leftarrow F(N)$.

`XKX`*-based nAEAD scheme. However, as mentioned in Section 1, it might be hard. When an adversary makes decryption queries with the same nonce, the encrypted checksums are the same. Thus, the randomnesses of the tags depend on the hash values of associated data. Since the BC's key to handle associated data is fixed, the birthday term regarding associated data by decryption queries might be introduced due to the PRF-PRP switch for the BC's outputs.

In this paper, in order to remove the birthday terms, we modify ΘCB3, where the has value is XOR-ed with the checksum (instead of the encrypted checksum). We call the variant ΘCB3†. Note that by this modification, the memory size is reduced by the hash value, since one does not keep a hash value of associated data when the checksum is encrypted.

4.1 Specification of `XKX`*-based ΘCB3†

We give the specification of ΘCB3† with `XKX`* by following the notations in [13]. For simplicity, we call it ΘCB3†. Let $\mathcal{N}$ be the set of nonces of ΘCB3† such that $0 \notin \mathcal{N}$. The sets of first tweaks and of second tweaks of `XKX`* are defined as

$$\mathcal{TW}_N := \mathcal{N} \cup \{0\}$$
$$\mathcal{TW}_{ctr} := \mathbb{N}_1 \cup (\mathbb{N}_0 \times \{*\}) \cup (\mathbb{N}_0 \times \{\$\}) \cup (\mathbb{N}_0 \times \{*\$\}) \cup \mathbb{N}_1 \cup (\mathbb{N}_0 \times \{*\})$$

Algorithm 1 $\Theta\mathrm{CB3}^\dagger$

Encryption $\Theta\mathrm{CB3}^\dagger.\mathtt{Enc}(N, A, M)$

$\Sigma \leftarrow \Theta\mathrm{CB3}^\dagger.\mathtt{Hash}(A)$; $K_N \leftarrow F(N)$; $C_* \leftarrow \lambda$; $M_1, \ldots, M_m, M_* \stackrel{n}{\leftarrow} M$
for $i = 1$ to m **do**
    $C_i \leftarrow E_{K_N}(M_i \oplus H(i)) \oplus H(i)$ ▷ XKX
    $\Sigma \leftarrow \Sigma \oplus M_i$
end for
if $M_* = \lambda$ **then**
    $T \leftarrow [E_{K_N}(\Sigma \oplus H(m, \$))]^\tau$ ▷ XK
else
    $\mathrm{Pad} \leftarrow E_{K_N}(0^n \oplus H(m, *))$ ▷ XK
    $C_* \leftarrow [\mathrm{Pad}]^{|M_*|} \oplus M_*$; $\Sigma \leftarrow \Sigma \oplus M_* \| 10^*$
    $T \leftarrow [E_{K_N}(\Sigma \oplus H(m, *\$))]^\tau$ ▷ XK
end if
return $(C_1 \| \cdots \| C_m \| C_*, T)$

Decryption $\Theta\mathrm{CB3}^\dagger.\mathtt{Dec}(N, A, M, T)$

$\Sigma \leftarrow \Theta\mathrm{CB3}^\dagger.\mathtt{Hash}(A)$; $K_N \leftarrow F(N)$; $M_* \leftarrow \lambda$; $C_1, \ldots, C_m, C_* \stackrel{n}{\leftarrow} C$
for $i = 1$ to m **do**
    $M_i \leftarrow E^{-1}_{K_N}(C_i \oplus H(i)) \oplus H(i)$ ▷ XKX
    $\Sigma \leftarrow \Sigma \oplus M_i$
end for
if $M_* = \lambda$ **then**
    $T^* \leftarrow [E_{K_N}(\Sigma \oplus H(m, \$))]^\tau$ ▷ XK
else
    $\mathrm{Pad} \leftarrow E_{K_N}(0^n \oplus H(m, *))$ ▷ XK
    $C_* \leftarrow [\mathrm{Pad}]^{|M_*|} \oplus M_*$; $\Sigma \leftarrow \Sigma \oplus M_* \| 10^*$
    $T^* \leftarrow [E_{K_N}(\Sigma \oplus H(m, *\$))]^\tau$ ▷ XK
end if
if $T^* = T$ **then return** $M_1 \| \cdots \| M_m \| M_*$
if $T^* \neq T$ **then return** $\perp$

Subroutine $\Theta\mathrm{CB3}^\dagger.\mathtt{Hash}(A)$

$K_0 \leftarrow F(0)$; $\Sigma_A \leftarrow 0^n$; $A_1, \ldots, A_a, A_* \stackrel{n}{\leftarrow} A$
for $i = 1$ to a **do** $\Sigma_A \leftarrow \Sigma_A \oplus E_{K_0}(A_i \oplus H(i))$ ▷ XK
if $A_* \neq \lambda$ **then** $\Sigma_A \leftarrow \Sigma_A \oplus E_{K_0}(A_* \| 10^* \oplus H(i, *))$ ▷ XK
return Σ_A

where $\mathbb{N}_1$ and $\mathbb{N}_0$ are positive and nonnegative integers, respectively. "0" is used to define a BC's key to handle associated data. Hence, $\Theta\mathrm{CB3}^\dagger$ uses six types of permutations with tweaks (N,i), $(N,i,*)$, $(N,i,\$)$, $(N,i,*\$)$, (i), and $(i,*)$. The first two permutations are used to encrypt plaintext blocks. The next two permutations are used to generate a tag. The last two permutations are used to handle associated data. In each procedure, the latter permutation is used to avoid an additional permutation call by the padding. The sets of keys, associated data, plaintexts and ciphertexts of $\Theta\mathrm{CB3}^\dagger$ is defined as $\mathcal{K} := \{0,1\}^k$, $\mathcal{A} := \{0,1\}^*$, $\mathcal{M} := \{0,1\}^*$ and $\mathcal{C} := \{0,1\}^*$. In $\Theta\mathrm{CB3}^\dagger$, plaintext blocks are encrypted by `XKX`, and other data blocks (a checksum and associated data blocks) are encrypted by `XK`. In ΘCB3, a one-zero padding 10^* is used, where $X\|10^*$ is a bit string that 1 is appended to the bit string X and an appropriate number of bits 0 is appended so that the bit length becomes n. $\Theta\mathrm{CB3}^\dagger$ is specified in Algorithm 1 and is illustrated in Fig. 1.

4.2 Security Bounds of $\Theta\mathrm{CB3}^\dagger$

The adversarial parameters are defined as follows.

- $q_\mathcal{E}$: the number of encryption queries.
- $q_\mathcal{D}$: the number of decryption queries.
- $q = q_\mathcal{E} + q_\mathcal{D}$.
- $\sigma_\mathcal{E}$: the number of BC calls by encryption queries.
- σ: the number of BC calls by all queries.
- $\ell_{\mathsf{H},\alpha}$: the number of BC calls in $\Theta\mathrm{CB3}^\dagger$.`Hash` at the α-th encryption query, where $\alpha \in [q_\mathcal{E}]$.
- $l_{\mathsf{H},\beta}$: the number of BC calls in $\Theta\mathrm{CB3}^\dagger$.`Hash` at the β-th decryption query, where $\beta \in [q_\mathcal{D}]$.
- $\ell_{\mathsf{E},\alpha}$: the number of BC calls except for those in $\Theta\mathrm{CB3}^\dagger$.`Hash` at the α-th encryption query, where $\alpha \in [q_\mathcal{E}]$.
- $l_{\mathsf{D},\beta}$: the number of BC calls except for those in $\Theta\mathrm{CB3}^\dagger$.`Hash` at the β-th decryption query, where $\beta \in [q_\mathcal{D}]$.
- $l_{\mathcal{D},\beta} := l_{\mathsf{H},\beta} + l_{\mathsf{D},\beta}$, where $\beta \in [q_\mathcal{D}]$.
- $\ell_\mathsf{E} := \max\{\ell_{\mathsf{E},\alpha} | \alpha \in [q_\mathcal{E}]\}$.
- $l_\mathsf{D} := \max\{l_{\mathsf{D},\beta} | \beta \in [q_\mathcal{D}]\}$.
- $\ell_\mathcal{E} := \max\{\ell_{\mathsf{E},\alpha} + \ell_{\mathsf{H},\alpha} | \alpha \in [q_\mathcal{E}]\}$.
- $l_\mathcal{D} := \max\{l_{\mathsf{D},\beta} + l_{\mathsf{H},\beta} | \beta \in [q_\mathcal{D}]\}$.

Theorem 2 (Privacy of $\Theta\mathrm{CB3}^\dagger$). *Assume that $\mathcal{H}$ is (ϵ,δ)-AXU. Let $\mathbf{A}$ be a priv-adversary that runs in time t. Then, there exist a $(\sigma_\mathcal{E}, t+O(\sigma_\mathcal{E}))$-sprp-adversary $\mathbf{A}_E$ and $(q_\mathcal{E}, t+O(\sigma_\mathcal{E}))$-prf-adversary $\mathbf{A}_F$ such that*

$$\mathbf{Adv}^{\mathsf{priv}}_{\Theta\mathrm{CB3}^\dagger}(\mathbf{A}) \le q_\mathcal{E} \cdot \mathbf{Adv}^{\mathsf{sprp}}_{E}(\mathbf{A}_E) + \mathbf{Adv}^{\mathsf{prf}}_{\mathcal{F}}(\mathbf{A}_F) + \sum_{\alpha=1}^{q_\mathcal{E}} \ell_{\mathsf{E},\alpha}^2 \cdot \max\{\epsilon,\delta\} \ .$$

Theorem 3 (Authenticity of ΘCB3†). *Assume that $\mathcal{H}$ is (ϵ, δ)-AXU. Let $\mathbf{A}$ be a auth-adversary that runs in time t. Then, there exist a $(\sigma, t + O(\sigma))$-sprp-adversary $\mathbf{A}_E$ and $(q, t + O(\sigma))$-prf-adversary $\mathbf{A}_F$ such that*

$$\begin{aligned}\mathbf{Adv}^{\mathsf{auth}}_{\Theta\mathrm{CB3}^\dagger} \leq & (q+1) \cdot \mathbf{Adv}^{\mathsf{sprp}}_{E}(\mathbf{A}_E) + \mathbf{Adv}^{\mathsf{prf}}_{\mathcal{F}}(\mathbf{A}_F) \\ & + \frac{q_{\mathcal{D}}(2^{n-\tau}+2)}{2^n - (\ell_{\mathcal{E}} + l_{\mathcal{D}})} + (\ell_{\mathsf{E}} + \ell_{\mathsf{H}}^2) \cdot q_{\mathcal{D}} \cdot \epsilon + \sum_{\beta=1}^{q_{\mathcal{D}}} 2 l_{\mathcal{D},\beta}^2 \cdot \epsilon \ .\end{aligned}$$

Before giving the security proofs, we study the security bounds. Assume that the SPRP-security and PRF-security terms are negligible, which can be achieved by using a BC with a long-size key such as $k = 2n$ (See Section 6 in [21] for the detail). For simplicity, we fix ℓ the number of blockcipher calls by a query, and use the optimal parameters for $\mathcal{H}$: $\epsilon = \delta = 1/2^n$. Then, the privacy bound becomes roughly $\ell^2 q_{\mathcal{E}}/2^n$, since $\ell_{\mathsf{E},\alpha} \leq \ell$. Regarding the authenticity bound, the term $\frac{q_{\mathcal{D}}(2^{n-\tau}+2)}{2^n-(\ell_{\mathcal{E}}+l_{\mathcal{D}})}$ becomes roughly $q/2^\tau$ and the terms $(\ell_{\mathsf{E}} + \ell_{\mathsf{H}}^2) \cdot q_{\mathcal{D}} \cdot \epsilon + \sum_{\beta=1}^{q_{\mathcal{D}}} 2 l_{\mathcal{D},\beta}^2 \cdot \epsilon$ become roughly $\ell^2 q_{\mathcal{D}}/2^n$, since $\ell_{\mathsf{E}}, \ell_{\mathsf{H}}, l_{\mathcal{D},\beta} \leq \ell$. Hence, the authenticity bound becomes roughly $q/2^\tau + \ell^2 q_{\mathcal{D}}/2^n$, and assuming $q/2^\tau \ll \ell^2 q_{\mathcal{D}}/2^n$, it is roughly $\ell^2 q_{\mathcal{D}}/2^n$. Hence the birthday terms $\sigma_A^2/2^n, \sigma_{\mathcal{D}}^2/2^n$ are absent in the security bounds.

4.3 Proof of Theorem 2

Firstly, $\mathtt{XKX}^*$ except for $\mathtt{XK}$ in ΘCB3†.$\mathtt{Hash}$ are replaced with a TRP $\widetilde{P} \xleftarrow{\$} \widetilde{\mathsf{Perm}}(\mathcal{TW}_N \times \mathcal{TW}_{ctr}, \{0,1\}^n)$. In this replacement, from Theorem 1, the following terms are introduced.

$$q_{\mathcal{E}} \cdot \mathbf{Adv}^{\mathsf{sprp}}_{E}(\mathbf{A}_E) + \mathbf{Adv}^{\mathsf{prf}}_{\mathcal{F}}(\mathbf{A}_F) + \sum_{\alpha=1}^{q_{\mathcal{E}}} \ell_{\mathsf{E},\alpha}^2 \cdot \max\{\epsilon, \delta\}$$

In the modified ΘCB3†, for each encryption query, the output blocks are defined by $\widetilde{P}$, and for each $\widetilde{P}$ call, a distinct tweak is used. Thereby, all outputs are randomly drawn (regardless of outputs of ΘCB3†.$\mathtt{Hash}$). Hence, the upper-bound in Theorem 2 is obtained.

4.4 Proof of Theorem 3

Let $\Pi_0 := \Theta\mathrm{CB3}^\dagger$, and

$$\mathsf{Game0} := \left(F \xleftarrow{\$} \mathcal{F}; H \xleftarrow{\$} \mathcal{H}; \mathbf{A}^{\Pi_0} \text{ forges} \right) \ .$$

This game is called Game 0.

We next consider Game 1. From Game 0 to Game 1, Minematsu's TBC, $\mathtt{Min}$, is replaced with a TRP. $\Pi_1 := (\Pi_1.\mathtt{Enc}, \Pi_1.\mathtt{Dec})$ denotes the resultant scheme

Algorithm 2 Scheme Π_1

Encryption $\Pi_1.\texttt{Enc}(N, A, M)$

$\Sigma \leftarrow \Pi_1.\texttt{Hash}(A)$; $C_* \leftarrow \lambda$; $M_1, \ldots, M_m, M_* \xleftarrow{n} M$
for $i = 1$ to m **do** $C_i \leftarrow \widetilde{P}_N(M_i \oplus H(i)) \oplus H(i)$; $\Sigma \leftarrow \Sigma \oplus M_i$
if $M_* = \lambda$ **then**
    $T \leftarrow \left[\widetilde{P}_N(\Sigma \oplus H(m, \$))\right]^{\tau}$
else
    $\mathrm{Pad} \leftarrow \widetilde{P}_N(0^n \oplus H(m, *))$
    $C_* \leftarrow [\mathrm{Pad}]^{|M_*|} \oplus M_*$; $\Sigma \leftarrow \Sigma \oplus M_* \| 10^*$
    $T \leftarrow \left[\widetilde{P}_N(\Sigma \oplus H(m, *\$))\right]^{\tau}$
end if
return $(C_1 \| \cdots \| C_m \| C_*, T)$

Decryption $\Pi_1.\texttt{Dec}(N, A, M, T)$

$\Sigma \leftarrow \Pi_1.\texttt{Hash}(A)$; $M_* \leftarrow \lambda$; $C_1, \ldots, C_m, C_* \xleftarrow{n} C$
for $i = 1$ to m **do** $M_i \leftarrow \widetilde{P}_N^{-1}(C_i \oplus H(i)) \oplus H(i)$; $\Sigma \leftarrow \Sigma \oplus M_i$
if $M_* = \lambda$ **then**
    $T^* \leftarrow \left[\widetilde{P}_N(\Sigma \oplus H(m, \$))\right]^{\tau}$
else
    $\mathrm{Pad} \leftarrow \widetilde{P}_N(0^n \oplus H(m, *))$;
    $C_* \leftarrow [\mathrm{Pad}]^{|M_*|} \oplus M_*$; $\Sigma \leftarrow \Sigma \oplus M_* \| 10^*$
    $T^* \leftarrow \left[\widetilde{P}_N(\Sigma \oplus H(m, *\$))\right]^{\tau}$
end if
if $T^* = T$ **then return** M
if $T^* \neq T$ **then return** $\perp$

Subroutine $\Pi_1.\texttt{Hash}(A)$

$\Sigma \leftarrow 0^n$; $A_1, \ldots, A_a, A_* \xleftarrow{n} A$
for $i = 1$ to a **do** $\Sigma \leftarrow \Sigma \oplus \widetilde{P}_0(A_i \oplus H(i))$
if $A_* \neq \lambda$ **then** $\Sigma \leftarrow \Sigma \oplus \widetilde{P}_0(A_* \| 10^* \oplus H(i, *))$
return Σ

using a TRP $\widetilde{P} \xleftarrow{\$} \widetilde{\mathsf{Perm}}(\mathcal{TW}_N, \{0,1\}^n)$, which is defined in Algorithm 2, where $\widetilde{P}_N(\cdot) := \widetilde{P}(N, \cdot)$. In Game 1, the following event is considered.

$$\mathsf{Game1} := \left(\widetilde{P} \xleftarrow{\$} \widetilde{\mathsf{Perm}}(\mathcal{TW}_N, \{0,1\}^n); H \xleftarrow{\$} \mathcal{H}; \mathbf{A}^{\Pi_1} \text{ forges}\right) .$$

$\Pr[\mathsf{Game0}] - \Pr[\mathsf{Game1}]$ can be upper-bounded by using the following lemma.

Lemma 1 (TSPRP-Security of `Min` **[19]).** *Let* $\mathbf{A}$ *be a* (μ, t)*-tsprp-adversary whose queries include* ν *distinct tweaks in* $\mathcal{TW}_N$*. Then there exist a* $(\mu, t+O(\mu))$*-sprp-adversary* $\mathbf{A}_E$ *and a* $(\nu, t + O(\mu))$*-prf-adversary* $\mathbf{A}_F$ *such that*

$$\mathbf{Adv}^{\widetilde{\mathsf{sprp}}}_{\texttt{Min}}(\mathbf{A}) \leq \nu \cdot \mathbf{Adv}^{\mathsf{sprp}}_{E}(\mathbf{A}_E) + \mathbf{Adv}^{\mathsf{prf}}_{\mathcal{F}}(\mathbf{A}_F) .$$

Hence, Min can be replaced with a TRP $\widetilde{P} \xleftarrow{\$} \widetilde{\mathsf{Perm}}(\mathcal{TW}_N, \{0,1\}^n)$ with the above security loss where $\nu = q+1$ and $\mu = \sigma$, that is,

$$\Pr[\mathsf{Game0}] - \Pr[\mathsf{Game1}] \le (q+1) \cdot \mathbf{Adv}_E^{\mathsf{sprp}}(\mathbf{A}_E) + \mathbf{Adv}_{\mathcal{F}}^{\mathsf{prf}}(\mathbf{A}_F) \ . \tag{1}$$

Next, $\Pr[\mathsf{Game1}]$ is upper-bounded. The probability can be upper-bounded by the similar analysis as PMAC [3] that considers a collision in inputs to $\widetilde{P}_N$ that define tags. If no such collision occurs, all tags are randomly drawn from roughly 2^n values, thereby the probability that $\mathbf{A}$ forgers is roughly $q_{\mathcal{D}}/2^n$. In the following, the detailed analysis of $\Pr[\mathsf{Game1}]$ is given.

Analysis of Game1. Let $x_i := M_i \oplus H(i)$, $y_i := C_i \oplus H(i)$, $x_* := H(j,*)$, $x_\$:= \varSigma \oplus H(m,\$)$ (if $M_* = \lambda$); $x_\$:= \varSigma \oplus H(m,*\$)$ (if $M_* \ne \lambda$), $w_i := A_i \oplus H(i)$, and $w_* := A_* \| 10^* \oplus H(a,*)$. See also Fig. 1 for these notations. Note that x_* is absent if $M_* = \lambda$. We first consider the case where $\mathbf{A}$ forges at the β-th decryption query where $\beta \in [q_{\mathcal{D}}]$. The event is denoted by $\mathsf{Forge}[\beta]$. Hereafter, a value v defined at the β-th decryption query is denoted by $\hat{v}$. Then the following cases are considered.

• Case 1: $\hat{N}$ is new, i.e., $\hat{N}$ is distinct from all nonces defined at the previous encryption queries. In this case, the following cases are considered.
— Subcase 1-1: $\hat{x}_\$ \notin \{\hat{x}_1, \hat{x}_2, \ldots, \hat{x}_{\hat{m}}, \hat{x}_*\}$. Since $\hat{x}_\$$ is a new input to $\widetilde{P}_N$, the output $\hat{T}$ is randomly drawn from at least $2^n - l_{\mathsf{D}}$ values, thereby we have $\Pr[\mathsf{Forge}[\beta]] \le 1/(2^n - l_{\mathsf{D}})$.
— Subcase 1-2: $\hat{x}_\$ \in \{\hat{x}_1, \hat{x}_2, \ldots, \hat{x}_{\hat{m}}, \hat{x}_*\}$. In this case, $\Pr[\mathsf{Forge}[\beta]]$ is upper-bounded by the probability that Subase 1-2 occurs. Assume that $\hat{x}_\$ = \hat{x}_i$ where $\hat{x}_i \in \{\hat{x}_1, \hat{x}_2, \ldots, \hat{x}_{\hat{m}}, \hat{x}_*\}$. $\hat{x}_\$$ has the form $\hat{x}_\$ = \hat{\varSigma} \oplus H(\hat{tw}_\$)$, and $\hat{x}_i$ has the form $\hat{x}_i = \hat{M}_i \oplus H(\hat{tw}_i)$ where $\hat{tw}_\$ \ne \hat{tw}_i$. $\hat{x}_\$ = \hat{x}_i$ implies that

$$\hat{\varSigma} \oplus H(\hat{tw}_\$) = \hat{M}_i \oplus H(\hat{tw}_i) \Rightarrow H(\hat{tw}_\$) \oplus H(\hat{tw}_i) = \hat{\varSigma} \oplus \hat{M}_i,$$

Since $\mathcal{H}$ is ϵ-AXU, the probability that Subcase 1-2 occurs is at most $l_{\mathsf{D},\beta} \cdot \epsilon$.

• Case 2: $\hat{N}$ is not new. In this case, the following cases are considered. Assume that the nonce defined at the α-th encryption query equals $\hat{N}$, where $\alpha \in [q_{\mathcal{E}}]$. Note that since $\mathbf{A}$ is nonce-respecting, the number of encryption queries whose nonces equal $\hat{N}$ is at most 1. Hereafter, a value v defined at the α-th encryption query is denoted by $\bar{v}$.
— Subcase 2-1: $\hat{x}_\$ \notin \{\hat{x}_1, \hat{x}_2, \ldots, \hat{x}_{\hat{m}}, \hat{x}_*, \bar{x}_1, \bar{x}_2, \ldots, \bar{x}_{\bar{m}}, \bar{x}_*, \bar{x}_\$\}$. Since $\hat{x}_\$$ is a new input to $P_{\hat{N}}$, the output is randomly drawn from at least $2^n - (\ell_{\mathsf{E}} + l_{\mathsf{D}})$, thereby $\Pr[\mathsf{Forge}[\beta]] \le 2^{n-\tau}/(2^n - (\ell_{\mathsf{E}} + l_{\mathsf{D}}))$.
— Subcase 2-2: $\hat{x}_\$ \in \{\hat{x}_1, \hat{x}_2, \ldots, \hat{x}_{\hat{m}}, \hat{x}_*, \bar{x}_1, \bar{x}_2, \ldots, \bar{x}_{\bar{m}}, \bar{x}_*\}$. Assume that $\hat{x}_\$ = x'$ where $x' \in \{\hat{x}_1, \hat{x}_2, \ldots, \hat{x}_{\hat{m}}, \hat{x}_*, \bar{x}_1, \bar{x}_2, \ldots, \bar{x}_{\bar{m}}, \bar{x}_*\}$. $\hat{x}_\$$ has the form $\hat{x}_\$ = \hat{\varSigma} \oplus H(\hat{tw}_\$)$, and x' has the form $x' = X' \oplus H(tw')$ for some n-bit value X' such that $\hat{tw}_* \ne tw'$. $\hat{x}_\$ = x'$ implies that

$$\hat{\varSigma} \oplus H(\hat{tw}_\$) = X' \oplus H(tw') \Rightarrow H(\hat{tw}_\$) \oplus H(tw') = \hat{\varSigma} \oplus X' \ .$$

$\Pr[\mathsf{Forge}[\beta]]$ is upper-bounded by the collision probability. Since $\mathcal{H}$ is (ϵ, δ)-AXU, the collision probability is at most $(\ell_{\mathsf{E},\alpha} + l_{\mathsf{D},\beta} - 2) \cdot \epsilon$.
— $\mathsf{Subcase\ 2\text{-}3}$: $\hat{x}_\$ = \bar{x}_\$$ and $\hat{tw}_\$ \neq \bar{tw}_\$$. $\hat{x}_*$ has the form $\hat{x}_* = \hat{\Sigma} \oplus H(\hat{tw}_\$)$, and $\bar{x}_*$ has the form $\bar{x}_* = \bar{\Sigma} \oplus H(\bar{tw}_\$)$. $\hat{x}_* = \bar{x}_*$ implies that

$$\hat{\Sigma} \oplus H(\hat{tw}_\$) = \bar{\Sigma} \oplus H(\bar{tw}_\$) \Rightarrow H(\hat{tw}_\$) \oplus H(\bar{tw}_\$) = \hat{\Sigma} \oplus \bar{\Sigma}.$$

$\Pr[\mathsf{Forge}[\beta]]$ is upper-bounded by the collision probability. Since $\mathcal{H}$ is (ϵ, δ)-AXU, the collision probability is at most ϵ.
— $\mathsf{Subcase\ 2\text{-}4}$: $\hat{x}_\$ = \bar{x}_\$$ and $\hat{tw}_\$ = \bar{tw}_\$$ and $\hat{A} = \bar{A}$. In this case, $\hat{\Sigma} = \bar{\Sigma}$, and by $\hat{tw}_\$ = \bar{tw}_\$$, $\hat{m} = \bar{m}$ and $\ell_{\mathsf{E},\alpha} = l_{\mathsf{D},\beta}$ are satisfied. Let $I := \{1, 2, \ldots, \hat{m}\}$. We remove trivial induces from I, i.e., induces $i \in I$ such that $\hat{M}_i = \bar{M}_i$ are removed. The resultant subset is denoted by I'. Then

$$\begin{aligned} \hat{\Sigma} = \bar{\Sigma} \Leftrightarrow \left(\bigoplus_{i=1}^{\hat{m}} \hat{M}_i\right) \oplus \hat{P} \oplus \Pi_1.\mathtt{Hash}(\hat{A}) &= \left(\bigoplus_{i=1}^{\bar{m}} \bar{M}_i\right) \oplus \bar{P} \oplus \Pi_1.\mathtt{Hash}(\bar{A}) \\ \Leftrightarrow \left(\bigoplus_{i \in I'} \hat{M}_i \oplus \bar{M}_i\right) &= \hat{P} \oplus \bar{P} \end{aligned} \tag{2}$$

where $\hat{P} = \hat{M}_* \| 10^*$ or 0^n, and $\bar{P} = \bar{M}_* \| 10^*$ or 0^n. $\Pr[\mathsf{Forge}[\beta]]$ is upper-bounded by $\Pr[(2)]$ (the probability that (2) is satisfied).

$\Pr[(2)]$ is upper-bounded. By $\hat{A} = \bar{A}$, $\hat{C} \neq \bar{C}$ is satisfied, and thus $I' \neq \emptyset$ is satisfied. Let $\hat{\mathcal{Y}} := \{\hat{y}_i | i \in I'\}$ and $\mathcal{Y} := \{\hat{y}_i, \bar{y}_i | i \in I'\}$ be multisets for I'. The following cases are considered.

- The first case is $\exists \hat{y}^\dagger \in \hat{\mathcal{Y}}, y^\ddagger \in \mathcal{Y} \backslash \{\hat{y}^\dagger\}$ s.t. $\hat{y}^\dagger = y^\ddagger$. In this case, $\Pr[(2)]$ is upper-bounded by the probability that $\hat{y}^\dagger = y^\ddagger$. $\hat{y}^\dagger$ has the form $\hat{y}^\dagger = \hat{C}^\dagger \oplus H(\hat{tw}^\dagger)$, and $y^\ddagger$ has the form $y^\ddagger = C^\ddagger \oplus H(\bar{tw}^\ddagger)$, where $\hat{tw}^\dagger \neq \bar{tw}^\ddagger$. $\hat{y}^\dagger = y^\ddagger$ implies that

  $$\hat{C}^\dagger \oplus H(\hat{tw}^\dagger) = C^\ddagger \oplus H(\bar{tw}^\ddagger) \Leftrightarrow H(\hat{tw}^\dagger) \oplus H(tw^\ddagger) = \hat{C}^\dagger \oplus C^\ddagger \ .$$

  Since $|\hat{\mathcal{Y}}| \leq l_{\mathsf{D},\beta} - 1$, $|\mathcal{Y}| \leq 2l_{\mathsf{D},\beta} - 3$ and $\mathcal{H}$ is (ϵ, δ)-AXU, in this case, $\Pr[(2)] \leq (l_{\mathsf{D},\beta} - 1)(2l_{\mathsf{D},\beta} - 3) \cdot \epsilon$.
- The second case is $\forall \hat{y}^\dagger \in \hat{\mathcal{Y}}, y^\ddagger \in \mathcal{Y} \backslash \{\hat{y}^\dagger\} : \hat{y}^\dagger \neq y^\ddagger$. In this case, $\hat{y}^\dagger \in \hat{\mathcal{Y}}$ is a new input to $P^{-1}_{\hat{N}}$, and thus the output is randomly drawn from at least $2^n - l_{\mathsf{D}}$ values. Hence, in this case, $\Pr[(2)] \leq 1/(2^n - l_{\mathsf{D}})$.

— $\mathsf{Subcase\ 2\text{-}5}$: $\hat{x}_\$ = \bar{x}_\$$ and $\hat{tw}_\$ = \bar{tw}_\$$ and $\hat{A} \neq \bar{A}$. In this case, $\hat{\Sigma} = \bar{\Sigma}$ and $\hat{m} = \bar{m}$ are satisfied. Let $I := \{1, 2, \ldots, \max\{\hat{a}, \bar{a}\}, *\}$ be the set of induces for associated data blocks. We first remove trivial induces from I, i.e., induces $i \in I$ s.t. $\hat{A}_i = \bar{A}_i$ are removed. The resultant subset is denoted by I'. By $\hat{A} \neq \bar{A}$,

$I' \neq \emptyset$ is satisfied. Then

$$\begin{aligned}\hat{\Sigma} = \bar{\Sigma} \Leftrightarrow \left(\bigoplus_{i=1}^{\hat{m}} \hat{M}_i\right) \oplus \hat{P} \oplus \Pi_1.\mathtt{Hash}(\hat{A}) &= \left(\bigoplus_{i=1}^{\bar{m}} \bar{M}_i\right) \oplus \bar{P} \oplus \Pi_1.\mathtt{Hash}(\bar{A}) \\ \Leftrightarrow \Pi_1.\mathtt{Hash}(\hat{A}) \oplus \Pi_1.\mathtt{Hash}(\bar{A}) &= \left(\bigoplus_{i=1}^{\hat{m}} \hat{M}_i \oplus \bar{M}_i\right) \oplus \hat{P} \oplus \bar{P} \end{aligned} \tag{3}$$

where $\hat{P} = \hat{M}_* \| 10^*$ or 0^n, and $\bar{P} = \bar{M}_* \| 10^*$ or 0^n. Hence, $\Pr[\mathsf{Forge}[\beta]]$ is upper-bounded by $\Pr[(3)]$ (the probability that (3) is satisfied), and similar to Sub-case 2-4, the probability can be upper-bounded by considering a collision in inputs to $\widetilde{P}_0$. The detail is given below. Let $\mathcal{W} := \{\hat{w}_i, \bar{w}_i | i \in I'\}$ be the multiset of inputs to $\widetilde{P}_0$ in $\Pi_1.\mathtt{Hash}$ with respect to induces I'. Then the following cases are considered.

- The first case is $\exists w^\dagger, w^\ddagger \in \mathcal{W}$ s.t. $w^\dagger = w^\ddagger$. This case is a collision in inputs to $\widetilde{P}_0$. $w^\dagger$ has the form $w^\dagger = A^\dagger \oplus H(tw^\dagger)$, and $w^\ddagger$ has the form $w^\ddagger = A^\ddagger \oplus H(tw^\ddagger)$, where $tw^\dagger \neq tw^\ddagger$. $w^\dagger = w^\ddagger$ implies that

$$A^\dagger \oplus H(tw^\dagger) = A^\ddagger \oplus H(tw^\ddagger) \Leftrightarrow H(tw^\dagger) \oplus H(tw^\ddagger) = A^\dagger \oplus A^\ddagger \ .$$

  In this case, $\Pr[(3)]$ is upper-bounded by the collision probability. Since $\mathcal{H}$ is ϵ-AXU, the collision probability is at most $\binom{\ell_{\mathsf{H},\alpha}+l_{\mathsf{H},\beta}}{2} \cdot \epsilon \leq 0.5(\ell_{\mathsf{H},\alpha}+l_{\mathsf{H},\beta})^2 \cdot \epsilon$.
- The second case is $\forall w^\dagger, w^\ddagger \in \mathcal{W} : w^\dagger \neq w^\ddagger$. In this case, for $w^\dagger \in \mathcal{W}$, $P_0(w^\dagger)$ is not canceled out and is randomly drawn from at least $2^n - (\ell_{\mathsf{H}} + l_{\mathsf{H}})$ values, since $|\mathcal{W}| \leq \ell_{\mathsf{H},\alpha} + l_{\mathsf{H},\beta} \leq \ell_{\mathsf{H}} + l_{\mathsf{H}}$. We thus have $\Pr[(3)] \leq 1/(2^n - (\ell_{\mathsf{H}} + l_{\mathsf{H}}))$.

Conclusion of the Proof. From the above analyses,

$$\begin{aligned}\Pr[\mathsf{Forge}[\beta] \wedge \mathsf{Case\ 1}] \leq& \frac{1}{2^n - l_{\mathsf{D}}} + l_{\mathsf{D},\beta} \cdot \epsilon \\ \Pr[\mathsf{Forge}[\beta] \wedge \mathsf{Case\ 2}] \leq& \frac{2^{n-\tau}}{2^n - (\ell_{\mathsf{E}} + l_{\mathsf{D}})} + (\ell_{\mathsf{E},\alpha} + l_{\mathsf{D},\beta} - 2) \cdot \epsilon + \epsilon \\ &+ (l_{\mathsf{D},\beta} - 1)(2l_{\mathsf{D},\beta} - 3) \cdot \epsilon + \frac{1}{2^n - l_{\mathsf{D}}} \\ &+ 0.5(\ell_{\mathsf{H},\alpha} + l_{\mathsf{H},\beta})^2 \cdot \epsilon + \frac{1}{2^n - (\ell_{\mathsf{H}} + l_{\mathsf{H}})} \\ \leq& \frac{2^{n-\tau} + 2}{2^n - (\ell_{\mathcal{E}} + l_{\mathcal{D}})} + (\ell_{\mathsf{E},\alpha} + 2l_{\mathsf{D},\beta}^2 + \ell_{\mathsf{H},\alpha}^2 + l_{\mathsf{H},\beta}^2) \cdot \epsilon \\ \leq& \frac{2^{n-\tau} + 2}{2^n - (\ell_{\mathcal{E}} + l_{\mathcal{D}})} + (\ell_{\mathsf{E}} + \ell_{\mathsf{H}}^2 + 2l_{\mathcal{D},\beta}^2) \cdot \epsilon \ .\end{aligned}$$

Summing the above bounds gives

$$
\begin{aligned}
\Pr[\mathsf{Game1}] &\le \sum_{\beta=1}^{q_{\mathcal{D}}} \Pr[\mathsf{Forge}[\beta]] \\
&\le \sum_{\beta=1}^{q_{\mathcal{D}}} \max\{\Pr[\mathsf{Forge}[\beta] \wedge \mathsf{Case\ 1}], \Pr[\mathsf{Forge}[\beta] \wedge \mathsf{Case\ 2}]\} \\
&\le \sum_{\beta=1}^{q_{\mathcal{D}}} \Pr[\mathsf{Forge}[\beta] \wedge \mathsf{Case\ 2}] \\
&\le \frac{q_{\mathcal{D}}(2^{n-\tau}+2)}{2^n - (\ell_{\mathcal{E}} + l_{\mathcal{D}})} + (\ell_{\mathsf{E}} + \ell_{\mathsf{H}}^2) \cdot q_{\mathcal{D}} \cdot \epsilon + \sum_{\beta=1}^{q_{\mathcal{D}}} 2 l_{\mathcal{D},\beta}^2 \cdot \epsilon \ . \qquad (4)
\end{aligned}
$$

Finally, the upper-bound in Theorem 3 is obtained by (1) and (4)

5 BC-based Instantiations

BC-based Instantiations of F. As mentioned in [21], F can be instantiated from a BC. Let $w_0, w_1, \ldots, w_{\lfloor k/n \rfloor} \in \{0,1\}^c$ be distinct bit strings for a positive integer c. The first tweak space is defined as $\mathcal{TW}_N := \{0,1\}^{n-c}$. Then the instantiations are given below.

- $F_K^{(1)}(N) = \left[Y_0 \| Y_1 \| \cdots \| Y_{\lfloor k/n \rfloor - 1}\right]^k$ where $Y_i = E_K(w_i \| N)$.
- $F_K^{(2)}(N) = \left[\left(Y_0 \oplus Y_1\right) \| \left(Y_0 \oplus Y_2\right) \| \cdots \| \left(Y_0 \oplus Y_{\lfloor k/n \rfloor}\right)\right]^k$ where $Y_i = E_K(w_i \| N)$.

Incorporating the above function into ΘCB3†, "0" is defined as some bit string $const_0 \in \{0,1\}^{n-c}$ and $\mathcal{N} := \{0,1\}^{n-c} \backslash \{const_0\}$. Note that $2^c \ge \lfloor k/n \rfloor$ for $F^{(1)}$, and $2^c - 1 \ge \lfloor k/n \rfloor$ for $F^{(2)}$.

As mentioned in [21], the security bound of $F^{(1)}$ is obtained by the PRP-PRF switch [2], and that of $F^{(2)}$ is obtained by the security result of `CENC` [23, 7, 9].

Lemma 2 (PRF Security of $F^{(1)}$ [2]). *For any (q,t)-prf-adversary $\mathbf{A}$, there exists a $(\lfloor k/n \rfloor \cdot q, t + O(q))$-prp-adversary $\mathbf{A}_E$ such that*

$$\mathbf{Adv}_{F^{(1)}}^{\mathsf{prf}}(\mathbf{A}) \le \mathbf{Adv}_E^{\mathsf{prp}}(\mathbf{A}_E) + \frac{\lfloor k/n \rfloor \cdot q^2}{2^{n+1}} \ .$$

Lemma 3 (PRF Security of $F^{(2)}$ [23, 7, 9]). *For any (q,t)-prf-adversary $\mathbf{A}$ such that $q \le 2^n/134$, there exists a $((\lfloor k/n \rfloor + 1)q, t + O(q))$-prp-adversary $\mathbf{A}_E$ such that*

$$\mathbf{Adv}_{F^{(2)}}^{\mathsf{prf}}(\mathbf{A}) \le \mathbf{Adv}_E^{\mathsf{prp}}(\mathbf{A}_E) + \frac{(\lfloor k/n \rfloor)^2 \cdot q}{2^n} \ .$$

Hence, incorporating these PRFs into `XKX`*, these terms are introduced into the security bounds.

BC-based Instantiations of H. The function H in $\mathtt{XKX}^*$ can be instantiated from a BC by the powering-up scheme [25], the gray-code-based scheme [13, 26], and the LFSR-based scheme [5, 6]. Consider the powering-up scheme. It uses the multiplications by $2, 3$ and 7 over $GF(2^n)$. H is realized as follow. Define $L = E_K(const_H)$ for some constant $const_H \in \{0,1\}^n$. Then, for a non-negative integer i, $H(i) := 2^i \cdot L$, $H(i,*) := 2^i \cdot 3 \cdot L$, $H(i,\$) := 2^i \cdot 7 \cdot L$, and $H(i,*\$) := 2^i \cdot 3 \cdot 7 \cdot L$. Regarding the probabilities ϵ and δ, replacing E_K with a random permutation, since L is randomly drawn from $\{0,1\}^n$, $\epsilon = \delta = 1/2^n$ is satisfied.

Remark. Using the above instantiation of F and the powering-up scheme together, $const_H$ should be distinct from all inputs to the BC in F, i.e., $const_H \neq w_i \| N$ for $\forall i \in \{0, 1, \ldots, \lfloor k/n \rfloor\}, N \in \mathcal{TW}_N$.

6 Conclusion

In this paper, we improved the security bounds of the $\mathtt{XKX}^*$-based AEAD scheme. The previous security bounds were given by the modular proof, which are roughly $\ell^2 q/2^n + \sigma_A^2/2^n + \sigma_{\mathcal{D}}^2/2^n$, where ℓ is the number of BC calls by a query, q is the number of queries, σ_A is the number of BC calls to handle associated data by encryption queries, and $\sigma_{\mathcal{D}}$ is the number of BC calls by decryption queries. The birthday terms $\sigma_A^2/2^n, \sigma_{\mathcal{D}}^2/2^n$ might become dominant, for example, when n is small and when DoS attacks are performed. In this paper, in order to remove the birthday terms, we modified ΘCB3 called ΘCB3†, and proved that for ΘCB3† with $\mathtt{XKX}^*$, the birthday terms can be removed, i.e., the security bounds become roughly $\ell^2 q/2^n$.

Acknowledgments

We would like to thank Atul Luykx for his comments and suggestions.

References

1. Bellare, M., Namprempre, C.: Authenticated encryption: Relations among notions and analysis of the generic composition paradigm. J. Cryptology 21(4), 469–491 (2008)
2. Bellare, M., Rogaway, P.: Code-based game-playing proofs and the security of triple encryption. IACR Cryptology ePrint Archive 2004, 331 (2004)
3. Black, J., Rogaway, P.: A block-cipher mode of operation for parallelizable message authentication. In: Advances in Cryptology - EUROCRYPT 2002, International Conference on the Theory and Applications of Cryptographic Techniques, Amsterdam, The Netherlands, April 28 - May 2, 2002, Proceedings. pp. 384–397 (2002)
4. Carter, L., Wegman, M.N.: Universal classes of hash functions. J. Comput. Syst. Sci. 18(2), 143–154 (1979)

5. Chakraborty, D., Sarkar, P.: A general construction of tweakable block ciphers and different modes of operations. IEEE Trans. Information Theory 54(5), 1991–2006 (2008)
6. Granger, R., Jovanovic, P., Mennink, B., Neves, S.: Improved masking for tweakable blockciphers with applications to authenticated encryption. In: Advances in Cryptology - EUROCRYPT 2016 - 35th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Vienna, Austria, May 8-12, 2016, Proceedings, Part I. pp. 263–293 (2016)
7. Iwata, T.: New blockcipher modes of operation with beyond the birthday bound security. In: Fast Software Encryption, 13th International Workshop, FSE 2006, Graz, Austria, March 15-17, 2006, Revised Selected Papers. pp. 310–327 (2006)
8. Iwata, T.: Authenticated encryption mode for beyond the birthday bound security. In: Progress in Cryptology - AFRICACRYPT 2008, First International Conference on Cryptology in Africa, Casablanca, Morocco, June 11-14, 2008. Proceedings. pp. 125–142 (2008)
9. Iwata, T., Mennink, B., Vizár, D.: CENC is optimally secure. IACR Cryptology ePrint Archive 2016, 1087 (2016)
10. Iwata, T., Minematsu, K.: Stronger security variants of GCM-SIV. IACR Trans. Symmetric Cryptol. 2016(1), 134–157 (2016)
11. Iwata, T., Yasuda, K.: BTM: A single-key, inverse-cipher-free mode for deterministic authenticated encryption. In: Selected Areas in Cryptography, 16th Annual International Workshop, SAC 2009, Calgary, Alberta, Canada, August 13-14, 2009, Revised Selected Papers. pp. 313–330 (2009)
12. Iwata, T., Yasuda, K.: HBS: A single-key mode of operation for deterministic authenticated encryption. In: Fast Software Encryption, 16th International Workshop, FSE 2009, Leuven, Belgium, February 22-25, 2009, Revised Selected Papers. pp. 394–415 (2009)
13. Krovetz, T., Rogaway, P.: The software performance of authenticated-encryption modes. In: Fast Software Encryption - 18th International Workshop, FSE 2011, Lyngby, Denmark, February 13-16, 2011, Revised Selected Papers. pp. 306–327 (2011)
14. Lampe, R., Seurin, Y.: Tweakable blockciphers with asymptotically optimal security. In: Fast Software Encryption - 20th International Workshop, FSE 2013, Singapore, March 11-13, 2013. Revised Selected Papers. pp. 133–151 (2013)
15. Landecker, W., Shrimpton, T., Terashima, R.S.: Tweakable blockciphers with beyond birthday-bound security. In: Advances in Cryptology - CRYPTO 2012 - 32nd Annual Cryptology Conference, Santa Barbara, CA, USA, August 19-23, 2012. Proceedings. pp. 14–30 (2012)
16. Liskov, M., Rivest, R.L., Wagner, D.: Tweakable block ciphers. In: Advances in Cryptology - CRYPTO 2002, 22nd Annual International Cryptology Conference, Santa Barbara, California, USA, August 18-22, 2002, Proceedings. pp. 31–46 (2002)
17. Lucks, S.: The sum of prps is a secure PRF. In: Advances in Cryptology - EUROCRYPT 2000, International Conference on the Theory and Application of Cryptographic Techniques, Bruges, Belgium, May 14-18, 2000, Proceeding. pp. 470–484 (2000)
18. Mennink, B., Reyhanitabar, R., Vizár, D.: Security of full-state keyed sponge and duplex: Applications to authenticated encryption. In: Advances in Cryptology - ASIACRYPT 2015 - 21st International Conference on the Theory and Application of Cryptology and Information Security, Auckland, New Zealand, November 29 - December 3, 2015, Proceedings, Part II. pp. 465–489 (2015)

19. Minematsu, K.: Beyond-birthday-bound security based on tweakable block cipher. In: Fast Software Encryption, 16th International Workshop, FSE 2009, Leuven, Belgium, February 22-25, 2009, Revised Selected Papers. pp. 308–326 (2009)
20. Minematsu, K.: Parallelizable rate-1 authenticated encryption from pseudorandom functions. In: Advances in Cryptology - EUROCRYPT 2014 - 33rd Annual International Conference on the Theory and Applications of Cryptographic Techniques, Copenhagen, Denmark, May 11-15, 2014. Proceedings. pp. 275–292 (2014)
21. Naito, Y.: Tweakable blockciphers for efficient authenticated encryptions with beyond the birthday-bound security. ePrint 2017/466 and IACR Trans. Symmetric Cryptol. 2017(2), 1–26 (2017)
22. Patarin, J.: Security of random feistel schemes with 5 or more rounds. In: Advances in Cryptology - CRYPTO 2004, 24th Annual International CryptologyConference, Santa Barbara, California, USA, August 15-19, 2004, Proceedings. pp. 106–122 (2004)
23. Patarin, J.: Introduction to mirror theory: Analysis of systems of linear equalities and linear non equalities for cryptography. IACR Cryptology ePrint Archive 2010, 287 (2010)
24. Rogaway, P.: Authenticated-encryption with associated-data. In: Proceedings of the 9th ACM Conference on Computer and Communications Security, CCS 2002, Washington, DC, USA, November 18-22, 2002. pp. 98–107 (2002)
25. Rogaway, P.: Efficient instantiations of tweakable blockciphers and refinements to modes OCB and PMAC. In: Advances in Cryptology - ASIACRYPT 2004, 10th International Conference on the Theory and Application of Cryptology and Information Security, Jeju Island, Korea, December 5-9, 2004, Proceedings. pp. 16–31 (2004)
26. Rogaway, P., Bellare, M., Black, J., Krovetz, T.: OCB: a block-cipher mode of operation for efficient authenticated encryption. In: CCS 2001, Proceedings of the 8th ACM Conference on Computer and Communications Security, Philadelphia, Pennsylvania, USA, November 6-8, 2001. pp. 196–205 (2001)
27. Wang, L., Guo, J., Zhang, G., Zhao, J., Gu, D.: How to build fully secure tweakable blockciphers from classical blockciphers. In: Advances in Cryptology - ASIACRYPT 2016 - 22nd International Conference on the Theory and Application of Cryptology and Information Security, Hanoi, Vietnam, December 4-8, 2016, Proceedings, Part I. pp. 455–483 (2016)