Analyzing the Provable Security Bounds of GIFT-COFB and Photon-Beetle

Akiko Inoue[1], Tetsu Iwata[2], and Kazuhiko Minematsu[1]

[1] NEC, Kawasaki, Japan
a_inoue@nec.com,k-minematsu@nec.com
[2] Nagoya University, Nagoya, Japan
tetsu.iwata@nagoya-u.jp

Abstract. We study the provable security claims of two NIST Lightweight Cryptography (LwC) finalists, GIFT-COFB and Photon-Beetle, and present several attacks whose complexities contradict their claimed bounds in their final round specification documents. For GIFT-COFB, we show an attack using q_e encryption queries and no decryption query to break privacy (IND-CPA). The success probability is $O(q_e/2^{n/2})$ for n-bit block while the claimed bound contains $O(q_e^2/2^n)$. This positively solves an open question posed in [Khairallah, ePrint 2021/648 (also accepted at FSE 2022)]. For Photon-Beetle, we show an attack using q_e encryption queries (using a small number of input blocks) followed by a single decryption query and no primitive query to break authenticity (INT-CTXT). The success probability is $O(q_e^2/2^b)$ for a b-bit block permutation, and it is significantly larger than what the claimed bound tells, which is independent of the number of encryption queries. We also show a simple tag guessing attack that violates the INT-CTXT bound when the rate $r = 32$. Then, we analyze other (improved/modified) bounds of Photon-Beetle shown in the subsequent papers [Chakraborty et al., ToSC 2020(2) and Chakraborty et al., ePrint 2019/1475]. As a side result of our security analysis of Photon-Beetle, we point out that a simple and efficient forgery attack is possible in the related-key setting.
We emphasize that our results do not contradict the claimed "bit security" in the LwC specification documents for any of the schemes that we studied. That is, we do not negate the claims that GIFT-COFB is $(n/2 - \log n)$-bit secure for $n = 128$, and Photon-Beetle is $(b/2 - \log b/2)$-bit secure for $b = 256$ and $r = 128$, where r is a rate. We also note that the security against related-key attacks is not included in the security requirements of NIST LwC, and is not claimed by the designers.

Keywords: Authenticated Encryption · Lightweight Cryptography · Provable Security · NIST

1 Introduction

NIST Lightweight cryptography[3] aims at standardizing authenticated encryption (AE) schemes for resource-constrained devices. In March 2021, NIST has an-

[3] https://csrc.nist.gov/projects/lightweight-cryptography

nounced ten finalists among the 32 second-round candidates. The finalists include GIFT-COFB [3] and Photon-Beetle [6]. GIFT-COFB is a block cipher-based AE that combines a variant of COFB mode [13] and the lightweight 128-bit block cipher GIFT [5]. Photon-Beetle is a permutation-based AE that combines Beetle mode [11] and the lightweight cryptographic permutation Photon [19], which is an ISO standard [1]. This paper studies the provable security bounds of GIFT-COFB and Photon-Beetle, and shows some attacks whose success probabilities are inconsistent with the presented security bounds in the final round specification documents of NIST LwC.

GIFT-COFB. For the original COFB and GIFT-COFB, the security bounds for the combined AE notion of IND-CPA and INT-CTXT were presented in [3, 13]. Assuming a nonce-respecting attacker and that the underlying block cipher is a random permutation, GIFT-COFB's AE bound is roughly $\sigma^2/2^n + nq_d/2^{n/2}$ for $\sigma = \sigma_e + \sigma_d + q_e + q_d$, where σ_e (resp. σ_d) denotes the total queried blocks in encryption (resp. decryption) queries, and q_e (resp. q_d) denotes the number of encryption (resp. decryption) queries. This bound suggests that if (1) σ_e reaches $2^{n/2}$, or (2) σ_d reaches $2^{n/2}$, or (3) q_d reaches $2^{n/2}/n$, the bound reaches 1 and hence no security guarantee is possible. The tightness of these conditions has been studied by Khairallah [21, 22, 23] and Inoue and Minematsu (IM21) [20]. Khairallah [21, 22, 23] showed attacks with $q_d = 2^{n/2}$ with about $\sigma_e = 2^{n/2}$ or $\sigma_e = 2^{n/4}$, called Weak Key attack and Mask collision attack [21, 22]. Khairallah finally showed one with $q_e = 1$, $\sigma_e = O(1)$ (a few blocks) and $q_d = 2^{n/2}$, called Mask Presuming attack [23]. The last one implies that the tightness condition (3) has only the small gap of $\log n$ factor. Inoue and Minematsu [20] studied the tightness of (1) and showed an attack with $\sigma_e = 2^{n/2}$ and $q_d = 1$. As in the previous attacks, this attack breaks the authenticity and matches the aforementioned bound. For (2) it remains unsolved, and [20] mentioned that it might be an artifact in the proofs.

We take a closer look at the condition (1). IM21's attack with q_e encryption queries and 1 decryption query has success probability roughly $q_e^2/2^n$. However, we found an improved attack that needs q_e encryption queries to break privacy (hence the combined AE notion) success probability roughly $q_e/2^{n/2}$. The existence of such an attack has been posed as an open problem by Khairallah [23]. We solved this positively. This implies a contradiction with the bound in the NIST LwC document although the bit-level security maintains. We give a brief analysis on the root of this contradiction in Sect. 3.2.

Photon-Beetle. For Photon-Beetle, the security proofs for the original version and the NIST LwC version have been shown in [6, 11, 12]. For b-bit block permutation with $b = 256$ and rate (which is the length of one message block processed in one permutation call) $r = 128$, the security bounds roughly tell $b/2 - \log b/2 = 121$-bit security for both IND-CPA and INT-CTXT. Dobraunig and Mennink commented on a constant factor related to a key recovery attack [18], and Mège analysed the security of the hash function [27].

We focus the authenticity bound shown in the final round NIST LwC submission document [6], which is roughly $q_p(q+q')/2^b + rq_p/2^{b/2} + q_p^r/2^{(b/2)\cdot(r-1)} + r\sigma'/2^{256-r}$, where q_p, q, q' and σ' denote the number of primitive queries, the number of encryption queries, the number of decryption queries, and the total number of blocks in decryption queries. The rate can be either $r = 128$ or 32, where $r = 128$ is the primary setting. The tag length is 128 bits for both cases. When $r = 128$, we observed that if $q_p = 0$, *i.e.* we do not query the primitive (permutation), the above authenticity bound reduces to the bound that has no contribution from encryption queries. We invalidate this by presenting a simple forgery using $2^{b/2}$ encryption queries and a single decryption query. The success probability is close to 1, while the claimed bound indicates a negligibly small probability with that complexity. This attack shows inconsistency with the claimed bound and implies the lack of the birthday term with respect to the block size, $O(q_e^2/2^b)$, in the claimed bound. Moreover, when $r = 32$, the INT-CTXT bound reduces to the bound that is smaller than $q'/2^{128}$, which is impossible to achieve for any AE of 128-bit tags. Thus, a simple tag guessing attack (*i.e.*, decryption queries with identical nonce, AD, ciphertext, and distinct tags) invalidates the claimed bound. This implies even the break of bit-level security suggested by the bound. However, the bit security shown in [6, Table 4.1] claims 128-bit authenticity. We clarify that we do not break the figure. Moreover, we study other (improved or modified) security bounds for Photon-Beetle shown in the subsequent papers [15, 16]. In [16], an improved bound AE bound is presented. The bound claims that the IND-CPA security is maintained beyond $2^{b/2}$ encryption queries, but this is not possible to achieve. The same paper presents a simplified AE bound, and we point out that this cannot be true. We then clarify that the ePrint version [15] of [16] addresses the issue, while we still see an issue in simplification.

As a side result of our security analysis of Photon-Beetle for $r = 128$, we point out that a simple and efficient forgery attack is possible in the related-key setting, in which the attacker can modify the key used in the oracle [7, 9, 26]. In Photon-Beetle, a fixed constant is xor'ed into the secret key when the input (both AD and a message) is empty, and our forgery makes use of this fact. See [4, 17, 24] for examples of related-key attacks on some AE schemes. In the domain of public-key authenticated encryption, see [25].

Our attacks do not depend on the primitives and do not break the primitives. The attack against GIFT-COFB does not work against the COFB versions in [13,14] because of the shorter nonce length than the NIST LwC version. Our attacks show some inconsistencies in the claimed security bounds of GIFT-COFB and Photon-Beetle. At the same time, we would like to emphasize that these results do not negate the claimed bit security levels of GIFT-COFB and Photon-Beetle. We also note that the security against related-key attacks is not included in the security requirements of NIST LwC, and is not claimed by the designers.

2 Preliminaries

2.1 Notations

Our notations largely follow the specifications of GIFT-COFB and Photon-Beetle [3, 6]. Let $[i] \coloneqq \{1, \ldots, i\}$ and $[\![i]\!] \coloneqq \{0, 1, \ldots, i\}$. Let $\{0,1\}^*$ denote the set of all bit strings. The set of bit strings whose length is a multiple of n is denoted as $(\{0,1\}^n)^*$. For $X \in \{0,1\}^*$, $|X|$ denotes its bit length. An empty string ε is a bit string of length zero; we have $|\varepsilon| = 0$. The block length of $X \in \{0,1\}^*$ in n-bit blocks is denoted as $|X|_n \coloneqq \lceil |X|/n \rceil$. A concatenation of two bit strings X and Y is written as $X \,\|\, Y$ or simply XY. Let $\mathsf{Trunc}_t(X)$ denote the first $t \in [\![|X|]\!]$ bits of X, where $\mathsf{Trunc}_0(X) = \varepsilon$. For two integers a and b, we write $a|b$ if a divides b. For a bit string X, $X \ll c$ denotes the left-shift of X by c bits. Bit rotation of X by c bits to the left (right) is denoted by $X \lll c$ ($X \ggg c$).

For $X \in \{0,1\}^*$, the parsing operation of X into n-bit blocks is denoted by $(X[1], \ldots, X[x]) \xleftarrow{n} X$. Here, if $X \neq \varepsilon$, $X[1] \,\|\, X[2] \,\|\, \ldots \,\|\, X[x] = X$ and $|X[i]| = n$ for $i < |X|_n$ and $|X[x]| \in [n]$ for $x = |X|_n$. By writing $X_1 \,\|\, X_2 \xleftarrow{a_1, a_2} X$ we mean the parsing such that $X_1 \,\|\, X_2 = X$ and $|X_1| = a_1$ and $|X_2| = a_2$. If $X = \varepsilon$, $x = 1$ and $|X[x]| = 0$ (*i.e.*, the parsing yields the same empty string). The sequence of i zeros is denoted by 0^i. We may use an integer $i \in \{0, 1, \ldots, 2^n - 1\}$ to mean an element of $\{0,1\}^n$, assuming the standard encoding, *e.g.*, for $n = 4$, 3 denotes 0011.

Galois field of 2^n elements. An element a in the Galois extension field $\mathrm{GF}(2^n)$ will be interchangeably denoted as an n-bit string $a_{n-1} \ldots a_1 a_0$ or an integer $\sum_{i=0}^{n-1} a_i 2^i$. Hence, by writing $2 \cdot a$ or $2a$ when no confusion is possible, we mean the multiplication of a by $2 = \mathtt{x}$. This operation is called *doubling* and has been frequently used by various modes for the "domain separation" task. See [28] for example. For $n = 64$ (that will be used for GIFT-COFB), we use the primitive polynomial $\mathtt{x}^{64} + \mathtt{x}^4 + \mathtt{x}^3 + \mathtt{x} + 1$ to define the field $\mathrm{GF}(2^n)$. In this case, the doubling $2 \cdot a$ is $(a \ll 1)$ if $\mathtt{msb}_1(a) = 0$ and $(a \ll 1) \oplus (0^{59}11011)$ if $\mathtt{msb}_1(a) = 1$, and the tripling $3 \cdot a$ means $2 \cdot a \oplus a$. Combined expressions such as $2^i \cdot 3^j \cdot a$ are defined analogously, namely i doublings and j triplings of a.

2.2 Cryptographic components

A keyed function with key space $\mathcal{K}$, domain $\mathcal{X}$, and range $\mathcal{Y}$ is a function $F : \mathcal{K} \times \mathcal{X} \to \mathcal{Y}$. We may write $F_K(X)$ for $F(K, X)$. If Mode is a mode of operation for F using a single key $K \in \mathcal{K}$ for F, we write $\mathsf{Mode}[F_K]$ instead of $\mathsf{Mode}[F]_K$. A block cipher is a keyed function $E : \mathcal{K} \times \mathcal{T} \times \mathcal{M} \to \mathcal{M}$ such that for each $K \in \mathcal{K}$, $E(K, \cdot)$ is a permutation over $\mathcal{M}$. A cryptographic permutation $P : \mathcal{M} \to \mathcal{M}$ is simply a (keyless) permutation. GIFT-COFB is based on a block cipher, while Photon-Beetle is based on a cryptographic permutation.

Let $\mathcal{A}$ be an adversary that queries c oracles, $O_1, \ldots, O_c$ in an arbitrarily order and outputs a certain final output. By writing $\mathcal{A}^{O_1, O_2, \cdots}$, we mean the final output of $\mathcal{A}$. Let $\mathrm{Perm}(n)$ be the set of all permutations over $\{0,1\}^n$. For block cipher $E : \mathcal{K} \times \mathcal{M} \to \mathcal{M}$, the PRP advantage is defined as

$$\mathbf{Adv}^{\mathtt{prp}}_E(\mathcal{A}) \coloneqq \Pr\left[K \xleftarrow{\$} \mathcal{K} : \mathcal{A}^{E_K(*)} \Rightarrow 1\right] - \Pr\left[\pi \xleftarrow{\$} \mathrm{Perm}(n) : \mathcal{A}^{\pi(*)} \Rightarrow 1\right].$$

The PRP advantage represents the indistinguishability of E_K from the uniform random permutation of the same message space for adversaries performing queries to encryption oracles (either $E_K(*)$ or $\pi(*)$).

2.3 Authenticated encryption

We briefly describe the syntax and security notions about authenticated encryption (AE). Our targets are both nonce-based AEs [8, 29], which requires nonce to be unique for each encryption. Let Π denote a nonce-based AE scheme consisting of an encryption function $\Pi.\mathcal{E}_K$ and a decryption function $\Pi.\mathcal{D}_K$, for key $K \xleftarrow{\$} \mathcal{K}$. For plaintext M with nonce N and associated data (AD) A, $\Pi.\mathcal{E}_K$ takes (N, A, M) and returns ciphertext C (typically $|C| = |M|$) and tag T. Here, AD is a part of the input that is not encrypted but must be authenticated (*e.g.*, a protocol header). The tuple (N, A, C, T) will be sent to the receiver. For decryption, $\Pi.\mathcal{D}_K$ takes (N, A, C, T) and returns a decrypted plaintext M if the authentication check is successful, and otherwise an error symbol, $\perp$.

Security notions. The security of AEs can be defined by two notions. The privacy[4] notion is the indistinguishability of encryption oracle $\Pi.\mathcal{E}_K$ from the random-bit oracle \$ which returns random $|M| + \tau$ bits for any query (N, A, M). The adversary is assumed to be nonce-respecting, i.e., nonces can be arbitrarily chosen but must be distinct for encryption queries. The privacy advantage is defined as

$$\mathbf{Adv}^{\mathtt{priv}}_{\Pi}(\mathcal{A}) \coloneqq \Pr\left[K \xleftarrow{\$} \mathcal{K} : \mathcal{A}^{\Pi.\mathcal{E}_K(\cdot,\cdot,\cdot)} \Rightarrow 1\right] - \Pr\left[\mathcal{A}^{\$(\cdot,\cdot,\cdot)} \Rightarrow 1\right],$$

which measures the hardness of breaking the privacy notion for $\mathcal{A}$. This notion corresponds to IND-CPA [8].

The authenticity notion is the probability of successful forgery via queries to $\Pi.\mathcal{E}_K$ and $\Pi.\mathcal{D}_K$ oracles. We define the authenticity advantage as

$$\mathbf{Adv}^{\mathtt{auth}}_{\Pi}(\mathcal{A}) \coloneqq \Pr\left[K \xleftarrow{\$} \mathcal{K} : \mathcal{A}^{\Pi.\mathcal{E}_K(\cdot,\cdot,\cdot),\Pi.\mathcal{D}_K(\cdot,\cdot,\cdot,\cdot)} \text{ forges}\right],$$

where $\mathcal{A}$ forges if it receives a value $M' \neq \perp$ from $\Pi.\mathcal{D}_K$. Here, to prevent trivial wins, if $(C, T) \leftarrow \Pi.\mathcal{E}_K(N, A, M)$ is obtained earlier, $\mathcal{A}$ cannot query (N, A, C, T) to $\Pi.\mathcal{D}_K$. The adversary must be nonce-respecting for encryption queries, but has no restriction on decryption queries. It corresponds to INT-CTXT notion [8].

It is also common to use a combined notion, sometimes called AE advantage, define as

$$\mathbf{Adv}^{\mathtt{ae}}_{\Pi}(\mathcal{A}) \coloneqq \Pr\left[K \xleftarrow{\$} \mathcal{K} : \mathcal{A}^{\Pi.\mathcal{E}_K(\cdot,\cdot,\cdot),\Pi.\mathcal{D}_K(\cdot,\cdot,\cdot,\cdot)} \Rightarrow 1\right] - \Pr\left[\mathcal{A}^{\$(\cdot,\cdot,\cdot),\perp} \Rightarrow 1\right],$$

where $\perp$ oracle denotes the oracle that always returns the rejection symbol. It is know that the sum of Privacy and Authenticity advantages is a bound of AE advantage [30], thus it compactly represents the security of an AE scheme as a whole.

[4] Following the literature (*e.g.*, [28]), we conventionally refer to it as privacy, but in practice, it may be more intuitive to call it confidentiality.

3 Analysis of GIFT-COFB

Specification. For reference, the specification of GIFT-COFB is shown in Appendix A (Figs. 4 and 5). The padding function $\mathtt{pad} : \{0,1\}^* \to (\{0,1\}^n)^*$ is a variant of so-called one-zero padding and defined as $\mathtt{pad}(X) = X$ if $X \neq \varepsilon$ and $|X| \bmod n = 0$, and otherwise $\mathtt{pad}(X) = X \,\|\, 10^{(n-(|X| \bmod n)-1)}$. The G in Fig. 4 denotes a matrix such that $G \cdot X \coloneqq (X[2], X[1] \lll 1)$ for $X[1], X[2] \xleftarrow{n/2} X$, $X \in \{0,1\}^n$. We also write $G(X)$ to mean $G \cdot X$.

We show our attack against GIFT-COFB that contradicts the claimed security bound. As mentioned earlier, this does not invalidate the claimed bit security levels, namely 64-bit IND-CPA security and 58-bit INT-CTXT security in the specification document.

3.1 Our attack

The security bound shown in the latest NIST LwC specification document is as follows (with minor changes in notations):

Theorem 1 (Chapter 4 in [3]).

$$\begin{aligned}\mathbf{Adv}^{\mathsf{ae}}_{\mathsf{GIFT\text{-}COFB}}(\mathcal{A}) \leq & \mathbf{Adv}^{\mathtt{prp}}_{\mathsf{GIFT}}(q', t') + \frac{\binom{q'}{2}}{2^n} + \frac{1}{2^{n/2}} + \frac{q_d(n+4)}{2^{n/2+1}} \\ & + \frac{3\sigma_e^2 + q_d + 2(q_e + \sigma_e + \sigma_d) \cdot \sigma_d}{2^n},\end{aligned}$$

where $q' = q_e + q_d + \sigma_e + \sigma_d$, which corresponds to the total number of block cipher calls through the game, and $t' = t + O(q')$. Note that the advantage has been taken by the maximum advantage over all the adversaries making q_e encryption queries, q_d decryption queries and running in time t, such σ_e, σ_d are the total number of blocks queried in the encryption and decryption queries, respectively.

The term $\mathbf{Adv}^{\mathtt{prp}}_{\mathsf{GIFT}}(q', t')$ denotes the maximum of PRP advantage for any adversary of q' queries and t' time complexity. When we only use encryption queries, the above bound effectively reduces to about $\sigma_e^2/2^n$ and hence about $q_e^2/2^n$ if each message is short. We present an attack using q_e encryption queries (where each message is short) with success probability about $q_e/2^{n/2}$. This contradicts the bound of Theorem 1, since $q_e^2/2^n \leq q_e/2^{n/2}$ necessarily holds when $1 \leq q_e \leq 2^{n/2}$. The attack proceeds as follows.

1. The attacker makes a query (N, A, M) to the encryption oracle such that $|A| = n$, $|M| = 2n$ and $M = M[1] \,\|\, M[2]$ (for arbitrarily chosen N, single-block A and two-block M), and it obtains corresponding (C, T), where $C = C[1] \,\|\, C[2]$, as shown in Fig. 1.
2. The attacker computes $Y[1]$, $Y[2]$, and $\mathtt{lsb}_{n/2}(X[2]) = \mathtt{lsb}_{n/2}(G(Y[1]) \oplus M[1])$. Note that $\mathtt{msb}_{n/2}(X[2])$ is unknown; nevertheless, the attacker can mount a privacy attack by using *the guessed* $X[2]$ as the nonce of the next encryption query.

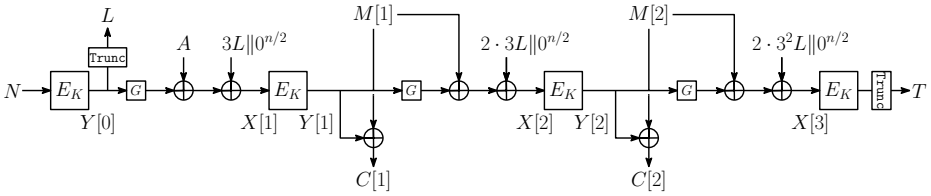

Fig. 1: The first encryption query of the attack against GIFT-COFB.

3. For $0 \leq i \leq 2^{n/2}-1$, the attacker queries (N_i, A_i, M_i), where $|A_i| = |M_i| = n$, to the encryption oracle such that

$$\begin{aligned} N_i &= (i)_{n/2} \,\|\, \mathtt{lsb}_{n/2}(X[2]), \quad L_i \coloneqq \mathsf{Trunc}_{n/2}(Y[2]), \\ A_i &= N_i \oplus G(Y[2]) \oplus 3L_i \,\|\, 0^{n/2}, \\ M_i &= N_i \oplus G(Y[2]) \oplus 3^2 L_i \,\|\, 0^{n/2}, \end{aligned}$$

where $(i)_{n/2}$ denotes $n/2$-bit string of a binary representation of i. The attacker obtains corresponding (C_i, T_i). In the real world, there always exists i such that $M_i \oplus C_i = Y[2]$ and $T_i = \mathsf{Trunc}_\tau(Y[2])$, where i fulfilling $N_i = X[2]$. In the ideal world, $\Pr[M_i \oplus C_i = Y[2], T_i = \mathsf{Trunc}_\tau(Y[2])] = 1/2^{n+\tau}$ holds for all i, and thus the attacker can find i such that $M_i \oplus C_i = Y[2]$ and $T_i = \mathsf{Trunc}_\tau(Y[2])$ holds with a negligibly small probability, $1/2^{n/2+\tau}$.

In the real world, the above attack fails when $N = X[2]$ accidentally holds because it prevents the attacker from using $X[2]$ for the next nonce. To prevent such a case, the attacker can query a longer plaintext in Step 1, and it can find $X[\cdot]$ s.t. $\mathtt{lsb}_{n/2}(X[\cdot]) \neq \mathtt{lsb}_{n/2}(N)$ with a sufficiently high probability.

We remark that this attack does not work against versions of COFB in TCHES 2017 [13] and Journal of Cryptology [14] because the nonce length of these versions is $n/2$ bits.

3.2 Brief analysis on security proof

As we mentioned in the previous section, the security bound shown in [3, Chapter 4] does not include the term $O(q_e/2^{n/2})$ nor $O(\sigma_e/2^{n/2})$. However, in [3, Sect. 4.2], the authors provide INT-CTXT bound, which includes the term $3\sigma_e/2^{64}$ assuming $n = 128$. This term is somehow missing in the final bound of the AE advantage that combines privacy and authenticity. Still, in any case, since our attack uses only encryption queries, the terms $O(q_e/2^{n/2})$ or $O(\sigma_e/2^{n/2})$ should appear in the IND-CPA security bound, originally presented in [3, Sect. 4.1]. Let us look into [2] which shows the full proof of GIFT-COFB. The authors define the following two events as the bad events.

B1: $X_{i_1}[j_1] = X_{i_2}[j_2]$ for some $(i_1, j_1) \neq (i_2, j_2)$ where $j_1,\ j_2 > 0$.
B2: $Y_{i_1}[j_1] = Y_{i_2}[j_2]$ for some $(i_1, j_1) \neq (i_2, j_2)$ where $j_1,\ j_2 > 0$.

Here, $X_i[j]$ and $Y_i[j]$ denote input and output of the j-th underlying block cipher call in the i-th encryption query. Also, $X_i[0] \coloneqq N_i$, where N_i is the nonce value in the i-th encryption query. As our attack shows, the attacker can produce a collision between $X_i[0]$ and $X_1[2]$ with probability $q_e/2^{n/2}$. One can speculate that this inconsistency could be fixed by setting $j_1,\ j_2 \geq 0$ in the above events (then it covers the presented attack), rather than $j_1,\ j_2 > 0$.

4 Analysis of Photon-Beetle

Specification. For reference, we present the AEAD specification of Photon-Beetle almost verbatim in Appendix A (Figs. 6 and 7). In the specification, $\mathtt{ozs}_r(X)$ for any X such that $|X| < r$, is another variant of one-zero padding, defined as $\mathtt{ozs}_r(X) = X \,\|\, 10^{r-|X|-1}$. The expression $\mathbf{E}?a : b$ evaluates to a if $\mathbf{E}$ holds and b otherwise. Similarly, $(\mathbf{E_1}\ \mathtt{and}\ \mathbf{E_2}?a : b : c : d)$ evaluates to a if $\mathbf{E_1} \wedge \mathbf{E_2}$ holds, b if $\mathbf{E_1} \wedge \overline{\mathbf{E_2}}$ holds, c if $\overline{\mathbf{E_1}} \wedge \mathbf{E_2}$, and d otherwise. The Shuffle in the ρ and ρ^{-1} functions is a function: $\{0,1\}^r \to \{0,1\}^r$. It is defined as $\mathsf{Shuffle}(S) = (S[2] \,\|\, S[1] \ggg 1)$, where $(S[1], S[2]) \xleftarrow{r/2} S$.

We show our attacks against Photon-Beetle that violate its claimed security bound in NIST LwC documentation [6]. We emphasize that our attacks do not violate the claimed "bit security" levels of Photon-Beetle, which are 121-bit IND-CPA and INT-CTXT security when $r = 128$, and 128-bit IND-CPA and INT-CTXT security when $r = 32$.

4.1 Claimed security bound and our attack

In [6], Photon-Beetle is claimed to be provably secure, with the security bound of

$$O\left(\frac{\sigma^2}{2^{256}} + \frac{q_p}{2^{256-r}} + \frac{q \cdot q_p}{2^{256}} + \frac{r q_p}{2^{128}} + \frac{\sigma_e^r}{2^{128(r-1)}}\right)$$

for privacy (IND-CPA), where σ is the total number of blocks in encryption queries, q_p is the number of offline queries, r is the rate ($r = 32$ or 128), q is the number of encryption queries, and σ_e is the total number of blocks in encryption queries [6, Sect. 4.1][5]. For authenticity (INT-CTXT), the claimed bound is

$$O\left(\frac{q_p(q+q')}{2^{256}} + \frac{r q_p}{2^{128}} + \frac{q_p^r}{2^{128(r-1)}} + \frac{r\sigma'}{2^{256-r}}\right), \tag{1}$$

where q_p is the number of offline queries, q is the number of encryption queries, q' is the number of decryption queries, r is the rate ($r = 32$ or 128), and σ' is the total number of blocks in decryption queries [6, Sect. 4.2].

We present two attacks that invalidate the bound in (1). The observation is that, when $q_p = 0$, *i.e.*, when the attacker does not make offline queries, then the

[5] We do not know the difference between σ and σ_e.

bound (1) is simplified into

$$O\left(\frac{r\sigma'}{2^{256-r}}\right). \tag{2}$$

We observe that the bound (2) claims that the authenticity security is maintained even if the attacker makes an unlimited number of encryption queries and that the success probability is smaller than $\sigma'/2^{128}$ when $r = 32$. In what follows, we present attacks based on these observations.

Birthday forgery against Photon-Beetle. The attack is as follows.

1. Let $q = 2^{b/2}$, and fix q distinct nonces $N_1, \ldots, N_q$, q distinct AD $A_1, \ldots, A_q$ with $|A_i| = b$, and q distinct messages $M_1, \ldots, M_q$ with $|M_i| = b + r$. The attacker chooses $M_1, \ldots, M_q$ of the form $M_i = M' \,\|\, M'_i$, where $|M'| = b$, $|M'_i| = r$, and $M'_1, \ldots, M'_q$ take q distinct values. That is, the first b bits of $M_1, \ldots, M_q$ take the same value M', and the corresponding portions of ciphertexts are used to detect a full-state collision.
2. Make q encryption queries $(N_1, A_1, M_1), \ldots, (N_q, A_q, M_q)$ and obtain (C_1, T_1), $\ldots, (C_q, T_q)$, where $|C_i| = b + r$.
3. Find (i, j) such that $C'_i = C'_j$, where C'_i is the first b bits of C_i, and the same for C'_j.
4. Output (N_i, A_i, C_j, T_j) (or (N_j, A_j, C_i, T_i)) as the forgery.

See Fig. 2 for the process of (N_i, A_i, M_i) and (N_j, A_j, M_j) when $r = 128$. With a high probability, we have a full-state collision, *i.e.*, we have (i, j) such that $S_i = S_j$ in the figure. The collision can be detected from C'_i and C'_j, which are the first b bits of C_i and C_j. If this happens, we see that the forgery in Step 4 succeeds.

The bound (2) claims that the success probability of the attack is negligibly small and at most $O(7r/2^{256-r})$ when $r = 128$ (or at most $O(6r/2^{256-r})$ depending on the interpretation of σ'), while the attack succeeds with an overwhelming probability. Therefore, the bound (1) is invalidated.

Tag guessing attack against Photon-Beetle with $r = 32$. When $r = 32$, the above setting of $q_p = 0$ makes the INT-CTXT bound (1) reduces to $32\sigma'/2^{256-32} = \sigma'/2^{219}$ which is smaller than $\sigma'/2^{128}$. When σ' is close to q', this implies a bound that is not possible to achieve with 128-bit tags. A simple tag guessing attack invalidates this bound, that is, q' decryption queries using identical (nonce, AD, ciphertext) tuple with distinct tags will succeed with probability about $q'/2^{128}$.

Discussion and implication. In [6, Sect. 4.2], the designers outline the proof of the bound (1). To quote:

> Also, if an adversary can obtain a state collision among the input/output of a permutation query with the state of an encryption query or decryption query, it can use the fact to mount an forgery attack.

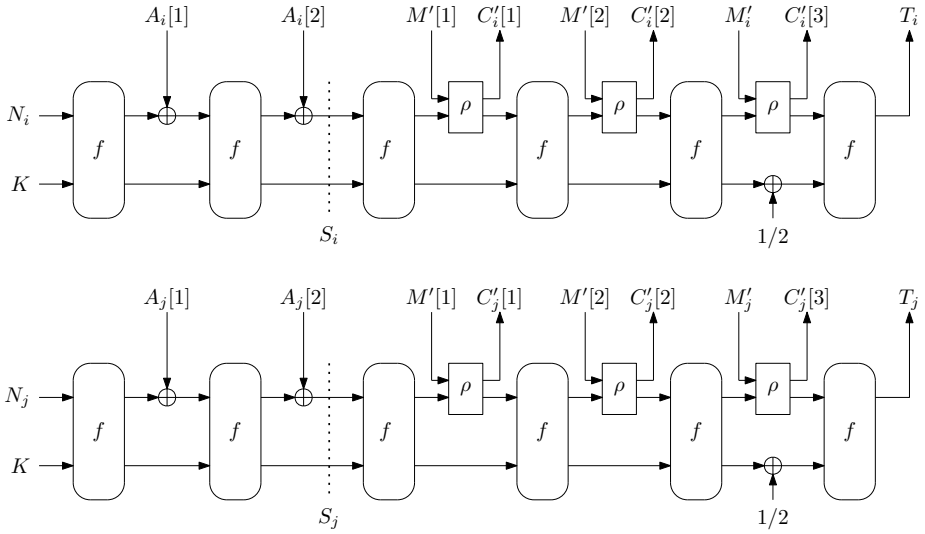

Fig. 2: Two encryption queries (N_i, A_i, M_i) and (N_j, A_j, M_j) when $r = 128$. Here, $A_i = A_i[1] \,\|\, A_i[2]$ and $M_i = M'[1] \,\|\, M'[2] \,\|\, M'_i$.

The argument here ignores a full-state collision among encryption queries, resulted in the first attack. Here is another quote from the same document:

> The trivial solution for forging is to guess the key or the tag which can be bounded by $\frac{q+q'}{2^{128}}$.

We do not find an issue here, while for $r = 32$, the bound (1) makes a stronger security claim than this argument.

We note that the above two attacks need 2^{128} complexity, and thus do not violate the claimed 121-bit security (when $r = 128$) or 128-bit security (when $r = 32$). However, our attacks show that the theoretical reasoning for the bit security in the NIST LwC document [6] is inaccurately mentioned.

4.2 Analysis of the bound in [16]

There are various provable security claims related to Beetle [6, 11, 12, 15, 16]. We do not consider the bound in [11, 12] for the difference in the specification.

For Photon-Beetle, we write the combined AE advantage as $\mathbf{Adv}^{\mathsf{ae}}_{\mathsf{Photon\text{-}Beetle}}$, which is the same as the case of combined AE notion defined in Sect. 2, except that the attacker has additional oracles to compute the forward and inverse directions of the permutation that is modeled as a public random permutation. In [16], improved provable security bounds of Photon-Beetle are presented. Corollary 1 in [16] claims that, in the combined AE notion, the success probability of the

attacker for the case $r = 128$ is

$$\begin{aligned}\mathbf{Adv}^{\mathsf{ae}}_{\mathsf{Photon\text{-}Beetle}}(\mathcal{A}) \le \frac{4\tau\sigma_d}{2^c} + \frac{4r\sigma_d}{2^c} + \frac{4b\sigma_d}{2^c} + \frac{q_p}{2^\kappa} + \frac{2q_d}{2^\tau} + \frac{2\sigma_d(\sigma+q_p)}{2^b} \\ + \frac{6\sigma_e q_p}{2^b} + \frac{8rq_p}{2^c} + \frac{4\tau q_p}{2^{b-\tau}} + \frac{\sigma_e + q_p}{2^b} + \frac{4rq_p\sigma_d}{2^{2c}}\,, \end{aligned} \tag{3}$$

where τ is the tag length, c is the capacity, r is the rate, $b = r + c$, κ is the key length, q_e is the number of encryption queries, q_d is the number of decryption queries, σ_e is the total number of blocks in encryption queries, σ_d is the total number of blocks in decryption queries, q_p is the number of offline queries, and $\sigma = \sigma_e + \sigma_d$.

When $q_p = 0$ and $q_d = \sigma_d = 0$, the bound (3) is

$$\mathbf{Adv}^{\mathsf{ae}}_{\mathsf{Photon\text{-}Beetle}}(\mathcal{A}) \le \frac{\sigma_e}{2^b}\,,$$

i.e., it claims IND-CPA security up to $\sigma_e = 2^b$, which is flawed as we show below.

We note that the birthday forgery attack in Sect. 4.1 implies a distinguishing attack with a comparable complexity as follows:

1. Let $q_e = 2^{b/2}$, and fix q_e distinct nonces $N_1, \ldots, N_{q_e}$, q_e distinct AD $A_1, \ldots, A_{q_e}$ with $|A_i| = b$. We also fix a message M with $|M| = b$.
2. Make q_e encryption queries $(N_1, A_1, M), \ldots, (N_{q_e}, A_{q_e}, M)$ and obtain (C_1, T_1), $\ldots, (C_{q_e}, T_{q_e})$, where $|C_i| = b$.
3. If there exists (i, j) such that $(C_i, T_i) = (C_j, T_j)$, then output 1 (real world). Otherwise, output 0 (ideal world).

Since the b-bit state collision can be expected in the real world, the attacker finds (i, j) in Step 3 with a high probability. The attack makes $q_e = 2^{b/2}$ encryption queries, no primitive query ($q_p = 0$), and no decryption query ($q_d = \sigma_d = 0$), violating the bound (3).

In [16, Sect. 7.2], the following AE bound is claimed for $r = 128$:

$$\mathbf{Adv}^{\mathsf{ae}}_{\mathsf{Photon\text{-}Beetle}}(\mathcal{A}) \le \frac{q_p}{2^\kappa} + \frac{13rq_p}{2^c} \tag{4}$$

When $q_p = 0$, the bound claims perfect security both in IND-CPA and INT-CTXT. Even the ideal AE scheme cannot have a perfect security bound in authenticity, and our birthday forgery in Sect. 4.1 invalidates the INT-CTXT claim, and the above distinguishing attack invalidates the IND-CPA claim.

The bound (4) is obtained from the bound (3) by using the relation

$$\sigma \le q_p\,, \tag{5}$$

which is not the case in our attacks. We do not see how the relation (5) can be ensured, as our attacks demonstrate that there are attackers with $q_p = 0$.

We clarify that the ePrint version [15] of [16] addresses the issue in the bound (3) with the following revised bound for $r = 128$:

$$\mathbf{Adv}^{\mathsf{ae}}_{\mathsf{Photon\text{-}Beetle}}(\mathcal{A}) \le \frac{8r\sigma_d}{2^c} + \frac{8b^3q_p^2\sigma_d}{2^{b+c}} + \frac{q_p}{2^\kappa} + \frac{2q_d}{2^r} + \frac{2\sigma(2\sigma+q_p)}{2^b} + \frac{q_p^2}{2^b} + \frac{6\sigma_e q_p}{2^b} + \frac{12rq_p}{2^c} + \frac{\sigma_e+q_p}{2^b} + \frac{4rq_p\sigma_d}{2^{2c}}\,, \quad (6)$$

i.e., the revised bound contains a term $\sigma^2/2^b$. A full-state collision in encryption queries is covered in the analysis of [16], and the above attack no longer applies. The source of the gap seems to be an error in the final step of the proof in [16] to take the summation of various terms, where a term $2\sigma_e^2/2^b$ has been somewhat missing.

In the ePrint version [15, Sect. 7.3.1], a simplified bound is presented. For $r = 128$, the bound is

$$\mathbf{Adv}^{\mathsf{ae}}_{\mathsf{Photon\text{-}Beetle}}(\mathcal{A}) \le \frac{q_p}{2^\kappa} + \frac{2\sigma}{2^r} + \frac{10b^2q_p^2}{2^b} + \frac{24rq_p}{2^c} + \frac{12\sigma q_p}{2^b}\,,$$

which is obtained from the bound (6) by using the relation (5). We do not have an attack for this, but we do not know its correctness, as there are attackers outside of the relation (5).

On SCHWAEMM. A NIST LwC finalist Sparkle [10] adopts Beetle. More specifically, the AE member of Sparkle, SCHWAEMM, uses Beetle with minor modifications. The specification document [10] does not present security bounds of SCHWAEMM nor mention the relationship with the original bounds of Beetle. Thus our analysis above does not have any implications to SCHWAEMM beyond the fact that it is based on Beetle. Moreover, as with the case of Photon-Beetle, we do not negate the bit security claims of SCHWAEMM.

4.3 Related-key attack

We present an efficient forgery attack against Photon-Beetle for $r = 128$ in the related-key setting [7, 9, 26]. In this setting, we consider the security notion as in Sect. 2, where we additionally assume that the adversary can modify the secret key. The encryption oracle $\Pi.\mathcal{E}_K(\cdot,\cdot,\cdot)$ takes (N, A, M) and returns $(C, T) = \Pi.\mathcal{E}_K(N, A, M)$. In the related-key setting, it additionally takes $\Delta \in \{0,1\}^k$, where k is the bit length of the secret key K. The related-key encryption oracle returns $(C, T) = \Pi.\mathcal{E}_{K\oplus\Delta}(N, A, M)$ for a query (Δ, N, A, M). The decryption oracle can also be defined to take additional input to modify the key, but we do not use this in our attack.

Our attack goes as follows:

1. Fix (Δ, N, A, M), where $\Delta = 1$, N can be any nonce, A is empty, and M can be any message such that $|M| \ge r$.

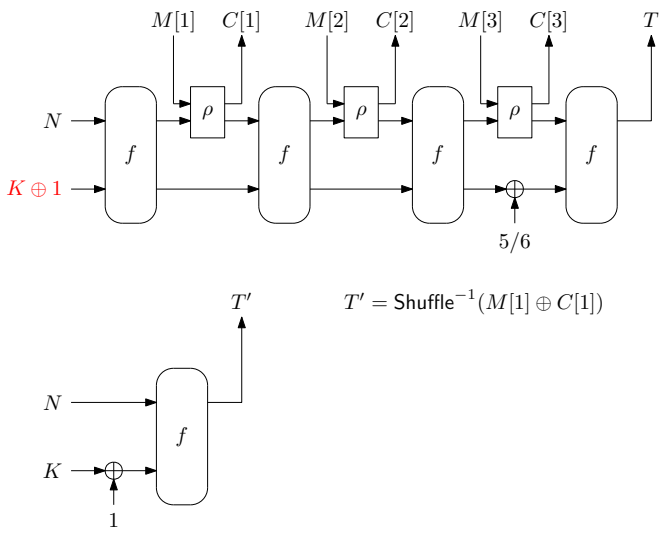

Fig. 3: The adversary makes a single encryption query with key $K \oplus 1$. This immediately allows a forgery for empty message and AD.

2. Make a related-key encryption query (Δ, N, A, M) and obtain (C, T). Let $M[1]$ be the first r bits of M, and $C[1]$ be the first r bits of C.
3. Return (N, A', C', T') as the forgery, where A' and C' are empty, and $T' = \mathsf{Shuffle}^{-1}(M[1] \oplus C[1])$.

See Fig. 3. We see that the encryption query with key $K \oplus 1$ simulates the process for the empty message and AD, and the forgery in Step 3 succeeds with probability 1. The attack makes one related-key encryption query, one decryption query, and the success probability is 1.

We remark that the impact is limited, as the attack only forges the empty AD and message. We also remark that the security against related-key attacks is not included in the security requirements of NIST LwC, and is not claimed by the designers. However, this type of weakness is avoided, *e.g.*, in SCHWAEMM.

5 Conclusions

We have investigated the provable security bounds in the specification documents of two NIST LwC finalists, GIFT-COFB and Photon-Beetle, and reported some attacks whose success probabilities are higher that what their bounds tell. We have also analyzed other bounds of Photon-Beetle shown in the subsequent papers and shown some attacks. As a side result, we presented a simple forgery attack against Photon-Beetle when $r = 128$. We remark that our attacks do not invalidate the claimed bit security levels of them, and the related-key security is not claimed by the designers.

Acknowledgements

We thank GIFT-COFB team and the authors of [15,16] for feedback on an earlier version of this paper. We thank the anonymous reviewers for helpful comments.

References

1. Information technology - Security techniques - Lightweight cryptography - Part 5: Hash-functions. ISO/IEC 29192-5:2016 (2016)
2. Banik, S., Chakraborti, A., Iwata, T., Minematsu, K., Nandi, M., Peyrin, T., Sasaki, Y., Sim, S.M., Todo, Y.: GIFT-COFB. Cryptology ePrint Archive, Report 2020/738 (2020), https://ia.cr/2020/738
3. Banik, S., Chakraborti, A., Iwata, T., Minematsu, K., Nandi, M., Peyrin, T., Sasaki, Y., Sim, S.M., Todo, Y.: GIFT-COFB v1.1. A Submission to the NIST Lightweight Cryptography Standardization Process (2021), https://csrc.nist.gov/CSRC/media/Projects/lightweight-cryptography/documents/finalist-round/updated-spec-doc/gift-cofb-spec-final.pdf
4. Banik, S., Maitra, S., Sarkar, S., Turan, M.S.: A chosen IV related key attack on Grain-128a. In: Boyd, C., Simpson, L. (eds.) ACISP 13. LNCS, vol. 7959, pp. 13–26. Springer, Heidelberg (Jul 2013). https://doi.org/10.1007/978-3-642-39059-3_2
5. Banik, S., Pandey, S.K., Peyrin, T., Sasaki, Y., Sim, S.M., Todo, Y.: GIFT: A small present - towards reaching the limit of lightweight encryption. In: Fischer, W., Homma, N. (eds.) CHES 2017. LNCS, vol. 10529, pp. 321–345. Springer, Heidelberg (Sep 2017). https://doi.org/10.1007/978-3-319-66787-4_16
6. Bao, Z., Chakraborti, A., Datta, N., Guo, J., Nandi, M., Peyrin, T., Yasuda, K.: PHOTON-Beetle Authenticated Encryption and Hash Family. A Submission to the NIST Lightweight Cryptography Standardization Process (2021), https://csrc.nist.gov/CSRC/media/Projects/lightweight-cryptography/documents/finalist-round/updated-spec-doc/photon-beetle-spec-final.pdf
7. Bellare, M., Kohno, T.: A theoretical treatment of related-key attacks: RKA-PRPs, RKA-PRFs, and applications. In: Biham, E. (ed.) EUROCRYPT 2003. LNCS, vol. 2656, pp. 491–506. Springer, Heidelberg (May 2003). https://doi.org/10.1007/3-540-39200-9_31
8. Bellare, M., Namprempre, C.: Authenticated encryption: Relations among notions and analysis of the generic composition paradigm. In: Okamoto, T. (ed.) ASIACRYPT 2000. LNCS, vol. 1976, pp. 531–545. Springer, Heidelberg (Dec 2000). https://doi.org/10.1007/3-540-44448-3_41
9. Biham, E.: New types of cryptanalytic attacks using related keys (extended abstract). In: Helleseth, T. (ed.) EUROCRYPT'93. LNCS, vol. 765, pp. 398–409. Springer, Heidelberg (May 1994). https://doi.org/10.1007/3-540-48285-7_34
10. Biryukov, C.B.A., dos Santos, L.C., Großschädl, J., Perrin, L., Udovenko, A., Velichkov, V., Wang, Q., Moradi, A., Shahmirzadi, A.R.: SPARKLE (SCHWAEMM and ESCH). A Submission to the NIST Lightweight Cryptography Standardization Process (2021), https://csrc.nist.gov/CSRC/media/Projects/lightweight-cryptography/documents/finalist-round/updated-spec-doc/sparkle-spec-final.pdf
11. Chakraborti, A., Datta, N., Nandi, M., Yasuda, K.: Beetle family of lightweight and secure authenticated encryption ciphers. IACR TCHES **2018**(2), 218–241 (2018). https://doi.org/10.13154/tches.v2018.i2.218-241, https://tches.iacr.org/index.php/TCHES/article/view/881

12. Chakraborti, A., Datta, N., Nandi, M., Yasuda, K.: Beetle family of lightweight and secure authenticated encryption ciphers. Cryptology ePrint Archive, Report 2018/805 (2018), https://eprint.iacr.org/2018/805
13. Chakraborti, A., Iwata, T., Minematsu, K., Nandi, M.: Blockcipher-based authenticated encryption: How small can we go? In: Fischer, W., Homma, N. (eds.) CHES 2017. LNCS, vol. 10529, pp. 277–298. Springer, Heidelberg (Sep 2017). https://doi.org/10.1007/978-3-319-66787-4_14
14. Chakraborti, A., Iwata, T., Minematsu, K., Nandi, M.: Blockcipher-based authenticated encryption: How small can we go? Journal of Cryptology **33**(3), 703–741 (Jul 2020). https://doi.org/10.1007/s00145-019-09325-z
15. Chakraborty, B., Jha, A., Nandi, M.: On the security of sponge-type authenticated encryption modes. Cryptology ePrint Archive, Report 2019/1475 (2019), https://eprint.iacr.org/2019/1475
16. Chakraborty, B., Jha, A., Nandi, M.: On the security of sponge-type authenticated encryption modes. IACR Trans. Symm. Cryptol. **2020**(2), 93–119 (2020). https://doi.org/10.13154/tosc.v2020.i2.93-119
17. Dobraunig, C., Eichlseder, M., Mendel, F.: Related-key forgeries for Prøst-OTR. In: Leander, G. (ed.) FSE 2015. LNCS, vol. 9054, pp. 282–296. Springer, Heidelberg (Mar 2015). https://doi.org/10.1007/978-3-662-48116-5_14
18. Dobraunig, C., Mennink, B.: Key recovery attack on PHOTON-Beetle. OFFICIAL COMMENT: PHOTON-Beetle (2020), https://csrc.nist.gov/CSRC/media/Projects/lightweight-cryptography/documents/round-2/official-comments/photon-beetle-round2-official-comment.pdf
19. Guo, J., Peyrin, T., Poschmann, A.: The PHOTON family of lightweight hash functions. In: Rogaway, P. (ed.) CRYPTO 2011. LNCS, vol. 6841, pp. 222–239. Springer, Heidelberg (Aug 2011). https://doi.org/10.1007/978-3-642-22792-9_13
20. Inoue, A., Minematsu, K.: GIFT-COFB is tightly birthday secure with encryption queries. Cryptology ePrint Archive, Report 2021/737 (2021), https://ia.cr/2021/737
21. Khairallah, M.: Weak keys in the rekeying paradigm: Application to COMET and mixFeed. IACR Trans. Symm. Cryptol. **2019**(4), 272–289 (2019). https://doi.org/10.13154/tosc.v2019.i4.272-289
22. Khairallah, M.: Observations on the tightness of the security bounds of GIFT-COFB and HyENA. Cryptology ePrint Archive, Report 2020/1463 (2020), https://eprint.iacr.org/2020/1463
23. Khairallah, M.: Security of COFB against chosen ciphertext attacks. Cryptology ePrint Archive, Report 2021/648 (2021), https://eprint.iacr.org/2021/648, (also accepted at FSE 2022)
24. Lee, Y., Jeong, K., Sung, J., Hong, S.: Related-key chosen IV attacks on Grain-v1 and Grain-128. In: Mu, Y., Susilo, W., Seberry, J. (eds.) ACISP 08. LNCS, vol. 5107, pp. 321–335. Springer, Heidelberg (Jul 2008)
25. Lu, X., Li, B., Jia, D.: KDM-CCA security from RKA secure authenticated encryption. In: Oswald, E., Fischlin, M. (eds.) EUROCRYPT 2015, Part I. LNCS, vol. 9056, pp. 559–583. Springer, Heidelberg (Apr 2015). https://doi.org/10.1007/978-3-662-46800-5_22
26. Lucks, S.: Ciphers secure against related-key attacks. In: Roy, B.K., Meier, W. (eds.) FSE 2004. LNCS, vol. 3017, pp. 359–370. Springer, Heidelberg (Feb 2004). https://doi.org/10.1007/978-3-540-25937-4_23
27. Mège, A.: OFFICIAL COMMENT: PHOTON-Beetle (2021), https://csrc.nist.gov/CSRC/media/Projects/lightweight-cryptography/documents/round-2/official-comments/photon-beetle-round2-official-comment.pdf

28. Rogaway, P.: Efficient instantiations of tweakable blockciphers and refinements to modes OCB and PMAC. In: Lee, P.J. (ed.) ASIACRYPT 2004. LNCS, vol. 3329, pp. 16–31. Springer, Heidelberg (Dec 2004). https://doi.org/10.1007/978-3-540-30539-2_2
29. Rogaway, P.: Nonce-based symmetric encryption. In: Roy, B.K., Meier, W. (eds.) FSE 2004. LNCS, vol. 3017, pp. 348–359. Springer, Heidelberg (Feb 2004). https://doi.org/10.1007/978-3-540-25937-4_22
30. Rogaway, P., Shrimpton, T.: A provable-security treatment of the key-wrap problem. In: Vaudenay, S. (ed.) EUROCRYPT 2006. LNCS, vol. 4004, pp. 373–390. Springer, Heidelberg (May / Jun 2006). https://doi.org/10.1007/11761679_23

A Specifications of GIFT-COFB and Photon-Beetle

Algorithm GIFT-COFB-$\mathcal{E}_K(N, A, M)$

$Y[0] \leftarrow E_K(N),\ L \leftarrow \mathsf{Trunc}_{n/2}(Y[0])$
$(A[1], \ldots, A[a]) \xleftarrow{n} \mathsf{pad}(A)$
if $M \neq \epsilon$ **then**
  $(M[1], \ldots, M[m]) \xleftarrow{n} \mathsf{pad}(M)$
for $i = 1$ **to** $a - 1$
  $L \leftarrow 2 \cdot L$
  $X[i] \leftarrow A[i] \oplus G \cdot Y[i-1] \oplus L \| 0^{n/2}$
  $Y[i] \leftarrow E_K(X[i])$
if $|A| \bmod n = 0$ **and** $A \neq \epsilon$ **then** $L \leftarrow 3 \cdot L$
else $L \leftarrow 3^2 \cdot L$
if $M = \epsilon$ **then** $L \leftarrow 3^2 \cdot L$
$X[a] \leftarrow A[a] \oplus G \cdot Y[a-1] \oplus L \| 0^{n/2}$
$Y[a] \leftarrow E_K(X[a])$
for $i = 1$ **to** $m - 1$
  $L \leftarrow 2 \cdot L$
  $C[i] \leftarrow M[i] \oplus Y[i + a - 1]$
  $X[i + a] \leftarrow M[i] \oplus G \cdot Y[i + a - 1] \oplus L \| 0^{n/2}$
  $Y[i + a] \leftarrow E_K(X[i + a])$
if $M \neq \epsilon$ **then**
  if $|M| \bmod n = 0$ **then** $L \leftarrow 3 \cdot L$
  else $L \leftarrow 3^2 \cdot L$
  $C[m] \leftarrow M[m] \oplus Y[a + m - 1]$
  $X[a + m] \leftarrow M[m] \oplus G \cdot Y[a + m - 1] \oplus L \| 0^{n/2}$
  $Y[a + m] \leftarrow E_K(X[a + m])$
  $C \leftarrow \mathsf{Trunc}_{|M|}(C[1] || \ldots || C[m])$
  $T \leftarrow \mathsf{Trunc}_{\tau}(Y[a + m])$
else $C \leftarrow \epsilon$, $T \leftarrow \mathsf{Trunc}_{\tau}(Y[a])$
return (C, T)

Algorithm GIFT-COFB-$\mathcal{D}_K(N, A, C, T)$

$Y[0] \leftarrow E_K(N),\ L \leftarrow \mathsf{Trunc}_{n/2}(Y[0])$
$(A[1], \ldots, A[a]) \xleftarrow{n} \mathsf{pad}(A)$
if $C \neq \epsilon$ **then**
  $(C[1], \ldots, C[c]) \xleftarrow{n} \mathsf{pad}(C)$
for $i = 1$ **to** $a - 1$
  $L \leftarrow 2 \cdot L$
  $X[i] \leftarrow A[i] \oplus G \cdot Y[i-1] \oplus L \| 0^{n/2}$
  $Y[i] \leftarrow E_K(X[i])$
if $|A| \bmod n = 0$ **and** $A \neq \epsilon$ **then** $L \leftarrow 3 \cdot L$
else $L \leftarrow 3^2 \cdot L$
if $C = \epsilon$ **then** $L \leftarrow 3^2 \cdot L$
$X[a] \leftarrow A[a] \oplus G \cdot Y[a-1] \oplus L \| 0^{n/2}$
$Y[a] \leftarrow E_K(X[a])$
for $i = 1$ **to** $c - 1$
  $L \leftarrow 2 \cdot L$
  $M[i] \leftarrow Y[i + a - 1] \oplus C[i]$
  $X[i + a] \leftarrow M[i] \oplus G \cdot Y[i + a - 1] \oplus L \| 0^{n/2}$
  $Y[i + a] \leftarrow E_K(X[i + a])$
if $C \neq \epsilon$ **then**
  if $|C| \bmod n = 0$ **then**
    $L \leftarrow 3 \cdot L$
    $M[c] \leftarrow Y[a + c - 1] \oplus C[c]$
  else
    $L \leftarrow 3^2 \cdot L$, $c' \leftarrow |C| \bmod n$
    $M[c] \leftarrow \mathsf{Trunc}_{c'}(Y[a + c - 1] \oplus C[c]) \| 10^{n-c'-1}$
  $X[a + c] \leftarrow M[c] \oplus G \cdot Y[a + c - 1] \oplus L \| 0^{n/2}$
  $Y[a + c] \leftarrow E_K(X[a + c])$
  $M \leftarrow \mathsf{Trunc}_{|C|}(M[1] || \ldots || M[c])$
  $T' \leftarrow \mathsf{Trunc}_{\tau}(Y[a + c])$
else $M \leftarrow \epsilon$, $T' \leftarrow \mathsf{Trunc}_{\tau}(Y[a])$
if $T' = T$ **then return** M, **else return** $\perp$

Fig. 4: Algorithms of GIFT-COFB [3, Fig. 2.3]

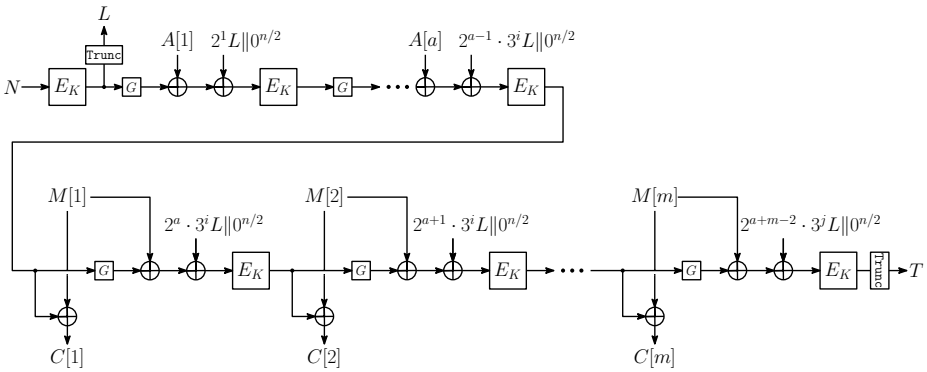

Fig. 5: GIFT-COFB.

Algorithm Photon-Beetle-$\mathcal{E}[r]_K(N, A, M)$

1. $\text{IV} \leftarrow N \,\|\, K;\ C \leftarrow \varepsilon$
2. **if** $(A = \varepsilon) \wedge (M = \varepsilon)$
3. $\quad T \leftarrow \mathsf{TAG}_{128}(\text{IV} \oplus 1)$; **return**$(\varepsilon, T)$
4. $c_0 \leftarrow ((M \neq \varepsilon) \wedge (r| \,|A|))?\ 1 : 2 : 3 : 4)$
5. $c_1 \leftarrow ((A \neq \varepsilon) \wedge (r| \,|M|))?\ 1 : 2 : 5 : 6)$
6. **if** $A \neq \varepsilon$
7. $\quad \text{IV} \leftarrow \mathsf{HASH}_r(\text{IV}, A, c_0)$
8. **if** $M \neq \varepsilon$
9. $\quad (M[1], \ldots, M[m]) \xleftarrow{r} M$
10. $\quad$ **for** $i = 1$ **to** m
11. $\quad\quad (Y, Z) \xleftarrow{r,256-r} \mathsf{Photon}_{256}(\text{IV})$
12. $\quad\quad (W, C[i]) \leftarrow \rho(Y, M[i])$
13. $\quad\quad \text{IV} \leftarrow W \,\|\, Z$
14. $\quad \text{IV} \leftarrow \text{IV} \oplus c_1$
15. $\quad C \leftarrow (C[1] \,\|\, \ldots \,\|\, C[m])$
16. $T \leftarrow \mathsf{TAG}_{128}(\text{IV})$
17. **return** (C, T)

Algorithm Photon-Beetle-$\mathcal{D}[r]_K(N, A, C, T)$

1. $\text{IV} \leftarrow N \,\|\, K;\ M \leftarrow \varepsilon$
2. **if** $(A = \varepsilon) \wedge (C = \varepsilon)$
3. $\quad T^* \leftarrow \mathsf{TAG}_{128}(\text{IV} \oplus 1)$
4. $\quad$ **return**$(T = T^*)?\ \varepsilon : \perp$
5. $c_0 \leftarrow ((C \neq \varepsilon) \wedge (r| \,|A|))?\ 1 : 2 : 3 : 4)$
6. $c_1 \leftarrow ((A \neq \varepsilon) \wedge (r| \,|C|))?\ 1 : 2 : 5 : 6)$
7. **if** $A \neq \varepsilon$
8. $\quad \text{IV} \leftarrow \mathsf{HASH}_r(\text{IV}, A, c_0)$
9. **if** $C \neq \varepsilon$
10. $\quad (C[1], \ldots, C[m]) \xleftarrow{r} C$
11. $\quad$ **for** $i = 1$ **to** m
12. $\quad\quad (Y, Z) \xleftarrow{r,256-r} \mathsf{Photon}_{256}(\text{IV})$
13. $\quad\quad (W, M[i]) \leftarrow \rho^{-1}(Y, C[i])$
14. $\quad\quad \text{IV} \leftarrow W \,\|\, Z$
15. $\quad \text{IV} \leftarrow \text{IV} \oplus c_1$
16. $\quad M \leftarrow (M[1] \,\|\, \ldots \,\|\, M[m])$
17. $T^* \leftarrow \mathsf{TAG}_{128}(\text{IV})$
18. **return** $(T = T^*)?\ M : \perp$

Algorithm $\mathsf{HASH}_r(\text{IV}, D, c_0)$

1. $D[1] \,\|\, \ldots \,\|\, D[d] \xleftarrow{r} \mathsf{ozs}_r(D)$
2. **for** $i = 1$ **to** d
3. $\quad (Y, Z) \xleftarrow{r,256-r} \mathsf{Photon}_{256}(\text{IV})$
4. $\quad W \leftarrow Y \oplus D[i]$
5. $\quad \text{IV} \leftarrow W \,\|\, Z$
6. $\text{IV} \leftarrow \text{IV} \oplus c_0$
7. **return** IV

Algorithm $\mathsf{TAG}_\tau(T[0])$

1. **for** $i = 1$ **to** $\lceil \tau/128 \rceil$
2. $\quad T[i] \leftarrow \mathsf{Photon}_{256}(T[i-1])$
3. $T \leftarrow \mathsf{Trunc}_{128}(T[1]) \,\|\, \ldots \,\|\, \mathsf{Trunc}_{128}(T[\tau/128])$
4. **return** T

Algorithm $\rho(S, U)$

1. $V \leftarrow \mathsf{Trunc}_{|U|}(\mathsf{Shuffle}(S)) \oplus U$
2. $S \leftarrow S \oplus \mathsf{ozs}_r(U)$
3. **return** (S, V)

Algorithm $\rho^{-1}(S, V)$

1. $U \leftarrow \mathsf{Trunc}_{|V|}(\mathsf{Shuffle}(S)) \oplus V$
2. $S \leftarrow S \oplus \mathsf{ozs}_r(U)$
3. **return** (S, U)

Fig. 6: Algorithms of Photon-Beetle [6, Fig. 3.6]

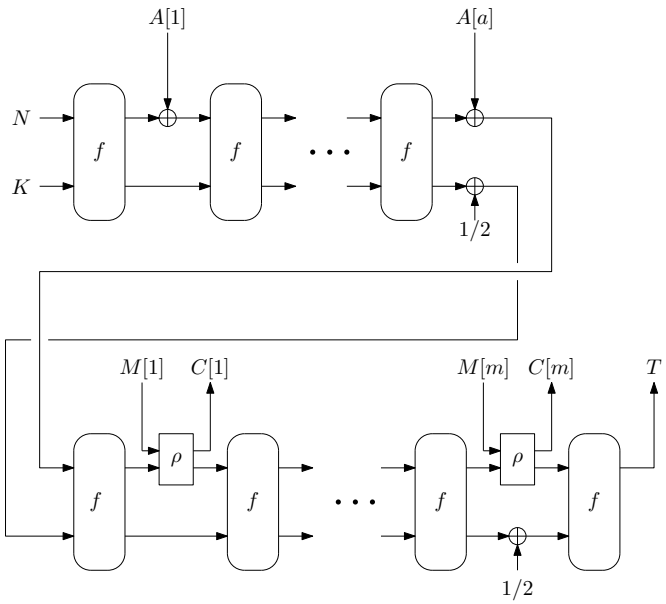

Fig. 7: Photon-Beetle.