Authenticated Encryption with Small Stretch (or, How to Accelerate AERO) [⋆]

Kazuhiko Minematsu

NEC Corporation, Japan
k-minematsu@ah.jp.nec.com

Abstract. Standard form of authenticated encryption (AE) requires the ciphertext to be expanded by the nonce and the authentication tag. These expansions can be problematic when messages are relatively short and communication cost is high. To overcome the problem we propose a new form of AE scheme, MiniAE, which expands the ciphertext only by the single variable integrating nonce and tag. An important feature of MiniAE is that it requires the receiver to be stateful not only for detecting replays but also for detecting forgery of any type. McGrew and Foley already proposed a scheme having this feature, called AERO, however, there is no formal security guarantee based on the provable security framework.

We provide a provable security analysis for MiniAE, and show several provably-secure schemes using standard symmetric crypto primitives. This covers a generalization of AERO, hence our results imply a provable security of AERO. Moreover, one of our schemes has a similar structure as OCB mode of operation and enables rate-1 operation, i.e. only one blockcipher call to process one input block. This implies that the computation cost of MiniAE can be as small as encryption-only schemes.

Keywords: Authenticated Encryption, Stateful Decryption, Provable Security, AERO, OCB

1 Introduction

Authenticated encryption (AE) is a symmetric-key cryptographic function for communication which provides both confidentiality and integrity of messages. A standard form of AE requires the ciphertext to be expanded by the amount of nonce, a never-repeating value maintained by the sender, and the authentication tag. This holds for popular schemes, such as CCM [2] and GCM [3]. The amount of expansion is small, say several dozen bytes. Nevertheless, it can be problematic when the messages are quite short. A typical example is wireless sensor network (WSN). For WSN, communication is much more energy-consuming than computation, and thus network packets are required to be very short. In fact, McGrew [4] provided examples of such real-life wireless protocols having maximum payload size ranging from 10 to 1K bytes. Struik [5] suggested that saving 8 bytes in communication may justify making encryption ten-times more expensive for WSN. Similar observation was given by Seys and Preneel [6]. Detailed energy cost evaluations for communication and cryptographic computation are given by [7,8].

As a solution to this problem, we propose MiniAE, a class of AE having smaller stretch than normal AEs. Its ciphertext is only expanded by the amount of single variable integrating nonce and tag. When ciphertext of MiniAE is stretched by s bits, it provides (about) s-bit authenticity and can securely encrypt at most 2^s messages, while nonce-based AE (NAE) needs $2s$-bit stretch for this purpose. A key difference from NAE is that MiniAE requires both sender and receiver to maintain a state (say counter), whereas NAE basically needs only the sender to be stateful. At first sight this might seem a big disadvantage, however, we remark that even NAE needs a stateful receiver when one wants to detect replays. In fact, replay detection via stateful receiver is employed by most of Internet protocols and wireless networks, such as Zigbee [1] and Bluetooth low energy (BLE) [2]. An important feature of MiniAE is that a receiver's state is not only used to detect replays but to detect forgeries of any other types. McGrew and Foley [9,4] already showed a similar idea and proposed a scheme called AERO. It can be seen as an encryption following encode-then-encipher (ETE) approach by Bellare and Rogaway [10], using XCB mode of operation [11] as its internal large keyed permutation. The approach of AERO is intuitively sound, however [9,4] do not provide a formal security analysis. As a consequence, it is not clear if ETE is essential for achieving the

[⋆] The proceeding version of this paper appears in ACISP 2016 [1]. This is the full version.

[1] http://www.zigbee.org

[2] http://www.bluetooth.com

goal, i.e. small stretch. This is undesirable since these schemes are likely to be used by resource-constrained devices.

We provide a formal model of MiniAE and basic security notions, namely the confidentiality, integrity and replay protection, and show provably secure constructions. Our model is different from Bellare, Kohno and Namprempre [12], which proposes a security model with stateful decryption tailored to analyze (a generalization of) SSH Binary Packet Protocol. More specifically, we propose three MiniAE schemes with concrete security proofs. The first scheme is based on ETE and can be seen as a simple generalization of AERO. This shows that AERO is indeed secure for our security notions. The second scheme, which we call $\mathsf{MiniCTR}$, is similar to a generic composition [13,14]. If it is instantiated by CTR mode encryption and a polynomial hash function, the computation cost of $\mathsf{MiniCTR}$ is almost the same as GCM [3]. The third scheme tries to further improve the efficiency. It is called $\mathsf{MiniOCB}$ for its structural similarity with OCB [15,16,17]. As well as OCB it is defined as a mode of tweakable blockcipher (TBC), and TBC can be instantiated by a blockcipher. It is parallelizable and rate-1, that is, it requires one blockcipher call for processing one plaintext block. The last two schemes show that ETE is not the exclusive approach to MiniAE, and a secure MiniAE can be as fast as nonce-based (unauthenticated) encryptions. We here stress that, unlike most NAEs, all our schemes are not capable of on-line encryption, and thus not desirable to handle long messages.

We remark that our basic security notions in Section 3.2 are extensions of standard NAE security notions, hence do not consider *misuse*. According to [9] AERO is expected to have a certain *misuse-resistance* beyond NAE. To fill the gap, Section 5 provides a short security analysis involving extended security notions covering misuse, and show a separation between the proposed schemes if we require these extended notions in addition to the basic ones.

2 Preliminaries

Let $\{0,1\}^*$ denote the set of all binary sequences including the empty string, ε. For $X \in \{0,1\}^*$, we write $|X|$ to denote the bit length of X, and let $|X|_n \overset{\text{def}}{=} \lceil |X|/n \rceil$. For $X, Y \in \{0,1\}^*$ we write $X \| Y$ to denote their concatenation. The first (last) i bits of X is denoted by $\mathtt{msb}_i(X)$ ($\mathtt{lsb}_i(X)$). We have $\mathtt{msb}_0(X) = \varepsilon$ and $\varepsilon \oplus X = \varepsilon$ for any X. For any $s > 0$, a partition of X into s-bit blocks is written as $(X[1], \ldots, X[x]) \overset{s}{\leftarrow} X$, where $|X[i]| = s$ for $i < x$ and $|X[x]| \leq s$. For $X = \varepsilon$, we let $X[1] \overset{s}{\leftarrow} X$ with $X[1] = \varepsilon$. Moreover, for X and Y such that $|X| \leq n$ and $|X| + |Y| \geq n$ we write $(\underline{X}, \underline{Y}) \overset{n}{\leftarrow} (X, Y)$ to denote the parsing into $\underline{X} = X \| \mathtt{msb}_{n-|X|}(Y)$ and $\underline{Y} = \mathtt{lsb}_{|Y|-(n-|X|)} Y$. The inverse parsing is written as $(X, Y) \overset{m}{\leftharpoondown} (\underline{X}, \underline{Y})$, where $|\underline{X}| \geq m$ and $X = \mathtt{msb}_m(\underline{X})$, $Y = \mathtt{lsb}_{|\underline{X}|-m}(\underline{X} \| \underline{Y})$. By writing $X10^*$ for $0 \leq |X| < n$ we mean a padding $10^{n-|X|-1}$ to X. We have $X10^* = X$ when $|X| = n$, and $\varepsilon 10^* = 10^{n-1}$. For a finite set $\mathcal{X}$ we write $X \overset{\$}{\leftarrow} \mathcal{X}$ to mean the uniform sampling of X over $\mathcal{X}$.

For keyed function $F : \mathcal{K} \times \mathcal{X} \rightarrow \mathcal{Y}$ with key $K \in \mathcal{K}$, we may simply write $F_K : \mathcal{X} \rightarrow \mathcal{Y}$ if key space is obvious, or even write as $F : \mathcal{X} \rightarrow \mathcal{Y}$ if being keyed is obvious. If $E_K : \mathcal{X} \rightarrow \mathcal{X}$ is a keyed permutation, or a blockcipher, E_K is a permutation over $\mathcal{X}$ for every $K \in \mathcal{K}$. Its inverse is denoted by E_K^{-1}. A tweakable keyed permutation or TBC [18], $\widetilde{E}_K : \mathcal{T} \times \mathcal{X} \rightarrow \mathcal{X}$, is a family of keyed permutation over $\mathcal{X}$ indexed by tweak $T \in \mathcal{T}$ and its encryption is written as $C = \widetilde{E}_K^T(M)$ for plaintext M, tweak T and ciphertext C. The decryption is written as $M = \widetilde{E}_K^{-1,T}(C)$. We consider $\mathcal{X}$ to be either a set of fixed length strings or variable length strings (though the original definition [18] assumes the fixed length). The latter is also called tweakable enciphering scheme (TES).

Random Functions. Let $\mathrm{Func}(n, m)$ be the set of all functions $\{0,1\}^n \rightarrow \{0,1\}^m$, and let $\mathrm{Perm}(n)$ be the set of all permutations over $\{0,1\}^n$. A uniform random function (URF) having n-bit input and m-bit output is a function family uniformly distributed over $\mathrm{Func}(n, m)$. It is denoted by $\mathsf{R} \overset{\$}{\leftarrow} \mathrm{Func}(n, m)$. An n-bit uniform random permutation (URP), denoted by P, is similarly defined as $\mathsf{P} \overset{\$}{\leftarrow} \mathrm{Perm}(n)$. We also define tweakable URP. Let $\mathcal{T}$ be a set of tweak and $\mathrm{Perm}^{\mathcal{T}}(n)$ be the set of all functions such that for any $f \in \mathrm{Perm}^{\mathcal{T}}(n)$ and $t \in \mathcal{T}$, $f(t, *)$ is a permutation. A tweakable n-bit URP with tweak $T \in \mathcal{T}$ is defined as $\widetilde{\mathsf{P}} \overset{\$}{\leftarrow} \mathrm{Perm}^{\mathcal{T}}(n)$.

Pseudorandom Function. For c oracles, $O_1, O_2, \ldots, O_c$, we write $\mathcal{A}^{O_1, O_2, \ldots, O_c}$ to represent the adversary $\mathcal{A}$ accessing these c oracles in an arbitrarily order. If O and O' are oracles having the same input and output domains, we say they are compatible. Let $F_K : \{0,1\}^n \rightarrow \{0,1\}^m$ and $G_{K'} : \{0,1\}^n \rightarrow \{0,1\}^m$ be two compatible keyed functions, with $K \in \mathcal{K}$ and $K' \in \mathcal{K}'$ (key spaces are not necessarily the same). Let

$\mathcal{A}$ be an adversary trying distinguish them using queries. Then the advantage of $\mathcal{A}$ is defined as

$$\begin{aligned}\mathtt{Adv}^{\mathtt{cpa}}_{F_K,G_{K'}}(\mathcal{A}) &\overset{\text{def}}{=} \Pr[\mathcal{A}^{F_K} \Rightarrow 1] - \Pr[\mathcal{A}^{G_{K'}} \Rightarrow 1],\\ \mathtt{Adv}^{\mathtt{cca}}_{F_K,G_{K'}}(\mathcal{A}) &\overset{\text{def}}{=} \Pr[\mathcal{A}^{F_K,F_K^{-1}} \Rightarrow 1] - \Pr[\mathcal{A}^{G_{K'},G_{K'}^{-1}} \Rightarrow 1],\end{aligned}$$

where the latter is defined if F and G are keyed permutation, and probabilities are defined over uniform samplings of keys and internal randomness of $\mathcal{A}$. If F and G are tweakable, a tweak for a query is arbitrarily chosen by the adversary for both $\mathtt{Adv}^{\mathtt{cpa}}$ and $\mathtt{Adv}^{\mathtt{cca}}$. For URF R compatible to F, let $\mathtt{Adv}^{\mathtt{prf}}_{F_K}(\mathcal{A}) \overset{\text{def}}{=} \mathtt{Adv}^{\mathtt{cpa}}_{F_K,\mathsf{R}}(\mathcal{A})$. In a similar manner, let tweakable URP $\widetilde{\mathsf{P}}$ compatible to TBC $\widetilde{E}_K$. Then we define

$$\mathtt{Adv}^{\mathtt{tprp}}_{\widetilde{E}_K}(\mathcal{A}) \overset{\text{def}}{=} \mathtt{Adv}^{\mathtt{cpa}}_{\widetilde{E}_K,\widetilde{\mathsf{P}}}(\mathcal{A}), \text{ and } \mathtt{Adv}^{\mathtt{tsprp}}_{\widetilde{E}_K}(\mathcal{A}) \overset{\text{def}}{=} \mathtt{Adv}^{\mathtt{cca}}_{\widetilde{E}_K,\widetilde{\mathsf{P}}}(\mathcal{A}),$$

We further extends these notions to the functions (or permutations) having variable-input length (VIL). For example, if F_K is a VIL keyed function : $\{0,1\}^* \to \{0,1\}^n$ we define $\mathtt{Adv}^{\mathtt{prf}}_{F_K}(\mathcal{A})$ as $\mathtt{Adv}^{\mathtt{cpa}}_{F_K,\mathsf{R}^*}(\mathcal{A})$, where R^* is an URF compatible to F_K which can be implemented by lazy sampling.

Time Complexity. If adversary $\mathcal{A}$ is with time complexity t, it means the total computation time and memory of $\mathcal{A}$ required for query generation and final decision, in some fixed model. If there is no description on time complexity of $\mathcal{A}$, it means $\mathcal{A}$ has no computational restriction. Conventionally we say F_K is a pseudorandom function (PRF) if $\mathtt{Adv}^{\mathtt{prf}}_{F_K}(\mathcal{A})$ is negligible for all practical adversaries (though the formal definition requires F_K to be a function family). Similarly we say F_K is a pseudorandom permutation (PRP) if $\mathtt{Adv}^{\mathtt{prp}}_{F_K}(\mathcal{A})$ is negligible and F_K is invertible. Strong PRP (SPRP), tweakable PRP (TPRP) and tweakable SPRP (TSPRP) are defined in a similar manner.

Universal Hash Function. Let $H : \mathcal{K} \times \mathcal{X} \to \{0,1\}^n$ be a keyed function, where key K is uniform over $\mathcal{K}$ and $\mathcal{X} \subseteq \{0,1\}^*$. We say H_K is $\epsilon(x)$-almost XOR universal (AXU) if

$$\max_{c\in\{0,1\}^n} \Pr_K[H_K(X) \oplus H_K(X') = c] \leq \epsilon(x) \tag{1}$$

holds for any distinct $X, X' \in \mathcal{X}$ with $\max\{|X|_n, |X'|_n\} = x$ and for some $\epsilon(*)$. If input is divided into two parts, e.g. $X = (X_1, X_2)$, $|X|_n$ means $|X_1|_n + |X_2|_n$.

Building TBC. All our constructions will use TBC. It can be built from scratch [19,20,21] or from a blockcipher. Suppose we want a TBC of n-bit block and tweak space (which is assumed to be a set of binary strings) $\mathcal{T}$. From the result of [18], using an n-bit blockcipher E_K and an independently-keyed $\epsilon(x)$-AXU hash function, $H_{K'} : \mathcal{T} \to \{0,1\}^n$, we can build TBC as

$$\widetilde{E}^T_{K,K'}(M) = E_K(S \oplus M) \oplus S, \text{ where } S = H_{K'}(T) \tag{2}$$

for encryption of plaintext M and tweak T. This has a TSPRP-advantage of $O(\epsilon(\ell) \cdot q^2)$ plus a CCA-advantage of E_K, for any adversary with q CCA queries using tweak of maximum block length ℓ. Typically we can use a polynomial hash function defined over GF(2^n) as a universal hash fulfilling (1) with $\epsilon(x) = x/2^n$. Alternatively we can use PRF as a computational counterpart, say CMAC. In some cases the use of two keys in (2) can be reduced to one [16,22].

3 Definition of MiniAE

3.1 Basic Model

The encryption function of MiniAE accepts nonce N, associated data (AD) A, and plaintext M, and generates ciphertext C and *encrypted nonce* L, where $N, L \in \mathcal{N}_{ae} = \{0,1\}^\nu$ for some fixed ν, $A \in \mathcal{A}_{ae}$, $M \in \mathcal{M}_{ae}$ with $|C| = |M|$. Typically $\mathcal{A}_{ae} = \mathcal{M}_{ae} = \{0,1\}^*$ and we may simply write $\mathcal{M}$ for $\mathcal{M}_{ae}$. A message sent over a communication channel is (A, L, C). Thus the expansion is ν bits. AD A and plaintext M can be empty, and if M is empty the corresponding C is also empty. We require unique nonce for each encryption. We define nonce increment function $\mu : \mathcal{N}_{ae} \to \mathcal{N}_{ae}$, which is a permutation over $\mathcal{N}_{ae}$ and has single cycle of length $|\mathcal{N}_{ae}|$. We assume μ and initial nonce value are public and fixed. If N is the nonce last used in encryption, the next nonce is $\mu(N)$. Typically, μ is a counter increment $\mu(N) = N + 1$ where $+$ is modulo 2^ν. As mentioned earlier we assume stateful decryption. On receiving

(A', L', C'), the stateful decryption function first computes the decrypted nonce $N' \in \mathcal{N}_{ae}$ using the key, and outputs the decrypted plaintext M' if N' is considered as valid, otherwise the default error symbol, $\perp$. The validity of N' is determined by comparison with the receiver state. Here stateful decryption is essential to detect replays, and we assume the receiver state is uniquely determined by the nonce in the previous successful decryption (thus a state is an element of $\mathcal{N}_{ae}$), which is typical in many replay protection schemes including AERO [9][3]. More generally, the receiver has a set of expected nonce values for each decryption. The set is defined as a function of the receiver state, and we write the function as $\rho : \mathcal{N}_{ae} \to 2^{\mathcal{N}_{ae}}$, where $2^{\mathcal{N}_{ae}}$ is the power set of $\mathcal{N}_{ae}$. The function ρ is public, and when N' is the value obtained by the decryption and $\widehat{N}$ is the last nonce accepted as valid, N' is determined as valid iff $N' \in \rho(\widehat{N})$ holds true. In this paper we assume $|\rho(N)| \le \omega$ holds for any N, where ω is called *verification range size*. Let us write i-th nonce used at encryption as N_i (e.g. $N_2 = \mu(N_1)$). Naturally we require that $N_{i+1}(= \mu(N_i)) \in \rho(N_i)$ for any i to accept the genuine ciphertext, and $N_j \notin \rho(N_i)$ for any $j \le i$ to reject replays without fail. In practice ρ determines the resilience against packet loss. If the synchronization is perfect between the sender and receiver, the simplest setting as $\rho(N) = \mu(N)$ with $\omega = 1$ works fine. However we often need to include $\{N_j\}$ for some $j > i+1$ for $\rho(N_i)$ when packets can lost in the channel. In this case $\rho(N) = \{\mu(N), \mu(\mu(N)), ...\}$ to tolerate the loss of consecutive $\omega - 1$ packets. This will increase a chance of success at forgery, roughly by a factor of ω.

Nonce Shorter than Block. As all of our constructions are defined over n-bit blocks for some n (say 128), we require $\nu \le n$, and $|N| + |M| \ge n$ holds, which means if $\nu < n$ we have a nonzero limit on the minimum plaintext length, or, in practice we may pad as [9]. Throughout the paper, we may implicitly use $(\underline{N}, \underline{M})$ to denote the result of parsing $(\underline{N}, \underline{M}) \xleftarrow{n} (N, M)$, provided N and M are clear from the context. Similarly we may use $(\underline{L}, \underline{C})$ to denote the result of parsing $(\underline{L}, \underline{C}) \xleftarrow{n} (L, C)$. We remark that when $\nu = n$, we have $\underline{N} = N$, $\underline{M} = M$ and $\underline{L} = L$, $\underline{C} = C$.

3.2 Security Notions

Following NAE security notions, we introduce two security notions, namely privacy and authenticity, to model the security of MiniAE. Here privacy notion reflects the pseudorandomness of ciphertexts, and authenticity notion reflects the hardness of forgery even if the receiver state are chosen by the adversary. We think this form of authenticity will be beneficial for its simplicity, strong assurance, and independence of the details of state management[4]. Let **MiAE** be an MiniAE with ν-bit nonce (with some key $K \xleftarrow{\$} \mathcal{K}$). The encryption and decryption algorithms are **MiAE**-$\mathcal{E}$ and **MiAE**-$\mathcal{D}$. Following Section 3.1, **MiAE**-$\mathcal{E}$ takes (N, A, M) and returns (L, C) with $|M| = |C|$ and $|N| = |L| = \nu$. **MiAE**-$\mathcal{D}$ takes $(\widehat{N}, A', L', C')$ with $|\widehat{N}| = |L'| = \nu$, where $\widehat{N}$ is a receiver state (i.e. a decrypted nonce) guessed by adversary. In practice $\widehat{N}$ is not sent over the communication channel. **MiAE**-$\mathcal{D}$ then computes the decrypted nonce, N', and see if $N' \in \rho(\widehat{N})$. If true it returns a decrypted message $M' \in \mathcal{M}_{ae}$ and otherwise $\perp$. Thus we have

$$\begin{aligned} (L, C) &\leftarrow \textbf{MiAE-}\mathcal{E}(N, A, M) \\ M'/\perp &\leftarrow \textbf{MiAE-}\mathcal{D}(\widehat{N}, A', L', C'). \end{aligned}$$

Privacy notion. Let $\mathcal{A}$ be a Priv-adversary who accesses **MiAE**-$\mathcal{E}$ using q encryption queries with distinct nonces (i.e. nonce-respecting). Here we assume nonces are not necessarily updated[5] by μ, in the same manner to [23]. The privacy notion for Priv-adversary $\mathcal{A}$ is defined as

$$\mathtt{Adv}^{\mathtt{priv}}_{\mathbf{MiAE}}(\mathcal{A}) \overset{\text{def}}{=} \Pr[\mathcal{A}^{\mathbf{MiAE}\text{-}\mathcal{E}} \Rightarrow 1] - \Pr[\mathcal{A}^{\$} \Rightarrow 1], \tag{3}$$

where random-bit oracle, \$, takes (N, A, M) and returns $(L, C) \xleftarrow{\$} \{0,1\}^{\nu} \times \{0,1\}^{|M|}$.

Authenticity notion. Let $\mathcal{A}$ be an Auth-adversary against **MiAE**. We write q encryption queries as $(N_1, A_1, M_1), \ldots, (N_q, A_q, M_q)$, and q' decryption queries as $(\widehat{N}_1, A'_1, L'_1, C'_1), \ldots, (\widehat{N}_{q'}, A'_{q'}, L'_{q'}, C'_{q'})$. We may say verification queries instead of decryption queries. We also let $(L_1, C_1), \ldots, (L_q, C_q)$ be the corresponding oracle answers for encryption queries. We assume $\mathcal{A}$ follows the two conditions.

[3] Decryption of [9] also maintains the most recent invalid nonce, in order to do resynchronization.

[4] In this sense our notions are similar to Rogaway's nonce-based encryption [23] as it allows a provable security analysis without taking into account the details of nonce generation.

[5] It is possible to define the adversary in our security notions strictly following the generation of nonce described at Section 3.1. Here we employ a more general definition for the simplicity.

Condition 1: Adversary is nonce-respecting for encryption queries (i.e. $N_i \neq N_j$ for any $i \neq j$)
Condition 2: For all $i = 1, \ldots, q'$, $(A'_i, L'_i, C'_i) \neq (A_j, L_j, C_j)$ holds for all j-th encryption queries before the i-th decryption query.

As well as the privacy notion, nonces in the encryption queries are not necessarily generated by μ. The second condition excludes the adversary's trivial win including a replay, that is, a decryption query $(\widehat{N}, A', L', C')$ with $(A', L', C') = (A, L, C)$ with $N \in \rho(\widehat{N})$ for some previous encryption query (N, A, M) and response (L, C). This is because a replay is always detected at the decryption side in actual use of *any* MiniAE scheme following Section 3.1. We also excluded the case $N \notin \rho(\widehat{N})$, as it will be always rejected (thus trivial loss). The authenticity notion is defined as

$$\mathtt{Adv}^{\mathtt{auth}}_{\mathbf{MiAE}}(\mathcal{A}) \overset{\text{def}}{=} \Pr[\mathcal{A}^{\mathbf{MiAE}\text{-}\mathcal{E}, \mathbf{MiAE}\text{-}\mathcal{D}} \text{ forges }], \tag{4}$$

where $\mathcal{A}$ forges if $\mathbf{MiAE}$-$\mathcal{D}$ returns output other than $\perp$ for a decryption query.

For both privacy and authenticity notions, we write the total input blocks, denoted by σ, to mean $\sum_i |N_i|_n + |A_i|_n + |M_i|_n$ for the privacy notion, and $\sum_i |N_i|_n + |A_i|_n + |M_i|_n + \sum_j |L'_j|_n + |A'_j|_n + |C'_j|_n$ for the authenticity notion.

MiniAE **as a large tweakable random permutation.** Since **MiAE** is a tweakable keyed permutation in general (where tweak is used for AD), we write $\mathtt{Adv}^{\mathtt{tprp}}_{\mathbf{MiAE}}(\mathcal{A})$ and $\mathtt{Adv}^{\mathtt{tsprp}}_{\mathbf{MiAE}}(\mathcal{A})$ to denote TPRP and TSPRP advantages of the underlying tweakable keyed permutation $\widetilde{\mathbf{E}}$. Note that $\widetilde{\mathbf{E}}$ is not always required to be strong with respect to these notions. In fact our results show that it can be much weaker.

IV-based Encryption. We also define IV-based encryption scheme: $\mathit{\Pi}_K : \mathcal{I} \times \mathcal{M} \to \mathcal{M}$ which is a permutation over $\mathcal{M}$ determined by $K \in \mathcal{K}$ and fixed-length initialization-vector (IV) $I \in \mathcal{I}$. Here IV is sampled uniformly random for every encryption. Let $\mathit{\Pi}_K$ oracle as the encryption oracle take $M \in \mathcal{M}$ and return (I, C), where $I \overset{\$}{\leftarrow} \mathcal{I}$ and $C \to \mathit{\Pi}_K(I, M)$. We remark that the adversary is not allowed to see I before querying M. We define the PRIV\$ advantage as the indistinguishability of $\mathit{\Pi}_K$ from the random-bit oracle (\$), which returns $|I| + |M|$-bit random sequence, i.e.,

$$\mathtt{Adv}^{\mathtt{priv\$}}_{\mathit{\Pi}_K}(\mathcal{A}) \overset{\text{def}}{=} \Pr[\mathcal{A}^{\mathit{\Pi}_K} \Rightarrow 1] - \Pr[\mathcal{A}^{\$} \Rightarrow 1]. \tag{5}$$

We say the total input block of $\mathcal{A}$ to mean the total number of plaintext blocks.

3.3 Remarks

Comparison with NAE security notions. Our security notions are quite similar to NAE security notions, e.g. [13,24]. For privacy notion, both NAE and MiniAE require that the outputs of encryption oracle are pseudorandom. For authenticity notion, both NAE and MiniAE require that a forgery is hard for nonce-respecting adversary. The standard authenticity notion for NAE considers stateless receiver, however if a certain NAE scheme is secure with respect to the standard authenticity notion, then it certainly detects replays if receiver is stateful and nonce is dealt with μ and ρ as described at Section 3.1. We remark that, when the receiver loses state NAE still can detect forgeries other than replays, while MiniAE can not: only unverified decryption is possible.

Comparison with alternative solutions. If sender and receiver are completely synchronized, we can use NAE and simply omit the nonce to be sent to save bandwidth (also called implicit sequence number [4]). However this is problematic when packets may lost. A mitigation is to send a partial information. This technique is employed by some popular protocols as [4] shows. Therefore it basically works for some settings, however it makes the messaging format dependent on the number of tolerable packet lost, which depends on the network condition and application and sometimes hard to determine in practice. Moreover, once the receiver loses the state, even the unverified decryption becomes impossible. In contrast, MiniAE allows ad-hoc mechanisms to handle packet lost without changing the message format, and unverified decryption without state, and allows efficient built-in resynchronization as shown by AERO.

Another solution to suppress expansion is Deterministic AE (DAE) proposed by Rogaway and Shrimpton [25]. In DAE there is no nonce and for plaintext M the encryption output is (C, T) where $|C| = |M|$ and T is the authentication tag of fixed length[6]. Since DAE encryption is deterministic, the standard privacy notion is impossible to achieve. DAE can prevent replay if the receiver keeps (hash values of)

[6] If DAE takes nonce as its input we call it MRAE (misuse-resistant AE) which has the same expansion as NAE.

all received ciphertexts, or using Bloom filter allowing some false negatives. Either option requires much larger memories or computations than the verification of MiniAE. Table 1 summarizes encryption schemes in the presence of stateful receiver for replay protection.

Table 1. Comparison of Encryption Schemes.

Scheme	Expansion	Privacy	Authenticity	Replay Protect	Dec w/o state
nonce-based Enc	$\|N\|$	✓	-	✓	✓
NAE	$\|N\| + \|T\|$	✓	✓	✓	✓
DAE	$\|T\|$	-	✓	difficult	✓
NAE+Nonce omit	$\|T\|$	✓	✓	✓	-
MiniAE	$\|N\|$	✓	✓	✓	✓

3.4 Applications to Low-power Wireless Sensor Network

To suppress communication overhead, link-layer security protocols for low-power WSN often employed NAE having a short nonce and short tag (see [26] for a good survey). We here present some examples.

– Zigbee for IEEE 802.15.4 uses AES-CCM. Typically a nonce is a concatenation of 4-byte frame counter (FC), 8-byte source address (SA) and 1-byte security control (SC), total 13 bytes, and tag has 4 bytes [27].
– BLE also uses AES-CCM with 13-byte nonce consisting of 39-bit counter, 1-bit direction indicator, and two 32-bit device IDs for sender and receiver devices. The tag is 4 bytes.
– Lightweight link-layer security protocols such as [28,29,30,31] use AEs with 2-byte counter and several other information to form nonce and has 4-byte tag.

For all of the above cases nonce N consists of a counter ctr and supplemental information sup. Here sup should not be encrypted and possibly repeated, hence it is rather attributed as AD. Suppose we have an NAE with $|\mathsf{ctr}| = \nu$, $|\mathsf{sup}| = a$, and tag length $|\mathsf{tag}| = \tau$. This expands the ciphertext by $\nu + \tau$ bits (Baseline, left of Fig. 1), excluding AD. Then there are two application scenarios of MiniAE. First, if both ν-bit counter and τ-bit tag are considered too short, MiniAE having $(\nu + \tau)$-bit nonce will enlarge the counter and tag spaces, keeping the same expansion (Scenario 1, middle of Fig. 1). Here sup is moved to AD. If ν-bit counter is sufficiently long and $\tau \le \nu$, MiniAE with ν-bit nonce will remove the expansion caused by tag, without harming the authenticity strength (Scenario 2, right of Fig. 1). For example, keeping Zigbee's communication overhead, MiniAE of scenario 1 realizes 64-bit counter and 64-bit authenticity, and for BLE, MiniAE of scenario 2 reduces the 4-byte expansion from each packet keeping the use of 39-bit counter. For the security protocols of [28,29,30,31] MiniAE of scenario 1 allows 48-bit counter and 48-bit authenticity. We finally remark that [9] suggests to use AERO for reducing the overhead of Internet protocols : for IPsec, using a standard such as ESP AES-GCM would cause more than 24-byte expansion, while AERO can reduce it to 12-byte. A secure MiniAE scheme has the same effect.

4 Building MiniAE

In this section we provide constructions of MiniAE. Throughout the section all schemes are assumed to have nonce of ν bits and verification range size ω. We assume ν and ω are fixed parameters, and also assume $\nu \le n$ for some fixed block length n, and plaintext M used in a scheme satisfies $|M| \ge \nu - n$.

4.1 MiniAE from large tweakable blockcipher

We start with a naive solution based on ETE approach mentioned earlier. We call the scheme MiniETE. More specifically, let $\widetilde{\mathbf{E}} : \mathcal{T} \times \mathcal{M} \to \mathcal{M}$ be a TBC, where $\mathcal{M} = \bigcup_{i \ge n} \{0,1\}^i$. The encryption of MiniETE using $\widetilde{\mathbf{E}}$ is defined as $(L \| C) = \widetilde{\mathbf{E}}^A(N \| M)$. For decryption, we perform $(N' \| M') = \widetilde{\mathbf{E}}^{-1,A}(L \| C)$ and see if $N' \in \rho(\widehat{N})$. This scheme is provably secure if $\widetilde{\mathbf{E}}$ is a TSPRP. Concrete security bounds of MiniETE are shown in the following propositions. Here, the proof of Proposition 1 is trivial and that of Proposition 2 is easily obtained as a variant of the proof of Theorem 2 thus we omit it here.

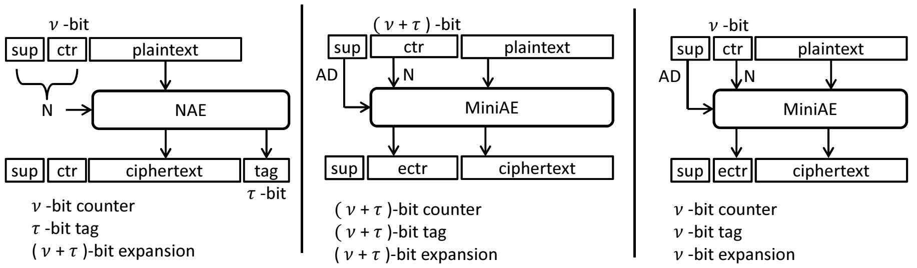

Fig. 1. (Left) Baseline, a nonce-based AE in WSN whose counter and tag may be short. (Middle) Scenario 1, MiniAE with a longer counter enables larger counter space and stronger authenticity while keeping the same overhead. (Right) Scenario 2, MiniAE has the same counter as Baseline, but reduces the overhead without harming the authenticity if counter is not shorter than tag.

Proposition 1. *Let* $\mathsf{MiniETE}[\widetilde{\mathbf{E}}]$ *be* $\mathsf{MiniETE}$ *using* $\widetilde{\mathbf{E}}$. *If* $\mathcal{A}$ *is a Priv-adversary with* q *encryption queries and* σ *total input blocks and time complexity* t, *we have*

$$\mathtt{Adv}^{\mathtt{priv}}_{\mathsf{MiniETE}[\widetilde{\mathbf{E}}]}(\mathcal{A}) \le \mathtt{Adv}^{\mathtt{tprp}}_{\widetilde{\mathbf{E}}}(\mathcal{B}) + \frac{q^2}{2^{n+1}},$$

where $\mathcal{B}$ *uses* q *encryption queries with* σ *total input blocks and time complexity* $t' = t + O(\sigma)$.

Proposition 2. *Let* $\mathcal{A}$ *be an Auth-adversary with* q *encryption and* q' *decryption queries,* σ *total input blocks and time complexity* t. *We assume* $(q + q') < 2^{\nu-1}$. *Then we have*

$$\mathtt{Adv}^{\mathtt{auth}}_{\mathsf{MiniETE}[\widetilde{\mathbf{E}}]}(\mathcal{A}) \le \mathtt{Adv}^{\mathtt{tsprp}}_{\widetilde{\mathbf{E}}}(\mathcal{B}) + \frac{2(q + q')(\omega + q')}{2^{\nu}},$$

where $\mathcal{B}$ *uses* q *encryption queries and* q' *decryption queries with* σ *total input blocks and time complexity* $t' = t + O(\sigma)$.

For instantiations of $\widetilde{\mathbf{E}}$, we could use known schemes [32,33,34,35] as internal wide-block TBC. As mentioned, this scheme is in fact a generalization of AERO which uses XCB [11] with AES-128. That is, AERO is provably secure in our security model, although there are minor differences and additional features[7]. MiniETE also has some similarities with ETE-based AE schemes, such as AEZ [36] and PIV [35].

4.2 MiniAE from encrypted counter

MiniETE is conceptually simple, however actual computation cost is rather high. A popular approach to $\widetilde{\mathbf{E}}$ shown by the seminal paper by Naor and Reingold [37] uses two universal hashing layers with one encryption layer, called Hash-Enc-Hash [32,38,39]. EME and CMC [33,34] do not use universal hash but require two blockcipher calls for each n-bit input block.

To improve the efficiency, we present a two-pass scheme which we call MiniCTR. The name comes from that it consists of encryption of nonce and an additive encryption. Specifically, MiniCTR uses an n-bit block, variable-length tweak TBC $\widetilde{E}_K : \mathcal{T} \times \{0,1\}^n \to \{0,1\}^n$ and a keyed function of n-bit input and variable-length output, $F_{K'} : \{0,1\}^n \to \{0,1\}^*$. Here we assume $\mathcal{T}$ is sufficiently large to encode a pair $(A, \underline{M})$. Two keys, K and K', are assumed to be independent. The algorithms of MiniCTR are shown in Fig. 2 and the encryption is also shown in Fig. 3. We write $\Pi[F_{K'}]$ to denote the underlying additive encryption, where $\underline{L}$ is used as n-bit IV. In Theorems 1 and 2 below, we prove the security of MiniCTR when $\widetilde{E}$ is TSPRP-secure and $\Pi[F_{K'}]$ is PRIV\$-secure.

DAE does not work. The presented scheme has a similar structure as DAE schemes [25,40,41] or randomized encryption by Desai [42]. However we can not directly use them as MiniAE. For example, (a generic form of) DAE with n-bit tag is obtained by changing line 2 of Fig. 2 as a Feistel round $\underline{L} \leftarrow \underline{N} \oplus F'(A, M)$ with $\underline{N}$ fixed to 0^n using another PRF F'. However the privacy of this scheme is easily broken if we query (N, A, M) and $(N \oplus c, A, M)$ for some non-constant c: the corresponding pair of $\underline{L}$ has a fixed difference c.

```
Algorithm MiniCTR-$\mathcal{E}_{\widetilde{E},F}(N,A,M)$

$(\underline{N},\underline{M}) \overset{n}{\leftharpoonup} (N,M)$
$\underline{L} \leftarrow \widetilde{E}^{(A,\underline{M})}(\underline{N})$
$\underline{C} \leftarrow F(\underline{L}) \oplus \underline{M}$
$(L,C) \overset{\nu}{\leftharpoondown} (\underline{L},\underline{C})$
return $(L,C)$
```

```
Algorithm MiniCTR-$\mathcal{D}_{\widetilde{E},F}(\widehat{N},A',L',C')$

$(\underline{L}',\underline{C}') \overset{n}{\leftharpoonup} (L',C')$
$\underline{M}' \leftarrow F(\underline{L}') \oplus \underline{C}'$
$\underline{N}' \leftarrow \widetilde{E}^{-1(A',\underline{M}')}(\underline{L}')$
if $\mathtt{msb}_\nu(\underline{N}') \in \rho(\widehat{N})$ then
    $(N',M') \overset{\nu}{\leftharpoondown} (\underline{N}',\underline{M}')$
    return $M'$
else return $\perp$
```

Fig. 2. Encryption and decryption algorithms of MiniCTR$[\widetilde{E},F]$

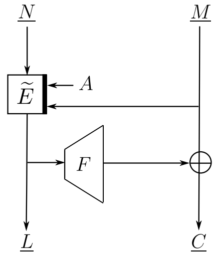

Fig. 3. The encryption algorithm of MiniCTR$[\widetilde{E},F]$, except the pre- and post-parsings.

Security. Let MiniCTR$[\widetilde{E},F]$ be MiniCTR using TBC $\widetilde{E}_K$ and $F_{K'}$. The security bounds for MiniCTR$[\widetilde{E},F]$ are presented in the following theorems.

Theorem 1. *If $\mathcal{A}$ is a Priv-adversary with q encryption queries and σ total input blocks and time complexity t, we have*

$$\mathtt{Adv}^{\mathtt{priv}}_{\mathsf{MiniCTR}[\widetilde{E},F]}(\mathcal{A}) \le \mathtt{Adv}^{\mathtt{tprp}}_{\widetilde{E}}(\mathcal{B}) + \mathtt{Adv}^{\mathtt{priv\$}}_{\Pi[F]}(\mathcal{C}) + \frac{q^2}{2^{n+1}}.$$

where $\mathcal{B}$ uses q queries with total input blocks σ and time complexity $t' = t + O(\sigma)$, and $\mathcal{C}$ uses q queries and σ total input blocks with time complexity $t' = t + O(\sigma)$.

Theorem 2. *Let $\mathcal{A}$ be an Auth-adversary with q encryption queries, q' decryption queries, σ total input blocks, and time complexity t. We assume $(q + q') < 2^{\nu-1}$. Then we have*

$$\mathtt{Adv}^{\mathtt{auth}}_{\mathsf{MiniCTR}[\widetilde{E},F]}(\mathcal{A}) \le \mathtt{Adv}^{\mathtt{tsprp}}_{\widetilde{E}}(\mathcal{B}) + \frac{2(q+q')(\omega+q')}{2^\nu},$$

where $\mathcal{B}$ uses q encryption queries and q' decryption queries, having time complexity $t' = t + O(\sigma)$.

We remark that the authenticity does not require any security property of F here: the reason is simple, since the authenticity of N is guaranteed even when adversary can access the key of F. The proofs of Theorems 1 and 2 are presented in Appendix A.

Instantiation. Typically, $\widetilde{E}$ is instantiated by n-bit blockcipher and n-bit polynomial hash with (2) and F is instantiated by CTR mode, e.g. $C = \Pi[F_{K'}](\underline{L}, M)$ with $C[i] = E_{K'}(\underline{N} \oplus i) \oplus M[i]$ for $i = 1, 2, \dots$ using blockcipher $E_{K'}$. In this case, the computation cost of MiniCTR for each n-bit plaintext block is one GF(2^n) multiplication and one blockcipher call, which is roughly the same as GCM. Combined with [18,22] and standard security result for CTR mode, e.g. [43], we can prove the birthday-type bounds of MiniCTR comparable to those of GCM [44] both for privacy and authenticity[8]. For GCM, 12-byte nonce and 16-byte tag is a popular setting, and MiniCTR with 16-byte nonce will reduce the ciphertext expansion from 28 to 16 bytes keeping a comparable level of security.

[7] For instance AERO's nonce is a sequence number, and appended to the plaintext. Moreover the receiver additionally keeps the most recent sequence number value which was rejected, in order to do resynchronization.

[8] Assuming GCM of ν-bit tag. We note that there is a difference in authentication strength due to the numerators of $1/2^\nu$, and GCM can be better e.g. when q' is huge.

```
Algorithm MiniOCB-$\mathcal{E}_{\widetilde{E}}(N, A, M)$

$(\underline{N}, \underline{M}) \xleftarrow{n} (N, M)$
$(M[1], M[2], \ldots, M[m]) \xleftarrow{n} \underline{M}$
$\Sigma \leftarrow M[1] \oplus \ldots \oplus M[m-1] \oplus M[m]10^*$
if $|M[m]| = n$ then $d \leftarrow 0$
else $d \leftarrow 1$
$\underline{L} \leftarrow \widetilde{E}^{(\Sigma, A, m, d)}(\underline{N})$
for $i = 1$ to $m - 1$ do
    $C[i] \leftarrow \widetilde{E}^{(\underline{L}, A, i, 2)}(M[i])$
$\mathrm{pad} \leftarrow \mathtt{msb}_{|M[m]|}(\widetilde{E}^{(\underline{L}, A, m, 2)}(0^n))$
$C[m] \leftarrow M[m] \oplus \mathrm{pad}$
$\underline{C} \leftarrow (C[1], \ldots, C[m])$
$(L, C) \xleftarrow{\nu} (\underline{L}, \underline{C})$
return $(L, C)$
```

```
Algorithm MiniOCB-$\mathcal{D}_{\widetilde{E}}(\widehat{N}, A', L', C')$

$(\underline{L'}, \underline{C'}) \xleftarrow{n} (L', C')$
$(C'[1], \ldots, C'[m']) \xleftarrow{n} \underline{C'}$
if $|C'[m']| = n$ then $d' \leftarrow 0$
else $d' \leftarrow 1$
for $i = 1$ to $m' - 1$ do
    $M'[i] \leftarrow \widetilde{E}^{-1(\underline{L'}, A', i, 2)}(C'[i])$
$\mathrm{pad}' \leftarrow \mathtt{msb}_{|C'[m']|}(\widetilde{E}^{(\underline{L'}, A', m', 2)}(0^n))$
$M'[m'] \leftarrow C'[m'] \oplus \mathrm{pad}'$
$\Sigma' \leftarrow M'[1] \oplus \ldots \oplus M'[m'-1] \oplus M'[m']10^*$
$\underline{N'} \leftarrow \widetilde{E}^{-1(\Sigma', A', m', d')}(\underline{L'})$
if $\mathtt{msb}_\nu(\underline{N'}) \in \rho(\widehat{N})$ then
    $\underline{M'} \leftarrow (M'[1], \ldots, M'[m])$
    $(N', M') \xleftarrow{\nu} (\underline{N'}, \underline{M'})$
    return $M'$
else return $\perp$
```

Fig. 4. Encryption and decryption algorithms of MiniOCB$[\widetilde{E}]$.

4.3 MiniAE from OCB mode

The computation cost of MiniCTR is similar to the generic composition of NAE, and thus there is still a significant difference from the nonce-based unauthenticated encryption. A natural question here is if we can further reduce the computation cost. We positively answer this question by showing a scheme achieving rate-1 operation, i.e. one blockcipher call per one input block. We call our proposal MiniOCB since the design is based on OCB [15,16,17]. MiniOCB is parallelizable for both encryption and decryption. MiniOCB uses n-bit TBC, $\widetilde{E}$, having variable-length tweak in $\mathcal{T} = \{0,1\}^n \times \mathcal{A}_{ae} \times \mathbb{N} \times \{0,1,2\}$ where $\mathbb{N} = \{1, 2, \ldots\}$. The encryption and decryption algorithms of MiniOCB are shown in Figs. 4 and Fig. 5. It needs one TBC call to process one input block, and if TBC is instantiated by a blockcipher it is still rate-1 with respect to the underlying blockcipher (see below).

Design. While MiniOCB is based on OCB, it has an important difference. OCB uses a TBC (which is instantiated by XEX mode [16]) that takes a tweak involving the nonce, whereas MiniOCB can not explicitly use the nonce as a part of a tweak. This is because the nonce can not be present clear in a ciphertext and the decryption should be done so that any small change to a ciphertext will make the decrypted nonce random. Instead we use encrypted nonce $\underline{L}$ to be a part of tweaks for plaintext encryption, and $\underline{L}$ is derived from an encryption of nonce with tweak involving the plaintext checksum, i.e., XOR of plaintext blocks, in the similar manner to OCB. A tweak of MiniOCB also contains $d = 0, 1, 2$ which is used to separate the roles of TBC calls. We remark that $\nu \leq n$ is required, as well as previous schemes.

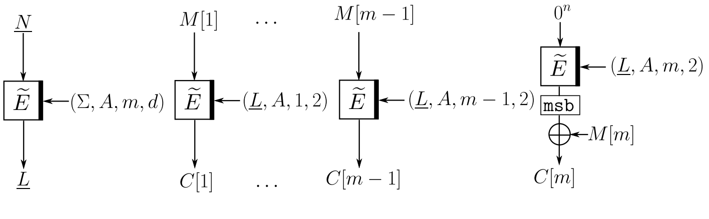

Fig. 5. The encryption algorithm of MiniOCB$[\widetilde{E}]$. Σ denotes the plaintext checksum, and d for encryption of $\underline{N}$ is 0 when $|M[m]| = n$ and 1 otherwise.

We present security bounds of MiniOCB in the following theorems. For simplicity we here provide a security bound for the case of single decryption query.

Theorem 3. *Let $\widetilde{E}$ be a TBC with n-bit block with tweak space $\mathcal{T} = \{0,1\}^n \times \mathcal{A}_{ae} \times \mathbb{N} \times \{0,1,2\}$, and let $\mathsf{MiniOCB}[\widetilde{E}]$ be $\mathsf{MiniOCB}$ using $\widetilde{E}$ with ν-bit nonce and verification range size ω. Then, for any Priv-adversary $\mathcal{A}$ with $q < 2^{n-1}$ encryption queries and σ total input blocks and time complexity t, we have*

$$\mathtt{Adv}^{\mathtt{priv}}_{\mathsf{MiniOCB}[\widetilde{E}]}(\mathcal{A}) \leq \mathtt{Adv}^{\mathtt{tprp}}_{\widetilde{E}}(\mathcal{B}) + \frac{q^2}{2^n},$$

for an adversary $\mathcal{B}$ using σ encryption queries with $t + O(\sigma)$ time.

Theorem 4. *For any Auth-adversary $\mathcal{A}$ with $q < 2^{n-1}$ encryption queries with σ input blocks, and single decryption query with σ' input blocks, and time complexity t, we have*

$$\mathtt{Adv}^{\mathtt{auth}}_{\mathsf{MiniOCB}[\widetilde{E}]}(\mathcal{A}) \leq \mathtt{Adv}^{\mathtt{tsprp}}_{\widetilde{E}}(\mathcal{B}) + \frac{2.5q^2}{2^n} + \frac{2\omega}{2^\nu},$$

for an adversary $\mathcal{B}$ using σ encryption queries and σ' decryption queries with $t + O(\sigma + \sigma')$ time.

The proofs of these theorems are presented in Appendix B.

A blockcipher-based instantiation of $\widetilde{E}$ used in $\mathsf{MiniOCB}$ can use the construction of (2). One may wonder if every tweak update of $\widetilde{E}$ in $\mathsf{MiniOCB}$ requires computation proportional to $|A|$, since A is a part of tweak. However this is not true for most universal hash functions and PRFs, as we can cache the intermediate result depending only on A (e.g. CMAC). Tweak update with respect to third and fourth coordinates can be done without needing additional blockcipher calls (say) by using GF doubling technique [16]. Therefore once we process A, the cost of tweak update is quite small. In Appendix C we provide a brief complexity analysis of our proposals with existing schemes.

5 Misuse resistance

Our security notions do not consider resistance against misuse, while it is an active topic in recent studies. We here consider nonce-misuse and decryption-misuse, as most common concepts thus far. The former was introduced by Rogaway and Shrimpton [25] which refers to the use of repeated nonce in encryption. The latter, a.k.a releasing unverified plaintext (RUP) introduced by Andreeva et. al. [45], refers to the setting that decryption oracle returns unverified plaintext in addition to the result of authentication, which is possible when decryption side has small memory while messages are long. In this section, we provide an analysis of proposed schemes from the viewpoint of misuse. We remark that AERO [9] claims security against these misuses but does not present a formal proof for them. Before we proceed, we point out that there should be a treading-off between strength of security notion and achievable efficiency. We should also think about inherent problem caused by nonce-misuse, i.e. false positive at replay detection even though the corresponding plaintexts are different. In addition, the possibility of decryption-misuse is small when messages are short (except implementation errors) and low-power WSN sometimes consists of unidirectional link from low-end sensors to (a much more powerful) aggregation server.

For our analysis we consider the following three misuse-related security notions and show a security separation between the proposed schemes according to the proposed notions. Though these notions are quite close to what have been discussed for standard nonce-based AE, they need to be explicitly defined due to the difference in the model (i.e. the stateful receiver). In this sense our notions are not new and variants can be considered.

Nonce-misuse. The first is nonce-misuse privacy (NM-Priv) written as $\mathtt{Adv}^{\mathtt{nm\text{-}priv}}_{\mathbf{MiAE}}(\mathcal{A})$. It is essentially the same as $\mathtt{Adv}^{\mathtt{priv}}_{\mathbf{MiAE}}(\mathcal{A})$, however, $\mathcal{A}$ has no restriction on N in queries and is only required to make each query (N, A, M) distinct. It is similar to deterministic privacy (detPriv) defined for DAE privacy [25].

Indistinguishability from large random permutation. The second is indistinguishability from tweakable random permutation, or TSPRP-security described at the end of Section 3.2. This notion is related to the decryption-misuse since the adversary can obtain the unverified plaintext, and also similar to the plaintext awareness 2 (PA2) [45], however a stronger one in the sense that $\mathcal{A}$ can repeat nonce, and unverified *tag* is also given in the decryption query.

Decryption-misuse. The third is the authenticity in the nonce-misuse and decryption-misuse (NMDM-Auth), which is a variant of INT-RUP [45]. Following [45], we first separate the decryption oracle into two oracles, that is, we define three oracles,

$$\begin{aligned}(L, C) \leftarrow & \mathbf{MiAE}\text{-}\mathcal{E}(N, A, M)\\ M' \leftarrow & \mathbf{MiAE}\text{-}\widetilde{\mathcal{D}}(\widehat{N}, A', L', C')\\ \top/\bot \leftarrow & \mathbf{MiAE}\text{-}\mathcal{V}(\widehat{N}, A', L', C'),\end{aligned}$$

where the combination of $\mathbf{MiAE}\text{-}\widetilde{\mathcal{D}}$ and $\mathbf{MiAE}\text{-}\mathcal{V}$ is the ordinary decryption oracle. Here $\top$ denotes the acceptance of ciphertext. We note that $\widetilde{\mathcal{D}}$ and $\mathcal{V}$ are always uniquely determined for any $\mathbf{MiAE}$, i.e., $\mathbf{MiAE}\text{-}\widetilde{\mathcal{D}}$ returns the result of underlying inverse permutation except the first ν bits corresponding decrypted nonce. We define

$$\mathtt{Adv}^{\mathtt{nmdm\text{-}auth}}_{\mathbf{MiAE}}(\mathcal{A}) \overset{\text{def}}{=} \Pr[\mathcal{A}^{\mathbf{MiAE}\text{-}\mathcal{E}, \mathbf{MiAE}\text{-}\widetilde{\mathcal{D}}, \mathbf{MiAE}\text{-}\mathcal{V}} \text{ forges }], \tag{6}$$

where $\mathcal{A}$ forges means that $\mathcal{A}$ receives $\top$ from $\mathbf{MiAE}\text{-}\mathcal{V}$. Here $\mathcal{A}$ is required to satisfy **(Condition 2)** of Authenticity notion for $\mathbf{MiAE}\text{-}\mathcal{V}$, that is, $\mathcal{A}$ never queries $(\widehat{N}, A', L', C')$ to $\mathbf{MiAE}\text{-}\mathcal{V}$ such that $(A', L', C') = (A, L, C)$ and $N \in \rho(\widehat{N})$ if $\mathcal{A}$ already queried (N, A, M) to $\mathbf{MiAE}\text{-}\mathcal{E}$ and received (L, C). Note that $\mathcal{A}$ has no limitation on queries to $\mathbf{MiAE}\text{-}\widetilde{\mathcal{D}}$, and **(Condition 1)** is not required.

Proposition 3. *TSPRP-security implies NM-Priv and NMDM-Auth. More concretely we have*

$$\begin{aligned}\mathtt{Adv}^{\mathtt{nm\text{-}priv}}_{\mathbf{MiAE}}(\mathcal{A}) &\le \mathtt{Adv}^{\mathtt{tprp}}_{\mathbf{MiAE}}(\mathcal{A}) + \frac{q^2}{2^{n+1}}.\\ \mathtt{Adv}^{\mathtt{nmdm\text{-}auth}}_{\mathbf{MiAE}}(\mathcal{A}) &\le \mathtt{Adv}^{\mathtt{tsprp}}_{\mathbf{MiAE}}(\mathcal{A}) + \frac{2(q+q')(\omega+q')}{2^n}.\end{aligned}$$

for any $\mathbf{MiAE}$ *and any adversary* $\mathcal{A}$ *using* q *encryption for the first bound, and in addition* q' *decryption queries to* $\mathbf{MiAE}\text{-}\widetilde{\mathcal{D}}$ *for the second bound.*

The first claim is trivial. For the second claim we have

$$\mathtt{Adv}^{\mathtt{nmdm\text{-}auth}}_{\mathbf{MiAE}}(\mathcal{A}) \le \mathtt{Adv}^{\mathtt{nmdm\text{-}auth}}_{\widetilde{\mathsf{P}}}(\mathcal{A}) + \mathtt{Adv}^{\mathtt{tsprp}}_{\mathbf{MiAE}}(\mathcal{A}')$$

where $\widetilde{\mathsf{P}}$ is tweakable URP over (N, M) with tweak A. The first term of right hand side is bounded by $\frac{2(q+q')(\omega+q')}{2^n}$ with a similar analysis as the proof of Theorem 2. With Proposition 3, we show the following security separation results about our schemes.

Theorem 5. *Assuming components of* MiniETE*, and* MiniCTR *and* MiniOCB *are ideal, we have*

1. MiniETE *is TSPRP-secure, thus it is also secure with respect to NM-Priv and NMDM-Auth.*
2. MiniCTR *and* MiniOCB *are not TSPRP-secure.*
3. MiniCTR *is secure w.r.t. NM-Priv and NMDM-Auth, and* MiniOCB *are not secure w.r.t. both notions.*

Proof. (sketch.) The first claim is trivial from Proposition 3. For the second claim, an adversary first performs encryption query to obtain (N, A, M, L, C) and then decryption query $(\widehat{N}, A', L', C')$ with $A' = A$, $L' = L$, $C' = C \oplus \delta$ for some differential δ. For the third, MiniCTR's NM-Priv proof is just the same as the proof of Theorem 1, from the fact that $\underline{L}$ is pseudorandom whenever (N, A, M) is distinct. For proving NMDM-Auth we extend the proof of Theorem 2 such that Cond1 in the proof can be relaxed to $(N_i, A_i, M_i) \neq (N_j, A_j, M_j)$ without changing the following analysis. Decryption-misuse has already been incorporated in the proof.

Intuitively, these result imply that MiniETE has the highest misuse resistance, and MiniCTR is weaker, but still reasonably-high misuse resistance. The difference is that, while MiniETE makes decrypted plaintext unpredictable even when verification fails, MiniCTR does not guarantee it. In contract MiniOCB has no misuse-resistance, in return for the highest efficiency.

6 Conclusion

In this paper, we have presented a new form of authenticated encryption scheme, called MiniAE, whose ciphertext expansion is the same as the length of single variable integrating nonce and tag, with the help of stateful decryption. While McGrew and Foley's AERO has the same feature, there is no formal treatment on the provable security. Focusing on the most fundamental security properties, i.e., pseudorandomness of ciphertexts under unique nonce, and a basic form of integrity protection including replay detection, we proposed three constructions of MiniAE, called MiniETE, MiniCTR and MiniOCB, where MiniETE is a generalization of AERO. Notably MiniOCB is based on OCB mode of operation and achieves rate-1 parallelizable encryption. This implies that MiniAE can be as efficient as nonce-based unauthenticated encryption.

Acknowledgements

The author would like to thank the anonymous reviewers of ACISP 2016 for useful comments, and Tetsu Iwata for fruitful discussions.

References

1. Minematsu, K.: Authenticated Encryption with Small Stretch (or, How to Accelerate AERO). In: ACISP (2). Volume 9723 of Lecture Notes in Computer Science., Springer (2016) 347–362
2. Dworkin, M.: Recommendation for Block Cipher Modes of Operation: The CCM Mode for Authentication and Confidentiality . NIST Special Publication 800-38C (2004)
3. Dworkin, M.: Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC. NIST Special Publication 800-38D (2007)
4. McGrew, D.: Low power wireless scenarios and techniques for saving bandwidth without sacrificing security. NIST Lightweight Cryptography Workshop 2015 (2015)
5. Struik, R.: Revisiting design criteria for AEAD ciphers targeting highly constrained networks. DIAC: Directions in Authenticated Ciphers (2013) `http://2013.diac.cr.yp.to/`.
6. Seys, S., Preneel, B.: Power consumption evaluation of efficient digital signature schemes for low power devices. In: WiMob (1), IEEE (2005) 79–86
7. Singelée, D., Seys, S., Batina, L., Verbauwhede, I.: The communication and computation cost of wireless security: extended abstract. In: WISEC, ACM (2011) 1–4
8. de Meulenaer, G., Gosset, F., Standaert, F., Pereira, O.: On the Energy Cost of Communication and Cryptography in Wireless Sensor Networks. In: WiMob, IEEE Computer Society (2008) 580–585
9. McGrew, D., Foley, J.: Authenticated Encryption with Replay prOtection (AERO). Internet-Draft (2013)
10. Bellare, M., Rogaway, P.: Encode-Then-Encipher Encryption: How to Exploit Nonces or Redundancy in Plaintexts for Efficient Cryptography. In: ASIACRYPT. Volume 1976 of Lecture Notes in Computer Science., Springer (2000) 317–330
11. McGrew, D.A., Fluhrer, S.R.: The Security of the Extended Codebook (XCB) Mode of Operation. In: Selected Areas in Cryptography. Volume 4876 of Lecture Notes in Computer Science., Springer (2007) 311–327
12. Bellare, M., Kohno, T., Namprempre, C.: Breaking and provably repairing the SSH authenticated encryption scheme: A case study of the Encode-then-Encrypt-and-MAC paradigm. ACM Trans. Inf. Syst. Secur. **7**(2) (2004) 206–241
13. Bellare, M., Namprempre, C.: Authenticated Encryption: Relations among Notions and Analysis of the Generic Composition Paradigm. In: ASIACRYPT. Volume 1976 of Lecture Notes in Computer Science., Springer (2000) 531–545
14. Bellare, M., Namprempre, C.: Authenticated Encryption: Relations among Notions and Analysis of the Generic Composition Paradigm. J. Cryptology **21**(4) (2008) 469–491
15. Rogaway, P., Bellare, M., Black, J.: OCB: A block-cipher mode of operation for efficient authenticated encryption. ACM Trans. Inf. Syst. Secur. **6**(3) (2003) 365–403
16. Rogaway, P.: Efficient Instantiations of Tweakable Blockciphers and Refinements to Modes OCB and PMAC. In: ASIACRYPT. Volume 3329 of Lecture Notes in Computer Science., Springer (2004) 16–31
17. Krovetz, T., Rogaway, P.: The Software Performance of Authenticated-Encryption Modes. In: FSE. Volume 6733 of Lecture Notes in Computer Science., Springer (2011) 306–327
18. Liskov, M., Rivest, R.L., Wagner, D.: Tweakable Block Ciphers. In: CRYPTO. Volume 2442 of Lecture Notes in Computer Science., Springer (2002) 31–46
19. Schroeppel, R.: Hasty Pudding Cipher. AES Submission (1998) `http://www.cs.arizona.edu/rcs/hpc/`.

20. Ferguson, N., Lucks, S., Schneier, B., Whiting, D., Bellare, M., Kohno, T., Callas, J., Walker, J.: Skein Hash Function. SHA-3 Submission (2008) `http://www.skein-hash.info/`.
21. Jean, J., Nikolic, I., Peyrin, T.: Tweaks and Keys for Block Ciphers: The TWEAKEY Framework. In: ASIACRYPT (2). Volume 8874 of Lecture Notes in Computer Science., Springer (2014) 274–288
22. Minematsu, K.: Improved Security Analysis of XEX and LRW Modes. In: Selected Areas in Cryptography. Volume 4356 of Lecture Notes in Computer Science., Springer (2006) 96–113
23. Rogaway, P.: Nonce-Based Symmetric Encryption. In: FSE. Volume 3017 of Lecture Notes in Computer Science., Springer (2004) 348–359
24. Bellare, M., Rogaway, P., Wagner, D.: The EAX Mode of Operation. In: FSE. Volume 3017 of Lecture Notes in Computer Science., Springer (2004) 389–407
25. Rogaway, P., Shrimpton, T.: A Provable-Security Treatment of the Key-Wrap Problem. In: EUROCRYPT. Volume 4004 of Lecture Notes in Computer Science., Springer (2006) 373–390
26. Jr., M.A.S., de Oliveira, B.T., Barreto, P.S.L.M., Margi, C.B., Carvalho, T.C.M.B., Näslund, M.: Comparison of Authenticated-Encryption schemes in Wireless Sensor Networks. In: LCN, IEEE Computer Society (2011) 450–457
27. Yuksel, E., Nielson, H.R., Nielson, F.: ZigBee-2007 Security Essentials. Proceedings of 13th Nordic Workshop on Secure IT-systems (2008)
28. Casado, L., Tsigas, P.: ContikiSec: A Secure Network Layer for Wireless Sensor Networks under the Contiki Operating System. In Jøsang, A., Maseng, T., Knapskog, S.J., eds.: Identity and Privacy in the Internet Age, 14th Nordic Conference on Secure IT Systems, NordSec 2009, Oslo, Norway, 14-16 October 2009. Proceedings. Volume 5838 of Lecture Notes in Computer Science., Springer (2009) 133–147
29. Vitaletti, A., Palombizio, G.: Rijndael for Sensor Networks: Is Speed the Main Issue? Electr. Notes Theor. Comput. Sci. **171**(1) (2007) 71–81
30. Karlof, C., Sastry, N., Wagner, D.: TinySec: a link layer security architecture for wireless sensor networks. In Stankovic, J.A., Arora, A., Govindan, R., eds.: Proceedings of the 2nd International Conference on Embedded Networked Sensor Systems, SenSys 2004, Baltimore, MD, USA, November 3-5, 2004, ACM (2004) 162–175
31. Vading, E., Enander, G.: An evaluation of low power, low-rate wireless data communication technologies for battery powered sensor networks. Master's Thesis, Lund University (2014)
32. Chakraborty, D., Sarkar, P.: HCH: A New Tweakable Enciphering Scheme Using the Hash-Encrypt-Hash Approach. In: INDOCRYPT. Volume 4329 of Lecture Notes in Computer Science., Springer (2006) 287–302
33. Halevi, S., Rogaway, P.: A Tweakable Enciphering Mode. In: CRYPTO. Volume 2729 of Lecture Notes in Computer Science., Springer (2003) 482–499
34. Halevi, S., Rogaway, P.: A Parallelizable Enciphering Mode. In: CT-RSA. Volume 2964 of Lecture Notes in Computer Science., Springer (2004) 292–304
35. Shrimpton, T., Terashima, R.S.: A Modular Framework for Building Variable-Input-Length Tweakable Ciphers. In: ASIACRYPT (1). Volume 8269 of Lecture Notes in Computer Science., Springer (2013) 405–423
36. Hoang, V.T., Krovetz, T., Rogaway, P.: Robust Authenticated-Encryption AEZ and the Problem That It Solves. In: EUROCRYPT (1). Volume 9056 of Lecture Notes in Computer Science., Springer (2015) 15–44
37. Naor, M., Reingold, O.: On the Construction of Pseudorandom Permutations: Luby-Rackoff Revisited. J. Cryptology **12**(1) (1999) 29–66
38. Wang, P., Feng, D., Wu, W.: HCTR: A Variable-Input-Length Enciphering Mode. In: CISC. Volume 3822 of Lecture Notes in Computer Science., Springer (2005) 175–188
39. Halevi, S.: Invertible Universal Hashing and the TET Encryption Mode. In: CRYPTO. Volume 4622 of Lecture Notes in Computer Science., Springer (2007) 412–429
40. Iwata, T., Yasuda, K.: HBS: A Single-Key Mode of Operation for Deterministic Authenticated Encryption. In: FSE. Volume 5665 of Lecture Notes in Computer Science., Springer (2009) 394–415
41. Iwata, T., Yasuda, K.: BTM: A Single-Key, Inverse-Cipher-Free Mode for Deterministic Authenticated Encryption. In: Selected Areas in Cryptography. Volume 5867 of Lecture Notes in Computer Science., Springer (2009) 313–330
42. Desai, A.: New Paradigms for Constructing Symmetric Encryption Schemes Secure against Chosen-Ciphertext Attack. In: CRYPTO. Volume 1880 of Lecture Notes in Computer Science., Springer (2000) 394–412
43. Bellare, M., Desai, A., Jokipii, E., Rogaway, P.: A Concrete Security Treatment of Symmetric Encryption. In: FOCS, IEEE Computer Society (1997) 394–403
44. Niwa, Y., Ohashi, K., Minematsu, K., Iwata, T.: GCM Security Bounds Reconsidered. In: FSE. Volume 9054 of Lecture Notes in Computer Science., Springer (2015) 385–407
45. Andreeva, E., Bogdanov, A., Luykx, A., Mennink, B., Mouha, N., Yasuda, K.: How to Securely Release Unverified Plaintext in Authenticated Encryption. In Sarkar, P., Iwata, T., eds.: Advances in Cryptology - ASIACRYPT 2014 - 20th International Conference on the Theory and Application of Cryptology and Information Security, Kaoshiung, Taiwan, R.O.C., December 7-11, 2014. Proceedings, Part I. Volume 8873 of Lecture Notes in Computer Science., Springer (2014) 105–125

46. Coron, J., Dodis, Y., Mandal, A., Seurin, Y.: A Domain Extender for the Ideal Cipher. In Micciancio, D., ed.: Theory of Cryptography, 7th Theory of Cryptography Conference, TCC 2010, Zurich, Switzerland, February 9-11, 2010. Proceedings. Volume 5978 of Lecture Notes in Computer Science., Springer (2010) 273–289

47. Minematsu, K., Iwata, T.: Tweak-Length Extension for Tweakable Blockciphers. In Groth, J., ed.: Cryptography and Coding - 15th IMA International Conference, IMACC 2015, Oxford, UK, December 15-17, 2015. Proceedings. Volume 9496 of Lecture Notes in Computer Science., Springer (2015) 77–93

A Proofs of Theorem 1 and Theorem 2

We here provide information-theoretic bounds where components are ideal. Deriving computational counterparts is standard.

Authenticity bound. We first consider $\mathsf{MiniCTR1} \overset{\text{def}}{=} \mathsf{MiniCTR}[\widetilde{\mathsf{P}}, \mathsf{R}]$, where $\widetilde{\mathsf{P}}$ is an n-bit block tweakable URP with variable-length tweak, and R is an n-bit input, variable-length output URF. However, it turns out that the authenticity of $\mathsf{MiniCTR1}$ is hold even we let the adversary freely access the oracle computing R. Therefore we consider $\mathsf{MiniCTR1}'$ which omits the computation of R of $\mathsf{MiniCTR1}$. It is a sort of MAC function, and using $\mathsf{MiniCTR1}'$ we think *tagging* oracle which takes a tagging query (t-query) $(\underline{N}, A, \underline{M})$ to return $\underline{L}$, and *verification* oracle which takes a verification query (v-query) $(\widehat{N}, A', \underline{L}', \underline{M}')$ to return $\top$ if $N' \in \rho(\widehat{N})$ holds for the decrypted nonce N', and otherwise $\bot$. For any F compatible to $\mathsf{MiniCTR1}'$ we define $\mathtt{Adv}^{\mathtt{auth}}_{F}(\mathcal{A})$ as the probability of obtaining $\bot$ by accessing tagging and verification oracles, where underlying $\mathcal{A}$ keeps the two conditions of the authenticity notion in Section 3.2 (by internally producing C). As $\underline{M}$ and $\underline{C}$ are one-to-one given $\underline{L}$, this implies that $\mathtt{Adv}^{\mathtt{auth}}_{\mathsf{MiniCTR1}}(\mathcal{A}) \leq \mathtt{Adv}^{\mathtt{auth}}_{\mathsf{MiniCTR1}'}(\mathcal{B})$ for some $\mathcal{B}$ using q t-queries and q' v-queries.

For simplicity we first prove the case for $\nu = n$, hence we have $\underline{N} = N$, $\underline{M} = M$ and $\underline{L} = L$. We abbreviate $(A, \underline{M})$ as V for t-query and $(A', \underline{M}')$ as V' for v-query. We also fix the adversary. By writing (N_i, V_i, L_i) we mean the i-th query is a t-query (N_i, V_i) and the response is L_i, and by writing $(\widehat{N}_j, L'_j, V'_j)$ we mean the j-th query is a v-query $(\widehat{N}_j, L'_j, V'_j)$. The response for $(\widehat{N}_j, L'_j, V'_j)$ is based on N'_j defined as $\widetilde{\mathsf{P}}^{-1,V'_j}(L'_j)$. Here the index sequence is shared for both t- and v-queries, thus it ranges from 1 to $q + q'$, and there may be undefined variables,e.g., if we perform t-query and then v-query, we have (N_1, V_1, L_1) and $(\widehat{N}_2, L'_2, V'_2)$ but (N_2, V_2, L_2) is undefined.

Let $\mathcal{V}$ be the set of possible values for V or V'. For any $v \in \mathcal{V}$, we denote the set of index for t-queries using $V = v$ by $\mathcal{I}[v]$ and the set of index for v-queries using $V' = v$ by $\mathcal{I}'[v]$. From the requirements for the adversary shown in Section 3.2 we only need to consider adversaries with following conditions:

Cond1 $N_i \neq N_j$ for any $i \neq j$

Cond2 if we have (N_i, V_i, L_i) and $(\widehat{N}_j, L'_j, V'_j)$, $V_i = V'_j$ for $i < j$, we have $L'_j \neq L_i$.

Note that we already excluded repeated queries, however a repeat $(L'_i, V'_i) = (L'_j, V'_j)$ is possible, and makes sense if $\widehat{N}_i \neq \widehat{N}_j$. We then define event $\mathcal{E}$ which corresponds to either (1) there is a verification query with response $\bot$, or (2) there is an index pair (i, j), $j < i$ such that $N_i = N'_j$ with $V_i = V'_j$. Here (1) corresponds to a successful forgery and (2) also implies it, by having one more v-query. We denote the event that $\mathcal{E}$ occurred in the first i queries by $\mathcal{E}_i$, and evaluate the conditional probability $\Pr[\mathcal{E}_i | \overline{\mathcal{E}_{i-1}}]$ given the fixed transcript up to $(i-1)$-th query-response pairs (with responses to v-queries fixed to $\bot$) and i-th query, where the probability space is defined by $\widetilde{\mathsf{P}}$.

First we consider the case that i-th query is a v-query $(\widehat{N}_i, L'_i, V'_i)$. To evaluate $\Pr[\mathcal{E}_i | \overline{\mathcal{E}_{i-1}}]$ we need to bound $\max_{x \in \{0,1\}^n} \Pr[N'_i = x | \overline{\mathcal{E}_{i-1}}]$. From the property of $\widetilde{\mathsf{P}}$, we observe that $\Pr[N'_i = x | \overline{\mathcal{E}_{i-1}}]$ is independent of queries using V or V' different from V_i. Let $A_i = \{j \in \mathcal{I}[V_i], j < i\}$ and $B_i = \{k \in \mathcal{I}'[V_i], k < i\}$ be the set of indexes for t- and v-queries using $V, V' = V_i$ done before $(\widehat{N}_i, L'_i, V'_i)$. Here we have $A_i \leq i \leq q$ and $B_i \leq i \leq q'$.

Given $\overline{\mathcal{E}_{i-1}}$, we know that $N'_i \neq N_j$ for any $j \in A_i$, and $N'_i \neq N'_k$ for any $k \in B_i$. Thus we have

$$\Pr[N'_i = x|\overline{\mathcal{E}_{i-1}}]$$

$$\le \sum_{\substack{x_j\in\{0,1\}^n, j\in A_i,\\ x'_k\in\{0,1\}^n : k\in B_i}} \Pr[N'_i = x|\overline{\mathcal{E}_{i-1}}, N_j = x_j, j\in A_i, N'_k = x'_k, k \in B_i]\Pr[N_j = x_j, j\in A_i, N'_k = x'_k, k\in B_i] \tag{7}$$

$$\le \max_{\substack{x_j\in\{0,1\}^n, j\in A_i,\\ x'_k\in\{0,1\}^n : k\in B_i}} \Pr[N'_i = x|\overline{\mathcal{E}_{i-1}}, N_j = x_j, j\in A_i, N'_k = x'_k, k \in B_i] \tag{8}$$

$$\le \frac{1}{2^n - (|A_i| + |B_i|)} \le \frac{1}{2^n - (q+q')} \le \frac{2}{2^n} \tag{9}$$

where the third inequality follows from the property of $\widetilde{\mathsf{P}}$ and the last follows from the assumption. As we are successful in forgery if N'_i is in a set of size ω specified by $\rho(\widehat{N}_i)$, in this case $\Pr[\mathcal{E}_i|\overline{\mathcal{E}_{i-1}}]$ is at most $2\omega/2^n$.

Secondly we consider that case that i-th query is a t-query (N_i, V_i). Then $\Pr[\mathcal{E}_i|\overline{\mathcal{E}_{i-1}}] = \Pr[N_i = N'_j,$ for some $j \in B_i|\overline{\mathcal{E}_{i-1}}]$, which is at most

$$\sum_{k\in B_i} \Pr[N'_k = N_i|\overline{\mathcal{E}_{i-1}}] \le \sum_{k\in B_i} \max_{x\in\{0,1\}^n} \Pr[N'_k = x|\overline{\mathcal{E}_{i-1}}] \le \frac{|B_i|}{2^n - (|A_i| + |B_i|)} \le \frac{2q'}{2^n} \tag{10}$$

from the same analysis as the first case and the assumption on $q + q'$. Therefore, we have

$$\mathtt{Adv}^{\mathtt{auth}}_{\mathsf{MiniCTR1'}}(\mathcal{A}) \le \sum_{i=1,\dots,q+q'} \Pr[\mathcal{E}_i|\overline{\mathcal{E}_{i-1}}] \le \frac{2(q+q')(\omega+q')}{2^n}, \tag{11}$$

which concludes the case $\nu = n$. For $\nu < n$, we substitute N, N', L, and L' in the above analysis with $\underline{N}$ and $\underline{N'}$ and so on (but keeping Cond1 and Cond2), and we obtain the bounds of (9) and (10) as those of $\underline{N'_j}$. In case i-th query is a v-query, the adversary is only required to guess the first ν bits of $\underline{N'_i}$, therefore $\Pr[\mathcal{E}_i|\overline{\mathcal{E}_{i-1}}]$ is at most $2\omega/2^\nu$. For the same reason, in case i-th query is a t-query, the probability is at most $2q'/2^\nu$. Thus $\mathtt{Adv}^{\mathtt{auth}}_{\mathsf{MiniCTR1'}}(\mathcal{A})$ is bounded by $2(q+q')(\omega+q')/2^\nu$, which concludes the proof.

Privacy bound. For proving privacy bound, we consider the indistinguishability of $\mathsf{MiniCTR2} \stackrel{\text{def}}{=} \mathsf{MiniCTR}[\widetilde{\mathsf{P}}, F]$ and $\mathit{\Pi}[F]$, which returns $(\underline{L}, \underline{C})$ where $\underline{L}$ is random and $\underline{C} = F(\underline{L}) \oplus \underline{M}$. Then we have

$$\mathtt{Adv}^{\mathtt{priv}}_{\mathsf{MiniCTR}[\widetilde{E},F]}(\mathcal{A}) \le \mathtt{Adv}^{\mathtt{tprp}}_{\widetilde{E}}(\mathcal{B}) + \mathtt{Adv}^{\mathtt{cpa}}_{\mathsf{MiniCTR2},\mathit{\Pi}[F]}(\mathcal{C}') + \mathtt{Adv}^{\mathtt{cpa}}_{\mathit{\Pi}[F],\$}(\mathcal{C}) \tag{12}$$

$$\le \mathtt{Adv}^{\mathtt{tprp}}_{\widetilde{E}}(\mathcal{B}) + \frac{q^2}{2^{n+1}} + \mathtt{Adv}^{\mathtt{priv\$}}_{\mathit{\Pi}[F]}(\mathcal{C}), \tag{13}$$

for $\mathcal{B}$, $\mathcal{C}$, and $\mathcal{C}'$ using q queries. This follows from the observation that the distinguishing advantage of $\widetilde{\mathsf{P}}$ outputs taking distinct inputs (and arbitrarily tweaks) from random are at most $q^2/2^{n+1}$ and $\mathtt{Adv}^{\mathtt{cpa}}_{\mathit{\Pi}[F],\$}(\mathcal{C})$ is essentially equivalent to $\mathtt{Adv}^{\mathtt{priv\$}}_{\mathit{\Pi}[F]}(\mathcal{C})$. This concludes the proof.

B Proofs of Theorem 3 and Theorem 4

We only consider information-theoretic bounds in the same manner as Appendix A.

Authenticity bound. We first derive authenticity bound. Let $\mathsf{MiniOCB1}$ denote $\mathsf{MiniOCB}[\widetilde{\mathsf{P}}]$. We point out that the optimal strategy for $\mathcal{A}$ is that it first performs q encryption queries, written as (N_i, A_i, M_i) for $i = 1, \dots, q$, and then the decryption query, written as $(\widehat{N}, A', L', C')$. Corresponding answers are written as (L_i, C_i). Assuming $\mathcal{A}$ has the optimal attacking strategy in this case, $(\widehat{N}, A', L', C')$ can be defined as a (deterministic) function of $\mathbf{Z} = ((N_1, A_1, M_1, L_1, C_1), \dots, (N_q, A_q, M_q, L_q, C_q))$. Let d denote the last component of tweak, and we call a tweak with $d \in \{0, 1\}$ an *N-tweak*, and a tweak with $d = 2$ is called an *M-tweak*. Note that an N-tweak and an M-tweak never collide. We write $T_i = (\mathit{\Sigma}_i, A_i, m_i, d_i)$ to denote the N-tweak for encryption of (A_i, N_i, M_i), where Σ_i and m_i and d_i are as defined in Fig. 4, and $T' = (\Sigma', A', m', d')$ to denote the N-tweak for the decryption of $(\widehat{N}, A', L', C')$. We have

$$\mathtt{Adv}^{\mathtt{auth}}_{\mathsf{MiniOCB1}}(\mathcal{A}) \leq \sum_{\mathbf{z}} \Pr_{\mathcal{A},\mathsf{MiniOCB1}}[N' \in \rho(\widehat{N}) | \mathbf{Z} = \mathbf{z}] \cdot \Pr_{\mathcal{A},\mathsf{MiniOCB1}}[\mathbf{Z} = \mathbf{z}] \tag{14}$$

where the sum is taken for all possible values of $\mathbf{Z} = \mathbf{z}$, and N' is the true nonce value obtained in the decryption of (A', L', C'). In what follows we simply write Pr to denote $\Pr_{\mathcal{A},\mathsf{MiniOCB1}}$. Let $\mathtt{Bad}_1$ denote the $\underline{L}$-collision event, i.e. $\underline{L}_i = \underline{L}_j$ for some $1 \leq i < j \leq q$. Note that this is a deterministic event given $\mathbf{Z}$. Then the right hand side of (14) is bounded by

$$\max_{\mathbf{z} \text{ satisfies } \overline{\mathtt{Bad}_1}} \Pr[N' \in \rho(\widehat{N}) | \mathbf{Z} = \mathbf{z}, \overline{\mathtt{Bad}_1}] + \Pr[\mathtt{Bad}_1]. \tag{15}$$

Since $N_i \neq N_j$ for all $i < j$, we have $\underline{N}_i \neq \underline{N}_j$ too. Therefore for any $1 \leq i < j \leq q$, $\underline{L}_i = \underline{L}_j$ can happen only when N-tweaks are different, with probability $1/2^n$. From this

$$\Pr[\mathtt{Bad}_1] \leq \binom{q}{2} \frac{1}{2^n} < \frac{q^2}{2^{n+1}} \tag{16}$$

holds true[9]. We then focus on $\Pr[N' \in \rho(\widehat{N}) | \mathbf{Z} = \mathbf{z}, \overline{\mathtt{Bad}_1}]$. Let $\mathtt{Bad}_2$ be a event that $(T', \underline{L}') = (T_i, \underline{L}_i)$ holds for some $i = 1, \dots, q$. Here a crucial observation is that, as long as $\overline{\mathtt{Bad}_2} \wedge \overline{\mathtt{Bad}_1}$ occurs, guessing $\underline{N}'$ is just a coin toss over a set $\mathcal{S} = \{0,1\}^n \setminus \{\underline{N}_i : i \text{ such that } T_i = T'\}$. Here $\mathcal{S}$ is a random variable depending on Σ', which also depends on the decryption result of $\underline{C}'$ using M-tweaks, however its size is at least $2^n - q$ hence the guess on $\underline{N}'$ succeeds with probability at most $1/(2^n - q)$, which is at most $2/2^n$ by assumption. Therefore, a guess on N' consisting of ω candidates (i.e. $\rho(\widehat{N})$) will succeed with probability at most $2\omega/2^\nu$. From this observation, we have

$$\begin{aligned}
&\Pr[N' \in \rho(\widehat{N}) | \mathbf{Z} = \mathbf{z}, \overline{\mathtt{Bad}_1}] && (17)\\
&\leq \Pr[N' \in \rho(\widehat{N}) | \mathbf{Z} = \mathbf{z}, \overline{\mathtt{Bad}_1} \wedge \overline{\mathtt{Bad}_2}] + \Pr[\mathtt{Bad}_2 | \mathbf{Z} = \mathbf{z}, \overline{\mathtt{Bad}_1}] && (18)\\
&\leq \frac{2\omega}{2^\nu} + \Pr[\mathtt{Bad}_2 | \mathbf{Z} = \mathbf{z}, \overline{\mathtt{Bad}_1}]. && (19)
\end{aligned}$$

Fix $\mathbf{z}$ and $i \leq q$ and let p denote event $[(T', \underline{L}') = (T_i, \underline{L}_i)]$. We perform a case analysis for p. Recall that as conditioned by $\overline{\mathtt{Bad}_1}$ the following analysis assumes $\underline{L}_i \neq \underline{L}_j$ for any $i < j$, and we assumed $(A', L', C') \neq (A_i, L_i, C_i)$ for all $i \leq q$.

Case 1: $(A', m', d', \underline{L}') \neq (A_i, m_i, d_i, \underline{L}_i)$ **for all** $i \leq q$**.**
This implies $\overline{\mathtt{Bad}_2}$, thus $p = 0$.

Case 2: $(A', m', d', \underline{L}') = (A_i, m_i, d_i, \underline{L}_i)$ **for some** $i \leq q$**.**
Let $h \leq q$ be the index such that $(A', m', d', \underline{L}') = (A_h, m_h, d_h, \underline{L}_h)$. As conditioned by $\overline{\mathtt{Bad}_1}$, h is unique. What we need is the probability of $\Sigma' = \Sigma_h$. As $\underline{L}' = \underline{L}_h$ holds, we must have $\underline{C}' \neq \underline{C}_h$. We need a further case analysis.

Case 2-1: $m_h = m' = 1$ **and** $(|\underline{C}'| = 0,\ |\underline{C}_h| \neq 0)$**, or** $(|\underline{C}'| \neq 0,\ |\underline{C}_h| = 0)$**.** Here $(|\underline{C}'| = 0, |\underline{C}_h| \neq 0)$ implies $\Sigma' = 10^{n-1}$, $d' = 1$ and $\Sigma_h = M_h[1]10^*$, with $0 < |M_h[1]|$. If $|M_h[1]| < n$ then $\Sigma' \neq \Sigma_h$ due to the padding. If $|M_h[1]| = n$ then $d_h = 0$, thus a contradiction. Therefore $p = 0$. The case $(|\underline{C}'| = 0$, $|\underline{C}_h| \neq 0)$ is just the opposite.

Case 2-2: $m_h = m' = 1$ **and** $(|\underline{C}'| \neq 0,\ |\underline{C}_h| \neq 0)$**.** If $|\underline{C}_h| \neq |\underline{C}'|$ the paddings in Σ' and Σ_h assure $\Sigma_h \neq \Sigma'$. If $|\underline{C}_h| = |\underline{C}'|$ but $\underline{C}_h \neq \underline{C}'$, then we can write $\Sigma' = (X \oplus \underline{C}')10^a$ and $\Sigma_h = (X \oplus \underline{C}_h)10^a$ for some (X, a) with $a < n - 1$. Thus $\Sigma' \neq \Sigma_h$ holds, and we have $p = 0$.

Case 2-3: $m_h = m' \geq 2$**.** We must have $1 \leq i \leq m'$ such that $C'[i] \neq C_h[i]$. If there exists $i < m'$ such that $C'[i] \neq C_h[i]$, the decryption of $C'[i]$ is done as $M'[i] = \widetilde{\mathsf{P}}^{-1,(\underline{L}',A',m',2)}(C'[i])$ and $M'[i]$ is independent (of any other variables in Σ' and Σ_h) and uniform over $\{0,1\}^n \setminus \{M_h[i]\}$ as we have $M_h[i] = \widetilde{\mathsf{P}}^{-1,(\underline{L}_h,A_h,m_h,2)}(C_h[i])$ and $(\underline{L}_h, A_h, m_h, 2) = (\underline{L}', A', m', 2)$, and this M-tweak is unique to the operations of these two blocks. This implies that p is bounded by the maximum point probability of $\Sigma' \oplus \Sigma_h$ that contains $M'[i]$, uniform over a set of size $2^n - 1$. Thus p is at most $1/(2^n - 1) \leq 2/2^n$.

[9] In fact a collision on $\underline{L}$ results in a simple forgery attack.

Otherwise, we have $C'[i] = C_h[i]$ for any $i = 1, \ldots, m'-1$ and $C'[m'] \neq C_h[m']$, i.e. the difference is only in the last block. Then the differences in Σ' and Σ_h is also in the last blocks to be xored, and we perform the analysis as **Case 2-2** to have $p = 0$.

Summarizing all cases, we have $p \leq 2/2^n$, which means

$$\Pr[\mathtt{Bad}_2|\mathbf{Z} = \mathbf{z}, \overline{\mathtt{Bad}_1}] \leq \frac{2q}{2^n}. \tag{20}$$

Combining (14) to (20) we have

$$\mathtt{Adv}^{\mathtt{auth}}_{\mathsf{MiniOCB1}}(\mathcal{A}) \leq \frac{q^2}{2^{n+1}} + \frac{2q}{2^n} + \frac{2\omega}{2^\nu} \leq \frac{2.5q^2}{2^n} + \frac{2\omega}{2^\nu}, \tag{21}$$

which concludes the proof of authenticity bound.

Privacy bound. To prove the privacy bound, we observe that $\{\underline{C}_i\}_{i=1,\ldots,q}$ is a set of independent and uniformly random sequences given $\overline{\mathtt{Bad}_1}$. As shown by the proof of authenticity bound, the probability of $\mathtt{Bad}_1$ is at most $q^2/2^{n+1}$, and given $\overline{\mathtt{Bad}_1}$, $\{\underline{L}_i\}_{i=1,\ldots,q}$ uniformly distributes over all distinct n-bit sequences, which implies that the distinguishing advantage $\{\underline{L}_i\}_{i=1,\ldots,q}$ between random sequences is at most $q^2/2^{n+1}$. Thus we have

$$\mathtt{Adv}^{\mathtt{priv}}_{\mathsf{MiniOCB1}}(\mathcal{A}) \leq \frac{q^2}{2^{n+1}} + \frac{q^2}{2^{n+1}} \leq \frac{q^2}{2^n}. \tag{22}$$

This concludes the proof of privacy bound.

C Efficiency comparison

We compare the computational complexities of our proposals with existing schemes. For evaluating the complexity we assume the following instantiation strategies.

- For MiniETE, we consider three approaches, namely (1) Hash-Enc-Hash approach using polynomial hash function over GF(2^n), and (2) and (3) using CMC [33] or EME [34]. Both require $2m + O(1)$ blockcipher calls for m-block message. For (2) and (3), they are defined over fixed, n-bit tweaks, however, there are generic tweak-length-extension schemes [46,47]. By using Coron et. al.'s scheme [46], which we call hash tweak (HT), we could extend tweak (fron n bits to an bits) using a multiplications if we use polynomial hash (HT$_{\mathrm{poly}}$) or using CMAC which needs a blockcipher calls (HT$_{\mathrm{cmac}}$).
- For MiniCTR and MiniOCB, we consider $\widetilde{E}$ (which has n-bit block, variable-length tweak) to be instantiated by LRW mode [18] with tweak hashing by polynomial hash (LRW$_{\mathrm{poly}}$) or CMAC (LRW$_{\mathrm{cmac}}$). In addition for MiniCTR, the additive encryption function F is instantiated by CTR mode.

Table 2 shows the comparison of complexities for encryption of m-block plaintext and a-block AD. Here $\#E$ and #Mul denote the required number of n-bit blockcipher invocations and multiplications over GF(2^n), and we ignore constants for the sake of simplicity.

Table 2. Efficiency Comparison of Schemes.

Scheme	$\#E$	#Mul	Ex. Instantiation
$\mathsf{MiniETE}(1)$	m	$2m+a$	Hash-Enc-Hash
$\mathsf{MiniETE}(2)$	$2m+a$	0	CMC or EME w/ $\mathrm{HT}_{\mathrm{cmac}}$
$\mathsf{MiniETE}(3)$	$2m$	a	CMC or EME w/ $\mathrm{HT}_{\mathrm{poly}}$
$\mathsf{MiniCTR}(1)$	m	$m+a$	$\mathrm{LRW}_{\mathrm{poly}}$, CTR encryption
$\mathsf{MiniCTR}(2)$	$2m+a$	0	$\mathrm{LRW}_{\mathrm{cmac}}$, CTR encryption
$\mathsf{MiniOCB}(1)$	m	a	$\mathrm{LRW}_{\mathrm{poly}}$
$\mathsf{MiniOCB}(2)$	$m+a$	0	$\mathrm{LRW}_{\mathrm{cmac}}$
GCM [3]	m	$m+a$	
SIV [25]	$2m+a$	0	
BTM [41]	m	$m+a$	
OCB [15,16,17]	$m+a$	0	