Authenticated Encryption without Tag Expansion (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. This paper studies a form of AE scheme whose ciphertext is only expanded by nonce, with the help of stateful receiver which also enables detection of replays. While there is a scheme having this feature, called AERO, proposed by McGrew and Foley, there is no formal treatment based on the provable security framework.
We propose a provable security framework for such AE schemes, which we call MiniAE, and show several secure schemes using standard symmetric crypto primitives. Most notably, one of our schemes has a similar structure as OCB mode of operation and uses only one blockcipher call to process one input block, thus the computation cost is comparable to the nonce-based encryption-only schemes.

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

1 Introduction

Suppose we have an application using a nonce-based symmetric encryption, e.g. CTR mode of operation, and we now want to switch to authenticated encryption (AE) to make it secure against tampering. This can be caused by, say, a change from wired to wireless communication. With standard AE schemes, e.g., CCM [3] or GCM [16], this inevitably expands the ciphertext caused by the (MAC) tag. However we sometime want to avoid such tag expansion. This is typical for low-power wireless sensor networks (WSNs) since communication cost is expensive and message is relatively short, as pointed out by literature such as [14]. For example, Struik [39] suggests that saving 8 bytes in communication may justify making encryption ten-times more expensive for WSN. Another potential reason is that a change in packet format should be difficult for legacy systems. For example, in industrial plant control, there are many legacy control protocols/systems called SCADA, and they often have no additional room for tag from both computational and communication restrictions (see also [39]). We may shorten the nonce to make a room for the tag, however it changes the nonce generation mechanism of the underlying devices, which can be difficult in practice (say nonce includes the ID of the target devices). Then how can we avoid tag expansion keeping strong security?

In this paper we provide a solution of this problem, by showing a class of AE whose ciphertext is only expanded by the amount of nonce, assuming both sender and receiver maintaining a state. We call it MiniAE. Here, we emphasize that a stateful receiver is necessary for detecting replays, even if we use a nonce-based AE (NAE) which originally requires only sender to be stateful. We are not going to say making receiver to be stateful is always possible. However at least in practice most common networks are equipped with replay protection using a counter, including many Internet protocols and low-power WSNs, such as Zigbee [43] and Bluetooth low energy [1]. A unique feature of our scheme is that a receiver's state is not only used to detect replays but to detect other forgeries. In particular, this idea is already seen in AERO by McGrew and Foley [29], using encode-then-encipher (ETE) approach by Bellare and Rogaway [10] instantiated by XCB [30]. However [29] does not provide a formal security analysis and therefore it is not clear if ETE is essential for achieving AE without tag expansion.

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 [7], 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 in our security notions. The second scheme, which

we call MiniCTR, is similar to a generic composition of AE [8, 9], more specifically a composition of tweakable blockcipher (TBC) introduced by Liskov, Rivest and Wagner [27] and an additive encryption. If we instantiate it using CTR mode of operation and a polynomial hash function, say GHASH [16], the computation cost of MiniCTR is almost the same as GCM [16]. The third scheme tries to further improve the efficiency. It is called MiniOCB and is similar to OCB [34, 32, 26]. As well as OCB it uses a TBC which can be instantiated by a blockcipher. It is parallelizable and requires one blockcipher call to process one plaintext block, hence the efficiency is quite close to the nonce-based encryption schemes. The last two schemes show that ETE is not the exclusive solution to our problem and the computation cost can in fact be reduced to those of ordinal nonce-based encryptions.

We remark that our basic security notions considered here are quite similar to those for NAE. According to [29] AERO is expected to have a certain *misuse-resistance* beyond NAE. Our basic notions described above do not consider misuse-resistance. However, in Section 5 we provide 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 $|X|_n \stackrel{\text{def}}{=} \lceil |X|/n \rceil$. 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 $(X[1], \ldots, X[x]) \xleftarrow{s} X$, where $|X[i]| = s$ for $i < x$ and $|X[x]| \leq s$. For $X = \varepsilon$, we let $X[1] \xleftarrow{s} X$ with $X[1] = \varepsilon$. Moreover, for X and Y such that $|X| \leq n$ and $|X| + |Y| \geq n$ we write $(\underline{X}, \underline{Y}) \xleftarrow{n} (X, Y)$ 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 $(X, Y) \xleftarrow{m} (\underline{X}, \underline{Y})$, 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 $0 \leq |X| \leq n$ 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 $X \xleftarrow{\$} \mathcal{X}$ to mean the uniform sampling of X over $\mathcal{X}$.

For keyed function $F : \mathcal{K} \times \mathcal{X} \to \mathcal{Y}$ with key $K \in \mathcal{K}$, we may simply write $F_K : \mathcal{X} \to \mathcal{Y}$ if key space is obvious, or even write as $F : \mathcal{X} \to \mathcal{Y}$ if being keyed is obvious. If $E_K : \mathcal{X} \to \mathcal{X}$ 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 tweakable blockcipher (TBC), $\widetilde{E}_K : \mathcal{T} \times \mathcal{X} \to \mathcal{X}$, 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 $M = \widetilde{E}_K^{-1,T}(C)$.

Random Functions. Let $\mathrm{Func}(n, m)$ be the set of all functions $\{0,1\}^n \to \{0,1\}^m$, 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 $\mathsf{R} \xleftarrow{\$} \mathrm{Func}(n, m)$. An n-bit uniform random permutation (URP), denoted by P, is similarly defined as $\mathsf{P} \xleftarrow{\$} \mathrm{Perm}(n)$. We also define tweakable URP. Let $\mathcal{T}$ be a set of tweak and $\mathrm{Perm}^{\mathcal{T}}(n)$ be a set of functions $\mathcal{T} \times \{0,1\}^n \to \{0,1\}^n$ 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 $\widetilde{\mathsf{P}} \xleftarrow{\$} \mathrm{Perm}^{\mathcal{T}}(n)$.

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 $F_K : \{0,1\}^n \to \{0,1\}^m$ and $G_{K'} : \{0,1\}^n \to \{0,1\}^m$ 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

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

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 $\mathtt{Adv}^{\mathtt{cca}}$. Using URF R compatible to F, we define

$$\mathtt{Adv}^{\mathtt{prf}}_{F_K}(\mathcal{A}) \stackrel{\text{def}}{=} \mathtt{Adv}^{\mathtt{cpa}}_{F_K, \mathsf{R}}(\mathcal{A}).$$

In a similar manner, let URP P compatible to blockcipher E_K and tweakable URP $\widetilde{\mathsf{P}}$ compatible to TBC $\widetilde{E}_K$. Then we define

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

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 $\ell(x)$-almost XOR universal (AXU) if

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

holds for any distinct $X, X' \in \mathcal{X}$ with $\max\{|X|_n, |X'|_n\} = x$. 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 such as [37, 5, 23], 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 [27], using an n-bit blockcipher E_K and an independently-keyed $\ell(x)$-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 $O(\ell_{\max} q^2/2^n)$ plus a CCA-advantage of E_K, for any adversary with q CCA queries using tweak of maximum block length $\ell_{\max}$. Note that $\mathcal{T}$ can be a set of variable strings, and $H_{K'}$ can be relaxed to be a computational universal hash, where (1) holds for some H_{ideal} which is hard to be distinguished from $H_{K'}$ via CPA queries. For instance $H_{K'}$ can be implemented by a polynomial hash function defined over GF(2^n), or a VIL-PRF based on an n-bit blockcipher, say CMAC [4]. The use of two keys can also be reduced by using (e.g.) the result of Rogaway [32].

3 Definition of MiniAE

3.1 Basic Interface

We first define a general I/O of MiniAE. 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 $|C| = |M|$. Here $\mathcal{A}_{ae} = \mathcal{M}_{ae} = \{0,1\}^*$. A message sent over a communication channel is (A, L, C). Thus the expansion caused by encryption is ν bits. AD A and plaintext M can be empty, and if M is empty the corresponding C is also empty. In the standard setting nonce is unique 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 [29][1]. 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 $|\rho(N)| \leq \omega$ 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 $j \leq i$ 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 Length Incompatibility. As all of our constructions are defined over n-bit blocks for some n (say 128), we require $\nu \leq n$, and $|N| + |M| \geq n$ holds, which means if $\nu < n$ we have a nonzero limit on the minimum plaintext length, or, in practice we may pad as [29]. 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

We introduce two security notions, privacy and authenticity, to model the security of MiniAE. Roughly saying, our privacy notion requires that the encryption outputs are pseudorandom as long as nonces are unique. Our authenticity notion is the probability of successful 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[2]. Let **MiAE** be an MiniAE with ν-bit nonce (with some key $K \xleftarrow{\$} \mathcal{K}$, which we omit the description). 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 $\perp$.

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[3] by μ, in the same manner to [33]. The privacy notion for Priv-adversary $\mathcal{A}$ is defined as

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

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 who accesses **MiAE**-$\mathcal{E}$ and **MiAE**-$\mathcal{D}$. We write q encryption queries as $(N_1, A_1, M_1), \ldots, (N_q, A_q, M_q)$. In the same fashion, we write q' decryption queries as $(\widehat{N}_1, A'_1, L'_1, C'_1), \ldots, (\widehat{N}_{q'}, A'_{q'}, L'_{q'}, C'_{q'})$. We also let $(L_1, C_1), \ldots, (L_q, C_q)$ be the corresponding oracle answers for encryption queries. We assume $\mathcal{A}$ against authenticity follows the two conditions. The first **(condition 1)** is nonce-respecting for encryption queries (i.e. $N_i \neq N_j$ for any $i \neq j$), and the second **(condition 2)** is that, 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 (A, N, 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 the description of Section 3.1. We also

[1] Decryption of [29] also maintains the most recent invalid nonce, in order to do resynchronization.

[2] In this sense our notions are similar to Rogaway's nonce-based encryption [33] as it allows a provable security analysis without taking into account the details of nonce generation.

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

excluded the case $N \notin \rho(\widehat{N})$, as it will be always rejected (thus trivial loss). The authenticity notion for the Auth-adversary $\mathcal{A}$ is defined as

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

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

3.3 Remarks

Comparison with NAE security notions. Our security notions are quite similar to standard notions for NAE shown by e.g. [11]. 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 (thus unverified decryption). In other words, a stateful receiver is crucial to MiniAE to detect any type of forgery.

Comparison with alternative solutions. If the receiver is completely synchronized with the sender, there is a trivial solution to save bandwidth: use NAE and omit nonce from the communication. However this is problematic when packets may lost. A mitigation is to send a partial information (say the last few bits) of nonce, as shown by [28]. This basically works, however it makes the messaging format dependent on the number of tolerable packet lost, which depends on the network condition and application. In some cases adequate control would be difficult. Moreover, once the receiver loses the state in these nonce omission schemes, even the (unverified) decryption becomes impossible. In contrast, MiniAE allows the parties to design the mechanism to handle packet lost without changing the message format, and even with the complete lost of receiver state, it allows (unverified) decryption, and allows efficient built-in resynchronization as shown by AERO.

Another potential solution to our problem is Deterministic AE (DAE) proposed by Rogaway and Shrimpton [35]. 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. Since DAE encryption is deterministic thus the same plaintext is always mapped to the same ciphertext, making the standard privacy notion impossible to achieve. DAE can prevent replay if the receiver keeps (hash values of) all received ciphertexts, or using Bloom filter allowing some false negatives. Either option requires much larger memories or computations than comparison of state variables.

Table 1 summarizes encryption schemes in the presence of stateful receiver for replay protection. Here the rightmost column indicates the ability to perform decryption under packet loss and unverified decryption under receiver state loss.

Table 1. Comparison of Encryption Schemes.

Scheme	Expansion	Privacy	Authenticity	Replay Protect	Dec under packet/state loss
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 [24] for a good survey). A short tag is clearly problematic, and to avoid overflow of short counter, we need another mechanism in addition to counter-based replay detection, say, frequent session key changes. We here present some examples.

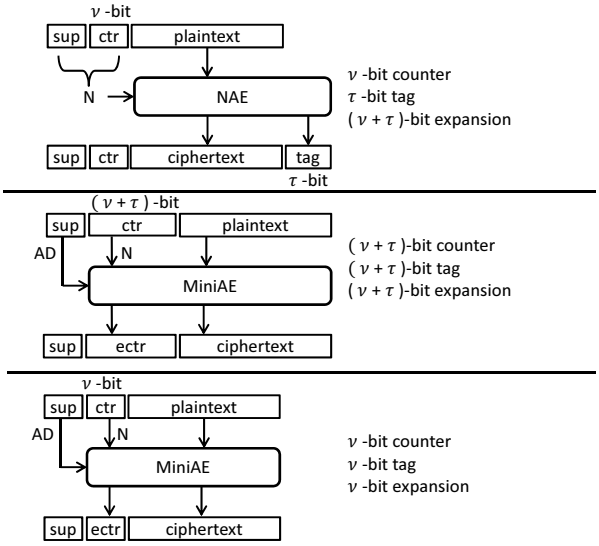

Fig. 1. (Top) Baseline, a nonce-based AE in WSN whose counter and tag may be short. (Middle) Scenario 1, MiniAE with a longer counter enables larger counter space and stronger authenticity while keeping the same overhead. (Bottom) Scenario 2, MiniAE has the same counter as Baseline, but reduces the overhead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

- Zigbee [2] for IEEE 802.15.4 uses AES-CCM. In a typical setting, a nonce (or IV) 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 [43].
- Bluetooth low energy (BLE) [1] 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 [12, 41, 25, 40] 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, the top of Figure 1), excluding AD. Then there are two application scenarios of MiniAE. First, if we consider both ν-bit counter and τ-bit tag are too short, then a 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 we consider ν-bit counter is sufficiently long and $\tau \leq \nu$, then a MiniAE with ν-bit nonce will remove the expansion caused by tag, without harming the authenticity strength (Scenario 2, bottom of Fig. 1). For example, keeping Zigbee's communication overhead, MiniAE of scenario 1 realizes 64-bit counter and 64-bit authenticity. In case of 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 [12, 41, 25, 40] MiniAE of scenario 1 allows 48-bit counter and 48-bit authenticity. We finally remark that [29] 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 also assume $\nu \leq n$ for some fixed block length n, and plaintext M used in a scheme satisfies $|M| \geq \nu - n$.

4.1 MiniAE from large blockcipher

To build a MiniAE scheme, a naive solution is to use a TBC with arbitrarily large-block, a.k.a Tweakable Enciphering Scheme (TES) in 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 TES, where $\mathcal{M} = \bigcup_{i \geq n} \{0,1\}^i$. For nonce N, tweak A and plaintext M, 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, proving Proposition 1 is trivial and the proof of Proposition 2 is easily obtained as a variant of that of Theorem 2 thus we omit them here.

Proposition 1. *Let* $\mathsf{MiniETE}[\widetilde{\mathbf{E}}]$ *be* 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*

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

where $\mathcal{B}$ *is an adversary using* q *encryption queries with 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*

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

where $\mathcal{B}$ *is an adversary using* q *encryption and* q' *decryption queries with time complexity* $t' = t + O(\sigma)$*.*

For instantiations of $\widetilde{\mathbf{E}}$, we could use any of existing schemes such as [13, 18, 19, 38]. As mentioned, this scheme is in fact a generalization of AERO which uses XCB [30] with AES-128 as an internal TES, which is also a variant of ETE-based AE [10]. That is, AERO is provably secure in our security model, although there are minor differences and additional features [4]. MiniETE also has some similarities with ETE-based AE schemes, such as AEZ [20] and one described at [38].

4.2 MiniAE from encrypted counter

The scheme of Section 4.1 is conceptually simple, however actual computation cost is rather high. A popular approach to $\widetilde{\mathbf{E}}$ shown by the seminal Naor and Reingold paper [31] uses two universal hashing layers with one encryption. Many concrete schemes based on [31] can be found [13, 42, 17] including AERO's XCB. With n-bit polynomial hash and n-bit blockcipher, these constructions require two $\mathrm{GF}(2^n)$ multiplications and one blockcipher call for each n-bit input block. EME and CMC modes [18, 19] 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 called MiniCTR. The name comes from that it consists of encryption of nonce and an additive encryption using encrypted nonce as an IV, where the latter can be instantiated by CTR mode. Specifically, it uses an n-bit block TBC $\widetilde{E}_K : \mathcal{T} \times \{0,1\}^n \to \{0,1\}^n$ and a PRF of n-bit input and variable-length output, $F_{K'}$. Here we assume $\mathcal{T}$ is sufficiently large to encode a pair $(A, \underline{M})$. The algorithms of MiniCTR are shown in Fig. 2 and the encryption is also shown in Fig. 3. If $\widetilde{E}$ is instantiated by n-bit blockcipher and n-bit polynomial hash with (2) and F is instantiated by CTR mode, computation cost of MiniCTR for each n-bit plaintext block is one $\mathrm{GF}(2^n)$ multiplication and one blockcipher call, which is roughly the same as GCM. The presented scheme has a similar structure as DAE schemes (SIV [35], HBS [22], BTM [21]) or randomized encryption by Desai [15]. 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'. For decryption inverse permutation is applied and we see if $N' = 0^n$. If it is used as an MiniAE scheme with nonce $\underline{N}$ instead of a fixed value, an adversary can easily break the authenticity.

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

[4] 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.

Algorithm MiniCTR-$\mathcal{E}_{\widetilde{E},F}(N,A,M)$	**Algorithm** MiniCTR-$\mathcal{D}_{\widetilde{E},F}(\widehat{N},A',L',C')$
1. $(\underline{N},\underline{M}) \xleftarrow{n} (N,M)$ 2. $\underline{L} \leftarrow \widetilde{E}^{(A,\underline{M})}(\underline{N})$ 3. $\underline{C} \leftarrow F(\underline{L}) \oplus \underline{M}$ 4. $(L,C) \xleftarrow{\nu} (\underline{L},\underline{C})$ 5. **return** (L,C)	1. $(\underline{L}',\underline{C}') \xleftarrow{n} (L',C')$ 2. $\underline{M}' \leftarrow F(\underline{L}') \oplus \underline{C}'$ 3. $\underline{N}' \leftarrow \widetilde{E}^{-1(A',\underline{M}')}(\underline{L}')$ 4. **if** $\mathtt{msb}_\nu(\underline{N}') \in \rho(\widehat{N})$ **then** 5. $\quad (N',M') \xleftarrow{\nu} (\underline{N}',\underline{M}')$ 6. $\quad$ **return** M' 7. **else return** $\perp$

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

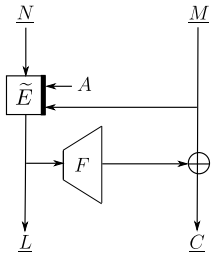

Fig. 3. The encryption algorithm of MiniCTR$[\widetilde{E},F]$, except the pre- and post-parsings.

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

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

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

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

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

The proofs of Theorems 1 and 2 are presented in Appendix A.

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, is possible. We call our proposal MiniOCB. As the name suggests the design of MiniOCB is based on OCB mode of operation [34, 32, 26] and uses a TBC as a core component. MiniOCB is parallelizable for both encryption and decryption and uses one TBC call to process one plaintext block. In MiniOCB, the underlying n-bit TBC is denoted by $\widetilde{E}$ and the tweak is an element of $\mathcal{T} = \{0,1\}^n \times \mathcal{A}_{ae} \times \mathbb{N} \times \{0,1,2\}$ where $\mathbb{N} = \{1,2,\dots\}$. The encryption and decryption algorithms of MiniOCB are shown in Fig. 4, and encryption is also shown in Fig. 5.

Design. While MiniOCB is based on OCB, it has an important difference. OCB uses a TBC (which is instantiated by XEX mode [32]) that takes a tweak involving the nonce, whereas MiniOCB can not

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

1. $(\underline{N}, \underline{M}) \xleftarrow{n} (N, M)$
2. $(M[1], M[2], \ldots, M[m]) \xleftarrow{n} \underline{M}$
3. $\Sigma \leftarrow M[1] \oplus \ldots \oplus M[m-1] \oplus M[m]10^*$
4. **if** $|M[m]| = n$ **then** $d \leftarrow 0$
5. **else** $d \leftarrow 1$
6. $\underline{L} \leftarrow \widetilde{E}^{(\Sigma, A, m, d)}(\underline{N})$
7. **for** $i = 1$ **to** $m - 1$ **do**
8. $\quad C[i] \leftarrow \widetilde{E}^{(\underline{L}, A, i, 2)}(M[i])$
9. $\text{pad} \leftarrow \mathtt{msb}_{|M[m]|}(\widetilde{E}^{(\underline{L}, A, m, 2)}(0^n))$
10. $C[m] \leftarrow M[m] \oplus \text{pad}$
11. $\underline{C} \leftarrow (C[1], \ldots, C[m])$
12. $(L, C) \xleftarrow{\nu} (\underline{L}, \underline{C})$
13. **return** (L, C)

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

1. $(\underline{L}', \underline{C}') \xleftarrow{n} (L', C')$
2. $(C'[1], \ldots, C'[m']) \xleftarrow{n} \underline{C}'$
3. **if** $|C'[m']| = n$ **then** $d' \leftarrow 0$
4. **else** $d' \leftarrow 1$
5. **for** $i = 1$ **to** $m' - 1$ **do**
6. $\quad M'[i] \leftarrow \widetilde{E}^{-1(\underline{L}', A', i, 2)}(C'[i])$
7. $\text{pad}' \leftarrow \mathtt{msb}_{|C'[m']|}(\widetilde{E}^{(\underline{L}', A', m', 2)}(0^n))$
8. $M'[m'] \leftarrow C'[m'] \oplus \text{pad}'$
9. $\Sigma' \leftarrow M'[1] \oplus \ldots \oplus M'[m'-1] \oplus M'[m']10^*$
10. $\underline{N}' \leftarrow \widetilde{E}^{-1(\Sigma', A', m', d')}(\underline{L}')$
11. **if** $\mathtt{msb}_\nu(\underline{N}') \in \rho(\widehat{N})$ **then**
12. $\quad \underline{M}' \leftarrow (M'[1], \ldots, M'[m])$
13. $\quad (N', M') \xleftarrow{\nu} (\underline{N}', \underline{M}')$
14. $\quad$ **return** M'
15. **else return** $\perp$

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

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 call, where $d = 0$ or 1 is used for nonce encryption depending on the last plaintext block length, and $d = 2$ is used for plaintext encryption. We remark that $\nu \leq n$ is required, as well as previous schemes.

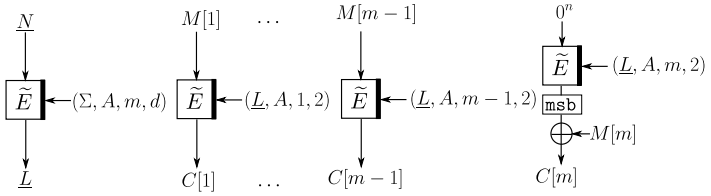

Fig. 5. The encryption algorithm of MiniOCB$[\widetilde{E}]$. Σ denotes the plaintext checksum, and d for encryption of $\underline{N}$ is 0 when $|M[m]| = n$ and 1 otherwise.

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* MiniOCB$[\widetilde{E}]$ *be* 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*

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

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 MiniOCB can use the construction of (2). One may wonder if every tweak update of $\widetilde{E}$ in MiniOCB requires computation proportional to $|A|$, since A is included in tweaks. However this is not true for most universal hash functions and PRFs, as we can cache the intermediate result depending only on A (consider CMAC, for example). We can also incorporate GF doubling technique for update of the third and fourth tweak elements, in a similar manner to OCB. Therefore once we process A, the cost of tweak update is quite small.

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 [35] which refers to the use of repeated nonce in encryption. The latter, a.k.a releasing unverified plaintext (RUP) introduced by Andreeva et. al. [6], refers to the setting that decryption oracle returns unverified plaintext in addition to the result of authentication, which is probable when decryption side has small memory while messages are long. In this section, we provide a short analysis of proposed schemes as an first step toward understanding misuse-resistance of MiniAE. We remark that AERO [29] 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. We remark that the selection is only to show a security separation between the proposed schemes. Our notions are not new and variants can be considered. 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 [35]. The second is indistinguishability from tweakable random permutation, or TSPRP-security. Note that, for any MiniAE, its encryption is a keyed permutation of (N, M) tweaked by A for encryption query (A, N, M). Thus we can always define the corresponding keyed inverse and thus we can naturally define $\mathtt{Adv}^{\mathtt{tsprp}}_{\mathbf{MiAE}}(\mathcal{A})$. This notion is related to the decryption-misuse since the adversary always get the unverified plaintext. It is also similar to the plaintext awareness 2 (PA2) [6], with more abilities to the adversary (i.e. $\mathcal{A}$ can repeat nonce, and unverified tag is also given in the decryption query). The third is the authenticity in the nonce-misuse and decryption-misuse (NMDM-Auth). It is a variant of INT-RUP [6] and defined as

$$\mathtt{Adv}^{\mathtt{nmdm\text{-}auth}}_{\mathbf{MiAE}}(\mathcal{A}) \stackrel{\mathrm{def}}{=} \Pr[\mathcal{A}^{\mathbf{MiAE}\text{-}\mathcal{E},\mathbf{MiAE}\text{-}\mathcal{D}'} \text{ forges }], \tag{5}$$

where **MiAE**-$\mathcal{D}'$ takes $(\widehat{N}, A', L', C')$ and returns unverified plaintext M' in addition to $\perp$ if authenticity check fails. Here $\mathcal{A}$ is required to satisfy **(condition 2)** of Authenticity notion, but **(condition 1)** is not required.

We naturally see that TSPRP-security implies NM-Priv and NMDM-Auth. Proofs are simple, and we omit them for limited space. We show the following security separation results about our schemes.

Theorem 5. *(informal)*

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 is trivial. For the second 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 security, and simpler MiniCTR is comparably secure. The difference is that, while MiniETE makes decrypted plaintext unpredictable even when verification fails, MiniCTR does not guarantee it. MiniOCB has no misuse-resistance, in return for the speed advantage.

6 Conclusion

In this paper, we have presented a new type of authenticated encryption scheme, called MiniAE, whose expansion is the same as the amount of nonce, with the help of stateful decryption. While McGrew and Foley's AERO is one of such schemes, 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 efficient and secure 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 operation. This implies that MiniAE can be, at least in theory, as efficient as nonce-based unauthenticated encryption.

It would be an interesting direction to extend our basic security notions to incorporate misuse. In Section 5 we provide a short analysis of proposed scheme from the viewpoint of misuse. It shows that, as claimed by AERO, MiniETE achieves a highest misuse-resistance, although a more efficient MiniCTR also has a comparable level of security.

Acknowledgements

The author would like to thank Tetsu Iwata for discussions and insights.

References

1. : Bluetooth low energy `http://www.bluetooth.com/Pages/Low-Energy.aspx/`.
2. : ZigBee Alliance `http://www.zigbee.org/`.
3. : Recommendation for Block Cipher Modes of Operation: The CCM Mode for Authentication and Confidentiality . NIST Special Publication 800-38C (2004) National Institute of Standards and Technology.
4. : Recommendation for Block Cipher Modes of Operation: The CMAC Mode for Authentication. NIST Special Publication 800-38B (2005) National Institute of Standards and Technology.
5. : Skein Hash Function. SHA-3 Submission (2008) `http://www.skein-hash.info/`.
6. 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
7. 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
8. Bellare, M., Namprempre, C.: Authenticated Encryption: Relations among Notions and Analysis of the Generic Composition Paradigm. In Okamoto, T., ed.: Advances in Cryptology - ASIACRYPT 2000, 6th International Conference on the Theory and Application of Cryptology and Information Security, Kyoto, Japan, December 3-7, 2000, Proceedings. Volume 1976 of Lecture Notes in Computer Science., Springer (2000) 531–545

9. Bellare, M., Namprempre, C.: Authenticated Encryption: Relations among Notions and Analysis of the Generic Composition Paradigm. J. Cryptology **21**(4) (2008) 469–491
10. Bellare, M., Rogaway, P.: Encode-Then-Encipher Encryption: How to Exploit Nonces or Redundancy in Plaintexts for Efficient Cryptography. In Okamoto, T., ed.: Advances in Cryptology - ASIACRYPT 2000, 6th International Conference on the Theory and Application of Cryptology and Information Security, Kyoto, Japan, December 3-7, 2000, Proceedings. Volume 1976 of Lecture Notes in Computer Science., Springer (2000) 317–330
11. Bellare, M., Rogaway, P., Wagner, D.: The EAX Mode of Operation. [36] 389–407
12. 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
13. Chakraborty, D., Sarkar, P.: HCH: A New Tweakable Enciphering Scheme Using the Hash-Encrypt-Hash Approach. In Barua, R., Lange, T., eds.: Progress in Cryptology - INDOCRYPT 2006, 7th International Conference on Cryptology in India, Kolkata, India, December 11-13, 2006, Proceedings. Volume 4329 of Lecture Notes in Computer Science., Springer (2006) 287–302
14. de Meulenaer, G., Gosset, F., Standaert, F., Pereira, O.: On the Energy Cost of Communication and Cryptography in Wireless Sensor Networks. In: IEEE International Conference on Wireless and Mobile Computing, Networking and Communications, WiMob 2008, Avignon, France, 12-14 October 2008, Proceedings, IEEE (2008) 580–585
15. Desai, A.: New Paradigms for Constructing Symmetric Encryption Schemes Secure against Chosen-Ciphertext Attack. In Bellare, M., ed.: Advances in Cryptology - CRYPTO 2000, 20th Annual International Cryptology Conference, Santa Barbara, California, USA, August 20-24, 2000, Proceedings. Volume 1880 of Lecture Notes in Computer Science., Springer (2000) 394–412
16. Dworkin, M.: Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC. NIST Special Publication 800-38D (2007)
17. Halevi, S.: Invertible Universal Hashing and the TET Encryption Mode. In Menezes, A., ed.: Advances in Cryptology - CRYPTO 2007, 27th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 19-23, 2007, Proceedings. Volume 4622 of Lecture Notes in Computer Science., Springer (2007) 412–429
18. Halevi, S., Rogaway, P.: A Tweakable Enciphering Mode. In Boneh, D., ed.: Advances in Cryptology - CRYPTO 2003, 23rd Annual International Cryptology Conference, Santa Barbara, California, USA, August 17-21, 2003, Proceedings. Volume 2729 of Lecture Notes in Computer Science., Springer (2003) 482–499
19. Halevi, S., Rogaway, P.: A Parallelizable Enciphering Mode. In Okamoto, T., ed.: Topics in Cryptology - CT-RSA 2004, The Cryptographers' Track at the RSA Conference 2004, San Francisco, CA, USA, February 23-27, 2004, Proceedings. Volume 2964 of Lecture Notes in Computer Science., Springer (2004) 292–304
20. Hoang, V.T., Krovetz, T., Rogaway, P.: Robust Authenticated-Encryption AEZ and the Problem That It Solves. In Oswald, E., Fischlin, M., eds.: Advances in Cryptology - EUROCRYPT 2015 - 34th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Sofia, Bulgaria, April 26-30, 2015, Proceedings, Part I. Volume 9056 of Lecture Notes in Computer Science., Springer (2015) 15–44
21. Iwata, T., Yasuda, K.: BTM: A single-key, inverse-cipher-free mode for deterministic authenticated encryption. In Jr., M.J.J., Rijmen, V., Safavi-Naini, R., eds.: Selected Areas in Cryptography, 16th Annual International Workshop, SAC 2009, Calgary, Alberta, Canada, August 13-14, 2009, Revised Selected Papers. Volume 5867 of Lecture Notes in Computer Science., Springer (2009) 313–330
22. Iwata, T., Yasuda, K.: HBS: A single-key mode of operation for deterministic authenticated encryption. In Dunkelman, O., ed.: Fast Software Encryption, 16th International Workshop, FSE 2009, Leuven, Belgium, February 22-25, 2009, Revised Selected Papers. Volume 5665 of Lecture Notes in Computer Science., Springer (2009) 394–415
23. Jean, J., Nikolic, I., Peyrin, T.: Tweaks and keys for block ciphers: The TWEAKEY framework. 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 II. Volume 8874 of Lecture Notes in Computer Science., Springer (2014) 274–288
24. 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 Chou, C.T., Pfeifer, T., Jayasumana, A.P., eds.: IEEE 36th Conference on Local Computer Networks, LCN 2011, Bonn, Germany, October 4-7, 2011, IEEE (2011) 450–457
25. 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
26. Krovetz, T., Rogaway, P.: The Software Performance of Authenticated-Encryption Modes. In Joux, A., ed.: FSE. Volume 6733 of Lecture Notes in Computer Science., Springer (2011) 306–327

27. Liskov, M., Rivest, R.L., Wagner, D.: Tweakable Block Ciphers. In Yung, M., ed.: Advances in Cryptology - CRYPTO 2002, 22nd Annual International Cryptology Conference, Santa Barbara, California, USA, August 18-22, 2002, Proceedings. Volume 2442 of Lecture Notes in Computer Science., Springer (2002) 31–46
28. Luk, M., Mezzour, G., Perrig, A., Gligor, V.D.: MiniSec: a secure sensor network communication architecture. In Abdelzaher, T.F., Guibas, L.J., Welsh, M., eds.: Proceedings of the 6th International Conference on Information Processing in Sensor Networks, IPSN 2007, Cambridge, Massachusetts, USA, April 25-27, 2007, ACM (2007) 479–488
29. McGrew, D., Foley, J.: Authenticated Encryption with Replay prOtection (AERO). Internet-Draft (2013)
30. McGrew, D.A., Fluhrer, S.R.: The Security of the Extended Codebook (XCB) Mode of Operation. In Adams, C.M., Miri, A., Wiener, M.J., eds.: Selected Areas in Cryptography, 14th International Workshop, SAC 2007, Ottawa, Canada, August 16-17, 2007, Revised Selected Papers. Volume 4876 of Lecture Notes in Computer Science., Springer (2007) 311–327
31. Naor, M., Reingold, O.: On the Construction of Pseudorandom Permutations: Luby-Rackoff Revisited. J. Cryptology **12**(1) (1999) 29–66
32. Rogaway, P.: Efficient Instantiations of Tweakable Blockciphers and Refinements to Modes OCB and PMAC. In Lee, P.J., ed.: ASIACRYPT. Volume 3329 of Lecture Notes in Computer Science., Springer (2004) 16–31
33. Rogaway, P.: Nonce-Based Symmetric Encryption. [36] 348–359
34. 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
35. Rogaway, P., Shrimpton, T.: A Provable-Security Treatment of the Key-Wrap Problem. In Vaudenay, S., ed.: Advances in Cryptology - EUROCRYPT 2006, 25th Annual International Conference on the Theory and Applications of Cryptographic Techniques, St. Petersburg, Russia, May 28 - June 1, 2006, Proceedings. Volume 4004 of Lecture Notes in Computer Science., Springer (2006) 373–390
36. Roy, B.K., Meier, W., eds.: Fast Software Encryption, 11th International Workshop, FSE 2004, Delhi, India, February 5-7, 2004, Revised Papers. In Roy, B.K., Meier, W., eds.: FSE. Volume 3017 of Lecture Notes in Computer Science., Springer (2004)
37. Schroeppel, R.: Hasty Pudding Cipher. AES Submission (1998) `http://www.cs.arizona.edu/rcs/hpc/`.
38. Shrimpton, T., Terashima, R.S.: A Modular Framework for Building Variable-Input-Length Tweakable Ciphers. In Sako, K., Sarkar, P., eds.: Advances in Cryptology - ASIACRYPT 2013 - 19th International Conference on the Theory and Application of Cryptology and Information Security, Bengaluru, India, December 1-5, 2013, Proceedings, Part I. Volume 8269 of Lecture Notes in Computer Science., Springer (2013) 405–423
39. Struik, R.: Revisiting design criteria for AEAD ciphers targeting highly constrained networks. DIAC: Directions in Authenticated Ciphers (2013) `http://2013.diac.cr.yp.to/`.
40. 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)
41. Vitaletti, A., Palombizio, G.: Rijndael for Sensor Networks: Is Speed the Main Issue? Electr. Notes Theor. Comput. Sci. **171**(1) (2007) 71–81
42. Wang, P., Feng, D., Wu, W.: HCTR: A Variable-Input-Length Enciphering Mode. In Feng, D., Lin, D., Yung, M., eds.: Information Security and Cryptology, First SKLOIS Conference, CISC 2005, Beijing, China, December 15-17, 2005, Proceedings. Volume 3822 of Lecture Notes in Computer Science., Springer (2005) 175–188
43. Yuksel, E., Nielson, H.R., Nielson, F.: ZigBee-2007 Security Essentials. Proceedings of 13th Nordic Workshop on Secure IT-systems

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 $\mathsf{MiniCTR1} \stackrel{\text{def}}{=} \mathsf{MiniCTR}[\widetilde{\mathsf{P}}, \mathsf{R}]$, 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, and 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 $\perp$, 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 $\perp$) and i-th query, where the probability space is defined by $\widetilde{\mathsf{P}}$.

First we consider that 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 $A = \{j \in \mathcal{I}[V_i], j < i\}$ and $B = \{k \in \mathcal{I}'[V_i], k < i\}$ be the set of indexes for t- and v-queries using $V, V' = V_i$ done before $(\widehat{N}_i, L'_i, V'_i)$.

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

$$
\begin{aligned}
&\Pr[N'_i = x|\overline{\mathcal{E}_{i-1}}] \\
&\leq \sum_{\substack{x_j \in \{0,1\}^n, j \in A, \\ x'_k \in \{0,1\}^n : k \in B}} \Pr[N'_i = x|\overline{\mathcal{E}_{i-1}}, N_j = x_j, j \in A, N'_k = x'_k, k \in B] \Pr[N_j = x_j, j \in A, N'_k = x'_k, k \in B] \quad (6) \\
&\leq \max_{\substack{x_j \in \{0,1\}^n, j \in A, \\ x'_k \in \{0,1\}^n : k \in B}} \Pr[N'_i = x|\overline{\mathcal{E}_{i-1}}, N_j = x_j, j \in A, N'_k = x'_k, k \in B] \quad (7) \\
&\leq \frac{1}{2^n - (|A| + |B|)} \leq \frac{1}{2^n - (q + q')} \leq \frac{2}{2^n} \quad (8)
\end{aligned}
$$

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 $\Pr[\mathcal{E}_i|\overline{\mathcal{E}_{i-1}}] = \Pr[N_i = N'_j, \text{ for some } j \in B|\overline{\mathcal{E}_{i-1}}]$, which is at most

$$
\sum_{k \in B} \Pr[N'_k = N_i|\overline{\mathcal{E}_{i-1}}] \leq \sum_{k \in B} \max_{x \in \{0,1\}^n} \Pr[N'_k = x|\overline{\mathcal{E}_{i-1}}] \leq \frac{|B|}{2^n - (|A| + |B|)} \leq \frac{2q'}{2^n} \quad (9)
$$

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

$$
\mathtt{Adv}^{\mathtt{auth}}_{\mathsf{MiniCTR1}'}(\mathcal{A}) \leq \sum_{i=1,\ldots,q+q'} \Pr[\mathcal{E}_i|\overline{\mathcal{E}_{i-1}}] \leq \frac{2(q+q')(\omega+q')}{2^n}, \quad (10)
$$

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 (8) and (9) 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 first observe that unless a collision of $\underline{L}$, $\underline{C}$ is perfectly random. As $\underline{L} = \widetilde{\mathsf{P}}^{(A,\underline{M})}(\underline{N})$ and $\underline{N}$ is not colliding, the collision probability of $\underline{L}$ for any adversary is bounded by $\binom{q}{2}/2^n < q^2/2^{n+1}$. Thus we have

$$\mathtt{Adv}^{\mathtt{priv}}_{\mathsf{MiniCTR1}}(\mathcal{A}) \leq \frac{q^2}{2^{n+1}}. \tag{11}$$

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 $i = 1, \ldots, q$, 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 $\mathbf{Z} = ((N_1, A_1, M_1, L_1, C_1), \ldots, (N_q, A_q, M_q, L_q, C_q))$. 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 $T_i = (\Sigma_i, A_i, m_i, d_i)$ 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

$$\mathtt{Adv}^{\mathtt{auth}}_{\mathsf{MiniOCB1}}(\mathcal{A}) \leq \sum_{\mathbf{z}} \Pr_{\mathcal{A},\mathsf{MiniOCB1}}[N' \in \rho(\widehat{N}) | \mathbf{Z} = \mathbf{z}] \cdot \Pr_{\mathcal{A},\mathsf{MiniOCB1}}[\mathbf{Z} = \mathbf{z}] \tag{12}$$

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 (12) is bounded by

$$\max_{\mathbf{z} \text{ satisfies } \overline{\mathtt{Bad}_1}} \Pr[N' \in \rho(\widehat{N}) | \mathbf{Z} = \mathbf{z}, \overline{\mathtt{Bad}_1}] + \Pr[\mathtt{Bad}_1]. \tag{13}$$

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

$$\Pr[\mathtt{Bad}_1] \leq \binom{q}{2} \frac{1}{2^n} < \frac{q^2}{2^{n+1}} \tag{14}$$

holds true[5]. 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 $i = 1, \ldots, q$. 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

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

$$\leq \Pr[N' \in \rho(\widehat{N}) | \mathbf{Z} = \mathbf{z}, \overline{\mathtt{Bad}_1} \wedge \overline{\mathtt{Bad}_2}] + \Pr[\mathtt{Bad}_2 | \mathbf{Z} = \mathbf{z}, \overline{\mathtt{Bad}_1}] \tag{16}$$

$$\leq \frac{2\omega}{2^\nu} + \Pr[\mathtt{Bad}_2 | \mathbf{Z} = \mathbf{z}, \overline{\mathtt{Bad}_1}]. \tag{17}$$

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 $p = 0$.

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

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 $1/(2^n - 1) \leq 2/2^n$. 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

$$\Pr[\mathtt{Bad}_2 | \mathbf{Z} = \mathbf{z}, \overline{\mathtt{Bad}_1}] \leq \frac{2q}{2^n}. \tag{18}$$

Combining (12) to (18) we have

$$\mathtt{Adv}^{\mathtt{auth}}_{\mathsf{MiniOCB1}}(\mathcal{A}) \leq \frac{q^2}{2^{n+1}} + \frac{2q}{2^n} + \frac{2\omega}{2^\nu} \leq \frac{2.5q^2}{2^n} + \frac{2\omega}{2^\nu}, \tag{19}$$

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

$$\mathtt{Adv}^{\mathtt{priv}}_{\mathsf{MiniOCB1}}(\mathcal{A}) \leq \frac{q^2}{2^{n+1}} + \frac{q^2}{2^{n+1}} \leq \frac{q^2}{2^n}. \tag{20}$$

This concludes the proof of privacy bound.