Revisiting the Security of OleF, t-OleF, and OlÆF

Tingting Guo[1], Peng Wang[2(✉)], Shuping Mao[3], Yan Jia[4,5], and Jiadong Han[2]

[1] Ningbo Key Laboratory of Information Technology Application Innovation and Security, School of Cyber Science and Engineering, Ningbo University of Technology, Ningbo, China
guotingting@nbut.edu.cn
[2] School of Cryptology, University of Chinese Academy of Sciences, Beijing, China
p-wang@ucas.ac.cn, hanjiadong25@mails.ucas.ac.cn
[3] Beijing Electronic Science and Technology Institute, Beijing, China
maoshuping19@mails.ucas.ac.cn
[4] State Key Laboratory of Cyberspace Security Defense, Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China
jiayan22@mails.ucas.ac.cn
[5] School of Cyber Security, University of Chinese Academy of Sciences, Beijing, China

Abstract. We revisit the security of OleF, t-OleF, and OlÆF and identify two independent security flaws. First, we establish a mask-compensation relation between continuation and termination processing in OleF and t-OleF. Exploiting this relation, we construct a two-encryption-query distinguisher against OleF with advantage $1 - 2^{-2n}$. We further give two concrete AXU hash families for which the same attack applies to t-OleF, revealing flaws in the original DOSPRP and SPRTOP security proofs. We extend these attacks to OlÆF through valid encodings, breaking its privacy and integrity. In particular, the integrity forgery succeeds with probability 1 under full-domain verification when the target expansion length is 1 and extends to larger expansion lengths under suffix-only verification. Second, we refute the claimed ideal RUP integrity bound for variable-expansion online authenticated encryption. Even with an ideal tweakable online permutation, prior unverified decryption queries at smaller expansion lengths can reveal part of the target authentication suffix. We construct counterexamples showing that a single unverified decryption query or a single non-target verification query, followed by one forgery attempt, achieves a success probability of 3/8, exceeding the claimed bound of 1/4. We also derive generalized expressions for the forgery success probabilities of both attacks. Finally, we discuss the corresponding repair requirements.

Keywords: online cipher; OleF; t-OleF; OlÆF; encode-then-encipher; RUP integrity; domain separation.

1 Introduction

Online cipher. Online ciphers [3] are length-preserving encryption schemes that process messages block by block, where each ciphertext block depends only on the current and preceding plaintext blocks. This property makes them suitable for encrypting long messages, devices with limited storage, and low-latency communication. We focus on the *online-but-last* convention [1], which allows the final block to be processed differently from preceding blocks. In this convention, the online cipher and the tweakable online cipher are illustrated in fig. 1. For diblock-online

ciphers, Bhaumik and Nandi [6,7] introduced diblock-online strong pseudorandom permutation (DOSPRP) security. Bhattacharjee et al. [5] extended the setting to tweakable online ciphers and defined strong pseudorandom tweakable online permutation (SPRTOP) security.

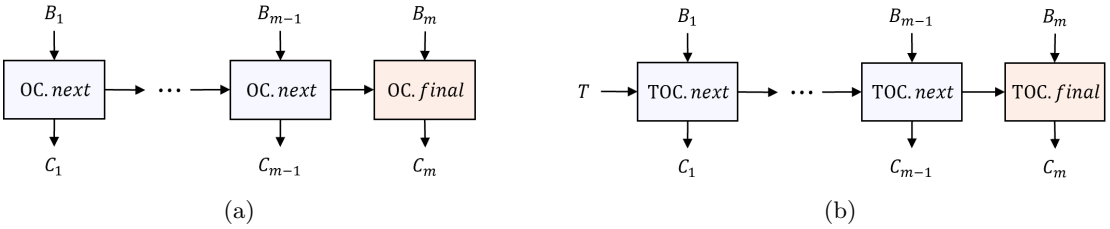

Fig. 1: (a) Online cipher and (b) tweakable online cipher, both following the online-but-last convention.

Online authenticated encryption. Online authenticated encryption (OAE) [8,10] aims to provide privacy and integrity while preserving online encryption. To support decryption in devices with limited memory or applications requiring immediate plaintext processing, Andreeva et al. [2] introduced the release of unverified plaintext (RUP) setting, where plaintext may be released before authentication is completed. Zhang and Wu [14] further studied the security of online AE schemes in this setting. Recently, Bhattacharjee et al. [5] proposed a variable-expansion OAE framework in the RUP setting based on the encode-then-encipher (EtE) paradigm. It encodes a message by adding public redundancy and then encrypts the result using a tweakable online cipher. The framework provides encryption, unverified decryption, and verification interfaces, with privacy and integrity defined against adversaries that may make unverified decryption queries.

From OleF to t-OleF and OlÆF. Bhaumik and Nandi [7] proposed OleF, an inverse-free diblock-online cipher that supports variable-length messages with either a complete or a partial final diblock. It uses a four-round Feistel-type structure to process each complete diblock and updates a chaining state between diblocks. OleF follows the online-but-last convention. Let T denote the chaining state updated after processing the preceding block. It uses b_2T for continuation and b_3T for termination. It was claimed to achieve DOSPRP security under a pseudorandom function (PRF) assumption on the underlying keyed function [7].

Bhattacharjee et al. [5] later proposed t-OleF, a tweakable version of OleF. The only difference lies in the initial chaining state: OleF uses zero, whereas t-OleF uses $H_{K'}(A)$, with $H_{K'}$ being an independently keyed almost-XOR-universal (AXU) hash function and A the associated data serving as the external tweak. The authors claimed that t-OleF achieves SPRTOP security under the PRF and AXU assumptions.

Furthermore, Bhattacharjee et al. [5] proposed OlÆF, a variable-expansion online authenticated encryption scheme based on the encode-then-encipher (EtE) paradigm [4]. The encoding layer appends a suffix of the form $10^{\tau-1}$ to the message, where τ denotes the ciphertext expansion length, while the enciphering layer uses t-OleF. The authors claimed that OlÆF achieves privacy and integrity security in the RUP setting under the PRF assumption on the underlying block cipher and the AXU assumption on the hash function.

Motivations. The claimed security of OleF, t-OleF, and OlÆF raises several concerns.

First, the ideal online-but-last model treats continuation and termination processing as independent, whereas OleF and t-OleF use the same round function and distinguish the two operations only through different masks b_2T and b_3T. This design may introduce exploitable relations between continuation and termination processing, potentially undermining the claimed DOSPRP and SPRTOP security of OleF and t-OleF and, consequently, the privacy and integrity of OlÆF.

Second, the actual verification algorithm of OlÆF checks only the authentication suffix, whereas the verification algorithm used in the generic EtE OAE security proof in the same work also checks whether the decoded value corresponds to a valid message in the message space. These two verification algorithms have different acceptance conditions, which may affect the integrity.

Moreover, the claimed ideal RUP integrity bound depends only on the number of forgery attempts and does not explicitly account for information leaked by prior unverified decryption queries or non-target verification queries. Since queries at different expansion lengths can use the same inverse permutation output, such leakage may affect the forgery success probability.

Our contributions. We revisit the security of OleF, t-OleF, and OlÆF and identify two independent security flaws. The main contributions are as follows.

(1) **Constructing structural attacks on OleF and t-OleF.** We identify a mask-compensation relation between continuation and termination processing in OleF and t-OleF, and exploit it to construct a two-encryption-query distinguisher against OleF with advantage $1 - 2^{-2n}$. For t-OleF, we give concrete AXU hash families for which the same attack applies. We further identify flaws in the original DOSPRP and SPRTOP security proofs, which fail to account for collisions between continuation and termination processing.

(2) **Extending the attacks to the privacy and RUP integrity of OlÆF.** We show that the attack on t-OleF can be extended to OlÆF through valid encodings, yielding a two-encryption-query privacy distinguisher with advantage $1 - 2^{-2n}$. We also construct an RUP integrity forgery requiring only one encryption query and one forgery attempt. The forgery succeeds with probability 1 when the target expansion length is $\tau^* = 1$, even under full-domain verification. Under the suffix-only verification specified by OlÆF, the forgery succeeds with probability 1 for all $1 \leq \tau^* \leq \min\{t, 2n\}$.

(3) **Refuting the claimed RUP integrity bound for variable-expansion EtE OAE.** We further examine the generic variable-expansion EtE OAE construction underlying OlÆF, even when its encipher layer is an ideal tweakable online permutation. We show that the claimed RUP integrity bound $q'/2^{\tau^*}$ with q' the number of forgery attempts and τ^* the target expansion length can be violated using either prior unverified decryption queries at smaller expansion lengths or non-target verification queries at different expansion lengths. In particular, a single query of either type followed by one forgery attempt achieves a success probability of 3/8, exceeding the claimed bound of 1/4. We also generalize both counterexamples and identify the flaw in the original RUP integrity proof.

(4) **Proposing repair requirements.** First, we lift the AXU condition on the hash function to coefficient-AXU to prevent publicly predictable mask differences. Second, we require full-domain verification to reject ciphertexts outside the legal encoding domain. Third, we bind

the expansion length to the tweak of the underlying online cipher to prevent cross-expansion leakage.

Table 1 summarizes the originally claimed security bounds, the resources and parameters of our attacks, and the corresponding distinguishing advantages or forgery success probabilities.

Table 1: Summary of the originally claimed security bounds, attack resources and parameters, and attack results for OleF, t-OleF, OlÆF and variable-expansion EtE OAE with an ideal tweakable online cipher (TOC). Here, q_e, q_d, q_v, and q' denote the numbers of encryption queries, unverified decryption queries, non-target verification queries, and forgery attempts at the target expansion length, respectively. The total number of queries is $q = q_e + q_d + q_v + q'$. The parameters σ_d and σ_b denote the total numbers of complete diblocks and blocks across all queries, respectively. Moreover, τ denotes the expansion length used in non-target decryption or verification queries, τ^* is the target expansion length for forgery, n is the input and output length of the underlying keyed function or block cipher, t is the upper bound of every expansion length and ϵ is the AXU bound of the universal hash family. Here, R_f and R_E denote the PRF advantages against the underlying keyed function f and block cipher E, respectively.

Construction	Security	Originally claimed bound	Our attack	
			Attack resources and parameters	Distinguishing advantage or forgery success probability
OleF	DOSPRP	$R_f + \frac{7\sigma_d^2}{2^n}$ [7]	$q_e = 2, \sigma_d = 3$	$1 - 2^{-2n}$
t-OleF	SPRTOP	$R_E + \frac{7\sigma_b^2}{2^n} + 3q^2\epsilon$ [5]	$q_e = 2, \sigma_b = 6$	$1 - 2^{-2n}$
OlÆF	privacy	$R_E + \frac{7\sigma_b^2}{2^n} + 3q^2\epsilon$ [5]	$q_e = 2, \sigma_b = 6$	$1 - 2^{-2n}$
	integrity (suffix-only)	$R_E + \frac{7\sigma_b^2}{2^n} + 3q^2\epsilon + \frac{q'}{2^{\tau^*}}$ [5]	$q_e = 1, q' = 1, \sigma_b = 6, 1 \le \tau^* \le \min\{t, 2n\}$	1
	integrity (full-domain variant)	Not explicitly claimed	$q_e = 1, q' = 1, \sigma_b = 6, \tau^* = 1$	1
Variable-expansion EtE OAE (ideal TOC)	integrity	$\frac{q'}{2^{\tau^*}}$ [5]	$q_d = 1, q' = 1, \tau = 1, \tau^* = 2$	3/8
			$q_d = \ell, q' = 1, \tau < \tau^*$	$\frac{1-\left(1-2^{\tau-\tau^*}\right)^{\ell+1}}{2^\tau}$
			$q_v = 1, q' = 1, \tau = 1, \tau^* = 2$	3/8
			$q_v = \ell, q' = 1, \tau \neq \tau^*$	$\frac{2^{(\ell+1)\tau}-1}{2^{\tau^*}+\ell\tau(2^\tau-1)}$

Related works. Public cryptanalysis of OleF, t-OleF and OlÆF remains limited. To the best of our knowledge, Guo et al. [9] reported a plaintext-recovery attack on OleF in the quantum chosen-ciphertext setting. We are not aware of any previously published classical attacks on OleF, t-OleF, or OlÆF in the literature we surveyed.

Reyhanitabar et al. [13] studied AE schemes with variable authentication tag lengths and constructed cross-expansion forgeries by exploiting a prefix relation between ciphertext differences under different tag lengths. Our counterexamples exploit a different mechanism: for the same associated-data value and ciphertext, the variable-expansion EtE OAE construction uses the same inverse-permutation output regardless of the expansion length. Consequently, an unverified decryption query at a shorter expansion length can reveal bits needed for a forgery at a longer expansion length, while a non-target verification query at a different expansion length can reveal information about the target authentication suffix through its accept/reject result.

For the online encryption mode POE and its authenticated encryption scheme POET, which also use AXU hash functions, Nandi [12] showed that AXU is insufficient to guarantee security.

For the online cipher POEx, which uses pAXU hash functions, a variant of AXU, Jha and Nandi [11] showed that pAXU is insufficient to guarantee security. However, the requirements these papers impose on the UHF differ from ours. The security claim challenged in [12] requires the hash function to satisfy the corresponding differential probability bound even after repeated composition, and the original security proof examined in [11] requires the pAXU property to hold for iterated inputs that depend on the hash key. Our requirement is instead that the XOR difference between hash outputs weighted by different coefficients have a bounded differential probability.

Organization. The rest of this paper is organized as follows. Section 2 introduces the necessary preliminaries and reviews OleF, t-OleF, and OlÆF. Section 3 presents structural attacks on OleF and t-OleF. Section 4 extends these attacks to the privacy and integrity of OlÆF. Section 5 identifies a flaw in the ideal RUP integrity bound for variable-expansion OAE. Section 6 discusses possible repairs. Finally, we conclude the paper in section 7.

2 Preliminary

2.1 Notations

We use '$\oplus$' for field addition and XOR of equal-length strings, and '$\|$' for concatenation. Let $\mathsf{lsb}_r(x)$ denote the least significant r bits of x and $\mathsf{msb}_r(x)$ denote the most significant r bits of x. Let trunc_r denote truncation that removes the least significant r bits. Let $|x|$ be the length of the binary string x. Let ε denote the empty string. Let n be the underlying block-cipher block length or the input and output length of the underlying keyed function. Each complete diblock consists of two n-bit blocks.

2.2 Security Definitions for Cryptographic Primitives

PRF Let $F : \mathcal{K} \times \{0,1\}^n \to \{0,1\}^n$ be a keyed function, where $\mathcal{K}$ is the key space. Let $\mathrm{Func}(n,n)$ denote the set of all functions from $\{0,1\}^n$ to $\{0,1\}^n$. The PRF advantage of an adversary $\mathcal{A}$ against F is defined as

$$\mathrm{Adv}_F^{\mathrm{prf}}(\mathcal{A}) = \left| \Pr_{K \xleftarrow{\$} \mathcal{K}} [\mathcal{A}^{F_K} = 1] - \Pr_{\rho \xleftarrow{\$} \mathrm{Func}(n,n)} [\mathcal{A}^{\rho} = 1] \right|.$$

The function family F is PRF-secure if this advantage is negligible for every efficient adversary $\mathcal{A}$.

AXU Hash Function Let $H : \mathcal{K}' \times \mathcal{X} \to \mathcal{Y}$ be a keyed hash function where $\mathcal{K}'$ is the key space, $\mathcal{X}$ is the domain and $\mathcal{Y}$ is the range. Hash function H is said to be ϵ-AXU if for any two distinct strings $X, X' \in \mathcal{X}$ and any $\Delta \in \mathcal{Y}$,

$$\Pr[K' \xleftarrow{\$} \mathcal{K}' : H_{K'}(X) \oplus H_{K'}(X') = \Delta] \leq \epsilon.$$

Security Definitions for (Tweakable) Online Cipher We now introduce the DOSPRP security definition for online ciphers and the SPRTOP security definition for tweakable online ciphers, respectively.

DOSPRP security [7]. Let $\mathrm{OC} : \mathcal{K} \times \mathcal{M}_D \mapsto \mathcal{M}_D$ be an online cipher, where $\mathcal{K}$ is the key space and message space is $\mathcal{M}_D = \{M \in \{0,1\}^* : |M| \geq 2n\}$. Its message is parsed into $M = M_1 \parallel \cdots \parallel M_r \parallel M_{r+1}$, where each nonfinal M_i is a diblock which satisfies $|M_i| = 2n$ and the final M_{r+1} satisfies $2n \leq |M_{r+1}| \leq 4n-1$. Under the online-but-last convention, nonfinal diblocks undergo continuation processing, while the final diblock undergoes termination processing, even when it is a complete diblock. Its ideal object is a random online permutation $\widetilde{\pi} \xleftarrow{\$} \mathrm{DOPerm}$, where DOPerm denotes the set of all online permutations $\widetilde{P} : \mathcal{M}_D \mapsto \mathcal{M}_D$ of the following form:

$$\widetilde{P}(M) = h_{\varepsilon}(M_1) \| h_{M_1}(M_2) \| \cdots \left\| h_{M_1 \| \cdots \| M_{r-1}}(M_r) \right\| g_{|M_{r+1}|, M_1 \| \cdots \| M_r}(M_{r+1}).$$

Here the tweakable permutation $h : \mathcal{T}' \times \{0,1\}^{2n} \to \{0,1\}^{2n}$ represents the continuation operations, and another independent tweakable permutation $g_s : \mathcal{T}' \times \{0,1\}^s \to \{0,1\}^s$ for $2n \leq s \leq 4n-1$ represents the termination operations. The set $\mathcal{T}'$ is an index space containing all diblock-aligned plaintext prefixes and unambiguous encodings of tweak-prefix pairs. In the degenerate case where the message only consists of M_1 with $2n \leq |M_1| \leq 4n-1$, $\widetilde{P}(M) = g_{|M_1|,\varepsilon}(M_1)$. The DOSPRP advantage of an adversary $\mathcal{A}$ against OC is defined as

$$\mathrm{Adv}^{\mathrm{dosprp}}_{\mathrm{OC}}(\mathcal{A}) = \left| \Pr_{K \xleftarrow{\$} \mathcal{K}} \left[\mathcal{A}^{\mathrm{OC}_K, \mathrm{OC}_K^{-1}} \Rightarrow 1 \right] - \Pr_{\widetilde{\pi} \xleftarrow{\$} \mathrm{DOPerm}} \left[\mathcal{A}^{\widetilde{\pi}, \widetilde{\pi}^{-1}} \Rightarrow 1 \right] \right|.$$

If this advantage is negligible for every efficient adversary $\mathcal{A}$, then OC achieves DOSPRP security.

SPRTOP security [5]. Let $\mathcal{T}$ be the external tweak space. Let $\mathrm{TOC} : \mathcal{K} \times \mathcal{T} \times \mathcal{M}_D \mapsto \mathcal{M}_D$ be a tweakable online cipher where $\mathcal{K}, \mathcal{M}_D$ and the message parsing are the same as in the DOSPRP definition. Under the online-but-last convention, nonfinal diblocks undergo continuation processing, while the final diblock undergoes termination processing, even when it is a complete diblock. Its ideal object is a random tweakable online permutation $\pi \xleftarrow{\$} \mathrm{TOPerm}$, where TOPerm denotes the set of all tweakable online permutations $P : \mathcal{T} \times \mathcal{M}_D \mapsto \mathcal{M}_D$ of the following form:

$$P_T(M) = h_T(M_1) \parallel h_{T \| M_1}(M_2) \parallel \cdots \parallel h_{T \| M_1 \| \cdots \| M_{r-1}}(M_r) \parallel g_{|M_{r+1}|, T \| M_1 \| \cdots \| M_r}(M_{r+1}).$$

Here h, g_s are the same as in the DOSPRP definition. In the degenerate case where the message only consists of M_1 with $2n \leq |M_1| \leq 4n-1$, $P_T(M) = g_{|M_1|,T}(M_1)$. The SPRTOP advantage of an adversary $\mathcal{A}$ against TOC is defined as

$$\mathrm{Adv}^{\mathrm{sprtop}}_{\mathrm{TOC}}(\mathcal{A}) = \left| \Pr_{K \xleftarrow{\$} \mathcal{K}} \left[\mathcal{A}^{\mathrm{TOC}_K, \mathrm{TOC}_K^{-1}} \Rightarrow 1 \right] - \Pr_{\pi \xleftarrow{\$} \mathrm{TOPerm}} \left[\mathcal{A}^{\pi, \pi^{-1}} \Rightarrow 1 \right] \right|.$$

If this advantage is negligible for every efficient adversary $\mathcal{A}$, then TOC achieves SPRTOP security.

Security Definitions for Variable-expansion OAE Based on the EtE Paradigm under RUP Reference [5] defines the syntax and security of variable-expansion OAE schemes in the

EtE mode. The EtE construction has two distinct layers. A public encoder first adds checkable redundancy. A length-preserving tweakable online cipher then enciphers the entire encoded string. Let A be the associated data, $\mathcal{T}''$ the expansion length space, $\tau \in \mathcal{T}''$ the expansion length, $\mathcal{M}_D^-$ the message space, $M \in \mathcal{M}_D^-$ the message, and C the ciphertext. Let $\phi : \mathcal{T}'' \times \mathcal{M}_D^- \to \mathcal{M}_D$ be the encoder, which is required to satisfy that for every expansion length $\tau \in \mathcal{T}''$ and message $M \in \mathcal{M}_D^-$ there exists some $Y \in \{0,1\}^\tau$ such that $\phi(\tau, M) = M \parallel Y$. For example, the encoder can be $\phi_{10^*}(\tau, M) = M \parallel 10^{\tau-1}$. The map ϕ is injective over pairs (τ, M). Let the encoded space be $\mathcal{D}_\tau = \{\phi(\tau, M) : M \in \mathcal{M}_D^-\}$. Let P be a tweakable online cipher. In the RUP setting, the adversary may make unverified decryption queries before authentication succeeds. The OAE scheme then consists of an encryption function Enc, a decryption function Dec, and a verification function Ver as follows.

$$\begin{aligned}
&\mathrm{Enc}(A, \tau, M) = P_A(\phi(\tau, M)),\\
&\mathrm{Dec}(A, \tau, C) = \mathrm{trunc}_\tau\left(P_A^{-1}(C)\right),\\
&\mathrm{Ver}(A, \tau, C) = \begin{cases} \mathsf{accept}, & P_A^{-1}(C) \in \mathcal{D}_\tau,\\ \mathsf{reject}, & \text{otherwise.} \end{cases}
\end{aligned}$$

The verification function of OAE is called *full-domain verification*, which outputs accept when

$$P_A^{-1}(C) \in \mathcal{D}_\tau \quad \Longleftrightarrow \quad \mathsf{lsb}_\tau(P_A^{-1}(C)) = 10^{\tau-1} \text{ and } m_{\min} \le |P_A^{-1}(C)| - \tau \le m_{\max}.$$

All queries are required to satisfy the input-length constraints of the corresponding interfaces.

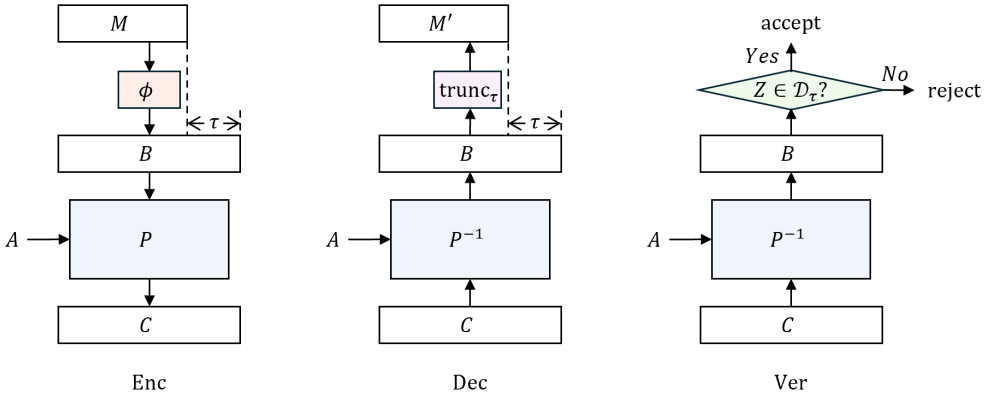

Fig. 2: Variable-expansion OAE based on the EtE paradigm under RUP.

Privacy. In the real world, the adversaries are allowed to make queries to Enc, Dec and Ver of OAE. In the ideal world, let $\pi \xleftarrow{\$}$ TOPerm and

$$\begin{aligned}
&\mathrm{Enc}_0(A, \tau, M) = \pi_A(\phi(\tau, M)),\\
&\mathrm{Dec}_0(A, \tau, C) = \mathrm{trunc}_\tau\left(\pi_A^{-1}(C)\right),\\
&\mathrm{Ver}_0(A, \tau, C) = \begin{cases} \mathsf{accept}, & \pi_A^{-1}(C) \in \mathcal{D}_\tau,\\ \mathsf{reject}, & \text{otherwise.} \end{cases}
\end{aligned}$$

The adversaries are allowed to make queries to $\mathrm{Enc}_0, \mathrm{Dec}_0$ and Ver_0. In both worlds, adversaries are not allowed to query the verification oracle with input (A, τ', C) on a ciphertext C obtained directly from an encryption query with input (A, τ, M). For an OAE scheme, the distinguishing advantage in the privacy security definition is

$$\mathrm{Adv}^{\text{priv-rup}}_{\mathrm{OAE}}(\mathcal{A}) = \left| \Pr_{K \xleftarrow{\$} \mathcal{K}} \left[\mathcal{A}^{\mathrm{Enc},\mathrm{Dec},\mathrm{Ver}} \Rightarrow 1\right] - \Pr_{\pi \xleftarrow{\$} \mathrm{TOPerm}} \left[\mathcal{A}^{\mathrm{Enc}_0,\mathrm{Dec}_0,\mathrm{Ver}_0} \Rightarrow 1\right] \right|.$$

If this advantage is negligible for every adversary $\mathcal{A}$, meaning no adversary can distinguish these two worlds, then OAE achieves privacy.

Integrity. Fix a target expansion value τ^* and give it to adversaries. The adversaries may query Enc, Dec and Ver. Verification queries at the target expansion length τ^* are counted as forgery attempts. They are not allowed to query $\mathrm{Ver}(A, \tau', C)$ on a ciphertext C obtained directly from $\mathrm{Enc}(A, \tau, M)$. Let Forge_{τ^*} denote the event that at least one admissible forgery attempt (A, τ^*, C) such that $\mathrm{Ver}(A, \tau^*, C) = \mathsf{accept}$. Thus the advantage in the integrity security definition is

$$\mathrm{Adv}^{\text{int-rup}}_{\mathrm{OAE}}(\mathcal{A}) = \Pr_{K \xleftarrow{\$} \mathcal{K}} \left[\mathrm{Forge}_{\tau^*}\right].$$

The OAE scheme is said to satisfy integrity if the forgery success probability is negligible for every efficient adversary.

2.3 OleF and t-OleF

OleF is an online cipher, and t-OleF is its tweakable version, that is, a tweakable online cipher. In OleF, the round function f can be any keyed n-bit function and need not be invertible. In t-OleF, the round function is $f = E_K$, where E is an n-bit block cipher under key K. Let $H_{K'}$ be an AXU hash function with n-bit outputs under an independent key K'. The public linear transformations b_1, b_2, b_3 are chosen according to the original constructions. In the field-multiplication instantiation considered here, they are represented by distinct nonzero field elements $b_1, b_2, b_3 \in \mathbb{F}_{2^n}$, and satisfy the additional conditions required by the original constructions. The notation $b_i X$ denotes multiplication by b_i in $\mathbb{F}_{2^n}$. Let A denote the associated data, which serves as the tweak for t-OleF and is not used by OleF. The initialisation routine is

$$\mathrm{Init}_{\mathrm{mode}}(A) = \begin{cases} 0, & \mathrm{mode} = \text{OleF}, \\ H_{K'}(A), & \mathrm{mode} = \text{t-OleF}. \end{cases} \tag{1}$$

The encryption and decryption procedures for messages consisting entirely of complete diblocks are given in algorithm 1, and the encryption process is illustrated in fig. 3. The processing of an incomplete final diblock follows the original constructions [7,5] and is omitted here, as our attacks use only complete diblocks. For complete diblocks, OleF and t-OleF use the same round function for continuation and termination processing, but with different masks. Specifically, continuation processing uses $a = b_2 T$, while termination processing uses $a = b_3 T$, where T is the current chaining state. The shared round function and its inverse are specified in algorithm 2, and the forward round function is illustrated in fig. 4.

The original papers claimed DOSPRP and SPRTOP security bounds for OleF and t-OleF, respectively. Specifically, Bhaumik and Nandi [7] gave the following DOSPRP bound for OleF

Algorithm 1 OleF/t-OleF encryption and decryption for complete diblocks.

procedure CIPHERENC(mode, A, B)
  $(B_1, \ldots, B_m) \leftarrow \text{Parse}(B)$
  $T_1 \leftarrow \text{Init}_{\text{mode}}(A)$
  for $j = 1, \ldots, m-1$ **do**
    $(T_{j+1}, C_j) \leftarrow \text{Round}(b_2, T_j, B_j)$
  end for
  $(T_{m+1}, C_m) \leftarrow \text{Round}(b_3, T_m, B_m)$
  return $C_1 \| \cdots \| C_m$
end procedure

procedure CIPHERDEC(mode, A, C)
  $(C_1, \ldots, C_m) \leftarrow \text{Parse}(C)$
  $T_1 \leftarrow \text{Init}_{\text{mode}}(A)$
  for $j = 1, \ldots, m-1$ **do**
    $(T_{j+1}, B_j) \leftarrow \text{InvRound}(b_2, T_j, C_j)$
  end for
  $(T_{m+1}, B_m) \leftarrow \text{InvRound}(b_3, T_m, C_m)$
  return $B_1 \| \cdots \| B_m$
end procedure

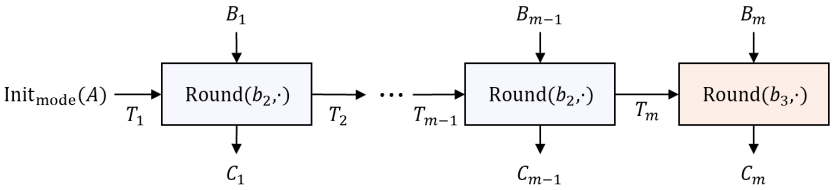

Fig. 3: Encryption of OleF and t-OleF for $m \geq 2$ complete diblocks, where $T_1 = 0$ for OleF and $T_1 = H_{K'}(A)$ for t-OleF.

with q queries containing a total of σ_d complete diblocks:

$$\text{Adv}^{\text{dosprp}}_{\text{OleF}}(\mathcal{A}) \leq \text{Adv}^{\text{prf}}_{f}(\mathcal{A}') + \frac{7\sigma_d^2}{2^n}. \tag{2}$$

For t-OleF instantiated with an n-bit block cipher E and an additional ϵ-AXU hash function H, Bhattacharjee et al. [5] claimed the following SPRTOP bound for q queries containing at most σ_b blocks:

$$\text{Adv}^{\text{sprtop}}_{\text{t-OleF}}(\mathcal{A}) \leq \text{Adv}^{\text{prf}}_{E}(\mathcal{A}') + \frac{7\sigma_b^2}{2^n} + 3q^2\epsilon. \tag{3}$$

2.4 OlÆF

OlÆF is obtained by instantiating the variable-expansion OAE based on the EtE paradigm defined in section 2.2 with the tweakable online cipher t-OleF. Fix integers $\ell_{\max} \geq 2$ and

Algorithm 2 Shared complete-diblock round and inverse for OleF and t-OleF; $c \in \{b_2, b_3\}$ and every call uses the same forward f.

procedure ROUND(c, T, B)
  $a \leftarrow cT$
  $(L, R) \leftarrow B$
  $X \leftarrow f(L) \oplus R \oplus a$
  $Y \leftarrow f(X) \oplus L$
  $V \leftarrow f(Y) \oplus X$
  $U \leftarrow b_1 f(V) \oplus Y \oplus a$
  $T^+ \leftarrow X \oplus Y$
  return $(T^+, U \| V)$
end procedure

procedure INVROUND(c, T, C)
  $a \leftarrow cT$
  $(U, V) \leftarrow C$
  $Y \leftarrow U \oplus b_1(f(V)) \oplus a$
  $X \leftarrow V \oplus f(Y)$
  $L \leftarrow Y \oplus f(X)$
  $R \leftarrow X \oplus f(L) \oplus a$
  $T^+ \leftarrow X \oplus Y$
  return $(T^+, L \| R)$
end procedure

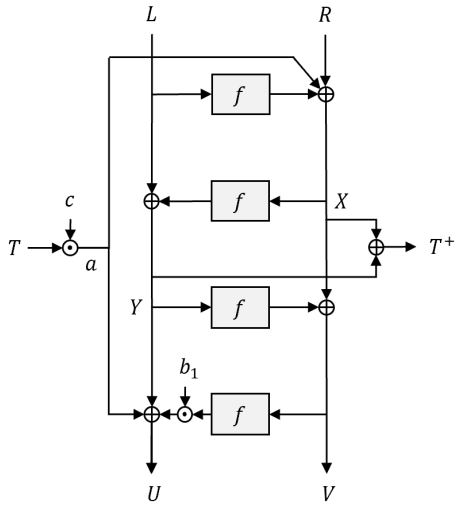

Fig. 4: Shared diblock round function Round$(\cdot,\cdot,\cdot)$ for OleF and t-OleF.

$1 \le t \le 2n(\ell_{\max}-1)$, and set $m_{\min} = 2n-1, m_{\max} = 2n\ell_{\max} - t$. The message space is $\mathcal{M}_D^- = \{M : m_{\min} \le |M| \le m_{\max}\}$, the expansion length space is $\mathcal{T}'' = \{\tau : 1 \le \tau \le t\}$. OlÆF uses the injective suffix-pad $\phi(\tau, M) = \phi_{10^*}(\tau, M) = M \,\|\, 10^{\tau-1}$. For independently sampled keys K and K' as above, the tweakable online cipher is instantiated as

$$P_A(B) = \text{CipherEnc}(\text{t-OleF}, A, B), \qquad P_A^{-1}(C) = \text{CipherDec}(\text{t-OleF}, A, C).$$

Here, CipherEnc and CipherDec refer to the full t-OleF algorithms, including the processing of incomplete final diblocks described in [7,5]. The algorithms given above specify their complete-diblock cases.

OlÆF inherits the encryption and unverified decryption interfaces from the generic EtE OAE construction, but its verification interface differs. It omits the message-length check and requires only that $\mathsf{lsb}_\tau(P_A^{-1}(C)) = 10^{\tau-1}$. To distinguish the two, we denote the *suffix-only verification* in OlÆF as

$$\text{Ver}_{\text{suf}}(A, \tau, C) = \begin{cases} \mathsf{accept}, & \mathsf{lsb}_\tau(P_A^{-1}(C)) = 10^{\tau-1}, \\ \mathsf{reject}, & \text{otherwise.} \end{cases}$$

The security analysis of OlÆF in [5] is based on the generic variable-expansion EtE OAE construction using a TOC. Specifically, [5, Theorem 2] states the following privacy and integrity bounds:

$$\text{Adv}^{\text{priv}}_{\text{OAE}}(\mathcal{A}) \le \text{Adv}^{\text{sprtop}}_{\text{TOC}}(\mathcal{A}') \tag{4}$$

and

$$\text{Adv}^{\text{int}}_{\text{OAE}}(\mathcal{A}) \le \text{Adv}^{\text{sprtop}}_{\text{TOC}}(\mathcal{A}') + \frac{q'}{2^{\tau^*}}, \tag{5}$$

where $\tau^* \le 2n$ and q' is the number of forgery attempts at the target expansion length τ^*. By combining these bounds with the claimed SPRTOP security of t-OleF in eq. (3), the authors derive the privacy and integrity bounds for OlÆF in [5, Theorem 3].

Remark. In fact, there is a mismatch between the generic EtE construction and the specified OlÆF scheme. As discussed above, OlÆF uses suffix-only verification, whereas the generic EtE

construction uses full-domain verification. Unlike full-domain verification, suffix-only verification may accept ciphertexts whose decoded values do not belong to $\mathcal{D}_\tau$. Consequently, the security reductions for the generic EtE construction cannot be directly applied to establish the claimed privacy and integrity bounds for OlÆF as specified.

3 Attacks on OleF and t-OleF

Under the online-but-last convention, continuation and termination processing are modeled by independent random permutations. However, OleF and t-OleF use the same round function with different masks for these two operations on complete diblocks. In this section, we identify a mask-compensation relation between the two operations and exploit it to construct a two-encryption-query distinguisher against OleF and, for certain AXU hash families, t-OleF.

3.1 Attack Essence: Mask Compensation Relationship in OleF and t-OleF

The round functions of both OleF and t-OleF, shown in algorithm 2, use the following core permutation $F_a : \{0,1\}^{2n} \to \{0,1\}^{2n}$:

$$X = f(L) \oplus R \oplus a, \qquad Y = f(X) \oplus L, \tag{6}$$

$$V = f(Y) \oplus X, \qquad U = b_1 f(V) \oplus Y \oplus a. \tag{7}$$

Two public translations on diblocks are defined as

$$S_\Delta(L,R) = (L, R \oplus \Delta), \qquad Q_\Delta(U,V) = (U \oplus \Delta, V). \tag{8}$$

The following lemma will establish that F_a has the mask compensation flaw. This flaw first underlies the attacks developed in section 3 against the DOSPRP security of OleF and the SPRTOP security of t-OleF. More significantly, it further compromises the privacy and integrity of OlÆF, as demonstrated in section 4.

Lemma 1 (Mask compensation). *For every f, every mask a, every Δ, and every diblock $B = (L,R)$,*

$$F_{a\oplus\Delta}(S_\Delta B) = Q_\Delta F_a(B). \tag{9}$$

The two computations also produce the same next state $X \oplus Y$.

Proof. For input $B = (L,R)$ to F_a, the values X, Y, U, V are given by eqs. (6) and (7). We now consider the input $B' = S_\Delta B = (L, R \oplus \Delta)$ to $F_{a\oplus\Delta}$, for which

$$X' = f(L) \oplus (R \oplus \Delta) \oplus (a \oplus \Delta) = X.$$

Consequently $Y' = f(X') \oplus L = Y$ and $V' = f(Y') \oplus X' = V$. Its remaining output is

$$U' = b_1 f(V') \oplus Y' \oplus a \oplus \Delta = U \oplus \Delta.$$

These two cases are shown on the left and right sides of fig. 5, respectively. Hence $F_{a\oplus\Delta}(S_\Delta B) = Q_\Delta F_a(B)$ always holds. The above derivation also yields $X' \oplus Y' = X \oplus Y$, which implies that the next state in OleF and t-OleF remains unchanged.

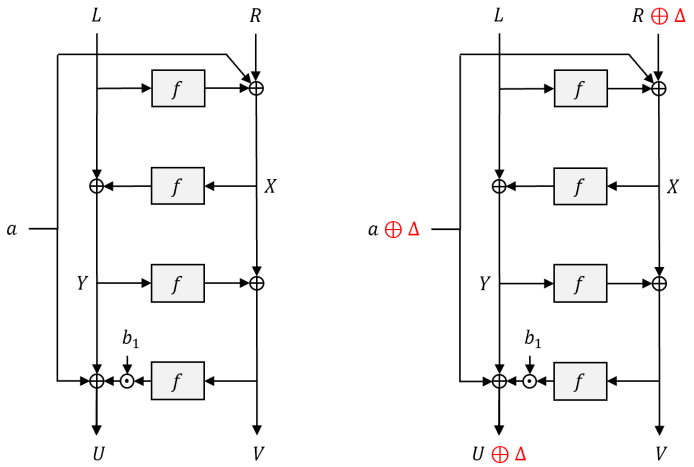

Fig. 5: Computation of $F_a(L, R)$ on the left and $F_{a\oplus\Delta}(L, R \oplus \Delta)$ on the right.

3.2 Two-Query Attack on the Security of OleF and t-OleF

We first introduce a generic attack strategy against OleF and t-OleF under a certain assumption, and then show separately that instances satisfying this assumption exist for OleF and for t-OleF.

The generic attack under assumption. Following the encryption procedure of OleF and t-OleF in algorithm 1, we make the following assumption.

Assumption. There exist associated data A and A' such that the mask difference $\Delta = b_2T_1 \oplus b_3T_1'$ is publicly computable, where $T_1 = \text{Init}_{\text{mode}}(A)$ and $T_1' = \text{Init}_{\text{mode}}(A')$.

Based on this assumption, the generic attack proceeds as follows.

(1) Query the encryption oracle on the associated data A and the two-diblock plaintext $B_1 \parallel B_2$. Then obtain the corresponding ciphertext $C_1 \parallel C_2$.
(2) Query the encryption oracle on the associated data A' and the one-diblock plaintext $B_1' := S_\Delta B_1$. Then obtain the corresponding ciphertext C_1'.
(3) Check whether $C_1' = Q_\Delta C_1$ holds.

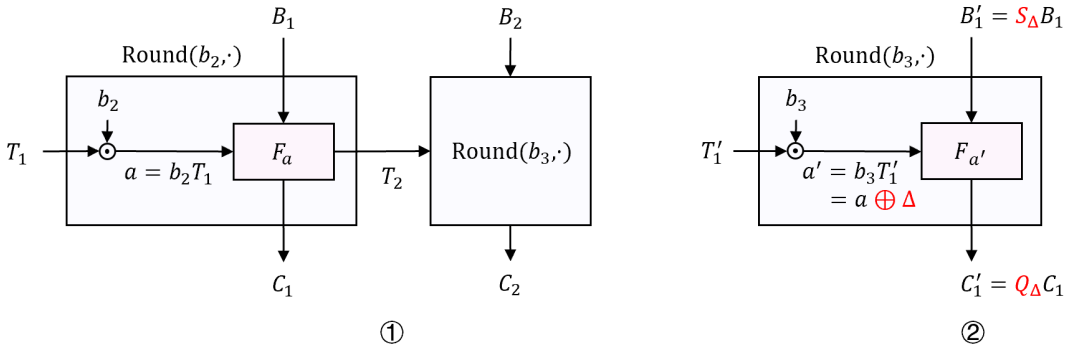

Fig. 6: Two encryption queries to OleF or t-OleF.

If the construction is OleF or t-OleF, the two encryption queries to the construction are illustrated in fig. 6. In this case, the first data block B_1 of the first query is followed by other data blocks, so the first mask is $a = b_2 T_1$. The first data block B'_1 of the second query is also the last data block, so the first mask is $a' = b_3 T'_1 = a \oplus \Delta$. Then by lemma 1 we have

$$C'_1 = F_{a'}(B'_1) = F_{a\oplus\Delta}(S_\Delta B_1) = Q_\Delta(F_a(B_1)) = Q_\Delta C_1$$

with certainty. In contrast, for a random online permutation or a random tweakable online permutation, C_1 is not the last ciphertext block and C'_1 is the last ciphertext block, so they are generated by two independent random permutations, and the probability that $C'_1 = Q_\Delta C_1$ holds is 2^{-2n}. Hence the distinguishing advantage of this attack is $1 - 2^{-2n}$.

Application of the attack to OleF and t-OleF. For OleF, $T_1 = T'_1 = 0$, hence $\Delta = b_2 \cdot 0 \oplus b_3 \cdot 0$ is public. Therefore, it admits a two-encryption-query distinguisher against DOSPRP security of advantage $1 - 2^{-2n}$.

For t-OleF, $T_1 = H_{K'}(A), T'_1 = H_{K'}(A')$. As examples, we give two concrete AXU hash functions that satisfy the attack assumption and allow the attack to succeed.

(1) A UHF $H : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^n$ defined as $H_{K'}(X) = K'X$. We first show that $H_{K'}(X) = K'X$ indeed satisfies AXU. For any $X \neq X'$, multiplication by $X \oplus X' \neq 0$ is a bijection, so $K'(X \oplus X')$ is uniformly distributed and the function is 2^{-n}-AXU. We then show that it satisfies the attack assumption. Let $A = A' = 0$. The mask difference $\Delta = b_2 \cdot K' \cdot 0 \oplus b_3 \cdot K' \cdot 0 = 0$ is then publicly known. Hence it admits a two-encryption-query distinguisher against SPRTOP security with advantage $1 - 2^{-2n}$. This attack holds for every value of the hash key, including when the primitive is replaced by a truly random function.

(2) A UHF $H : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^n$ defined as $H_{K'}(X) = K'X \oplus v$ where $v \in \{0,1\}^n$ is a fixed public constant. This family is also 2^{-n}-AXU, since the fixed constant v cancels in the hash difference. We show that it satisfies the attack assumption. Fix any $A \in \{0,1\}^n$ and let $A' = \frac{b_2}{b_3} A$. The mask difference is then

$$\Delta = b_2 \cdot (K'A \oplus v) \oplus b_3 \cdot (K'A' \oplus v) = b_2 \cdot (K'A \oplus v) \oplus b_3 \cdot (K' \cdot \frac{b_2}{b_3} A \oplus v) = (b_2 \oplus b_3)v,$$

which is publicly computable. Hence it admits a two-encryption-query distinguisher against SPRTOP security with advantage $1 - 2^{-2n}$. This attack holds for every value of the hash key.

Flaws in the original OleF and t-OleF security proofs. Both security proofs derive their distinguishing advantage bounds by estimating the probabilities of bad events, including collisions of the internal variable X. However, neither proof adequately accounts for collisions between continuation and termination processing.

In the OleF proof [7], the badB Type 1 analysis in Appendix A.2 overlooks the case where the same first diblock is processed as a nonterminal diblock in one query and as a terminal diblock in another. Since both computations start from the zero initial state, their internal variables X are identical. Thus, the collision occurs with probability 1, rather than at most 2^{-n} as claimed in the original proof.

In the t-OleF proof [5], the badB13 analysis in Appendix A considers collisions of X in the first diblocks of two queries using hash terms with the same coefficient b_2. However, continuation and termination processing involve different coefficients b_2 and b_3, and ordinary AXU does not guarantee the required probability bound for $b_2 H_{K'}(A) \oplus b_3 H_{K'}(A')$. As shown by our two AXU

hash families, this difference can be publicly computed and compensated for in the plaintext using lemma 1, forcing $X = X'$ with probability 1. The same collision occurs in OleF, where the mask difference is 0. Thus, the original collision analyses fail to cover these cases, invalidating the claimed DOSPRP and SPRTOP security bounds in eqs. (2) and (3) under the stated assumptions.

4 Attacks on OlÆF

The previous section showed that t-OleF is insecure for certain hash function instances satisfying AXU. However, these attacks do not immediately imply attacks on OlÆF, since its encipher layer operates only on valid encodings. In this section, we show that the inputs required by the attack on t-OleF can be constructed as valid encodings of OlÆF, thereby extending the attack to both its privacy and integrity.

4.1 Two-Query Attack on the Privacy of OlÆF

Building on the attack on t-OleF in section 3.2, we present the following attack on the privacy of OlÆF. Let $P_A(B) = \text{CipherEnc}(\text{t-OleF}, A, B)$, and let A, A', Δ be the values involved in the attack of section 3.2.

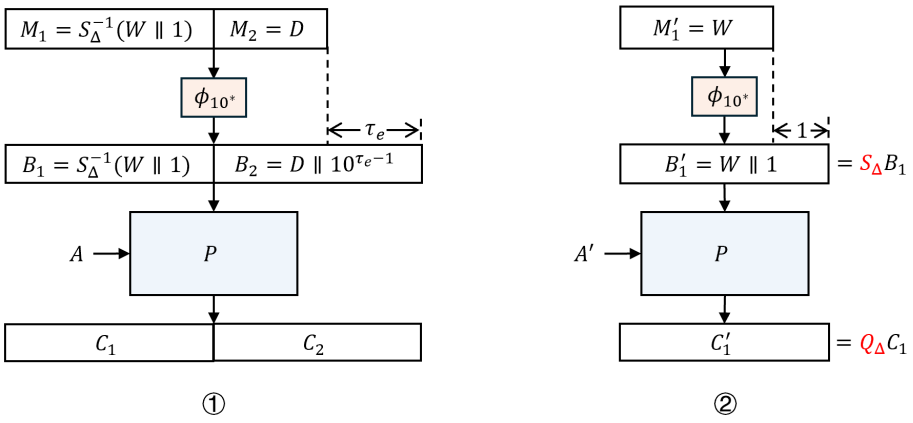

Fig. 7: Two encryption queries to OlÆF.

(1) Determine the messages to be queried as follows. Fix arbitrary $W \in \{0,1\}^{2n-1}, D \in \{0,1\}^{2n-\tau_e}$ where $\tau_e := \max\{1, 4n - m_{\max}\}$ and let one query message be $M_1' = W$, whose encoding is one-diblock $B_1' = \phi_{10^*}(1, M_1') = W \| 1$. Let $B_1 \| B_2 = S_\Delta^{-1}(W \| 1) \| (D \| 10^{\tau_e - 1})$ be another two-diblock encoded value satisfying $B_1' = S_\Delta B_1$. Then the corresponding pre-encoding message $M_1 \| M_2 = S_\Delta^{-1}(W \| 1) \| D$ satisfies $\phi_{10^*}(\tau_e, M_1 \| M_2) = B_1 \| B_2$. The parameter τ_e is chosen to satisfy the expansion length and message length requirements of OlÆF and to ensure that $|M_2|$ is nonnegative for any $\ell_{\max} \geq 2$ and $1 \leq t \leq 2n(\ell_{\max} - 1)$, as proved in section A.

(2) Query the encryption oracle on the associated data A, expansion length τ_e and the message $M_1 \| M_2$, and obtain the corresponding ciphertext $C_1 \| C_2$.

(3) Query the encryption oracle on the associated data A', expansion length $\tau' = 1$ and the message M'_1, and obtain the corresponding ciphertext C'_1.
(4) Check whether $C'_1 = Q_\Delta C_1$ holds.

The above attack constructs two suitable messages whose values after the encoding layer are exactly the two inputs to the attack on the encipher layer t-OleF in section 3.2, so the attack of section 3.2 applies directly. Hence this attack on privacy also achieves a distinguishing advantage as high as $1 - 2^{-2n}$. In the real world, the two encryption queries to OlÆF in this attack are illustrated in fig. 7.

4.2 Two-Query Attack on the Integrity of OlÆF

Building on the attack on the privacy of OlÆF in section 4.1, this section presents an attack on the integrity of OlÆF. The original OlÆF paper [5] adopts suffix-only verification as described in section 2.4, which is easier to pass than full-domain verification. For completeness, we consider integrity attacks in both cases, depending on whether the verification function uses full-domain verification or suffix-only verification.

Attack under full-domain verification. Let $P_A(B) = \text{CipherEnc}(\text{t-OleF}, A, B)$, and let $A, A', \Delta, W, D, \tau_e, M_1, M_2, B_1, B_2$ be the values involved in the attack of section 4.1. The encryption query and the successful forgery against OlÆF with target expansion $\tau^* = 1$ in this attack are illustrated in fig. 8.

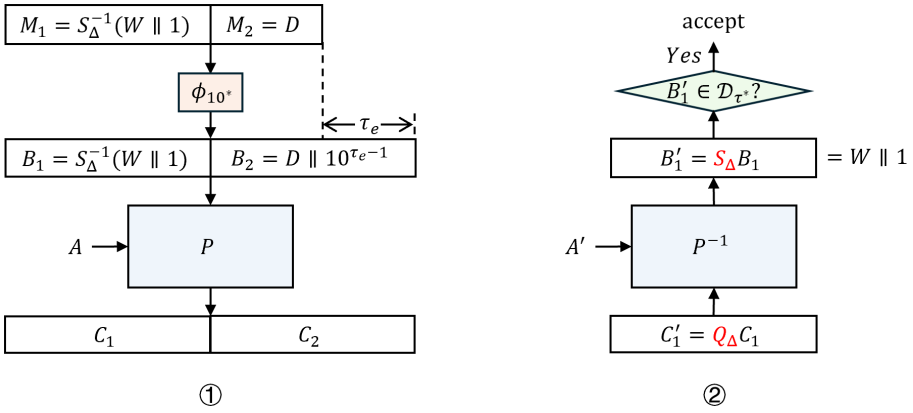

Fig. 8: The encryption query and the successful forgery against OlÆF.

(1) Make an encryption query identical to the first encryption query in the attack of section 4.1. That is, query the encryption oracle on $(A, \tau_e, M_1 \parallel M_2)$, and obtain the corresponding ciphertext $C_1 \parallel C_2$.
(2) Forge $(A', \tau^* = 1, C'_1 = Q_\Delta C_1)$.

We show that (A', τ^*, C'_1) is a successful forgery. By the relation between the two plaintext-ciphertext pairs in the attack of section 4.1, we have

$$B'_1 = P_{A'}^{-1}(C'_1) = P_{A'}^{-1}(Q_\Delta C_1) = S_\Delta(\mathsf{msb}_{2n}(P_A^{-1}(C_1 \parallel C_2))) = S_\Delta B_1 = W \parallel 1,$$

which can be generated from the message $M'_1 = W$ by the encoding function ϕ_{10^*}. Besides, the short decoded message length satisfies $|B'_1| - \tau^* = 2n - 1 = m_{\min}$. Hence $\mathrm{Ver}(A', \tau^*, C'_1) = \mathsf{accept}$, which implies that (A', τ^*, C'_1) is a successful forgery. Hence, this forgery attack against integrity succeeds with probability 1.

Attack under suffix-only verification. Clearly, if the verification function for the forgery in the above attack is changed to suffix-only verification, $\mathrm{Ver}_{\mathrm{suf}}(A', \tau^*, C'_1) = \mathsf{accept}$ still holds. Consider a slight adjustment to the above attack that changes the target expansion from $\tau^* = 1$ to $1 \le \tau^* \le \min\{t, 2n\}$. For $1 < \tau^* \le \min\{t, 2n\}$, we adjust the attack by choosing a $2n$-bit string B'_1 satisfying $\mathsf{lsb}_{\tau^*}(B'_1) = 10^{\tau^*-1}$, setting $B_1 = S_\Delta^{-1}(B'_1)$, and constructing the encryption query as in Section 4.1 using this B_1. By Lemma 1, the forged ciphertext $C'_1 = Q_\Delta C_1$ decrypts to B'_1. Since $|B'_1| - \tau^* = 2n - \tau^* < m_{\min}$, full-domain verification rejects the forgery, whereas suffix-only verification accepts it. That is, the forgery against OlÆF is still deemed successful when the target expansion is enlarged.

5 Flaw in the RUP Integrity Bound for Variable-Expansion EtE OAE

The preceding section presented attacks against the concrete OlÆF construction. This section instead examines the generic variable-expansion EtE OAE construction used in the security analysis of OlÆF in [5]. We show that even when its underlying encipher is an ideal tweakable online permutation $\pi \xleftarrow{\$} \mathrm{TOPerm}$, the claimed RUP integrity bound $\frac{q'}{2^{\tau^*}}$ is incorrect, where q' denotes the number of forgery attempts at the target expansion length τ^*. We first present a simple counterexample using an unverified decryption query and then generalize it to multiple queries and more general parameters. We next show that the claimed bound can be violated even without unverified decryption queries, by using non-target verification queries before the forgery. Finally, we explain the underlying flaw in the original RUP integrity proof revealed by these counterexamples.

5.1 Counterexamples Using Unverified Decryption Queries

A Simple Counterexample We first give a simple counterexample showing that the RUP integrity bound in [5] is incorrect. In this attack, the adversary makes only one unverified decryption query before one forgery. In this example, the target expansion length is $\tau^* = 2$. For concreteness, fix $n \ge 2$, $\ell_{\max} = 3$, and $t = 2$, in which case $m_{\min} = 2n - 1$ and $m_{\max} = 6n - 2$. The attack proceeds as follows.

(1) Pick two distinct associated-data values A, A' and two arbitrary ciphertexts C, C' of $4n$ bits each.
(2) Query the decryption oracle on $(A, \tau = 1, C)$, and obtain the corresponding unverified message M.
(3) If $\mathsf{lsb}_1(M) = 1$, forge $(A, \tau^* = 2, C)$; else, forge $(A', \tau^* = 2, C')$.

According to [5], in the ideal RUP setting the integrity bound is determined solely by the number of forgery attempts and equals $\frac{1}{2^{\tau^*}} = \frac{1}{4}$. However, we show below that the forgery success probability of our attack exceeds $\frac{1}{4}$.

(1) The case $\mathsf{lsb}_1(M) = 1$ happens with probability $\frac{1}{2}$ by the randomness of π_A^{-1}. Let $b_2 \,\|\, b_1 \coloneqq \mathsf{lsb}_{\tau^*=2}(\pi_A^{-1}(C))$ where $b_i \in \{0, 1\}$. Since the forgery and the unverified decryption query

use the same inverse-permutation output $\pi_A^{-1}(C)$ and differ only in the expansion length in this case, the decryption query reveals that $b_2 = \mathsf{lsb}_1(M) = 1$. Even given b_2, the last bit b_1 is independent of it and takes a random value, so $\Pr[b_1 = 0 \mid b_2 = 1] = \frac{1}{2}$. Therefore, the probability that this case occurs and the forgery succeeds is $\Pr[(\mathsf{lsb}_1(M) = 1) \wedge (b_2 \parallel b_1 = 10)] = \frac{1}{2} \times \frac{1}{2} = \frac{1}{4}$.

(2) The other case, $\mathsf{lsb}_1(M) = 0$, occurs with probability $\frac{1}{2}$ by the randomness of π_A^{-1}. In this case, the forgery uses a different inverse permutation $\pi_{A'}^{-1}$ which is independent of that used in the unverified decryption query since $A' \neq A$. Let $b'_2 \parallel b'_1 \coloneqq \mathsf{lsb}_{\tau^*=2}(\pi_{A'}^{-1}(C'))$ where $b'_i \in \{0,1\}$. Therefore, $\Pr[b'_2 \parallel b'_1 = 10 \mid \mathsf{lsb}_1(M) = 0] = \frac{1}{4}$ and the joint probability of this case and a successful forgery is $\Pr[(\mathsf{lsb}_1(M) = 0) \wedge (b'_2 \parallel b'_1 = 10)] = \frac{1}{2} \times \frac{1}{4} = \frac{1}{8}$.

In conclusion, the forgery succeeds with probability $\frac{1}{4} + \frac{1}{8} = \frac{3}{8}$, which exceeds the bound $\frac{1}{4}$ proved in [5].

A Generalized Counterexample We generalize the above counterexample as follows. Let the target expansion length be $1 < \tau^* \leq \min\{t, 2n\}$ and let $1 \leq \tau < \tau^*$ be another expansion length. Let $\ell \geq 1$ be the number of unverified decryption queries.

(1) Pick $\ell + 1$ distinct associated-datas $A_1, \ldots, A_\ell, A_{\ell+1}$ and $\ell + 1$ arbitrary complete-diblock ciphertexts $C_1, \ldots, C_\ell, C_{\ell+1}$ of a common length s, provided that such a length s exists and satisfies $m_{\min} \leq s - \tau^* < s - \tau \leq m_{\max}$.

(2) Query the decryption oracle on $(A_1, \tau, C_1), \ldots, (A_\ell, \tau, C_\ell)$ and obtain the corresponding unverified messages $M_1, \ldots, M_\ell$.

(3) If at least one of $\mathsf{lsb}_{\tau^*-\tau}(M_i) = 10^{\tau^*-\tau-1}$ for $i \in \{1, \ldots, \ell\}$ holds, let j be the smallest such index and forge (A_j, τ^*, C_j). Otherwise, forge $(A_{\ell+1}, \tau^*, C_{\ell+1})$.

According to [5], in the ideal RUP setting the integrity bound is $\frac{1}{2^{\tau^*}}$. We now compute the forgery success probability of this attack.

(1) Let E denote the event that there exists an index $i \in \{1, \ldots, \ell\}$ such that $\mathsf{lsb}_{\tau^*-\tau}(M_i) = 10^{\tau^*-\tau-1}$. The event E occurs with probability $1 - \left(1 - \frac{1}{2^{\tau^*-\tau}}\right)^\ell$. Let $b_{j,\tau^*} \parallel \cdots \parallel b_{j,1} \coloneqq \mathsf{lsb}_{\tau^*}(\pi_{A_j}^{-1}(C_j))$ where $b_{j,i} \in \{0,1\}$. The occurrence of event E implies $b_{j,\tau^*} \parallel \cdots \parallel b_{j,\tau+1} = 10^{\tau^*-\tau-1}$. Since j is determined by the revealed bits, the remaining τ bits are still independent and uniformly random. Hence,

$$\Pr\left[b_{j,\tau} \parallel \cdots \parallel b_{j,1} = 0^\tau \mid b_{j,\tau^*} \parallel \cdots \parallel b_{j,\tau+1} = 10^{\tau^*-\tau-1}\right] = \frac{1}{2^\tau}.$$

Hence, the probability that event E occurs and the forgery succeeds is

$$p_1 \coloneqq \Pr[E \wedge (b_{j,\tau^*} \parallel \cdots \parallel b_{j,1} = 10^{\tau^*-1})] = \left(1 - \left(1 - \frac{1}{2^{\tau^*-\tau}}\right)^\ell\right) \times \frac{1}{2^\tau}.$$

(2) The event $\bar{E}$ occurs with probability $\left(1 - \frac{1}{2^{\tau^*-\tau}}\right)^\ell$. Let $b_{\ell+1,\tau^*} \parallel \cdots \parallel b_{\ell+1,1} \coloneqq \mathsf{lsb}_{\tau^*}(\pi_{A_{\ell+1}}^{-1}(C_{\ell+1}))$ where $b_{\ell+1,i} \in \{0,1\}$. The probability that $\bar{E}$ occurs and the forgery succeeds is

$$p_2 \coloneqq \Pr[\bar{E} \wedge (b_{\ell+1,\tau^*} \parallel \cdots \parallel b_{\ell+1,1} = 10^{\tau^*-1})] = \left(1 - \frac{1}{2^{\tau^*-\tau}}\right)^\ell \times \frac{1}{2^{\tau^*}}.$$

In conclusion, the forgery succeeds with probability $p_{\tau,\tau^*,\ell} \coloneqq p_1 + p_2 = \frac{1}{2^\tau}\left(1 - \left(1 - \frac{1}{2^{\tau^*-\tau}}\right)^{\ell+1}\right)$, which exceeds the ideal RUP integrity bound $\frac{1}{2^{\tau^*}}$ claimed in [5].

5.2 Counterexamples Using Non-Target Verification Queries

A Simple Counterexample We now give another simple counterexample that requires no unverified decryption queries and uses only one non-target verification query and one forgery attempt. Using the same parameters and the same choices of A, A', C, C' as the simple counterexample in Section 5.1, we modify Steps (2) and (3) as follows.

(2) Query the verification oracle on $(A, \tau = 1, C)$ and obtain the verification result.
(3) If the verification result is reject, forge $(A, \tau^* = 2, C)$; otherwise, forge $(A', \tau^* = 2, C')$.

Let $b_2 \parallel b_1 \coloneqq \mathsf{lsb}_2(\pi_A^{-1}(C))$, where $b_i \in \{0, 1\}$. The verification query returns reject with probability $\frac{1}{2}$, which reveals that $b_1 = 0$. Since b_2 is independent of b_1 and uniformly random, the forgery $(A, \tau^* = 2, C)$ succeeds with conditional probability $\frac{1}{2}$. Otherwise, the verification query returns accept with probability $\frac{1}{2}$. Since $A' \neq A$, the forgery $(A', \tau^* = 2, C')$ succeeds with conditional probability $\frac{1}{4}$. Therefore, the overall forgery success probability is $\frac{1}{2} \times \frac{1}{2} + \frac{1}{2} \times \frac{1}{4} = \frac{3}{8}$, which exceeds the claimed bound $\frac{1}{4}$.

A Generalized Counterexample We now generalize the simple verification-based counterexample. Let the target expansion length be $1 \leq \tau^* \leq \min\{t, 2n\}$ and let $1 \leq \tau \leq t$ with $\tau \neq \tau^*$ be another expansion length. Let $\ell \geq 1$ be the number of non-target verification queries. As in Step (1) of the preceding generalized counterexample in Section 5.1, pick $\ell + 1$ distinct associated-datas $A_1, \ldots, A_{\ell+1}$ and $\ell + 1$ arbitrary complete-diblock ciphertexts $C_1, \ldots, C_{\ell+1}$ of a common length s, except that s must satisfy $m_{\min} \leq s - \max\{\tau, \tau^*\} < s - \min\{\tau, \tau^*\} \leq m_{\max}$. We then modify Steps (2) and (3) as follows.

(2) Query the verification oracle on $(A_1, \tau, C_1), \ldots, (A_\ell, \tau, C_\ell)$ and obtain the corresponding verification results.
(3) If at least one verification query returns reject, let j be the smallest such index and forge (A_j, τ^*, C_j). Otherwise, forge $(A_{\ell+1}, \tau^*, C_{\ell+1})$.

Let E' denote the event that at least one of the ℓ verification queries returns reject, i.e., $\mathsf{lsb}_\tau(\pi_{A_i}^{-1}(C_i)) \neq 10^{\tau-1}$ for some i. The forgery success probability is analyzed as follows.

(1) The event E' occurs with probability

$$\Pr[E'] = 1 - \left(\frac{1}{2^\tau}\right)^\ell$$

by the independence of the inverse permutations. Since the verification query on (A_j, τ, C_j) returns reject, we know that $\mathsf{lsb}_\tau(\pi_{A_j}^{-1}(C_j)) \neq 10^{\tau-1}$. Since $\tau \neq \tau^*$, the suffix conditions for the two expansion lengths $\mathsf{lsb}_\tau(B) = 10^{\tau-1}$ and $\mathsf{lsb}_{\tau^*}(B) = 10^{\tau^*-1}$ cannot hold simultaneously for any B. Therefore, given that the verification query returns reject, the probability that the same ciphertext passes the target verification is

$$\Pr\left[\mathsf{lsb}_{\tau^*}(\pi_{A_j}^{-1}(C_j)) = 10^{\tau^*-1} \;\middle|\; \mathsf{lsb}_\tau(\pi_{A_j}^{-1}(C_j)) \neq 10^{\tau-1}\right] = \frac{2^\tau}{2^{\tau^*}(2^\tau - 1)}$$

whether $\tau < \tau^*$ or $\tau > \tau^*$. Since the inverse permutations corresponding to distinct associated-datas are independent, selecting the first rejected candidate does not change the above conditional probability. Therefore, this case contributes the following probability to the overall forgery success:

$$p_1' \coloneqq \Pr\left[E' \wedge \left(\mathsf{lsb}_{\tau^*}(\pi_{A_j}^{-1}(C_j)) = 10^{\tau^*-1}\right)\right] = \frac{2^{\ell\tau} - 1}{2^{\tau^* + (\ell-1)\tau}(2^\tau - 1)}.$$

(2) When E' does not occur, the adversary forges $(A_{\ell+1}, \tau^*, C_{\ell+1})$. Since $A_{\ell+1}$ is distinct from all previously queried associated-datas, this forgery succeeds with conditional probability $\frac{1}{2^{\tau^*}}$. Hence, the probability that $\bar{E}'$ occurs and the forgery succeeds is

$$p'_2 \coloneqq \Pr\left[\bar{E}' \wedge \left(\mathsf{lsb}_{\tau^*}(\pi^{-1}_{A_{\ell+1}}(C_{\ell+1})) = 10^{\tau^*-1}\right)\right] = \left(\frac{1}{2^{\tau}}\right)^{\ell} \times \frac{1}{2^{\tau^*}} = \frac{1}{2^{\tau^*+\ell\tau}}.$$

Therefore, the overall forgery success probability is

$$p'_{\tau,\tau^*,\ell} \coloneqq p'_1 + p'_2 = \frac{2^{(\ell+1)\tau} - 1}{2^{\tau^*+\ell\tau}(2^{\tau} - 1)}.$$

For every $\ell \geq 1$, this probability strictly exceeds $\frac{1}{2^{\tau^*}}$. Thus, the claimed integrity bound can be violated without any unverified decryption queries.

5.3 Flaw in the Original RUP Integrity Proof

According to [5], the integrity bound in this ideal RUP setting is $\frac{q'}{2^{\tau^*}}$, where q' is the number of forgery attempts at the target expansion length τ^*. This bound can be understood as follows. If the target expansion is τ^*, a valid encoding must end with 10^{τ^*-1}, so an attacker appears to need to guess τ^* bits, and a single verification succeeds with probability $2^{-\tau^*}$. With q' target-expansion verifications in total, this yields the bound $\frac{q'}{2^{\tau^*}}$. The intuition behind this bound is that the last τ^* bits derived from each forgery attempt are uniformly random and independent of the previous query transcript.

The counterexamples in sections 5.1 and 5.2 show that the claimed bound can be violated if the adversary has previously made unverified decryption queries at a smaller expansion length or verification queries at a different expansion length. Both types of attacks exploit the fact that the adversary can reuse a previously queried associated-data-ciphertext pair (A, C) in its forgery with a probability determined by the preceding query results. Whenever this happens, the preceding query and the forgery necessarily use the same inverse-permutation output $\pi_A^{-1}(C)$ despite their different expansion lengths. The unverified decryption queries reveal part of the target authentication suffix, while the non-target verification queries reveal information about the suffix through their verification results. Consequently, the target authentication suffix is not necessarily uniformly random given the previous query transcript, and the forgery success probability can exceed the claimed bound.

6 Repair Requirements

This section presents some repair requirements to address the attacks described in the previous sections.

Requiring coefficient-AXU for UHF. The core of the distinguishing attack on t-OleF in section 3.2 is the existence of two associated data values A and A' such that $b_2 H_{K'}(A) \oplus b_3 H_{K'}(A') = \Delta$ holds with probability 1, where Δ is a public value. Hence the AXU property is insufficient for a universal hash function H with n-bit output. Therefore, to prevent this attack while preserving the original masking structure, we impose the stronger coefficient almost XOR

universal (CAXU) property on H as our first repair requirement: for any two distinct input pairs (X, c) and (X', c'), where $c, c' \in b_2, b_3$ are nonzero, and any $\Delta \in \{0,1\}^n$,

$$\Pr_{K'}[cH_{K'}(X) \oplus c'H_{K'}(X') = \Delta] \leq \epsilon.$$

This property is stronger than AXU, which covers only $X \neq X'$ with $c = c'$. It additionally covers $X = X'$ with $c \neq c'$ and $X \neq X'$ with $c \neq c'$, both outside the scope of AXU.

Here we present an instantiation of a UHF satisfying CAXU with $\epsilon = \frac{1}{2^n}$, namely $H_{K_1,K_2}(X) = K_1X \oplus K_2$, where K_1, K_2 are independently and uniformly sampled from $\{0,1\}^n$ and kept secret. Let $\alpha \coloneqq cX \oplus c'X'$ and $\beta = c \oplus c'$. Then

$$\Pr_{K_1,K_2}[cH_{K_1,K_2}(X) \oplus c'H_{K_1,K_2}(X') = \Delta] = \Pr_{K_1,K_2}[\alpha K_1 \oplus \beta K_2 = \Delta]. \tag{10}$$

When $c \neq c'$, we have $\beta \neq 0$, so (10) $= \Pr_{K_1,K_2}[K_2 = \beta^{-1}(\Delta \oplus \alpha K_1)] = \frac{1}{2^n}$. For the remaining case $c = c'$, the condition $(X, c) \neq (X', c')$ leads to $X \neq X'$. Then $\beta = 0$ and $\alpha = c(X \oplus X')$. Since $c \neq 0$ and $X \oplus X' \neq 0$, we have $\alpha \neq 0$, and hence (10) $= \Pr_{K_1,K_2}[K_1 = \alpha^{-1}\Delta] = \frac{1}{2^n}$. In summary, this H satisfies CAXU with $\epsilon = \frac{1}{2^n}$.

Adding complete domain checks. Section 4.2 shows that if the verification function is suffix-only acceptance, the integrity forgery attack on OlÆF still works for target expansion length $\tau^* > 1$. Under full-domain verification, however, the same attack fails in this case. Therefore, the second repair requirement is to change the verification function of OlÆF to full-domain verification.

Note that this repair does not prevent all attacks on the privacy and integrity of OlÆF presented in section 4. In particular, full-domain verification cannot prevent the integrity forgery at target expansion $\tau^* = 1$, since the forged ciphertext decrypts to a valid message of length at least $2n - 1$. It also cannot prevent the privacy distinguishing attack in section 4.1, which does not involve verification. Moreover, full-domain verification cannot prevent the cross-expansion counterexamples presented in section 5, since the corresponding queries and forgeries satisfy the required message-length constraints.

Incorporating the expansion parameter τ into the tweak. The flaw in the RUP integrity bound for the generic variable-expansion EtE OAE construction, discussed in section 5, arises because the enciphering permutation P_A depends on A but not on τ. Hence even for $\tau \neq \tau'$, both $\mathrm{Dec}(A, \tau, C)$ and $\mathrm{Ver}(A, \tau', C)$ act through the same inverse $P_A^{-1}(C)$, which causes cross-expansion leakage. Therefore the third repair requirement is to treat (A, τ) as a whole as the tweak of the permutation P, that is, to replace P_A with $P_{A,\tau}$. This requires an injective encoding of (A, τ) into the tweak space, and any hash function used to process the tweak must support the enlarged input domain. In this way, as long as $\tau \neq \tau'$, the inverses $P_{A,\tau}^{-1}$ and $P_{A,\tau'}^{-1}$ behave as two independent random online permutations in the ideal setting, which prevents the cross-expansion information reuse exploited by the counterexamples in section 5.

Although repair requirements have been given, this paper does not prove that the repaired t-OleF or OlÆF is fully secure. A complete security proof for the repaired constructions remains an open problem.

7 Conclusions

This paper revisits the security of OleF, t-OleF, and OlÆF and identifies two main independent security flaws. First, using different masks for continuation and termination processing does not necessarily prevent exploitable algebraic relations between them. We exploit such a relation to construct structural attacks on OleF and t-OleF, and further extend these attacks to the privacy and RUP integrity of OlÆF. Second, in the generic variable-expansion EtE OAE construction, the enciphering layer depends only on the associated data and not on the expansion length. Thus, prior unverified decryption queries or non-target verification queries at different expansion lengths can reveal information about the target authentication suffix through the same inverse-permutation output. We construct counterexamples showing that the claimed RUP integrity bound can be violated even when the underlying encipher is an ideal tweakable online permutation. We also identify a mismatch between the full-domain verification used in the generic EtE security analysis and the suffix-only verification specified for OlÆF. To address these issues, we propose repair requirements. Whether these measures jointly suffice to guarantee privacy and integrity under RUP remains an open problem.

References

1. Andreeva, E., Barwell, G., Bhaumik, R., Nandi, M., Page, D., Stam, M.: Turning online ciphers off. IACR Transactions on Symmetric Cryptology **2017**(2), 105–142 (2017). https://doi.org/10.13154/tosc.v2017.i2.105-142
2. Andreeva, E., Bogdanov, A., Luykx, A., Mennink, B., Mouha, N., Yasuda, K.: How to securely release unverified plaintext in authenticated encryption. In: Advances in Cryptology—ASIACRYPT 2014, Part I. Lecture Notes in Computer Science, vol. 8873, pp. 105–125. Springer (2014). https://doi.org/10.1007/978-3-662-45611-8_6
3. Bellare, M., Boldyreva, A., Knudsen, L., Namprempre, C.: Online ciphers and the Hash-CBC construction. In: Advances in Cryptology—CRYPTO 2001. Lecture Notes in Computer Science, vol. 2139, pp. 292–309. Springer (2001). https://doi.org/10.1007/3-540-44647-8_18
4. Bellare, M., Rogaway, P.: Encode-then-encipher encryption: How to exploit nonces or redundancy in plaintexts for efficient cryptography. In: Advances in Cryptology—ASIACRYPT 2000. Lecture Notes in Computer Science, vol. 1976, pp. 317–330. Springer (2000). https://doi.org/10.1007/3-540-44448-3_24
5. Bhattacharjee, A., Bhaumik, R., Collins, D., Nandi, M.: Provably secure online authenticated encryption and bidirectional online channels. In: Selected Areas in Cryptography—SAC 2024, Part II. Lecture Notes in Computer Science, vol. 15517, pp. 56–82. Springer (2025). https://doi.org/10.1007/978-3-031-82841-6_3
6. Bhaumik, R., Nandi, M.: OleF: An inverse-free online cipher. Cryptology ePrint Archive, Paper 2016/1090 (2016), https://eprint.iacr.org/2016/1090
7. Bhaumik, R., Nandi, M.: OleF: an inverse-free online cipher. an online SPRP with an optimal inverse-free construction. IACR Transactions on Symmetric Cryptology **2016**(2), 30–51 (2017). https://doi.org/10.46586/tosc.v2016.i2.30-51
8. Fleischmann, E., Forler, C., Lucks, S.: McOE: A family of almost foolproof on-line authenticated encryption schemes. In: Fast Software Encryption—FSE 2012. Lecture Notes in Computer Science, vol. 7549, pp. 196–215. Springer (2012). https://doi.org/10.1007/978-3-642-34047-5_12
9. Guo, T., Wang, P., Jing, J., Mao, S., Liu, G.: Classical and quantum full plaintext recovery for low-round Feistel-type designs. In: Advances in Cryptology—CRYPTO 2026, Part V. Lecture Notes in Computer Science, vol. 16804, pp. 468–498. Springer (2026). https://doi.org/10.1007/978-3

-032-35409-9_16, https://eprint.iacr.org/2026/1040, full version: ePrint 2026/1040, revision of 8 June 2026
10. Hoang, V.T., Reyhanitabar, R., Rogaway, P., Vizár, D.: Online authenticated-encryption and its nonce-reuse misuse-resistance. In: Advances in Cryptology—CRYPTO 2015, Part I. Lecture Notes in Computer Science, vol. 9215, pp. 493–517. Springer (2015). https://doi.org/10.1007/978-3-662-47989-6_24, https://eprint.iacr.org/2015/189
11. Jha, A., Nandi, M.: On rate-1 and beyond-the-birthday bound secure online ciphers using tweakable block ciphers. Cryptography and Communications **10**(5), 731–753 (2018). https://doi.org/10.1007/s12095-017-0275-0
12. Nandi, M.: Forging attacks on two authenticated encryption schemes COBRA and POET. In: Advances in Cryptology – ASIACRYPT 2014. Lecture Notes in Computer Science, vol. 8873, pp. 126–140. Springer (2014). https://doi.org/10.1007/978-3-662-45611-8_7
13. Reyhanitabar, R., Vaudenay, S., Vizár, D.: Authenticated encryption with variable stretch. In: Advances in Cryptology – ASIACRYPT 2016. pp. 396–425. Springer (2016). https://doi.org/10.1007/978-3-662-53887-6_15
14. Zhang, J., Wu, W.L.: Security of online AE schemes in RUP setting. In: Cryptology and Network Security—CANS 2016. Lecture Notes in Computer Science, vol. 10052, pp. 319–334. Springer (2016). https://doi.org/10.1007/978-3-319-48965-0_19

A Validity of the Attack Parameter τ_e

The first message $M_1 \parallel M_2$ in this attack has length $4n - \tau_e$. By the message length requirement for OlÆF described in Section 2.4, it must satisfy

$$2n - 1 \leq 4n - \tau_e \leq m_{\max}, \tag{11}$$

with expansion length

$$1 \leq \tau_e \leq t. \tag{12}$$

Hence, combining eqs. (11) and (12), we obtain the following equivalent condition: $\max\{1, 4n - m_{\max}\} \leq \tau_e \leq \min\{t, 2n + 1\}$. We choose $\tau_e = \max\{1, 4n - m_{\max}\}$, which satisfies the lower bound. We next show that it also satisfies the upper bound.

(1) Proof of $\tau_e \leq t$: OlÆF already requires

$$m_{\max} = 2n\ell_{\max} - t, \tag{13}$$
$$1 \leq t \leq 2n(\ell_{\max} - 1) \tag{14}$$

with

$$\ell_{\max} \geq 2, \tag{15}$$

as described in Section 2.4. Eqs. (13) and (15) give $4n - m_{\max} \leq t$. and eq. (14) gives $1 \leq t$. Therefore $\max\{1, 4n - m_{\max}\} \leq t$, that is, $\tau_e \leq t$.

(2) Proof of $\tau_e \leq \min\{t, 2n+1\}$: From eq. (13), we obtain $4n - m_{\max} = t - 2n(\ell_{\max} - 2)$. Combining this with eq. (14) further yields $4n - m_{\max} \leq 2n$. Therefore $\tau_e = \max\{1, 4n - m_{\max}\} \leq 2n$, and hence $\tau_e \leq \min\{t, 2n + 1\}$ necessarily holds.

Furthermore, the established bound $\tau_e \leq 2n$ ensures that the length of M_2, given by $2n - \tau_e$, is nonnegative.