Linking Online Misuse-Resistant Authenticated Encryption and Blockwise Attack Models

Guillaume Endignoux and Damian Vizár
EPFL, Switzerland, \{guillaume.endignoux,damian.vizar\}@epfl.ch

Abstract. Real-world applications of authenticated encryption often require the encryption to be computable online, e.g. to compute the $i^{th}$ block of ciphertext after having processed the first $i$ blocks of plaintext. A significant line of research was dedicated to identifying security notions for online authenticated encryption schemes, that capture various security goals related to real-life scenarios. Fouque, Joux, Martinet and Valette proposed definitions of privacy and integrity against adversaries that can query their oracles in a blockwise-adaptive manner, to model memory-constrained applications. A decade later, Fleischmann, Forler and Lucks proposed the notion of online nonce misuse-resistant authenticated encryption (OAE) to capture the security of online authenticated encryption under nonce-reuse. In this work we investigate the relation between these notions. We first recast the blockwise notions of Fouque et al. to make them compatible with online authenticated encryption schemes that support headers. We then show that OAE and the conjunction of the blockwise notions are “almost” equivalent. We identify the missing property on the side of blockwise notions, and formalize it under the name PR-TAG. With PR-TAG being just an auxiliary definition, the equivalence we finally show suggests that OAE and the blockwise model for online authenticated encryption capture essentially the same notion of security.

Keywords: Symmetric-key Cryptography · Authenticated Encryption · Online Encryption · Security Notions

1 Introduction

Authenticated encryption (AE) is a symmetric-key cryptographic primitive that provides confidentiality (privacy\(^1\)) and integrity (together with authenticity) protection of processed data. After its initial recognition and formalization \[\text{[BN00, BR00, KY01]}\], AE became a popular research target. In particular, a significant amount of effort has been invested in the research of security goals for AE, resulting in a number of security notions, e.g. the nonce-based AE with associated data (AEAD) \[\text{[Rog02]}\], notions capturing security against blockwise attacks \[\text{[FJMV03, FJP04]}\], the nonce misuse-resistant AE (MRAE) \[\text{[RS06]}\], online nonce misuse-resistant AE (OAE) \[\text{[FFL12]}\], robust AE (RAE) \[\text{[HKR15]}\], online AE (OAE2) \[\text{[HRRV15]}\], or security notions for streaming channels \[\text{[FGMP15]}\]. These notions capture security of AE in the context of diverse usage scenarios and adversarial powers. The recent CAESAR competition \[\text{[Ber]}\] has been both an evidence of the popularity of AE and a catalyst for new research activity.

Online Computable AE. A majority of existing AE schemes internally parse the plaintext into smaller, fixed-size blocks during encryption, and likewise produce the ciphertext as a sequence of such blocks. In many of these schemes (including GCM \[\text{[MV04]}\] and many

\(^1\)In this paper, we use the word privacy interchangeably with confidentiality.
of the CAESAR candidates, e.g. \([\text{ABL}^+, \text{IMG}^+, \text{BDP}^+, \text{KR}]\), the encryption algorithm additionally computes the ciphertext blocks in an online manner, i.e. the \(i^{th}\) block of ciphertext can be computed and written immediately after the first \(i\) blocks of plaintext were processed. We call AE schemes with this property online AE schemes.

The onlineness of encryption is necessary in constrained applications, where it is of importance to compute the ciphertext blocks with constant latency (e.g. streaming), or where a constant memory implementation is required. Online encryption algorithms are also frequent in AE schemes targeted at high performance.

**Blockwise-Adaptive Attack Models.** While onlineness is useful as a practical feature, it can impact security. Joux, Martinet and Valette observed that if an application, such as a smart card, outputs a ciphertext block each time it is fed a plaintext block [JMV02]. They introduced the blockwise-adaptive adversaries and exhibited efficient blockwise-adaptive attacks on real-world schemes which were originally proven secure in models that consider the plaintexts/ciphertexts to be atomic.

A year later, Fouque, Joux, Martinet and Valette (FJMV) proposed security notions for privacy (dubbed \text{IND-BCPA} and \text{IND-BCCA}) and integrity (dubbed \text{B-INT-CTXT}) of randomized AE schemes against blockwise-adaptive adversaries [FJMV03]. Their privacy notions were defined using the left-or-right style of indistinguishability for symmetric key encryption [BDJR97]. Later on, Fouque, Joux and Poupart extended the framework of blockwise security notions for privacy to model the type of adversarial access, distinguishing between the cases where the adversary can query an infinite stream of blocks in a single message, where it can encrypt several messages sequentially, and where it can encrypt several messages concurrently [FJP04]. In the same year, Boldyreva and Taesombut proposed a relaxed version of the original \text{IND-BCCA} notion [BT04]. In 2007, Bard proposed a new framework for studying security of online encryption schemes against blockwise-adaptive attacks [Bar07]. However, this framework treated privacy of encryption-only schemes, but not integrity.

**Online Authenticated Encryption.** The notion of AEAD is perhaps the most popular design target for AE. The popularity stems from its simplicity: it allows one to construct simple, efficient, online AE schemes with deterministic encryption algorithms, such that they only require a unique initialization vector (a.k.a. a nonce) to be used with every encryption to be secure. Even though this simple requirement appears to be rather achievable, Rogaway and Shrimpton pointed out that if violated, it will lead to a complete break of numerous existing AEAD schemes. They proposed the MRAE notion to mitigate the impact of nonce repetition [RS06]. An MRAE scheme will only reveal unavoidable information if nonces get repeated: the complete repetition of all inputs. However, an inherent property of any MRAE scheme is that it cannot be online, as every ciphertext bit must depend on every bit of plaintext.

Fleischmann, Forler and Lucks (FFL) sought to overcome this functional limitation by proposing a security notion for schemes that are online, yet still retain some (lower) level of resistance in case of nonce-misuse [FFL12]. Their OAE notion combines the usual definition for integrity of ciphertexts with an extended version of the notion of online ciphers [BBKN12]. Another particularity of OAE is that it conflates nonce and associated data (previously separate inputs of encryption and decryption algorithms) into a single input called “header”. The notion was quickly targeted by several designs [AFF+15, ABL+13]. FFL assert that the security guarantees of OAE under nonce misuse offer meaningful protection, and that these guarantees are the best possible in the given setting. Both these claims have been disputed, and especially the former has been a subject of controversy [HRRV15, AFL+17], leaving a question mark over the usefulness of online
Deflating the Notion-Space. In this paper, we investigate the relations between the blockwise-adaptive notions for online AE by FJMV and the notion of OAE by FFL. Our motivation is twofold; first, the controversial results about nonce misuse-resistance of any online AE invite to investigate other security properties captured by OAE, and second, we believe that it is of importance to reduce the redundancy among the numerous security notions for online AE, by determining the relations between these notions.

However, a direct comparison of the notions of FFL and FJMV is not possible. The notions by FJMV are defined for randomized AE schemes with no support for headers, while the OAE notion works with deterministic AE schemes that do support headers. We therefore recast the notions of FJMV into the setting of deterministic online AE schemes and define what we believe to be their most natural extension. We then compare these restated blockwise notions (dubbed B-INT-CTX and D-LORS-BCPA in this paper) with OAE. We first show a rather intuitive result: that OAE implies both B-INT-CTX and D-LORS-BCPA. We then show, by means of a counterexample, that the conjunction of the latter two notions does not imply OAE. We identify the property used for the counterexample as a minor problem related to the privacy of authentication tags, and formalize the missing property in an auxiliary notion dubbed PR-TAG. We finally show the equivalence between OAE and the conjunction of B-INT-CTX, D-LORS-BCPA and PR-TAG.

Related Work. Fouque, Joux and Poupard (FJP) show that in case of deterministic online ciphers (as defined in [BBKN12]), the blockwise-adaptive security notion and the conventional security notion are equivalent by a quadratic reduction [FJP04]. Our result resembles theirs in the complexity of the reduction, and the two results are closely related. However, their analysis deals only with (privacy of) plain online ciphers, while our work establishes relations between notions for authenticated encryption. In addition, FJP analyze the relation between standard and blockwise-adaptive versions of the same notion, while our analysis links two notions of different nature (left-or-right vs indistinguishability from a random primitive).

Our Contribution. The main result of our paper is the equivalence between OAE and the adapted version of the blockwise-adaptive AE notions of FJMV, extended with PR-TAG. Considering that PR-TAG captures a property of the authentication tag which is not related to adversarial blockwise adaptability, this equivalence points out that the security guarantees captured by the notion of OAE are essentially equivalent with those captured by the notion of FJMV.

Organization of the Paper. In Section 2, we give a few preliminary notations and results. In Section 3 we give the definition of OAE and the adapted definitions of B-INT-CTX and D-LORS-BCPA. In Section 4 we establish the relations between the notions.

2 Preliminaries

We define the notations and recall some concepts that are referred to in the paper. Throughout the paper, we denote by \( \{0,1\}^* \) the set of finite strings, including the empty string \( \varepsilon \). For some positive integer \( n \), we denote by \( B_n = \{0,1\}^n \) the set of \( n \)-bit blocks and by \( B_n^* \subset \{0,1\}^* \) the set of finite strings whose length is a multiple of \( n \). We often call such \( n \) the block length. Strings from \( B_n^* \) naturally split into blocks: given \( M \in B_n^* \), we denote by \( M[i] \in B_n \) the \( i \)th block of \( M \) (starting at index 1), and we let \( M[i...j] = M[i] || \ldots || M[j] \),
We define the following functions on $B^n$: the block count $\text{BlCount} : B^n \rightarrow \mathbb{N}$ as $\text{BlCount}(M) = p$ for all $M \in B^n$, the longest common prefix between two strings $\text{LCP}_n : B^n \times B^n \rightarrow B^n$ defined by $\text{LCP}_n(M_1, M_2) = M$ where $M_1 = M||M'_1$, $M_2 = M||M'_2$, and $M'_1$ and $M'_2$ have no common prefix (i.e. $M'_1 = \epsilon$ or $M'_2 = \epsilon$ or $M([1] \neq M([2])$), and the length of the longest common prefix $\text{LLCP}_n : B^n \times B^n \rightarrow \mathbb{N} = \text{BlCount} \circ \text{LCP}_n$.

Online permutations have been extensively studied in [BBKN12], we recall some relevant properties here. We denote by $\text{Perm}[n]$ the set of permutations of $B_n$. We denote by $\text{OPerm}[n]$ the set of online, length-preserving permutations of $B_n$, i.e. for all $\pi \in \text{OPerm}[n]$ we have that (online) for all $M_1, M_2 \in B_n$, the first $\text{LLCP}_n(M_1, M_2)$ blocks of $\pi(M_1)$ and $\pi(M_2)$ are identical, (length-preserving), for all $M \in B_n$, $\text{BlCount}(\pi(M)) = \text{BlCount}(M)$. We note that $\text{OPerm}[n]$ is stable by the composition operation $\circ$ and that $\text{LLCP}_n$ is stable by composition with an online permutation, i.e. $\forall \pi \in \text{OPerm}[n], \forall M_1, M_2 \in B_n, \text{LLCP}_n(\pi(M_1), \pi(M_2)) = \text{LLCP}_n(M_1, M_2)$.

Canonical tree representation. We denote by $\text{Tree}[n]$ the set of infinite $2^n$-ary trees, such that every tree $T \in \text{Tree}[n]$ verifies the following properties: each node $N$ of $T$ is labeled with a permutation $T[N] \in \text{Perm}[n]$, and the $2^n$ (outgoing) edges starting from a node $N$ are each labeled with a distinct $x \in B_n$.

We recall that there is a bijection $T \mapsto \pi_T$ between $\text{Tree}[n]$ and $\text{OPerm}[n]$. Given a tree $T \in \text{Tree}[n]$, its image $\pi_T$ can be obtained by traversing the tree. More precisely, for any $M \in B_n^*$, we evaluate $C = \pi_T(M)$ as follows. We first set $i = 1$ and the root of $T$ as the current node $N_1$, and then for each input block $M[i]$ we use the permutation $T[N_i]$ of the current node to compute $C[i] = T[N_i](M[i])$ and follow the edge labeled with $M[i]$ to move to the next node $N_{i+1}$ (Figure 1).

Given a permutation $\pi \in \text{OPerm}[n]$ and a message $M \in B_n$ with $l = \text{BlCount}(M)$, we denote by $\pi[M] \in \text{Perm}[n]$ the permutation defined by $\pi[M] : B \mapsto \pi(M||B)[l+1]$. In other words, if $T$ is the labeled tree associated to $\pi$, then $\pi[M]$ is the label of the node reached by following the edges $M[1], ..., M[l]$ starting from the root of $T$. For example, on Figure 1, we identify $\pi[\epsilon] = T[N_1], \pi[M[1]] = T[N_2]$ and $\pi[M[1..2]] = T[N_3]$.

We note that while $\text{OPerm}[n]$ is an infinite set, it can be sampled “uniformly” thanks to the canonical tree representation and lazy sampling. To sample a random $\pi \in \text{OPerm}[n]$, we start with an unlabeled tree, and as we walk through the tree according to the incoming queries, we lazily sample the labeling permutations that are needed. With the example of Figure 1, computing $\pi_T(M[1..2])$ requires to sample $T[N_1]$ and $T[N_2]$. 

![Figure 1: Tree representation of an online permutation and computation of $C = \pi_T(M)$](image-url)
2.2 Online authenticated encryption schemes

Given a tag-length parameter $\tau$, we let $T = B_\tau$ and call it the tag space. We will denote by $H$ a non-empty set called the header space. As a practical example, $H$ can be $\{0, 1\}^*$ or $B_h$ for some $h$. We let $C = B_n^* \times T$ and call it the ciphertext space, i.e. a ciphertext logically consists of a ciphertext core – that comprises encrypted blocks – and a tag. Hence we decompose ciphertexts with the canonical projections $\text{Core} : C \rightarrow B_n^*$ and $\text{Tag} : C \rightarrow T$.

An online authenticated encryption scheme is a triplet $\Pi = (K, E, D)$, where $K$ is a finite key space, $E : K \times H \times B_n^* \rightarrow C$ is a deterministic encryption algorithm, and $D : K \times H \times C \rightarrow B_n^* \cup \{\perp\}$ is a deterministic decryption algorithm. $T$ is the tag space of $\Pi$, $H$ is its header space and $n$ is its block size. We require that $\Pi$ is correct, i.e. for all $K \in K$, $H \in H$ and $M \in B_n^*$, we have $D(K, H, E(K, H, M)) = M$. We further require that the encryption is online, i.e. for all $K \in K$ and $H \in H$, $\text{Core} \circ E(K, H, \cdot) \in \text{OPerm}[n]$, or informally that the $i$-th block of the ciphertext core only depends on the first $i$ blocks of the plaintext.

2.3 Security Definitions

We formalize security with help of code-based games proposed by [BR04]. A game $G_\Pi$ for a scheme $\Pi$ consists of an Initialize procedure, procedures that model oracle queries and a Finalize procedure. All these procedures are defined in terms of the scheme $\Pi$.

When we say that an adversary $\mathcal{A}$ plays the game $G_\Pi$, we mean the sequential execution of: first the Initialize procedure, then the algorithm of the adversary using the oracle procedures defined by $G_\Pi$, and last the Finalize procedure using as input the final output of the adversary. The result returned by the Finalize procedure is called the output of this execution. For some games, we do not specify the Finalize procedure: in that case it is the trivial procedure that forwards the output of the adversary. For an adversary $\mathcal{A}$, we denote by $\Pr[\mathcal{A}^{G_\Pi} \Rightarrow x]$ the probability that when $\mathcal{A}$ plays the game $G$ instantiated with the scheme $\Pi$, the output of the Finalize procedure is $x$.

Two games $G$ and $G'$ are said to be identical until $\mathsf{bad}$ if they both contain a statement $\mathsf{bad} \leftarrow \text{true}$ such that their code is identical until this statement is executed. We recall the fundamental lemma of game playing.

Lemma 1 (Fundamental lemma of game playing [BR04]). Let $G$ and $G'$ be two games identical until $\mathsf{bad}$. Then for any adversary $\mathcal{A}$ and any output $x$:

$$\Pr[\mathcal{A}^{G_\Pi} \Rightarrow x] - \Pr[\mathcal{A}^{G'_\Pi} \Rightarrow x] \leq \Pr[\mathcal{A}^{G_\Pi} \text{ sets } \mathsf{bad}]$$

Randomized algorithms We denote by $x \xleftarrow{\$} S$ the sampling of an element $x$ from a set $S$ with uniform distribution. We note that we assume the use of lazy sampling when the set $S$ is “large” (e.g. $\text{PERM}[n]$) or infinite endowed with a natural definition of uniform distribution (e.g. $\text{OPerm}[n]$), and for statements of the form $\text{for } a \in A \text{ do } x \xleftarrow{\$} S$ when $A$ is a “large” set. All such samplings in a single algorithm are always independent.

Resource-parametrized adversarial advantage For each security property $\text{Prop}$, we define the advantage of an adversary $\mathcal{A}$ in attacking $\text{Prop}$ of a scheme $\Pi$. This is a real number that we denote by $\text{Adv}^{\text{Prop}}_\Pi(\mathcal{A})$. To capture the security of a scheme $\Pi$ for the notion $\text{Prop}$, we define the maximum advantage that can be achieved by any adversary $\mathcal{A}$ that is only limited in its resources. These resources are always: $t$, the time complexity of $\mathcal{A}$; $q$, the number of multi-block messages queried by $\mathcal{A}$; $b$, the total number of blocks queried by $\mathcal{A}$; $\mu$, the maximum number of blocks in a message queried by $\mathcal{A}$.

An adversary that uses at most these resources is called a $(t, q, b, \mu)$-adversary. We denote
Game OAE-REAL\(\Pi\)
proc Initialize
\( K \xleftarrow{\$} \mathcal{K} \)
proc Enc\((H, M)\)
return \( E(K, H, M) \)
proc Dec\((H, C)\)
return \( D(K, H, C) \)

Game OAE-IDEAL\(\Pi\)
proc Initialize
for \( H \in \mathcal{H} \) do \( \pi_H \xleftarrow{\$} \text{OPERM}[\mathbf{n}] \)
for \( (H, M) \in \mathcal{H} \times B^n \) do \( T_{H,M} \xleftarrow{\$} T \)
proc Enc\((H, M)\)
return \((\pi_H(M), T_{H,M})\)
proc Dec\((H, C)\)
return \( \bot \)

Figure 2: Games OAE-REAL\(\Pi\) (left) and OAE-IDEAL\(\Pi\) (right) used to define OAE.

by \( \text{Adv}^{\Pi}_{\text{OAE}}(t, q, b, \mu) \) the maximum of \( \text{Adv}^{\Pi}_{\text{PROP}}(\mathcal{A}) \) over \((t, q, b, \mu)\)-adversaries \( \mathcal{A} \). We further say that a scheme \( \Pi \) is \((t, q, b, \mu; \varepsilon)\)-secure for PROP if \( \text{Adv}^{\Pi}_{\text{PROP}}(t, q, b, \mu) \leq \varepsilon \).

3 Existing security notions and their variants
In this section, we present the existing security notions investigated in this paper, namely the recent notion of misuse-resistant online authenticated encryption [FFL12], and several notions of blockwise privacy and integrity, that were proposed a decade earlier [FJMV03, FJP04]. We adapt the latter definitions to the syntax of online authenticated encryption schemes as proposed in [FFL12].

3.1 Misuse-resistant online authenticated encryption
The notion of (nonce) misuse-resistant online authenticated encryption was given by Fleischmann et al. [FFL12]. Hoang et al. pointed out that the original definition was incomplete and reformulated it [HRRV15]. We retain the latter version of the notion. Roughly speaking, a scheme \( \Pi = (K, E, D) \) is OAE secure if the outputs of \( E \) resemble core ciphertexts computed by a random online permutation sampled independently for each header, followed by random tags sampled independently for each header-message pair, and if it is simultaneously hard to guess inputs to \( D \) that decrypt correctly. This is formalized in Definition 1.

Definition 1 (OAE [FFL12]). Consider the games OAE-REAL and OAE-IDEAL defined in Figure 2. For an online authenticated encryption scheme \( \Pi = (K, E, D) \), and an adversary \( \mathcal{A} \) that never queries the Dec oracle with a result of a previous Enc query, we define the OAE advantage of \( \mathcal{A} \) against \( \Pi \) as:

\[
\text{Adv}^{\Pi}_{\text{OAE}}(\mathcal{A}) = \Pr[\mathcal{A}^{\text{OAE-REAL}} \Rightarrow 1] - \Pr[\mathcal{A}^{\text{OAE-IDEAL}} \Rightarrow 1].
\]

3.2 Blockwise integrity of ciphertexts
FJMV proposed a notion of integrity in the context of a blockwise chosen plaintext attack [FJMV03]. Compared to the classical integrity of ciphertexts notion [BN00, KY01], the adversary is given a blockwise encryption oracle along with a standard decryption oracle, and wins if it can produce an existential forgery.

In its original form, the notion of FJMV is defined for randomized schemes that do not support headers. We therefore recast the integrity notion of FJMV to make it compatible
with deterministic online AE schemes that take a header along with the message to encrypt (or decrypt). We give in Definition 2 what we believe to be the natural extension of the original notion of FJMV.

Definition 2 (B-INT-CTX). Consider the game B-INT-CTX defined in Figure 3. For an encryption scheme $\Pi = (K, E, D)$, we define the B-INT-CTX advantage of an adversary $\mathcal{A}$ against $\Pi$ as:

$$\text{Adv}_{\Pi}^{\text{B-INT-CTX}}(\mathcal{A}) = \Pr[\mathcal{A}^{\text{B-INT-CTX}}_{\Pi} \Rightarrow 1].$$

The Enc and the GetTag queries together form a complete blockwise encryption oracle. The current adversarial plaintext is encrypted and accumulated block-by-block in the variable $\tilde{M}$ through Enc queries, while a GetTag query returns the tag for the current (possibly empty) plaintext. The first if-statement in a Dec query makes sure that the adversarial forgery attempt is valid.

### 3.3 Blockwise privacy

FJMV proposed a notion of indistinguishability of ciphertexts in the context of a blockwise chosen plaintext attack [FJMV03]. The adversary was given more adaptive power: it was allowed to get the encryption of each block before requesting the encryption of the next block. The definition of FJMV uses the left-or-right style of indistinguishability, common for early notions of privacy [BDJR97, BN00]. The definition did not, however, explicitly state whether it allowed the adversary to query an infinite stream of blocks only, or if starting new blockwise queries was allowed. Later, Fouque, Joux and Poupart described several versions of the blockwise indistinguishability notion, that allowed for both infinite streams of blocks and multiple queries, further distinguishing between sequential and concurrent execution of multiple queries [FMP03, FJP04]. We focus on the version that allows to make multiple queries in a sequential manner that we call LORS-BCPA (as in [FJP04]).

**Recasting LORS-BCPA** As before, we attempt to recast the original definition for randomized encryption algorithms to make it compatible with deterministic online au-

---

**Figure 3:** Game B-INT-CTX used to define blockwise integrity of ciphertexts.
then the underlying cipher in a permutation, so two ciphertexts are equal if and only if the
same header and repeating a message only on the left side. Fixing the header turns
to trivially break any (not just online) deterministic AE scheme using two queries with
imposing the non-repetition of headers seems to be an unnecessarily
attacks to be mounted. An example of a subtler attack that leverages the online property
limiting constraint, that prevents more than just the unavoidable trivial attacks. In case
proves to be a non-trivial task because of the left-or-right flavour of the indistinguishability.
Authenticated encryption schemes that take a header alongside a plaintext as input. This
proves to be a non-trivial task because of the left-or-right flavour of the indistinguishability.
Allowing the adversary to issue encryption queries with no restrictions would allow it
to trivially break any (not just online) deterministic AE scheme using two queries with
the same header and repeating a message only on the left side. Fixing the header turns
the underlying cipher in a permutation, so two ciphertexts are equal if and only if the
corresponding plaintexts are equal. Figure 4 shows an example of such an attacker $\mathcal{A}_r$, where $0_H$ is an arbitrary element of the header space $\mathcal{H}$.

Such attacks can be thwarted by forbidding the adversary to repeat headers in its
queries. However, imposing the non-repetition of headers seems to be an unnecessarily
limiting constraint, that prevents more than just the unavoidable trivial attacks. In case
of deterministic online AE schemes, it is the online property itself that allows trivial
attacks to be mounted. An example of a subtler attack that leverages the online property
is described in Figure 4; adversary $\mathcal{A}_r'$ only repeats the first block on the left side, and
leverages the online property of the scheme.

We propose to restrict the LORS-BCPA notion to a class of adversaries specified
in Definition 3 whose queries verify certain conditions that avoid trivial attacks, but
can for example repeat headers. We will call adversaries that respect these conditions
online-respecting adversaries.

**Definition 3** (Online-respecting adversary). Consider an adversary $\mathcal{A}$ that plays the
LORS-BCPA game and queries a sequence of multi-block messages $((H_i, M_{0,i}, M_{1,i})) \in (\mathcal{H} \times B^* \times B^*_n)^*$ to the LR oracle. We say that $\mathcal{A}$ is online-respecting if for all pairs of indices $(i,j)$ such that $H_i = H_j$, we have \(\text{LLCP}_n(M_{0,i}, M_{0,j}) = \text{LLCP}_n(M_{1,i}, M_{1,j})\).

In other words, if two left messages requested with the same header share a common
prefix of $l$ blocks, then the associated right messages must also share a common prefix of
exactly $l$ blocks, and conversely. The online-respecting property can be rephrased using
online permutations, as in Proposition 1.

**Proposition 1.** Consider an adversary $\mathcal{A}$ that plays the LORS-BCPA game and queries
a sequence of multi-block messages $((H_i, M_{0,i}, M_{1,i})) \in (\mathcal{H} \times B^*_n \times B^*_n)^*$ to the LR oracle. Then $\mathcal{A}$ is online-respecting if and only if for every header $H \in \mathcal{H}$ there exists an
online-permutation $\sigma_H \in \text{OPERM}[n]$ such that for every query (indexed by $i$) we have
$M_{1,i} = \sigma_H(M_{0,i})$.

**Proof.** If the queries respect this condition, the adversary is online-respecting because for
indices $(i,j)$ such that $H_i = H_j = H$, the length of common prefixes verify the equality
$\text{LLCP}_n(M_{1,i}, M_{1,j}) = \text{LLCP}_n(\sigma_H(M_{0,i}), \sigma_H(M_{0,j})) = \text{LLCP}_n(M_{0,i}, M_{0,j})$.
Conversely, if the adversary is online-respecting, let’s show that suitable online-
permutations $\sigma_H$ exist. For each header $H \in \mathcal{H}$ and each message $M \in B^*_n$, we identify
constraints that the permutation $\sigma_H[M]$ must satisfy. Let $l = \text{BlCount}(M)$ and $S_{H,M}$ be

\[
\begin{align*}
\text{Adversary } \mathcal{A}_r & \quad \text{for } i \in \{0, \ldots, 2\} \text{ do } M_i \leftarrow \mathcal{H}(0_H, M_0, M_1) \\
& \quad \triangleright \text{ such that } M_i \text{ are distinct} \\
C_a & \leftarrow \text{LR}(0_H, M_0, M_1) \\
C_b & \leftarrow \text{LR}(0_H, M_0, M_2) \\
\text{output } C_a & \neq C_b \\
\text{Adversary } \mathcal{A}_r' & \quad \text{for } i \in \{0, \ldots, 6\} \text{ do } M_i \leftarrow \mathcal{H}(0_H, M_0, M_1) \\
& \quad \triangleright \text{ such that } M_i \text{ are distinct} \\
C_a & \leftarrow \text{LR}(0_H, M_0 || M_1 || M_2) \\
C_b & \leftarrow \text{LR}(0_H, M_0 || M_4 || M_5 || M_6) \\
\text{output } C_a[0] & \neq C_b[0]
\end{align*}
\]

**Figure 4:** Header-repeating adversaries against blockwise left-or-right notions.
Game LORS-BCPA$_\Pi$

```plaintext
proc Initialize
    \( K \xrightarrow{\$} K \)
    \( b \xrightarrow{\$} \{0, 1\} \)
    \( \tilde{H} \leftarrow \bot \)
    \( \tilde{M} \leftarrow \varepsilon \)
    \( j \leftarrow 0 \)
proc LR(\( H, P_0, P_1 \)) \( \triangleright P_0, P_1 \in B_n \)
    if \( \tilde{H} = \bot \) then \( \tilde{H} \leftarrow H \)
    \( \tilde{M} \leftarrow \tilde{M}|P_b \)
    \( C \leftarrow \text{Core}(\mathcal{E}(K, \tilde{H}, \tilde{M})) \)
    \( j \leftarrow j + 1 \)
    return \( C[j] \)
proc GetTag(\( H \))
    if \( \tilde{H} = \bot \) then \( \tilde{H} \leftarrow H \)
    \( T \leftarrow \text{TAG}(\mathcal{E}(K, \tilde{H}, \tilde{M})) \)
    \( \tilde{H} \leftarrow \bot \)
    \( \tilde{M} \leftarrow \varepsilon \)
    \( j \leftarrow 0 \)
    return \( T \)
proc Finalize(\( \beta \))
    return \( \beta = b \)
```

Figure 5: Game LORS-BCPA$_\Pi$ used to define blockwise privacy.

the set of queries \( i \) such that \( H_i = H \) and \( M \) is a strict prefix of \( M_{0,i} \). Then by the online-respecting property, for all \( i, j \in S_{H,M}, M_{0,i}[l+1] = M_{0,j}[l+1] \Leftrightarrow M_{1,i}[l+1] = M_{1,j}[l+1], \) because \( M_{0,i}[1..l] = M_{0,j}[1..l] = M. \) This implies that there exists a bijection (i.e. a permutation) \( \sigma_H[M] \) such that for all \( i \in S_{H,M}, M_{1,i}[l+1] = \sigma_H[M](M_{0,i}[l+1]). \) With such construction of the nodes \( \sigma_H[M] \), the online permutations \( \sigma_H \) verify \( M_{1,i} = \sigma_H(M_{0,i}). \)

We finally define the D-LORS-BCPA (a.k.a. deterministic-LORS-BCPA) notion for deterministic online AE schemes that considers online-respecting adversaries in Definition 4.

**Definition 4 (D-LORS-BCPA).** Consider the game LORS-BCPA defined in Figure 5. For an encryption scheme \( \Pi = (K,E,D) \), we define the D-LORS-BCPA advantage of an online-respecting adversary \( \mathcal{A} \) against \( \Pi \) as:

\[
\text{Adv}^{\text{D-LORS-BCPA}}_{\Pi}(\mathcal{A}) = 2 \cdot \Pr[\mathcal{A}^{\text{LORS-BCPA}} \Rightarrow 1] - 1.
\]

Intuitively, the LORS-BCPA security game preserves the left-or-right character of the original notions by FJMV and an adversary can construct queries adaptively, block-by-block. The D-LORS-BCPA notion then additionally requires the adversary to be online-respecting, which is necessary to prevent trivial victories against deterministic schemes.

**Remark 1.** FJMV have shown that blockwise security notions are stronger than non-blockwise ones in the case of randomized encryption [FJP04]. However, they noted that in the case of deterministic schemes with online-respecting adversaries, a blockwise encryption oracle can be straightforwardly simulated using an atomic encryption oracle by keeping a variable like \( \tilde{M} \) in Figure 5. This reduction is, however, not tight but quadratic, as it grows the number of adaptively queried blocks \( m \) to \( \sum_{i=1}^{m} = \frac{m(m+1)}{2} \) non-adaptively queried blocks. Hence, studying blockwise security notions is still relevant to obtain tighter relations.

## 4 Relations between blockwise notions and OAE

In this section, we first prove that OAE security implies D-LORS-BCPA and B-INT-CTX security, up to a quadratic increase in resources. We then show that the converse
is not true, and propose a new auxiliary notion called PR-TAG. Finally, we show that there is an equivalence between OAE security and the conjunction of D-LORS-BCPA, B-INT-CTX and PR-TAG security.

### 4.1 Separating OAE and Blockwise Notions

**OAE → D-LORS-BCPA**

**Theorem 1.** Let \( \Pi \) be an online authenticated encryption scheme. Then

\[
\text{Adv}^{\text{D-LORS-BCPA}}_{\Pi}(t, q, b, \mu) \leq 2 \cdot \text{Adv}^{\text{OAE}}_{\Pi}(t', q', b', \mu)
\]

where \( t' = t + c \cdot q', q' = q + b, b' = b + \min\left(q \cdot \frac{\mu(n+1)}{2}, \frac{b(b+1)}{2}\right) \) for a positive constant \( c \).

**Proof.** Let \( \mathcal{A}_1 \) be an online-respecting \((t, q, b, \mu)\)-D-LORS-BCPA adversary against scheme \( \Pi \). We construct adversary \( \mathcal{B}_1 \) as shown on Figure 6. The number of messages queried by \( \mathcal{B}_1 \) is at most \( q' \), because at most \( b \) messages are queried via \( \text{LR} \), and at most \( q \) messages are queried via \( \text{GetTag} \). The time complexity of \( \mathcal{B}_1 \) is \( t' = t + c \cdot q' \) because a constant time is spent for each query of \( \mathcal{B}_1 \). The number of blocks queried by \( \mathcal{B}_1 \) is at most \( b' \) because at most \( b \) blocks are queried via \( \text{GetTag} \) and for each message query, at most \( \sum_{i=1}^{b} i \) blocks are queried via \( \text{LR} \), and in total at most \( \sum_{i=1}^{b} i \) blocks are queried via \( \text{LR} \). The maximum number of blocks in a single query remains unchanged. We then have the following relations between the advantages of \( \mathcal{A}_1 \) and \( \mathcal{B}_1 \).

\[
\text{Adv}^{\text{OAE}}_{\Pi}(\mathcal{B}_1) = \Pr[\mathcal{A}_1^{\text{OAE-REAL}}_{\Pi} \Rightarrow 1] - \Pr[\mathcal{A}_1^{\text{OAE-IDEAL}}_{\Pi} \Rightarrow 1]
\]

\[
= \Pr[\mathcal{A}_1^{\text{OAE-REAL}}_{\Pi} \Rightarrow 1] - \frac{1}{2}
\]

\[
= \Pr[\mathcal{A}_1^{\text{LORS-BCPA}}_{\Pi} \Rightarrow 1] - \frac{1}{2}
\]

\[
= \frac{1}{2} \text{Adv}^{\text{D-LORS-BCPA}}_{\Pi}(\mathcal{A}_1)
\]

The first equality comes from the fact that if \( \mathcal{B}_1 \) interacts with the OAE-IDEAL game, the distribution of the replies of \( \mathcal{A}_1 \)'s LR oracle is independent from \( b \). Indeed, \( \mathcal{A}_1 \) is online-respecting so by Proposition 1, the left and right queries are identical up to online-permutations \( \sigma_H \), and by Lemma 5, the distribution of the replies is independent from \( b \).
### Figure 7: Adversary $B_2$ for the proof of Theorem 2.

The second equality comes from the construction of adversary $B_1$, that perfectly simulates the LORS-BCPA game when the underlying oracle is the OAE-REAL game.

**OAE → B-INT-CTXT**

**Theorem 2.** Let $\Pi$ be an online authenticated encryption scheme. Then

$$\text{Adv}_{\Pi}^{\text{B-INT-CTXT}}(t, q, b, \mu) \leq \text{Adv}_{\Pi}^{\text{OAE}}(t', q', b', \mu)$$

where $t' = t + c \cdot q'$, $q' = q + b$, $b' = b + \min \left( q \cdot \frac{\mu(\mu + 1)}{2}, \frac{b(b+1)}{2} \right)$ for a positive constant $c$.

**Proof.** Let $\mathcal{A}_2$ be a $(t, q, b, \mu)$-B-INT-CTXT adversary against scheme $\Pi$. We construct adversary $B_2$ as shown on Figure 7. The number of messages queried by $B_2$ is at most $q'$, because at most $b$ messages are queried via $\text{Enc}$, and at most $q$ messages are queried via $\text{GetTag}$ and $\text{Dec}$. The time complexity of $B_2$ is at most $t' = t + c \cdot q'$ because a constant time is spent for each query of $B_2$. The number of blocks queried by $B_2$ is at most $b'$ because at most $b$ blocks are queried via $\text{GetTag}$ and $\text{Dec}$ and for each message query, at most $\sum_{i=1}^{\mu} i$ blocks are queried via $\text{Enc}$, and in total at most $\sum_{i=1}^{b} i$ blocks are queried via $\text{Enc}$. The maximum number of blocks in a single query remains unchanged. We then have the following relations between the advantages of $\mathcal{A}_2$ and $B_2$.

$$\text{Adv}_{\Pi}^{\text{OAE}}(\mathcal{A}_2) = \Pr[\mathcal{A}_2^{\text{OAE-REAL}} \Rightarrow 1] - \Pr[\mathcal{B}_2^{\text{OAE-IDEAL}} \Rightarrow 1]$$

$$= \Pr[\mathcal{A}_2^{\text{B-INT-CTXT}} \Rightarrow 1] - 0$$

$$= \text{Adv}_{\Pi}^{\text{B-INT-CTXT}}(\mathcal{A}_2)$$

In the case of the OAE-IDEAL game, the adversary $B_2$ always outputs 0. In the case of the OAE-REAL game, adversary $B_2$ outputs 1 if and only if adversary $\mathcal{A}_2$ forges a valid ciphertext-tag pair.

**D-LORS-BCPA + B-INT-CTXT → OAE**

**Proposition 2.** If there exists an online authenticated encryption scheme $\Pi$ which is D-LORS-BCPA and B-INT-CTXT secure, then there exists a scheme $\Pi'$ which is D-LORS-BCPA and B-INT-CTXT secure but not OAE secure.
Figure 8: Definitions of $\Pi'$ (left and middle) and adversary $A_3$ (right) for the proof of Proposition 2.

Proof. Let $\Pi = (K, E, D)$ be a D-LORS-BCPA and B-INT-CTXT secure scheme, with tag-length $\tau$. We will construct a scheme $\Pi' = (K, E', D')$ with tag-length $\tau + 1$ that is D-LORS-BCPA and B-INT-CTXT secure, but not OAE secure.

The idea is to append a constant bit to the tag, so that it is easily distinguishable from a random string (Figure 8). Clearly, appending this constant bit does not change the D-LORS-BCPA nor B-INT-CTXT advantages. However, adversary $A_3$ (Figure 8) can obtain a constant advantage over OAE by making a single query. The OAE advantage of adversary $A_3$ is equal to $\frac{1}{2}$ since $\beta$ is always equal to 1 in the OAE-REAL game, and is uniformly distributed in the OAE-IDEAL game, so $\text{Adv}_{OAE}^{OAE}(A_3) = \Pr[A_3^{OAE-REAL} \Pi' \Rightarrow 1] - \Pr[A_3^{OAE-IDEAL} \Pi' \Rightarrow 1] = 1 - \frac{1}{2} = \frac{1}{2}$.

4.2 Towards an equivalence result

As shown in Proposition 2, D-LORS-BCPA security and B-INT-CTXT security combined are not sufficient to provide OAE security. As suggested by the proof of Proposition 2, this comes from the fact that the tag does not have to be uniformly distributed to provide D-LORS-BCPA and B-INT-CTXT. We define a new auxiliary notion called PR-TAG that captures the pseudo-randomness of the tag in a chosen plaintext attack, and we show an equivalence result with OAE.

**Pseudo-random tag** We introduce the PR-TAG (pseudo-random tag) notion a.k.a. indistinguishability from a random tag. More precisely, an adversary wins the PR-TAG game if it can distinguish real ciphertexts from pairs composed of the real core ciphertext and a random tag. This is captured in Definition 5.

**Definition 5 (PR-TAG).** Consider the games PR-TAG-REAL and PR-TAG-IDEAL defined in Figure 9. For an online authenticated encryption scheme $\Pi = (K, E, D)$, we define the PR-TAG advantage of an adversary $\mathcal{A}$ against $\Pi$ as:

$$\text{Adv}_{\Pi}^{PR-TAG}(\mathcal{A}) = \Pr[\mathcal{A}^{PR-TAG-REAL} \Pi \Rightarrow 1] - \Pr[\mathcal{A}^{PR-TAG-IDEAL} \Pi \Rightarrow 1]$$

**OAE → PR-TAG** We first prove that OAE security implies PR-TAG security, up to twice the advantage.

**Theorem 3.** Let $\Pi$ be an online authenticated encryption scheme. Then

$$\text{Adv}_{\Pi}^{PR-TAG}(t, q, b, \mu) \leq 2 \cdot \text{Adv}_{\Pi}^{OAE}(t', q, b, \mu)$$

where $t' = t + c \cdot q$ for a positive constant $c$. 

```markdown
<table>
<thead>
<tr>
<th>Algorithm $\mathcal{E}'(K, H, M)$</th>
<th>Algorithm $\mathcal{D}'(K, H, C)$</th>
<th>Adversary $\mathcal{A}_3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C \leftarrow E(K, H, M)$</td>
<td>$T</td>
<td></td>
</tr>
<tr>
<td>$T' \leftarrow \text{Tag}(C)$</td>
<td>if $b \neq 1$ then return 1</td>
<td>$T</td>
</tr>
<tr>
<td>return $(\text{Core}(C), T')$</td>
<td>$C' \leftarrow (\text{Core}(C), T)$</td>
<td>output $\beta$</td>
</tr>
</tbody>
</table>
```
**Game PR-TAG-REAL**<sub>Π</sub>

<table>
<thead>
<tr>
<th>proc Initialize</th>
</tr>
</thead>
<tbody>
<tr>
<td>( K \xleftarrow{$} \mathcal{K} )</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>proc Enc ((H, M))</th>
</tr>
</thead>
<tbody>
<tr>
<td>return ( E(K, H, M) )</td>
</tr>
</tbody>
</table>

**Game PR-TAG-IDEAL**<sub>Π</sub>

<table>
<thead>
<tr>
<th>proc Initialize</th>
</tr>
</thead>
<tbody>
<tr>
<td>( K \xleftarrow{$} \mathcal{K} )</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>for ((H, M) \in \mathcal{H} \times B_n^*) do</th>
</tr>
</thead>
<tbody>
<tr>
<td>( T_{H,M} \xleftarrow{$} \mathcal{T} )</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>proc Enc ((H, M))</th>
</tr>
</thead>
<tbody>
<tr>
<td>( C \leftarrow \text{Core}(E(K, H, M)) )</td>
</tr>
<tr>
<td>return ((C, T_{H,M}))</td>
</tr>
</tbody>
</table>

**Figure 9:** Games PR-TAG-REAL<sub>Π</sub> (left) and PR-TAG-IDEAL<sub>Π</sub> (right) used to define PR-TAG.

**Figure 10:** Adversary \( B_4 \) for the proof of Theorem 3.

**Proof.** Let \( \mathcal{A}_4 \) be a \((t, q, b, \mu)\)-PR-TAG adversary against a scheme \( \Pi \). We note that since all the oracles available to PR-TAG adversaries are contained in the OAE-REAL and OAE-IDEAL games, we can also view \( \mathcal{A}_4 \) as an OAE adversary. We insert the OAE-IDEAL game in the expression of the PR-TAG advantage of \( \mathcal{A}_4 \) and we will bound each half of this new expression:

\[
\text{Adv}_{\Pi}^{\text{PR-TAG}}(\mathcal{A}_4) = \text{Pr}[\mathcal{A}_4^{\text{PR-TAG-REAL}} \Rightarrow 1] - \text{Pr}[\mathcal{A}_4^{\text{PR-TAG-IDEAL}} \Rightarrow 1] \\
= \text{Pr}[\mathcal{A}_4^{\text{PR-TAG-REAL}} \Rightarrow 1] - \text{Pr}[\mathcal{A}_4^{\text{OAE-IDEAL}} \Rightarrow 1] \\
+ \text{Pr}[\mathcal{A}_4^{\text{OAE-IDEAL}} \Rightarrow 1] - \text{Pr}[\mathcal{A}_4^{\text{PR-TAG-IDEAL}} \Rightarrow 1].
\]

First, we note that by construction, the game OAE-REAL perfectly simulates the game PR-TAG-REAL for adversaries that only use the \text{Enc} oracle, such as adversary \( \mathcal{A}_4 \), which means that \( \text{Pr}[\mathcal{A}_4^{\text{PR-TAG-REAL}} \Rightarrow 1] = \text{Pr}[\mathcal{A}_4^{\text{OAE-REAL}} \Rightarrow 1] \) and we obtain

\[
\text{Pr}[\mathcal{A}_4^{\text{PR-TAG-REAL}} \Rightarrow 1] - \text{Pr}[\mathcal{A}_4^{\text{OAE-IDEAL}} \Rightarrow 1] \leq \text{Adv}_{\Pi}^{\text{OAE}}(t', q, b, \mu).
\]

Second, given the PR-TAG adversary \( \mathcal{A}_4 \), we construct the OAE adversary \( B_4 \) as shown on Figure 10. The time complexity of \( B_4 \) is \( t' = t + c \cdot q \) because a constant time is spent for each query of \( \mathcal{A}_4 \). The number of message queries and the number of block queries remain unchanged.

By construction, adversary \( B_4 \) perfectly simulates game PR-TAG-IDEAL when the underlying game is OAE-REAL. Note that the output bit of adversary \( \mathcal{A}_4 \) is flipped by adversary \( B_4 \), i.e. \( \text{Pr}[\mathcal{A}_4^{\text{PR-TAG-IDEAL}} \Rightarrow 1] = 1 - \text{Pr}[\mathcal{B}_4^{\text{OAE-REAL}} \Rightarrow 1] \).

When the underlying game is OAE-IDEAL, adversary \( B_4 \) gives answers to \( \mathcal{A}_4 \) that are identically distributed to the OAE-IDEAL game, because \( B_4 \) redundantly replaces a random tag that was independent from the random core by another independent random tag, and thus \( \text{Pr}[\mathcal{A}_4^{\text{OAE-IDEAL}} \Rightarrow 1] = 1 - \text{Pr}[\mathcal{B}_4^{\text{OAE-IDEAL}} \Rightarrow 1] \). It follows that

\[
\text{Pr}[\mathcal{A}_4^{\text{OAE-IDEAL}} \Rightarrow 1] - \text{Pr}[\mathcal{A}_4^{\text{PR-TAG-IDEAL}} \Rightarrow 1] \leq \text{Adv}_{\Pi}^{\text{OAE}}(t', q, b, \mu).
\]
Games $G_0$ and $G_1$

\begin{tabular}{|l|l|}
\hline
\multicolumn{1}{|l|}{proc Initialize} & \multicolumn{1}{|l|}{Game $G_2$} \\
\hline
\begin{aligned}
K & \xleftarrow{} K \\
X & \xleftarrow{} \emptyset
\end{aligned} & \begin{aligned}
K & \xleftarrow{} K \\
\text{proc Enc}(H, M) & \begin{aligned}
C & \xleftarrow{} \mathcal{E}(K, H, M) \\
X & \leftarrow X \cup \{(H, C)\}
\end{aligned} \\
\text{proc Dec}(H, C) & \begin{aligned}
M & \leftarrow D(K, H, C) \\
\text{if } (H, C) \in X \text{ then } M & \leftarrow \bot \\
\text{if } M \neq \bot \text{ then } \text{bad} & \leftarrow \text{true}; M \leftarrow \bot
\end{aligned}
\end{aligned}
\end{tabular}

\begin{figure}[h]
\centering
\begin{tabular}{|l|l|}
\hline
proc Initialize & proc Initialize \\
$K \xleftarrow{} K$ & $K \xleftarrow{} K$ \\
$X \xleftarrow{} \emptyset$ & proc $\text{Enc}(H, M)$ \\
& return $\mathcal{E}(K, H, M)$ \\
proc $\text{Enc}(H, M)$ & proc $\text{Dec}(H, C)$ \\
$C \xleftarrow{} \mathcal{E}(K, H, M)$ & return $\bot$ \\
$X \leftarrow X \cup \{(H, C)\}$ & \\
\hline
\end{tabular}
\caption{Games $G_0$, $G_1$ (left) and $G_2$ (right) for the reduction of B-INT-CTXT. The boxed statement is present in game $G_1$ only.}
\end{figure}

\section{D-LORS-BCPA + B-INT-CTXT + PR-TAG $\rightarrow$ OAE}

\textbf{Theorem 4.} Let $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be an online authenticated encryption scheme. Then

$$
\text{Adv}_{\Pi}^{\text{OAE}}(t, q, b, \mu) \leq \text{Adv}_{\Pi}^{\text{B-INT-CTXT}}(t_c, q_c, b_c, \mu_c) + \text{Adv}_{\Pi}^{\text{PR-TAG}}(t_t, q_t, b_t, \mu_t) + \text{Adv}_{\Pi}^{\text{D-LORS-BCPA}}(t_p, q_p, b_p, \mu_p)
$$

where $t_c = t + c_c \cdot (q + b)$, $q_c = q$, $b_c = b$, $\mu_c = \mu$; $t_t = t + c_t \cdot q$, $q_t = q$, $b_t = b$, $\mu_t = \mu$; $t_p = t + c_p \cdot (q + b)$, $q_p = q$, $b_p = b$, $\mu_p = \mu$ for positive constants $c_c, c_t, c_p$.

\textbf{Proof.} Let $\mathcal{A}$ be a $(t, q, b, \mu)$-OAE adversary against scheme $\Pi$. We define games $G_i$ for $i \in \{0, \ldots, 4\}$ on Figure 11, Figure 13 and Figure 14. We have, by Lemma 2, Lemma 3 and Lemma 4 proven below that:

$$
\begin{aligned}
\text{Pr}[\mathcal{A}^{\text{OAE-REAL}}_{\Pi} \Rightarrow 1] & \leq \text{Adv}_{\Pi}^{\text{B-INT-CTXT}}(t_c, q_c, b_c, \mu_c), \\
\text{Pr}[\mathcal{A}^{\text{G2n}}_{\Pi} \Rightarrow 1] & \leq \text{Adv}_{\Pi}^{\text{B-INT-CTXT}}(t_t, q_t, b_t, \mu_t), \\
\text{Pr}[\mathcal{A}^{\text{G2n}}_{\Pi} \Rightarrow 1] & \leq \text{Adv}_{\Pi}^{\text{D-LORS-BCPA}}(t_p, q_p, b_p, \mu_p),
\end{aligned}
$$

which gives us

$$
\text{Adv}_{\Pi}^{\text{OAE}}(\mathcal{A}) \leq \text{Adv}_{\Pi}^{\text{B-INT-CTXT}}(t_c, q_c, b_c, \mu_c) + \text{Adv}_{\Pi}^{\text{PR-TAG}}(t_t, q_t, b_t, \mu_t) + \text{Adv}_{\Pi}^{\text{D-LORS-BCPA}}(t_p, q_p, b_p, \mu_p),
$$

\ \square

\textbf{Lemma 2.} Let $\mathcal{A}$ be a $(t, q, b, \mu)$-OAE adversary. Let $G_0$, $G_1$ and $G_2$ be the games defined in Figure 11. We have

$$
\text{Pr}[\mathcal{A}^{\text{OAE-REALn}}_{\Pi} \Rightarrow 1] - \text{Pr}[\mathcal{A}^{\text{G2n}}_{\Pi} \Rightarrow 1] \leq \text{Adv}_{\Pi}^{\text{B-INT-CTXT}}(t', q, b, \mu)
$$

where $t' = t + c_c \cdot (q + b)$.

\textbf{Proof.} Let $\mathcal{A}'$ be the adversary defined from $\mathcal{A}$ on Figure 12. The time complexity of $\mathcal{A}'$ is $t' = t + c_c \cdot (q + b)$ because a constant time is spent for each block queried by $\mathcal{A}$ to call $\text{Enc}$, as well as for each query of $\mathcal{A}$ to call $\text{GetTag}$ and $\text{Dec}$. The number of message queries and the number of block queries remain unchanged.
By definition, we recall that \( \mathcal{A} \) never queries to the Dec oracle the result of a previous Enc query. We then have:

\[
\Pr[\mathcal{A}^{\text{OAE-REAL}} \Rightarrow 1] = \Pr[\mathcal{A}^{\text{OAE}} \Rightarrow 1]
\]

\[
= \Pr[\mathcal{A}^{\text{G}1} \Rightarrow 1] + (\Pr[\mathcal{A}^{\text{G}1} \Rightarrow 1] - \Pr[\mathcal{A}^{\text{G}0} \Rightarrow 1])
\]

\[
\leq \Pr[\mathcal{A}^{\text{G}1} \Rightarrow 1] + \Pr[\mathcal{A}^{\text{OAE}} \text{ sets bad}]
\]

where the last inequality comes from Lemma 1. Then, we note that \( \Pr[A^{\text{G}1} \Rightarrow 1] = \Pr[A^{\text{G}0} \Rightarrow 1] \) because the Dec oracle of \( G_1 \) always returns \( \bot \). We have \( \Pr[A^{\text{G}0} \text{ sets bad}] \leq \text{Adv}_{\Pi}^{\text{B-INT-ctxt}}(A_c) \) because game \( G_0 \) sets bad whenever a ciphertext forgery is found by adversary \( \mathcal{A} \), and consequently by adversary \( \mathcal{A}_c \). These inequalities prove that \( \Pr[A^{\text{OAE-REAL}} \Rightarrow 1] - \Pr[A^{\text{OAE}} \Rightarrow 1] \leq \text{Adv}_{\Pi}^{\text{PR-TAG}}(t', q, b, \mu) \).

**Lemma 3.** Let \( \mathcal{A} \) be a \((t, q, b, \mu)\)-OAE adversary. Let \( G_3 \) be the game defined in Figure 13. We have:

\[
\Pr[\mathcal{A}^{\text{G}2} \Rightarrow 1] - \Pr[\mathcal{A}^{\text{G}3} \Rightarrow 1] \leq \text{Adv}_{\Pi}^{\text{PR-TAG}}(t', q, b, \mu)
\]

where \( t' = t + c_t \cdot q \).

**Proof.** Let \( \mathcal{A}_t \) be the adversary defined from \( \mathcal{A} \) on Figure 13. The time complexity of \( \mathcal{A}_t \) is \( t' = t + c_t \cdot q \) because a constant time is spent for each query of \( \mathcal{A} \). The number of messages queried by \( \mathcal{A}_t \) is at most \( q \), the number of blocks queried by \( \mathcal{A}_t \) is at most \( b \) and the maximum number of blocks in a query of \( \mathcal{A}_t \) is at most \( \mu \).

We have the following perfect simulations of the PR-TAG games: \( \Pr[A^{\text{G}2} \Rightarrow 1] = \Pr[A_t^{\text{PR-TAG-REAL}} \Rightarrow 1] \) and \( \Pr[A^{\text{G}3} \Rightarrow 1] = \Pr[A_t^{\text{PR-TAG-IDEAL}} \Rightarrow 1] \). From this, and
where $t$

40 Linking OAE and Blockwise Attack Models

Adv

Pr[Pr[

For $H \in \mathcal{H}$ do $\sigma_\mathcal{H} \leftarrow \text{OPERM}[n]$

for $(H, M) \in \mathcal{H} \times B_n^*$ do $T_{H, M} \leftarrow \mathcal{T}$

Run $\mathcal{A}$

On query $\text{Enc}(H, M)$ of $\mathcal{A}$

$l \leftarrow \text{BlCount}(M)$

$M[1] || \ldots || M[l] \leftarrow M$

$R[1] || \ldots || R[l] \leftarrow \sigma_\mathcal{H}(M)$

for $j \in \{1, \ldots, l\}$ do

$C[j] \leftarrow \text{LR}(H, R[j], M[j])$

$T \leftarrow \text{GetTag}(H)$

return $(C, T_{H, M})$ to $A$

On query $\text{Dec}(H, C)$ of $\mathcal{A}$

return $\perp$ to $A$

On Finalize($\beta$) of $\mathcal{A}$

output $\beta$

Figure 14: Game $G_4$ (left) and adversary $\mathcal{A}_p$ (right) for the proof of Lemma 4.

from Definition 5 it follows that $\Pr[A_{\text{G2n}^{\perp} \Rightarrow 1}] - \Pr[A_{\text{G2n}^{\leftarrow 1} \Rightarrow 1}] \leq \text{Adv}_{\mathcal{H}}^{\text{PR-TAG}}(A_t) \leq \text{Adv}_{\mathcal{H}}^{\text{PR-TAG}}(t', q, b, \mu)$.

Lemma 4. Let $\mathcal{A}$ be a $(t, q, b, \mu)$-OAE adversary. We have:

$$\Pr[\mathcal{A}_{\text{G2n}^{\perp} \Rightarrow 1}] - \Pr[\mathcal{A}_{\text{OAE-IDEAL}^{\perp} \Rightarrow 1}] \leq \text{Adv}_{\mathcal{H}}^{\text{LORS-BCPA}}(t', q, b, \mu)$$

where $t' = t + c_p \cdot (q + b)$.

Proof. Let $\mathcal{A}_p$ be the adversary defined from $\mathcal{A}$ on Figure 14. The time complexity of $\mathcal{A}_p$ is $t' = t + c_p \cdot (q + b)$ because a constant time is spent for each block queried by $\mathcal{A}$ to call LR, as well as for each query of $\mathcal{A}$ to call GetTag or upon Dec. The number of messages queried by $\mathcal{A}_p$ is at most $q$, the number of blocks queried by $\mathcal{A}_p$ is at most $b$ and the maximum number of blocks in a query of $\mathcal{A}_p$ is at most $\mu$.

By Proposition 1, $\mathcal{A}_p$ is online-respecting. Let $G_4$ be the game defined on Figure 14. If we denote by $b$ the bit chosen by the LORS-BCPA game played by $\mathcal{A}_p$, we have the following perfect simulations. If $b = 1$, then the LR oracle always encrypts the message $M$ provided to the Enc oracle by adversary $\mathcal{A}'$, so game $G_3$ is simulated. If $b = 0$, then the LR oracle always encrypts $\sigma_\mathcal{H}(M)$ for the header $H$ and the message $M$ provided to the Enc oracle by adversary $\mathcal{A}'$, so game $G_4$ is simulated and we get $\Pr[A_{\text{G2n}^{\perp} \Rightarrow 1}] = \Pr[A_{\text{LORS-BCPA}^{\perp} \Rightarrow 1} | b = 1]$ and $\Pr[A_{\text{G2n}^{\leftarrow 1} \Rightarrow 1}] = \Pr[A_{\text{LORS-BCPA}^{\perp} \Rightarrow 0} | b = 0]$. Since $\Pr[b = 0] = \Pr[b = 1] = \frac{1}{2}$, we have:

$$\Pr[A_{\text{LORS-BCPA}^{\perp} \Rightarrow 1}] = \Pr[A_{\text{LORS-BCPA}^{\perp} \Rightarrow 1} | b = 1] \cdot \frac{1}{2} + \Pr[A_{\text{LORS-BCPA}^{\perp} \Rightarrow 1} | b = 0] \cdot \frac{1}{2}$$
from which we can conclude that:

$$\text{Adv}^{\text{D-LORS-BCPA}}_\Pi (A_p) = 2 \cdot \Pr[A_p^{\text{LORS-BCPA}} \Rightarrow 1] - 1$$

$$= \Pr[A_p^{\text{LORS-BCPA}} \Rightarrow 1 | b = 1] + \Pr[A_p^{\text{LORS-BCPA}} \Rightarrow 1 | b = 0] - 1$$

$$= \Pr[A_p^{\text{LORS-BCPA}} \Rightarrow 1 | b = 1] - \Pr[A_p^{\text{LORS-BCPA}} \Rightarrow 0 | b = 0]$$

$$= \Pr[A_G^{\text{3Π}} \Rightarrow 1] - \Pr[A_O^{\text{4Π}} \Rightarrow 1]$$

Recall that for a fixed $K \in \mathcal{K}$ and $H \in \mathcal{H}$, the core encryption algorithm is an online permutation, i.e. $\rho_{K,H} := \text{Core}(E(K,H,\cdot)) \in \text{OPerm}[n]$ (see Subsection 2.2). In game $G_4$, we sample a random $\sigma_H \in \text{OPerm}[n]$ and compute $C = \rho_{K,H} \circ \sigma_H(M)$, whereas in game OAE-IDEAL we sample a random $\pi_H \in \text{OPerm}[n]$ and compute $C = \pi_H(M)$. By Lemma 5 of Appendix A, these two games produce identical distributions and have equivalent complexity, which shows that $\Pr[A_G^{\text{4Π}} \Rightarrow 1] = \Pr[A_O^{\text{4Π}} \Rightarrow 1]$. It follows that

$$\Pr[A_G^{\text{3Π}} \Rightarrow 1] - \Pr[A_O^{\text{4Π}} \Rightarrow 1] = \text{Adv}^{\text{D-LORS-BCPA}}_\Pi (A_p) \leq \text{Adv}^{\text{D-LORS-BCPA}}_\Pi (t',q,b,\mu).$$

4.3 On the minimality of PR-TAG

In the proof of Lemma 4, we did not make full use of the GetTag oracle. Indeed, adversary $A_p$ only queries this oracle to start a new sequential query, but discards the value of the tag. For this reason, a weaker privacy notion than D-LORS-BCPA, one that completely ignores the authentication tag, would be sufficient to prove the equivalence with OAE: we can replace the GetTag oracle in the LORS-BCPA game of Figure 5 by a Reset oracle that starts a new query but does not output the tag. Such a weakening of privacy of core ciphertexts is possible, because privacy of tags is already captured by the PR-TAG notion. This raises a question: can the PR-TAG notion be weakened without breaking the equivalence result, or is it only possible to weaken the privacy of the ciphertext core, but PR-TAG cannot be unchanged? Determining the overlap of the PR-TAG property with both B-INT-CTX and D-LORS-BCPA remains an interesting open question.

This also highlights an interesting observation: to show equivalence with OAE, we need to capture the randomness of the authentication tag with an indistinguishable-from-random type of notion, whereas a left-or-right type of notion is sufficient for the core ciphertext. This stems from the fact that, by definition, the core of an online AE scheme has to be an online permutation for a fixed key and header (otherwise encryption or decryption cannot be computed in an online manner). This, together with the result shown in Appendix A, facilitates the equivalence between games OAE-REAL and $G_4$ used in the proof of Lemma 4.

5 Conclusion

We have shown that, if projected to the setting of deterministic online AE schemes with headers, the blockwise-adaptive security notions of FJMV are equivalent to the notion of OAE in the essentials. We have identified the PR-TAG property, as a missing component that is necessary for the equivalence to be properly proven, and we have observed that this property is not related to the blockwise-adaptiveness of the adversary. We leave the establishment of the exact relations of PR-TAG to the remaining blockwise notions as an open question.
Acknowledgments.

Damian Vizár is supported in part by Microsoft Research under MRL Contract No. 2014-006 (DP1061305). We would like to thank the anonymous reviewers for their constructive comments. We would also like to thank Reza Reyhanitabar for initial discussion leading to this work.

References


A Composition of a random online permutation and a fixed online permutation

Lemma 5. Let \( \rho \) be a random permutation sampled from \( \text{OPERM}[n] \) with any distribution. Let \( \sigma \) be a random permutation independent from \( \rho \) and uniformly distributed in \( \text{OPERM}[n] \) in the sense of Subsection 2.1. Then the composition \( \pi := \rho \circ \sigma \) is uniformly distributed in \( \text{OPERM}[n] \). Besides, the algorithmic complexity to evaluate lazily \( \rho \circ \sigma \) is the same as a lazy evaluation of the equivalent random permutation \( \pi \) uniformly distributed in \( \text{OPERM}[n] \), up to a constant factor.

Proof. Let’s consider the tree representations of these online permutations as described in Subsection 2.1. Let \( T \) be the tree associated to the online permutation \( \rho \circ \sigma \). Each node of \( T \) is labeled with a permutation \( (\rho \circ \sigma)[M] \), where \( M \in B_n^* \) is the input message that leads to this node when one traverses \( T \) from the root.

Given \( M \in B_n^* \), let \( N := \sigma(M) \) and \( P := \rho \circ \sigma(M) \). For all \( M' \in B_n \), let \( N', P' \in B_n \) be defined by \( N' := \sigma(M|M') \) and \( P' := \rho \circ \sigma(M|M') \). Then we have by definition:

\[
N' = \sigma[M|M'] \\
P' = \rho[N](N') \\
P' = (\rho \circ \sigma)[M|M']
\]

This implies that:

\[
(\rho \circ \sigma)[M] = \rho[N] \circ \sigma[M] = \rho[\sigma(M)] \circ \sigma[M]
\]

Consequently, we first have that for all \( M \in B_n^* \), \( (\rho \circ \sigma)[M] \) is uniformly distributed in \( \text{PERM}[n] \). Indeed, \( \sigma[M] \) is uniformly distributed in \( \text{PERM}[n] \) and independent from \( \sigma(M) \), and \( \rho \) is independent from \( \sigma \), so \( \rho[\sigma(M)] \circ \sigma[M] \) is uniformly distributed in \( \text{PERM}[n] \).

Second, the family \( (\rho \circ \sigma)[M]_{M \in B_n^*} \) is a family of independent random permutations. Indeed, each node of \( \rho \circ \sigma \) is labeled with a distinct message \( M \), and for each message \( M \) the permutations \( \rho[\sigma(M)] \) and \( \sigma[M] \) are only used to compute \( (\rho \circ \sigma)[M] \) i.e. they are not used to compute any other \( (\rho \circ \sigma)[M'] \) for \( M' \neq M \). Since the family \( (\sigma[M])_{M \in B_n^*} \) is a family of independent random permutations, the composed family \( (\rho[\sigma(M)] \circ \sigma[M])_{M \in B_n^*} \) is also a family of independent random permutations.

Last, if we let \( \pi := \rho \circ \sigma \), the evaluation of \( \pi \) by lazy sampling of \( \pi \) and the evaluation of \( \rho \circ \sigma \) by lazy sampling of \( \rho \) and \( \sigma \) have the same complexity, up to a constant factor. Indeed, for any message \( M \in B_n^* \), if we let \( l := \text{BLCOUNT}(M) \), to evaluate \( \pi[M] \) we need to sample the permutations \( \pi[e], ..., \pi[M[1...l-1]] \in \text{PERM}[n] \). To evaluate \( (\rho \circ \sigma)(M) \), we need to know the permutations \( (\rho \circ \sigma)[e], ..., (\rho \circ \sigma)[M[1...l-1]] \), and for this we need to sample exactly \( \sigma[e], ..., \sigma[M[1...l-1]] \), as well as \( \rho[\sigma(e)], ..., \rho[\sigma(M[1...l-1])] \). In other words, to evaluate \( \rho \circ \sigma \) on a sequence of messages, we need to explore in the tree of \( \sigma \) the same paths that we explore in the tree of \( \pi \) to evaluate \( \pi \) on those messages, and we need to explore in the tree of \( \rho \) the permutation by \( \sigma \) of these paths. This ensures that the complexity of a lazy sampling is the same in both cases, up to a constant factor. \( \Box \)