

Authenticated Encryption: Relations among Notions and Analysis of the Generic Composition Paradigm

Mihir Bellare and Chanathip Namprempre

Dept. of Computer Science & Engineering, University of California, San Diego, 9500 Gilman Drive,
La Jolla, CA 92093, USA

mihir@cs.ucsd.edu

and

Electrical Engineering, Faculty of Engineering, Thammasat University, Klong Luang, Patumtani 12121,
Thailand

nchanath@engr.tu.ac.th

Communicated by H. Krawczyk

Received 15 July 2007 and revised 24 June 2008

Online publication 30 July 2008

Abstract. An authenticated encryption scheme is a symmetric encryption scheme whose goal is to provide both privacy and integrity. We consider two possible notions of authenticity for such schemes, namely integrity of plaintexts and integrity of ciphertexts, and relate them, when coupled with IND-CPA (indistinguishability under chosen-plaintext attack), to the standard notions of privacy IND-CCA and NM-CPA (indistinguishability under chosen-ciphertext attack and nonmalleability under chosen-plaintext attack) by presenting implications and separations between all notions considered. We then analyze the security of authenticated encryption schemes designed by “generic composition,” meaning making black-box use of a given symmetric encryption scheme and a given MAC. Three composition methods are considered, namely *Encrypt-and-MAC*, *MAC-then-encrypt*, and *Encrypt-then-MAC*. For each of these and for each notion of security, we indicate whether or not the resulting scheme meets the notion in question assuming that the given symmetric encryption scheme is secure against chosen-plaintext attack and the given MAC is unforgeable under chosen-message attack. We provide proofs for the cases where the answer is “yes” and counterexamples for the cases where the answer is “no.”

Key words. Symmetric encryption, Message authentication, Authenticated encryption, Concrete security.

M. Bellare’s work was supported in part by a 1996 Packard Foundation Fellowship in Science and Engineering, NSF CAREER Award CCR-9624439, NSF grants CNS-0524765 and CNS-0627779, and a gift from Intel Corporation.

C. Namprempre’s work was supported in part by grants of the first author and the Thailand Research Fund.

1. Introduction

We use the term *authenticated encryption scheme* to refer to a shared-key based transform whose goal is to provide *both* privacy *and* authenticity of the encapsulated data. In such a scheme, the *encryption* process applied by the sender takes the key and a plaintext to return a ciphertext, while the *decryption* process applied by the receiver takes the same key and a ciphertext to return either a plaintext or a special symbol indicating that it considers the ciphertext invalid or not authentic.

The design of such schemes has attracted a lot of attention historically. Many early schemes, based on adding “redundancy” to the message before applying CBC encryption, were broken. Today authenticated encryption schemes continue to be the target of design and standardization efforts. A popular modern design paradigm is “generic composition,” where a privacy-only symmetric encryption scheme (for example, a block cipher mode of operation like CBC) is combined with a message authentication (MA) scheme (for example, HMAC [8] or CBC-MAC).

The goal of symmetric encryption is usually viewed as privacy, but an authenticated encryption scheme is simply a symmetric encryption scheme meeting additional authenticity goals. The first part of this paper formalizes several different possible notions of authenticity for symmetric encryption schemes and integrates them into the existing mosaic of notions by relating them to the main known notions of privacy for symmetric encryption, via implications and separations in the style initiated in [10]. The second part of this paper analyzes several generic composition methods with regard to meeting the previous notions. Let us now look at these items in more detail.

1.1. Relations among Notions

Privacy goals for symmetric encryption schemes include indistinguishability and non-malleability, each of which can be considered under either chosen-plaintext or (adaptive) chosen-ciphertext attack, leading to four notions of security we abbreviate IND-CPA, IND-CCA, NM-CPA, and NM-CCA. (The original definitions were in the asymmetric setting [24,28,29,45] but can be “lifted” to the symmetric setting using the encryption oracle based template of [9]). The relations among these notions are well understood [10,24,37].

We consider two notions of integrity (we use the terms authenticity and integrity interchangeably) for symmetric encryption schemes. INT-PTXT (integrity of plaintexts) requires that it be computationally infeasible to produce a ciphertext decrypting to a message which the sender had never encrypted, while INT-CTXT (integrity of ciphertexts) requires that it be computationally infeasible to produce a ciphertext not previously produced by the sender, regardless of whether or not the underlying plaintext is “new.” (In both cases, the adversary is allowed a chosen-message attack.) The first of these notions is the more natural security requirement, while the interest of the second, stronger notion is perhaps more in the implications we discuss below.

These notions of authenticity are by themselves quite disjoint from the notions of privacy; for example, sending the message in the clear with an accompanying (strong) MAC achieves INT-CTXT but no kind of privacy. To make for useful comparisons, we consider each notion of authenticity coupled with IND-CPA, the weakest widely-accepted notion of privacy; namely the notions on which we focus for comparison purposes are $\text{INT-PTXT} \wedge \text{IND-CPA}$ and $\text{INT-CTXT} \wedge \text{IND-CPA}$. (Read “ \wedge ” as “and”.)

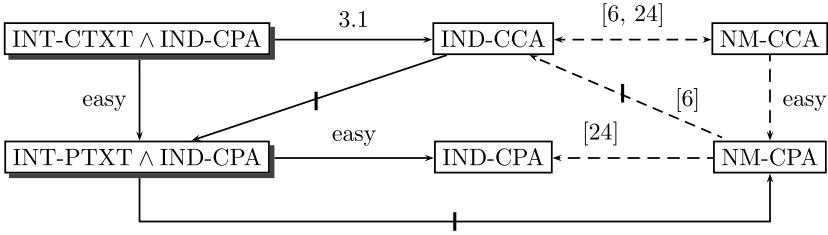


Fig. 1. Relations among notions of symmetric encryption: An arrow denotes an implication, while a barred arrow denotes a separation. The full arrows are relations proved in this paper. Positive results are annotated with the number of the corresponding Theorem. Separation results are derived in Sect. 3. Dotted arrows are reminders of existing relations, annotated with citations to the papers establishing them.

Following the approach of [10] in studying and specifying relative strengths among security notions, Fig. 1 shows the graph of relations between these notions and the above-mentioned older ones. An “implication” $A \rightarrow B$ means that every symmetric encryption scheme meeting notion A also meets notion B . A “separation” $A \not\rightarrow B$ means that there exists a symmetric encryption scheme meeting notion A but not notion B . (This under the minimal assumption that some scheme meeting notion A exists since otherwise the question is moot.) Only a minimal set of relations is explicitly indicated; the relation between any two notions can be derived from the shown ones. (For example, IND-CCA does not imply INT-CTXT \wedge IND-CPA because otherwise, by following arrows, we would get IND-CCA \rightarrow INT-PTXT \wedge IND-CPA contradicting a stated separation.) The dotted lines are reminders of existing relations.

A few points may be worth highlighting. Integrity of ciphertexts—even when coupled only with the weak privacy requirement IND-CPA—emerges as the most powerful notion. Not only does it imply security against chosen-ciphertext attack, but it is strictly stronger than this notion. Nonmalleability—whether under chosen-plaintext or chosen-ciphertext attack—does not imply any type of integrity. The intuitive reason is that nonmalleability only prevents the generation of ciphertexts whose plaintexts are meaningfully related to those of some challenge ciphertexts, while integrity requires it to be hard to generate ciphertexts of new plaintexts even if these are unrelated to plaintexts underlying any existing ciphertexts. Finally, integrity of plaintexts does not imply integrity of ciphertexts.

1.2. Analysis of Generic Composition

There are many possible ways to design authenticated encryption schemes. We focus in this paper on “generic composition”: simply combine a standard symmetric encryption scheme with a MA scheme in some way. There are a few possible ways to do it, and our goal is to analyze and compare their security. The motivation, as we will argue, is that these “obvious” methods, as often the case in practice, remain the most pragmatic from the point of view of performance and security architecture design.

Generic Composition Assume that we are given a symmetric encryption scheme \mathcal{SE} whose encryption and decryption algorithms we denote by \mathcal{E} and \mathcal{D} , respectively. (Typ-

ically this will be a block cipher mode of operation such as CBC or CTR.) Also assume that we are given a message authentication scheme \mathcal{MA} whose tagging and tag-verifying algorithms we denote by \mathcal{T} and \mathcal{V} , respectively. (Possibilities include the CBC-MAC, HMAC [8], or UMAC [20]). We assume that the encryption scheme meets the weakest notion considered to provide an acceptable level of privacy for encryption, namely IND-CPA. This is an appropriate assumption because standard modes of operations such as CBC and CTR do meet the notion (assuming that the underlying block cipher is a secure PRF) [9] but do not meet stronger notions. We assume that the MA scheme meets a notion of unforgeability under chosen message attack. (We will consider both a weak and a strong version of this notion. Standard constructs such as HMAC and CBC-MAC meet both [3,8,11].) We want to “compose” (meaning, appropriately combine) the given encryption and MA schemes to create an authenticated encryption scheme meeting either $\text{INT-CTXT} \wedge \text{IND-CPA}$ or $\text{INT-PTXT} \wedge \text{IND-CPA}$. Below are the composition methods we consider. We call them “generic” because the algorithms of the authenticated encryption scheme appeal to the given ones as black-boxes only. In each case, K_e is a key for encryption, and K_m is a key for message authentication. We stress that these keys are independently chosen. Here $\bar{\mathcal{E}}$ is the encryption algorithm of the authenticated encryption scheme, while the “decrypt+verify” process specifies a decryption algorithm $\bar{\mathcal{D}}$. The latter will either return a plaintext or a special symbol indicating that it considers the ciphertext not authentic.

- *Encrypt-and-MAC (E&M)*: $\bar{\mathcal{E}}(K_e \| K_m, M) = \mathcal{E}(K_e, M) \| \mathcal{T}(K_m, M)$.¹ Namely, encrypt the plaintext and append a MAC of the plaintext. “Decrypt+verify” is performed by first decrypting to get the plaintext and then verifying the tag. The Transport Layer of SSH uses a variant of this method [51].
- *MAC-then-encrypt (MtE)*: $\bar{\mathcal{E}}(K_e \| K_m, M) = \mathcal{E}(K_e, M \| \mathcal{T}(K_m, M))$. Namely, append a MAC to the plaintext and then encrypt them together. “Decrypt+verify” is performed by first decrypting to get the plaintext and candidate tag, and then verifying the tag. SSL uses a variant of this method [26].
- *Encrypt-then-MAC (EtM)*: $\bar{\mathcal{E}}(K_e \| K_m, M) = C \| \mathcal{T}(K_m, C)$, where $C = \mathcal{E}(K_e, M)$. Namely, encrypt the plaintext to get a ciphertext C and append a MAC of C . “Decrypt+verify” is performed by first verifying the tag and then decrypting C . IPSEC uses a variant of this method [38].

Security Results Figure 2 summarizes the security results for the three composite authenticated encryption schemes. (We omit NM-CCA since it is equivalent to IND-CCA.) The top table of the figure shows the results assuming that the base MAC is weakly unforgeable (WUF-CMA), while the bottom table shows the results assuming that the MAC is strongly unforgeable (SUF-CMA). WUF-CMA is the standard notion [11]—it should be computationally infeasible for the adversary to find a message-tag pair in which the message is “new,” even after a chosen-message attack. SUF-CMA requires that it be computationally infeasible for the adversary to find a new message-tag pair

¹ Here (and everywhere in this paper) “||” denotes an operation that combines several strings into one in such a way that the constituent strings are uniquely recoverable from the final one. (If lengths of all strings are fixed and known, concatenation will serve the purpose.)

Composition Method	Privacy			Integrity	
	IND-CPA	IND-CCA	NM-CPA	INT-PTXT	INT-CTXT
<i>Encrypt-and-MAC</i>	insecure	insecure	insecure	secure	insecure
<i>MAC-then-encrypt</i>	secure	insecure	insecure	secure	insecure
<i>Encrypt-then-MAC</i>	secure	insecure	insecure	secure	insecure

Composition Method	Privacy			Integrity	
	IND-CPA	IND-CCA	NM-CPA	INT-PTXT	INT-CTXT
<i>Encrypt-and-MAC</i>	insecure	insecure	insecure	secure	insecure
<i>MAC-then-encrypt</i>	secure	insecure	insecure	secure	insecure
<i>Encrypt-then-MAC</i>	secure	secure	secure	secure	secure

Fig. 2. Summary of security results for the composite authenticated encryption schemes. The given encryption scheme is assumed to be IND-CPA for both tables, while the given MAC is assumed to be weakly unforgeable for the top table and strongly unforgeable for the bottom table.

even after a chosen-message attack. (The message does not have to be new as long as the output tag was not previously attached to this message by the legitimate parties.) This notion seems to have first appeared in [22], albeit in the asymmetric setting. We note that any pseudorandom function is a strongly unforgeable MAC, and most practical MACs seem to be strongly unforgeable. Therefore, analyzing the composition methods under this notion is a realistic and useful approach. Entries in the above tables have the following meaning:

- *Secure*: The composite encryption scheme in question is proven to meet the security requirement in question, assuming only that the base encryption scheme is IND-CPA secure and the base MA scheme is (weakly or strongly, as the case may be) unforgeable under chosen-message attack.
- *Insecure*: There exist *some* IND-CPA secure symmetric encryption and some (WUF-CMA or SUF-CMA) MA scheme such that the composite scheme based on them does not meet the security requirement in question.

In Sect. 4, we justify the above claims.

Why Generic Composition? The use of a generic composition method is advantageous from the point of view both of performance and of security architecture. The performance benefit arises from the presence of fast MACs such as HMAC [8] and UMAC [20]. The architectural benefits arise from the stringent notion of security being used. To be secure, the composition must be secure for *all* possible secure instantiations of its constituent primitives. (If it is secure for some instantiations but not others, we declare it insecure.) An application can thus choose a symmetric encryption scheme and a message authentication scheme independently (these are usually already supported by existing security analyses) and then appeal to some fixed and standard composition technique to combine them. No tailored security analysis of the composed scheme is required.

Interpreting Our Results As we can see from the bottom table of Fig. 2, EtM is secure from all points of view, making it a good choice for a standard. However, if $\text{INT-PTXT} \wedge \text{IND-CPA}$ suffices, then MtE is also appropriate.

We stress that our results are about what happens in general. For example, to say that E&M is IND-CPA insecure means that the composition is insecure for *some* choices of base schemes, not all. There could exist specific choices of base schemes whose E&M composition is in fact IND-CPA secure.

Quantitative Results and Comparisons Above we have discussed our results at a qualitative level. Each result also has a quantitative counterpart; these are what our theorems actually state and prove. These “concrete security” analyses enable a designer to estimate the security of the authenticated encryption scheme in terms of that of its components. All the reductions in this paper are tight, meaning there is little to no loss of security.

1.3. Prior Related Work

The notions IND-CCA and NM-CCA were denoted IND-CCA2 and NM-CCA2, respectively, in [10]. The chosen-ciphertext attacks here are the adaptive kind [45]. Consideration of nonadaptive chosen-ciphertext attacks [43] leads to two more notions, denoted IND-CCA1 and NM-CCA1 by [10], who worked out the relations between six notions of privacy, these two and the four we consider here. (Their results hold for both the asymmetric and the symmetric settings, as mentioned before.) Three additional notions of privacy are considered and related to these six by [37]. In this paper, we have for simplicity avoided consideration of all the possible notions of privacy, focusing instead on what we consider the (four) main ones and their relations to the notions of authenticity.

Authenticity of an encryption scheme has been understood as a goal by designers for many years. The INT-CTXT notion seems to have first appeared in [5,36]. (These two works are concurrent and independent.) Katz and Yung [36] consider two other notions of authenticity not considered here. They also observe the implication $\text{INT-CTXT} \wedge \text{IND-CPA} \rightarrow \text{IND-CCA}$ and present an authenticated encryption scheme called RPC. Signcryption [52] is an asymmetric analog of authenticated encryption.

1.4. Subsequent Related Work

A preliminary version of our paper appeared in 2000 [4]. Subsequent to this, there has been a lot of work on authenticated encryption. We summarize some of it below.

Extensions Rogaway [46] introduces an extension of the notion of authenticated encryption called authenticated encryption with associated data (AEAD). Here the data has two fields, a header and a plaintext. Integrity is required for the whole, but privacy only for the plaintext. Rogaway and Shrimpton [47] explore the problem of cryptographic key transport, referred to as the key wrap problem, and introduce a notion of deterministic authenticated encryption, which they prove to be equivalent to key wrapping.

Generic Composition Canetti and Krawczyk [21] show that EtM implements a “secure channel,” and Krawczyk [40] shows that E&M and MtE in general do not. Krawczyk [40], however, finds some particular instantiations of MtE that do implement secure channels. An, Dodis, and Rabin [2] analyze generic-composition-based signcryption.

Other General Approaches An and Bellare [1] analyze the “encryption with redundancy” paradigm in which one attempts to get an authenticated encryption scheme by adding some redundancy to the plaintext before encrypting. Bellare and Rogaway [5] introduce the “encode-then-encipher” paradigm where an authenticated encryption scheme is obtained by adding randomness and redundancy to the plaintext and then enciphering (rather than encrypting).

E&M in SSH Our results about E&M might make one pessimistic about the security of SSH, which as we said above, is E&M-based. However, SSH in fact uses a variant of E&M, and a direct analysis provided by [13] shows that this variant is in fact secure in most ways. This work also extends ours to allow stateful verification and considers notions of security that require protection against replay attacks.

Dedicated Schemes Dedicated schemes are ones that attempt to directly achieve $\text{IND-CPA} \wedge \text{INT-CTXT}$. These include IACBC [31,32,35], OCB [48], XCBC [27], CCM [50], Helix [25], GCM [42], CWC [39], and EAX [14]. Some of these are more efficient than schemes obtained by generic composition, having effectively the same cost as privacy-only schemes. For a more detailed discussion of dedicated schemes, see [17].

IND-CCA Authenticated encryption is not the only approach to achieving IND-CCA. Direct approaches yielding more compact schemes have been provided by Desai [23].

2. Definitions

Conventions Unless otherwise indicated, an algorithm may be randomized. An adversary is an algorithm. By $y \stackrel{\$}{\leftarrow} A(x_1, x_2, \dots)$ we mean we execute algorithm A with fresh coins on inputs x_1, x_2, \dots , and let y denote the output obtained.

Games Our definitions and proofs will be in the code-based game-playing style of [6]. We recall some background here. A game—look at Fig. 7 for an example—has an **Initialize** procedure, procedures to respond to adversary oracle queries, and a **Finalize** procedure. A game G is executed with an adversary A as follows. First, **Initialize** executes, and its outputs are the inputs to A . Then A executes, its oracle queries being answered by the corresponding procedures of G . When A terminates, its output becomes the input to the **Finalize** procedure. The output of the latter, denoted G^A , is called the output of the game, and we let “ $G^A \Rightarrow y$ ” denote the event that this game output takes value y . Boolean flags are assumed initialized to **false**. Games G_i, G_j are *identical until bad* if their code differs only in statements that follow the setting of **bad** to true. For example, games G_0, G_1 of Figure 7 are identical until bad. The following is the Fundamental Lemma of game-playing of [6].

Lemma 2.1 [6]. *Let G_i, G_j be games that are identical until bad, and A an adversary. Then for any y ,*

$$\Pr[G_i^A \Rightarrow y] - \Pr[G_j^A \Rightarrow y] \leq \Pr[G_j \text{ sets bad}].$$

Concrete Security A security notion is captured by defining a game and a related advantage for an adversary. The understanding is that “secure” means the advantage is “small” for any adversary of “practical” resources. Resources mean running time and number of oracle queries. By convention, running time is that of the execution of the adversary with the game, meaning it includes the time used by the game procedures in this execution. It also includes the size of the adversary description, all this in some fixed RAM computation model. (The description size is included in order to appropriately “charge” adversaries for including large, pre-computed look-up tables in their descriptions.)

This means “secure” has no formal definition. One can provide one by introducing a security parameter and lifting all definitions to an asymptotic setting, but practical symmetric primitives (e.g., block ciphers, cryptographic hash functions) have no security parameter, and so we have preferred to use the concrete setting. Theorems underlying positive results (an implication or a claim that a scheme meets a notion of security under some assumption) are in the concrete style, showing how the advantage and resources of constructed adversaries relate to the original one. Negative results (separations or claims that a composition method fails to preserve a security property) are done less formally. We present the counter-example scheme, which is usually derived from a given underlying scheme, provide the attack, and then state the concrete security result showing that the counter-example scheme retains relevant security attributes of the underlying scheme.

Syntax of Symmetric Encryption Schemes A symmetric encryption scheme $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ consists of three algorithms. The randomized key generation algorithm \mathcal{K} takes no input and returns a key K . The encryption algorithm \mathcal{E} could be randomized or stateful. It takes the key K and a plaintext M to return a ciphertext C . If randomized, it flips coins anew on each invocation. If stateful, it uses and then updates a state that is maintained across invocations. The decryption algorithm \mathcal{D} is deterministic and stateless. It takes the key K and a string C to return either the corresponding plaintext or the symbol \perp ; we write $M \leftarrow \mathcal{D}(K, C)$. We require that $\mathcal{D}(K, \mathcal{E}(K, M)) = M$ with probability one for all M , where the probability is over the choice of K and the coins of \mathcal{E} . (We are assuming that the message space is $\{0, 1\}^*$ for simplicity.) An authenticated encryption scheme is syntactically identical to an encryption scheme as defined above; we will use the term only to emphasize cases where we are targeting authenticity goals.

Privacy of Symmetric Encryption Schemes We use the “left-or-right” version of indistinguishability from [9] but formulate the definitions using games. The **Initialize** procedure of Game $\text{IND-CPA}_{\mathcal{SE}}$ of Fig. 3 picks a random key K and challenge bit b . The adversary A can then query the **LR** oracle with any pair M_0, M_1 of messages of equal length, and the oracle returns an encryption of M_b . (The adversary may query this

<p><u>proc Initialize</u> $K \xleftarrow{\\$} \mathcal{K}; b \xleftarrow{\\$} \{0, 1\}$</p> <p><u>proc LR</u>(M_0, M_1) $C \xleftarrow{\\$} \mathcal{E}(K, M_b); \text{Return } C$</p> <p><u>proc Finalize</u>(d) Return ($d = b$)</p>	<p><u>proc Initialize</u> $K \xleftarrow{\\$} \mathcal{K}; b \xleftarrow{\\$} \{0, 1\}; S \leftarrow \emptyset$</p> <p><u>proc LR</u>($M_0, M_1$) $C \xleftarrow{\\$} \mathcal{E}(K, M_b); S \leftarrow S \cup \{C\}; \text{Return } C$</p> <p><u>proc Dec</u>($C$) If $C \notin S$ then $M \leftarrow \mathcal{D}(K, C)$ else $M \leftarrow \perp$ Return M</p> <p><u>proc Finalize</u>(d) Return ($d = b$)</p>
--	---

Fig. 3. Game IND-CPA $_{\mathcal{SE}}$ (left) and Game IND-CCA $_{\mathcal{SE}}$ (right) where $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$.

oracle multiple times, and each encryption uses fresh coins. If encryption is stateful, the state is maintained by the game.) The game returns true if the adversary's output d equals the challenge bit b , and false otherwise. Let

$$\text{Adv}_{\mathcal{SE}}^{\text{ind-cpa}}(A) = 2 \cdot \Pr[\text{IND-CPA}_{\mathcal{SE}}^A \Rightarrow \text{true}] - 1. \quad (1)$$

Game IND-CCA $_{\mathcal{SE}}$ additionally provides the adversary with oracle **Dec** and augments **LR** to do some bookkeeping. Let

$$\text{Adv}_{\mathcal{SE}}^{\text{ind-cca}}(A) = 2 \cdot \Pr[\text{IND-CCA}_{\mathcal{SE}}^A \Rightarrow \text{true}] - 1. \quad (2)$$

A convention, used throughout this paper, is that the length of a query M_0, M_1 to a left-or-right encryption oracle is defined as $|M_0|$. (This equals $|M_1|$ since the messages must have the same length.) This convention is used in measuring the total length of the queries submitted by the adversary. Note that for IND-CCA, challenge-decryption is disallowed *only* in the second phase [16]. IND-CPA secure symmetric encryption schemes include the CBC and CTR modes of operation [9].

Nonmalleability We will not use definitions of nonmalleability as per [10,24,37] but instead use the equivalent indistinguishability under parallel chosen-ciphertext attack characterization of [7] adapted to the symmetric setting. This facilitates our proofs and analyses and also facilitates concrete security measurements. Game NM-CPA $_{\mathcal{SE}}$ of Fig. 4 provides the adversary with the usual **LR** oracle to which it can query any pair of equal-length messages. It also provides a parallel decryption oracle **Dec**^{*} to which the adversary is allowed *only one* query, this to consist of a vector \vec{C} of ciphertexts. The oracle returns the corresponding plaintext vector. Here and later, the notation $|\vec{X}|$ denotes the number of components of a vector \vec{X} , and $\vec{X}[i]$ denotes the i th component of \vec{X} . Via the flag pdec, the game takes away access to **LR** once A has made its **Dec**^{*} query. However, it continues to have access to the plain encryption oracle **Enc**. For any adversary A , we let

$$\text{Adv}_{\mathcal{SE}}^{\text{nm-cpa}}(A) = 2 \cdot \Pr[\text{NM-CPA}_{\mathcal{SE}}^A \Rightarrow \text{true}] - 1.$$

<p><u>proc Initialize</u> $K \xleftarrow{\\$} \mathcal{K}; b \xleftarrow{\\$} \{0, 1\}; S \leftarrow \emptyset$</p> <p><u>proc LR</u>(M_0, M_1) If pdec then $C \leftarrow \perp$ Else $C \xleftarrow{\\$} \mathcal{E}(K, M_b); S \leftarrow S \cup \{C\}$ Return C</p> <p><u>proc Enc</u>(M) $C \xleftarrow{\\$} \mathcal{E}(K, M);$ Return C</p>	<p><u>proc Dec</u>[*](C^*) pdec \leftarrow true For $i = 1$ to C^* do If $C^*[i] \in S$ then $M^*[i] \leftarrow \perp$ else $M^*[i] \leftarrow \mathcal{D}(K, C^*[i])$ Return M^*</p> <p><u>proc Finalize</u>(d) Return ($d = b$)</p>
--	---

Fig. 4. Game $\text{NM-CPA}_{\mathcal{SE}}$ where $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$.

<p><u>proc Initialize</u> $K \xleftarrow{\\$} \mathcal{K}; S \leftarrow \emptyset$</p> <p><u>proc Enc</u>(M) $C \xleftarrow{\\$} \mathcal{E}(K, M); S \leftarrow S \cup \{M\};$ Return C</p> <p><u>proc VF</u>(C) $M \leftarrow \mathcal{D}(K, C)$ If $M \neq \perp$ and $M \notin S$ then win \leftarrow true Return ($M \neq \perp$)</p> <p><u>proc Finalize</u> Return win</p>	<p><u>proc Initialize</u> $K \xleftarrow{\\$} \mathcal{K}; S \leftarrow \emptyset$</p> <p><u>proc Enc</u>(M) $C \xleftarrow{\\$} \mathcal{E}(K, M); S \leftarrow S \cup \{C\};$ Return C</p> <p><u>proc VF</u>(C) $M \leftarrow \mathcal{D}(K, C)$ If $M \neq \perp$ and $C \notin S$ then win \leftarrow true Return ($M \neq \perp$)</p> <p><u>proc Finalize</u> Return win</p>
---	---

Fig. 5. Game $\text{INT-PTXT}_{\mathcal{SE}}$ (left) and Game $\text{INT-CTXT}_{\mathcal{SE}}$ (right), where $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$.

We do not explicitly define NM-CCA because we do not use it barring its appearance in Fig. 1, and the latter only summarizes known relations about this notion. Briefly, the game additionally provides the adversary with a standard decryption oracle and enforces the usual rule, namely that this oracle returns \perp if queried with a value previously returned by **LR**.

Integrity We specify the notions for integrity (authenticity) of a symmetric encryption scheme using the games of Fig. 5. Adversary A wins in the $\text{INT-PTXT}_{\mathcal{SE}}$ game if it submits to **VF** a ciphertext C whose decryption is a message $M \neq \perp$ not previously queried to **Enc**. It wins the $\text{INT-CTXT}_{\mathcal{SE}}$ game if it submits to **VF** a ciphertext C not previously returned by **Enc**. For any adversary A , we let

$$\text{Adv}_{\mathcal{SE}}^{\text{int-ptxt}}(A) = \Pr[\text{INT-PTXT}_{\mathcal{SE}}^A \Rightarrow \text{true}] \quad \text{and}$$

$$\text{Adv}_{\mathcal{SE}}^{\text{int-ctxt}}(A) = \Pr[\text{INT-CTXT}_{\mathcal{SE}}^A \Rightarrow \text{true}].$$

Syntax of Message Authentication Schemes A message authentication (MA) scheme $\mathcal{MA} = (\mathcal{K}, \mathcal{T}, \mathcal{V})$ consists of three algorithms. The randomized key generation algorithm \mathcal{K} takes no input and returns a key K . The tagging algorithm \mathcal{T} could be either

<pre> <u>proc Initialize</u> $K \xleftarrow{\\$} \mathcal{K}; S \leftarrow \emptyset$ <u>proc Tag(M)</u> $\tau \xleftarrow{\\$} \mathcal{T}(K, M); S \leftarrow S \cup \{M\};$ Return τ <u>proc VF(M, τ)</u> $b \leftarrow \mathcal{V}(K, M, \tau)$ If $b = 1$ and $M \notin S$ then win \leftarrow true Return b <u>proc Finalize</u> Return win </pre>	<pre> <u>proc Initialize</u> $K \xleftarrow{\\$} \mathcal{K}; S \leftarrow \emptyset$ <u>proc Tag(M)</u> $\tau \xleftarrow{\\$} \mathcal{T}(K, M); S \leftarrow S \cup \{(M, \tau)\};$ Return τ <u>proc VF(M, τ)</u> $b \leftarrow \mathcal{V}(K, M, \tau)$ If $b = 1$ and $(M, \tau) \notin S$ then win \leftarrow true Return b <u>proc Finalize</u> Return win </pre>
---	---

Fig. 6. Game $\text{WUF-CMA}_{\mathcal{MA}}$ (left) and $\text{SUF-CMA}_{\mathcal{MA}}$ (right), where $\mathcal{MA} = (\mathcal{K}, \mathcal{T}, \mathcal{V})$.

randomized or stateful. It takes the key K and a message M to return a tag τ . The verification algorithm \mathcal{V} is deterministic and stateless. It takes the key K , a message M , and a candidate tag τ for M to return a bit v . We require that $\mathcal{V}(K, M, \mathcal{T}(K, M)) = 1$ with probability one for all $M \in \{0, 1\}^*$, where the probability is over the choice of K and the coins of \mathcal{T} . We also require that for all K, τ , we have $\mathcal{V}(K, M, \tau) = 0$ if $M = \perp$. The scheme is said to be a MAC if the tagging algorithm is deterministic and stateless and $\mathcal{V}(K, M, \tau)$ returns 1 if and only if $\tau = \mathcal{T}(K, M)$. We sometimes also call the tag τ a MAC.

Unforgeability We specify the security notions of a message authentication scheme using games. Game $\text{WUF-CMA}_{\mathcal{MA}}$ of Fig. 6 captures the standard notion of unforgeability under chosen-message attacks, namely the adaptation of the notion of [30] to the symmetric setting as per [11]. This notion considers the adversary successful if it forges a tag of a message that it did not query to its **Tag** oracle. Game $\text{SUF-CMA}_{\mathcal{MA}}$ captures a stronger notion in which the adversary only needs forge a new message-tag pair to be successful. Note that we allow multiple verification queries. (WUF-CMA is a qualitatively different notion, and SUF-CMA a quantitatively different one, otherwise [12].) For any adversary F , we let

$$\text{Adv}_{\mathcal{MA}}^{\text{wuf-cma}}(F) = \Pr[\text{WUF-CMA}_{\mathcal{MA}}^F \Rightarrow \text{true}] \quad \text{and}$$

$$\text{Adv}_{\mathcal{MA}}^{\text{suf-cma}}(F) = \Pr[\text{SUF-CMA}_{\mathcal{MA}}^F \Rightarrow \text{true}].$$

It is easy to see that SUF-CMA implies WUF-CMA , meaning that any SUF-CMA secure MA scheme is also WUF-CMA secure. There are many practical MACs that are SUF-CMA secure under standard assumptions, for example, HMAC [3,8], CBC-MAC [11,15], EMAC [44], XCBC [18], PMAC [19], TMAC [41], OMAC [33], and CMAC [49]. UMAC [20] and RMAC [34] are randomized WUF-CMA MA schemes.

3. Relations among Notions of Symmetric Encryption

In this section, we detail the results summarized in Fig. 1 and provide proofs. We begin with the implications and then move to the separations.

<pre> proc Initialize Games $G_0, \boxed{G_1}$ 000 $K \xleftarrow{\\$} \mathcal{K}; b \xleftarrow{\\$} \{0, 1\}; S \leftarrow \emptyset$ proc LR(M_0, M_1) 010 $C \xleftarrow{\\$} \mathcal{E}(K, M_b); S \leftarrow S \cup \{C\}; \text{Return } C$ proc Dec(C) 020 If $C \notin S$ then $M \leftarrow \mathcal{D}(K, C)$ else $M \leftarrow \perp$ 021 If $M \neq \perp$ then bad \leftarrow true; $M \leftarrow \perp$ 022 Return M proc Finalize(d) 030 Return ($d = b$) </pre>	<pre> proc Initialize Game G_2 200 $K \xleftarrow{\\$} \mathcal{K}; b \xleftarrow{\\$} \{0, 1\}$ proc LR(M_0, M_1) 210 $C \xleftarrow{\\$} \mathcal{E}(K, M_b); \text{Return } C$ proc Dec(C) 220 Return \perp proc Finalize(d) 230 Return ($d = b$) </pre>
---	---

Fig. 7. Games G_0, G_1 , and G_2 for the proof of Theorem 3.1. Game G_1 contains the code in the box while G_0 does not.

$INT\text{-}CTXT \wedge IND\text{-}CPA \rightarrow IND\text{-}CCA$: The following theorem implies that any encryption scheme that is both IND-CPA secure and INT-CTXT secure is also IND-CCA secure, meaning that weak privacy coupled with strong integrity implies strong privacy.

Theorem 3.1. *Let $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be an encryption scheme. Let A be an ind-cca adversary against \mathcal{SE} running in time t and making q_e **Enc** queries and q_d **Dec** queries. Then, we can construct an int-ctxt adversary A_c and an ind-cpa adversary A_p such that*

$$\mathbf{Adv}_{\mathcal{SE}}^{\text{ind-cca}}(A) \leq 2 \cdot \mathbf{Adv}_{\mathcal{SE}}^{\text{int-ctxt}}(A_c) + \mathbf{Adv}_{\mathcal{SE}}^{\text{ind-cpa}}(A_p), \quad (3)$$

and A_c runs in time $O(t)$ and makes q_e **Enc** queries and q_d **VF** queries, while A_p runs in time $O(t)$ and makes q_e **LR** queries.

Proof. Consider games G_0, G_1, G_2 of Fig. 7. We have

$$\begin{aligned} \Pr[\text{IND-CCA}_{\mathcal{SE}}^A \Rightarrow \text{true}] &= \Pr[G_0^A \Rightarrow \text{true}] \\ &= \Pr[G_1^A \Rightarrow \text{true}] + (\Pr[G_0^A \Rightarrow \text{true}] - \Pr[G_1^A \Rightarrow \text{true}]) \\ &\leq \Pr[G_1^A \Rightarrow \text{true}] + \Pr[G_1^A \text{ sets bad}], \end{aligned} \quad (4)$$

where the last equation follows from Lemma 2.1 because G_0, G_1 are identical until **bad**. Now, notice that in the procedure **Dec** of G_1 , the returned value is always \perp . Therefore,

$$\Pr[G_1^A \Rightarrow \text{true}] = \Pr[G_2^A \Rightarrow \text{true}]. \quad (5)$$

We will design A_c and A_p so that

$$\Pr[G_1^A \text{ sets bad}] \leq \Pr[\text{INT-CTXT}_{\mathcal{SE}}^{A_c} \Rightarrow \text{true}] \quad \text{and} \quad (6)$$

$$\Pr[G_2^A \Rightarrow \text{true}] \leq \Pr[\text{IND-CPA}_{\mathcal{SE}}^{A_p} \Rightarrow \text{true}]. \quad (7)$$

Then, Equation (3) follows from (1), (2), and (4)–(7). We now describe A_c and A_p .

Adversary A_c starts by picking a random bit b and then runs A answering its queries as follows. For an **LR** query M_0, M_1 , adversary A_c submits M_b to its **Enc** oracle and returns the response to A . For a **Dec** query C , adversary A_c submits C to its **VF** oracle and (regardless of the response) returns \perp to A .

We define A_p as follows. It simply runs A answering A 's **LR** queries using its own **LR** oracle and answering A 's **Dec** queries with \perp . It outputs whatever A outputs. \square

Separation Approach To show that a security notion **A** does not imply a security notion **B**, we construct a scheme $\overline{\mathcal{SE}}$ that meets notion **A** but for which we can exhibit an attack showing that it does not meet notion **B**. Of course, the statement that $\mathbf{A} \not\rightarrow \mathbf{B}$ is vacuously and uninterestingly true if there does not exist any scheme secure under the notion **A** in the first place. So we make the minimal assumption whenever we show a separation $\mathbf{A} \not\rightarrow \mathbf{B}$ that there exists some scheme secure under the notion **A**, and obtain $\overline{\mathcal{SE}}$ by modifying this given scheme. We note that the scheme $\overline{\mathcal{SE}}$ may be artificial. But the point we are making is that it is not possible to prove $\mathbf{A} \rightarrow \mathbf{B}$, and even an artificial example is enough for that.

IND-CCA $\not\rightarrow$ INT-PTXT: Let $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be a given IND-CCA secure symmetric encryption scheme. We define a scheme \mathcal{SE}_1 such that \mathcal{SE}_1 is IND-CCA secure but is not INT-PTXT secure. The idea is simple. A certain known string (or strings) will be viewed by $\overline{\mathcal{D}}$ as valid and decrypted to certain known messages, so that forgery is easy. But these ‘‘ciphertexts’’ will never be produced by the encryption algorithm, so privacy will not be affected. Here are the details. The new scheme $\mathcal{SE}_1 = (\mathcal{K}, \mathcal{E}_1, \mathcal{D}_1)$ has the same key generation algorithm as the old scheme and the following modified encryption and decryption algorithms:

$$\begin{array}{l|l} \text{Algorithm } \mathcal{E}_1(K, M) & \text{Algorithm } \mathcal{D}_1(K, C) \\ C' \xleftarrow{\$} \mathcal{E}(K, M) & \text{Parse } C \text{ as } b\|C' \text{ where } b \text{ is a bit} \\ C \leftarrow 0\|C' & \text{If } b = 0 \text{ then } M \leftarrow \mathcal{D}(K, C') \text{ else } M \leftarrow 0 \\ \text{Return } C & \text{Return } M \end{array}$$

We present an attack on \mathcal{SE}_1 in the form of an adversary A who defeats the integrity of plaintexts (meaning, wins Game $\text{INT-PTXT}_{\mathcal{SE}_1}$) with probability one using only one query to the verification oracle. All A does is submit the string 10 to its **VF** oracle. We observe that $\overline{\mathcal{D}}(K, 10) = 0$, meaning 10 is a valid ciphertext, and it decrypts to a message (namely 0) that the adversary has not queried of its oracle. So

$$\text{Adv}_{\mathcal{SE}_1}^{\text{int-ptxt}}(A) = 1.$$

To prove that \mathcal{SE}_1 is IND-CCA secure, we show that given any adversary A attacking \mathcal{SE}_1 using time t and making q_e **LR** queries and q_d **Dec** queries, there is an adversary B attacking \mathcal{SE} such that

$$\text{Adv}_{\mathcal{SE}_1}^{\text{ind-cca}}(A) \leq \text{Adv}_{\mathcal{SE}}^{\text{ind-cca}}(B)$$

and B runs in time $O(t)$ and makes q_e **LR** queries and q_d **Dec** queries. Adversary B works as follows:

Adversary B

Run A

On query $\mathbf{LR}(M_0, M_1)$

$C \xleftarrow{\$} 0 \parallel \mathbf{LR}(M_0, M_1)$; Return C to A

On query $\mathbf{Dec}(C)$

Parse C as $b \parallel C'$ where b is a bit

If $b = 0$ then $M \leftarrow \mathbf{Dec}(C')$ else $M \leftarrow 0$

Return M to A

Until A halts and outputs a bit b

Return b

Above, B is using its own \mathbf{LR} oracle to answer calls that A makes to its \mathbf{LR} oracle.

$INT\text{-}PTXT \wedge IND\text{-}CPA \not\rightarrow NM\text{-}CPA$: We can easily show this separation as usual by starting with an encryption that is both $INT\text{-}PTXT$ and $IND\text{-}CPA$ secure and modifying it so that it retains security under $INT\text{-}PTXT \wedge IND\text{-}CPA$ but is insecure under $NM\text{-}CPA$. However, notice from Fig. 2 that via the MtE composition method, one can construct a scheme that is $INT\text{-}PTXT \wedge IND\text{-}CPA$ but not $NM\text{-}CPA$, and thus this separation result holds.

4. Security of the Composite Schemes

We now detail the results for the composite schemes as summarized in Fig. 2. Throughout this section, $\mathcal{SE} = (\mathcal{K}_e, \mathcal{E}, \mathcal{D})$ is a given symmetric encryption scheme which is $IND\text{-}CPA$ secure, and $\mathcal{MA} = (\mathcal{K}_m, \mathcal{T}, \mathcal{V})$ is a given message authentication scheme which is $WUF\text{-}CMA$ or $SUF\text{-}CMA$ secure. We refer to these as the base schemes. Associated to them is $\overline{\mathcal{SE}} = (\overline{\mathcal{K}}, \overline{\mathcal{E}}, \overline{\mathcal{D}})$, an encryption scheme constructed according to one of the three methods we are considering. The presentation below is method by method, and in each case, we begin by specifying the method in more detail. We first provide theorems proving positive results for the method then either describe the counter-examples for negative results or explain how the implications can be derived from already proved results in combination with results from Sect. 3.

In presenting a counter-example (meaning a claim that a certain composition method is insecure under some notion of security \mathbf{A}), we use the following paradigm. We present a symmetric encryption scheme \mathcal{SE}' and a MAC \mathcal{MA}' such that \mathcal{SE}' is $IND\text{-}CPA$ secure and \mathcal{MA}' is $WUF\text{-}CMA$ or $SUF\text{-}CMA$ secure but we can present an attack on the composite scheme based on them showing that the composite scheme does not meet notion \mathbf{A} . Of course, we make the minimal assumptions that there exist some scheme \mathcal{SE} that is $IND\text{-}CPA$ secure and some scheme \mathcal{MA} that is $WUF\text{-}CMA$ or $SUF\text{-}CMA$ secure, since otherwise the claim is vacuous. We construct \mathcal{SE}' from \mathcal{SE} and \mathcal{MA}' from \mathcal{MA} . Whenever possible, we present the counter-example for the case in which \mathcal{MA}' is $SUF\text{-}CMA$ because the negative result then follows for the case in which \mathcal{MA}' is $WUF\text{-}CMA$. (If the composite scheme does not meet notion \mathbf{A} when \mathcal{MA}' is $SUF\text{-}CMA$, then it cannot hope to do so when \mathcal{MA}' is $WUF\text{-}CMA$.)

In some cases, the constructions are artificial. But what we want to assess is whether it is possible to prove that the composite scheme meets notion **A** assuming *only* that the constituent encryption scheme is IND-CPA secure and that the constituent MA scheme is WUF-CMA or SUF-CMA secure, and a result of the type just explained shows that such a proof is not possible.

4.1. *Encrypt-and-MAC (E&M)*

The *Encrypt-and-MAC* (E&M) composition of base schemes $\mathcal{SE} = (\mathcal{K}_e, \mathcal{E}, \mathcal{D})$ and $\mathcal{MA} = (\mathcal{K}_m, \mathcal{T}, \mathcal{V})$ is the encryption scheme $\overline{\mathcal{SE}} = (\overline{\mathcal{K}}, \overline{\mathcal{E}}, \overline{\mathcal{D}})$ whose constituent algorithms are as follows:

Algorithm $\overline{\mathcal{K}}$ $K_e \xleftarrow{\$} \mathcal{K}_e$ $K_m \xleftarrow{\$} \mathcal{K}_m$ Return $K_e \ K_m$	Algorithm $\overline{\mathcal{E}}(K_e \ K_m, M)$ $C' \xleftarrow{\$} \mathcal{E}(K_e, M)$ $\tau \xleftarrow{\$} \mathcal{T}(K_m, M)$ $C \leftarrow C' \ \tau$ Return C	Algorithm $\overline{\mathcal{D}}(K_e \ K_m, C)$ Parse C as $C' \ \tau$ $M \leftarrow \mathcal{D}(K_e, C')$ $v \leftarrow \mathcal{V}(K_m, M, \tau)$ If $v = 1$ then return M else return \perp .
---	--	---

E&M provides INT-PTXT: E&M does provide integrity of plaintexts. It inherits the integrity of the MAC in a direct way, with no degradation in security. This is independent of the symmetric encryption scheme: whether the latter is secure or not does not affect the integrity of the composite scheme.

Theorem 4.1. *Let $\mathcal{SE} = (\mathcal{K}_e, \mathcal{E}, \mathcal{D})$ be a symmetric encryption scheme, let $\mathcal{MA} = (\mathcal{K}_m, \mathcal{T}, \mathcal{V})$ be a message authentication scheme, and let $\overline{\mathcal{SE}} = (\overline{\mathcal{K}}, \overline{\mathcal{E}}, \overline{\mathcal{D}})$ be the encryption scheme obtained from \mathcal{SE} and \mathcal{MA} via the *Encrypt-and-MAC* composition method. Given any adversary A against $\overline{\mathcal{SE}}$, we can construct an adversary F such that*

$$\mathbf{Adv}_{\overline{\mathcal{SE}}}^{\text{int-ptxt}}(A) \leq \mathbf{Adv}_{\mathcal{MA}}^{\text{wuf-cma}}(F), \quad (8)$$

and F makes the same numbers of **Tag** and **VF** queries as A does **Enc** and **VF**, respectively.

Proof. Adversary F works as follows:

Adversary F

$K_e \xleftarrow{\$} \mathcal{K}_e$

Run A

On query **Enc**(M)

$C' \xleftarrow{\$} \mathcal{E}(K_e, M)$; $\tau \xleftarrow{\$} \mathbf{Tag}(M)$; Return $C' \| \tau$ to A

On query **VF**(C)

Parse C as $C' \| \tau$; $M \leftarrow \mathcal{D}(K_e, C')$; $v \leftarrow \mathbf{VF}(M, \tau)$; Return v to A

Until A halts

Consider a ciphertext $C = C' \| \tau$ that yields a successful forgery of a new plaintext M . This means that M was never queried to **Enc**, which implies that F never queried it to

Tag either. Therefore, the pair (M, τ) is a valid weak forgery, and (8) is justified. To justify the claims about the resource parameters used by F , we note that, as per our conventions, the resources for both adversaries include the running time of the game procedures. \square

E&M Does Not Provide IND-CPA: E&M does not preserve privacy because the MAC could reveal information about the plaintext. This is true regardless of whether the MAC is weakly or strongly unforgeable. We can certainly use this idea to show that E&M does not provide IND-CPA by starting with a (weak or strong) unforgeable MA scheme then modifying it so that it deliberately leaks some bits of the plaintext through the tag (doing so will not affect unforgeability), but clearly the resulting composite scheme will not be IND-CPA secure if the modified MA scheme is used as a base scheme.

However, we will prove a stronger statement here. Not only do there exist schemes for which the E&M method fails to provide IND-CPA, but it will fail to be so for most of the commonly defined MA schemes, including CBC-MAC and HMAC, because the latter are MACs. Indeed, an adversary can use the MAC present in the ciphertext of the composite scheme to see whether the same message has been encrypted twice, something which should not be possible if the scheme is to meet a strong notion of privacy like IND-CPA. This attack is successful regardless of whether the underlying MAC is weakly or strongly unforgeable. Here are the details. Assuming that \mathcal{MA} is a MAC, the IND-CPA attack is as follows. Adversary A picks distinct, equal-length messages x, y . It makes **LR** query x, y to get back a ciphertext $C_1 = C'_1 \parallel \tau_1$ and then makes **LR** query x, x to get back a ciphertext $C_2 = C'_2 \parallel \tau_2$. If $\tau_1 = \tau_2$, it returns 0, else it returns 1. Then,

$$\text{Adv}_{\mathcal{SE}}^{\text{ind-cpa}}(A) \geq 1 - \text{Adv}_{\mathcal{MA}}^{\text{wuf-cma}}(F),$$

where F is the following adversary: it makes query x to **Tag** to get τ and then makes query (y, τ) to **VF**. The WUF-CMA security of \mathcal{MA} implies that F has low advantage, so A has high advantage.

E&M Provides Neither IND-CCA nor NM-CPA: Since both IND-CCA and NM-CPA imply IND-CPA, the above means that E&M provides *neither* IND-CCA *nor* NM-CPA secure.

E&M Does Not Provide INT-CTXT: E&M also fails to provide integrity of ciphertexts. This is because there are secure encryption schemes with the property that a ciphertext can be modified without changing its decryption. When such an encryption scheme is used as the base symmetric encryption scheme, an adversary can query the encryption oracle, modify part of the response, and still submit the result to the verification oracle as a valid ciphertext. We provide details below.

Let $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be the given IND-CPA secure symmetric encryption scheme. We define a scheme \mathcal{SE}_2 such that \mathcal{SE}_2 is IND-CPA secure but is not INT-CTXT secure. The idea is to prepend a redundant bit to ciphertexts. This bit is ignored by the decryption algorithm, resulting in the ability to create two different ciphertexts of the same message, which in turn defeats the integrity of ciphertexts. Here are the details.

The new scheme $\mathcal{SE}_2 = (\mathcal{K}, \mathcal{E}_2, \mathcal{D}_2)$ has the same key generation algorithm as the old scheme and the following modified encryption and decryption algorithms:

$$\begin{array}{l|l} \text{Algorithm } \mathcal{E}_2(K, M) & \text{Algorithm } \mathcal{D}_2(K, C) \\ C \xleftarrow{\$} \mathcal{E}(K, M) & \text{Parse } C \text{ as } b\|C' \text{ where } b \text{ is a bit} \\ \text{Return } 0\|C & M \leftarrow \mathcal{D}(K, C'); \text{ return } M \end{array}$$

We claim that \mathcal{SE}_2 is IND-CPA secure but not INT-CTXT secure. The proof that \mathcal{SE}_2 is still IND-CPA secure is simple and is omitted. We now show that \mathcal{SE}_2 is not INT-CTXT secure through the following attack that has advantage one:

Adversary A

$$\bar{C} \xleftarrow{\$} \mathbf{Enc}(0); \text{ Parse } \bar{C} \text{ as } 0\|C; \mathbf{VF}(1\|C)$$

Note that this attack does not violate integrity of plaintexts because the plaintexts underlying ciphertexts $0\|C$ and $1\|C$ are the same.

4.2. MAC-then-Encrypt (MtE)

The *MAC-then-encrypt* (MtE) composition of base schemes $\mathcal{SE} = (\mathcal{K}_e, \mathcal{E}, \mathcal{D})$ and $\mathcal{MA} = (\mathcal{K}_m, \mathcal{T}, \mathcal{V})$ is the encryption scheme $\overline{\mathcal{SE}} = (\overline{\mathcal{K}}, \overline{\mathcal{E}}, \overline{\mathcal{D}})$ whose constituent algorithms are as follows:

$$\begin{array}{l|l|l} \text{Algorithm } \overline{\mathcal{K}} & \text{Algorithm } \overline{\mathcal{E}}(K_e\|K_m, M) & \text{Algorithm } \overline{\mathcal{D}}(K_e\|K_m, C) \\ K_e \xleftarrow{\$} \mathcal{K}_e & \tau \xleftarrow{\$} \mathcal{T}(K_m, M) & M' \leftarrow \mathcal{D}(K_e, C) \\ K_m \xleftarrow{\$} \mathcal{K}_m & C \xleftarrow{\$} \mathcal{E}(K_e, M\|\tau) & \text{Parse } M' \text{ as } M\|\tau \\ \text{Return } K_e\|K_m & \text{Return } C & v \leftarrow \mathcal{V}(K_m, M, \tau) \\ & & \text{If } v = 1 \text{ then return } M \text{ else return } \perp. \end{array}$$

MtE Provides INT-PTXT and IND-CPA: MtE provides both privacy against chosen-plaintext attack and integrity of plaintexts. More precisely, if the underlying MA scheme is WUF-CMA secure, then the composite scheme is INT-PTXT secure. Furthermore, if the underlying encryption scheme is IND-CPA secure, then so is the composite scheme.

Theorem 4.2. *Let $\mathcal{SE} = (\mathcal{K}_e, \mathcal{E}, \mathcal{D})$ be a symmetric encryption scheme, let $\mathcal{MA} = (\mathcal{K}_m, \mathcal{T}, \mathcal{V})$ be a message authentication scheme, and let $\overline{\mathcal{SE}} = (\overline{\mathcal{K}}, \overline{\mathcal{E}}, \overline{\mathcal{D}})$ be the encryption scheme obtained from \mathcal{SE} and \mathcal{MA} via the MAC-then-encrypt composition method. Given any adversary I against $\overline{\mathcal{SE}}$, we can construct an adversary F such that*

$$\mathbf{Adv}_{\overline{\mathcal{SE}}}^{\text{int-ptxt}}(I) \leq \mathbf{Adv}_{\mathcal{MA}}^{\text{wuf-cma}}(F), \quad (9)$$

and F makes the same numbers of **Tag** and **VF** queries as I does **Enc** and **VF**, respectively. Similarly, given any adversary A against $\overline{\mathcal{SE}}$, we can construct an adversary A_p such that

$$\mathbf{Adv}_{\overline{\mathcal{SE}}}^{\text{ind-cpa}}(A) \leq \mathbf{Adv}_{\mathcal{SE}}^{\text{ind-cpa}}(A_p), \quad (10)$$

and A_p uses the same resources as A , except that each encryption query of A_p is l bits longer than that of A , where l is the length of a tag in the scheme \mathcal{MA} .

Proof. We construct F and A_p as follows:

Adversary F

$K_e \xleftarrow{\$} \mathcal{K}_e$
 Run I
 On query $\mathbf{Enc}(M)$
 $M' \xleftarrow{\$} M \parallel \mathbf{Tag}(M)$; $C' \xleftarrow{\$} \mathcal{E}(K_e, M')$
 Return C' to I
 On query $\mathbf{VF}(C)$
 $M' \leftarrow \mathcal{D}(K_e, C)$; Parse M' as $M \parallel \tau$
 $v \leftarrow \mathbf{VF}(M, \tau)$; Return v to I
 Until I halts

Adversary A_p

$K_m \xleftarrow{\$} \mathcal{K}_m$
 Run A
 On query $\mathbf{LR}(M_0, M_1)$
 $\tau_0 \xleftarrow{\$} \mathcal{T}(K_m, M_0)$; $\tau_1 \xleftarrow{\$} \mathcal{T}(K_m, M_1)$
 $M'_0 \leftarrow M_0 \parallel \tau_0$; $M'_1 \leftarrow M_1 \parallel \tau_1$
 $C \xleftarrow{\$} \mathbf{LR}(M'_0, M'_1)$; Return C to A
 Until A halts and returns b
 Return b

It is easy to see that (9) and (10) follow. The claimed resource usage of F and A_p can also be easily justified. \square

MtE Does Not Provide NM-CPA: The base encryption scheme might be malleable, and this will be inherited by the composite scheme. Here are the details. Let $\mathcal{SE} = (\mathcal{K}_e, \mathcal{E}, \mathcal{D})$ be the given IND-CPA secure symmetric encryption scheme, and let $\mathcal{SE}_2 = (\mathcal{K}, \mathcal{E}_2, \mathcal{D}_2)$ be the scheme as defined towards the end of Sect. 4.1, namely a scheme that is based on \mathcal{SE} but that prepends a redundant bit which is ignored by the decryption algorithm. We already noted that, if \mathcal{SE} is IND-CPA secure, then so is \mathcal{SE}_2 . Let $\overline{\mathcal{SE}}$ be the scheme obtained by MtE based on \mathcal{SE}_2 and \mathcal{MA} . We show that $\overline{\mathcal{SE}}$ is not NM-CPA secure by presenting an attack in the form of an adversary A who violates its nonmalleability (meaning, wins Game $\text{NM-CPA}_{\overline{\mathcal{SE}}}$) with probability one. It works as follows:

Adversary A

$C \xleftarrow{\$} \mathbf{LR}(0, 1)$; Parse C as $x \parallel C'$ where x is a bit
 $\vec{C}[1] \leftarrow 1 \parallel C'$; $\vec{P} \leftarrow \mathbf{Dec}^*(\vec{C})$; Return $\vec{P}[1]$.

MtE Provides Neither IND-CCA nor INT-CTXT Since IND-CCA implies NM-CPA, MtE does *not* provide IND-CCA security either. Furthermore, the fact that it provides IND-CPA security but not NM-CPA security implies that it does not provide INT-CTXT security.

4.3. Encrypt-then-MAC (EtM)

The *Encrypt-then-MAC* (EtM) composition of base schemes $\mathcal{SE} = (\mathcal{K}_e, \mathcal{E}, \mathcal{D})$ and $\mathcal{MA} = (\mathcal{K}_m, \mathcal{T}, \mathcal{V})$ is the encryption scheme $\overline{\mathcal{SE}} = (\overline{\mathcal{K}}, \overline{\mathcal{E}}, \overline{\mathcal{D}})$ whose constituent algorithms are as follows:

Algorithm $\overline{\mathcal{K}}$

$K_e \xleftarrow{\$} \mathcal{K}_e$
 $K_m \xleftarrow{\$} \mathcal{K}_m$
 Return $K_e \parallel K_m$

Algorithm $\overline{\mathcal{E}}(K_e \parallel K_m, M)$

$C' \xleftarrow{\$} \mathcal{E}(K_e, M)$
 $\tau' \xleftarrow{\$} \mathcal{T}(K_m, C')$
 $C \leftarrow C' \parallel \tau'$
 Return C

Algorithm $\overline{\mathcal{D}}(K_e \parallel K_m, C)$

Parse C as $C' \parallel \tau'$
 $M \leftarrow \mathcal{D}(K_e, C')$
 $v \leftarrow \mathcal{V}(K_m, C', \tau')$
 If $v = 1$ then return M else return \perp .

The security results for the two composition methods we have covered so far, i.e., E&M and MtE, hold whether or not we assume the base MAC scheme to be weakly or strongly unforgeable. For EtM, however, we have different security results depending on our assumption about the MAC, as indicated in the two tables in Fig. 2. For clarity, we separate the results accordingly here.

EtM with WUF-CMA Base MAC Provides IND-CPA and INT-PTXT: The following theorem implies that EtM inherits the IND-CPA security of the base encryption scheme and is INT-PTXT secure if the base MA scheme is WUF-CMA secure.

Theorem 4.3. *Let $\mathcal{SE}(\mathcal{K}_e, \mathcal{E}, \mathcal{D})$ be a symmetric encryption scheme, let $\mathcal{MA} = (\mathcal{K}_m, \mathcal{T}, \mathcal{V})$ be a message authentication scheme, and let $\overline{\mathcal{SE}} = (\overline{\mathcal{K}}, \overline{\mathcal{E}}, \overline{\mathcal{D}})$ be the authenticated encryption scheme obtained from \mathcal{SE} and \mathcal{MA} via the encrypt-then-MAC composition method. Given any adversary A against $\overline{\mathcal{SE}}$, we can construct an adversary A_p such that*

$$\mathbf{Adv}_{\overline{\mathcal{SE}}}^{\text{ind-cpa}}(A) \leq \mathbf{Adv}_{\mathcal{SE}}^{\text{ind-cpa}}(A_p), \quad (11)$$

and A_p makes the same number of queries as A does. Similarly, given any adversary I against $\overline{\mathcal{SE}}$, we can construct an adversary F such that

$$\mathbf{Adv}_{\overline{\mathcal{SE}}}^{\text{int-ptxt}}(I) \leq \mathbf{Adv}_{\mathcal{MA}}^{\text{wuf-cma}}(F), \quad (12)$$

and F makes the same numbers of **Tag** and **VF** queries as I does **Enc** and **VF** queries, respectively, except that each **Tag** query of F is l bits longer than that of I , where l is the difference in bits of the length of a ciphertext and that of a plaintext in the scheme \mathcal{SE} .

Proof. We construct A_p and F as follows:

Adversary A_p $K_m \xleftarrow{\$} \mathcal{K}_m$ Run A On query LR (M_0, M_1) $C \xleftarrow{\$} \mathbf{LR}(M_0, M_1)$ $\tau \xleftarrow{\$} \mathcal{T}(K_m, C)$ Return $C \parallel \tau$ to A Until A halts and returns b Return b	Adversary F $K_e \xleftarrow{\$} \mathcal{K}_e$ Run I On query Enc (M) $C' \xleftarrow{\$} \mathcal{E}(K_e, M)$; $\tau \xleftarrow{\$} \mathbf{Tag}(C')$; Return $C' \parallel \tau$ to I On query VF (C) Parse C as $C' \parallel \tau'$; $v \leftarrow \mathbf{VF}(C', \tau')$; Return v to I Until I halts
--	--

Equation (11) is easily verified. For Equation (12), let $C = C' \parallel \tau'$ be a **VF** query of I that leads to its winning game $\text{INT-PTXT}_{\overline{\mathcal{SE}}}$. Let $M = \mathcal{D}(K_e, C')$. The unique decryptability of \mathcal{SE} means that if M was not an **Enc** query of I , then C' was not a **Tag** query of F . So F wins $\text{WUF-CMA}_{\mathcal{MA}}$ whenever I wins $\text{INT-PTXT}_{\overline{\mathcal{SE}}}$. \square

EtM with WUF-CMA Base MAC Does Not Provide NM-CPA: However, a weakly unforgeable base MA scheme is not enough to obtain a NM-CPA secure composite scheme under EtM. To illustrate this, let $\mathcal{MA} = (\mathcal{K}, \mathcal{T}, \mathcal{V})$ be the given WUF-CMA secure MA scheme. We define a WUF-CMA MA scheme \mathcal{MA}' such that the composite scheme $\overline{\mathcal{SE}}$ formed by EtM based on \mathcal{SE} and \mathcal{MA}' is not NM-CPA secure. The idea is that a redundant bit is appended to the tag by the tagging algorithm and ignored by the verification algorithm. Here are the details. The new MA scheme $\mathcal{MA}' = (\mathcal{K}, \mathcal{T}', \mathcal{V}')$ has the same key generation algorithm as that of the original scheme, but its tagging and verifying algorithms are as follows:

$$\begin{array}{l|l} \text{Algorithm } \mathcal{T}'(K, M) & \text{Algorithm } \mathcal{V}'(K, M, \tau) \\ \tau \xleftarrow{\$} \mathcal{T}(K, M) & \text{Parse } \tau \text{ as } \tau' \| b \text{ where } b \text{ is a bit.} \\ \text{Return } \tau \| 0 & \text{Return } \mathcal{V}(K, M, \tau') \end{array}$$

To prove that the composite scheme $\overline{\mathcal{SE}}$ constructed from \mathcal{SE} and \mathcal{MA}' using EtM method is not NM-CPA secure, we present an attack on $\overline{\mathcal{SE}}$ in the form of an adversary A who violates the nonmalleability of \mathcal{SE} (meaning, wins Game $\text{NM-CPA}_{\overline{\mathcal{SE}}}$) with probability one. It works as follows:

Adversary A

$$\begin{array}{l} C \xleftarrow{\$} \mathbf{LR}(0, 1); \text{ Parse } C \text{ as } C' \| \tau; \text{ Parse } \tau \text{ as } \tau' \| x \text{ where } x \text{ is a bit} \\ \tau \leftarrow \tau' \| 1; \vec{C}[1] \leftarrow C' \| \tau; \vec{P} \leftarrow \mathbf{Dec}^*(\vec{C}); \text{ Return } \vec{P}[1] \end{array}$$

The proof that \mathcal{MA}' is WUF-CMA secure is easy and is omitted.

EtM with WUF-CMA Base MAC Provides Neither IND-CCA nor INT-CTXT: Suppose that the base MA scheme is only assumed to be weakly unforgeable. Since IND-CCA implies NM-CPA, EtM does not provide IND-CCA security. Also, since $\text{INT-CTXT} \wedge \text{IND-CPA}$ implies NM-CPA and since EtM provides IND-CPA, EtM does not provide INT-CTXT security.

EtM with SUF-CMA Base MAC Provides INT-CTXT: The following theorem states that EtM provides IND-CCA and INT-CTXT security assuming a strongly unforgeable base MA scheme.

Theorem 4.4. *Let $\mathcal{SE} = (\mathcal{K}_e, \mathcal{E}, \mathcal{D})$ be a symmetric encryption scheme, let $\mathcal{MA} = (\mathcal{K}_m, \mathcal{T}, \mathcal{V})$ be a message authentication scheme, and let $\overline{\mathcal{SE}} = (\overline{\mathcal{K}}, \overline{\mathcal{E}}, \overline{\mathcal{D}})$ be the authenticated encryption scheme obtained from \mathcal{SE} and \mathcal{MA} via the encrypt-then-MAC composition method. Given any adversary I , we can construct an adversary F such that*

$$\mathbf{Adv}_{\overline{\mathcal{SE}}}^{\text{int-ctxt}}(I) \leq \mathbf{Adv}_{\mathcal{MA}}^{\text{suf-cma}}(F), \tag{13}$$

and F makes the same numbers of **Tag** and **VF** queries as I does **Enc** and **VF** queries, respectively, except that each **Tag** query of F is l bits longer than that of I , where l is the difference in bits of the length of a ciphertext and that of a plaintext in the scheme \mathcal{SE} .

Proof. Adversary F is the same as the one described in the proof of (12) in Theorem 4.3. Let $C = C' \parallel \tau'$ be a **VF** query of I that leads to its winning game $\text{INT-CTXT}_{\overline{SE}}$. If C was not returned to I by **Enc**, then F did not query **Tag** with C' . So F was $\text{SUF-CMA}_{\mathcal{M}, \mathcal{A}}$ whenever I wins $\text{INT-CTXT}_{\overline{SE}}$. \square

EtM with SUF-CMA Base MAC Provides IND-CPA, IND-CCA, NM-CPA, INT-PTXT: Theorem 4.3 says that EtM provides IND-CPA security. Consequently, Theorem 4.4 and Theorem 3.1 together imply that EtM provides IND-CCA security. Now, since IND-CCA security implies NM-CPA, EtM provides NM-CPA security. Finally, since INT-CTXT security implies INT-PTXT security, EtM also provides INT-PTXT security.

References

- [1] J.H. An and M. Bellare. Does encryption with redundancy provide authenticity? In *Advances in Cryptology—EUROCRYPT 2002*, ed. by L.R. Knudsen, Amsterdam, The Netherlands, Apr. 28–May 2, 2002. Lecture Notes in Computer Science, vol. 2332 (Springer, Berlin, 2002), pp. 512–528
- [2] J.H. An, Y. Dodis, T. Rabin, On the security of joint signature and encryption, in *Advances in Cryptology—EUROCRYPT 2002*, ed. by L.R. Knudsen, Amsterdam, The Netherlands, Apr. 28–May 2, 2002. Lecture Notes in Computer Science, vol. 2332 (Springer, Berlin, 2002), pp. 83–107
- [3] M. Bellare, New proofs for NMAC and HMAC: Security without collision-resistance, in *Advances in Cryptology—CRYPTO*, ed. by C. Dwork, Santa Barbara, CA, USA, Aug. 20–24, 2006. Lecture Notes in Computer Science, vol. 4117 (Springer, Berlin, 2006), pp. 602–619
- [4] M. Bellare, C. Namprempre, Authenticated encryption: Relations among notions and analysis of the generic composition paradigm, in *Advances in Cryptology—ASIACRYPT 2000*, ed. by T. Okamoto, Kyoto, Japan, Dec. 3–7, 2000. Lecture Notes in Computer Science, vol. 1976 (Springer, Berlin, 2000), pp. 531–545
- [5] M. Bellare, P. Rogaway, Encode-then-encipher encryption: How to exploit nonces or redundancy in plaintexts for efficient cryptography, in *Advances in Cryptology—ASIACRYPT 2000*, ed. by T. Okamoto, Kyoto, Japan, Dec. 3–7, 2000. Lecture Notes in Computer Science, vol. 1976 (Springer, Berlin, 2000), pp. 317–330
- [6] M. Bellare, P. Rogaway, The security of triple encryption and a framework for code-based game-playing proofs, in *Advances in Cryptology—EUROCRYPT 2006*, ed. by S. Vaudenay, St. Petersburg, Russia, May 29–June 1, 2006. Lecture Notes in Computer Science, vol. 4004 (Springer, Berlin, 2006), pp. 409–426. Available as Cryptology ePrint Report 2005/334
- [7] M. Bellare, A. Sahai, Non-malleable encryption: Equivalence between two notions, and an indistinguishability-based characterization, in *Advances in Cryptology—CRYPTO'99*, ed. by M.J. Wiener, Santa Barbara, CA, USA, Aug. 15–19, 1999. Lecture Notes in Computer Science, vol. 1666 (Springer, Berlin, 1999), pp. 519–536. Available as Cryptology ePrint Report 2006/228
- [8] M. Bellare, R. Canetti, H. Krawczyk, Keying hash functions for message authentication, in *Advances in Cryptology—CRYPTO'96*, ed. by N. Koblitz, Santa Barbara, CA, USA, Aug. 18–22, 1996. Lecture Notes in Computer Science, vol. 1109 (Springer, Berlin, 1996), pp. 1–15
- [9] M. Bellare, A. Desai, E. Jorjipii, P. Rogaway, A concrete security treatment of symmetric encryption, in *38th Annual Symposium on Foundations of Computer Science*, Miami Beach, Florida, Oct. 19–22, 1997 (IEEE Computer Society, Los Alamitos, 1997), pp. 394–403
- [10] M. Bellare, A. Desai, D. Pointcheval, P. Rogaway, Relations among notions of security for public-key encryption schemes, in *Advances in Cryptology—CRYPTO'98*, ed. by H. Krawczyk, Santa Barbara, CA, USA, Aug. 23–27, 1998. Lecture Notes in Computer Science, vol. 1462 (Springer, Berlin, 1998), pp. 26–45
- [11] M. Bellare, J. Kilian, P. Rogaway, The security of the cipher block chaining message authentication code. *J. Comput. Syst. Sci.* **61**(3), 362–399 (2000)

- [12] M. Bellare, O. Goldreich, A. Mityagin, The power of verification queries in message authentication and authenticated encryption, 2004. Available as Cryptology ePrint Report 2004/309
- [13] M. Bellare, T. Kohno, C. Namprempre, Breaking and provably repairing the SSH authenticated encryption scheme: A case study of the Encode-then-Encrypt-and-MAC paradigm. *ACM Trans. Inf. Syst. Secur.* **7**(2), 206–241 (2004)
- [14] M. Bellare, P. Rogaway, D. Wagner, The EAX mode of operation, in *Fast Software Encryption 2004*, ed. by B.K. Roy, W. Meier, New Delhi, India, Feb. 5–7, 2004. Lecture Notes in Computer Science, vol. 3017 (Springer, Berlin, 2004), pp. 389–407
- [15] M. Bellare, K. Pietrzak, P. Rogaway, Improved security analyses for CBC MACs, in *Advances in Cryptology—CRYPTO 2005*, ed. by V. Shoup, Santa Barbara, CA, USA, Aug. 14–18, 2005. Lecture Notes in Computer Science, vol. 3621 (Springer, Berlin, 2005), pp. 527–545
- [16] M. Bellare, D. Hoffheinz, E. Kiltz, IND-CCA revisited: When and how should challenge decryption be disallowed? Manuscript, 2007
- [17] J. Black, Authenticated encryption, in *Encyclopedia of Cryptography and Security*, ed. by H.C. van Tilborg (Springer, Berlin, 2005)
- [18] J. Black, P. Rogaway, CBC MACs for arbitrary-length messages: The three-key constructions, in *Advances in Cryptology—CRYPTO 2000*, ed. by M. Bellare, Santa Barbara, CA, USA, Aug. 20–24, 2000. Lecture Notes in Computer Science, vol. 1880 (Springer, Berlin, 2000), pp. 197–215
- [19] J. Black, P. Rogaway, A block-cipher mode of operation for parallelizable message authentication, in *Advances in Cryptology—EUROCRYPT 2002*, ed. by L.R. Knudsen, Amsterdam, The Netherlands, Apr. 28–May 2, 2002. Lecture Notes in Computer Science, vol. 2332 (Springer, Berlin, 2002), pp. 384–397
- [20] J. Black, S. Halevi, H. Krawczyk, T. Krovetz, P. Rogaway, UMAC: Fast and secure message authentication, in *Advances in Cryptology—CRYPTO'99*, ed. by M.J. Wiener, Santa Barbara, CA, USA, Aug. 15–19, 1999. Lecture Notes in Computer Science, vol. 1666 (Springer, Berlin, 1999), pp. 216–233
- [21] R. Canetti, H. Krawczyk, Analysis of key-exchange protocols and their use for building secure channels, in *Advances in Cryptology—EUROCRYPT 2001*, ed. by B. Pfitzmann, Innsbruck, Austria, May 6–10, 2001. Lecture Notes in Computer Science, vol. 2045 (Springer, Berlin, 2001), pp. 451–472
- [22] R. Cramer, I. Damgård, Secure signature schemes based on interactive protocols, in *Advances in Cryptology—CRYPTO'95*, ed. by D. Coppersmith, Santa Barbara, CA, USA, Aug. 27–31, 1995. Lecture Notes in Computer Science, vol. 963 (Springer, Berlin, 1995), pp. 297–310
- [23] A. Desai, New paradigms for constructing symmetric encryption schemes secure against chosen-ciphertext attack, in *Advances in Cryptology—CRYPTO 2000*, ed. by M. Bellare, Santa Barbara, CA, USA, Aug. 20–24, 2000. Lecture Notes in Computer Science, vol. 1880 (Springer, Berlin, 2000), pp. 394–412
- [24] D. Dolev, C. Dwork, M. Naor, Nonmalleable cryptography. *SIAM J. Comput.* **30**(2), 391–437 (2000)
- [25] N. Ferguson, D. Whiting, B. Schneier, J. Kelsey, S. Lucks, T. Kohno, Helix: Fast encryption and authentication in a single cryptographic primitive, in *Fast Software Encryption 2003*, ed. by T. Johansson, Lund, Sweden, Feb. 24–26, 2003. Lecture Notes in Computer Science, vol. 2887 (Springer, Berlin, 2003), pp. 330–346
- [26] A. Freier, P. Karlton, P. Kocher, The SSL protocol: Version 3.0, 1996
- [27] V. Gligor, P. Donescu, Fast encryption and authentication: XCBC encryption and XECB authentication modes, in *Fast Software Encryption 2001*, ed. by M. Matsui, Yokohama, Japan, Apr. 2–4, 2001. Lecture Notes in Computer Science, vol. 2355 (Springer, Berlin, 2001)
- [28] O. Goldreich, A uniform complexity treatment of encryption and zero-knowledge. *J. Cryptol.* **6**(1), 21–53 (1993)
- [29] S. Goldwasser, S. Micali, Probabilistic encryption. *J. Comput. Syst. Sci.* **28**, 270–299 (1984)
- [30] S. Goldwasser, S. Micali, R.L. Rivest, A digital signature scheme secure against adaptive chosen-message attacks. *SIAM J. Comput.* **17**(2), 281–308 (1988)
- [31] S. Halevi, An observation regarding Jutla's modes of operation, 2001. Available as Cryptology ePrint Report 2001/015
- [32] J. Hastad, The security of the IAPM and IACBC modes. *J. Cryptol.* **20**(2), 153–163 (2007)
- [33] T. Iwata, K. Kurosawa, OMAC: One-key CBC MAC, in *Fast Software Encryption 2003*, ed. by T. Johansson, Lund, Sweden, Feb. 24–26, 2003. Lecture Notes in Computer Science, vol. 2887 (Springer, Berlin, 2003), pp. 129–153

- [34] E. Jaulmes, A. Joux, F. Valette, On the security of randomized CBC-MAC beyond the birthday paradox limit: A new construction, in *Fast Software Encryption 2002*, ed. by J. Daemen, V. Rijmen, Leuven, Belgium, Feb. 4–6, 2002. Lecture Notes in Computer Science, vol. 2365 (Springer, Berlin, 2002), pp. 237–251
- [35] C. Jutla, Encryption modes with almost free message integrity, in *Advances in Cryptology—EUROCRYPT 2001*, ed. by B. Pfitzmann, Innsbruck, Austria, May 6–10, 2001. Lecture Notes in Computer Science, vol. 2045 (Springer, Berlin, 2001), pp. 529–544
- [36] J. Katz, M. Yung, Unforgeable encryption and chosen ciphertext secure modes of operation, in *Fast Software Encryption*, ed. by B. Schneier, New York, NY, USA, Apr. 10–12, 2000. Lecture Notes in Computer Science, vol. 1978 (Springer, Berlin, 2000), pp. 284–299
- [37] J. Katz, M. Yung, Characterization of security notions for probabilistic private-key encryption. *J. Cryptol.* **19**(1), 67–95 (2006)
- [38] S. Kent, IP encapsulating security payload (ESP). RFC 4303, Dec. 2005
- [39] T. Kohno, J. Viega, D. Whiting, CWC: A high-performance conventional authenticated encryption mode, in *Fast Software Encryption 2004*, ed. by B.K. Roy, New Delhi, India, Feb. 5–7, 2004. Lecture Notes in Computer Science, vol. 3017 (Springer, Berlin, 2004), pp. 408–426
- [40] H. Krawczyk, The order of encryption and authentication for protecting communications (or: How secure is SSL?), in *Advances in Cryptology—CRYPTO 2001*, ed. by J. Kilian, Santa Barbara, CA, USA, Aug. 19–23, 2001. Lecture Notes in Computer Science, vol. 2139 (Springer, Berlin, 2001), pp. 310–331
- [41] K. Kurosawa, T. Iwata, TMAC: Two-key CBC MAC, in *Topics in Cryptology—CT-RSA 2003*, ed. by M. Joye, San Francisco, CA, USA, Apr. 13–17, 2003. Lecture Notes in Computer Science, vol. 2612 (Springer, Berlin, 2003), pp. 33–49
- [42] D. McGrew, J. Viega, The security and performance of the Galois/Counter Mode (GCM) of operation, in *Progress in Cryptology—INDOCRYPT 2004: 5th International Conference in Cryptology in India*, ed. by A. Canteaut, K. Viswanathan, Chennai, India, Dec. 20–22, 2004. Lecture Notes in Computer Science, vol. 3348 (Springer, Berlin, 2004), pp. 343–355
- [43] M. Naor, M. Yung, Public-key cryptosystems provably secure against chosen ciphertext attacks, in *22nd Annual ACM Symposium on Theory of Computing*, Baltimore, Maryland, USA, May 14–16, 1990 (ACM Press, New York, 1990)
- [44] E. Petrank, C. Rackoff, CBC MAC for real time data sources. *J. Cryptol.* **13**(3), 315–338 (2000)
- [45] C. Rackoff, D.R. Simon, Non-interactive zero-knowledge proof of knowledge and chosen ciphertext attack, in *Advances in Cryptology—CRYPTO’91*, ed. by J. Feigenbaum, Santa Barbara, CA, USA, Aug. 11–15, 1991. Lecture Notes in Computer Science, vol. 576 (Springer, Berlin, 1991), pp. 433–444
- [46] P. Rogaway, Authenticated-encryption with associated-data, in *ACM CCS 2002: 9th Conference on Computer and Communications Security*, ed. by V. Atluri, Washington, D.C., USA, Nov. 18–22, 2002 (ACM Press, New York, 2002), pp. 98–107
- [47] P. Rogaway, T. Shrimpton, A provable-security treatment of the key-wrap problem, in *Advances in Cryptology—EUROCRYPT 2006*, ed. by S. Vaudenay, St. Petersburg, Russia, May 29–June 1, 2006. Lecture Notes in Computer Science, vol. 4004 (Springer, Berlin, 2006), pp. 373–390
- [48] P. Rogaway, M. Bellare, J. Black, T. Krovetz, OCB: A block-cipher mode of operation for efficient authenticated encryption, in *ACM CCS 2001: 8th Conference on Computer and Communications Security*, ed. by M. Reiter, Philadelphia, PA, USA, Nov. 5–8, 2001 (ACM Press, New York, 2001), pp. 196–205
- [49] J. Song, R. Poovendran, J. Lee, T. Iwata, The advanced encryption standard-cipher-based message authentication code-pseudo-random function-128 (AES-CMAC-PRF-128) algorithm for the Internet key exchange protocol (IKE). RFC 4615, 2006
- [50] D. Whiting, R. Housley, N. Ferguson, AES encryption & authentication using CTR mode & CBC-MAC. IEEE P802.11 doc 02/001r2, May 2002
- [51] T. Ylonen, C. Lonvick, The secure shell (SSH) transport layer protocol. RFC 4253, Jan. 2006
- [52] Y. Zheng, Digital signcryption or how to achieve $\text{cost}(\text{signature} \ \&\amp; \ \text{encryption}) \ll \text{cost}(\text{signature}) + \text{cost}(\text{encryption})$, in *Advances in Cryptology—CRYPTO’97*, ed. by B.S. Kaliski, Santa Barbara, CA, USA, Aug. 17–21, 1997. Lecture Notes in Computer Science, vol. 1294 (Springer, Berlin, 1997), pp. 165–179