An extended abstract of this paper appears in Tatsuaki Okamoto, editor, *Advances in Cryptology — ASIACRYPT 2000*, Volume 1976 of Lecture Notes in Computer Science, pages 531–545, Kyoto, Japan, December 3–7, 2000. Springer-Verlag, Berlin, Germany. This is the full version.

Authenticated Encryption: Relations among notions and analysis of the generic composition paradigm

Mihir Bellare[*] Chanathip Namprempre[†]

July 14, 2007

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) to the standard notions of privacy (IND-CCA, NM-CPA) 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 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 counter-examples for the cases where the answer is "no."

Keywords: Symmetric encryption, message authentication, authenticated encryption, concrete security.

[*] Dept. of Computer Science & Engineering, University of California, San Diego, 9500 Gilman Drive, La Jolla, California 92093, USA. E-Mail: mihir@cs.ucsd.edu. URL: http://www-cse.ucsd.edu/users/mihir. 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.

[†] Electrical Engineering, Faculty of Engineering, Thammasat University, Klong Luang, Patumtani 12121, Thailand. E-mail: nchanath@engr.tu.ac.th. URL: http://chanathip.ee.engr.tu.ac.th. Supported in part by grants of the first author and the Thailand Research Fund.

Contents

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 CBC encrypting, 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 [4] 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 of [6]. 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, NM-CCA. (The original definitions were in the asymmetric setting [26, 25, 42, 21] but can be "lifted" to the symmetric setting using the encryption oracle based template of [5]). The relations among these notions are well-understood [6, 21, 33].

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 notion of privacy; namely the notions on which we focus for comparison purposes are INT-PTXT $\wedge$ IND-CPA and INT-CTXT $\wedge$ IND-CPA. (Read "$\wedge$" as "and".)

Figure 1 shows the graph of relations between these notions and the above-mentioned older ones in the style of [6]. An "implication" $\mathbf{A} \rightarrow \mathbf{B}$ means that every symmetric encryption scheme meeting notion $\mathbf{A}$ also meets notion $\mathbf{B}$. A "separation" $\mathbf{A} \not\rightarrow \mathbf{B}$ means that there exists a symmetric encryption scheme meeting notion $\mathbf{A}$ but not notion $\mathbf{B}$. (This under the minimal assumption that some scheme meeting notion $\mathbf{A}$ exists since otherwise the question is moot.) Only a minimal set

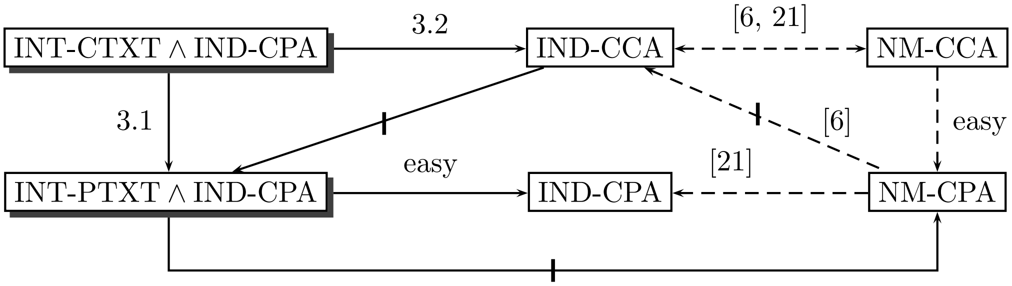

Figure 1: **Relations among notions of symmetric encryption:** An arrow denotes an implication while a barred arrow denotes a separation. The full arrows are relations proved in this paper. Positive results are annotated with the number of the corresponding Theorem. Separation results are derived in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

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 ∧ IND-CPA because otherwise, by following arrows, we would get IND-CCA → INT-PTXT ∧ 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. Non-malleability —whether under chosen-plaintext or chosen-ciphertext attack— does not imply any type of integrity. The intuitive reason is that non-malleability 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 an 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 we are given a symmetric encryption scheme $\mathcal{SE}$ whose encryption and decryption algorithms we denote by $\mathcal{E}$ and $\mathcal{D}$, respectively. (Typically this will be a block cipher mode of operation such as CBC or CTR.) Also assume 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 [4], or UMAC [15]). We assume the encryption scheme meets the weakest notion of privacy, namely IND-CPA. This is an appropriate assumption because standard modes of operations such as CBC and CTR do meet the notion [5] but do not meet stronger notions. We assume 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 [7, 4, 3].) We want to "compose" (meaning, appropriately combine)

Composition Method	**Privacy**			**Integrity**	
	IND-CPA	IND-CCA	NM-CPA	INT-PTXT	INT-CTXT
Encrypt-and-MAC	insecure	insecure	insecure	secure	insecure
MAC-then-encrypt	secure	insecure	insecure	secure	insecure
Encrypt-then-MAC	secure	insecure	insecure	secure	insecure

Composition Method	**Privacy**			**Integrity**	
	IND-CPA	IND-CCA	NM-CPA	INT-PTXT	INT-CTXT
Encrypt-and-MAC	insecure	insecure	insecure	secure	insecure
MAC-then-encrypt	secure	insecure	insecure	secure	insecure
Encrypt-then-MAC	secure	secure	secure	secure	secure

Figure 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.

the given encryption and MA schemes to create an authenticated encryption scheme meeting either INT-CTXT$\wedge$IND-CPA or INT-PTXT$\wedge$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.

— *Encrypt-and-MAC (E&M):* $\overline{\mathcal{E}}(K_e \| K_m, M) = \mathcal{E}(K_e, M) \| \mathcal{T}(K_m, M)$.[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 [48].

— *MAC-then-encrypt (MtE):* $\overline{\mathcal{E}}(K_e \| K_m, M) = \mathcal{E}(K_e, M \| \mathcal{T}(K_m, M))$. 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 [23].

— *Encrypt-then-MAC (EtM):* $\overline{\mathcal{E}}(K_e \| K_m, M) = C \| \mathcal{T}(K_m, C)$ 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 [35].

Here $\overline{\mathcal{E}}$ is the encryption algorithm of the authenticated encryption scheme while the "decrypt+verify" process specifies a decryption algorithm $\overline{\mathcal{D}}$. The latter will either return a plaintext or a special symbol indicating that it considers the ciphertext not authentic.

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

[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.)

the bottom table shows the results assuming that the MAC is strongly unforgeable (SUF-CMA). WUF-CMA is the standard notion [7]— 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 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 [19], 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 exists *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 Section 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 [4] and UMAC [15]. 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 Figure 2, EtM is secure from all points of view, making it a good choice for a standard. However, if INT-PTXT $\wedge$ IND-CPA suffices, then MtE is appropriate too.

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, NM-CCA were denoted IND-CCA2 and NM-CCA2, respectively, in [6]. The chosen-ciphertext attacks here are the adaptive kind [42]. Consideration of non-adaptive chosen-ciphertext attacks [40] leads to two more notions, denoted IND-CCA1 and NM-CCA1 by [6], 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 [33]. 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 [11, 34]. (These two works are concurrent and independent.) Katz and Yung [34] consider two other notions of authenticity not considered here. They also observe the implication INT-CTXT $\wedge$ IND-CPA $\rightarrow$ IND-CCA and present an authenticated encryption scheme called RPC. Signcryption [49] is an asymmetric analog of authenticated encryption.

1.4 Subsequent related work

A preliminary version of our paper appeared in 2000 [9]. Subsequent to this, there has been a lot of work on authenticated encryption. We summarize some of it below.

Extensions. Rogaway [43] 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 [45] 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 [18] show that EtM implements a "secure channel," and Krawczyk [37] shows that E&M and MtE in general do not. Krawczyk [37], 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 redundency" paradigm in which one attempts to get an authenticated encryption scheme by adding some redundancy to the plaintext before encrypting. Bellare and Rogaway [11] 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 [8] 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 IND-CPA $\wedge$ INT-CTXT. These include IACBC [32, 28, 29], OCB [44], XCBC [24], CCM [47], Helix [22], GCM [39], CWC [36] and EAX [13]. Some of these are more efficient than schemes obtained by generic composition, having effectively the same cost as privacy-only schemes.

IND-CCA. Authenticated encryption is not the only approach to achieving IND-CCA. Direct approaches yielding more compact schemes have been provided by Desai [20].

2 Definitions

Conventions. Unless otherwise indicated, an algorithm may be randomized. An adversary is an algorithm. By $y \overset{\$}{\leftarrow} A(x_1, x_2, \ldots)$, we mean we execute algorithm A with fresh coins on inputs

$x_1, x_2, \ldots$, and let y denote the output obtained. By $a_1 \| \ldots \| a_n$, we denote a string encoding of $a_1, \ldots, a_n$ from which the latter are uniquely recoverable.

GAMES. Our definitions and proofs will be in the code-based game-playing style of [12]. We recall some background here. A game —look at Figure 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 "$\mathrm{G}^A \Rightarrow y$" denote the event that this game output takes value y. Boolean flags are assumed initialized to false. Games $\mathrm{G}_i, \mathrm{G}_j$ are *identical until* bad if their code differs only in statements that follow the setting of bad to true. For example, games $\mathrm{G}_0, \mathrm{G}_1$ of Figure 7 are identical until bad. The following is the Fundamental Lemma of game-playing of [12].

Lemma 2.1 [12] Let $\mathrm{G}_i, \mathrm{G}_j$ be identical until bad games, and A an adversary. Then for any y

$$\Pr[\,\mathrm{G}_i^A \Rightarrow y\,] - \Pr[\,\mathrm{G}_j^A \Rightarrow y\,] \;\leq\; \Pr[\,\mathrm{G}_j \text{ sets } \mathsf{bad}\,]\;. \quad \blacksquare$$

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 includes the time used by the game procedures in this execution. It also includes the size of the adversary description (code), all this in some fixed RAM computation model.

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 $\mathcal{E}$. 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 [5], but formulate the definitions using games. The **Initialize** procedure of Game

```
proc Initialize
$K \xleftarrow{\$} \mathcal{K}$ ; $b \xleftarrow{\$} \{0,1\}$

proc LR($M_0, M_1$)
$C \xleftarrow{\$} \mathcal{E}(K, M_b)$ ; Return $C$

proc Finalize($d$)
Return $(d = b)$
```

```
proc Initialize
$K \xleftarrow{\$} \mathcal{K}$ ; $b \xleftarrow{\$} \{0,1\}$ ; $S \leftarrow \emptyset$

proc LR($M_0, M_1$)
$C \xleftarrow{\$} \mathcal{E}(K, M_b)$ ; $S \leftarrow S \cup \{C\}$ ; Return $C$

proc Dec($C$)
If $C \notin S$ then $M \leftarrow \mathcal{D}(K, C)$ else $M \leftarrow \perp$
Return $M$

proc Finalize($d$)
Return $(d = b)$
```

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

```
proc Initialize
$K \xleftarrow{\$} \mathcal{K}$ ; $b \xleftarrow{\$} \{0,1\}$ ; $S \leftarrow \emptyset$

proc LR($M_0, M_1$)
If $\mathsf{pdec}$ then $C \leftarrow \perp$
Else $C \xleftarrow{\$} \mathcal{E}(K, M_b)$ ; $S \leftarrow S \cup \{C\}$
Return $C$

proc Enc($M$)
$C \xleftarrow{\$} \mathcal{E}(K, M)$ ; Return $C$
```

```
proc Dec*($C^*$)
$\mathsf{pdec} \leftarrow \mathsf{true}$
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 $M^*$

proc Finalize($d$)
Return $(d = b)$
```

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

IND-CPA$_{\mathcal{SE}}$ of Figure 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 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

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

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

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

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. IND-CPA secure symmetric encryption schemes include the CBC and CTR modes of operation [5].

Non-malleability. We will not use definitions of non-malleability as per [6, 21, 33] but instead use the equivalent indistinguishability under parallel chosen-ciphertext attack characterization of

```
proc Initialize
$K \xleftarrow{\$} \mathcal{K}$ ; $S \leftarrow \emptyset$

proc Enc($M$)
$C \xleftarrow{\$} \mathcal{E}(K, M)$ ; $S \leftarrow S \cup \{M\}$ ; Return $C$

proc VF($C$)
$M \leftarrow \mathcal{D}(K, C)$
If $M \neq \perp$ and $M \notin S$ then win $\leftarrow$ true
Return ($M \neq \perp$)

proc Finalize
Return win
```

```
proc Initialize
$K \xleftarrow{\$} \mathcal{K}$ ; $S \leftarrow \emptyset$

proc Enc($M$)
$C \xleftarrow{\$} \mathcal{E}(K, M)$ ; $S \leftarrow S \cup \{C\}$ ; Return $C$

proc VF($C$)
$M \leftarrow \mathcal{D}(K, C)$
If $M \neq \perp$ and $C \notin S$ then win $\leftarrow$ true
Return ($M \neq \perp$)

proc Finalize
Return win
```

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

[14], adapted to the symmetric setting. This facilitates our proofs and analyses and also facilitates concrete security measurements. Game NM-CPA$_{\mathcal{SE}}$ 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 C^* of ciphertexts. The oracle returns the corresponding plaintext vector. Here and later, the notation $|X^*|$ denotes the number of components of a vector X^* and $X^*[i]$ denotes the i-th component of 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

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

We do not explicitly define NM-CCA because we do not use it barring its appearance in Figure 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 Figure 5. Adversary A wins in the 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 INT-CTXT$_{\mathcal{SE}}$ game if it submits to **VF** a ciphertext C not previously returned by **Enc**. For any adversary A, we let

$$\mathbf{Adv}^{\text{int-ptxt}}_{\mathcal{SE}}(A) = \Pr[\,\text{INT-PTXT}^A_{\mathcal{SE}} \Rightarrow 1\,] \quad \text{and} \quad \mathbf{Adv}^{\text{int-ctxt}}_{\mathcal{SE}}(A) = \Pr[\,\text{INT-CTXT}^A_{\mathcal{SE}} \Rightarrow 1\,] \,.$$

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 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 call the tag τ a MAC too.

```
proc Initialize
$K \xleftarrow{\$} \mathcal{K}$ ; $S \leftarrow \emptyset$

proc Tag($M$)
$\tau \xleftarrow{\$} \mathcal{T}(K, M)$ ; $S \leftarrow S \cup \{M\}$ ; Return $\tau$

proc VF($M, \tau$)
$b \leftarrow \mathcal{V}(K, M, \tau)$
If $b = 1$ and $M \notin S$ then $\mathsf{win} \leftarrow \mathsf{true}$
Return $b$

proc Finalize
Return $\mathsf{win}$
```

```
proc Initialize
$K \xleftarrow{\$} \mathcal{K}$ ; $S \leftarrow \emptyset$

proc Tag($M$)
$\tau \xleftarrow{\$} \mathcal{T}(K, M)$ ; $S \leftarrow S \cup \{(M, \tau)\}$ ; Return $\tau$

proc VF($M, \tau$)
$b \leftarrow \mathcal{V}(K, M, \tau)$
If $b = 1$ and $(M, \tau) \notin S$ then $\mathsf{win} \leftarrow \mathsf{true}$
Return $b$

proc Finalize
Return $\mathsf{win}$
```

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

Unforgeability. We specify the security notions of a message authentication scheme using games. Game WUF-CMA$_{\mathcal{MA}}$ of Figure 6 captures the standard notion of unforgeability under chosen-message attacks, namely the adaptation of the notion of [27] to the symmetric setting as per [7]. This notion considers the adversary successful if it forges a tag of a message that it did not query to its **Tag** oracle. Game SUF-CMA$_{\mathcal{MA}}$ captures a stronger notion in which, to be successful, not only the message but also the tag in the forgery have to be new. For any adversary F, we let

$$\mathbf{Adv}^{\text{wuf-cma}}_{\mathcal{MA}}(F) = \Pr[\,\text{WUF-CMA}^{F}_{\mathcal{MA}} \Rightarrow 1\,] \quad \text{and} \quad \mathbf{Adv}^{\text{suf-cma}}_{\mathcal{MA}}(F) = \Pr[\,\text{SUF-CMA}^{F}_{\mathcal{MA}} \Rightarrow 1\,] \,.$$

It is easy to see that SUF-CMA implies WUF-CMA, meaning 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 [4, 3], CBC-MAC [7, 10], EMAC [41], XCBC [16], PMAC [17], TMAC [38], OMAC [30], and CMAC [46]. UMAC [15] and RMAC [31] are randomized WUF-CMA MA schemes.

3 Relations among notions of symmetric encryption

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

INT-CTXT → INT-PTXT. The following theorem implies that any encryption scheme which is INT-CTXT secure is also INT-PTXT secure.

Theorem 3.1 Let $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be an encryption scheme. Then, for any A,

$$\mathbf{Adv}^{\text{int-ptxt}}_{\mathcal{SE}}(A) \leq \mathbf{Adv}^{\text{int-ctxt}}_{\mathcal{SE}}(A) \,. \quad \blacksquare$$

Proof of Theorem 3.1: Let C be a winning **VF** query made by A in Game INT-PTXT$_{\mathcal{SE}}$ and let $M = \mathcal{D}(K, C)$. Let X be the set of all **Enc** queries made prior to the winning **VF** query, and let Y be the set of all responses to these queries. We claim that $M \notin X$ implies $C \notin Y$, meaning A wins Game INT-CTXT$_{\mathcal{SE}}$ whenever it wins Game INT-PTXT$_{\mathcal{SE}}$. The claim is true because decryption is unique, deterministic, and stateless. Indeed, suppose $C \in Y$. This means there is an $x \in X$ such that C is an encryption of x under K. But if so, $\mathcal{D}(K, C)$ must equal x, meaning $M = \mathcal{D}(K, C)$ is $x \in X$. $\blacksquare$

```
proc Initialize                                    Games $G_0$, $\boxed{G_1}$
    $K \xleftarrow{\$} \mathcal{K}$ ; $b \xleftarrow{\$} \{0,1\}$ ; $S \leftarrow \emptyset$

proc LR($M_0, M_1$)
    $C \xleftarrow{\$} \mathcal{E}(K, M_b)$ ; $S \leftarrow S \cup \{C\}$ ; Return $C$

proc Dec($C$)
    If $C \notin S$ then $M \leftarrow \mathcal{D}(K, C)$ else $M \leftarrow \bot$
    If $M \neq \bot$ then bad $\leftarrow$ true ; $\boxed{M \leftarrow \bot}$
    Return $M$

proc Finalize($d$)
    Return $(d = b)$
```

```
proc Initialize                                    Game $G_2$
    $K \xleftarrow{\$} \mathcal{K}$ ; $b \xleftarrow{\$} \{0,1\}$

proc LR($M_0, M_1$)
    $C \xleftarrow{\$} \mathcal{E}(K, M_b)$ ; Return $C$

proc Dec($C$)
    Return $\bot$

proc Finalize($d$)
    Return $(d = b)$
```

Figure 7: Games G_0, G_1, and G_2 for the proof of Theorem 3.2. Game G_1 contains the code in the box while G_0 does not.

INT-CTXT ∧ IND-CPA → IND-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 weak privacy coupled with strong integrity implies strong privacy.

Theorem 3.2 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

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

Furthermore, 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 of Theorem 3.2: Consider games G_0, G_1, G_2 of Figure 7. We have

$$\begin{aligned}
\Pr[\,\text{IND-CCA}^{A}_{\mathcal{SE}} \Rightarrow \mathsf{true}\,] &= \Pr[\,G_0^A \Rightarrow \mathsf{true}\,] \\
&= \Pr[\,G_1^A \Rightarrow \mathsf{true}\,] + \Big(\Pr[\,G_0^A \Rightarrow \mathsf{true}\,] - \Pr[\,G_1^A \Rightarrow \mathsf{true}\,]\Big) \\
&\leq \Pr[\,G_1^A \Rightarrow \mathsf{true}\,] + \Pr[\,G_1^A \text{ sets } \mathsf{bad}\,] \,,
\end{aligned} \tag{3}$$

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 $\bot$. Therefore,

$$\Pr[\,G_1^A \Rightarrow \mathsf{true}\,] = \Pr[\,G_2^A \Rightarrow \mathsf{true}\,] \,. \tag{4}$$

We will design A_c and A_p so that

$$\Pr[\,G_1^A \text{ sets } \mathsf{bad}\,] \;\leq\; \Pr[\,\text{INT-CTXT}^{A_c}_{\mathcal{SE}} \Rightarrow \mathsf{true}\,] \text{ and} \tag{5}$$

$$\Pr[\,G_2^A \Rightarrow \mathsf{true}\,] \;\leq\; \Pr[\,\text{IND-CPA}^{A_p}_{\mathcal{SE}} \Rightarrow \mathsf{true}\,] \,. \tag{6}$$

Then, Equation (2) follows from Equations (1), (3), (4), (5), and (6). We now describe A_c and A_p.

Adversary A_c starts by picking a random bit b 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. ▌

Separation approach. We use the approach of [6] to show separations. Namely, 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 $\mathbf{A} \not\to \mathbf{B}$ is vacuously and un-interestingly 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 $\mathbf{A} \not\to \mathbf{B}$ 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 $\mathbf{A} \to \mathbf{B}$, and even an artificial example is enough for that.

IND-CCA $\not\to$ 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:

```
Algorithm $\mathcal{E}_1(K, M)$
    $C' \xleftarrow{\$} \mathcal{E}(K, M)$
    $C \leftarrow 0 \| C'$
    Return $C$

Algorithm $\mathcal{D}_1(K, C)$
    Parse $C$ as $b \| C'$ where $b$ is a bit
    If $b = 0$ then $M \leftarrow \mathcal{D}(K, C')$ else $M \leftarrow 0$
    Return $M$
```

We present an attack on $\mathcal{SE}_1$, in the form of an adversary A who defeats the integrity of plaintexts (meaning, wins Game 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

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

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

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

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 $\mathbf{LR}(M_0, M_1)$
        $C \xleftarrow{\$} 0 \| \mathbf{LR}(M_0, M_1)$ ;  Return $C$ to $A$
    On query $\mathbf{Dec}(C)$
        Parse $C$ as $b \| C'$ 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 **LR** oracle to answer calls that A makes to its **LR** oracle.

$\underline{\mathsf{INT\text{-}PTXT} \wedge \mathsf{IND\text{-}CPA} \not\rightarrow \mathsf{NM\text{-}CPA}.}$ Let $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be an encryption scheme that is both INT-PTXT and IND-CPA secure. We define a scheme $\mathcal{SE}_2$ such that $\mathcal{SE}_2$ is INT-PTXT and IND-CPA secure but is not NM-CPA 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 defeats the non-malleability. 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:

```
Algorithm $\mathcal{E}_2(K, M)$
    $C \xleftarrow{\$} \mathcal{E}(K, M)$
    Return $0 \| C$

Algorithm $\mathcal{D}_2(K, C)$
    Parse $C$ as $b \| C'$ where $b$ is a bit
    $M \leftarrow \mathcal{D}(K, C')$ ; return $M$
```

To prove that $\mathcal{SE}_2$ is not NM-CPA secure, we present an attack on $\mathcal{SE}_2$ in the form of an adversary A who violates its non-malleability (meaning, wins Game NM-CPA$_{\mathcal{SE}_2}$) with probability one. It works as follows:

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

To prove that $\mathcal{SE}_2$ is indeed IND-CPA (resp. INT-PTXT) secure, we will show that given any adversary A_p (resp. A_c) attacking $\mathcal{SE}_2$, we can construct an adversary B_p (resp. B_c) attacking $\mathcal{SE}$ such that

$$\mathbf{Adv}^{\text{ind-cpa}}_{\mathcal{SE}_2}(A_p) \;\leq\; \mathbf{Adv}^{\text{ind-cpa}}_{\mathcal{SE}}(B_p) \quad \text{and} \quad \mathbf{Adv}^{\text{int-ptxt}}_{\mathcal{SE}_2}(A_c) \;\leq\; \mathbf{Adv}^{\text{int-ptxt}}_{\mathcal{SE}}(B_c) \; .$$

Furthermore, the running time of B_p (resp. B_c) is big-oh of that of A_p (resp. A_c) and the constructed adversaries make the same number of oracle queries as the given ones. Adversaries B_p and B_c work as follows:

```
Adversary $B_p$
    Run $A_p$
    On query $\mathbf{LR}(M_0, M_1)$
        $C \xleftarrow{\$} 0 \| \mathbf{LR}(M_0, M_1)$
        Return $C$ to $A_p$
    Until $A_p$ halts and outputs a bit $b$
    Return $b$

Adversary $B_c$
    Run $A_c$
    On query $\mathbf{Enc}(M)$
        $C \xleftarrow{\$} 0 \| \mathbf{Enc}(M)$ ; Return $C$ to $A_c$
    On query $\mathbf{VF}(C)$
        Parse $C$ as $b \| C'$ where $b$ is a bit.
        Return $\mathbf{VF}(C')$ to $A_c$
    Until $A_c$ halts.
```

4 Security of the Composite Schemes

We now detail the results for the composite schemes as summarized in Figure 2. Throughout this section, $\mathcal{SE} = (\mathcal{K}_e, \mathcal{E}, \mathcal{D})$ is a given symmetric encryption scheme which is IND-CPA secure, and

$\mathcal{MA} = (\mathcal{K}_m, \mathcal{T}, \mathcal{V})$ is a given message authentication scheme which is WUF-CMA or SUF-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 Section 3.

In presenting a counter-example (meaning a claim that a certain composition method is insecure under some notion of security **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-CPA secure and $\mathcal{MA}'$ is WUF-CMA or SUF-CMA secure but we can present an attack on the composite scheme based on them showing that the composite scheme does not meet notion **A**. Of course, we make the minimal assumptions that some scheme $\mathcal{SE}$ that is IND-CPA secure, and some scheme $\mathcal{MA}$ that is WUF-CMA or SUF-CMA secure, exist, 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-CMA because the negative result then follows for the case in which $\mathcal{MA}'$ is WUF-CMA. (If the composite scheme does not meet notion **A** when $\mathcal{MA}'$ is SUF-CMA, then it cannot hope to do so when $\mathcal{MA}'$ is WUF-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 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

$$\mathbf{Adv}^{\text{int-ptxt}}_{\overline{\mathcal{SE}}}(A) \leq \mathbf{Adv}^{\text{wuf-cma}}_{\mathcal{MA}}(F) \; . \tag{7}$$

Furthermore, F uses the same resources as A. ▮

Proof of Theorem 4.1: Adversary F works as follows:

```
Adversary $F$
    $K_e \xleftarrow{\$} \mathcal{K}_e$
    Run $A$
    On query $\mathbf{Enc}(M)$
        $C' \xleftarrow{\$} \mathcal{E}(K_e, M)$ ; $\tau \xleftarrow{\$} \mathbf{Tag}(M)$ ; Return $C' \| \tau$ to $A$
    On query $\mathbf{VF}(C)$
        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 Equation (7) 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. ■

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 provide details assuming that the MAC is strongly unforgeable below.

Let $\mathcal{MA} = (\mathcal{K}_m, \mathcal{T}, \mathcal{V})$ be a given MA scheme. We define an MA scheme $\mathcal{MA}'$ which is the same as the given one except that it prepends the first bit of the message to the tag. Formally $\mathcal{MA}' = (\mathcal{K}_m, \mathcal{T}', \mathcal{V}')$ has the same key generation algorithm as the given MA scheme and the following tagging and verification algorithms:

```
Algorithm $\mathcal{T}'(K, M)$
    Parse $M$ as $x \| M'$ where $x$ is a bit
    Return $x \| \mathcal{T}(K, M)$
```

```
Algorithm $\mathcal{V}'(K, M, \tau)$
    Parse $M$ as $x \| M'$ where $x$ is a bit
    Parse $\tau$ as $s \| \tau'$ where $s$ is a bit
    If $x = s$ and $\mathcal{V}(K, M, \tau') = 1$ then return 1
    Else return 0
```

It is easy to see that if $\mathcal{MA}$ is SUF-CMA secure then $\mathcal{MA}'$ is SUF-CMA secure. However, if $\mathcal{MA}'$ is used as the base message authentication scheme in the E&M composition method, the resulting symmetric encryption scheme will not achieve IND-CPA because the first bit of the message is provided to the adversary via the MAC. The adversary can use this to break the scheme in the IND-CPA sense as follows. It queries its **LR** oracle with two messages M_0, M_1 such that the first bit of M_0 is 0 and the first bit of M_1 is 1. It gets back ciphertext $C = C' \| \tau$. It lets s be the first bit of τ. As per our construction above, s is the first bit of M_b and hence $s = b$, so the adversary returns s. The advantage of this adversary is one.

In fact, we can make a stronger statement. 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 $\mathcal{MA}$ 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 $C_1 = C_1' \| \tau_1$ and then makes **LR** query x, x to get back a ciphertext $C_2 = C_2' \| \tau_2$. If $\tau_1 = \tau_2$, it returns 0, else it returns 1. Then,

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

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 does not provide IND-CCA and 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. Let $\mathcal{SE}_2 = (\mathcal{K}, \mathcal{E}_2, \mathcal{D}_2)$ be derived from $\mathcal{SE}$ as in the INT-PTXT $\wedge$ IND-CPA $\not\rightarrow$ NM-CPA proof of Section 3. We claim that $\mathcal{SE}_2$ is IND-CPA secure but not INT-CTXT secure. The latter claim is justified by the following attack that has advantage one:

```
Adversary $A$
    $\overline{C} \xleftarrow{\$} \mathbf{Enc}(0)$ ; Parse $\overline{C}$ 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. Finally, we note that the proof that the modified scheme $\mathcal{SE}_2$ is still IND-CPA secure is easy and is omitted.

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:

```
Algorithm $\overline{\mathcal{K}}$
    $K_e \xleftarrow{\$} \mathcal{K}_e$
    $K_m \xleftarrow{\$} \mathcal{K}_m$
    Return $K_e\|K_m$
```

```
Algorithm $\overline{\mathcal{E}}(K_e\|K_m, M)$
    $\tau \xleftarrow{\$} \mathcal{T}(K_m, M)$
    $C \xleftarrow{\$} \mathcal{E}(K_e, M\|\tau)$
    Return $C$
```

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

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

$$\mathbf{Adv}^{\text{int-ptxt}}_{\overline{\mathcal{SE}}}(I) \ \le\ \mathbf{Adv}^{\text{wuf-cma}}_{\mathcal{MA}}(F) \ . \tag{8}$$

Furthermore, F uses the same resources as I. Similarly, given any adversary A against $\overline{\mathcal{SE}}$, we can construct an adversary A_p such that

$$\mathbf{Adv}^{\text{ind-cpa}}_{\overline{\mathcal{SE}}}(A) \ \le\ \mathbf{Adv}^{\text{ind-cpa}}_{\mathcal{SE}}(A_p) \ . \tag{9}$$

Furthermore, 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 of Theorem 4.2: We construct F and A_p as follows:

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

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

It is easy to see that Equations (8) and (9) follow. The claimed resource usage of F and A_p can also be easily justified. ▌

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 as defined in Section 3. 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 non-malleability (meaning, wins Game NM-CPA$_{\overline{\mathcal{SE}}}$) with probability one. It works as follows:

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

MtE does not provide IND-CCA and 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 $K_e \| K_m$
```

```
Algorithm $\overline{\mathcal{E}}(K_e \| K_m, M)$
    $C' \xleftarrow{\$} \mathcal{E}(K_e, M)$
    $\tau' \xleftarrow{\$} \mathcal{T}(K_m, C')$
    $C \leftarrow C' \| \tau'$
    Return $C$
```

```
Algorithm $\overline{\mathcal{D}}(K_e \| K_m, C)$
    Parse $C$ as $C' \| \tau'$
    $M \leftarrow \mathcal{D}(K_e, C')$
    $v \leftarrow \mathcal{V}(K_m, C', \tau')$
    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 Figure 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

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

Furthermore, A_p uses the same resources as A. Similarly, given any adversary I against $\overline{\mathcal{SE}}$, we can construct an adversary F such that

$$\mathbf{Adv}^{\text{int-ptxt}}_{\overline{\mathcal{SE}}}(I) \;\leq\; \mathbf{Adv}^{\text{wuf-cma}}_{\mathcal{MA}}(F) \; . \tag{11}$$

Furthermore, F uses the same resources as I 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 of Theorem 4.3: We construct A_p and F as follows:

```
Adversary $A_p$
    $K_m \xleftarrow{\$} \mathcal{K}_m$
    Run $A$
    On query $\mathbf{LR}(M_0, M_1)$
        $C \xleftarrow{\$} \mathbf{LR}(M_0, M_1)$
        $\tau \xleftarrow{\$} \mathcal{T}(K_m, C)$
        Return $C \| \tau$ to $A$
    Until $A$ halts and returns $b$
    Return $b$
```

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

Equation (10) is easily verified. For Equation (11), let $C = C' \| \tau'$ be a **VF** query of I that leads to its winning game 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 WUF-CMA$_{\mathcal{MA}}$ whenever I wins INT-PTXT$_{\overline{\mathcal{SE}}}$. ▌

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:

```
Algorithm $\mathcal{T}'(K, M)$
    $\tau \xleftarrow{\$} \mathcal{T}(K, M)$
    Return $\tau \| 0$
```

```
Algorithm $\mathcal{V}'(K, M, \tau)$
    Parse $\tau$ as $\tau' \| b$ where $b$ is a bit.
    Return $\mathcal{V}(K, M, \tau')$
```

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 non-malleability of $\overline{\mathcal{SE}}$ (meaning, wins Game NM-CPA$_{\overline{\mathcal{SE}}}$) with probability one. It works as follows:

```
Adversary $A$
    $C \xleftarrow{\$} \mathbf{LR}(0,1)$ ; Parse $C$ as $C' \| \tau$ ; Parse $\tau$ as $\tau' \| x$ where $x$ is a bit
    $\tau \leftarrow \tau' \| 1$ ; $C^*[1] \leftarrow C' \| \tau$ ; $P^* \leftarrow \mathbf{Dec}^*(C^*)$ ; Return $P^*[1]$
```

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

EtM with WUF-CMA base MAC does not provide IND-CCA and INT-CTXT. Suppose 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 INT-CTXT $\wedge$ 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

$$\mathbf{Adv}^{\text{int-ctxt}}_{\overline{\mathcal{SE}}}(I) \;\le\; \mathbf{Adv}^{\text{suf-cma}}_{\mathcal{MA}}(F) \; . \tag{12}$$

Furthermore, F uses the same resources as I 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 of Theorem 4.4: Adversary F is the same as the one described in the proof of Equation (11) in Theorem 4.3. Let $C = C' \| \tau'$ be a **VF** query of I that leads to its winning game INT-CTXT$_{\overline{\mathcal{SE}}}$. If C was not returned to I by **Enc** then F did not query **Tag** with C'. So F was SUF-CMA$_{\mathcal{MA}}$ whenever I wins INT-CTXT$_{\overline{\mathcal{SE}}}$. ▮

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 3.2 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 L. R. Knudsen, editor, *Advances in Cryptology – EUROCRYPT 2002*, volume 2332 of *Lecture Notes in Computer Science*, pages 512–528, Amsterdam, The Netherlands, Apr. 28 – May 2, 2002. Springer-Verlag, Berlin, Germany.

[2] J. H. An, Y. Dodis, and T. Rabin. On the security of joint signature and encryption. In L. R. Knudsen, editor, *Advances in Cryptology – EUROCRYPT 2002*, volume 2332 of *Lecture Notes in Computer Science*, pages 83–107, Amsterdam, The Netherlands, Apr. 28 – May 2, 2002. Springer-Verlag, Berlin, Germany.

[3] M. Bellare. New proofs for NMAC and HMAC: Security without collision-resistance. In C. Dwork, editor, *Advances in Cryptology – CRYPTO 2006*, volume 4117 of *Lecture Notes in Computer Science*, pages 602–619. Springer-Verlag, Berlin, Germany, Aug. 20–24, 2006.

[4] M. Bellare, R. Canetti, and H. Krawczyk. Keying hash functions for message authentication. In N. Koblitz, editor, *Advances in Cryptology – CRYPTO'96*, volume 1109 of *Lecture Notes in Computer Science*, pages 1–15, Santa Barbara, CA, USA, Aug. 18–22, 1996. Springer-Verlag, Berlin, Germany.

[5] M. Bellare, A. Desai, E. Jokipii, and P. Rogaway. A concrete security treatment of symmetric encryption. In *38th Annual Symposium on Foundations of Computer Science*, pages 394–403, Miami Beach, Florida, Oct. 19–22, 1997. IEEE Computer Society Press.

[6] M. Bellare, A. Desai, D. Pointcheval, and P. Rogaway. Relations among notions of security for public-key encryption schemes. In H. Krawczyk, editor, *Advances in Cryptology – CRYPTO'98*, volume 1462 of *Lecture Notes in Computer Science*, pages 26–45, Santa Barbara, CA, USA, Aug. 23–27, 1998. Springer-Verlag, Berlin, Germany.

[7] M. Bellare, J. Kilian, and P. Rogaway. The security of the cipher block chaining message authentication code. In Y. Desmedt, editor, *Advances in Cryptology – CRYPTO'94*, volume 839 of *Lecture Notes in Computer Science*, pages 341–358, Santa Barbara, CA, USA, Aug. 21–25, 1994. Springer-Verlag, Berlin, Germany.

[8] M. Bellare, T. Kohno, and C. Namprempre. Breaking and provably repairing the SSH authenticated encryption scheme: A case study of the Encode-then-Encrypt-and-MAC paradigm. *ACM Transactions on Information and System Security*, 7(2), 2004.

[9] M. Bellare and C. Namprempre. Authenticated encryption: Relations among notions and analysis of the generic composition paradigm. In T. Okamoto, editor, *Advances in Cryptology – ASIACRYPT 2000*, volume 1976 of *Lecture Notes in Computer Science*, pages 531–545, Kyoto, Japan, Dec. 3–7, 2000. Springer-Verlag, Berlin, Germany.

[10] M. Bellare, K. Pietrzak, and P. Rogaway. Improved security analyses for CBC MACs. In V. Shoup, editor, *Advances in Cryptology – CRYPTO 2005*, volume 3621 of *Lecture Notes in Computer Science*, pages 527–545. Springer-Verlag, Berlin, Germany, Aug. 14–18, 2005.

[11] M. Bellare and P. Rogaway. Encode-then-encipher encryption: How to exploit nonces or redundancy in plaintexts for efficient cryptography. In T. Okamoto, editor, *Advances in Cryptology – ASIACRYPT 2000*, volume 1976 of *Lecture Notes in Computer Science*, pages 317–330, Kyoto, Japan, Dec. 3–7, 2000. Springer-Verlag, Berlin, Germany.

[12] M. Bellare and P. Rogaway. Code-based game-playing proofs and the security of triple encryption. In S. Vaudenay, editor, *Advances in Cryptology – EUROCRYPT 2006*, volume 4004 of *Lecture Notes in Computer Science*, St. Petersburg, Russia, May 29 – June 1, 2006. Springer-Verlag, Berlin, Germany. Available as Cryptology ePrint Report 2005/334.

[13] M. Bellare, P. Rogaway, and D. Wagner. The EAX mode of operation. In B. K. Roy and W. Meier, editors, *Fast Software Encryption 2004*, volume 3017 of *Lecture Notes in Computer Science*, pages 389–407. Springer-Verlag, Berlin, Germany, Feb. 5–7, 2004.

[14] M. Bellare and A. Sahai. Non-malleable encryption: Equivalence between two notions, and an indistinguishability-based characterization. IACR ePrint Report 2006/228. Preliminary version in E. Brickell, editor, Advances in Cryptology – CRYPTO '99, volume 740 of *Lecture Notes in Computer Science*, pages 519–536. Springer-Verlag, Berlin, Germany, Aug. 1999.

[15] J. Black, S. Halevi, H. Krawczyk, T. Krovetz, and P. Rogaway. UMAC: Fast and secure message authentication. In M. J. Wiener, editor, *Advances in Cryptology – CRYPTO'99*, volume 1666 of *Lecture Notes in Computer Science*, pages 216–233, Santa Barbara, CA, USA, Aug. 15–19, 1999. Springer-Verlag, Berlin, Germany.

[16] J. Black and P. Rogaway. CBC MACs for arbitrary-length messages: The three-key constructions. In M. Bellare, editor, *Advances in Cryptology – CRYPTO 2000*, volume 1880 of *Lecture Notes in Computer Science*, pages 197–215, Santa Barbara, CA, USA, Aug. 20–24, 2000. Springer-Verlag, Berlin, Germany.

[17] J. Black and P. Rogaway. A block-cipher mode of operation for parallelizable message authentication. In L. R. Knudsen, editor, *Advances in Cryptology – EUROCRYPT 2002*, volume 2332 of *Lecture Notes in Computer Science*, pages 384–397, Amsterdam, The Netherlands, Apr. 28 – May 2, 2002. Springer-Verlag, Berlin, Germany.

[18] R. Canetti and H. Krawczyk. Analysis of key-exchange protocols and their use for building secure channels. In B. Pfitzmann, editor, *Advances in Cryptology – EUROCRYPT 2001*, volume 2045 of *Lecture Notes in Computer Science*, pages 451–472. Springer-Verlag, Berlin, Germany, May 6–10, 2001.

[19] R. Cramer and I. Damgård. Secure signature schemes based on interactive protocols. In D. Coppersmith, editor, *Advances in Cryptology – CRYPTO'95*, volume 963 of *Lecture Notes in Computer Science*, pages 297–310, Santa Barbara, CA, USA, Aug. 27–31, 1995. Springer-Verlag, Berlin, Germany.

[20] A. Desai. New paradigms for constructing symmetric encryption schemes secure against chosen-ciphertext attack. In M. Bellare, editor, *Advances in Cryptology – CRYPTO 2000*, volume 1880 of *Lecture Notes in Computer Science*, pages 394–412, Santa Barbara, CA, USA, Aug. 20–24, 2000. Springer-Verlag, Berlin, Germany.

[21] D. Dolev, C. Dwork, and M. Naor. Nonmalleable cryptography. *SIAM Journal on Computing*, 30(2):391–437, 2000.

[22] N. Ferguson, D. Whiting, B. Schneier, J. Kelsey, S. Lucks, and T. Kohno. Helix: Fast encryption and authentication in a single cryptographic primitive. In T. Johansson, editor, *Fast Software Encryption 2003*, volume 2887 of *Lecture Notes in Computer Science*, pages 330–346. Springer-Verlag, Berlin, Germany, Feb. 24–26, 2003.

[23] A. Freier, P. Karlton, and P. Kocher. The SSL protocol: Version 3.0, 1996.

[24] V. Gligor and P. Donescu. Fast encryption and authentication: XCBC encryption and XECB authentication modes. In M. Matsui, editor, *Fast Software Encryption 2001*, volume 2355 of *Lecture Notes in Computer Science*. Springer-Verlag, Berlin, Germany, Apr. 2–4, 2001.

[25] O. Goldreich. A uniform complexity treatment of encryption and zero-knowledge. *Journal of Cryptology*, 6(1):21–53, 1993.

[26] S. Goldwasser and S. Micali. Probabilistic encryption. *Journal of Computer and System Sciences*, 28:270–299, 1984.

[27] S. Goldwasser, S. Micali, and R. L. Rivest. A digital signature scheme secure against adaptive chosen-message attacks. *SIAM Journal on Computing*, 17(2):281–308, Apr. 1988.

[28] S. Halevi. An observation regarding Jutla's modes of operation. IACR ePrint Report 2001/015, 2001.

[29] J. Hastad. The security of the IAPM and IACBC modes. *Journal of Cryptology*, 20(2):153–163, 2007.

[30] T. Iwata and K. Kurosawa. OMAC: One-key CBC MAC. In T. Johansson, editor, *Fast Software Encryption 2003*, volume 2887 of *Lecture Notes in Computer Science*, pages 129–153. Springer-Verlag, Berlin, Germany, Feb. 24–26, 2003.

[31] E. Jaulmes, A. Joux, and F. Valette. On the security of randomized CBC-MAC beyond the birthday paradox limit: A new construction. In J. Daemen and V. Rijmen, editors, *Fast Software Encryption 2002*, volume 2365 of *Lecture Notes in Computer Science*, pages 237–251. Springer-Verlag, Berlin, Germany, Feb. 4–6, 2002.

[32] C. Jutla. Encryption modes with almost free message integrity. In B. Pfitzmann, editor, *Advances in Cryptology – EUROCRYPT 2001*, volume 2045 of *Lecture Notes in Computer Science*, pages 529–544. Springer-Verlag, Berlin, Germany, May 6–10, 2001.

[33] J. Katz and M. Yung. Complete characterization of security notions for probabilistic private-key encryption. In *32nd Annual ACM Symposium on Theory of Computing*, pages 245–254, Portland, Oregon, USA, May 21–23, 2000. ACM Press.

[34] J. Katz and M. Yung. Unforgeable encryption and chosen ciphertext secure modes of operation. In B. Schneier, editor, *Fast Software Encryption 2000*, volume 1978 of *Lecture Notes in Computer Science*, pages 284–299. Springer-Verlag, Berlin, Germany, Apr. 10–12, 2000.

[35] S. Kent. IP encapsulating security payload (ESP). RFC 4303, Dec. 2005.

[36] T. Kohno, J. Viega, and D. Whiting. CWC: A high-performance conventional authenticated encryption mode. In B. K. Roy and W. Meier, editors, *Fast Software Encryption 2004*, volume 3017 of *Lecture Notes in Computer Science*, pages 408–426. Springer-Verlag, Berlin, Germany, Feb. 5–7, 2004.

[37] H. Krawczyk. The order of encryption and authentication for protecting communications (or: How secure is SSL?). In J. Kilian, editor, *Advances in Cryptology – CRYPTO 2001*, volume 2139 of *Lecture Notes in Computer Science*, pages 310–331, Santa Barbara, CA, USA, Aug. 19–23, 2001. Springer-Verlag, Berlin, Germany.

[38] K. Kurosawa and T. Iwata. TMAC: Two-key CBC MAC. In M. Joye, editor, *Topics in Cryptology – CT-RSA 2003*, volume 2612 of *Lecture Notes in Computer Science*, pages 33–49, San Francisco, CA, USA, Apr. 13–17, 2003. Springer-Verlag, Berlin, Germany.

[39] D. McGrew and J. Viega. The security and performance of the Galois/Counter Mode (GCM) of operation. In A. Canteaut and K. Viswanathan, editors, *Progress in Cryptology - INDOCRYPT 2004: 5th International Conference in Cryptology in India*, volume 3348 of *Lecture Notes in Computer Science*, pages 343–355. Springer-Verlag, Berlin, Germany, Dec. 20–22, 2004.

[40] M. Naor and 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.

[41] E. Petrank and C. Rackoff. CBC MAC for real time data sources. *Journal of Cryptology*, 13(3):315–338, 2000.

[42] C. Rackoff and D. R. Simon. Non-interactive zero-knowledge proof of knowledge and chosen ciphertext attack. In J. Feigenbaum, editor, *Advances in Cryptology – CRYPTO'91*, volume 576 of *Lecture Notes in Computer Science*, pages 433–444, Santa Barbara, CA, USA, Aug. 11–15, 1991. Springer-Verlag, Berlin, Germany.

[43] P. Rogaway. Authenticated-encryption with associated-data. In *ACM CCS 02: 9th Conference on Computer and Communications Security*, pages 98–107. ACM Press, Nov. 18–22, 2002.

[44] P. Rogaway, M. Bellare, J. Black, and T. Krovetz. OCB: A block-cipher mode of operation for efficient authenticated encryption. In *ACM CCS 01: 8th Conference on Computer and Communications Security*, pages 196–205, Philadelphia, PA, USA, Nov. 5–8, 2001. ACM Press.

[45] P. Rogaway and T. Shrimpton. A provable-security treatment of the key-wrap problem. In S. Vaudenay, editor, *Advances in Cryptology – EUROCRYPT 2002*, volume 4004 of *Lecture Notes in Computer Science*, pages 373–390, St. Petersburg, Russia, May 29 – June 1, 2006. Springer-Verlag, Berlin, Germany.

[46] J. Song, R. Poovendran, J. Lee, and 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.

[47] D. Whiting, R. Housley, and N. Ferguson. AES encryption & authentication using CTR mode & CBC-MAC. IEEE P802.11 doc 02/001r2, May 2002.

[48] T. Ylonen and C. Lonvick. The secure shell (SSH) transport layer protocol. RFC 4253, Jan. 2006.

[49] Y. Zheng. Digital signcryption or how to achieve cost(signature & encryption) << cost(signature) + cost(encryption). In B. S. Kaliski, editor, *Advances in Cryptology – CRYPTO'97*, volume 1294 of *Lecture Notes in Computer Science*, pages 165–179. Springer-Verlag, Berlin, Germany, Aug. 17–21, 1997.