Preliminary version.

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

Mihir Bellare* Chanathip Namprempre†

May 26, 2000

Abstract

We consider two possible notions of authenticity for symmetric encryption schemes, namely integrity of plaintexts and integrity of ciphertexts, and relate them to the standard notions of privacy for symmetric encryption schemes 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 plaintext*, *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 at 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 NSF CAREER Award CCR-9624439 and a 1996 Packard Foundation Fellowship in Science and Engineering.

†Dept. of Computer Science & Engineering, University of California at San Diego, 9500 Gilman Drive, La Jolla, California 92093, USA. E-Mail: meaw@cs.ucsd.edu. URL: http://www-cse.ucsd.edu/users/cnamprem. Supported in part by above-mentioned grants of first author.

Contents

1 Introduction

We use the term *authenticated encryption scheme* to refer to a transform whose goal is to provide *both* privacy *and* authenticity of the encapsulated data. Practitioners have been designing such schemes for a long time. (Early efforts were typically based on adding "redundancy" to the message before CBC encrypting, while modern designs combine MACs with standard block cipher modes of operation.) Theoretical recognition of authenticated encryption as a goal in its own right is more recent but growing quickly (cf. [5, 13, 11]). We are lead, even if belatedly, to acknowledge authenticity as an important addition to the list of security goals of a symmetric encryption scheme.[1]

CONTRIBUTIONS IN BRIEF. 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 of security for symmetric encryption by relating them to the main existing notions of privacy, via implications and separations in the style of [3]. The second part of this paper is motivated by emerging standards such as [14] which design authenticated encryption schemes by what we call "generic composition"— we analyze, with regard to meeting the previous notions, several generic composition methods. Let us now look at these items in more detail.

1.1 Relations among notions

We consider the symmetric setting (i.e. private key). Privacy goals for symmetric encryption schemes include indistinguishability [10] and non-malleability [8], each of which can be considered under either chosen-plaintext or chosen-ciphertext attack, leading to four notions of security we abbreviate IND-CPA, IND-CCA, NM-CPA, NM-CCA.[2] Definitions can be provided following [2]. (Their encryption oracle based template can be used to "lift" definitions for the asymmetric setting to the symmetric setting.) The relations among these notions are well-understood [2, 9]. (These papers state results for the asymmetric setting, but as noted in [3] it is an easy exercise to transfer them to the symmetric setting. Details can be found in [12].)

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 now discuss.

These notions of authenticity are by themselves quite disjoint from the notions of privacy; for example a scheme could achieve INT-CTXT yet be sending the message in the clear. 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 [2]. 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 there exists a symmetric encryption

[1] An authenticated encryption scheme, at least in the symmetric setting which we consider for the bulk of this paper, is simply a symmetric encryption scheme meeting this additional goal. So from the security point of view we need only talk of symmetric encryption and add authenticity to the list of its goals.

[2] To avoid a blowup in the number of notions, we are restricting attention to what we consider the main ones. Chosen-ciphertext attack here means the adaptive kind [16], denoted CCA2 in [3]; we will not consider the non-adaptive kind [15]. We will also not consider distinctions between adaptive and non-adaptive access to encryption oracles [12].

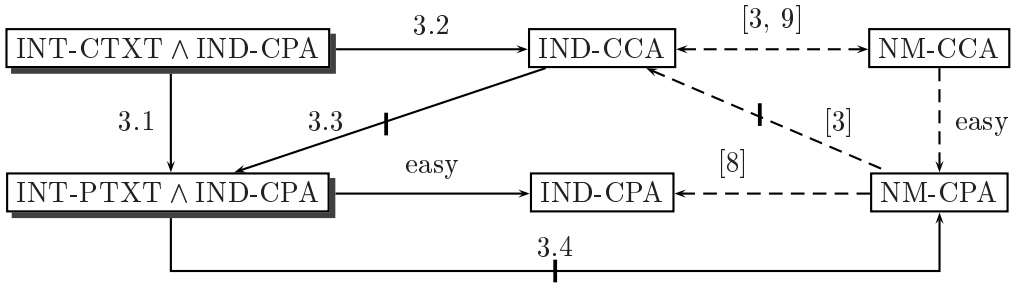

Figure 1: **Relations among notions of symmetric encryption:** An arrow denotes an implication while a barred arrow denotes a separation. The full arrows are relations proved in this paper, annotated with the number of the corresponding Proposition or Theorem, while dotted arrows are reminders of existing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

---

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 ∧ 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 while the numbers annotating the dark lines are pointers to Propositions or Theorems in this paper.

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.

INT-PTXT is considered in [5] and INT-CTXT in [13, 11]. The implication INT-CTXT∧IND-CPA → IND-CCA was independently observed in [13] and the same paper also defines some variants of INT-CTXT which we do not consider.

1.2 Analysis of generic composition

There are many possible approaches to the design of authenticated encryption schemes. As indicated above, the earliest designs were based on adding "redundancy" to the message before encrypting with some block cipher mode of operation. Attacks were found on many of these designs. Enciphering (rather than encrypting) after adding randomness and redundancy can, however, be proven to yield a scheme meeting INT-CTXT∧IND-CPA [5]. Some more specific constructions can be found in [13, 11].

We focus in this paper on a much simpler and better known method which we call "generic composition:" simply combine a standard symmetric encryption scheme with a MAC in some way. There are a few possible ways to do it, and our goal is to analyze and compare their security. The motivation is practical. We will argue that notwithstanding the alternatives, it is this "obvious" method which —as often the case in practice— remains the most pragmatic from the point of view of performance and security architecture design.

Generic composition. Assume we are given a symmetric encryption scheme $\mathcal{SE}$ specified by an

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

Figure 2: Summary of security results for the composed authenticated encryption schemes under the assumption that the given encryption scheme is IND-CPA and the given MAC is unforgeable.

encryption algorithm $\mathcal{E}$ and a decryption algorithm $\mathcal{D}$. (Typically this will be a block cipher mode of operation.) Also assume we are given a message authentication scheme $\mathcal{MA}$ specified by a tagging algorithm $\mathcal{T}$ and a tag verifying algorithm $\mathcal{V}$ and meeting some appropriate notion of unforgeability under chosen-message attack. (Possibilities include the CBC-MAC, HMAC [1], or UMAC [7].) We want to "compose" (meaning, appropriately combine) these 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. (After we present the results we will explain why this is important.) In each case K_e is a key for encryption and K_m is a key for message authentication—

— *Encrypt and MAC plaintext:* $\overline{\mathcal{E}}_{K_e,K_m}(M) = \mathcal{E}_{K_e}(M)\|\mathcal{T}_{K_m}(M)$.[3] 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.

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

— *Encrypt-then-MAC:* $\overline{\mathcal{E}}_{K_e,K_m}(M) = C\|\mathcal{T}_{K_m}(C)$ 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. This is the method of the Internet RFC [14].

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 unauthentic.

Security results. Figure 2 summarizes what we show about the security of the three composite authenticated encryption schemes. Entries in this table have the following meaning:

— *Secure:* The composite encryption scheme in question is proven to meet the security requirement in question, assuming only that the component encryption scheme meets IND-CPA and the message authentication scheme is unforgeable under chosen-message attack.

— *Insecure:* There exists *some* symmetric encryption scheme and some message authentication scheme such that each is secure when taken individually, but when they are used as components under the composition method in question, the resulting authenticated encryption scheme does not meet the security requirement in question.

[3] 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.)

As we can see from Figure 2, the *encrypt-then-MAC* method of [14] is secure from all points of view, making it a good choice for a standard.

The use of a generic composition method secure in the sense above is advantageous — as compared to the other methods to achieve authenticated encryption that we have discussed— from the point of view both of performance and of security architecture. The performance benefit arises from the presence of fast MACs such as UMAC [7] using which the cost of authenticated encryption is essentially the cost of encryption. 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.

In Section 4 we state formal theorems to support the above claims, providing quantitative bounds for the positive results, and counter-examples with attacks for the negative result.

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.

2 Definitions

We present definitions for symmetric encryption, first specifying the *syntax* —meaning what kinds of algorithms make up the scheme— and then specifying formal security measures using a concrete framework which permits quantitative security assessments. Associated to each scheme and each notion of security is an advantage function that measures the maximum possible success probability of an adversary as a function of the resources it invests in the attack. Security as we have discussed it in Section 1 can be interpreted as the requirement that the corresponding advantage function is negligible as a function of the security parameter for an adversary of polynomial resources.

Definition 2.1 [Sntax of symmetric encryption [2]] A *symmetric encryption scheme* $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ consists of three algorithms as follows:

- The *key generation* algorithm $\mathcal{K}$ is randomized. It returns a key K; we write $K \xleftarrow{R} \mathcal{K}$.
- The *encryption* algorithm $\mathcal{E}$ could be randomized or stateful. It takes the key K and a *plaintext* M to return a *ciphertext* C; we write $C \xleftarrow{R} \mathcal{E}_K(M)$.
- The *decryption* algorithm $\mathcal{D}$ is deterministic. It takes the key K and a string C to return either the corresponding plaintext M or the symbol $\perp$; we write $x \leftarrow \mathcal{D}_K(C)$ where $x \in \{0,1\}^* \cup \{\perp\}$.

We require that $\mathcal{D}_K(\mathcal{E}_K(M)) = M$ for all $M \in \{0,1\}^*$. ▌

For brevity, we often omit the word "symmetric" and just say "encryption scheme." 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 both privacy and authenticity. Now we address these two security goals.

Privacy is measured via *indistinguishability* in the "left-or-right" model of [2]. We now recall their definition, which is for privacy under chosen-plaintext attacks. The adversary is allowed queries of the form (x_0, x_1) where these are equal-length messages. Two games are considered. In the first,

each query is responded to by encrypting the left message; in the second, it is the right message. Formally, we define the *left-or-right* oracle $\mathcal{E}_K(\mathcal{LR}(\cdot,\cdot,b))$, where $b \in \{0,1\}$, to take input (x_0, x_1) and do the following: if $b = 0$ it computes $C \leftarrow \mathcal{E}_K(x_0)$ and returns C; else it computes $C \leftarrow \mathcal{E}_K(x_1)$ and returns C. (It is understood that the oracle picks any coins that $\mathcal{E}$ might need if $\mathcal{E}$ is randomized, or updates its state appropriately if $\mathcal{E}$ is stateful.) We consider an encryption scheme to be "good" if a "reasonable" adversary cannot obtain "significant" advantage in distinguishing the cases $b = 0$ and $b = 1$ given access to the oracle.

To model chosen-ciphertext attacks we allow the adversary to also have access to a decryption oracle. Note that if the adversary queries the decryption oracle at a ciphertext output by the left-or-right oracle, then it can obviously easily win the game. Therefore, we disallow it from doing so. Any other query is permissible.

Definition 2.2 [Privacy of a Symmetric Encryption Scheme [2]] Let $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be a symmetric encryption scheme. Let b be a bit. Let A_{cpa} be an adversary that has access to the oracle $\mathcal{E}_K(\mathcal{LR}(\cdot,\cdot,b))$ and let A_{cca} be an adversary that has access to the oracles $\mathcal{E}_K(\mathcal{LR}(\cdot,\cdot,b))$ and $\mathcal{D}_K(\cdot)$. Now, we consider the following experiments:

Experiment $\mathbf{Exp}^{\text{ind-cpa}}_{\mathcal{SE}}(A,b)$	Experiment $\mathbf{Exp}^{\text{ind-cca}}_{\mathcal{SE}}(A,b)$
$K \xleftarrow{R} \mathcal{K}$	$K \xleftarrow{R} \mathcal{K}$
$x \leftarrow A_{\text{cpa}}^{\mathcal{E}_K(\mathcal{LR}(\cdot,\cdot,b))}$	$x \leftarrow A_{\text{cca}}^{\mathcal{E}_K(\mathcal{LR}(\cdot,\cdot,b)),\mathcal{D}_K(\cdot)}$
Return x	Return x

Above it is mandated that A_{cca} never queries $\mathcal{D}_K(\cdot)$ on a ciphertext C output by the $\mathcal{E}_K(\mathcal{LR}(\cdot,\cdot,b))$ oracle, and that the two messages queried of $\mathcal{E}_K(\mathcal{LR}(\cdot,\cdot,b))$ always have equal length. We define the *advantage* of A_{cpa}, and the *advantage* of A_{cca}, respectively, via

$$\begin{aligned}
\mathbf{Adv}^{\text{ind-cpa}}_{\mathcal{SE}}(A_{\text{cpa}}) &= \Pr\left[\mathbf{Exp}^{\text{ind-cpa}}_{\mathcal{SE}}(A_{\text{cpa}},1)=1\right] - \Pr\left[\mathbf{Exp}^{\text{ind-cpa}}_{\mathcal{SE}}(A_{\text{cpa}},0)=1\right] \\
\mathbf{Adv}^{\text{ind-cca}}_{\mathcal{SE}}(A_{\text{cca}}) &= \Pr\left[\mathbf{Exp}^{\text{ind-cca}}_{\mathcal{SE}}(A_{\text{cca}},1)=1\right] - \Pr\left[\mathbf{Exp}^{\text{ind-cca}}_{\mathcal{SE}}(A_{\text{cca}},0)=1\right].
\end{aligned}$$

We define the *advantage function of the scheme for privacy under chosen-plaintext attacks*, and the *advantage function of the scheme for privacy under chosen-ciphertext attacks*, respectively, as follows. For any $t, q_e, q_d, \mu \geq 0$,

$$\mathbf{Adv}^{\text{ind-cpa}}_{\mathcal{SE}}(t,q_e,\mu) = \max_{A_{\text{cpa}}}\{\mathbf{Adv}^{\text{ind-cpa}}_{\mathcal{SE}}(A_{\text{cpa}})\} \quad \Big| \quad \mathbf{Adv}^{\text{ind-cca}}_{\mathcal{SE}}(t,q_e,q_d,\mu) = \max_{A_{\text{cca}}}\{\mathbf{Adv}^{\text{ind-cca}}_{\mathcal{SE}}(A_{\text{cca}})\}$$

where the maximum is over all $A_{\text{cpa}}, A_{\text{cca}}$ with "time complexity" t, each making at most q_e queries to the $\mathcal{E}_K(\mathcal{LR}(\cdot,\cdot,b))$ oracle, totaling at most μ bits, and, in the case of A_{cca}, also making at most q_d queries to the $\mathcal{D}_K(\cdot)$ oracle. ∎

The advantage function is the maximum probability that the security of the scheme $\mathcal{SE}$ can be compromised by an adversary using the indicated resources. The "time complexity" is the worst case total execution time of the experiment, plus the size of the code of the adversary, in some fixed RAM model of computation. We stress that the the total execution time of the experiment includes the time of *all* operations in the experiment, including the time for key generation and the computation of answers to oracle queries. This convention for measuring time complexity is used for all definitions in this paper.

We define non-malleability not directly as per [8] or even [3], but via the equivalent indistinguishability under parallel chosen-ciphertext attack characterization of [6]. This is mainly in order to facilitate the concrete security measurements which, under the more direct definitions, require more parameters. The notations $\vec{\mathcal{E}}_K(\mathcal{LR}(\cdot,\cdot,b))$ and $\vec{\mathcal{D}}_K(\cdot)$ denote similar oracles to those in Definition 2.2 except that the individual inputs (except for the bit b) and outputs are vectors of strings.

Definition 2.3 [Non-Malleability of a Symmetric Encryption Scheme [6]] Let $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be a symmetric encryption scheme. Let b be a bit. Let $A_{\text{cpa}} = (A_{\text{cpa}_1}, A_{\text{cpa}_2})$ be an adversary that has access to the oracle $\vec{\mathcal{E}}_K(\mathcal{LR}(\cdot,\cdot,b))$ and let $A_{\text{cca}} = (A_{\text{cca}_1}, A_{\text{cca}_2})$ be an adversary that has access to the oracles $\vec{\mathcal{E}}_K(\mathcal{LR}(\cdot,\cdot,b))$ and $\vec{\mathcal{D}}_K(\cdot)$. Now, we consider the following experiments:

Experiment $\mathbf{Exp}^{\text{nm-cpa}}_{\mathcal{SE}}(A_{\text{cpa}}, b)$	Experiment $\mathbf{Exp}^{\text{nm-cca}}_{\mathcal{SE}}(A_{\text{cca}}, b)$
$K \xleftarrow{R} \mathcal{K}$	$K \xleftarrow{R} \mathcal{K}$
$(\vec{c}, s) \leftarrow A_{\text{cpa}_1}^{\vec{\mathcal{E}}_K(\mathcal{LR}(\cdot,\cdot,b))}$	$(\vec{c}, s) \leftarrow A_{\text{cca}_1}^{\vec{\mathcal{E}}_K(\mathcal{LR}(\cdot,\cdot,b)),\vec{\mathcal{D}}_K(\cdot)}$
$\vec{p} \leftarrow \vec{\mathcal{D}}_K(\vec{c})$	$\vec{p} \leftarrow \vec{\mathcal{D}}_K(\vec{c})$
$x \leftarrow A_{\text{cpa}_2}(\vec{p}, \vec{c}, s)$	$x \leftarrow A_{\text{cca}_2}(\vec{p}, \vec{c}, s)$
`Return` x	`Return` x

Above it is mandated that the vector $\vec{c}$ output by A_{cpa_1} does not contain any of the ciphertexts output by the $\vec{\mathcal{E}}_K(\mathcal{LR}(\cdot,\cdot,b))$ oracle, and that the pairs of messages queried of $\vec{\mathcal{E}}_K(\mathcal{LR}(\cdot,\cdot,b))$ are always of equal length. We define the *advantage* of A_{cpa}, and the *advantage* of A_{cca}, respectively, via

$$\begin{aligned}
\mathbf{Adv}^{\text{nm-cpa}}_{\mathcal{SE}}(A_{\text{cpa}}) &= \Pr\left[\mathbf{Exp}^{\text{nm-cpa}}_{\mathcal{SE}}(A_{\text{cpa}}, 1) = 1\right] - \Pr\left[\mathbf{Exp}^{\text{nm-cpa}}_{\mathcal{SE}}(A_{\text{cpa}}, 0) = 1\right] \\
\mathbf{Adv}^{\text{nm-cca}}_{\mathcal{SE}}(A_{\text{cca}}) &= \Pr\left[\mathbf{Exp}^{\text{nm-cca}}_{\mathcal{SE}}(A_{\text{cca}}, 1) = 1\right] - \Pr\left[\mathbf{Exp}^{\text{nm-cca}}_{\mathcal{SE}}(A_{\text{cca}}, 0) = 1\right] .
\end{aligned}$$

We define the *advantage function of the scheme for non-malleability under chosen-plaintext attacks*, and the *advantage function of the scheme for non-malleability under chosen-ciphertext attacks*, respectively, as follows. For any $t, q_e, q_d, \mu \geq 0$,

$$\mathbf{Adv}^{\text{nm-cpa}}_{\mathcal{SE}}(t, q_e, \mu) = \max_{A_{\text{cpa}}}\{\mathbf{Adv}^{\text{nm-cpa}}_{\mathcal{SE}}(A_{\text{cpa}})\} \quad \Big| \quad \mathbf{Adv}^{\text{nm-cca}}_{\mathcal{SE}}(t, q_e, q_d, \mu) = \max_{A_{\text{cca}}}\{\mathbf{Adv}^{\text{nm-cca}}_{\mathcal{SE}}(A_{\text{cca}})\}$$

where the maximum is over all $A_{\text{cpa}}, A_{\text{cca}}$ with time complexity t, each making at most q_e queries to the $\vec{\mathcal{E}}_K(\mathcal{LR}(\cdot,\cdot,b))$ oracle, totaling at most μ bits, and, in the case of A_{cca}, also making at most q_d queries to the $\vec{\mathcal{D}}_K(\cdot)$ oracle. ∎

Now we specify security definitions for integrity (authenticity) of a symmetric encryption scheme. The model is similar to that used for message authentication except that the messages are no longer in the clear, but specified implicitly via ciphertexts. The adversary is allowed to mount a chosen-message attack on the scheme, modeled by giving it access to an encryption oracle $\mathcal{E}_K(\cdot)$. Success is measured by its ability to make the decryption oracle output a new plaintext rather than reject by outputting $\perp$. To capture this, we introduce an oracle $\mathcal{D}^*_K(\cdot)$ defined as follows:

Oracle $\mathcal{D}^*_K(C)$
  `If` $\mathcal{D}_K(C) \neq \perp$, `then` return 1.
    `Else` return 0.

The adversary A is given access to this oracle (but not to the decryption oracle itself). It is considered successful if it can make the oracle accept a ciphertext query C that was not "legitimately produced" (i.e. $\mathcal{D}^*_K(C) = 1$). There are two possible restrictions for a valid ciphertext C. One convention is to consider an adversary A successful if the plaintext corresponding to C was never queried of the encryption oracle. A scheme secure in this manner is said to preserve the *integrity of plaintexts*. The other convention is to consider A successful if it simply submits a new valid ciphertext C. A scheme secure in this manner is said to preserve the *integrity of ciphertexts*.

Definition 2.4 [Integrity of an Authenticated Encryption Scheme] Let $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be a symmetric encryption scheme, and let $\mathcal{D}^*_K(\cdot)$ be as defined above. Let A_{ptxt} and A_{ctxt} be adversaries each of which has access to two oracles: $\mathcal{E}_K(\cdot)$ and $\mathcal{D}^*_K(\cdot)$. Consider the following experiments:

Experiment $\mathbf{Exp}^{\text{int-ptxt}}_{\mathcal{SE}}(A_{\text{ptxt}})$
$K \xleftarrow{R} \mathcal{K}$
If $A_{\text{ptxt}}^{\mathcal{E}_K(\cdot),\mathcal{D}_K^*(\cdot)}$ makes a query C to
the oracle $\mathcal{D}_K^*(\cdot)$ such that
– $\mathcal{D}_K^*(C)$ returns 1, and
– $M \stackrel{\text{def}}{=} \mathcal{D}_K(C)$ was never a query to $\mathcal{E}_K(\cdot)$
then return 1 else return 0.

Experiment $\mathbf{Exp}^{\text{int-ctxt}}_{\mathcal{SE}}(A_{\text{ctxt}})$
$K \xleftarrow{R} \mathcal{K}$
If $A_{\text{ctxt}}^{\mathcal{E}_K(\cdot),\mathcal{D}_K^*(\cdot)}$ makes a query C to
the oracle $\mathcal{D}_K^*(\cdot)$ such that
– $\mathcal{D}_K^*(C)$ returns 1, and
– C was never a response of $\mathcal{E}_K(\cdot)$
then return 1 else return 0.

We define the *advantage* of A_{ptxt}, *advantage* of A_{ctxt}, and the *advantage function of the scheme for integrity of plaintexts and ciphertexts*, respectively, as follows. For any $t, q_e, q_d, \mu \geq 0$, let

$$\mathbf{Adv}^{\text{int-ptxt}}_{\mathcal{SE}}(A_{\text{ptxt}}) = \Pr\left[\mathbf{Exp}^{\text{int-ptxt}}_{\mathcal{SE}}(A_{\text{ptxt}}) = 1\right] \qquad \mathbf{Adv}^{\text{int-ctxt}}_{\mathcal{SE}}(A_{\text{ctxt}}) = \Pr\left[\mathbf{Exp}^{\text{int-ctxt}}_{\mathcal{SE}}(A_{\text{ctxt}}) = 1\right]$$
$$\mathbf{Adv}^{\text{int-ptxt}}_{\mathcal{SE}}(t, q_e, q_d, \mu) = \max_{A_{\text{ptxt}}}\{\mathbf{Adv}^{\text{int-ptxt}}_{\mathcal{SE}}(A_{\text{ptxt}})\} \qquad \mathbf{Adv}^{\text{int-ctxt}}_{\mathcal{SE}}(t, q_e, q_d, \mu) = \max_{A_{\text{ctxt}}}\{\mathbf{Adv}^{\text{int-ctxt}}_{\mathcal{SE}}(A_{\text{ctxt}})\}$$

where the maximum is over all A_{ptxt}, A_{ctxt} with time complexity t, each making at most q_e queries to the oracle $\mathcal{E}_K(\cdot)$ and at most q_d queries to the oracle $\mathcal{D}_K^*(\cdot)$ such that the sum of the lengths of all oracle queries is at most μ bits. ▌

A scheme is said to meet a notion such as IND-CPA, IND-CCA, NM-CPA, NM-CCA, INT-PTXT, or INT-CTXT if the advantage —measured as defined above for the notion in question— of any polynomial time adversary is negligible. For the purpose of this definition, it is assumed that there is some implicit security parameter as a function of which these measurements are made.

3 Relations among notions

In this section we state the formal versions of the results summarized in Figure 1. We begin with the implications and then move to the separations. All proofs are in Appendix B. The first implication, below, is a triviality:

Theorem 3.1 [INT-CTXT ⇒ INT-PTXT] Let $\mathcal{SE}$ be an encryption scheme. For any $t, q_e, q_d, \mu \geq 0$,

$$\mathbf{Adv}^{\text{int-ptxt}}_{\mathcal{SE}}(t, q_e, q_d, \mu) \quad \leq \quad \mathbf{Adv}^{\text{int-ctxt}}_{\mathcal{SE}}(t, q_e, q_d, \mu)\ .$$

The next implication is more interesting:

Theorem 3.2 [INT-CTXT ∧ IND-CPA ⇒ IND-CCA] Let $\mathcal{SE}$ be an encryption scheme. For any $t, q_e, q_d, \mu \geq 0$,

$$\mathbf{Adv}^{\text{ind-cca}}_{\mathcal{SE}}(t, q_e, q_d, \mu) \leq 2 \cdot \mathbf{Adv}^{\text{int-ctxt}}_{\mathcal{SE}}(t, q_e, q_d, \mu) + \mathbf{Adv}^{\text{ind-cpa}}_{\mathcal{SE}}(t, q_e, \mu)\ .$$

We use the following approach to show separations. To show that a security notion A, for instance, does not imply a security notion B, we construct a scheme $\mathcal{SE}$ that can be proven secure under the notion A but not under the notion B. Of course, the statement that $A \not\Rightarrow B$ 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 $A \not\Rightarrow B$ that there exists some scheme secure under the notion A.

Proposition 3.3 [IND-CCA ⇏ INT-PTXT] Given a symmetric encryption scheme $\mathcal{SE}$, we can construct a symmetric encryption scheme $\mathcal{SE}'$ such that, for any $t, q_e, q_d, \mu \geq 0$,

$$\mathbf{Adv}^{\text{ind-cca}}_{\mathcal{SE}'}(t, q_e, q_d, \mu) \quad \leq \quad \mathbf{Adv}^{\text{ind-cca}}_{\mathcal{SE}}(t, q_e, q_d, \mu) \tag{1}$$

but for some t'

$$\mathbf{Adv}^{\text{int-ptxt}}_{\mathcal{SE}'}(t', 0, 1, 2) \quad = \quad 1\ . \tag{2}$$

Proposition 3.4 [INT-PTXT $\wedge$ IND-CPA $\not\Rightarrow$ NM-CPA] Given a symmetric encryption scheme $\mathcal{SE}$, we can construct a symmetric encryption scheme $\mathcal{SE}'$ such that, for any $t_p, t_i, q, q_e, q_d, \mu_p, \mu_i \geq 0$,

$$\mathbf{Adv}^{\text{ind-cpa}}_{\mathcal{SE}'}(t_p, q, \mu_p) \leq \mathbf{Adv}^{\text{ind-cpa}}_{\mathcal{SE}}(t_p, q, \mu_p) \quad (3)$$

$$\mathbf{Adv}^{\text{int-ptxt}}_{\mathcal{SE}'}(t_i, q_e, q_d, \mu_i) \leq \mathbf{Adv}^{\text{int-ptxt}}_{\mathcal{SE}}(t_i, q_e, q_d, \mu_i) \quad (4)$$

but for some t'

$$\mathbf{Adv}^{\text{nm-cpa}}_{\mathcal{SE}'}(t', 1, 1) \geq 1 - \mathbf{Adv}^{\text{ind-cpa}}_{\mathcal{SE}'}(t', 1, 1)\ . \quad (5)$$

4 Security of the Composite Schemes

We now present the formal security results of the composite schemes as summarized in Figure 2. The theorems are presented in their full quantitative form and are phrased in terms of the definitions of Section 2. Their proofs can be found in Appendix B. Definition for message authentication schemes which we use below can be found in Appendix A.

Throughout this section, $\mathcal{SE} = (\mathcal{K}_e, \mathcal{E}, \mathcal{D})$ is a given symmetric encryption scheme, $\mathcal{MA} = (\mathcal{K}_m, \mathcal{T}, \mathcal{V})$ is a given message authentication scheme, and $\overline{\mathcal{SE}} = (\overline{\mathcal{K}}, \overline{\mathcal{E}}, \overline{\mathcal{D}})$ is a composite scheme. The assumption that a given encryption scheme $\mathcal{SE}$ is secure corresponds, intuitively, to assuming that $\mathbf{Adv}^{\text{ind-cpa}}_{\mathcal{SE}}(t, q, \mu)$ is small for "reasonably high" values of their argument parameters. Similarly, for a secure MAC scheme $\mathcal{MA}$, we assume that $\mathbf{Adv}_{\mathcal{MA}}(t, q_t, q_v, \mu)$ is small for large parameters. We wish to assess the privacy of $\overline{\mathcal{SE}}$ under chosen-plaintext and chosen-ciphertext attacks, its non-malleability, and its integrity given these assumptions. A claim of security for $\overline{\mathcal{SE}}$ under some security measure corresponds to an upper bound on the corresponding advantage function, provided as a function of the given advantage functions of the given schemes $\mathcal{SE}$ and $\mathcal{MA}$. A claim of insecurity under some measure corresponds to a lower bound on the advantage function of $\mathcal{SE}$ under this measure. The presentation below is method by method, and in each case we begin by specifying the method in more detail.

We make the simplifying assumption that $\mathcal{D}$ never returns $\perp$. It can take any string as input, and the output is always some string. (This is without loss of generality because we can modify $\mathcal{D}$ so that instead of returning $\perp$ it just returns some default message. Security under chosen-plaintext attack is unaffected.) However, $\overline{\mathcal{D}}$ can and will return $\perp$ at times and this is crucial for integrity.

Encrypt and MAC Plaintext. The composite scheme is defined as follows:

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

This composition method does not preserve privacy because the MAC could reveal information about the plaintext. The simplest illustration of this is to consider the case where the message authentication scheme is deterministic. (Most practical ones are, in fact, deterministic, including CBC-MAC and HMAC.) In that case, 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 security in the left-or-right model.

The proof of Proposition 4.1 makes this more precise. It considers an arbitrary symmetric encryption scheme and an arbitrary but deterministic message authentication scheme, and then presents

an adversary attacking the privacy of the corresponding *encrypt and MAC plaintext* scheme. This adversary uses minimal resources: just two chosen plaintexts, each one bit long, and time complexity t enough to cover two applications each of the encryption and tagging functions. It is successful unless the message authentication scheme is very weak in terms of integrity, so for all practical purposes the success probability of our attack is one. (The term $\mathbf{Adv}_{\mathcal{MA}}(t,1,1,1)$ measures the probability of breaking the MAC using a chosen-message attack consisting of a single, one-bit message, and should be viewed as essentially zero.) The proof can be found in Appendix B. We present the formal statement of this result below.

Proposition 4.1 [*Encrypt and MAC plaintext* method is insecure under IND-CPA and IND-CCA] Let $\mathcal{SE}$ be a symmetric encryption scheme, and let $\mathcal{MA}$ be a deterministic message authentication scheme. Let $\overline{\mathcal{SE}}$ be the composite scheme obtained from these by the *encrypt and MAC plaintext* composition method. For any $t \geq t_0$ —where t_0 is a small value specified in the proof— we have

$$\mathbf{Adv}^{\text{ind-cpa}}_{\overline{\mathcal{SE}}}(t,2,2) \quad \geq \quad 1 - \mathbf{Adv}_{\mathcal{MA}}(t,1,1,1) \tag{6}$$

$$\mathbf{Adv}^{\text{ind-cca}}_{\overline{\mathcal{SE}}}(t,2,0,2) \quad \geq \quad 1 - \mathbf{Adv}_{\mathcal{MA}}(t,1,1,1)\ . \tag{7}$$

Proposition 4.1 justifies the claim that the *encrypt and MAC plaintext* method does not preserve privacy. In general, privacy of the *encrypt and MAC plaintext* method will be compromised whenever the MAC reveals partial information about the plaintext, for example, outputs some part of it.

This composition method also fails in general 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. This attack is possible regardless of the assumption on the MAC.

The following proposition makes this more precise. It says that as long as a secure symmetric encryption scheme $\mathcal{SE}$ exists, there also exists another secure symmetric encryption scheme $\mathcal{SE}'$ such that the composite scheme formed by *encrypt and MAC plaintext* based on $\mathcal{SE}'$ and the given MAC scheme can be attacked in terms of integrity of ciphertexts.

Proposition 4.2 [*Encrypt and MAC plaintext* method is insecure under INT-CTXT]
Let $\mathcal{SE} = (\mathcal{K}_e, \mathcal{E}, \mathcal{D})$ and $\mathcal{MA} = (\mathcal{K}_m, \mathcal{T}, \mathcal{V})$ be a symmetric encryption scheme and a message authentication scheme, respectively. We can construct a symmetric encryption scheme $\mathcal{SE}'$ based on $\mathcal{SE}$ such that, for large enough t and for any $q, \mu \geq 0$,

$$\mathbf{Adv}^{\text{ind-cpa}}_{\mathcal{SE}}(t,q,\mu) \quad = \quad \mathbf{Adv}^{\text{ind-cpa}}_{\mathcal{SE}'}(t,q,\mu)$$

but for some t',

$$\mathbf{Adv}^{\text{int-ctxt}}_{\overline{\mathcal{SE}}}(t',1,1,l+1) \quad = \quad 1$$

where $\overline{\mathcal{SE}}$ is a composite scheme constructed via the *encrypt and MAC plaintext* composition method based on the schemes $\mathcal{MA}$ and $\mathcal{SE}'$, and l is the length of a ciphertext of the scheme $\overline{\mathcal{SE}}$.

The first equation says that the modified scheme $\mathcal{SE}'$ is still secure against chosen-plaintext attack, having preserved the security of $\mathcal{SE}$. The second equation says that there is an attack on the composite scheme with regard to integrity of ciphertexts.

Proposition 4.3 [*Encrypt and MAC plaintext* method is insecure under NM-CPA and NM-CCA]

Let $\overline{\mathcal{SE}}$ be a composite scheme obtained via the *encrypt and MAC plaintext* composition method. Then, for large enough $t_1, t_2, q_1, q_2, q_2', \mu_1$ and μ_2, we have

$$\begin{aligned}\mathbf{Adv}^{\text{nm-cpa}}_{\overline{\mathcal{SE}}}(t_1, q_1, \mu_1) &\geq \mathbf{Adv}^{\text{ind-cpa}}_{\overline{\mathcal{SE}}}(t_1, q_1, \mu_1)\\ \mathbf{Adv}^{\text{nm-cca}}_{\overline{\mathcal{SE}}}(t_2, q_2, q_2', \mu_2) &\geq \mathbf{Adv}^{\text{nm-cpa}}_{\overline{\mathcal{SE}}}(t_2, q_2, \mu_2)\ .\end{aligned}$$

The first equation, together with Proposition 4.1, imply that the *encrypt and MAC plaintext* composition method is insecure under NM-CPA. Similarly, the second equation, together with the first, imply that this composition method is insecure under NM-CCA.

Nevertheless, the *encrypt and MAC plaintext* composition method does preserve integrity of plaintexts, in the sense that 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 4.4 [*Encrypt and MAC plaintext* method is secure under INT-PTXT]
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 $\overline{\mathcal{SE}}$ be the encryption scheme obtained from $\mathcal{SE}$ and $\mathcal{MA}$ via the *encrypt and MAC plaintext* composition method. For any $t, q_e, q_d, \mu \geq 0$, the following holds:

$$\mathbf{Adv}^{\text{int-ptxt}}_{\overline{\mathcal{SE}}}(t, q_e, q_d, \mu) \ \leq\ \mathbf{Adv}_{\mathcal{MA}}(t, q_e, q_d, \mu)\ .$$

MAC-THEN-ENCRYPT. The composite scheme is defined as follows:

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

This composition method does not preserve the integrity of ciphertexts for the same reason as in the case of the *encrypt and MAC plaintext* method. In fact, the same attack described in the proof of Proposition 4.2 is also effective here.

In terms of privacy against chosen-ciphertext attacks, this composition method also fails. The reason is the same as before: one can query two messages to the left-or-right encryption oracle, modify part of the response, submit the result to the decryption oracle as a valid query, and completely determine whether the given encryption oracle is a left or a right one. Proposition 4.5 states the implication of this attack, which is presented in Appendix B, more precisely.

Proposition 4.5 [*MAC-then-encrypt* method is insecure under IND-CCA and INT-CTXT]
Let $\mathcal{SE} = (\mathcal{K}_e, \mathcal{E}, \mathcal{D})$ and $\mathcal{MA} = (\mathcal{K}_m, \mathcal{T}, \mathcal{V})$ be a symmetric encryption scheme and a message authentication scheme, respectively. We can construct a symmetric encryption scheme $\mathcal{SE}'$ based on $\mathcal{SE}$ such that, for large enough t and for any $q, \mu \geq 0$,

$$\mathbf{Adv}^{\text{ind-cpa}}_{\mathcal{SE}}(t, q, \mu) \ =\ \mathbf{Adv}^{\text{ind-cpa}}_{\mathcal{SE}'}(t, q, \mu)$$

but for some t_p and t_i,

$$\begin{aligned}\mathbf{Adv}^{\text{ind-cca}}_{\overline{\mathcal{SE}}}(t_p, 1, 1, l+1) &= 1\\ \mathbf{Adv}^{\text{int-ctxt}}_{\overline{\mathcal{SE}}}(t_i, 1, 1, l+1) &= 1\end{aligned}$$

where $\overline{\mathcal{SE}}$ is a composite scheme constructed via the *MAC-then-encrypt* composition method based on the schemes $\mathcal{MA}$ and $\mathcal{SE}'$, and l is the length of a ciphertext of the scheme $\overline{\mathcal{SE}}$.

The first equation says that the modified scheme $\mathcal{SE}'$ is still secure, having preserved the security of $\mathcal{SE}$. The last two equations say that there are attacks on the composite scheme with regard to chosen-ciphertext privacy and integrity of ciphertexts, respectively.

Proposition 4.6 [*MAC-then-encrypt* method is insecure under NM-CPA]
Let $\mathcal{SE} = (\mathcal{K}_e, \mathcal{E}, \mathcal{D})$ and $\mathcal{MA} = (\mathcal{K}_m, \mathcal{T}, \mathcal{V})$ be a symmetric encryption scheme and a message authentication scheme, respectively. We can construct a symmetric encryption scheme $\mathcal{SE}'$ based on $\mathcal{SE}$ such that, for large enough t and for any $q, \mu \geq 0$,

$$\mathbf{Adv}^{\text{ind-cpa}}_{\mathcal{SE}'}(t, q, \mu) \quad \leq \quad \mathbf{Adv}^{\text{ind-cpa}}_{\mathcal{SE}}(t, q, \mu)$$

but for some t',

$$\mathbf{Adv}^{\text{nm-cpa}}_{\overline{\mathcal{SE}}}(t', 1, 1) \geq 1 - \mathbf{Adv}^{\text{ind-cpa}}_{\mathcal{SE}'}(t', 1, 1)$$

where $\overline{\mathcal{SE}}$ is a composite scheme constructed via the *MAC-then-encrypt* composition method based on $\mathcal{MA}$ and $\mathcal{SE}'$.

Proposition 4.7 [*MAC-then-encrypt* method is insecure under NM-CCA]
Let $\overline{\mathcal{SE}}$ be a composite scheme obtained via the *MAC-then-encrypt* composition method. Then, for large enough t, q_e, q_d, and μ, we have

$$\mathbf{Adv}^{\text{nm-cca}}_{\overline{\mathcal{SE}}}(t, q_e, q_d, \mu) \quad \geq \quad \mathbf{Adv}^{\text{ind-cca}}_{\overline{\mathcal{SE}}}(t, q_e, q_d, \mu) \ .$$

The equation above, together with Proposition 4.5, imply that the *MAC-then-encrypt* composition method is insecure under NM-CCA.

Nevertheless, the *MAC-then-encrypt* composition method does preserve both privacy against chosen-plaintext attack and integrity of plaintexts as stated in the following theorem.

Theorem 4.8 [*MAC-then-encrypt* method is secure under INT-PTXT and IND-CPA]
Let $\mathcal{MA} = (\mathcal{K}_m, \mathcal{T}, \mathcal{V})$ be a message authentication scheme, and let $\mathcal{SE} = (\mathcal{K}_e, \mathcal{E}, \mathcal{D})$ be a symmetric encryption scheme secure against chosen-plaintext attacks. Let $\overline{\mathcal{SE}}$ be the encryption scheme obtained from $\mathcal{SE}$ and $\mathcal{MA}$ via the *MAC-then-encrypt* composition method. For any $t_i, t_p, q, q_e, q_d, \mu_i, \mu_p \geq 0$, the following holds:

$$\mathbf{Adv}^{\text{int-ptxt}}_{\overline{\mathcal{SE}}}(t_i, q_e, q_d, \mu_i) \quad \leq \quad \mathbf{Adv}_{\mathcal{MA}}(t_i, q_e, q_d, \mu_i) \tag{8}$$

$$\mathbf{Adv}^{\text{ind-cpa}}_{\overline{\mathcal{SE}}}(t_p, q, \mu_p) \quad \leq \quad \mathbf{Adv}^{\text{ind-cpa}}_{\mathcal{SE}}(t_p, q, \mu_p) \tag{9}$$

Encrypt-then-MAC. The composite scheme is defined as follows:

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

The following theorem implies that the *encrypt-then-MAC* composition method yields a secure authenticated encryption scheme. For brevity, we do not state explicitly in the theorem that this composition method is also secure under NM-CPA and NM-CCA because it follows directly from the results proven in [3], i.e. that IND-CCA implies NM-CPA and NM-CCA.

Theorem 4.9 [*Encrypt-then-mac* method is secure under IND-CPA, IND-CCA, INT-PTXT, and INT-CTXT]
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 $\overline{\mathcal{SE}}$ be the authenticated encryption scheme obtained from $\mathcal{SE}$ and $\mathcal{MA}$ via the *encrypt-then-MAC* composition method. For any set of parameters all ≥ 0, the following holds:

$$\mathbf{Adv}^{\text{int-ptxt}}_{\overline{\mathcal{SE}}}(t_1, q_1, q'_1, \mu_1) \leq \mathbf{Adv}_{\mathcal{MA}}(t_1, q_1, q'_1, \mu_1) \quad (10)$$

$$\mathbf{Adv}^{\text{int-ctxt}}_{\overline{\mathcal{SE}}}(t_2, q_2, q'_2, \mu_2) \leq \mathbf{Adv}_{\mathcal{MA}}(t_2, q_2, q'_2, \mu_2) \quad (11)$$

$$\mathbf{Adv}^{\text{ind-cpa}}_{\overline{\mathcal{SE}}}(t_3, q_3, \mu_3) \leq \mathbf{Adv}^{\text{ind-cpa}}_{\mathcal{SE}}(t_3, q_3, \mu_3) \quad (12)$$

$$\mathbf{Adv}^{\text{ind-cca}}_{\overline{\mathcal{SE}}}(t_4, q_4, q'_4, \mu_4) \leq 2 \cdot \mathbf{Adv}_{\mathcal{MA}}(t_4, q_4, q'_4, \mu_4) + \mathbf{Adv}^{\text{ind-cpa}}_{\mathcal{SE}}(t_4, q_4, \mu_4) \quad (13)$$

References

[1] M. Bellare, R. Canetti and H. Krawczyk, "Keying hash functions for message authentication," *Advances in Cryptology – Crypto '96*, Lecture Notes in Computer Science Vol. 1109, N. Koblitz ed., Springer-Verlag, 1996.

[2] M. Bellare, A. Desai, E. Jokipii and P. Rogaway, "A concrete security treatment of symmetric encryption: Analysis of the DES modes of operation," *Proceedings of the 38th Symposium on Foundations of Computer Science*, IEEE, 1997.

[3] M. Bellare, A. Desai, E. Pointcheval and P. Rogaway, "Relations among notions of security for public-key encryption schemes," *Advances in Cryptology – Crypto '98*, Lecture Notes in Computer Science Vol. 1462, H. Krawczyk ed., Springer-Verlag, 1998.

[4] M. Bellare, J. Kilian, P. Rogaway, "The security of the cipher block chaining message authentication code," *Advances in Cryptology – Crypto '94*, Lecture Notes in Computer Science Vol. 839, Y. Desmedt ed., Springer-Verlag, 1994.

[5] M. Bellare and P. Rogaway, "Encode-then-encipher encryption: How to exploit nonces or redundancy in plaintexts for efficient cryptography," Manuscript, December 1998. Available from authors.

[6] M. Bellare and A. Sahai, "Non-Malleable Encryption: Equivalence between Two Notions, and an Indistinguishability-Based Characterization," *Advances in Cryptology – Crypto '99*, Lecture Notes in Computer Science Vol. 1666, M. Wiener ed., Springer-Verlag, 1999.

[7] J. Black, S. Halevi, H. Krawczyk, T. Krovetz and P. Rogaway, "UMAC: Fast and secure message authentication," *Advances in Cryptology – Crypto '99*, Lecture Notes in Computer Science Vol. 1666, M. Wiener ed., Springer-Verlag, 1999.

[8] D. Dolev, C. Dwork, and M. Naor, "Non-malleable cryptography," *Proceedings of the 23rd Annual Symposium on the Theory of Computing*, ACM, 1991.

[9] D. Dolev, C. Dwork, and M. Naor, "Non-malleable cryptography," Manuscript, 1999, to appear in *SIAM J. Comput.*

[10] S. Goldwasser and S. Micali, "Probabilistic encryption," *Journal of Computer and System Science*, Vol. 28, 1984, pp. 270-299.

[11] C. Jutla, "Encryption modes with almost free message integrity," Manuscript, May 2000.

[12] J. Katz and M. Yung, "Complete characterization of security notions for probabilistic private-key encryption," *Proceedings of the 32nd Annual Symposium on the Theory of Computing*, ACM, 2000.

[13] J. Katz and M. Yung, "Unforgeable Encryption and Adaptively Secure Modes of Operation," *Fast Software Encryption '00*, Lecture Notes in Computer Science Vol. ??, B. Schneier ed., Springer-Verlag, 2000.

[14] S. Kent and R. Atkinson, "IP Encapsulating Security Payload (ESP)," Request for Comments 2406, November 1998.

[15] M. Naor and M. Yung, "Public-key cryptosystems provably secure against chosen ciphertext attacks," *Proceedings of the 22nd Annual Symposium on the Theory of Computing*, ACM, 1990.

[16] C. Rackoff and D. Simon, "Non-Interactive zero-knowledge proof of knowledge and chosen ciphertext attack," *Advances in Cryptology – Crypto '91*, Lecture Notes in Computer Science Vol. 576, J. Feigenbaum ed., Springer-Verlag, 1991.

A Definitions for Message Authentication Schemes

Definition A.1 [Message Authentication Scheme] A *message authentication scheme* $\mathcal{MA} = (\mathcal{K}, \mathcal{T}, \mathcal{V})$ consists of three algorithms as follows:

- The *key generation* algorithm $\mathcal{K}$ is randomized. It returns a key K; we write $K \xleftarrow{R} \mathcal{K}$.
- The *tagging* algorithm $\mathcal{T}$ could be either randomized or stateful. It takes the key K and a message M to return a *tag* σ; we write $\sigma \xleftarrow{R} \mathcal{T}_K(M)$.
- The *verification* algorithm $\mathcal{V}$ is deterministic. It takes the key K, a message M, and a candidate tag σ for M to return a bit v; we write $v \leftarrow \mathcal{V}_K(M, \sigma)$.

We require that $\mathcal{V}_K(M, \mathcal{T}_K(M)) = 1$ for all $M \in \{0,1\}^*$. The scheme is said to be deterministic if the tagging algorithm is deterministic and verification is done via tag re-computation. We sometimes call a message authentication scheme a MAC, and also sometimes call the tag σ a MAC. ▌

Security for message authentication considers an adversary F who is allowed a chosen-message attack, modeled by allowing it access to an oracle for $\mathcal{T}_K(\cdot)$. F is "successful" if it can make the verifying oracle $\mathcal{V}_K(\cdot,\cdot)$ accept a pair (M, σ) that was not "legitimately produced." There are two possible conventions with regard to what "legitimately produced" can mean, leading to two measures of advantage. The "standard" measure is that the message M is "new," meaning F never made query M of its tagging oracle. A more stringent measure considers the adversary successful even if the message is not new, as long as the tag is new. This type of *strong forgery* means that the adversary wins as long as σ was never returned by the tagging oracle in response to query M. We use only this strong notion in this paper as reflected in the definition below.

Definition A.2 [Message Authentication Scheme Security] Let $\mathcal{MA} = (\mathcal{K}, \mathcal{T}, \mathcal{V})$ be a message authentication scheme, and let F be an adversary that has an access to two oracles: $\mathcal{T}_K(\cdot)$ and $\mathcal{V}_K(\cdot,\cdot)$. Consider the following experiment:

Experiment $\mathbf{Exp}_{\mathcal{MA}}(F)$
  $K \xleftarrow{R} \mathcal{K}$
  If $F^{\mathcal{T}_K(\cdot),\mathcal{V}_K(\cdot,\cdot)}$ makes a query (M, σ) to the oracle $\mathcal{V}_K(\cdot,\cdot)$ such that
    – $\mathcal{V}_K(M, \sigma)$ returns 1, and
    – σ was never returned by the oracle $\mathcal{T}_K(\cdot)$ in response to query M
  `then` return 1 `else` return 0.

We define the *advantage* of F, and the *advantage function of the scheme*, respectively, as follows. For any $t, q_t, q_v, \mu \geq 0$, let

$$\mathbf{Adv}_{\mathcal{MA}}(F) = \Pr[\,\mathbf{Exp}_{\mathcal{MA}}(F) = 1\,]$$
$$\mathbf{Adv}_{\mathcal{MA}}(t, q_t, q_v, \mu) = \max_F\{\mathbf{Adv}_{\mathcal{MA}}(F)\}$$

where the maximum is over all F with time complexity t, making at most q_t oracle queries to $\mathcal{T}_K(\cdot)$ and at most q_v oracle queries to $\mathcal{V}_K(\cdot,\cdot)$ such that the sum of the lengths of all oracle queries is at most μ bits. ▌

B Proofs

Proof of Theorem 3.1: This is true because an adversary that violates integrity of plaintexts of a scheme $\mathcal{SE}$ also violates integrity of ciphertexts of the scheme. In particular, if a forgery C is used in a successful violation of integrity of plaintexts of a scheme $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$, then it can also be used to violate integrity of ciphertexts of $\mathcal{SE}$. The reason is that if $M \stackrel{\text{def}}{=} \mathcal{D}_K(C)$ was never queried to the oracle $\mathcal{E}_K$, then the oracle never outputs C as a response. Thus, C is a valid forgery violating integrity of ciphertexts of $\mathcal{SE}$. ▌

Proof of Theorem 3.2: We prove this theorem by constructing two adversaries, one violating integrity of ciphertexts of the scheme and the other violating the privacy of the scheme under the chosen-plaintext attack model. Let A_i be the first adversary and A_p be the second. Let A be an adversary achieving the best possible success in violating the privacy of the scheme under the chosen-ciphertext attack model, meaning it uses resources at most t, q_e, q_d, μ and advantage $\mathbf{Adv}^{\text{ind-cca}}_{\mathcal{SE}}(A)$ equal to $\mathbf{Adv}^{\text{ind-cca}}_{\mathcal{SE}}(t, q_e, q_d, \mu)$. The two adversaries A_i and A_p will use A to achieve its goals. The constructions for A_i and A_p are as follows:

Adversary $A_i^{\mathcal{E}_K(\cdot), \mathcal{D}^*_K(\cdot)}$
  $b' \xleftarrow{R} \{0,1\}$
  For $i = 1, \ldots, q_e + q_d$ do
    When A makes a query M_i, M_i'
    to the oracle $\mathcal{E}_K(\mathcal{LR}(\cdot,\cdot,b))$ do
      If $b' = 0$,
        then $A \leftarrow \mathcal{E}_K(M_i)$,
        else $A \leftarrow \mathcal{E}_K(M_i')$.
    When A makes a query C_i
    to the oracle $\mathcal{D}_K(\cdot)$ do
      $v \leftarrow \mathcal{D}^*_K(C_i)$
      If $v = 0$,
        then $A \leftarrow \perp$,
        else stop.

Adversary $A_p^{\mathcal{E}_K(\mathcal{LR}(\cdot,\cdot,b))}$
  For $i = 1, \ldots, q_e + q_d$ do
    When A makes a query M_i, M_i'
    to the oracle $\mathcal{E}_K(\mathcal{LR}(\cdot,\cdot,b))$ do
      $A \leftarrow \mathcal{E}_K(\mathcal{LR}(M_i, M_i', b))$
    When A makes a query C_i
    to the oracle $\mathcal{D}_K(\cdot)$ do
      $A \leftarrow \perp$
  $A \rightarrow b'$
  Return b'

We will prove that

$$2 \cdot \mathbf{Adv}^{\text{int-ctxt}}_{\mathcal{SE}}(A_i) + \mathbf{Adv}^{\text{ind-cpa}}_{\mathcal{SE}}(A_p) \geq \mathbf{Adv}^{\text{ind-cca}}_{\mathcal{SE}}(A) \tag{14}$$

Consider the experiment $\mathbf{Exp}^{\text{ind-cca}}_{\mathcal{SE}}(A, b)$ where the bit b is chosen at random. Let E denote an event that A makes at least one valid decryption oracle query, i.e. $\mathcal{D}_K(C) \neq \perp$. We relate the probability that A makes a correct guess (i.e. outputs a correct bit $b' = b$) to its advantage as follows:

$$\begin{aligned}
\Pr[\, A \text{ outputs } b' = b \,] &= \Pr[\, b' = 1 \wedge b = 1 \,] + \Pr[\, b' = 0 \wedge b = 0 \,] \\
&= \Pr[\, b' = 1 \mid b = 1 \,] \Pr[\, b = 1 \,] + \Pr[\, b' = 0 \mid b = 0 \,] \Pr[\, b = 0 \,] \\
&= \frac{1}{2}(1 - \Pr[\, b' = 1 \mid b = 0 \,]) + \frac{1}{2}(\Pr[\, b' = 1 \mid b = 1 \,]) \\
&= \frac{1}{2}\mathbf{Adv}^{\text{ind-cca}}_{\mathcal{SE}}(A) + \frac{1}{2}
\end{aligned} \tag{15}$$

Now we analyze the event that A outputs a correct guess in our experiment. First, we know that

$$\Pr[\, A \text{ outputs } b' = b \,] = \Pr[\, A \text{ outputs } b' = b \ \wedge\ E \,] + \Pr[\, A \text{ outputs } b' = b \ \wedge\ \neg E \,] \tag{16}$$

Second, we claim that the following inequalities hold:

$$\begin{aligned} \Pr[\, A \text{ outputs } b' = b \ \wedge\ E \,] &\leq \Pr[\, E \,] = \Pr[\, A_i \text{ succeeds} \,] \\ &= \mathbf{Adv}^{\text{int-ctxt}}_{\mathcal{SE}}(A_i) \qquad (17) \\ \Pr[\, A \text{ outputs } b' = b \ \wedge\ \neg E \,] &\leq \Pr[\, A_p \text{ outputs } b' = b \,] \\ &= \frac{1}{2}\mathbf{Adv}^{\text{ind-cpa}}_{\mathcal{SE}}(A_p) + \frac{1}{2} \qquad (18) \end{aligned}$$

Substituting quantities on the right hand side of Equation (16) with quantities from inequalities (17) and (18), we obtain

$$\Pr[\, A \text{ outputs } b' = b \,] \ \leq\ \mathbf{Adv}^{\text{int-ctxt}}_{\mathcal{SE}}(A_i) + \tfrac{1}{2}\mathbf{Adv}^{\text{ind-cpa}}_{\mathcal{SE}}(A_p) + \tfrac{1}{2}$$

Then, applying Equation (15) and some algebraic manipulation leads to Equation (14). We now justify the claimed inequalities (17) and (18) by analyzing each of them in turn. To justify the inequality (17), we observe that A_i simulates A in the exact same environment as that of the experiment $\mathbf{Exp}^{\text{ind-cca}}_{\mathcal{SE}}(A, b)$. Therefore, if A submits a valid ciphertext as a decryption query (i.e. the event E occurs), A_i can then use this ciphertext as a query to its verification oracle, and so Equation (17) follows. Similarly for the inequality (18), when event E does not occur, A_p simulates A in the exact same environment as that of the experiment $\mathbf{Exp}^{\text{ind-cca}}_{\mathcal{SE}}(A, b)$. Therefore, if A is able to guess the correct bit $b' = b$, so will A_p, and Equation (18) follows. This concludes the proof for Equation (14). Note that here we rely on the assumption that A never queries the decryption oracle on an output of its oracle $\mathcal{E}_K(\mathcal{LR}(\cdot, \cdot, b))$. Without this assumption, A_p would need to memorize its responses to A's encryption queries and then compare them to every decryption query C_i submitted by A. It does so hoping that it can return a correct message M whose encryption is C_i. However, with all its efforts, A_p will not be able to return the correct message since it does not know if the left or the right message the oracle has encrypted during the experiment to obtain C_i.

To justify the message complexity of A_i and A_p, we note that each of A_i and A_p uses the same number of queries as that of A (A_i to its $\mathcal{E}_K(\cdot)$ and $\mathcal{D}^*_K(\cdot)$ oracles and A_p to its $\mathcal{E}_K(\mathcal{LR}(\cdot, \cdot, b))$ oracle). For time complexity, we simply note that we measure the time for each *entire* experiment. Therefore, Equation (14) leads to Theorem 3.2. We omit details. ▌

Proof of Proposition 3.3: Let $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be the given symmetric encryption scheme. We need to define the scheme $\mathcal{SE}'$. The idea is simple. A certain known string (or strings) will be viewed by $\mathcal{D}'$ 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 $\mathcal{SE}' = (\mathcal{K}, \mathcal{E}', \mathcal{D}')$ has the same key generation algorithm as the old scheme and the following modified encryption and decryption algorithms:

Algorithm $\mathcal{E}'_K(M)$	Algorithm $\mathcal{D}'_K(C)$
$C' \leftarrow \mathcal{E}_K(M)$	Parse C as $b\|\|C'$ where b is a bit
$C \leftarrow 0\|\|C'$	If $b = 0$ `then` $M \leftarrow \mathcal{D}_K(C')$; `return` M
`Return` C	`Else return` 0

To justify Equation (2) we present an attack on $\mathcal{SE}'$, in the form of an adversary A who defeats the integrity of plaintexts with probability one. It works as follows:

Adversary $A^{\mathcal{E}'_K(\cdot), \mathcal{D}^{*'}_K(\cdot)}$
  Submit 10 to $\mathcal{D}^{*'}_K(\cdot)$.

We observe that $\mathcal{D}'_K(10) = 0$, meaning 10 is a valid ciphertext, and it decrypts to a message (namely 0) that the adversary has not queried of its oracle. So $\mathbf{Adv}^{\text{int-ptxt}}_{\mathcal{SE}'}(A) = 1$. Also, A makes zero query to $\mathcal{E}'_K(\cdot)$ and one query to $\mathcal{D}^{*'}_K(\cdot)$ totaling 2 bits.

To justify Equation (1), it suffices to associate to any adversary A attacking $\mathcal{SE}'$ an adversary B attacking $\mathcal{SE}$ such that

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

and B has the same resource utilization as A. Adversary B works like this–

Adversary $B^{\mathcal{E}_K(\mathcal{LR}(\cdot,\cdot,b)),\mathcal{D}_K(\cdot)}$
  For $i = 1, \ldots, q_e + q_d$ do
    When A makes a query M_i, M'_i to its LR-encryption oracle
      $A \leftarrow 0 \| \mathcal{E}_K(\mathcal{LR}(M_i, M'_i, b))$
    When A makes a query C_i to its decryption oracle
      Parse C as $b_i \| C'_i$ where b_i is a bit
      If $b = 0$ then $A \leftarrow \mathcal{D}_K(C'_i)$
      Else $A \leftarrow 0$
  Return whatever A returns

As the code shows it is easy for B to use its own oracles to provide A with the answers to A's oracle queries. Thus B is successful with the same probability as A. ∎

Proof of Proposition 3.4: Let $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be the given symmetric encryption scheme. We need to define the scheme $\mathcal{SE}'$. The idea is simple. A redundant bit prepended to ciphertexts is ignored by $\mathcal{D}'$, resulting in the ability to create two different ciphertexts of the same message, which defeats the non-malleability. Here now are the details.

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

Algorithm $\mathcal{E}'_K(M)$	Algorithm $\mathcal{D}'_K(C)$
$C \leftarrow \mathcal{E}_K(M)$	Parse C as $b \| C'$ where b is a bit
Return $0 \| C$	$M \leftarrow \mathcal{D}_K(C')$; return M

To justify Equation (5), we present an attack on $\mathcal{SE}'$, in the form of an adversary $A = (A_{\text{cpa}_1}, A_{\text{cpa}_2})$ who violates its non-malleability with high probability. It works as follows:

Adversary $A^{\vec{\mathcal{E}}'_K(\mathcal{LR}(\cdot,\cdot,b))}_{\text{cpa}_1}$	Adversary $A_{\text{cpa}_2}(\vec{p}, \vec{c}, s)$
$\vec{v_1} \leftarrow \{0\}$; $\vec{v_2} \leftarrow \{1\}$	If the first element of $\vec{p} = 0$
$\vec{c} \leftarrow \vec{\mathcal{E}}'_K(\mathcal{LR}(\vec{v_1}, \vec{v_2}, b))$	then return 0
For each element c of $\vec{c}$	else return 1.
Parse c as $b \| c'$	
$b' \leftarrow b \oplus 1$	
$\vec{c'} \leftarrow \{b' \| c'\}$	
Return $(\vec{c'}, \epsilon)$	

The A_{cpa_1} part of the adversary A queries its oracle with a pair of vectors with just one element each. It then flips the first bit of the resulting vector of ciphertext and outputs it. A_{cpa_2} then compares its plaintext input to the known strings (0 and 1). The probability that A_{cpa_2} outputs a wrong answer is only at most the insecurity of the scheme in terms of chosen-plaintext privacy. This justifies Equation (5).

To justify Equation (3) and Equation (4), it suffices to associate to any adversary A_p attacking the privacy of $\mathcal{SE}'$ an adversary B_p attacking the privacy of $\mathcal{SE}$ such that

$$\mathbf{Adv}^{\text{ind-cpa}}_{\mathcal{SE}'}(A_p) \le \mathbf{Adv}^{\text{ind-cpa}}_{\mathcal{SE}}(B_p)$$

where B_p has the same resource utilization as A_i, and also to associate to any adversary A_i attacking the integrity of plaintexts of $\mathcal{SE}'$ and adversary B_i attacking the integrity of plaintexts of $\mathcal{SE}$ such that B_i has the same resource utilization as A_i and

$$\mathbf{Adv}^{\text{int-ptxt}}_{\mathcal{SE}'}(A_i) \le \mathbf{Adv}^{\text{int-ptxt}}_{\mathcal{SE}}(B_i)$$

The adversaries B_p and B_i work as follows:

Adversary $B_p^{\mathcal{E}_K(\mathcal{LR}(\cdot,\cdot,b)),\mathcal{D}_K(\cdot)}$
  For $i = 1, \ldots, q_e + q_d$ do
    When A_p makes a query (M_i, M_i')
    to the oracle $\mathcal{E}'_K(\mathcal{LR}(\cdot,\cdot,b)))$ do
      $A_p \leftarrow 0 \| \mathcal{E}_K(\mathcal{LR}(M_i, M_i', b))$
    When A_p makes a query C_i
    to the oracle $\mathcal{D}'_K(\cdot)$ do
      Parse C_i as $b \| C_i'$ where b is a bit.
      $A_p \leftarrow \mathcal{D}_K(C_i')$.
  Return whatever A_p returns.

Adversary $B_i^{\mathcal{E}_K(\cdot),\mathcal{D}^*_K(\cdot)}$
  For $i = 1, \ldots, q_t + q_v$ do
    When A_i makes a query M_i
    to the oracle $\mathcal{E}'_K(\cdot)$ do
      $A_i \leftarrow 0 \| \mathcal{E}_K(M_i)$
    When A_i makes a query C_i
    to the oracle $\mathcal{D}^{*'}_K(\cdot)$ do
      Parse C_i as $b \| C_i'$ where b is a bit.
      $A_i \leftarrow \mathcal{D}^*_K(C_i')$.

As the code shows it is easy for B_p and B_i to use their own oracles to provide A_p and A_i with the answers to A_p's and A_i's oracle queries. Thus B_p and B_i are successful with the same probability as A_p and A_i respectively. ▌

Proof of Proposition 4.1: We describe an attack on the privacy of $\overline{\mathcal{SE}}$. Recall that as per Definition 2.2 the adversary has access to the left-right oracle $\overline{\mathcal{E}}_{\langle K_e, K_m\rangle}(\mathcal{LR}(\cdot,\cdot,b))$. In this case, given messages x_0, x_1, the oracle returns $\mathcal{E}_{K_e}(x_b) \| \mathcal{T}_{K_m}(x_b)$. The attack is described by the following adversary:

Adversary $A^{\overline{\mathcal{E}}_{\langle K_e, K_m\rangle}(\mathcal{LR}(\cdot,\cdot,b))}$
  $C_0 \| \sigma_0 \leftarrow \overline{\mathcal{E}}_{\langle K_e, K_m\rangle}(\mathcal{LR}(0,0,b))$ // Query left-right oracle with both messages set to 0
  $C_1 \| \sigma_1 \leftarrow \overline{\mathcal{E}}_{\langle K_e, K_m\rangle}(\mathcal{LR}(0,1,b))$ // Query left-right oracle with messages $(0,1)$
  If $\sigma_1 = \sigma_2$ then return 0 else return 1

If $b = 0$, then the determinism of the $\mathcal{T}$ function means that $\sigma_0 = \sigma_1$ so the output of A is 0. If $b = 1$, then A outputs 1 unless it happens that the messages 0 and 1 have the same MAC, namely $\mathcal{T}_{K_m}(0) = \mathcal{T}_{K_m}(1)$. But if the latter were true, the message authentication scheme is clearly insecure: we could query the tagging function at 0 and then forge the MAC of 1. This can be formalized to show that the success probability of A is at least $1 - \mathbf{Adv}_{\mathcal{MA}}(t, 1, 1, 1)$, and Equation (6) follows. We set t_0 to the time complexity of the above adversary. We omit the details.

The same adversary will also succeed in the chosen-ciphertext attack model. (It simply does not use its decryption oracle.) Therefore, Equation (7) is justified. ▌

Proof of Proposition 4.2: Given an encryption scheme $\mathcal{SE} = (\mathcal{K}_e, \mathcal{E}, \mathcal{D})$, we define the encryption scheme $\mathcal{SE}' = (\mathcal{K}_e, \mathcal{E}', \mathcal{D}')$ to be a scheme with the same key generation algorithm as that of $\mathcal{SE}$ and the same encryption and decryption algorithms as those of the scheme $\mathcal{SE}'$ defined in the proof of Proposition 3.4. Then, we provide the following adversary A attacking the composite scheme $\overline{\mathcal{SE}}$ constructed based on the schemes $\mathcal{SE}'$ and $\mathcal{MA}$:

Adversary $A^{\overline{\mathcal{E}}_{\langle K_e,K_m\rangle}(\cdot),\overline{\mathcal{D}}^*_{\langle K_e,K_m\rangle}(\cdot)}$
$\quad \overline{C} \leftarrow \overline{\mathcal{E}}_{\langle K_e,K_m\rangle}(0)$
$\quad$ Parse $\overline{C}$ as $0\|C$
$\quad$ Submit $1\|C$ as a query to the oracle $\overline{\mathcal{D}}^*_{\langle K_e,K_m\rangle}(\cdot)$.

The ciphertext submitted to $\overline{\mathcal{D}}^*_{\langle K_e,K_m\rangle}(\cdot)$ is new, meaning was never output by the encryption oracle $\overline{\mathcal{E}}_{\langle K_e,K_m\rangle}(\cdot)$, because it begins with a 1 while all outputs of $\overline{\mathcal{E}}_{\langle K_e,K_m\rangle}(\cdot)$ begin with a 0. Furthermore, it is valid because the decryption algorithm $\mathcal{D}'$ by definition ignores the first bit of any ciphertext it is given. Therefore, A violates the integrity of ciphertexts of $\overline{\mathcal{SE}}$ with probability 1. (Note this A 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 $\mathcal{SE}'$ is still secure against chosen-plaintext attack is easy and is omitted. ▌

Proof of Proposition 4.3: This follows directly from the implication NM-CPA ⇒ IND-CPA and the implication NM-CCA ⇒ NM-CPA. ▌

Proof of Theorem 4.4: We let A be an adversary achieving the best possible success in violating the integrity of plaintexts of the scheme $\overline{\mathcal{SE}}$ as defined in Definition 2.4, while using resources t, q_e, q_d, μ. We will construct a forger F attacking the message authentication scheme $\mathcal{MA}$ so that the resources used by F are at most t, q_e, q_d, μ, and furthermore

$$\mathbf{Adv}^{\text{int-ptxt}}_{\overline{\mathcal{SE}}}(A) \;\le\; \mathbf{Adv}_{\mathcal{MA}}(F)\,. \tag{19}$$

The theorem follows.

Adversary F has access to the oracles $\mathcal{T}_{K_m}(\cdot)$ and $\mathcal{V}_{K_m}(\cdot,\cdot)$ where K_m is a random key for $\mathcal{MA}$. It will pick a key K_e for the encryption algorithm $\mathcal{E}$. Using this key and its own oracles it can simulate the oracles $\overline{\mathcal{E}}_{\langle K_e,K_m\rangle}(\cdot)$ and $\overline{\mathcal{D}}^*_{\langle K_e,K_m\rangle}(\cdot)$ that A needs, and thus answer A's oracle queries. In more detail, it works as follows:

Adversary $F^{\mathcal{T}_{K_m}(\cdot),\mathcal{V}_{K_m}(\cdot,\cdot)}$
$\quad K_e \xleftarrow{R} \mathcal{K}_e$
$\quad$ `For` $i = 1,\ldots,q_e+q_d$ `do`
$\quad\quad$ When A makes a query M_i to its $\overline{\mathcal{E}}_{\langle K_e,K_m\rangle}(\cdot)$ oracle `do`
$\quad\quad\quad C'_i \leftarrow \mathcal{E}_{K_e}(M_i)\,;\; \tau_i \leftarrow \mathcal{T}_{K_m}(M_i)\,;\; A \leftarrow C'_i\|\tau_i$
$\quad\quad$ When A makes a query C_i to its $\overline{\mathcal{D}}^*_{\langle K_e,K_m\rangle}(\cdot)$ oracle `do`
$\quad\quad\quad$ Parse C_i as $C'_i\|\tau_i$; $M_i \leftarrow \mathcal{D}_{K_e}(C'_i)\,;\; v_i \leftarrow \mathcal{V}_{K_m}(M_i,\tau_i)\,;\; A \leftarrow v_i$

Consider a ciphertext $C_i = C'_i\|\tau_i$ that yields a successful forgery of a new plaintext M_i. This means that M_i was never queried to $\overline{\mathcal{E}}_{\langle K_e,K_m\rangle}(\cdot)$, which implies that F never queried it to $\mathcal{T}_{K_m}(\cdot)$ either. Therefore, the pair (M_i,τ_i) is a valid forgery, and Equation (19) is justified. It remains to justify the claims about the resource parameters used by F. The key thing to remember in verifying that this works out as claimed is that, as per our definitions, the resources for both adversaries pertain to the entire experiment which measures their success. ▌

Proof of Proposition 4.5: Given an encryption scheme $\mathcal{SE} = (\mathcal{K}_e,\mathcal{E},\mathcal{D})$, we define the encryption scheme $\mathcal{SE}' = (\mathcal{K}_e,\mathcal{E}',\mathcal{D}')$ to be a scheme with the same key generation algorithm as that of $\mathcal{SE}$ and the same encryption and decryption algorithms as those of the scheme $\mathcal{SE}'$ defined in the proof of Proposition 3.4. Then, we provide the following adversaries A_p and A_i attacking the composite scheme $\overline{\mathcal{SE}}$ constructed based on the schemes $\mathcal{SE}'$ and $\mathcal{MA}$:

Adversary $A_p^{\overline{\mathcal{E}}_{\langle K_e,K_m\rangle}(\mathcal{LR}(\cdot,\cdot,b)),\overline{\mathcal{D}}_{\langle K_e,K_m\rangle}(\cdot)}$
$C \leftarrow \overline{\mathcal{E}}_{\langle K_e,K_m\rangle}(\mathcal{LR}(0,1,b))$
Parse C as $b'\|C'$ where b' is a bit.
$b'' \leftarrow b' \oplus 1$
$M \leftarrow \overline{\mathcal{D}}_{\langle K_e,K_m\rangle}(b''\|C')$
`If` $M = 0$, `then` return 0, `else` return 1.

Adversary $A_i^{\overline{\mathcal{E}}_{\langle K_e,K_m\rangle}(\cdot),\overline{\mathcal{D}}^*_{\langle K_e,K_m\rangle}(\cdot)}$
$C \leftarrow \overline{\mathcal{E}}_{\langle K_e,K_m\rangle}(0)$
Parse C as $b\|C'$ where b' is a bit.
$b \leftarrow b' \oplus 1$
Submit $b\|C'$ to $\overline{\mathcal{D}}^*_{\langle K_e,K_m\rangle}(\cdot)$

Since the first bit of the ciphertext is ignored, the ciphertext that the adversary has constructed should still decrypt to the original message, and thus verify to be valid. Therefore, A_p can tell with complete confidence which plaintext was encrypted by its left-or-right oracle, and similarly A_i can successfully submit a new and valid ciphertext to its $\overline{\mathcal{D}}^*_{\langle K_e,K_m\rangle}(\cdot)$ oracle. Let t_p and t_i be the total time in each of the experiments, and Proposition 4.5 follows. The proof that the modified scheme $\mathcal{SE}'$ is still secure against chosen-plaintext attack is easy and is omitted. ▌

Proof of Proposition 4.6: Let $\mathcal{SE} = (\mathcal{K}_e, \mathcal{E}, \mathcal{D})$ be the given symmetric encryption scheme. We need to define the scheme $\mathcal{SE}' = (\mathcal{K}_e, \mathcal{E}', \mathcal{D}')$. The idea is simple. A redundant bit prepended to ciphertexts is ignored by $\mathcal{D}'$, resulting in the ability to create two different ciphertexts of the same message, which defeats the non-malleability. The proof is, for the most part, the same as that of Proposition 3.4. Thus, we simply note the following: let t' be the total time in the experiment, with this attack, we have $\mathbf{Adv}^{\text{nm-cpa}}_{\overline{\mathcal{SE}}}(t', 1, 1) \geq 1 - \mathbf{Adv}^{\text{ind-cpa}}_{\overline{\mathcal{SE}}}(t', 1, 1)$. This, together with the inequality $\mathbf{Adv}^{\text{ind-cpa}}_{\overline{\mathcal{SE}}}(t', q, \mu) \leq \mathbf{Adv}^{\text{ind-cpa}}_{\mathcal{SE}'}(t', q, \mu)$ implied by Equation (9), lead to the second equation in Proposition 4.6. We omit details. ▌

Proof of Proposition 4.7: This follows directly from the implication NM-CCA ⇒ IND-CCA. ▌

Proof of Theorem 4.8: Equation (8): Similar to the proof of Theorem 4.4, we prove this theorem by constructing an adversary F attacking the scheme $\mathcal{MA}$ using an adversary A attacking the integrity of plaintexts $\overline{\mathcal{SE}}$ as follows:

Adversary $F^{\mathcal{T}_{K_m}(\cdot),\mathcal{V}_{K_m}(\cdot,\cdot)}$
$K_e \xleftarrow{R} \mathcal{K}_e$
`For` $i = 1, \ldots, q_e + q_d$ `do`
When A makes a query M_i to its oracle $\overline{\mathcal{E}}_{\langle K_e,K_m\rangle}(\cdot)$ `do`
$M'_i \leftarrow M_i\|\mathcal{T}_{K_m}(M_i)$
$C'_i \leftarrow \mathcal{E}_{K_e}(M'_i)$
$A \leftarrow C'_i$
When A makes a query C_i to its oracle $\overline{\mathcal{D}}^*_{\langle K_e,K_m\rangle}(\cdot)$ `do`
$M'_i \leftarrow \mathcal{D}_{K_e}(C_i)$
Parse M'_i as $M_i\|\tau_i$.
$v_i \leftarrow \mathcal{V}_{K_m}(M_i, \tau_i)$
$A \leftarrow v_i$

Here F executes the encryption algorithm $\mathcal{E}_{K_e}$ under a random key K_e and invokes its oracles $\mathcal{T}_{K_m}(\cdot)$ and $\mathcal{V}_{K_m}(\cdot,\cdot)$ to respond to the oracle queries of A. Now consider a ciphertext C_i that yields a successful forgery of a new plaintext M_i. Since M_i is new, the pair (M_i, τ_i) where τ_i is obtained from appropriately parsing $\mathcal{D}_{K_e}(C_i)$ as described in the above algorithm is a valid strong forgery. We omit details. ▌

Proof of Theorem 4.8: Equation (9): We prove this claim by constructing an adversary A_p attacking the base scheme $\mathcal{SE}$ using the adversary A attacking the privacy of $\overline{\mathcal{SE}}$ against chosen-plaintext attacks. The construction is as follows:

Adversary $A_p^{\mathcal{E}_{K_e}(\mathcal{LR}(\cdot,\cdot,b))}$
  $K_m \xleftarrow{R} \mathcal{K}_m$
  For $i = 1, \ldots, q$ do
    When A makes a query (M_i, M_i') to its oracle do
      $\tau_1 \leftarrow \mathcal{T}_{K_m}(M_i)$; $\tau_2 \leftarrow \mathcal{T}_{K_m}(M_i')$
      $M_1 \leftarrow M_i \| \tau_1$; $M_2 \leftarrow M_i' \| \tau_2$
      $C_i \leftarrow \mathcal{E}_{K_e}(\mathcal{LR}(M_1, M_2, b))$
      $A \leftarrow C_i$
  $A \rightarrow b'$
  Return b'

Here A_p executes the tagging algorithm $\mathcal{T}$ under a random key K_m and invokes its oracle $\mathcal{E}_{K_e}(\mathcal{LR}(\cdot,\cdot,b))$ to respond to the oracle queries of A. For each query, it computes the tags of both messages queried by A to generate inputs to its oracle and then lets its oracle decide which input to encrypt. It then outputs A's guess as its own. Since A is simulated in the exact same environment as that of experiment $\mathbf{Exp}^{\text{ind-cpa}}_{\overline{\mathcal{SE}}}(A, b)$ where the bit b is chosen at random, if A can guess correctly, so can A_p. The justification of resource usage is similar to the argument in the proof of Theorem 4.4, and Equation (9) follows. We omit details. ▌

Proof of Theorem 4.9: Equation (10) and Equation (11): Similar to Theorem 4.4, we prove the two equations by constructing an adversary F attacking $\mathcal{MA}$ using an adversary A that successfully violates the integrity of either plaintexts or ciphertexts of the scheme $\overline{\mathcal{SE}}$. F picks a key K_e for the encryption algorithm, then it simulates A. In more details, it works as follows:

Adversary $F^{\mathcal{T}_{K_m}(\cdot),\mathcal{V}_{K_m}(\cdot,\cdot)}$
  $K_e \xleftarrow{R} \mathcal{K}_e$
  For $i = 1, \ldots, q_e + q_d$ do
    When A makes a query M_i to its oracle $\overline{\mathcal{E}}_{\langle K_e, K_m\rangle}(\cdot)$ do
      $C_i' \leftarrow \mathcal{E}_{K_e}(M_i)$; $\tau_i \leftarrow \mathcal{T}_{K_m}(C_i')$; $A \leftarrow C_i' \| \tau_i$
    When A makes a query C_i to its oracle $\overline{\mathcal{D}}^*_{\langle K_e, K_m\rangle}(\cdot)$ do
      Parse C_i as $C_i' \| \tau_i'$; $v_i \leftarrow \mathcal{V}_{K_m}(C_i', \tau_i')$; $A \leftarrow v_i$

We analyze this construction in two cases. First, we consider the case where A violates the integrity of plaintexts of $\overline{\mathcal{SE}}$. This means that A can submit a ciphertext $C_i = C_i' \| \tau_i'$ that yields a successful forgery of a new plaintext M_i. Since a ciphertext can only correspond to at most one message, this means that C_i' is also new. Therefore, the pair (C_i', τ_i') is a valid forgery. Second, we consider the case where A violates the integrity of ciphertexts. This means that A can submit a new ciphertext $C_i = C_i' \| \tau_i'$. The pair (C_i', τ_i') is then a valid forgery. The justification of resource usage of F is similar to that in the proof of Theorem 4.4, and Equation (10) and Equation (11) follow. We omit details. ▌

Proof of Theorem 4.9: Equation (12): We prove this claim by constructing an adversary A_p attacking the base scheme $\mathcal{SE}$ using an adversary A violating the privacy of $\overline{\mathcal{SE}}$ against chosen-plaintext attacks. This construction works independent of the MAC scheme. The construction in detail is as follows:

Adversary $A_p^{\mathcal{E}_{K_e}(\mathcal{LR}(\cdot,\cdot,b))}$
  $K_m \xleftarrow{R} \mathcal{K}_m$
  For $i = 1, \ldots, q$ do
    When A makes a query (M_i, M_i') to its oracle do
      $C_i \leftarrow \mathcal{E}_{K_e}(\mathcal{LR}(M_i, M_i', b))$; $\tau_i \leftarrow \mathcal{T}_{K_m}(C_i)$; $A \leftarrow C_i \| \tau_i$

$A \to b'$
Return b'

Here A_p executes the tagging algorithm $\mathcal{T}$ under a random key K_m and invokes its oracle $\mathcal{E}_{K_e}(\mathcal{LR}(\cdot,\cdot,b))$ to respond to the oracle queries of A. One can verify that A is simulated in the exact same environment as that of experiment $\mathbf{Exp}^{\text{ind-cpa}}_{\mathcal{SE}}(A,b)$ where the bit b is chosen at random. Therefore, if A can successfully determine the bit b, so can A_p. The justification of resource usage of A_p is similar to that in the proof of Theorem 4.4, and Equation (12) follows. We omit details. ▌

Proof of Theorem 4.9: Equation (13): This equation is a corollary of Theorem 3.2, Equation (11), and Equation (12). ▌