Security Notions for Identity Based Encryption

David Galindo and Ichiro Hasuo

Institute for Computing and Information Sciences
Radboud University Nijmegen
P.O.Box 9010, 6500 GL Nijmegen, The Netherlands
{d.galindo,ichiro}@cs.ru.nl

Abstract. Identity Based Encryption (IBE) has attracted a lot of attention since the publication of the scheme by Boneh and Franklin. So far, only indistinguishability based security notions have been considered in the literature, and it has not been investigated whether these definitions are appropriate. For this purpose, we define the goals of semantic security and non-malleability for IBE. We then compare the security notions resulting from combining those goals with the attacks previously considered in the literature (full and selective-identity attacks), providing either an implication or a separation. Remarkably, we show that the strongest security levels with respect to selective-identity attacks (i.e. chosen-ciphertext security) do not imply the weakest full-identity security level (i.e. one-wayness). With the aim of comprehensiveness, notions of security for IBE in the context of encryption of multiple messages and/or to multiple receivers are finally presented, as well as their relationship with the standard IBE security notion. The results obtained substantiate indistinguishability against full-identity chosen ciphertext attacks as the appropriate security notion for IBE.

Keywords: foundations, identity-based encryption, one-wayness, indistinguishability, non-malleability, semantic security, selective-identity attacks, full-identity attacks, implications and separations.

1 Introduction

The concept of Identity Based Encryption (IBE) was proposed by Shamir in [Sha85], aimed at simplifying certificate management in e-mail related systems. The idea is that an arbitrary string such as an e-mail address or a telephone number could serve as a public key for an encryption scheme. Once a user U receives a communication encrypted using its identity id_U, the user authenticates itself to a Key Generation Center (KGC) from which it obtains the corresponding private key d_U.

The problem was not satisfactorily solved until the work by Boneh and Franklin [BF03]. They proposed formal security notions for IBE systems and designed a secure IBE scheme. Since then, IBE has attracted a lot of attention, and a large number of IBE schemes and related systems (such as [Gen03,AP03]) have been proposed. So far, only the indistinguishability based security notions proposed in [BF03], as well as the variations obtained from [CHK03], have been considered in the literature, and it has not been investigated whether these definitions are appropriate. To better understand what does mean the "appropriateness" of an encryption security notion it is worthwhile to remind how the standard security notion for Public Key Encryption (PKE) was adopted.

A look at PKE security notions. A useful way to describe the security of a cryptographic scheme is by combining the possible *goals* and *attack models*. In the case of PKE schemes, the most important goals considered are: *indistinguishability* (IND), *semantic security* (SS) [GM84] and *non-malleability* (NM) [DDN00]. Semantic security formalizes the inability of an adversary to learn any information about the plaintext m hidden in a challenge ciphertext c, while indistinguishability is a technical goal, aimed at capturing a strong form of privacy and being easier to work and reason with than semantic security. Non-malleability formalizes that an adversary can not build up a ciphertext $c' \neq c$ whose decryption is meaningfully related to m. This notion models the tamper-proofness of a scheme. Regarding attacks, *chosen-plaintext attacks* (CPA) [GM84] and *chosen-ciphertext attacks* (CCA) [RS92] are the most well-known models. Under CPA the adversary is given the public key of the scheme, and thus can encrypt messages on its own, while in CCA the adversary gets in addition access to a decryption oracle, to which it can submit any ciphertext of its choice except for the challenge ciphertext c. Combining these goals and attacks one obtains six security notions: IND-CPA, IND-CCA, SS-CPA, SS-CCA, NM-CPA, NM-CCA.

The equivalence between IND-CPA and SS-CPA was studied in [GM84], and the relations between IND and NM under any form of attack were presented in [BDPR98]. In particular, they showed that NM implies IND under any attack form considered there, but that the opposite direction only holds for the CCA case. For this reason, IND-CCA was considered the right notion of security for general purpose PKE. However, in [BDPR98] was emphasized that a definition for SS-CCA had not been proposed yet, and thus was not known if IND-CCA and SS-CCA were actually equivalent notions. Despite this fact, it became “cryptographic folklore” that those notions were equivalent, and this equivalence was even explicitly used in the literature, for instance in [SJ00,KI01,DT03]. Finally, it was not until [WSI02,GLN02] that an SS-CCA definition was given and proved to be equivalent to IND-CCA.

State of the art on IBE security notions. The first IBE security notions were proposed in [BF03]. These new notions were inspired on the existing PKE definitions, with the novelties that the adversary, regardless of the attack model, is given access to an *extraction oracle*, which on input an identity $\mathsf{id} \in \{0,1\}^*$ outputs the corresponding private key. Moreover, the adversary selects the identity $\mathsf{id}_{\mathsf{ch}}$ on which it wants to be challenged, so that now the challenge consists of a pair $(\mathsf{id}_{\mathsf{ch}}, c)$, where c denotes a ciphertext encrypted under identity $\mathsf{id}_{\mathsf{ch}}$. In [BF03] the adversary is allowed to adaptively select $\mathsf{id}_{\mathsf{ch}}$, probably depending on the information received so far (such as decryption keys), while in [CHK03] the adversary must commit ahead of time to the challenge identity. The latter model is referred to as *selective-identity* attacks (sID), while the original model is called *full-identity* attacks (ID). With respect to IBE goals, only one-wayness and indistinguishability [BF03] have been defined so far. Thus, the security definitions mostly considered up to now in the literature are: IND-ID-CPA, IND-sID-CPA,IND-ID-CCA,IND-sID-CCA.

Currently, IND-ID-CCA is thought to be the right security notion for IBE. However, there is no work, in the sense of [BDPR98], aiming at establishing the strength of this security notion, and it also lacks to relate the full-identity and the (purportedly) weaker selective-identity models. Moreover, although no definition for semantic security in the context of IBE encryption has been given to the best of our knowledge, in the original work by Boneh and Franklin the security notions IND-ID-CPA, (IND-ID-CCA) are presented as equivalent to SS-ID-CPA, (SS-ID-CCA).

Our contributions. Building from the works [BDPR98,GLN02,BF03], we investigate the security foundations underlying IBE encryption. For this purpose, we define the goals of semantic security and non-malleability for IBE, building from well-known public key encryption goals. These definitions are presented in Section 3.

With respect to the relation between selective and full-identity attacks, we show in Section 4 that the strongest security levels with respect to sID attacks (i.e. chosen-ciphertext security) do not even imply the weakest ID security level (i.e. one-wayness). It turns out then that selective-identity security is a strictly weaker security requirement than full-identity security. Notwithstanding, sID security suffices for other purposes, for instance for building IND-CCA PKE schemes [CHK04], and there exists an efficient generic transformation in the Random Oracle Model [BR93] from sID security to ID security [BB04a]. There are several efficient schemes in the literature meeting full-identity security in the Random Oracle Model [BR93], such as [BF03,BB04a,LQ05,Gal05,CC05]. Currently, only the works [BB04a,BB04b,Wat05] show schemes with ID security in the standard model.

In Section 5, we compare the full-identity security notions resulting from our work, providing either an implication or a separation. Our results are summarized in Table 1. As in [BDPR98], this diagram must be seen as a directed graph. An arrow is an implication, and there is path between **A** and **B** if and only if the security notion **A** implies the security notion **B**. A hatched arrow represents a separation which is proved in this paper. Dotted arrows refer to trivial implications. For each pair of notions we obtain an implication or a separation, which is either explicitly found in the diagram or deduced from it. For instance, it turns out that IND-ID-CPA does not imply IND-ID-CCA. Otherwise, following the arrows in the diagram, there would be a path between IND-ID-CPA and NM-ID-CPA, which is impossible due to Theorem 11. As a particular case, we show that SS-ID-CCA and IND-ID-CCA are equivalent under our definition, thus proving the intuition stated in [BF03]. Recently it has come to our attention that in an independent work, [ACH+05] shows similar results to those we present in this section.

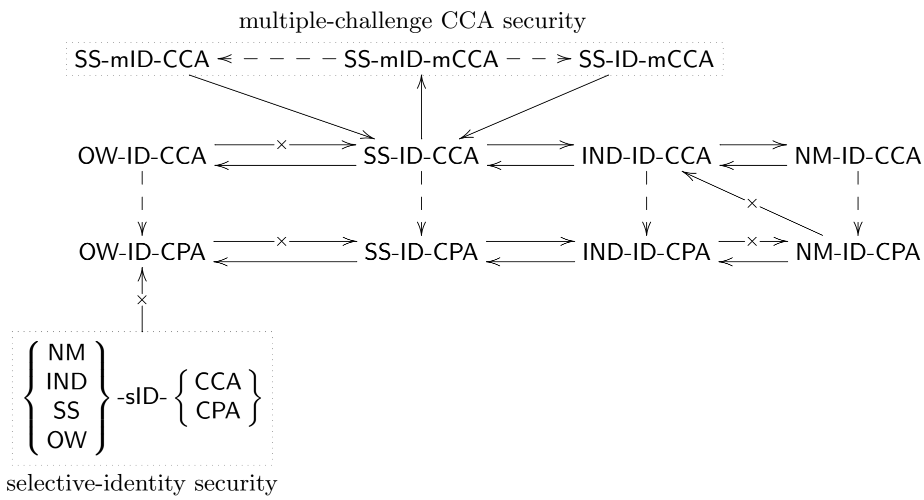

Table 1. Relations among security notions

In the last place, we study in Section 6 the robustness of IND-ID-CCA secure schemes in the context of encryption of multiple messages and/or to multiple receivers. Concretely, inspired by [GLN02], we propose several new attack models for the case of active adversaries: *multiple-identity* (mID-CCA) attacks (the adversary can adaptively query for encryptions of the same plaintext under different identities)[1]; *multiple-plaintext* (ID-mCCA) attacks (the adversary chooses one fixed identity, and can adaptively query encryption of different plaintexts under that identity) and *multiple-identity-plaintext* attacks (mID-mCCA) (the adversary can adaptively query encryption of different plaintexts under different identities). It is shown that any IND-ID-CCA scheme also meets those stronger security levels.

The proof techniques used throughout the paper are indebted to the techniques used in [BDPR98] and [GLN02]. Still, the fact that PKE and IBE are essentially different cryptographic primitives, makes some subtleties to appear, which demand to be carefully examined. For instance, a strong separation such as the one we present between sID and ID security, is not known for PKE security notions. As a general conclusion, the broad body of evidences presented in this work confirm indistinguishability against full-identity chosen-ciphertext attacks as the appropriate security notion for IBE.

2 Preliminaries

We start by fixing some notation and recalling basic concepts.

Algorithmic notation. Assigning a value a to a variable x will be in general denoted by $x \leftarrow a$. If A is a non-empty set, then $x \leftarrow A$ denotes that x has been uniformly chosen in A. If D is a probability distribution over A, then $x \leftarrow D$ means that x has been chosen in A by sampling the distribution D. Finally, if $\mathcal{A}$ is an algorithm, $x \leftarrow \mathcal{A}$ means that $\mathcal{A}$ has been executed on some specified input and its output has been assigned to the variable x.

Negligible functions. The class of negligible functions on a parameter $\ell \in \mathbb{Z}^+$, denoted as $\mathsf{negl}(\ell)$, is the set of the functions $\epsilon : \mathbb{Z}^+ \to \mathbb{R}^+$ such that, for any polynomial $p \in \mathbb{R}[\ell]$, there exist $M \in \mathbb{R}^+$ such that $\epsilon(\ell) < \frac{M}{p(\ell)}$ for all $\ell \in \mathbb{Z}^+$. Let $\mathsf{poly}(\ell)$ the class of functions $p : \mathbb{Z}^+ \to \mathbb{R}^+$ upper bounded in $\mathbb{Z}^+$ by some polynomial in $\mathbb{R}[\ell]$. Hereafter, $U(\ell)$ denotes the uniform distribution on $\{0,1\}^\ell$.

Set sequences. As usual, $\{0,1\}^\star$ and $\{0,1\}^\ell$ will respectively denote the set of all finite binary strings and the set of binary strings with length ℓ. If $m \in \{0,1\}^\star$, then $|m|$ denotes its length. A string set sequence, $\mathcal{M} = \{\mathcal{M}_\ell\}_{\ell \in \mathbb{Z}^+}$, is a *polynomial size set* (hereafter simply named as set) if there exist an integer valued function $p_\mathcal{M}(\ell) \in \mathsf{poly}(\ell)$ such that $\mathcal{M}_\ell \subseteq \{0,1\}^{p_\mathcal{M}(\ell)}$ for all $\ell \in \mathbb{Z}^+$. The cardinality of a set sequence A (as a function of ℓ) will be denoted by $|A|$.

Miscellaneous notations. We denote a sequence of bit strings by $\overrightarrow{-}$ symbol. For example, the notation $\overrightarrow{m} \leftarrow D(\mathsf{mk}, \mathsf{id_{ch}}, \overrightarrow{c})$ means that $\overrightarrow{m}$ is the sequence of the results when we apply the algorithm $D(\mathsf{mk}, \mathsf{id_{ch}}, -)$ to each component of $\overrightarrow{c}$. The bitwise complement of a bit string x is denoted by $\overline{x}$.

Identity based encryption (IBE). An IBE scheme is specified by four probabilistic polynomial time (PPT) algorithms:

Setup (S) takes a security parameter 1^ℓ and returns the system parameters pms and master-key mk. The parameters pms include includes the security parameter 1^ℓ; the description of sets $\mathcal{ID}, \mathcal{M}, \mathcal{C}$, which denote the set of identities, messages and ciphertexts respectively. pms is publicly available, while the mk is kept secret by the KGC.

Extract (X) takes as inputs pms, mk and $\mathsf{id} \in \mathcal{ID}_\ell$; it outputs the private key d_id corresponding to the identity id.

[1] This security definition has been previously considered in [BSS05], but no proof of equivalence to IND-ID-CCA was given. Moreover, the attack we consider is stronger since it gives more power to the adversary.

Encrypt (E takes as inputs pms, an identity $\mathsf{id} \in \mathcal{ID}_\ell$ and $m \in \mathcal{M}_\ell$. It returns a ciphertext $c \in \mathcal{C}_\ell$.

Decrypt (D) takes as inputs pms, a private key d_{id} and $c \in \mathcal{C}_\ell$, and it returns $m \in \mathcal{M}_\ell$ or reject when c is not a legitimate ciphertext. For the sake of consistency, these algorithms must satisfy that for all $\mathsf{id} \in \mathcal{ID}_\ell, m \in \mathcal{M}_\ell$,

$$D(\mathsf{pms}, d_{\mathsf{id}}, c) = m, \quad \text{where} \quad c = E(\mathsf{pms}, \mathsf{id}, m) \quad \text{and} \quad d_{\mathsf{id}} = X(\mathsf{pms}, \mathsf{mk}, \mathsf{id})$$

2.1 Attacks models for identity based encryption

The attack models mostly used for IBE in the literature can be classified by combining the following items:

- The adversary is given/not-given to access a decryption oracle;
- The adversary can adaptively select the identity it wants to attack or it is forced to commit ahead of time to the identity under attack,

which results then in 4 different attacks models. Regardless of the attack model under consideration, the adversary is supposed to have access to an extraction oracle, which on input an identity id it outputs the corresponding decryption key d_{id}. The decryption oracle, on inputs an identity id and a ciphertext c, it outputs $D(\mathsf{pms}, d_{\mathsf{id}}, c)$. The adversary can query these oracles polynomially many times and in an adaptive manner [BF03]. There is no need to include in the attack model an encryption oracle, since in an IBE scheme the adversary is able to simulate this oracle after knowing the public parameters pms of the scheme. The attack models under consideration are referred to as sID-CPA,ID-CPA,sID-CCA,ID-CCA.

2.2 One-wayness - OW

In the following, one-wayness security definitions for IBE are presented. As far as we know, only one-wayness against full-identity chosen-plaintext attacks (referred to as OW-ID-CPA in the following definition) has been previously considered in the literature. Following the notation in [BDPR98], $\mathcal{O}_i = \varepsilon$ means $\mathcal{O}_i$ is the function which returns the empty string ε on any input.

Definition 1 (OW-{ID, sID}-{CCA, CPA}) *Let $\Pi = (S, X, E, D)$ be an IBE scheme, and let $\mathcal{A} = (\mathcal{A}_0, \mathcal{A}_1, \mathcal{A}_2)$ be any 3-tuple of PPT oracle algorithms. For* $\mathsf{ATK} = \mathsf{sID\text{-}CPA}, \mathsf{ID\text{-}CPA}, \mathsf{sID\text{-}CCA}, \mathsf{ID\text{-}CCA}$, *we say Π is* OW-ATK *secure if for any 3-tuple of PPT oracle algorithms $\mathcal{A}$*

$$\Pr\left[m' = m \;\middle|\; \begin{array}{c} (\mathsf{id}, \gamma) \leftarrow \mathcal{A}_0(1^\ell) \\ (\mathsf{pms}, \mathsf{mk}) \leftarrow S(1^\ell); \\ (\mathsf{id}_{\mathsf{ch}}, \sigma) \leftarrow \mathcal{A}_1^{\mathcal{O}_1, \mathcal{O}_2}(\mathsf{pms}, \mathsf{id}, \gamma) \\ m \leftarrow P(U_{\mathsf{poly}}(\ell)); \quad c \leftarrow E(\mathsf{pms}, \mathsf{id}_{\mathsf{ch}}, m); \\ m' \leftarrow \mathcal{A}_2^{\mathcal{O}_1, \mathcal{O}_2}(\sigma, (\mathsf{id}_{\mathsf{ch}}, c)) \end{array} \right] \in \mathsf{negl}(\ell),$$

and

If ATK=sID-CPA	*then*	$\mathcal{O}_1(\cdot) = X(\mathsf{pms}, \mathsf{mk}, \cdot)$,	$\mathcal{O}_2(\cdot) = \varepsilon$	*and*	$\mathsf{id}_{\mathsf{ch}} := \mathsf{id}$
If ATK=ID-CPA	*then*	$\mathcal{O}_1(\cdot) = X(\mathsf{pms}, \mathsf{mk}, \cdot)$,	$\mathcal{O}_2(\cdot) = \varepsilon$		
If ATK=sID-CCA	*then*	$\mathcal{O}_1(\cdot) = X(\mathsf{pms}, \mathsf{mk}, \cdot)$,	$\mathcal{O}_2(\cdot) = D(\mathsf{pms}, \mathsf{mk}, \cdot)$	*and*	$\mathsf{id}_{\mathsf{ch}} := \mathsf{id}$
If ATK=ID-CCA	*then*	$\mathcal{O}_1(\cdot) = X(\mathsf{pms}, \mathsf{mk}, \cdot)$,	$\mathcal{O}_2(\cdot) = D(\mathsf{pms}, \mathsf{mk}, \cdot)$		

In the above expression σ and γ denote some state information. By giving mk *as input to the decryption algorithm D we mean that it can decrypt ciphertexts related to any identity. Additional requirements are that neither $\mathcal{A}_1$ nor $\mathcal{A}_2$ are allowed to query $\mathcal{O}_1$ on the challenge identity* $\mathsf{id_{ch}}$*, and $\mathcal{A}_2$ can not query $\mathcal{O}_2$ on the challenge pair* $(\mathsf{id_{ch}}, c)$*. These queries may be asked adaptively, that is, each query may depend on the answers obtained to the previous queries.*

2.3 Indistinguishability - IND

In the following we recall the indistinguishability security notions obtained from the attacks previously considered in the literature.

Definition 2 (IND-{ID, sID}-{CCA, CPA}) *Let $\Pi = (S, X, E, D)$ be an IBE scheme, and let $\mathcal{A} = (\mathcal{A}_0, \mathcal{A}_1, \mathcal{A}_2)$ be any 3-tuple of PPT oracle algorithms. For* $\mathsf{ATK} = \mathsf{sID\text{-}CPA}, \mathsf{ID\text{-}CPA}, \mathsf{sID\text{-}CCA}, \mathsf{ID\text{-}CCA}$*, we say Π is* IND-ATK *secure if for any 3-tuple of PPT oracle algorithms $\mathcal{A}$,*
$\left|p_\ell^{(1)} - p_\ell^{(2)}\right| \in \mathsf{negl}(\ell)$*, where*

$$p_\ell^{(i)} = \Pr\left[v = 1 \;\middle|\; \begin{array}{c} (\mathsf{id}, \gamma) \leftarrow \mathcal{A}_0(1^\ell) \\ (\mathsf{pms}, \mathsf{mk}) \leftarrow S(1^\ell); \\ ((m^{(1)}, m^{(2)}, \mathsf{id_{ch}}), \sigma) \leftarrow \mathcal{A}_1^{\mathcal{O}_1, \mathcal{O}_2}(\mathsf{pms}, \mathsf{id}, \gamma) \\ c \leftarrow E(\mathsf{pms}, \mathsf{id_{ch}}, m^{(i)}); \\ v \leftarrow \mathcal{A}_2^{\mathcal{O}_1, \mathcal{O}_2}(\sigma, (\mathsf{id_{ch}}, c)) \end{array} \right]$$

In the last expression, the oracles $\mathcal{O}_1, \mathcal{O}_2$ as well as the access to them are as in Definition 1. Additionally, $m^{(1)}$ and $m^{(2)}$ are required to have the same length; neither $\mathcal{A}_1$ nor $\mathcal{A}_2$ are allowed to query $\mathcal{O}_1$ on the challenge identity $\mathsf{id_{ch}}$*, and $\mathcal{A}_2$ can not query $\mathcal{O}_2$ on the challenge pair* $(\mathsf{id_{ch}}, c)$*. These queries may be asked adaptively, that is, each query may depend on the answers obtained to the previous queries.*

3 Semantic Security and Non-Malleability for Identity Based Encryption

In this section, semantic security as well as non-malleability for IBE schemes are defined for the first time to the best of our knowledge. These definitions are obtained by adapting the definitions of semantic security [GM84,GLN02] and non-malleability [BDPR98,BS99] for PKE to the attack scenario proposed in [BF03].

3.1 Semantic Security - SS

In the following we rephrase the definitions in [GLN02] for the IBE setting. In our scenario, a CPA attacker is given access to one oracle for extraction of private keys, while a CCA adversary is additionally given access to a decryption oracle. The attack is broken in two stages:

Stage 1: The adversary conducts some computation using its oracles, and terminates this stage by giving a challenge template. This template consists of challenge identity $\mathsf{id_{ch}}$ and three circuits (P, L, F): P is a *sampling circuit*, and L, F are circuits with a number of input bits that equals the number of output bits in P. The idea is that P specifies a probability space on plaintexts, while L specifies partial information (i.e., "information leak") regarding the plaintext that is given to the adversary, and F specifies partial information (regarding the plaintext) that the adversary claims to be able to learn.

Stage 2: In the second stage the adversary is given an encryption c of a plaintext m under an identity $\mathsf{id_{ch}}$ along with $L(m)$, where m is selected according to P and the adversary does not know the decryption key for $\mathsf{id_{ch}}$. Both CPA and CCA adversaries are not allowed to query the extraction oracle on $\mathsf{id_{ch}}$, while the CCA adversary can not query the decryption oracle on the pair $(\mathsf{id_{ch}}, c)$.

Roughly speaking, an IBE scheme is said to be *semantically secure under* CPA *(respectively* CCA*)* if for every efficient CPA (respectively CCA) attacker as above, there exists a corresponding benign adversary that "performs as well" without seeing the ciphertexts. Concretely, the benign adversary is not given any oracle access, it produces a challenge template (P, L, F) and is supposed to guess $F(m)$ given $L(m)$. Additional requirements are that the benign adversary is supposed to produce (P, L, F) according to the same distribution as the real adversary, and to be as successful as the real adversary in guessing $F(m)$.

Note that the benign adversary models an ideal situation in which it produces the same challenge template as the real adversary, but it is given a "perfectly secure encryption" of the plaintext m, that is, it is given nothing [Gol93].

Definition 3 (SS-{ID, sID}-{CCA, CPA}) *Let $\Pi = (S, X, E, D)$ be an IBE scheme, and let $\mathcal{B} = (\mathcal{B}_0, \mathcal{B}_1, \mathcal{B}_2)$ be a 3-tuple of PPT oracle algorithms. For* $\mathsf{ATK} = \mathsf{sID\text{-}CPA}, \mathsf{ID\text{-}CPA}, \mathsf{sID\text{-}CCA}, \mathsf{ID\text{-}CCA}$, *we say Π is* SS-ATK *secure if for any 3-tuple of PPT oracle algorithms $\mathcal{B}$ there exists a 3-tuple of PPT algorithms $\mathcal{B}' = (\mathcal{B}'_0, \mathcal{B}'_1, \mathcal{B}'_2)$, such that the following conditions hold:*

$$\mathsf{Pr}\left[v = F(m) \;\middle|\; \begin{array}{c} (\mathsf{id}, \gamma) \leftarrow \mathcal{B}_0(1^\ell); \\ (\mathsf{pms}, \mathsf{mk}) \leftarrow S(1^\ell); \\ \big((P, \mathsf{id_{ch}}, L, F), \sigma\big) \leftarrow \mathcal{B}_1^{\mathcal{O}_1, \mathcal{O}_2}(\mathsf{pms}, \mathsf{id}, \gamma) \\ (c, \beta) \leftarrow \big(E(\mathsf{pms}, \mathsf{id_{ch}}, m), L(m)\big), \quad \textit{where} \\ m \leftarrow P(U_{\mathsf{poly}(\ell)}); \\ v \leftarrow \mathcal{B}_2^{\mathcal{O}_1, \mathcal{O}_2}(\sigma, (\mathsf{id_{ch}}, c), \beta) \end{array} \right]$$
$$< \mathsf{Pr}\left[v = F(m) \;\middle|\; \begin{array}{c} \big((P, L, F), \sigma'\big) \leftarrow \mathcal{B}'_1(1^\ell); \\ m \leftarrow P(U_{\mathsf{poly}(\ell)}); \\ v \leftarrow \mathcal{B}'_2\big(\sigma', L(m)\big) \end{array} \right] + \varepsilon(\ell),$$

where $\varepsilon(\ell) \in \mathsf{negl}(\ell)$, and for every ℓ the (P, L, F) part in the random variables $\mathcal{B}'_1(1^\ell)$ and $\mathcal{B}_1^{\mathcal{O}_1,\mathcal{O}_2}(\mathsf{pms}, \mathsf{id})$ are identically distributed. The oracles $\mathcal{O}_1, \mathcal{O}_2$ as well as the access to them are as in Definition 1. Additionally, neither $\mathcal{B}_1$ nor $\mathcal{B}_2$ are allowed to query $\mathcal{O}_1$ on the challenge identity $\mathsf{id_{ch}}$, and $\mathcal{B}_2$ can not query $\mathcal{O}_2$ on the challenge pair $(\mathsf{id_{ch}}, c)$. These queries may be asked adaptively.

3.2 Non-malleability - NM

The notion of non-malleability for PKE was introduced in [DDN00]. In [BDPR98] an alternative definition was proposed, which was shown to be equivalent to the original one [BS99]. The new definition was simpler since it did not use simulators. Intuitively, an encryption scheme is non-malleable if, given a ciphertext, the adversary cannot conjure up another ciphertext whose decryption is meaningfully related to the decryption of the given one. This notion models the tamper-proofness of a scheme.

In this paper non-malleability is defined without using simulators, adapting the definition for public-key schemes in [BDPR98] to the IBE framework. After receiving a challenge pair $(\mathsf{id_{ch}}, c)$, the aim of the adversary is to output the description of a relation R and a vector of ciphertexts $\overrightarrow{c}$ (no component of which is the challenge ciphertext c) such that the relation $R(m, \overrightarrow{m})$ holds, where $\overrightarrow{m} \leftarrow D(\mathsf{mk}, \mathsf{id_{ch}}, \overrightarrow{c})$ and m is the plaintext hidden in the challenge ciphertext. The adversary is successful if it can do this with probability significantly greater than that with which $R(m_0, \overrightarrow{m})$ holds for some random m_0 of the same length as m. In the following the notation $D(\mathsf{mk}, \mathsf{id_{ch}}, c) = \perp$ means that c is not a legitimate ciphertext,

Definition 4 (NM-{ID, sID}-{CCA, CPA}) *Let $\Pi = (S, X, E, D)$ be an IBE scheme, and let $\mathcal{C} = (\mathcal{C}_0, \mathcal{C}_1, \mathcal{C}_2)$ be any 3-tuple of PPT oracle algorithms. For* ATK = sID-CPA, ID-CPA, sID-CCA, ID-CCA, *we say Π is* NM-ATK *secure if for any 3-tuple of PPT oracle algorithms $\mathcal{C}$, the advantage*

$$\mathsf{Pr}\left[v = 1 \;\middle|\; \begin{array}{c} (\mathsf{id}, \gamma) \leftarrow \mathcal{C}_0(1^\ell); \qquad (\mathsf{pms}, \mathsf{mk}) \leftarrow S(1^\ell); \\ (P, \mathsf{id_{ch}}, \sigma) \leftarrow \mathcal{C}_1^{\mathcal{O}_1, \mathcal{O}_2}(\mathsf{pms}, \mathsf{id}, \gamma); \\ m \leftarrow P(U_{\mathsf{poly}(\ell)}); \qquad c \leftarrow E(\mathsf{pms}, \mathsf{id_{ch}}, m); \\ (R, \overrightarrow{c}) \leftarrow \mathcal{C}_2^{\mathcal{O}_1, \mathcal{O}_2}(\sigma, (\mathsf{id_{ch}}, c)); \qquad \overrightarrow{m} \leftarrow D(\mathsf{mk}, \mathsf{id_{ch}}, \overrightarrow{c}); \\ \texttt{if} \quad c \notin \overrightarrow{c} \wedge \perp \notin \overrightarrow{m} \wedge R(m, \overrightarrow{m}) \quad \texttt{then} \quad v \leftarrow 1 \quad \texttt{else} \quad v \leftarrow 0; \end{array} \right]$$
$$- \mathsf{Pr}\left[v = 1 \;\middle|\; \begin{array}{c} (\mathsf{id}, \gamma) \leftarrow \mathcal{C}_0(1^\ell); \qquad (\mathsf{pms}, \mathsf{mk}) \leftarrow S(1^\ell); \\ (P, \mathsf{id_{ch}}, \sigma) \leftarrow \mathcal{C}_1^{\mathcal{O}_1, \mathcal{O}_2}(\mathsf{pms}, \mathsf{id}, \gamma); \\ m, m_0 \leftarrow P(U_{\mathsf{poly}(\ell)}); \qquad c \leftarrow E(\mathsf{pms}, \mathsf{id_{ch}}, m); \\ (R, \overrightarrow{c}) \leftarrow \mathcal{C}_2^{\mathcal{O}_1, \mathcal{O}_2}(\sigma, (\mathsf{id_{ch}}, c)); \qquad \overrightarrow{m} \leftarrow D(\mathsf{mk}, \mathsf{id_{ch}}, \overrightarrow{c}); \\ \texttt{if} \quad c \notin \overrightarrow{c} \wedge \perp \notin \overrightarrow{m} \wedge R(m_0, \overrightarrow{m}) \quad \texttt{then} \quad v \leftarrow 1 \quad \texttt{else} \quad v \leftarrow 0; \end{array} \right]$$

is negligible as a function on ℓ. In the last expression, the oracles $\mathcal{O}_1, \mathcal{O}_2$ as well as the access to them is defined as in Definition 1, and P specifies a probability space on $\mathcal{M}_\ell$. Additionally, neither $\mathcal{C}_1$ nor $\mathcal{C}_2$ are allowed to query $\mathcal{O}_1$ on the challenge identity $\mathsf{id_{ch}}$, and $\mathcal{C}_2$ can not query $\mathcal{O}_2$ on the challenge pair $(\mathsf{id_{ch}}, c)$. These queries may be asked adaptively.

Remark 1. The security notions described here and in the previous section can be easily defined in the Random Oracle Model [BR93], where all parties have access to a random function H from strings to strings. The definitions are modified by including in the experiments defining the advantage an initial step, in which some random functions H are chosen from the set of all functions from some appropriate domain to appropriate range. Then H-oracles are provided to the algorithms and they may depend on H. It is easily verified that all of the implications and separations provided in this work also hold in the Random Oracle Model.

4 A separation between selective-identity and full-identity security notions

In this section we show a strong separation between selective-identity and full-identity attack scenarios. The separation is as follows: for any goal $\in$ {IND, SS, NM}, it turns out that goal-sID-CCA *does not imply* OW-ID-CPA, that is, the strongest selective-identity security levels are strictly weaker than the weakest full-identity security level.

Theorem 5 *For any* goal $\in$ {IND, SS, NM}*,* goal-sID-CCA *does not imply* OW-ID-CPA*.*

Proof: Assume that there exists an IBE scheme $\Pi = (S, X, E, D)$ which is goal-sID-CCA secure for some goal $\in$ {IND, SS, NM} (otherwise the claim is trivially true). We construct another IBE scheme $\Pi' = (S', X', E', D')$ which is goal-sID-CCA but not OW-ID-CPA, whose existence proves the theorem. The scheme Π' is defined as follows:

$S'(1^\ell)$	$X'(\mathsf{pms}', \mathsf{mk}', \mathsf{id})$	$E'(\mathsf{pms}', \mathsf{id}, m)$	$D'(\mathsf{pms}', d_{\mathsf{id}}, C)$
$(\mathsf{pms}, \mathsf{mk}) \leftarrow S(1^\ell)$; $\mathsf{id}^+ \leftarrow \mathcal{ID}_\ell$; $d^+ \leftarrow X(\mathsf{pms}, \mathsf{mk}, \mathsf{id}^+)$; $\mathsf{pms}' \leftarrow (\mathsf{pms}, \mathsf{id}^+, d^+)$; `return` $(\mathsf{pms}', \mathsf{mk})$	`return` $X(\mathsf{pms}, \mathsf{mk}, \mathsf{id})$;	`return` $E(\mathsf{pms}, \mathsf{id}, m)$	`return` $D(\mathsf{pms}, d_{\mathsf{id}}, C)$

It is trivial to check that Π' qualifies as an IBE scheme if Π does. From the definition of Π' and the definition of sID attacks[2], it also holds that $\mathcal{ID}' = \mathcal{ID} := \{\mathcal{ID}_\ell\}_{\ell \in \mathbb{Z}^+} = \{\{0,1\}^{p(\ell)}\}_{\ell \in \mathbb{Z}^+}$ for a certain $p(\ell) \in \mathsf{poly}(\ell)$. Π' is not OW-ID-CPA due to the following successful adversary $\mathcal{A} = (\mathcal{A}_0, \mathcal{A}_1, \mathcal{A}_2)$:

```
Algorithm  A_1^{O'_1,O'_2}(pms')
   id_ch ← id+;      σ ← (d+, pms);
   m^(1), m^(2) ← M_ℓ, s.t. |m^(1)| = |m^(2)|;
   return  ((m^(1), m^(2), id_ch), σ)
```

```
Algorithm  A_2^{O'_1,O'_2}(σ, id_ch, c)
   return  D(pms, d+, c)
```

A simple calculation shows that the OW-ID-CPA advantage of the adversary $\mathcal{A} = (\mathcal{A}_0, \mathcal{A}_1, \mathcal{A}_2)$ is 1. The basic idea is that $\mathcal{A}_1$ knows the decryption key related to id^+ once it gets pms'. Then it sets $\mathsf{id}_{\mathsf{ch}} := \mathsf{id}^+$, it qualifies as a OW-ID-CCA adversary ($\mathcal{A}_1$ did not query id^+ to its oracle $\mathcal{O}'_1$) and finally it can decrypt any ciphertext related to the challenge identity, thus effectively breaking the one-wayness of Π'. It remains to show that the new scheme Π' is secure in the sense of goal-sID-CCA. We argue by contradiction: if we have a successful goal-sID-CCA attacker $\mathcal{C} = (\mathcal{C}_0, \mathcal{C}_1, \mathcal{C}_2)$ for this new scheme, then from that we can construct a successful goal-sID-CCA $\mathcal{B} = (\mathcal{B}_0, \mathcal{B}_1, \mathcal{B}_2)$ attacker for the original scheme Π. The details are given in Appendix **??**. □

Notice that in the previous proof, the algorithm $\mathcal{A}_0$ was not specified, since it is useless in full-identity attack scenarios. In the following we will focus on full-identity scenarios, so the algorithm with subscript 0 in every adversary tuple is dropped for the sake of simplicity.

5 Relations between full-identity security notions

In this section implications and separations for one-wayness, indistinguishability, semantic security and non-malleability for full-identity chosen-plaintext and chosen-ciphertext attacks are presented.

Theorem 6 OW-ATK *does not imply* IND-ATK*, for* ATK = ID-CPA, ID-CCA*.*

[2] See for instance Definition 4 in [CHK04,BK05].

Proof: This proof is straightforward, and the reader is referred to the final version of the paper. □

Theorem 7 IND-ATK *entails* SS-ATK, *for* ATK = ID-CPA, ID-CCA.

Proof: We present the proof for the ID-CCA case. The modification for the ID-CPA case is easy by dropping the access to decryption oracle.

Given an SS-ID-CCA adversary $(\mathcal{B}_1, \mathcal{B}_2)$, we construct its benign simulator $(\mathcal{B}'_1, \mathcal{B}'_2)$ as follows. The algorithm $\mathcal{B}'_2$ simulates $\mathcal{B}_2$ by feeding the encryption of "fake plaintext" $1^{|m|}$ instead of a real plaintext $m \leftarrow P(U_{\mathsf{poly}(\ell)})$: since the scheme is IND-ID-CCA this should not affect the advantage.

```
Algorithm  B'_1(1^ℓ)
    (pms, mk) ← S(1^ℓ);   ((P, id_ch, L, F), σ) ← B_1^{O_1, O_2}(pms);
    n ← (the number of output bits in P);
    return  ((P, L, F), (σ, pms, mk, id_ch, 1^n))
```

```
Algorithm  B'_2((σ, pms, mk, id_ch, 1^n), β)
    v ← B_2^{O_1, O_2}(σ, (id_ch, (E(pms, id_ch, 1^n), β));
    return  v
```

It is obvious that the (P, L, F) part of the output of $\mathcal{B}_1$ and $\mathcal{B}'_1$ are identically distributed. We shall show that the simulator thus defined is as successful as the actual adversary. The advantage of the simulator $(\mathcal{B}'_1, \mathcal{B}'_2)$ is evaluated as follows.

$$\mathsf{Pr}\left[v = F(m) \;\middle|\; \begin{array}{c} (\mathsf{pms}, \mathsf{mk}) \leftarrow S(1^\ell); \\ \big((P, \mathsf{id_{ch}}, L, F), \sigma\big) \leftarrow \mathcal{B}_1^{X_{\mathsf{mk}}, D_{\mathsf{mk}}}(\mathsf{pms}); \\ m \leftarrow P(U_{\mathsf{poly}(\ell)}); \\ v \leftarrow \mathcal{B}_2^{X_{\mathsf{mk}}, D_{\mathsf{mk}}}(\sigma, \mathsf{id_{ch}}, \big(E(\mathsf{pms}, \mathsf{id_{ch}}, 1^{|m|}), L(m)\big)\big) \end{array} \right] \tag{1}$$

This probability is the same as the advantage of the original adversary $(\mathcal{B}_1, \mathcal{B}_2)$, except that $\mathcal{B}_2$ is given the encryption of $1^{|m|}$ instead of m.

Claim 1 *The two ensembles* $V_\ell = \big[\big(\sigma, \mathsf{id_{ch}}, (E(\mathsf{pms}, \mathsf{id_{ch}}, 1^{|m|}), L(m)), F(m)\big) \,\big|\, \mathsf{Exp}\big]$, $W_\ell = \big[\big(\sigma, \mathsf{id_{ch}}, (E(\mathsf{pms}, \mathsf{id_{ch}}, m), L(m)), F(m)\big) \,\big|\, \mathsf{Exp}\big]$, *under the following experiment*

```
Experiment  Exp
    (pms, mk) ← S(1^ℓ);    m ← P(U_poly(ℓ));    ((P, id_ch, L, F), σ) ← B_1^{X_mk, D_mk}(pms);
```

cannot be distinguished by any PPT algorithm $\mathcal{T}^{X_{\mathsf{mk}}, D_{\mathsf{mk}}}\big(\sigma, \mathsf{id_{ch}}, (\alpha, \beta), \gamma\big)$ *which is not allowed to query* $X_{\mathsf{mk}}(\mathsf{id_{ch}})$ *nor* $D_{\mathsf{mk}}(\mathsf{id_{ch}}, \alpha)$.

It suffices to show Claim 1: if it is true, in particular the following $\mathcal{T}$ does not distinguish V_ℓ and W_ℓ.

```
Algorithm  T^{O_1,O_2}(σ, id_ch, (α, β), γ)
    v ← A_2^{O_1,O_2}(σ, id_ch, (α, β));    if  v = γ  then  d ← 1  else  d ← 0;
    return  d
```

Hence the advantage of the simulator and that of the actual adversary are indistinguishable. This proves the theorem.

Claim 1 is proved by contradiction, using indistinguishability. Assume a successful distinguisher $\mathcal{T}$ of V_ℓ and W_ℓ exists. Then we can construct a successful $\mathsf{IND\text{-}ID\text{-}CCA}$ distinguisher $(\mathcal{A}_1, \mathcal{A}_2)$.

$$\begin{array}{|l|}\hline \texttt{Algorithm}\quad \mathcal{A}_1^{O_1,O_2}(\mathsf{pms}) \\ \quad \big((P, \mathsf{id}_{\mathsf{ch}}, L, F), \sigma\big) \leftarrow \mathcal{B}_1^{O_1,O_2}(\mathsf{pms}); \\ \quad m \leftarrow P(U_{\mathsf{poly}(\ell)}); \\ \quad \texttt{return}\quad \big((1^{|m|}, m, \mathsf{id}_{\mathsf{ch}}), (\sigma, L, F, m)\big) \\ \hline \end{array} \qquad \begin{array}{|l|}\hline \texttt{Algorithm}\quad \mathcal{A}_2^{O_1,O_2}\big((\sigma, L, F, m), (\mathsf{id}_{\mathsf{ch}}, \alpha)\big) \\ \quad v \leftarrow \mathcal{T}^{O_1,O_2}(\sigma, \mathsf{id}_{\mathsf{ch}}, (\alpha, L(m)), F(m)); \\ \quad \texttt{return}\quad v \\ \hline \end{array}$$

$(\mathcal{A}_1, \mathcal{A}_2)$ does not make those oracle queries an $\mathsf{IND\text{-}ID\text{-}CCA}$ distinguisher is prohibited to make, because $\mathcal{B}_1$ or $\mathcal{T}$ does not.

For probabilities $p_\ell^{(1)}, p_\ell^{(2)}$ in the definition of $\mathsf{IND\text{-}ID\text{-}CCA}$, the following equations are straightforward.

$$p_\ell^{(1)} = \mathsf{Pr}\left[\mathcal{T}^{X_{\mathsf{mk}}, D_{\mathsf{mk}}}(V_\ell) = 1\right] \qquad\qquad p_\ell^{(2)} = \mathsf{Pr}\left[\mathcal{T}^{X_{\mathsf{mk}}, D_{\mathsf{mk}}}(W_\ell) = 1\right]$$

Therefore the success of $\mathcal{T}$ implies the success of $(\mathcal{A}_1, \mathcal{A}_2)$, which is a contradiction. □

Theorem 8 $\mathsf{SS\text{-}ATK}$ *entails* $\mathsf{IND\text{-}ATK}$*, for* $\mathsf{ATK} = \mathsf{ID\text{-}CPA}, \mathsf{ID\text{-}CCA}$.

Proof: The proof is presented for $\mathsf{ATK{=}ID\text{-}CCA}$. The modification for $\mathsf{ID\text{-}CPA}$ case is easy.

We argue by contradiction. Assume that an IBE scheme (S, X, E, D) has a successful $\mathsf{IND\text{-}ID\text{-}CCA}$ distinguisher $(\mathcal{A}_1, \mathcal{A}_2)$. We shall construct an $\mathsf{SS\text{-}ID\text{-}CCA}$ adversary whose advantage is distinguishably larger than that of any benign simulator. The construction is rather obvious: the first part of the adversary outputs a challenge template $(P, \mathsf{id}_{\mathsf{ch}}, L, F)$ such that

- P chooses one out of $m^{(1)}$ and $m^{(2)}$, the two challenge plaintexts $\mathcal{A}_1$ outputs, with the uniform probability;
- L outputs the constant value so that the benign simulator cannot gain any information about the plaintext;
- $F(m^{(1)}) = 1$ and $F(m^{(2)}) = 0$.

However in the formal proof some subtle details need careful consideration, which can be found below.

We can assume without loss of generality that $\mathcal{A}_2$ always outputs either 0 or 1. If it is not the case we define a new oracle PPT $\mathcal{A}_2'$ which invokes $\mathcal{A}_2$ and outputs 1 if $\mathcal{B}_2$ outputs 1, and outputs 0 otherwise: obviously the pair $(\mathcal{A}_1, \mathcal{A}_2')$ is again a successful distinguisher. We can also assume, from the success of $\mathsf{IND\text{-}ID\text{-}CCA}$ distinguisher $(\mathcal{A}_1, \mathcal{A}_2)$, that for some polynomial q and infinitely many ℓ's the following holds: $p_\ell^{(1)} - p_\ell^{(2)} \geq \frac{1}{q(\ell)}$. Note that we no longer take the absolute value.

Additionally, for the technical reason, we assume that the algorithm $\mathcal{A}_1$ always outputs distinct challenge plaintexts (i.e. $m^{(1)} \neq m^{(2)}$). Otherwise we define a new distinguisher

$(\mathcal{A}'_1, \mathcal{A}'_2)$ as follows.

$$
\begin{array}{|l|}
\hline
\texttt{Algorithm} \quad \mathcal{A}'^{O_1,O_2}_1(\mathsf{pms}) \\
\quad \big((m^{(1)}, m^{(2)}, \mathsf{id}_{\mathsf{ch}}), \sigma\big) \leftarrow \mathcal{A}_1^{O_1,O_2}(\mathsf{pms}); \\
\quad \texttt{if} \quad m^{(1)} \neq m^{(2)} \quad \texttt{then} \quad d \leftarrow 0 \quad \texttt{else} \quad d \leftarrow 1; \\
\quad \texttt{if} \quad d = 0 \quad \texttt{then} \quad m' \leftarrow m^{(2)} \quad \texttt{else} \quad m' \leftarrow \overline{m^{(1)}}; \\
\quad \texttt{return} \quad \big((m^{(1)}, m', \mathsf{id}_{\mathsf{ch}}), (\sigma, d)\big) \\
\hline
\end{array}
$$

$$
\begin{array}{|l|}
\hline
\texttt{Algorithm} \quad \mathcal{A}'^{O_1,O_2}_2\big((\sigma, d), (\mathsf{id}_{\mathsf{ch}}, c)\big) \\
\quad \texttt{if} \quad d = 0 \quad \texttt{then} \quad v \leftarrow \mathcal{A}_2^{O_1,O_2}(\sigma, (\mathsf{id}_{\mathsf{ch}}, c)) \quad \texttt{else} \quad v \leftarrow U_1; \\
\quad \texttt{return} \quad v \\
\hline
\end{array}
$$

Obviously the advantage of $(\mathcal{A}'_1, \mathcal{A}'_2)$ is identical to that of $(\mathcal{A}_1, \mathcal{A}_2)$, and $\mathcal{A}'_1$ is ensured to output distinct challenge plaintexts,

Using the distinguisher we construct an SS-ID-CCA adversary $(\mathcal{B}_1, \mathcal{B}_2)$ as follows.

$$
\begin{array}{|l|}
\hline
\texttt{Algorithm} \quad \mathcal{B}_1^{O_1,O_2}(\mathsf{pms}) \\
\quad \big((m^{(1)}, m^{(2)}, \mathsf{id}_{\mathsf{ch}}), \sigma\big) \leftarrow \mathcal{A}_1^{O_1,O_2}(\mathsf{pms}); \\
\quad n \leftarrow \left|m^{(1)}\right|; \\
\quad P \leftarrow \text{a circuit with one input bit such that } P(0) = m^{(1)} \text{ and } P(1) = m^{(2)}; \\
\quad L \leftarrow \text{a circuit with } n \text{ input bits, which outputs constantly 0}; \\
\quad F \leftarrow \text{a circuit which outputs 1 for input } m^{(1)}, \text{ 0 for the other inputs}; \\
\quad \texttt{return} \quad \big((P, \mathsf{id}_{\mathsf{ch}}, L, F), \sigma\big) \\
\hline
\end{array}
$$

$$
\begin{array}{|l|}
\hline
\texttt{Algorithm} \quad \mathcal{B}_2^{O_1,O_2}(\sigma, \mathsf{id}_{\mathsf{ch}}, (\alpha, \beta)) \\
\quad v \leftarrow \mathcal{A}_2^{O_1,O_2}(\sigma, (\mathsf{id}_{\mathsf{ch}}, \alpha)); \\
\quad \texttt{return} \quad v \\
\hline
\end{array}
$$

Note that, by our assumption that $m^{(1)} \neq m^{(2)}$ in the output of $\mathcal{A}_1$, for an output $\big((P, \mathsf{id}_{\mathsf{ch}}, L, F), \sigma\big)$ of $\mathcal{B}_1$ we always have $F(P(0)) = F(m^{(1)}) = 1$ and $F(P(1)) = F(m^{(2)}) = 0$. $(\mathcal{B}_1, \mathcal{B}_2)$ does not make prohibited oracle queries because $(\mathcal{A}_1, \mathcal{A}_2)$ does not.

Let us denote the following experiments by $\mathsf{Exp}^{(i)}$ for $i = 1, 2$, where i denotes which plaintext is chosen.

$$
\begin{array}{|l|}
\hline
\texttt{Experiment} \quad \mathsf{Exp}^{(i)} \\
\quad (\mathsf{pms}, \mathsf{mk}) \leftarrow S(1^\ell); \qquad ((m^{(1)}, m^{(2)}, \mathsf{id}_{\mathsf{ch}}), \sigma) \leftarrow \mathcal{A}_1^{X_{\mathsf{mk}}, D_{\mathsf{mk}}}(\mathsf{pms}); \\
\quad c \leftarrow E(\mathsf{pms}, \mathsf{id}_{\mathsf{ch}}, m^{(i)}); \qquad v \leftarrow \mathcal{A}_2^{X_{\mathsf{mk}}, D_{\mathsf{mk}}}(\sigma, (\mathsf{id}_{\mathsf{ch}}, c)); \\
\hline
\end{array}
$$

Obviously $p_\ell^{(i)}$ (as in the definition of IND-ID-CCA) is equal to $\mathsf{Pr}[v = 1 \mid \mathsf{Exp}^{(i)}]$. Now the advantage of the above adversary $(\mathcal{B}_1, \mathcal{B}_2)$ for given ℓ is calculated as follows.

$$
\begin{aligned}
&\mathsf{Pr}\left[v = F(m) \;\middle|\; \begin{array}{c} (\mathsf{pms}, \mathsf{mk}) \leftarrow S(1^\ell); \\ ((P, \mathsf{id}_{\mathsf{ch}}, L, F), \sigma) \leftarrow \mathcal{B}_1^{X_{\mathsf{mk}}, D_{\mathsf{mk}}}(\mathsf{pms}); \\ m \leftarrow P(U_{\mathsf{poly}(\ell)}); \\ v \leftarrow \mathcal{B}_2^{X_{\mathsf{mk}}, D_{\mathsf{mk}}}(\sigma, \mathsf{id}_{\mathsf{ch}}, (E(\mathsf{pms}, \mathsf{id}_{\mathsf{ch}}, m), L(m))) \end{array} \right] \\
&\overset{(\dagger)}{=} \frac{1}{2} \mathsf{Pr}\left[v = 1 \;\middle|\; \mathsf{Exp}^{(1)} \right] + \frac{1}{2} \mathsf{Pr}\left[v = 0 \;\middle|\; \mathsf{Exp}^{(2)} \right] \\
&\overset{(\ddagger)}{=} \frac{1}{2} \mathsf{Pr}\left[v = 1 \;\middle|\; \mathsf{Exp}^{(1)} \right] + \frac{1}{2} \left(1 - \mathsf{Pr}\left[v = 1 \;\middle|\; \mathsf{Exp}^{(2)} \right] \right) = \frac{1}{2} + \frac{1}{2} \big(p_\ell^{(1)} - p_\ell^{(2)} \big) \ .
\end{aligned}
$$

Here (†) holds because $m^{(1)} \neq m^{(2)}$, and (‡) holds because $\mathcal{B}_2$ always outputs either 0 or 1. By assumption, the last quantity is distinguishably larger than 1/2, as a function on ℓ.

Let $(\mathcal{B}'_1, \mathcal{B}'_2)$ be an arbitrary benign simulator of $(\mathcal{B}_1, \mathcal{B}_2)$. Its advantage is evaluated as follows, using the fact that the (P, L, F) part in the outputs of $\mathcal{B}_1$ and $\mathcal{B}'_1$ are identically distributed.

$$\begin{aligned}
&\mathsf{Pr}\left[v = F(m) \;\middle|\; \begin{array}{c} ((P,L,F),\sigma') \leftarrow \mathcal{B}'_1(1^\ell); \\ m \leftarrow P(U_{\mathsf{poly}(\ell)}); \\ v \leftarrow \mathcal{B}'_2(\sigma', L(m)) \end{array} \right] \\
&= \frac{1}{2}\,\mathsf{Pr}\left[v = 1 \;\middle|\; \begin{array}{c} \big((P,L,F),\sigma'\big) \leftarrow \mathcal{B}'_1(1^\ell); \\ m \leftarrow P(0); \\ v \leftarrow \mathcal{B}'_2(\sigma', 0) \end{array} \right] + \frac{1}{2}\,\mathsf{Pr}\left[v = 0 \;\middle|\; \begin{array}{c} \big((P,L,F),\sigma'\big) \leftarrow \mathcal{B}'_1(1^\ell); \\ m \leftarrow P(1); \\ v \leftarrow \mathcal{B}'_2(\sigma', 0) \end{array} \right] \\
&\leq \frac{1}{2}\,\mathsf{Pr}\left[v = 1 \;\middle|\; \begin{array}{c} \big((P,L,F),\sigma'\big) \leftarrow \mathcal{B}'_1(1^\ell); \\ v \leftarrow \mathcal{B}'_2(\sigma', 0) \end{array} \right] + \frac{1}{2}\left(1 - \mathsf{Pr}\left[v = 1 \;\middle|\; \begin{array}{c} \big((P,L,F),\sigma'\big) \leftarrow \mathcal{B}'_1(1^\ell); \\ v \leftarrow \mathcal{B}'_2(\sigma', 0) \end{array} \right] \right) \\
&= \frac{1}{2}\ .
\end{aligned}$$

Hence the success rate of the actual adversary is distinguishably larger than that of any benign simulator, which contradicts our assumption of SS-ID-CCA. This concludes the proof. □

Theorem 9 NM-ATK *entails* IND-ATK, *for both* ATK = ID-CPA, ID-CCA.

Proof: The proof is presented for the case ATK = ID-CCA. For ATK = ID-CPA the modification is easy by just dropping oracle accesses.

By contradiction. Assume we have a successful IND-ATK distinguisher $(\mathcal{A}_1, \mathcal{A}_2)$. Then using them we can construct a successful NM-ATK adversary $(\mathcal{C}_1, \mathcal{C}_2)$ as follows.

```
Algorithm  C1^{O1,O2}(pms)
    ((m^(1), m^(2), id_ch), σ) ← A1^{O1,O2}(pms);
    P ← a circuit with one input bit such that P(0) = m^(1) and P(1) = m^(2);
    return  (P, id_ch, (σ, m^(1), m^(2)))
```

```
Algorithm  C2^{O1,O2}((σ, m^(1), m^(2)), id_ch, c)
    v ← A2^{O1,O2}(σ, (id_ch, c));
    if  v = 1  then  d ← E(pms, id_ch, overline{m^(1)})  else  d ← E(pms, id_ch, overline{m^(2)});
    R ← Comp, where Comp(x, y) <=def=> x = overline{y} ;
    return  (R, d)
```

The pair $(\mathcal{C}_1, \mathcal{C}_2)$ thus defined does not make prohibited oracle queries (as an NM-ID-CCA adversary) because $(\mathcal{A}_1, \mathcal{A}_2)$ does not (as an IND-ID-CCA distinguisher).

The trick is that we take as R a relation other than the equality: this ensures that the ciphertext d output by $\mathcal{C}_2$ is distinct from the challenge ciphertext c given to $\mathcal{C}_2$. It is straightforward to show that this pair $(\mathcal{C}_1, \mathcal{C}_2)$ is indeed successful: details follow next.

As in the proof of Theorem 8 we can assume that, the successful IND-ID-CCA distinguisher $(\mathcal{A}_1, \mathcal{A}_2)$ is such that: 1) $\mathcal{A}_2$ always outputs either 0 or 1; and 2) the function $p_\ell^{(1)} - p_\ell^{(2)}$ over ℓ (rather than its absolute value) is not negligible; and 3) $\mathcal{A}_1$ always outputs distinct challenge plaintexts $m^{(1)} \neq m^{(2)}$. Let us denote the following experiment by $\mathsf{Exp}^{(i)}$, for $i = 1, 2$. The parameter i represents which plaintext is chosen by the challenger.

$$
\begin{array}{|l|}
\hline
\texttt{Experiment} \quad \mathsf{Exp}^{(i)} \\
\quad (\mathsf{pms}, \mathsf{mk}) \leftarrow S(1^\ell); \qquad \big((m^{(1)}, m^{(2)}, \mathsf{id_{ch}}), \sigma\big) \leftarrow \mathcal{A}_1^{X_{\mathsf{mk}}, D_{\mathsf{mk}}}(\mathsf{pms}); \\
\quad c \leftarrow E(\mathsf{pms}, \mathsf{id_{ch}}, m^{(i)}); \qquad v' \leftarrow \mathcal{A}_2^{X_{\mathsf{mk}}, D_{\mathsf{mk}}}(\sigma, (\mathsf{id_{ch}}, c)); \\
\quad \texttt{if} \quad v' = 1 \quad \texttt{then} \quad c' \leftarrow E(\mathsf{pms}, \mathsf{id_{ch}}, \overline{m^{(1)}}) \quad \texttt{else} \quad c' \leftarrow E(\mathsf{pms}, \mathsf{id_{ch}}, \overline{m^{(2)}}); \\
\quad m' \leftarrow D(\mathsf{mk}, \mathsf{id_{ch}}, c'); \\
\hline
\end{array}
$$

Now for the NM-ID-CCA adversary $(\mathcal{C}_1, \mathcal{C}_2)$ constructed using $(\mathcal{A}_1, \mathcal{A}_2)$, its advantage is calculated as follows.

$$
\begin{aligned}
& \frac{1}{2} \sum_{i=1,2} \mathsf{Pr}\left[c \neq c' \wedge \perp \neq m' \wedge \mathsf{Comp}(m^{(i)}, m') \,\middle|\, \mathsf{Exp}^{(i)}\right] \\
& - \frac{1}{4} \sum_{i,j=1,2} \mathsf{Pr}\left[c \neq c' \wedge \perp \neq m' \wedge \mathsf{Comp}(m^{(j)}, m') \,\middle|\, \mathsf{Exp}^{(i)}\right] \\
& \overset{(*)}{=} \frac{1}{4} \sum_{i=1,2} \mathsf{Pr}\left[\mathsf{Comp}(m^{(i)}, m') \,\middle|\, \mathsf{Exp}^{(i)}\right] - \frac{1}{4} \sum_{(i,j)=(1,2),(2,1)} \mathsf{Pr}\left[c \neq c' \wedge \mathsf{Comp}(m^{(j)}, m') \,\middle|\, \mathsf{Exp}^{(i)}\right] \\
& \overset{(\dagger)}{\geq} \frac{1}{4} p_\ell^{(1)} + \frac{1}{4}(1 - p_\ell^{(2)}) - \frac{1}{4}(1 - p_\ell^{(1)}) - \frac{1}{4} p_\ell^{(2)} = \frac{1}{2}(p_\ell^{(1)} - p_\ell^{(2)}) \ .
\end{aligned}
$$

In the equality $(*)$ we use the facts that m' in $\mathsf{Exp}^{(i)}$ must be either $m^{(1)}$ or $m^{(2)}$ hence not $\perp$, and that if $\mathsf{Comp}(m^{(i)}, m')$ then $c \leftarrow E(\mathsf{pms}, \mathsf{id_{ch}}, m^{(i)})$ and $c' \leftarrow E(\mathsf{pms}, \mathsf{id_{ch}}, m')$ cannot be identical.[3] For the inequality $(\dagger)$ we first drop the condition $c \neq c'$ in the second probability, and then use the equalities such as $\mathsf{Pr}\left[\mathsf{Comp}(m^{(2)}, m') \mid \mathsf{Exp}^{(2)}\right] = \mathsf{Pr}\left[v' \neq 1 \mid \mathsf{Exp}^{(2)}\right] = 1 - p_\ell^{(2)}$. By the success of $(\mathcal{A}_1, \mathcal{A}_2)$ this function is not negligible, which contradicts that the scheme is NM-ID-CCA. □

Theorem 10 IND-ID-CCA *implies* NM-ID-CCA.

Proof: Assume the existence of a successful NM-ID-CCA adversary $(\mathcal{C}_1, \mathcal{C}_2)$. Using it we construct a successful IND-ID-CCA distinguisher $(\mathcal{A}_1, \mathcal{A}_2)$ as follows. The point is, since $\mathcal{A}_2$ has an access to decryption oracle, it can decrypt the ciphertexts output by $\mathcal{C}_2$ which are related to the challenge ciphertext.

$$
\begin{array}{|l|}
\hline
\texttt{Algorithm} \quad \mathcal{A}_1^{O_1, O_2}(\mathsf{pms}) \\
\quad (P, \mathsf{id_{ch}}, \sigma) \leftarrow \mathcal{C}_1^{O_1, O_2}(\mathsf{pms}); \\
\quad m^{(1)} \leftarrow P(U_{\mathsf{poly}(\ell)}); \qquad m^{(2)} \leftarrow P(U_{\mathsf{poly}(\ell)}); \\
\quad \texttt{return} \quad \big((m^{(1)}, m^{(2)}, \mathsf{id_{ch}}), (m^{(1)}, m^{(2)}, \sigma)\big) \\
\hline
\end{array}
$$

$$
\begin{array}{|l|}
\hline
\texttt{Algorithm} \quad \mathcal{A}_2^{O_1, O_2}\big((m^{(1)}, m^{(2)}, \sigma), (\mathsf{id_{ch}}, c)\big) \\
\quad (R, \overrightarrow{c}) \leftarrow \mathcal{C}_2^{O_1, O_2}\big(\sigma, \mathsf{id_{ch}}, c\big); \qquad \overrightarrow{m} \leftarrow O_2(\mathsf{id_{ch}}, \overrightarrow{c}); \\
\quad \texttt{if} \quad \big(\perp \notin \overrightarrow{m} \wedge R(m^{(1)}, \overrightarrow{m})\big) \quad \texttt{then} \quad v \leftarrow 1 \quad \texttt{else} \quad v \leftarrow 0; \\
\quad \texttt{return} \quad v \\
\hline
\end{array}
$$

[3] Note that $\mathsf{Comp}(m^{(j)}, m')$ does not ensure $c \neq c'$ when $j \neq i$. Here $c \leftarrow E(\mathsf{pms}, \mathsf{id_{ch}}, m^{(i)})$, $c' \leftarrow E(\mathsf{pms}, \mathsf{id_{ch}}, m')$ and it is possible that $m^{(i)} = m' = \overline{m^{(j)}}$.

We can assume that in every execution $(R, \overrightarrow{c}) \leftarrow \mathcal{C}_2^{O_1,O_2}(\sigma, \mathsf{id_{ch}}, c)$ we have $c \notin \overrightarrow{c}$.[4] This yields that $(\mathcal{A}_1, \mathcal{A}_2)$ thus defined does not make prohibited oracle queries. The advantage of $(\mathcal{A}_1, \mathcal{A}_2)$ (before taking the absolute value), $p_\ell^{(1)} - p_\ell^{(2)}$, is identical to that of $(\mathcal{C}_1, \mathcal{C}_2)$, hence must be non-negligible. This is a contradiction. □

Theorem 11 IND-ID-CPA *does not imply* NM-ID-CPA.

Proof: Assume that there exists an IBE scheme (S, X, E, D) which is IND-ID-CPA (otherwise the claim is trivially true). We construct another IBE scheme (S, X, E', D') which is IND-ID-CPA but not NM-ID-CPA, whose existence proves the theorem.

```
Algorithm  E'(pms, id, m)
   c1 ← E(pms, id, m);      c2 ← E(pms, id, m̄);
   return  (c1, c2)
```

```
Algorithm  D'(mk, id, (c1, c2))
   m ← D(mk, id, c1);
   return  m;
```

The algorithms E', D' thus defined obviously satisfy the condition that a ciphertext, when decrypted, yields the original plaintext.

The scheme (S, X, E', D') is not NM-ID-CPA due to the following successful adversary: a simple calculation shows that the advantage of this adversary $(\mathcal{C}_1, \mathcal{C}_2)$ is $1 - 1/2^\ell$, where $1/2^\ell$ is the probability a random (fake) plaintext m_0 is accidentally equal to the real plaintext m.

```
Algorithm  C1^{O1,O2}(pms)
   P ← the identity circuit with ℓ input bits;
   id_ch ← U_ℓ;      σ ← the empty string;
   return  (P, id_ch, σ)
```

```
Algorithm  C2^{O1,O2}(σ, id_ch, (c1, c2))
   return  (Comp, (c2, c1))
```

It remains to show that the new scheme (S, X, E', D') is IND-ID-CPA. We argue by contradiction: if we have a successful IND-ID-CPA distinguisher $(\mathcal{A}_1, \mathcal{A}_2)$ for this new scheme, then from that we can construct a successful distinguisher for the original scheme (S, X, E, D). For the notational convenience, let us denote by $q_\ell^{(i,j)}$ the following probability, for each $i = 1, 2$ and $j = 1, 2$.

$$q_\ell^{(i,j)} = \mathsf{Pr}\left[v = 1 \;\middle|\; \begin{array}{c} (\mathsf{pms}, \mathsf{mk}) \leftarrow S(1^\ell); \qquad \big((m^{(1)}, m^{(2)}, \mathsf{id_{ch}}), \sigma\big) \leftarrow \mathcal{A}_1^{X_{\mathsf{mk}}}(\mathsf{pms}); \\ c_1 \leftarrow E(\mathsf{pms}, \mathsf{id_{ch}}, m^{(i)}); \qquad c_2 \leftarrow E(\mathsf{pms}, \mathsf{id_{ch}}, \overline{m^{(j)}}); \\ v \leftarrow \mathcal{A}_2^{X_{\mathsf{mk}}}\big(\sigma, (\mathsf{id_{ch}}, (c_1, c_2))\big); \end{array} \right] .$$

Then the advantage of the distinguisher $(\mathcal{A}_1, \mathcal{A}_2)$ is now denoted by

$$q_\ell^{(1,1)} - q_\ell^{(2,2)} = \big(q_\ell^{(1,1)} - q_\ell^{(1,2)}\big) + \big(q_\ell^{(1,2)} - q_\ell^{(2,2)}\big) .$$

Since this probability is non-negligible, either $q_\ell^{(1,1)} - q_\ell^{(1,2)}$ or $q_\ell^{(1,2)} - q_\ell^{(2,2)}$ must be non-negligible. We shall show that in either case we can construct a successful IND-ID-CPA distinguisher $(\mathcal{A}_1', \mathcal{A}_2')$ for (S, X, E, D) using $(\mathcal{A}_1, \mathcal{A}_2)$, which contradicts our assumption.

[4] We can modify $\mathcal{C}_2$ such as, if $c \in \overrightarrow{c}$ then the output is $(\mathsf{Empty}, \overline{c})$, where Empty is the empty relation. Obviously this modification does not affect the advantage of the adversary.

If the former probability is non-negligible, define $(\mathcal{A}'_1, \mathcal{A}'_2)$ as follows. It is straight forward to see that its advantage is equal to $q_\ell^{(1,1)} - q_\ell^{(1,2)}$ hence non-negligible.

$$
\begin{array}{|l|}
\hline
\texttt{Algorithm}\quad \mathcal{A}'^{O_1}_1(\mathsf{pms}) \\
\quad \big((m^{(1)}, m^{(2)}, \overline{\mathsf{id}_{\mathsf{ch}}}), \sigma\big) \leftarrow \mathcal{A}_1^{O_1}(\mathsf{pms}); \\
\quad \texttt{return}\quad \big((\overline{m^{(1)}, m^{(2)}}, \mathsf{id}_{\mathsf{ch}}), (m^{(1)}, m^{(2)}, \sigma)\big) \\
\hline
\end{array}
\qquad
\begin{array}{|l|}
\hline
\texttt{Algorithm}\quad \mathcal{A}'^{O_1}_2\big((m^{(1)}, m^{(2)}, \sigma), (\mathsf{id}_{\mathsf{ch}}, c)\big) \\
\quad c_1 \leftarrow E(\mathsf{pms}, \mathsf{id}_{\mathsf{ch}}, m^{(1)}); \qquad c_2 \leftarrow c; \\
\quad v \leftarrow \mathcal{A}_2^{O_1}\big(\sigma, (\mathsf{id}_{\mathsf{ch}}, (c_1, c_2))\big); \\
\quad \texttt{return}\quad v \\
\hline
\end{array}
$$

If the latter probability $q_\ell^{(1,2)} - q_\ell^{(2,2)}$ is non-negligible, $(\mathcal{A}'_1, \mathcal{A}'_2)$ is defined as follows. Its advantage is equal to $q_\ell^{(1,2)} - q_\ell^{(2,2)}$ hence non-negligible.

$$
\begin{array}{|l|}
\hline
\texttt{Algorithm}\quad \mathcal{A}'^{O_1}_1(\mathsf{pms}) \\
\quad \big((m^{(1)}, m^{(2)}, \mathsf{id}_{\mathsf{ch}}), \sigma\big) \leftarrow \mathcal{A}_1^{O_1}(\mathsf{pms}); \\
\quad \texttt{return}\quad \big((m^{(1)}, m^{(2)}, \mathsf{id}_{\mathsf{ch}}), (m^{(1)}, m^{(2)}, \sigma)\big) \\
\hline
\end{array}
\qquad
\begin{array}{|l|}
\hline
\texttt{Algorithm}\quad \mathcal{A}'^{O_1}_2\big((m^{(1)}, m^{(2)}, \sigma), \overline{(\mathsf{id}_{\mathsf{ch}}}, c)\big) \\
\quad c_1 \leftarrow c; \qquad c_2 \leftarrow E(\mathsf{pms}, \mathsf{id}_{\mathsf{ch}}, \overline{m^{(2)}}); \\
\quad v \leftarrow \mathcal{A}_2^{O_1}\big(\sigma, (\mathsf{id}_{\mathsf{ch}}, (c_1, c_2))\big); \\
\quad \texttt{return}\quad v \\
\hline
\end{array}
$$

This concludes the proof. ◻

The proof for the next theorem is substantially more complex than the analogous result for public-key schemes [BDPR98, Theorem 3.6]. This is due to an important difference between IBE and PKE attack models, namely the existence of extraction oracles.

Theorem 12 NM-ID-CPA *does not imply* IND-ID-CCA.

Proof: Assume we have an IBE scheme (S, X, E, D) which is NM-ID-CPA. Using this we construct a new scheme (S', X', E', D') which is NM-ID-CPA but not IND-ID-CCA.

We can assume that, for each $\ell \in \mathbb{Z}^+$, we have a family of pseudo-random functions $F^\ell = \{F_K \mid K \in \{0,1\}^\ell\}$, where $F_K : \{0,1\}^{\mathsf{poly}(\ell)} \to \{0,1\}^{\mathsf{poly}(\ell)}$. Notice this does not imply an extra assumption, since the existence of a NM-ID-CPA secure IBE implies a IND-ID-CPA secure IBE by Theorem 9. This implies secure public key signature schemes [BF03], and this in turn implies one-way functions. Finally the existence of a family of pseudo-random functions is obtained by applying classical results such as [GGM86].

The idea of the construction is as follows. The new decryption algorithm, when queried on the value $F_K(\mathsf{id})$, reveals the decryption key for id. The value $F_K(\mathsf{id})$, which is unique to each id and kept secret, can only be known by making a decryption query on a publicly known value e. We have these two steps (e to $F_K(\mathsf{id})$ to the decryption key), instead of only one step (e to the decryption key), so that the scheme is easily shown to be NM-ID-CPA. This procedure of obtaining the decryption key can be done only by an ID-CCA adversary: an ID-CPA adversary can try extraction queries, but it is prohibited to make it on the challenge identity hence the values $F_K(\mathsf{id})$ it obtains are irrelevent to the desired $F_K(\mathsf{id}_{\mathsf{ch}})$.

$$
\begin{array}{|l|}
\hline
\texttt{Algorithm}\quad S'(1^\ell) \\
\quad (\mathsf{pms}, \mathsf{mk}) \leftarrow S(1^\ell); \qquad e \leftarrow U_\ell; \qquad K \leftarrow U_\ell; \\
\quad \texttt{return}\quad \big((\mathsf{pms}, e), (\mathsf{mk}, e, K)\big) \\
\hline
\end{array}
\qquad
\begin{array}{|l|}
\hline
\texttt{Algorithm}\quad X'\big(\mathsf{pms}, (\mathsf{mk}, e, K), \mathsf{id}\big) \\
\quad d \leftarrow X(\mathsf{mk}, \mathsf{id}); \\
\quad \texttt{return}\quad \big(d, e, F_K(\mathsf{id})\big) \\
\hline
\end{array}
$$

$$
\begin{array}{|l|}
\hline
\texttt{Algorithm}\quad E'\big((\mathsf{pms}, e), \mathsf{id}, m\big) \\
\quad c \leftarrow E(\mathsf{pms}, \mathsf{id}, m); \\
\quad \texttt{return}\quad (0, c) \\
\hline
\end{array}
\qquad
\begin{array}{|l|}
\hline
\texttt{Algorithm}\quad D'\big(\mathsf{pms}, (d, e, g), (i, c)\big) \\
\quad \texttt{if}\quad i = 0 \quad \texttt{then}\quad m \leftarrow D(d, c) \\
\quad \texttt{else}\quad \texttt{if}\quad c = e \quad \texttt{then}\quad m \leftarrow g \\
\qquad\qquad \texttt{else}\quad \texttt{if}\quad c = g \quad \texttt{then}\quad m \leftarrow d \\
\qquad\qquad\qquad \texttt{else}\quad m \leftarrow \perp; \\
\quad \texttt{return}\quad m \\
\hline
\end{array}
$$

Note that in the actual use the value g in D' above will be $F_K(\mathsf{id})$. The scheme (S', X', E', D') is not IND-ID-CCA due to the existence of the following successful distinguisher $(\mathcal{A}_1, \mathcal{A}_2)$. The oracles O'_1, O'_2 will be $X'_{(\mathsf{mk},e,K)}, D'_{(\mathsf{mk},e,K)}$, respectively, where the decryption oracle $D'_{(\mathsf{mk},e,K)}$ is defined by $D'_{(\mathsf{mk},e,K)}(\mathsf{id}, c) = D'\big(\mathsf{pms}, X'((\mathsf{mk}, e, K), \mathsf{id}), c\big)$.

```
Algorithm  A_1^{O'_1,O'_2}((pms, e))
  m^(1) ← (the uniform distribution over the message set specified in pms);
  id_ch ← U_ℓ;
  return  ((m^(1), overline{m^(1)}, id_ch), (m^(1), e))
```

```
Algorithm  A_2^{O'_1,O'_2}((m^(1), e), (id_ch, (0, c)))
  a ← O'_2(id_ch, (1, e));      b ← O'_2(id_ch, (1, a));      m ← D(b, c);
  if  m = m^(1)  then  v ← 1  else  v ← 0;
  return  v
```

In the description of $\mathcal{A}_2$, the value a will be $F_K(\mathsf{id_{ch}})$ and b will be $X(\mathsf{mk}, \mathsf{id})$ in an actual attack. D is the decryption algorithm of the original IBE scheme. Obviously the advantage of this distinguisher $(\mathcal{A}_1, \mathcal{A}_2)$ is 1, which is non-negligible.

It remains to show that the new scheme (S', X', E', D') is NM-ID-CPA. We argue by contradiction. Let $(\mathcal{C}_1, \mathcal{C}_2)$ be a successful NM-ID-CPA adversary for (S', X', E', D'). We construct an NM-ID-CPA adversary $(\mathcal{C}'_1, \mathcal{C}'_2)$ for (S, X, E, D) as follows, and the adversary $(\mathcal{C}'_1, \mathcal{C}'_2)$ will be shown to be successful.

```
Algorithm  C'_1^{O_1}(pms)
  e ← U_ℓ;      K ← U_ℓ;
  (P, id_ch, σ) ← C_1^{O'_1^{e,K}}((pms, e));
  return  (P, id_ch, (σ, pms, e, K))
```

```
Algorithm  C'_2^{O_1}((σ, pms, e, K), id_ch, c)
  (R, →c) ← C_2^{O'_1^{e,K}}(σ, id_ch, (0, c));
  for  1 ≤ i ≤ |→c|  do
    if  →c[i] = (0, c')  then  →d[i] ← c'
    else  if  →c[i] = (1, e)  then  →d[i] ← O_2(id_ch, F_K(id_ch))
            else  →d[i] ← c;
  return  (R, →d)
```

Here, ${O'}_1^{e,K}$ (meant to be $X'_{(\mathsf{mk},e,K)}$) is emulated using O_1 (meant to be X_{mk}) as ${O'}_1^{e,K}(\mathsf{id}) = \big(O_1(\mathsf{id}), e, F_K(\mathsf{id})\big)$.

We shall show that the NM-ID-CPA adversary $(\mathcal{C}'_1, \mathcal{C}'_2)$ for (S, X, E, D) defined above is successful. We denote by Exp_1 the experiment under which the advantage of $(\mathcal{C}'_1, \mathcal{C}'_2)$ (attacking (S, X, E, D)) is evaluated, and by Exp_2 the one for $(\mathcal{C}_1, \mathcal{C}_2)$ (attacking (S', X', E', D')). By expanding the definition we obtain the following description.

$$\begin{array}{|l|}\hline \texttt{Experiment}\quad \mathsf{Exp}_1 \\ \quad (\mathsf{pms},\mathsf{mk}) \leftarrow S(1^\ell); \qquad e \leftarrow U_\ell; \qquad K \leftarrow U_\ell; \\ \quad (P,\mathsf{id}_{\mathsf{ch}},\sigma) \leftarrow \mathcal{C}_1^{X'_{(\mathsf{mk},e,K)}}\big((\mathsf{pms},e)\big); \\ \quad m, m_0 \leftarrow P(U_{\mathsf{poly}(\ell)}); \qquad c \leftarrow E(\mathsf{pms},\mathsf{id}_{\mathsf{ch}},m); \\ \quad (R,\overrightarrow{c}) \leftarrow \mathcal{C}_2^{X'_{(\mathsf{mk},e,K)}}\big(\sigma,\mathsf{id}_{\mathsf{ch}},(0,c)\big); \\ \quad \overrightarrow{d} \leftarrow (\text{constructed from } \overrightarrow{c} \text{ as in } \mathcal{C}'_2); \\ \quad \overrightarrow{m} \leftarrow D(\mathsf{mk},\mathsf{id}_{\mathsf{ch}},\overrightarrow{d}); \\ \hline \end{array}$$

$$\begin{array}{|l|}\hline \texttt{Experiment}\quad \mathsf{Exp}_2 \\ \quad (\mathsf{pms},\mathsf{mk}) \leftarrow S(1^\ell); \qquad e \leftarrow U_\ell; \qquad K \leftarrow U_\ell; \\ \quad (P,\mathsf{id}_{\mathsf{ch}},\sigma) \leftarrow \mathcal{C}_1^{X'_{(\mathsf{mk},e,K)}}\big((\mathsf{pms},e)\big); \\ \quad m, m_0 \leftarrow P(U_{\mathsf{poly}(\ell)}); \qquad c \leftarrow E(\mathsf{pms},\mathsf{id}_{\mathsf{ch}},m); \\ \quad (R,\overrightarrow{c}) \leftarrow \mathcal{C}_2^{X'_{(\mathsf{mk},e,K)}}\big(\sigma,\mathsf{id}_{\mathsf{ch}},(0,c)\big); \\ \quad \overrightarrow{m} \leftarrow D'\big((\mathsf{mk},e,K),\mathsf{id}_{\mathsf{ch}},\overrightarrow{c}\big); \\ \hline \end{array}$$

In the sequel we denote a probability function evaluated under Exp_1 by $\mathsf{Pr}_1[\cdot]$, and one evaluated under Exp_2 by $\mathsf{Pr}_2[\cdot]$.

We make case-distinction: under Exp_1 or Exp_2, exactly one of the following events E_1, E_2, E_3 happens.

$$\begin{aligned} E_1 &\overset{\text{def.}}{=} (\overrightarrow{c} \text{ contains only } (0,*) \text{ or } (1,e)) \ , \\ E_2 &\overset{\text{def.}}{=} (\overrightarrow{c} \text{ contains only } (0,*) \text{ or } (1,e) \text{ or } (1,F_K(\mathsf{id}_{\mathsf{ch}})), \text{ but not } E_1) \ , \\ E_3 &\overset{\text{def.}}{=} (\text{neither } E_1 \text{ nor } E_2) \ . \end{aligned}$$

It is easy to see $\mathsf{Pr}_1[E_j] = \mathsf{Pr}_2[E_j]$ for each $j = 1,2,3$.

For notational convenience we define the following probabilities, for $j = 1,2,3$.

$$\begin{aligned} p(1,j) &= \mathsf{Pr}_1\left[c \notin \overrightarrow{d} \wedge \bot \notin \overrightarrow{m} \wedge R(m,\overrightarrow{m}) \mid E_j\right] - \mathsf{Pr}_1\left[c \notin \overrightarrow{d} \wedge \bot \notin \overrightarrow{m} \wedge R(m_0,\overrightarrow{m}) \mid E_j\right] \ , \\ p(2,j) &= \mathsf{Pr}_2\left[(0,c) \notin \overrightarrow{c} \wedge \bot \notin \overrightarrow{m} \wedge R(m,\overrightarrow{m}) \mid E_j\right] - \mathsf{Pr}_2\left[(0,c) \notin \overrightarrow{c} \wedge \bot \notin \overrightarrow{m} \wedge R(m_0,\overrightarrow{m}) \mid E_j\right] \ . \end{aligned}$$

The probability $p(1,j)$ is the (conditional) advantage of $(\mathcal{C}'_1, \mathcal{C}'_2)$ under the event E_j, and $p(2,j)$ is that of $(\mathcal{C}_1, \mathcal{C}_2)$. Hence, by the case-distinction, the advantage of $(\mathcal{C}'_1, \mathcal{C}'_2)$ is equal to $\sum_{j=1}^3 p(1,j) \cdot \mathsf{Pr}_1[E_j]$, and that of $(\mathcal{C}_1, \mathcal{C}_2)$ is $\sum_{j=1}^3 p(2,j) \cdot \mathsf{Pr}_2[E_j]$.

First we consider the case E_1 happens. In that case, the sequence of plaintexts $\overrightarrow{m}$ obtained from $\overrightarrow{c}$ via $\overrightarrow{d}$ in Exp_1 is directly obtained by $\overrightarrow{m} \leftarrow D'\big((\mathsf{mk},e,K),\mathsf{id}_{\mathsf{ch}},\overrightarrow{c}\big)$. Hence the gap between $p(1,1)$ and $p(2,1)$ comes only from the conditions $c \notin \overrightarrow{d}$ in $p(1,1)$ and $(0,c) \notin \overrightarrow{c}$ in $p(2,1)$. More precisely, only the case that, $\overrightarrow{c}$ contains $(1,e)$ and moreover $c = E\big(\mathsf{pms},\mathsf{id}_{\mathsf{ch}},F_K(\mathsf{id}_{\mathsf{ch}})\big)$, contributes to the gap. In this case the plaintext m must be identical to $F_K(\mathsf{id}_{\mathsf{ch}})$: this happens with only a negligible probability since the value $F_K(\mathsf{id}_{\mathsf{ch}})$ remains secret to the adversary. Therefore we conclude that $|p(1,1) - p(2,1)|$ is negligible.

Next we consider the case E_2 happens. In this case $\overrightarrow{c}$ must contain $\big(1,F_K(\mathsf{id}_{\mathsf{ch}})\big)$. However, just like in the previous paragraph, the algorithms $\mathcal{C}_1$ and $\mathcal{C}_2$ have no information about

$F_K(\mathsf{id_{ch}})$ hence this happens with only a negligible probability. To summarize, the probability $\mathsf{Pr}_1[E_2] = \mathsf{Pr}_2[E_2]$ are both negligible.

Finally, we show that $p(1,3) = p(2,3) = 0$. The probability $p(1,3)$ is 0 because, by the definition of $\mathcal{C}_2'$ (especially the construction of $\overrightarrow{d}$ from $\overrightarrow{c}$), $\overrightarrow{d}$ contains c when E_3 happens. The probability $p(2,3)$ is 0 because, when E_3 happens, $\overrightarrow{c}$ contains illegitimate ciphertext so $\perp \in \overrightarrow{m}$.

Combining the previous three paragraphs and that $\mathsf{Pr}_1[E_j] = \mathsf{Pr}_2[E_j]$, we have shown that the advantage $\sum_{j=1}^{3} p(1,j) \cdot \mathsf{Pr}_1[E_j]$ of $(\mathcal{C}_1', \mathcal{C}_2')$ and $\sum_{j=1}^{3} p(2,j) \cdot \mathsf{Pr}_2[E_j]$ of $(\mathcal{C}_1, \mathcal{C}_2)$ have only a negligible gap. By assumption the latter is non-negligible the former must also be non-negligible. This concludes the proof. $\square$

6 Semantical security of IBE schemes under multiple-challenge CCA

In this section we present three notions of semantic security under multiple-challenge CCA, following the public-key version [GLN02]. Here an adversary is allowed to make polynomially many challenge templates. Moreover each template is answered with a challenge ciphertext *immediately* (not after making all the templates), and the next challenge template can be generated according to the preceding templates and their answers. After this stage of asking many challenge templates *adaptively* and *in a related manner*, the adversary tries to guess information about the unrevealed plaintexts which have been used in answering challenge templates.

We shall introduce three different types of multiple-challenge chosen ciphertext attacks:

- In the *multiple-identity* version (mID-CCA), the challenger chooses one fixed plaintext, and the adversary can adaptively query its encryption under different identities polynomially many times.
- In the *multiple-plaintext* version (ID-mCCA), the adversary chooses one fixed identity, and can adaptively query encryption of different plaintexts under that identity polynomially many times.
- In the *multiple-identity-plaintext* version (mID-mCCA), the adversary can adaptively query encryption of different plaintexts under different identities polynomially many times.

Obviously the first two are special cases of the last one. For each attack model we introduce the notion of semantic security, namely SS-mID-CCA, SS-ID-mCCA, and SS-mID-mCCA. We show that these three notions are all equivalent to semantic security under single-challenge CCA, and hence also to the technical notion of indistinguishability.

In the definition of SS-ID-CCA an adversary consists of two oracle PPT's $\mathcal{B}_1$ and $\mathcal{B}_2$, in such a way that $\mathcal{B}_1$ outputs a challenge template, the challenger chooses a plaintext and presents its encryption, and then $\mathcal{B}_2$ tries to guess information about the plaintext. In the multiple-challenge case this interaction is modelled by providing the adversary with a "*tester*" algorithm $T_{r,\mathsf{pms}}$ or T_r as its oracle. $T_{r,\mathsf{pms}}$ is given to an actual adversary (which obtains a ciphertext in addition to information leak), while T_r is given to its benign simulator (which only sees information leak). A challenge template is then sent to one of these oracles as a query (called "*challenge query*").

```
Algorithm   T_{r,pms}(P, id_ch, L)
  return    (E(pms, id_ch, P(r)), L(r))
```

```
Algorithm   T_r(P, L)
  return    L(r)
```

Intuitively the parameter r of a tester is understood as the multiple-challenge version of the coin tosses that the challenger uses to select plaintexts. It is a sufficiently long sequence of coin tosses $(r^1, r^2, \ldots, r^t)$ which is unrevealed to the adversary. Given the i-th challenge template $(P^i, \mathsf{id}^i_{\mathsf{ch}}, L^i)$ (or (P^i, L^i) from a simulator), the challenger chooses a plaintext by $P^i(r^1, r^2, \ldots, r^i)$ using the first i coin tosses in r. Note that now L leaks information on coin tosses r rather than plaintexts $P^i(r^1, r^2, \ldots, r^i)$.[5]

For multiple-challenge CCA adversaries, it is quite natural to put the same restrictions on the adversary's oracle invocations as in the single-challenge version, namely:

– Extraction queries on challenge identities cannot be made;
– Decryption queries on challenge ciphertexts (obtained as answers to challenge queries) cannot be made.

However, as is shown in [GLN02], the second restriction is not necessary under multiple-plaintext CCA: such a decryption query by an actual adversary can be simulated by a benign simulator (which has only access to a tester oracle T_r) by making a challenge query on a extremely informative information leak L, namely $L = P$. For the sake of simplicity, we ignore this point of lifting restriction on decryption queries for the time being: this point is explained in detail in Appendix A.1. It seems that the first restriction on extraction queries is still necessary.

Definition 13 (Semantic security under multiple-challenge CCA) *An IBE scheme* (S, X, E, D) *is said to be* semantically secure under multiple-identity multiple-plaintext chosen ciphertext attacks *(*SS-mID-mCCA*) if the following holds. For every oracle PPT algorithm* $\mathcal{D}$ *("*SS-mID-mCCA *adversary") with the following restriction on oracle queries: in any execution of* $\mathcal{D}^{O_1,O_2,O_3}(\mathsf{pms})$*, for each challenge query* $(c, b) \leftarrow O_3(P, \mathsf{id}_{\mathsf{ch}}, L)$ *by* $\mathcal{D}$*,* $\mathcal{D}$ *is prohibited to make*

– *the extraction query* $O_1(\mathsf{id}_{\mathsf{ch}})$ *regardless of before or after the challenge query, or*
– *the decryption query* $O_2(\mathsf{id}_{\mathsf{ch}}, c)$ *after the challenge query*

there exists a PPT algorithm $\mathcal{D}'$ *("benign simulator of* $\mathcal{D}$*") which is equally successful as* $\mathcal{D}$*, in the following sense.*

1. *The difference between the advantage of the actual adversary* $\mathcal{D}$ *and that of the benign simulator* $\mathcal{D}'$*, namely*

$$\mathsf{Pr}\left[v = F(r) \,\middle|\, \begin{array}{c}(\mathsf{pms}, \mathsf{mk}) \leftarrow S(1^\ell); \qquad r \leftarrow U_{\mathsf{poly}(\ell)};\\ (F, v) \leftarrow \mathcal{D}^{X_{\mathsf{mk}}, D_{\mathsf{mk}}, T_{r,\mathsf{pms}}}(\mathsf{pms})\end{array}\right] - \mathsf{Pr}\left[v = F(r) \,\middle|\, \begin{array}{c}r \leftarrow U_{\mathsf{poly}(\ell)};\\ (F, v) \leftarrow \mathcal{D}'^{T_r}(1^\ell)\end{array}\right]$$

is negligible as a function over ℓ*.*

2. *The two ensembles over* $\ell \in \mathbb{Z}^+$*:*

$$\left[(t, F) \,\middle|\, \begin{array}{c}(\mathsf{pms}, \mathsf{mk}) \leftarrow S(1^\ell); \qquad r \leftarrow U_{\mathsf{poly}(\ell)};\\ (F, v) \leftarrow \mathcal{D}^{X_{\mathsf{mk}}, D_{\mathsf{mk}}, T_{r,\mathsf{pms}}}(\mathsf{pms}) \textit{ with trace } t\end{array}\right] \quad \textit{and}$$

$$\left[(t, F) \,\middle|\, \begin{array}{c}r \leftarrow U_{\mathsf{poly}(\ell)};\\ (F, v) \leftarrow \mathcal{D}'^{T_r}(1^\ell) \textit{ with trace } t\end{array}\right]$$

[5] As is shown in the later definition, the same goes to the information to guess: it is about the coin tosses r (i.e. $F(r)$ to guess) rather than plaintexts (i.e. $F(P(r))$ to guess).

are computationally indistinguishable (i.e. indistinguishable by any PPT algorithm). Here the trace *of an execution of the actual adversary $\mathcal{D}$ is the sequence of (P, L)-part of the challenge queries $(P, \mathsf{id_{ch}}, L)$ made by $\mathcal{D}$. The* trace *of an execution of the simulator $\mathcal{D}'$ is simply the sequence of challenge queries $\mathcal{D}'$ makes.*

Semantic security under multiple-identity chosen ciphertext attacks (SS-mID-CCA) *is defined analogously except that an adversary $\mathcal{D}$ is restricted to have the same plaintext circuit P and the same information leakage circuit L in all the challenge queries in the trace of an execution of $\mathcal{D}$ (the challenge identity $\mathsf{id_{ch}}$ can vary).*[6]

Semantic security under multiple-identity chosen ciphertext attacks (SS-ID-mCCA) *is analogous to SS-mID-mCCA except that an adversary $\mathcal{D}$ must have the same challenge identity $\mathsf{id_{ch}}$ in all the challenge queries in the trace of an execution of $\mathcal{D}$ (P and L can vary).*

Remark 2. Note that our mID-CCA attack is stronger than the attack consider in [BSS05], since in the latter case the adversary has to commit at once to the identities on which it wants to be challenged, while in the present case the i-th identity can be chosen depending on the challenges received so far.

Fig. 1. Implications between security notions (hence all equivalent)

$$\begin{tikzcd}
& \text{SS-mID-CCA} \arrow[d] & \text{SS-mID-mCCA} \arrow[l] \arrow[d] \\
\text{IND-ID-CCA} \arrow[r, shift left] & \text{SS-ID-CCA} \arrow[l, shift left] \arrow[ur] & \text{SS-ID-mCCA} \arrow[l]
\end{tikzcd}$$

Theorem 14 *The five security notions in Figure 1 for identity-based encryption schemes under a-posteriori chosen ciphertext attacks are all equivalent.*

It is obvious that SS-mID-mCCA entails SS-mID-CCA and SS-ID-mCCA. We shall show the remaining implications in the following lemmas.

Lemma 15 *SS-mID-CCA entails SS-ID-CCA. Also, SS-ID-mCCA entails SS-ID-CCA.*

Proof: The proof (by contradiction) is much like that for Theorem 8. Assume that the IBE scheme is not SS-ID-CCA. By Theorem 7 we have a successful IND-ID-CCA distinguisher $(\mathcal{A}_1, \mathcal{A}_2)$, with which we construct a multiple-challenge adversary $\mathcal{D}$. It is shown that $\mathcal{D}$ always has a trace of length 1 (hence qualifies as both SS-mID-CCA and SS-ID-mCCA adversaries), and that $\mathcal{D}$ is distinguishably more successful than any benign simulator.

[6] Note that, since P is deterministic, the plaintext $P(r)$ which is encrypted by the tester oracle $T_{r,\mathsf{pms}}$ stays the same throughout the execution.

Assume that the $\mathsf{IND\text{-}ID\text{-}CCA}$ distinguisher $(\mathcal{A}_1, \mathcal{A}_2)$ is successful. Using this we construct the following multiple-challenge CCA adversary.

```
Algorithm  D^{O1,O2,O3}(pms)
   ((m^(1), m^(2), id_ch), σ) ← A_1^{O1,O2}(pms);
   P, L, F ← (the same as in the proof of Theorem 8);
   (c, b) ← O3(P, id_ch, L ∘ P);      v ← A_2^{O1,O2}(σ, (id_ch, c));
   return  (F ∘ P, v)
```

Here $L \circ P$ denotes the sequential composition of circuits in the order of P after L. This $\mathcal{D}$ indeed qualifies both as an $\mathsf{SS\text{-}mID\text{-}CCA}$ adversary and as an $\mathsf{SS\text{-}ID\text{-}mCCA}$ adversary, due to the following reason. If it makes both an extraction query $O_1(\mathsf{id_{ch}})$ and challenge query $O_3(*, \mathsf{id_{ch}}, *)$ for some $\mathsf{id_{ch}}$ (which is prohibited), then the former is invoked by either $\mathcal{A}_1$ or $\mathcal{A}_2$, while the latter is invoked on the fourth line of the description hence $\mathsf{id_{ch}}$ is in the output of $\mathcal{A}_1$. This violates the condition of an $\mathsf{IND\text{-}ID\text{-}CCA}$ distinguisher. Similarly we can show that $\mathcal{D}$ does not make a decryption query $O_2(\mathsf{id_{ch}}, c)$ on a challenge ciphertext after a challenge query $(c, b) \leftarrow O_3(*, \mathsf{id_{ch}}, *)$. Moreover, $\mathcal{D}$ makes a challenge query only once, which ensures the condition that P and L (or $\mathsf{id_{ch}}$) must stay the same in all the challenge queries.

We can assume the following additional properties of $(\mathcal{A}_1, \mathcal{A}_2)$ as in Theorem 8:

- $\mathcal{A}_2$ always outputs either 0 or 1;
- $p_\ell^{(1)} - p_\ell^{(2)} \geq 1/q(\ell)$ (instead of its absolute value) holds for some polynomial q and infinitely many ℓ's;
- $\mathcal{A}_1$ always outputs distinct challenge plaintexts (i.e. $m^{(1)} \neq m^{(2)}$).

For the multiple-challenge adversary $\mathcal{D}$ defined above and given ℓ, its advantage is calculated as follows.

$$\mathsf{Pr}\left[v = F(r) \,\middle|\, \begin{array}{c} (\mathsf{pms}, \mathsf{mk}) \leftarrow S(1^\ell); \\ r \leftarrow U_{\mathsf{poly}(\ell)}; \\ (F, v) \leftarrow \mathcal{D}^{X_{\mathsf{mk}}, D_{mk}, T_{r,\mathsf{pms}}}(\mathsf{pms}) \end{array} \right] = \frac{1}{2} + \frac{1}{2}(p_\ell^{(1)} - p_\ell^{(2)}) \ ,$$

where the probabilities $p_\ell^{(i)}$ are as in the definition of $\mathsf{IND\text{-}ID\text{-}CCA}$. By the assumption that $(\mathcal{A}_1, \mathcal{A}_2)$ is successful, the advantage of $\mathcal{D}$ is distinguishably larger than $1/2$.

Let $\mathcal{D}'$ be an arbitrary benign simulator of $\mathcal{D}$. Its advantage is evaluated as follows.

$$\begin{aligned} &\mathsf{Pr}\left[v = F'(r) \,\middle|\, \begin{array}{c} r \leftarrow U_1; \\ (F', v) \leftarrow \mathcal{D}'^{T_r}(1^\ell) \end{array} \right] \\ &\overset{(\dagger)}{=} \frac{1}{2}\mathsf{Pr}\left[v = 1 \,\middle|\, (F', v) \leftarrow \mathcal{D}'^{T_0}(1^\ell) \right] + \frac{1}{2}\mathsf{Pr}\left[v = 0 \,\middle|\, (F', v) \leftarrow \mathcal{D}'^{T_1}(1^\ell) \right] \\ &\overset{(\ddagger)}{=} \frac{1}{2}\mathsf{Pr}\left[v = 1 \,\middle|\, (F', v) \leftarrow \mathcal{D}'^{T_0}(1^\ell) \right] + \frac{1}{2}\mathsf{Pr}\left[v = 0 \,\middle|\, (F', v) \leftarrow \mathcal{D}'^{T_0}(1^\ell) \right] \leq \frac{1}{2} \ . \end{aligned}$$

Here (†) holds because the F'-part of the output of $\mathcal{D}'$ is by definition identically distributed as that of $\mathcal{D}$, hence $F'(0) = F(P(0)) = 1$ and $F'(1) = F(P(1)) = 0$. The equation (‡) holds because $T_r(P, L) = L(r)$ and the traces of $\mathcal{D}$ and $\mathcal{D}'$ are identically distributed: the only challenge query $\mathcal{D}'$ makes is on the leakage circuit L in the definition of $\mathcal{D}$, which outputs the constant value, hence the tester oracles T_0 and T_1 gives the same answer. This inequality yields that any simulator $\mathcal{D}'$ can never be as successful as the actual adversary $\mathcal{D}$, which is a contradiction. □

Lemma 16 *SS-ID-CCA entails SS-mID-mCCA.*

Proof: The proof follows the idea of [GLN02]. Assume that an IBE scheme (S, X, E, D) is $\mathsf{SS\text{-}ID\text{-}CCA}$, hence has $\mathsf{IND\text{-}ID\text{-}CCA}$. For a given $\mathsf{SS\text{-}mID\text{-}mCCA}$ adversary $\mathcal{D}$, we construct its benign simulator $\mathcal{D}'$ which invokes $\mathcal{D}$ emulating the tester oracle by giving encryption of the fake plaintext 1^n. Just as in the proof of Theorem 7 we use the indistinguishability between the encryption of this fake plaintext and the actual challenge ciphertext.

The main technical challenge is that here the tester oracle is invoked polynomially many times. To overcome we use a hybrid argument. Let $\Pi^i_{r,\mathsf{pms}}$ be the history-dependent tester oracle which answers using the actual plaintext for the first i challenge queries but answers using the fake plaintext for the rest. If the difference between using $\Pi^i_{r,\mathsf{pms}}$ as a tester oracle and using $\Pi^{i+1}_{r,\mathsf{pms}}$ instead is negligible (which is shown using indistinguishability), then so is the difference between using $\Pi^0_{r,\mathsf{pms}}$ (i.e. the emulated oracle given to the simulator $\mathcal{D}'$) and using $\Pi^{\mathsf{poly}(\ell)}_{r,\mathsf{pms}}$ (i.e. the oracle given to the actual adversary $\mathcal{D}$). That is the main idea of the proof, and the details are found next.

Given an $\mathsf{SS\text{-}mID\text{-}mCCA}$ adversary $\mathcal{D}$, we shall construct its benign simulator $\mathcal{D}'$. First we define a "fake tester" $T'_{r,\mathsf{pms}}$ by

$$T'_{r,\mathsf{pms}}(P, \mathsf{id_{ch}}, L) = \big(E(\mathsf{pms}, \mathsf{id_{ch}}, 1^{|P(r)|}), T_r(P, L)\big) = \big(E(\mathsf{pms}, \mathsf{id_{ch}}, 1^{|P(r)|}), L(r)\big) \ .$$

It is emulated (using the oracle T_r) and plugged in $\mathcal{D}$ in the following definition of the simulator $\mathcal{D}'$.

`Algorithm` $\mathcal{D}'^{T_r}(1^\ell)$
$(\mathsf{pms}, \mathsf{mk}) \leftarrow S(1^\ell)$; $\qquad (F, v) \leftarrow \mathcal{D}^{X_{\mathsf{mk}}, D_{\mathsf{mk}}, T'_{r,\mathsf{pms}}}(\mathsf{pms})$;
`return` (F, v)

We shall show that this simulator $\mathcal{D}'$ has the same trace and is as successful as $\mathcal{D}$. To that end, let two experiments Exp, Exp' be as follows:

`Experiment` Exp
$(\mathsf{pms}, \mathsf{mk}) \leftarrow S(1^\ell)$; $\qquad r \leftarrow U_{\mathsf{poly}(\ell)}$;
$(F, v) \leftarrow \mathcal{D}^{X_{\mathsf{mk}}, D_{\mathsf{mk}}, T_{r,\mathsf{pms}}}(\mathsf{pms})$ with trace t;

`Experiment` Exp'
Same as Exp, except that we have $T'_{r,\mathsf{pms}}$ in place of $T_{r,\mathsf{pms}}$

And then consider the following random variables.

$$V_\ell = \big[(t, F, v, r) \,\big|\, \mathsf{Exp}\big] \qquad\qquad W_\ell = \big[(t, F, v, r) \,\big|\, \mathsf{Exp}'\big]$$

The following equality follows from the definition of $\mathcal{D}'$:

$$W_\ell = \left[(t, F, v, r) \,\middle|\, \begin{array}{c} r \leftarrow U_{\mathsf{poly}(\ell)}; \\ (F, v) \leftarrow \mathcal{D}'^{T_r}(1^\ell) \text{ with trace } t \end{array}\right] \ .$$

Hence to prove the lemma it suffices to show that the ensembles $\{V_\ell\}_{\ell\in\mathbb{Z}^+}$ and $\{W_\ell\}_{\ell\in\mathbb{Z}^+}$ are computationally indistinguishable.[7]

[7] For example, to show that the simulator $\mathcal{D}'$ is as successful as $\mathcal{D}$, consider the algorithm $\mathcal{T}(t, F, v, r)$ which outputs 1 only when $v = F(r)$. If $\mathcal{D}$ is distinguishably more successful than $\mathcal{D}'$ then this algorithm $\mathcal{T}$ distinguishes V_ℓ from W_ℓ, which is a contradiction. The fact that (t, F) is indistinguishably distributed for $\mathcal{D}$ and $\mathcal{D}'$ is shown similarly.

To prove the indistinguishablity of $\{V_\ell\}_{\ell\in\mathbb{Z}^+}$ and $\{W_\ell\}_{\ell\in\mathbb{Z}^+}$, we use an hybrid argument as is already outlined. Let $\Pi^i_{r,\mathsf{pms}}$ be a history-dependent *i-th hybrid oracle* ($i \in \mathbb{N}$) which behaves as $T_{r,\mathsf{pms}}$ for the first i queries, i.e. $\Pi^i_{r,\mathsf{pms}}(P, \mathsf{id_{ch}}, L) = \big(E(\mathsf{pms}, \mathsf{id_{ch}}, P(r)), L(r)\big)$, and then behaves as $T'_{r,\mathsf{pms}}$ for the rest, i.e. $\Pi^i_{r,\mathsf{pms}}(P, \mathsf{id_{ch}}, L) = \big(E(\mathsf{pms}, \mathsf{id_{ch}}, 1^{|P(r)|}), L(r)\big)$. Let us denote the following *i-th hybrid experiment* (of Exp and Exp') by $\mathsf{Exp}^{(i)}$, for $i \in \mathbb{N}$.

```
Experiment  Exp^(i)
   Same as Exp, except that we have Π^i_{r,pms} in place of T_{r,pms}
```

Define $Y^{(i)}_\ell \overset{\text{def.}}{=} [(t, F, v, r) \mid \mathsf{Exp}^{(i)}]$. This is the i-th hybrid of V_ℓ and W_ℓ. In particular $Y^{(0)}_\ell = W_\ell$ and $V_\ell = Y^{(\mathsf{poly}(\ell))}_l$ (since $\mathcal{D}$ queries its tester oracle only polynomially many times). Hence it suffices to show: for $i \in \mathbb{N}$, two ensembles $\{Y^{(i)}_\ell\}_{\ell\in\mathbb{Z}^+}$ and $\{Y^{(i+1)}_\ell\}_{\ell\in\mathbb{Z}^+}$ are computationally indistinguishable.

We argue by contradiction. Assume that a PPT algorithm $\mathcal{T}$ distinguishes the two ensembles. We construct a successful $\mathsf{IND\text{-}ID\text{-}CCA}$ distinguisher $(\mathcal{A}_1, \mathcal{A}_2)$ using $\mathcal{T}$, whose existence is a contradiction. In the following the polynomial $t_\mathcal{D}$ maps a natural number n to the maximal number of steps in the execution of $\mathcal{D}$ with input of length n. Internal coin tosses that $\mathcal{D}$ makes with input of length n are hence described as an element of $\{0,1\}^{t_\mathcal{D}(n)}$. We denote by $\mathcal{D}_s$ the algorithm $\mathcal{D}$ which executes according to specific coin tosses s (hence $\mathcal{D}_s$ is deterministic).

```
Algorithm  A_1^{O_1,O_2}(pms)
   r ← U_{poly(ℓ)};      s ← U_{t_D(|pms|)};
   Execute D_s^{O_1, O_2, T_{r,pms}}(pms) until it makes the (i+1)-th query
            to the oracle T_{r,pms}, and stop before its answer is obtained;
   h ← (all the oracle queries and their answers obtained
            in the course of the execution);
   if  (the (i+1)-th query to T_{r,pms} is indeed made)
         [ (P, id_ch, L) ← (the (i+1)-th query to T_{r,pms});
         [ return  ((P(r), 1^{|P(r)|}, id_ch), (r, s, h, pms))     ]
   else
         [ return  ((0, 0, 0), (r, s, h, pms)) ]
```

```
Algorithm  A_2^{O_1,O_2}((r, s, h, pms), (id_ch, c))
   Execute D_s^{O_1, O_2, T_{r,pms}}(pms) until it makes the (i+1)-th query
            to T_{r,pms}, using the coin tosses s and emulating the oracles by the record h
            (hence this execution of D is exactly the same as that in A_1);
   For (i+1)-th query (P, id_ch, L) to T_{r,pms} by D, feed D with the answer (c, L(r));
   Continue execution of D using the oracles O_1, O_2, T'_{r,pms}
            (note that the tester oracle is now replaced by the fake one);
   (F, v) ← (the output of the above execution of D);
   t ← (the trace of the above execution of D);
   d ← T(t, F, v, r);      return  d
```

In the above we explicitly specify the coin tosses s and the record of oracle invocations h so that in $\mathcal{A}_2$ we can get exactly the same execution of $\mathcal{D}$ as in $\mathcal{A}_1$. We need the record h since some oracles are probabilistic.

In the execution of $\mathcal{A}_2$, the first two (extraction and decryption) oracles given to $\mathcal{D}$ are such that:

- at first they answer according to the record h;
- after runnding out of the record h, queries are answered by invoking $\mathcal{A}_2$'s oracles O_1 or O_2, respectively.

We denote these (history-dependent) oracles by $O_1|_h$ and $O_2|_h$, respectively. Similarly, in the execution of $\mathcal{A}_2$, the fourth "tester" oracle given to $\mathcal{D}$ is such that:

- the first i queries are answered using the record h;
- the $(i+1)$-th query $(P, \mathsf{id}_{\mathsf{ch}}, L)$ is answered with $(c, L(r))$;
- the rest are answered by $T'_{r,\mathsf{pms}}$.

We denote this oracle by $\varDelta_{r,\mathsf{pms},i,h,c}$.

First we check that the pair $(\mathcal{A}_1, \mathcal{A}_2)$ thus defined indeed qualifies as an $\mathsf{IND\text{-}ID\text{-}CCA}$ distinguisher, in that each of algorithms does not make prohibited queries. $\mathcal{A}_2$ does not query $O_1(\mathsf{id}_{\mathsf{ch}})$: if it does, then in the execution of $\mathcal{D}^{O_1|_h,\, O_2|_h,\, \varDelta_{r,\mathsf{pms},i,h,c}}(\mathsf{pms})$ the algorithm $\mathcal{D}$ makes both queries $\varDelta_{r,\mathsf{pms},i,h,c}(P, \mathsf{id}_{\mathsf{ch}}, L)$ (which is the $(i+1)$-th query on that oracle)[8] and $O_1|_h(\mathsf{id}_{\mathsf{ch}})$. This violates the condition on the $\mathsf{SS\text{-}mID\text{-}mCCA}$ adversary $\mathcal{D}$. Similarly $\mathcal{A}_1$ is shown not to make a query $O_1(\mathsf{id}_{\mathsf{ch}})$. $\mathcal{A}_2$ does not query $O_2(\mathsf{id}_{\mathsf{ch}}, c)$: if it does, then $\mathcal{D}^{O_1|_h,\, O_2|_h,\, \varDelta_{r,\mathsf{pms},i,h,c}}(\mathsf{pms})$ queries $O_2|_h(\mathsf{id}_{\mathsf{ch}}, c)$ *after* it makes the $(i+1)$-th challenge query $(P, \mathsf{id}_{\mathsf{ch}}, L)$ to its tester oracle,[9] and that challenge query is answered with $(c, L(r))$. This violates the condition of the $\mathsf{SS\text{-}mID\text{-}mCCA}$ adversary $\mathcal{D}$.

For the pair $(\mathcal{A}_1, \mathcal{A}_2)$ thus defined, the probabilities $p_\ell^{(1)}$ and $p_\ell^{(2)}$ in the definition of $\mathsf{IND\text{-}ID\text{-}CCA}$ is calculated as follows.

$$p_\ell^{(1)} = \Pr\left[\mathcal{T}(Y_\ell^{(i+1)}) = 1\right] \qquad\qquad p_\ell^{(2)} = \Pr\left[\mathcal{T}(Y_\ell^{(i)}) = 1\right]$$

By the success of $\mathcal{T}$, the distinguisher $(\mathcal{A}_1, \mathcal{A}_2)$ is successful. This contradicts that the scheme has $\mathsf{IND\text{-}ID\text{-}CCA}$. The lemma follows. □

Acknowledgements. The first author thanks Eike Kiltz for helpful comments and for simplifying the separation result in Theorem 5.

References

[ACH+05] N. Attrapadung, Y. Cui, G. Hanaoka, H. Imai, K. Matsuura, P. Yang and R. Zhang. Relations among notions of security for identity based encryption schemes. Cryptology ePrint Archive, Report 2005/258, 2005. http://eprint.iacr.org/.

[AP03] S. AlRiyami and K.G. Paterson. Certificateless public key cryptography. In *ASIACRYPT 2003*, vol. 2894 of *LNCS*, pp. 452–473, 2003. Full version available at `http://eprint.iacr.org/`.

[BB04a] D. Boneh and X. Boyen. Efficient selective-ID secure identity-based encryption without Random Oracles. In *EUROCRYPT 2004*, vol. 3027 of *LNCS*, pp. 223–238, 2004.

[BB04b] D. Boneh and X. Boyen. Secure identity based encryption without Random Oracles. In *CRYPTO 2004*, vol. 3152 of *LNCS*, pp. 443–459, 2004.

[BDPR98] M. Bellare, A. Desai, D. Pointcheval and P. Rogaway. Relations among notions of security for public-key encryption schemes. In - *CRYPTO 1998*, vol. 1462 of *LNCS*, pp. 26–45, 1998.

[8] If $\mathcal{D}$ halts before making the $(i+1)$-th query to the tester oracle, then $\mathcal{A}_2$ makes no oracle queries because all the oracle queries $\mathcal{D}$ makes are answered with the record h.

[9] Otherwise $\mathcal{D}$'s query to $O_2|_h$ is answered using the record h, not by invoking $\mathcal{A}_2$'s oracle O_2.

[BF03] D. Boneh and M. Franklin. Identity-Based encryption from the Weil pairing. *SIAM Journal of Computing*, 32(3):586–615, 2003. This is the full version of an extended abstract of the same title presented at *Crypto'01*.

[BK05] D. Boneh and J. Katz. Improved efficiency for CCA-secure cryptosystems built using identity based encryption. In *CT-RSA 2005*, vol. 3376 of *LNCS*, pp. 87–103, 2005.

[BR93] M. Bellare and P. Rogaway. Random oracles are practical: A paradigm for designing efficient protocols. In *Proceedings of the 1st ACM CCS*, pp. 62–73. ACM Press, 1993.

[BS99] M. Bellare and A. Sahai. Non-malleable encryption: Equivalence between two notions, and an indistinguishability-based characterization. In *CRYPTO 1999*, vol. 1666 of *LNCS*, pp. 519–536, 1999.

[BSS05] J. Baek, R. SafaviNaini and W. Susilo. Efficient multi-receiver identity-based encryption and its application to broadcast encryption. In *PKC 2005*, vol. 3386 of *LNCS*, pp. 380–397, 2005.

[CC05] L. Chen and Z. Cheng. Security proof of Sakai-Kasahara's identity-based encryption scheme. In *IMA Int. Conf. 2005*, LNCS, 2005. To appear.

[CHK03] R. Canetti, S. Halevi and J. Katz. A forward-secure public-key encryption scheme. In *EUROCRYPT 2003*, vol. 2656 of *LNCS*, pp. 255–271, 2003.

[CHK04] R. Canetti, S. Halevi and J. Katz. Chosen-ciphertext security from identity-based encryption. In *EUROCRYPT 2004*, vol. 3027 of *LNCS*, pp. 207–222, 2004.

[DDN00] D. Dolev, C. Dwork and M. Naor. Non-malleable cryptography. *SIAM Journal of Computing*, 30(2):391–437, 2000. An extended version appeared in *Proc. of the 23rd ACM Symposium on Theory of Computing*, 1991.

[DT03] X. Ding and G. Tsudik. Simple identity-based cryptography with mediated RSA. In *CT-RSA 2003*, vol. 1992 of *LNCS*, pp. 193–210, 2003.

[Gal05] D. Galindo. Boneh-Franklin identity based encryption revisited. In *ICALP 2005*, vol. 3580 of *LNCS*, pp. 791–802, 2005.

[Gen03] C. Gentry. Certificate-based encryption and the certificate revocation problem. In *EUROCRYPT 2003*, vol. 2656 of *LNCS*, pp. 272–293, 2003.

[GGM86] O. Goldreich, S. Goldwasser and S. Micali. How to construct random functions. *Journal of the ACM*, 33(4):792–807, 1986.

[GLN02] O. Goldreich, Y. Lustig and M. Naor. On chosen ciphertext security of multiple encryptions. Cryptology ePrint Archive, Report 2002/089, 2002. `http://eprint.iacr.org/`.

[GM84] S. Golwasser and S. Micali. Probabilistic encryption. *Journal of Computer and System Sciences*, 28:270–299, 1984.

[Gol93] O. Goldreich. A uniform-complexity treatment of encryption and zero-knowledge. *Journal of Cryptology*, 6(1):21–53, 1993.

[KI01] K. Kobara and H. Imai. Semantically secure McEliece public-key cryptosystems-conversions for McEliece PKC. In *PKC 2001*, vol. 1992 of *LNCS*, pp. 19–35, 2001.

[LQ05] B. Libert and J.J. Quisquater. Identity based encryption without redundancy. In *ACNS 2005*, vol. 3531 of *LNCS*, pp. 285–300, 2005.

[RS92] C. Rackoff and D.R. Simon. Non-interactive zero-knowledge proof of knowledge and chosen ciphertext attack. In *CRYPTO 1991*, vol. 576 of *LNCS*, pp. 433–444, 1992.

[Sha85] A. Shamir. Identity-based cryptosystems and signature schemes. In *CRYPTO 1984*, vol. 196 of *LNCS*, pp. 47–53, 1985.

[SJ00] C.P. Schnorr and M. Jakobsson. Security of signed El-Gamal encryption. In *ASIACRYPT 2000*, vol. 1976 of *LNCS*, pp. 73–89, 2000.

[Wat05] B. Waters. Efficient identity-based encryption without Random Oracles. In *EUROCRYPT 2005*, vol. 3494 of *LNCS*, pp. 114–127, 2005.

[WSI02] Y. Watanabe, J. Shikata and H. Imai. Equivalence between semantic security and indistinguishability against chosen ciphertext attacks. In *PKC 2003*, vol. 2567 of *LNCS*, pp. 71–84, 2002.

A Appendix

A.1 Lifting restriction on decryption queries for SS-mID-mCCA adversary

As is shown in [GLN02] and briefly explained above, we can assume more powerful SS-mID-mCCA adversaries, namely those without the restriction that they cannot make decryption queries on challenge ciphertexts, and still we obtain the equivalent definition of the notion of SS-mID-mCCA.

For that stronger adversary $\mathcal{D}$, the definition of its trace is now augmented as follows: when $\mathcal{D}$ makes a decryption query $(\mathsf{id}_{\mathsf{ch}}, c)$ for a challenge ciphertext c (i.e. after making a challenge query $(c, *) \leftarrow O_3(P, \mathsf{id}_{\mathsf{ch}}, *)$), then we add to its trace a component (P, P). This modification is based on the idea that we can simulate this decryption query by replacing it by a challenge query with extremely informative information leak, namely P itself.

The following lemma puts the idea precise.

Lemma 17 *For a (stronger) SS-mID-mCCA adversary $\mathcal{D}$ without any restriction on decryption queries, then there exists a (weaker) SS-mID-mCCA adversary $\tilde{\mathcal{D}}$ which:*

- *is as successful as $\mathcal{D}$,*
- *has the same trace as $\mathcal{D}$, and*
- *and follows the restriction on decryption queries, that is, after making a challenge query $(c, *) \leftarrow T_{r,\mathsf{pms}}(*, \mathsf{id}_{\mathsf{ch}}, *)$ a decryption query $D_{\mathsf{mk}}(\mathsf{id}_{\mathsf{ch}}, c)$ is never made.*[10]

Proof: We replace a pattern of oracle queries on the left by that on the right and obtain the execution of $\tilde{\mathcal{D}}^{X_{\mathsf{mk}}, D_{\mathsf{mk}}, T_{r,\mathsf{pms}}}$.

$$\begin{array}{ccc}
\boxed{\mathcal{D}^{X_{\mathsf{mk}}, D_{\mathsf{mk}}, T_{r,\mathsf{pms}}}} & & \boxed{\tilde{\mathcal{D}}^{X_{\mathsf{mk}}, D_{\mathsf{mk}}, T_{r,\mathsf{pms}}}} \\
\vdots & & \vdots \\
(c, b) \leftarrow T_{r,\mathsf{pms}}(P, \mathsf{id}_{\mathsf{ch}}, L); & & (c, b) \leftarrow T_{r,\mathsf{pms}}(P, \mathsf{id}_{\mathsf{ch}}, L); \\
\vdots & & \vdots \\
p \leftarrow D_{\mathsf{mk}}(\mathsf{id}_{\mathsf{ch}}, c); & \overset{\text{replaced by}}{\mapsto} & \left\{ \begin{array}{c} (c', p') \leftarrow T_{r,\mathsf{pms}}(P, \mathsf{id}_{\mathsf{ch}}, P); \\ p \leftarrow p'; \end{array} \right. \\
\vdots & & \vdots
\end{array}$$

On both sides $c = E(\mathsf{pms}, \mathsf{id}_{\mathsf{ch}}, P(r)), b = L(r)$ and $p = P(r)$. Since P and L are circuits (hence deterministic), on both sides the random variables (c, b, p) are identically distributed. Hence we can take $\tilde{\mathcal{D}}$ instead of $\mathcal{D}$ as an adversary, which proves the lemma. □

[10] This weaker notion of adversary with restriction on decryption queries, which is used in the main text of this paper for the simplicity reason, is called *canonical* in [GLN02].