The Exact Security of an Identity Based Signature and its Applications

Benoît Libert[1,2] Jean-Jacques Quisquater [1]

[1]UCL Crypto Group
Place du Levant, 3. B-1348 Louvain-La-Neuve. Belgium
{libert,jjq}@dice.ucl.ac.be
[2] Laboratoire d'Informatique de l'École Polytechnique (LIX)
F-91128 Palaiseau CEDEX, France

Abstract. This paper first positively answers the previously open question of whether it was possible to obtain an optimal security reduction for an identity based signature (IBS) under a reasonable computational assumption. We revisit the Sakai-Ogishi-Kasahara IBS that was recently proven secure by Bellare, Namprempre and Neven through a general framework applying to a large family of schemes. We show that their modified SOK-IBS scheme can be viewed as a one-level instantiation of Gentry and Silverberg's alternative hierarchical IBS the exact security of which was never considered before. We also show that this signature is as secure as the one-more Diffie-Hellman problem. As an application, we propose a modification of Boyen's "Swiss Army Knife" identity based signature encryption (IBSE) that presents better security reductions and satisfies the same strong security requirements with a similar efficiency.

Keywords. ID-based cryptography, signatures, signcryption, security proofs, exact security

1 Introduction

Identity based cryptography has become a very fashionable topic in the last couple of years. The motivation of this concept, introduced by Shamir in 1984 ([55]), was to simplify key management and avoid the use of digital certificates. The trick was to let a public key be publicly and uniquely derivable from a human-memorizable binary sequence corresponding to an information non-ambiguously identifying its owner (e-mail address, IP address combined to a user name, social security number,...) while the associated private keys can only be computed by a trusted Private Key Generator (PKG) thanks to a master secret. This paradigm allows bypassing the trust problems that arise in traditional certificate-based public key infrastructures (PKIs). Indeed, since a public key 'is' its owner's identity, it becomes useless to bind them by a digital certificate. Although a PKG's public key still has to be certified, the need of digital certificates is really reduced as reasonably many users may depend on the same PKG.

Since the concept's appearance in 1984, several practical identity based signature schemes (IBS) have been devised in the late 80's ([29], [31]) and also after 2001 ([17],[34],[54],[50]). On the other hand, finding a practical identity based encryption scheme (IBE) remained an open challenge until 2001 when Boneh and Franklin ([13]) proposed to use bilinear maps over algebraic curves to elegantly solve the challenge. After that, these fashionable bilinear maps provided plenty of other applications (that are not listed here but their references can be found in [3]) including various particular kinds of signatures: blind, ring, undeniable, proxy,etc.

Along the evolution of public key cryptography from 1976, there has been a graduate evolution tending to a necessity to provide security proofs for asymmetric cryptosystems in the sense that the existence of an attacker against them would imply a probabilistic polynomial time algorithm to solve a hard number theoretic problem. In 1993, motivated by the perspective to achieve provable security for efficient protocols, Bellare and Rogaway introduced the random oracle model ([8]) that was previously implicitly suggested in [29] and in which hash functions are used as black box by attackers for whom they are also indistinguishable from perfectly random functions. Although it is well known that security in the random oracle model does not imply security in the real world as shown by several papers ([16],[4]) exhibiting pathological cases of provably secure schemes for which no secure implementation exists, it still seems to be a good principle to give security proofs 'at least' in the random oracle model when proposing a new asymmetric cryptosystem.

In the area of provable security, the last couple of years saw the rise of a new trend consisting of providing tight security reductions for asymmetric cryptosystems ([9],[51],etc.): the security of a cryptographic protocol is said to be tightly related to a hard number theoretic problem if an attacker against the scheme implies an efficient algorithm solving the problem with roughly the same advantage. This led several authors to provide search for new security proofs for systems that were already well known to be secure in the random oracle model or for some of their variants ([22],[23],[46]) or to devise new schemes that, although appearantly less efficient than existing ones at first sight, provide much better security guarantees for the same security parameters and are then eventually more efficient for a similar desired level of security ([32],[36],etc.).

Although concerned with the provable security of identity based signatures, the research community did not really focus on providing really strong security arguments for the various IBS proposed in the literature up to now. Indeed, Paterson's IBS still has no formal security proof while Hess and Cha-Cheon gave proofs under the Diffie-Hellman assumption for their respective scheme but these proofs were both obtained through Pointcheval and Stern's forking lemma ([52],[53]) which does not yield tight security reductions as already argued in several previous papers ([32],[36]). Libert and Quisquater ([40]) recently proposed a scheme that may be viewed as an identity based transformation of the Goh-Jarecki Diffie-Hellman based signature ([32]). Unfortunately, their scheme is computationally expensive and its security relies on the Bilinear Diffie-Hellman assumtion instead of the weaker Computational Diffie-Hellman one.

At Eurocrypt 2004, Bellare, Namprempre and Neven ([6]) defined a framework to provide security proofs for a large family of IBS by considering the security against passive, active and concurrent attacks of underlying 'convertible' identifications schemes (i.e. that can be converted into identity based identification schemes (IBI)). Unfortunately, as mentioned in [48], their framework does not end up with explicitly tight security bounds for the resulting family of IBS that includes the schemes originally described [34],[17],[56],[31],... and a variant of the one in [54]. We think the latter is of particular interest since it is possible to prove its security without considering the underlying identification scheme: in fact, explicitly tight reductions can be obtained without using the BNN framework in a black-box fashion. That is one of the concerns of the present paper.

Another security result was recently achieved by Kurosawa and Heng ([38]) for the Cha-Cheon scheme by also considering the underlying identity based identification scheme. They exhibited a polynomial time reduction from the Diffie-Hellman problem to a chosen-message attacker that avoids the use of the forking technique but their reduction is still quite loose: an attacker with a given advantage ϵ is used to build an algorithm to solve the Diffie-Hellman problem with probability $O(c \cdot \epsilon^2/q_E q_H^2)$ where c is a constant, q_E denotes a bound on the number of identities corrupted by the adversary and q_H is the number of hash queries.

In the present work, we show that Bellare et al.'s modifed Sakai-Ogishi-Kasahara identity based signature ([54]), denoted SOK-IBS in this paper, has a much tighter security proof under the Diffie-Hellman assumption: from an attacker with advantage ϵ, we build a polynomial time algorithm for the Diffie-Hellman problem with an advantage $O(c \cdot \epsilon/q_E)$ and we stress that a fully optimal reduction from a potentially stronger but reasonable assumption exists. According to [36], we think that a tight reduction from a given assumption is preferable to a loose reduction w.r.t. a weaker assumption.

A second contribution of this paper is to revisit Boyen's "Swiss Army Knife" identity based signature-encryption (IBSE) protocol by depicting a scheme achieving the same functionalities and satisfying the same strong security requirements while having much better security reductions than Boyen's IBSE (indeed, the only drawback of the latter construction is to have a poor exact security). Our IBSE follows Boyen's construction. For applications requiring authentication and privacy without necessitating all of IBSE's features, we describe a slightly more efficient 'monolithic' identity based signcryption (IBSC) based on SOK-IBS that is also provably secure in a strong security model but consists of a single layer design.

Before starting with showing our proofs for the modified SOK-IBS, we first recall the properties of bilinear maps that turned out to be (almost) unavoidable tools in the design of identity based cryptosystems. The next section gives formal definitions of presumed hard computational problems from which our reductions are made and then recalls an extended security model for identity based signatures. The improved security analysis of the modified SOK-IBS is given in paragraph 3.2. Finally, its application to ID-based signature/encryption is presented in section 4.

2 Preliminaries

2.1 Bilinear maps and Diffie-Hellman problems

Let k be a security parameter and q be a $k-$bit prime number. Let us consider groups $\mathbb{G}_1$ and $\mathbb{G}_2$ of the same prime order q. For our purposes, we need a bilinear map $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$ satisfying the following properties:

1. Bilinearity: $\forall\ P, Q \in \mathbb{G}_1,\ \forall\ a, b \in \mathbb{Z}_q^*$, we have $\hat{e}(aP, bQ) = \hat{e}(P, Q)^{ab}$.
2. Non-degeneracy: for any $P \in \mathbb{G}_1$, $\hat{e}(P, Q) = 1$ for all $Q \in \mathbb{G}_1$ iff $P = \mathcal{O}$.
3. Computability: there is an efficient algorithm to compute $\hat{e}(P, Q)\ \ \forall\ P, Q \in \mathbb{G}_1$.

As shown in [13], such non-degenerate admissible maps over cyclic groups can be obtained from the Weil or the Tate pairing over supersingular elliptic curves or abelian varieties.

We now recall the definitions of the Computational Diffie-Hellman problems and of one of its variants.

Definition 1. *Let us consider cyclic group $\mathbb{G}_1$ of prime order q,*
- *The **Computational Diffie-Hellman problem** (CDH) in $\mathbb{G}_1$ is, given $\langle P, aP, bP \rangle$ for unknown $a, b \in \mathbb{Z}_q$, to compute $abP \in \mathbb{G}_1$.*
- *The **one more CDH problem** (1m-CDH) is, given $\langle P, aP \rangle \in \mathbb{G}_1$ for an unknown $a \in \mathbb{Z}_q$, and access to a target oracle $\mathcal{T}_{\mathbb{G}_1}$ returning randomly chosen elements $Y_i \in \mathbb{G}_1$ (for $i = 1, \ldots, q_t$, q_t being the exact number of queries to this oracle) as well as a multiplication oracle $\mathcal{H}_{\mathbb{G}_1,a}(.)$ answering $aW \in \mathbb{G}_1$ when queried on an input $W \in \mathbb{G}_1$, to produce a list $((Z_1, j_1), \ldots, (Z_{q_t}, j_{q_t}))$ of q_t pairs such that $Z_i = aY_{j_i} \in \mathbb{G}_1$ for all $i = 1, \ldots, q_t$, $1 \leq j_i \leq q_t$ and $q_m < q_t$ where q_m denotes the number of queries made to the multiplication oracle.*

The one more CDH problem was introduced in [10] to prove the security of a Blind signature ([18]) obtained from the BLS signature ([14]). Its commonly assumed intractability was more recently used in [38] and [6] to prove the security of identification schemes built on top of identity based signatures. In the upcoming sections, we will sometimes refer to this assumption as the "one more CDH assumption".

2.2 Security notions for identity based signatures

We consider the notion of strong existential unforgeability already considered in [1] and [11] that is slightly stronger than the usual notion of existential unforgeability under chosen-message attacks introduced in [33].

Definition 2. *An identity based signature scheme is said to be **strongly existentially unforgeable** under chosen-message attacks if no probabilistic polynomial time (PPT) adversary has a non-negligible advantage in this game:*

1. *The challenger runs the setup algorithm to generate the system's parameters and sends them to the adversary.*
2. *The adversary $\mathcal{F}$ performs a series of queries:*
   - *Key extraction queries: $\mathcal{F}$ produces an identity ID and receives the private key d_{ID} corresponding to ID.*
   - *Signature queries: $\mathcal{F}$ produces a message M and an identity ID and receives a signature on M that was generated by the signature oracle using the private key corresponding to the identity ID.*

3. *After a polynomial number of queries, $\mathcal{F}$ produces a tuple (ID^*, M^*, σ^*) made of an identity ID^*, whose corresponding private key was never asked during stage 2, and a message-signature pair (M^*, σ^*) such that σ^* was not returned by the signature oracle on the input (M^*, ID^*) during stage 2 for the identity ID^*.*

The forger $\mathcal{F}$ wins the game if the signature verification algorithm outputs 1 when it is run on the tuple (ID^*, M^*, σ^*). The forger's advantage is defined to be its probability of producing a forgery taken over the coin-flippings of the challenger and $\mathcal{F}$.

The above security notion was not considered in previous papers tackling with identity based signatures but it is interesting to notice that the schemes proposed in [17] and [34] are also provably secure in this strengthened model.

3 An identity based signature with tight security reductions

The present section revisits the modified Sakai-Ogishi-Kasahara signature ([6]) by considering it as a one level instantiation of a (randomized) version of Gentry and Silverberg's alternative hierarchical IBS ([30]). This scheme is the same as the one obtained by applying Bellare et al.'s extended Fiat-Shamir heuristic ([6]) to the SOK identity based identification scheme (that is only secure against passive attacks as shown in [6]).

3.1 The scheme

The signature that was commonly called SOK-IBS in ([6]) (for Sakai-Ogishi-Kasahara Identity Based Signature) is made of four algorithms that are depicted on figure 1.

Setup: given a security parameter k, the PKG chooses groups $\mathbb{G}_1$ and $\mathbb{G}_2$ of prime order $q > 2^k$, a generator P of $\mathbb{G}_1$, a randomly chosen master key $s \in \mathbb{Z}_q^*$ and the associated public key $P_{pub} = sP$. It also picks cryptographic hash functions of same domain and range $H_1, H_2 : \{0,1\}^* \rightarrow \mathbb{G}_1^*$. The system's public parameters are

$$\texttt{params} = (\mathbb{G}_1, \mathbb{G}_2, \hat{e}, P, P_{pub}, H_1, H_2).$$

Keygen: given an user's identity ID, the PKG computes $Q_{ID} = H_1(ID) \in \mathbb{G}_1$ and the associated private key $d_{ID} = sQ_{ID} \in \mathbb{G}_1$ that is transmitted to the user.

Sign: in order to sign a message M,

1. pick $r \xleftarrow{R} \mathbb{Z}_q$ and compute $U = rP \in \mathbb{G}_1$ and then $H = H_2(ID, M, U) \in \mathbb{G}_1$.
2. Compute $V = d_{ID} + rH \in \mathbb{G}_1$.

The signature on M is the pair $\sigma = \langle U, V \rangle \in \mathbb{G}_1 \times \mathbb{G}_1$.

Verify: to verify a signature $\sigma = \langle U, V \rangle \in \mathbb{G}_1 \times \mathbb{G}_1$ on a message M for an identity ID, the verifier first takes $Q_{ID} = H_1(ID) \in \mathbb{G}_1$ and $H = H_2(ID, M, U) \in \mathbb{G}_1$. He then accepts the signature if $\hat{e}(P, V) = \hat{e}(P_{pub}, Q_{ID})\hat{e}(U, H)$ and rejects it otherwise.

Fig. 1. The SOK-IBS scheme

From an efficiency point of view, the signature issuing algorithm has the same complexity as Cha and Cheon's one ([17]) while the verification algorithm is slightly more expensive: it requires 3 pairing computations against only two for the verification algorithms of the schemes described in [17] and [34]. As will be shown in the next subsection, the advantage of the present scheme over the previously cited ones is to yield much better security guarantees in the random oracle model for similar security parameters.

Interestingly, although derived from an identity based identification scheme (IBI) that is only secure against passive attacks, the modified SOK-IBS has better reductions than other IBS for which the underlying IBI is secure against stronger attacks. In fact, it can also be regarded as an identity based extension of a randomized version of Boneh et al.'s short signature ([14]). That is the reason why so tight security reductions can be obtained.

3.2 The exact security of SOK-IBS

This security analysis first presents a security reduction from the Diffie-Hellman problem to a chosen-message attacker against SOK-IBS that is more efficient than any other known security reduction (including those given in [38],[6]) for existing identity based signatures ([17],[34],etc.). In a second step, we explain how to achieve an optimal reduction from the one more Diffie-Hellman problem.

Theorem 1. *In the random oracle model, if a PPT forger $\mathcal{F}$ has an advantage ϵ in forging a signature in an attack modelled by the game of definition 2 when running in a time t and asking q_{H_i} queries to random oracles H_i (i=1,2), q_E queries to the key extraction oracle and q_S queries to the signature oracle, then the CDH problem can be solved with an advantage*

$$\epsilon' > \frac{\epsilon - \big(q_S(q_{H_2} + q_S) + 1\big)/2^k}{e(q_E + 1)}$$

within a time $t' < t + (q_{H_1} + q_{H_2} + q_E + 2q_S)t_m + (q_S + 1)t_{mm}$ where e denotes the base of the natural logarithm, t_m is the time to compute a scalar multiplication in $\mathbb{G}_1$ and t_{mm} is the time to perform a multi-exponentiation in $\mathbb{G}_1$.

Proof. We start by describing how a forger $\mathcal{F}$ can be used by a probabilistic polynomial time algorithm $\mathcal{B}$ to solve the CDH problem. Let $(X = xP, Y = yP) \in \mathbb{G}_1 \times \mathbb{G}_1$ be a random instance of the CDH problem taken as input by $\mathcal{B}$. The latter initializes $\mathcal{F}$ with $P_{pub} = X$ as a system's overall public key. The forger $\mathcal{F}$ then starts performing queries such as those described in definition 2. These queries are answered by $\mathcal{B}$ as follows (without loss of generality, we assume that, for any key extraction query or signature query involving an identity, a H_1 oracle query was previously issued for the same identity):

- queries on oracle H_1: when an identity ID is submitted to the H_1 oracle, as in Coron's proof technique ([22]), $\mathcal{B}$ flips a coin $T \in \{0, 1\}$ that yields 0 with probability δ and 1 with probability $1 - \delta$. $\mathcal{B}$ then picks $u \xleftarrow{R} \mathbb{Z}_q^*$. If $T = 0$ then the hash value $H_1(ID)$ is defined as being $uP \in \mathbb{G}_1$. If $T = 1$, then $\mathcal{B}$ returns $uY \in \mathbb{G}_1$. In both cases, $\mathcal{B}$ inserts a tuple (ID, u, T) in a list L_1 to keep track of the way it answered the query.
- Key extraction queries: when $\mathcal{F}$ requests the private key associated to an identity ID, $\mathcal{B}$ recovers the corresponding (ID, u, T) from L_1 (recall that such a tuple must exist because of the aforementioned assumption). If $T = 1$, then $\mathcal{B}$ outputs "failure" and halts because it is unable to coherently answer the query. Otherwise, it means that $H_1(ID)$ was previously defined to be $uP \in \mathbb{G}_1$ and $uP_{pub} = uX \in \mathbb{G}_1$ is then returned to $\mathcal{F}$ as a private key associated to ID.
- queries on oracle H_2: when a tuple (ID, M, U) is submitted to the H_2 oracle, $\mathcal{B}$ first scans a list L_2 to check whether H_2 was already defined for that input. If it was, the previously defined value is returned. Otherwise, $\mathcal{B}$ picks a random $v \xleftarrow{R} \mathbb{Z}_q^*$, stores the tuple (ID, M, U, v) in the list L_2 and returns $vP \in \mathbb{G}_1$ as a hash value to $\mathcal{F}$.
- Signature queries: when $\mathcal{F}$ queries the signature oracle on a message M for an identity ID, $\mathcal{F}$ first recovers the previously defined value $Q_{ID} = H_1(ID) \in \mathbb{G}_1$ from L_1. It then chooses $t, \nu \xleftarrow{R} \mathbb{Z}_q^*$ before setting $V = tP_{pub} = tX \in \mathbb{G}_1$, $U = \nu P_{pub} = \nu X \in \mathbb{G}_1$ and defining the hash value $H_2(ID, M, U)$ as $\nu^{-1}(tP - Q_{ID}) \in \mathbb{G}_1$ ($\mathcal{B}$ halts and outputs "failure" if H_2 turns out to be already defined for the input (ID, M, U)). The pair (U, V) is returned to $\mathcal{F}$ and appears as a valid signature from the latter's point of view.

Eventually, the forger $\mathcal{F}$ produces a message M^*, an identity ID^* and a fake signature $\langle U^*, V^* \rangle$ for the pair (M^*, ID^*) and $\mathcal{B}$ then recovers the triple (ID^*, u^*, T^*) from L_1. If $T^* = 0$, then $\mathcal{B}$ outputs "failure" and stops. Otherwise, it goes on and the list L_2 must contain an entry (ID^*, M^*, U^*, v^*) with overwhelming probability (otherwise, $\mathcal{B}$ stops and outputs "failure"). Hence, since $H^* = H_2(ID^*, M^*, U^*)$ was defined to be $v^*P \in \mathbb{G}_1$, if $\mathcal{F}$ succeeded in the game with the view it was provided with, $\mathcal{B}$ knows that

$$\hat{e}(P, V^*) = \hat{e}(X, Q_{ID^*})\hat{e}(U^*, H^*)$$

with $H^* = v^*P \in \mathbb{G}_1$ and $Q_{ID^*} = u^*Y \in \mathbb{G}_1$ for some known elements $u^*, v^* \in \mathbb{Z}_q^*$. Then, it also knows that

$$\hat{e}(P, V^* - v^*U^*) = \hat{e}(X, u^*Y)$$

and that $u^{*-1}(V^* - v^*U^*) \in \mathbb{G}_1$ is the solution to the CDH instance $(X, Y) \in \mathbb{G}_1 \times \mathbb{G}_1$.

When assessing $\mathcal{B}$'s probability of failure, one readily checks that its probability to fail in handling a signing query because of a conflict on H_2 is at most $q_S(q_{H_2} + q_S)/2^k$ (as L_2 never contains more than $q_{H_2} + q_S$ entries) while the probability for $\mathcal{F}$ to output a valid forgery $\langle U^*, V^* \rangle$ on M^* without asking the corresponding $H_2(ID^*, M^*, U^*)$ query is at most $1/2^k$. Finally, by an analysis similar to Coron's one ([22]), the probability $\delta^{q_E}(1 - \delta)$ for $\mathcal{B}$ not to fail in a key extraction query or because $\mathcal{F}$ produces its forgery on a 'bad' identity ID^* is greater than $1 - 1/e(q_E + 1)$ when the optimal probability $\delta_{opt} = q_E/(q_E + 1)$ is used when handling key extraction queries. Eventually, it comes that $\mathcal{B}$'s advantage is at most

$$\frac{\epsilon - \big(q_S(q_{H_2} + q_S) + 1\big)/2^k}{e(q_E + 1)}. \qquad \square$$

Efficiency of the reduction. We note that the obtained reduction is tighter than for any previously known ID-based signature scheme: at this stage, our bound on ϵ' is already much better than Kurosawa and Heng's one ([38]) that was $O(\epsilon^2/eq_Eq_H^2)$. As an example, for $k = 160$, if we allow $q_{H_1}, q_{H_2} < 2^{60}$ and $q_E, q_S < 2^{30}$, we have $q_S(q_{H_2}+q_S)/2^k < 2\times 2^{90}/2^{160} = 2^{-69}$. If we assume that the advantage of an attacker in solving CDH is at most $\epsilon' < 2^{-60}$, we obtain that $(\epsilon - 2^{-69})/2^{32} \leq \epsilon' < 2^{-60}$ and the probability for an attacker to break SOK-IBS is bounded [1] by $\epsilon \leq 2^{-28}+2^{-69} < 2\times 2^{-28} = 2^{-27}$. Such a reduction will be called "sub-optimal" in the sequel.

Achieving an optimal reduction. The theorem below shows that an optimal reduction exists from the potentially stronger one more CDH assumption. The advantage in solving the one more CDH problem is, up to a negligible term, as large as the forger's advantage. The proof is quite simple.

Theorem 2. *In the random oracle model, if a PPT adversary $\mathcal{F}$ has an advantage ϵ against the strong unforgeability of SOK-IBS in a chosen-message attack when running in a time t, asking q_{H_i} queries to random oracles H_i (i=1,2), q_E key extraction queries and q_S queries to the signature oracle, then there is an algorithm $\mathcal{B}$ to solve the one more CDH problem with an advantage $\epsilon' > \epsilon - \big(q_S(q_{H_2}+q_S)+1\big)/2^k$ in a time $t' < t+(q_{H_2}+2q_S)t_m+(q_S+1)t_{mm}$ where t_m is the cost of a scalar multiplication in $\mathbb{G}_1$ and t_{mm} is the time to perform a multi-exponentiation in $\mathbb{G}_1$.*

Proof. Let $\langle P, X = aP, \mathcal{T}_{\mathbb{G}_1}, \mathcal{H}_{\mathbb{G}_1,a}(.)\rangle$ be an instance of the one more CDH problem. To solve it, the simulator $\mathcal{B}$ runs $\mathcal{F}$ with the domain-wide key $P_{pub} = aP \in \mathbb{G}_1$. The forger $\mathcal{F}$ then starts querying the various oracles that are simulated as follows:

- queries on oracle H_1: when a new identity ID_i is submitted to this oracle, $\mathcal{B}$ queries the target oracle $\mathcal{T}_{\mathbb{G}_1}$ (recall that this oracle takes no input) and forwards the obtained random element $Y_i \in \mathbb{G}_1$ as an answer to $\mathcal{F}$. The pair (ID_i, Y_i) is stored in a list L_1. If the same identity is submitted to H_1 again, the stored answer is returned.

[1] We have to mention that the upper bound on $\mathcal{B}$'s probability to fail in simulating the signing oracle could be improved by using another proof technique due to Coron ([23]) but, in the present situation, this would degrade the bound t' on $\mathcal{B}$'s running time.

- Private key queries on identities ID_i: we assume ID_i was previously submitted to the H_1 oracle. The corresponding $Y_i \in \mathbb{G}_1$ that was obtained from $\mathcal{T}_{\mathbb{G}_1}$ is recovered from L_1 and sent by $\mathcal{B}$ to the multiplication oracle $\mathcal{H}_{\mathbb{G}_1,a}(.)$ whose output $aY_i \in \mathbb{G}_1$ is returned to $\mathcal{F}$ as a private key for ID_i. The elements (ID_i, Y_i, aY_i) are stored in a list L_E.
- H_2 queries and signing queries are dealt with exactly as in the proof of theorem 1.

Since $\mathcal{F}$ is assumed to produce a forgery for an uncorrupted identity ID^*, and since we can assume that $H_1(ID^*)$ was asked during the game, it follows that the number q_{H_1} of target oracle queries made by $\mathcal{B}$ is strictly smaller than the number q_E of queries to $\mathcal{H}_{\mathbb{G}_1,a}(.)$. Furthermore, the private key $d_{ID^*} = V^* - v^*U^*$ associated to the uncorrupted identity ID^* can be extracted from the outputted forgery $(M^*, \langle U^*, V^*\rangle)$ and from the content of the list L_2 (where $H_2(ID^*, M^*, U^*)$ was defined to be $v^*P \in \mathbb{G}_1$) since we have the equality $\hat{e}(P, V^* - v^*U^*) = \hat{e}(X, Y^*)$ and Y^* is the value of $H_1(ID^*)$ fixed by $\mathcal{T}_{\mathbb{G}_1}$.

□

We now obtain an excellent exact security: with $k = 160$ and the previously given values of q_{H_i}, for $i = 1, 2$ and q_S, assuming that the one more CDH problem cannot be solved with a probability ϵ' greater than 2^{-60}, then no attacker can break the scheme with a better advantage than $\epsilon < 2^{-60} + 2^{-69} < 2^{-59}$. Finding a so tight security reduction for an identity based signature appeared as an open problem before this work. Finding an optimal reduction under a more standard assumption still remains an open challenge.

4 Identity based signcryption schemes with tighter security reductions

Since 2002, a couple of identity based protocols jointly performing signature and encryption have been studied ([15],[20],[42],[39],[47],[19],[44]). In 2003 ([15]), Boyen proposed a nice two-layer design of signature/encryption (IBSE) that provably satisfies strong security notions among which chosen-ciphertext security, signature non-repudiation, ciphertext authentication and ciphertext anonymity. This scheme also provides detachable signatures that cannot be linked to the original ciphertext (this property was called ciphertext unlinkability in [15]). The only drawback of Boyen's IBSE is not to have tight security reductions (just like its improvement, recently proposed in [19], that does not have the ciphertext unlinkability property), especially in the proof of unforgeability that relies on Pointcheval and Stern's forking lemma ([52],[53]). In fact, none of the previously known identity based signcryption schemes (IBSC) can be proved to be unforgeable with a tight reduction.

SOK-IBS provides a very natural construction based on a randomness re-use to perform encryption. This construction results in an identity based signature/encryption that provably satisfies the same security requirements as Boyen's scheme with a similar efficiency and much better reductions (especially for the non-repudiation aspect). In the security analysis, because of space limitation, we restricted ourselves to only detail proofs of CCA-security (even against "insider" attacks as for the schemes in [15] and [20]) and signature strong unforgeability but our scheme can also be shown to be also anonymous (i.e. a ciphertext conveys no information on its intended recipient nor about who its originator is) and to allow detachable signatures that are unlinkable to their associated ciphertexts.

By a simple tranformation, the SOK-IBSE construction can be turned into a somewhat more efficient monolithic identity based signcryption (IBSC) that offers better security guarantees than other previous provably secure IBSC proposals ([42],[39],etc.) and that is also provably anonymous (unlike the solution proposed in [41],[44],[20]). It is useful for applications that do not require to extract signed message from ciphertexts in such a way that extracted signatures cannot be linked to the original ciphertext they are embedded in. A strengthened security model for such monolithic IBSC schemes is presented in an extended version of the paper.

Interestingly, the new IBSC construction not only allows to send a single signed message to multiple recipients but, unlike Boyen's constructions and all known identity based signcryption schemes, it allows to send n possibly different signcrypted messages to n distinct recipients using a single random coin for both authentication and encryption purposes.

Before presenting our solution, let us first recall definitions of underlying hard problems on which the security of our scheme is shown to rely.

4.1 Underlying hard problems

Definition 3. *Given groups $\mathbb{G}_1$ and $\mathbb{G}_2$ of prime order q, a bilinear map $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$ and a generator P of $\mathbb{G}_1$,*

- *The **Bilinear Diffie-Hellman problem** (BDH) is, given $\langle P, aP, bP, cP \rangle$ for unknown $a, b, c \in \mathbb{Z}_q$, to compute $\hat{e}(P, P)^{abc} \in \mathbb{G}_2$.*
- *The **Decision Bilinear Diffie-Hellman problem** (DBDH) is, given $P, aP, bP, cP \in \mathbb{G}_1$ and $h \in \mathbb{G}_2$ to decide whether $h = \hat{e}(P, P)^{abc}$ or not. According to the terminology of [49], tuples of the form $\langle P, aP, bP, cP, \hat{e}(P, P)^{abc} \rangle \in {\mathbb{G}_1}^4 \times \mathbb{G}_2$ are called Bilinear Diffie-Hellman tuples.*
- *The **Gap Bilinear Diffie-Hellman problem** (GBDH) is, given $\langle P, aP, bP, cP \rangle$, for unknown $a, b, c \in \mathbb{Z}_q^*$, to compute $\hat{e}(P, P)^{abc}$ with the help of a DBDH oracle that is able to decide within a unit time whether a tuple $\langle P, a'P, b'P, c'P, h' \rangle \in {\mathbb{G}_1}^4 \times \mathbb{G}_2$ is such that $h' = \hat{e}(P, P)^{a'b'c'}$ or not.*

The DBDH problem was recently used in [12] to prove the security of an identity based encryption scheme in a standard computational model. Its was previously considered in [40] where the GBDH problem (whose denomination emanates from a terminology due to Okamoto and Pointcheval ([49])) was considered for the first time.

4.2 The SOK-IBSE scheme

We call this new scheme SOK-IBSE (for SOK-like Identity Based Signature-Encryption). The protocol is given by the six algorihtms depicted on figure 2. As done in [15], we assume the receiver of a ciphertext has no a priori knowledge on the sender's identity that is thus encrypted together with the plaintext.

A random salt τ of length δ is also encrypted together with the message and the sender's identity at the encryption operation. This random string, that is hashed with the sender and receiver's identities when computing the X component of the ciphertext from the U component of the signature, aims at providing the ciphertext unlinkability property by preventing someone observing a signed message $(M, \langle U, V \rangle)$ for a signer's identity ID_A to know how this message-signature pair should be encrypted into a ciphertext intended to some receiver ID_B. That is why it is also encrypted into one of the ciphertext's components to allow the receiver to detach the signature.

We re-use Boyen's construction ([15]) but, in order to achieve better security reductions, we also hash the ciphertext's X component and the recipient's public key Q_{ID_B} together with the pairing's result at the encryption operation.

Efficiency discussions. From an efficiency point of view, for a given security parameter k, SOK-IBSE has exactly the same cost as Boyen's scheme while ciphertexts produced by the latter are δ bits shorter. A naive composition of SOK-IBS with the Boneh-Franklin IBE or with any of its variants (e.g. the one in [2]) could not have final ciphertexts shorter than $4\ell + n_0 + n_1$ (indeed, the whole signature would need to be encrypted with the message and the sender's identity for ciphertext unlinkability purposes) while those of SOK-IBSE are no longer than $3\ell + n_0 + n_1$ in the worst case (we can assume $\delta \leq \ell$ as explained below).

Other comparisons with Boyen's IBSE. The size δ of the random seed, that is exactly the difference between the length of ciphertexts in Boyen's scheme and their size in SOK-IBSE, is flexible: it is an unlinkabilitiy parameter depending on the specific application: if detached signatures are required to be computationally unlinkable to their ciphertexts, it is recommended to take $\delta = 160$. If senders only need to be able to deny having created any ciphertext from a

Setup: is identical to its counterpart in SOK-IBS except that five hash functions are needed: $H_1 : \{0,1\}^* \to \mathbb{G}_1$, $H_2 : \{0,1\}^{n_0+n_1+\ell} \to \mathbb{G}_1$, $H_3 : \mathbb{G}_1{}^2 \times \mathbb{G}_2 \to \{0,1\}^\ell$, $H_4 : \mathbb{G}_1 \to \{0,1\}^\lambda$ and $H_5 : \{0,1\}^{2n_1+\delta} \to \mathbb{Z}_q^*$ where ℓ is the bitlength of $\mathbb{G}_1$'s elements. The space of plaintexts is $\mathcal{M} := \{0,1\}^{n_0}$, and the ciphertext space is $\mathcal{C} := \mathbb{G}_1 \times \{0,1\}^\ell \times \{0,1\}^{n_0+n_1+\delta}$ where n_1 denotes the maximum length of identifiers and δ is another security parameter explained below. A symmetric cipher $(\mathcal{E}, \mathcal{D})$ of keylength λ is also chosen.

Keygen: is the same as in SOK-IBS.

Sign: given a message M and the sender's private key d_{ID_A},
1. pick $r \xleftarrow{R} \mathbb{Z}_q^*$, compute $U = rP$ and $H = H_2(ID_A, M, U) \in \mathbb{G}_1$,
2. set $V = d_{ID_A} + rH \in \mathbb{G}_1$,
3. return $\langle M, r, U, V, ID_A \rangle$. to Encrypt

Encrypt: given $\langle M, r, U, V, ID_A \rangle$ and the recipient's identity $Q_{ID_B} = H_1(ID_B) \in \mathbb{G}_1$,
1. compute $x = H_5(ID_A, ID_B, \tau) \in \mathbb{Z}_q$ for $\tau \xleftarrow{R} \{0,1\}^\delta$ and $X = xU \in \mathbb{G}_1$,
2. set $W = V \oplus H_3(X, Q_{ID_B}, g^{xr})$, with $g = \hat{e}(P_{pub}, Q_{ID_B})$
3. compute $Z = \mathcal{E}_\kappa(M||ID_A||\tau)$, with $\kappa = H_4(V) \in \{0,1\}^\lambda$, and the final ciphertext is $\langle X, W, Z \rangle$.

Decrypt: given a ciphertext $\langle X, W, Z \rangle$ and the private key d_{ID_B},
1. compute $V = W \oplus H_3(X, Q_{ID_B}, g')$ with $g' = \hat{e}(X, d_{ID_B})$ and then $(M||ID_A||\tau) = \mathcal{D}_\kappa(Z)$ with $\kappa = H_4(V) \in \{0,1\}^\lambda$,
2. compute $Q_{ID_A} = H_1(ID_A) \in \mathbb{G}_1$, $U = x^{-1}X \in \mathbb{G}_1$ with $x = H_5(ID_A, ID_B, \tau) \in \mathbb{Z}_q^*$.
3. return $\langle M, ID_A, U, V \rangle$ to Verifiy

Verify: given $\langle M, ID_A, U, V \rangle$,

Return 1 if $\hat{e}(P, V) = \hat{e}(P_{pub}, Q_{ID_A})\hat{e}(U, H)$. where $H = H_2(ID_A, M, U) \in \mathbb{Z}_q^*$ and $Q_{ID_A} = H_1(ID_A) \in \mathbb{G}_1$ and 0 otherwise.

Fig. 2. The SOK-IBSE Signature/Encryption scheme

third party's view and to discourage anyone to try and link their detached signatures to any ciphertext, a short seed of 20 bits is sufficient. Finally, if a user does not need his/her signatures to be unlinkable to the ciphertext they are embedded in, one can set $\delta = 0$ and $x = 1$ to obtain a monolithic single-layer identity based signcryption (IBSC). It is then recommended to hash the recipient's identity ID_B together with the sender's one in the evaluation of $H \in \mathbb{G}_1$ upon the ciphertext's construction in order for the notion of ciphertext strong unforgeability to be satisfied in the monolithic security model (which is described in an extended version of this paper). The resulting efficient IBSC is described in appendix C.

We must conceed that this method to achieve ciphertext unlinkabiltiy is not practical for multi-recipient signature/encryption purposes (unless one is not concerned with the leakage of the information that all components of the multi-recipient ciphertext hide the same signed message) but, in such a scenario, one can easily use Boyen's method (that consists of setting $x = H_5(\hat{e}(d_{ID_A}, Q_{ID_B})) = H_5(\hat{e}(Q_{ID_A}, d_{ID_B}))$ instead of using a hidden random seed τ) without being concerned with the irreflexivity assumption (i.e. the assumption that a ciphertext's sender and receiver are always distinct entities) that causes a further degradation when dropped in [15] (indeed the use of a decision BDH oracle helps in achieving a perfect simulation).

A multi-authority scalable IBSE. Interestingly, the use of a random seed to handle unlinkability concerns allows the sender and the receiver not to depend on the same PKG since the hash value of the sender's identity is never paired with the receiver's private key (and vice versa). As a result, if all users have confidence in the public keys of all PKGs (this could be achieved by the use of PKG certificates), a sender can be imagined to append a tag indicating which PKG he/she depends on to his/her identity when completing the encryption operation. A tradeoff between identity and certificateful public key cryptography is then obtained: digital certificates are again necessary but their use is much more moderate than in traditional PKIs (since only PKGs public keys need to be certified) and, for large scale applications requiring both privacy and authentication, this is much more flexible than a sequential composition of hierarchical ID-based signature and encryption. What is more, this new certificate-and-identity based signature/encryption solution preserves the anonymity of ciphertexts unlike hierarchical schemes (recall that, in Gentry and Silverberg's ones, a ciphertext's length reveals the depth of its intended recipient in the hierarchy).

The security proofs presented in the next section only consider the case of a single authority but they can be easily adapted to the multi-authority setting.

4.3 Security analysis of SOK-IBSE

We refer to appendix A for a recall of the security models of chosen-ciphertext security and signature non-repudiation defined by Boyen ([15]) for IBSE schemes. The following theorems, for which the proofs can be found in appendix, claim that SOK-IBSE satisfies these two notions with tighter security bounds than the original IBSE. It can also be shown to satisfy the other notions of ciphertext authentication and ciphertext anonymity that are also formally described in [15].

Theorem 3. *In the random oracle model, if a chosen-ciphertext adversary $\mathcal{A}$ has an advantage ϵ against SOK-IBSE when running in a time t, making q_{H_i} queries to the random oracles H_i ($i = 1, \dots, 4$), q_{SE} signature/encryption queries and q_{DV} decryption/verification queries, then there exists a PPT algorithm $\mathcal{B}$ solving the GBDH problem with an advantage*

$$\epsilon' \geq \frac{1}{e(q_E+1)}\Big(\epsilon - q_{SE}\frac{q_{SE}+q_{H_2}}{2^k} - q_{DV}(\frac{q_{H_4}}{2^\ell} + \frac{1}{2^k})\Big)$$

within a time $t' \leq t + (q_{H_1} + q_E + 3q_{SE})t_m + q_{DV}t_{exp} + 2q_{DV}t_p + q_{H_3}O(1)$ where e denotes the base of the natural logarithm, t_p is the time to perform a pairing evaluation, t_m is the cost of a multiplication in $\mathbb{G}_1$ and t_{exp} is the time to perform an exponentiation in $\mathbb{G}_2$.

Proof. given in appendix. □

We observe that the reduction is much tighter than for any previously known identity based authenticated encryption scheme: for common security parameters $k = \ell = 160$, if we allow $q_{H_i} \leq 2^{60}$ for $i = 1, \dots, 4$ and $q_{SE}, q_{DV} \leq 2^{30}$, we have $\epsilon' \geq 2^{-32}\epsilon - 2^{-100}$. Unfortunately, the bound remains sub-optimal since we still have a loss of 2^{30}.

The next theorem claims the strong existential unforgeability of SOK-IBSE.

Theorem 4. *In the random oracle model, if an adversary $\mathcal{F}$ has an advantage ϵ against the strong existential unforgeabiltiy (ESUF-IBSC-CMCA) of SOK-IBSE when running in a time t, making q_{H_i} queries to the random oracles H_i ($i = 1, \dots, 4$), q_{SE} signature/encryption queries and q_{DV} decryption/verification queries, then there exists an algorithm $\mathcal{B}$ to solve the GBDH problem in a time with an advantage*

$$\epsilon' \geq \frac{1}{e(q_E+1)}\Big(\epsilon - q_{SE}\frac{q_{SE}+q_{H_2}}{2^k} - q_{DV}(\frac{q_{H_4}}{2^\ell} + \frac{1}{2^k}) - 2^{-\ell} - 2^{-k} - 2^{n_0+n_1+\delta-2\lambda}\Big)$$

within a time $t' \leq t + (q_{H_1} + q_E + 3q_{SE})t_m + q_{DV}t_{exp} + 2(q_{DV}+1)t_p + q_{H_3}O(1)$ where e is the base of the natural logarithm, t_p is the time to perform a pairing evaluation, t_m denotes the cost of a multiplication in $\mathbb{G}_1$ and t_{exp} is the time to perform an exponentiation in $\mathbb{G}_2$.

Proof. given in appendix □

4.4 A randomness re-using multi-recipient IBSC

In 2002, Kurosawa ([37]) proposed discrete logarithm based randomness re-using multi-recipient encryption schemes obtained from the El Gamal ([28]) and the Cramer-Shoup ([24]) cryptosystems. This kind of scheme aims at using a single random coin to encrypt several possibly different messages intended to distinct receivers. Kurosawa also described a formal security model that does not consider the most powerful attacks. At PKC 2003, Bellare et al. ([5]) presented a more rigorous and stringent security model for this kind of multi-recipient encryption scheme by considering the fact that an adversary may be one of the recipients of a multi-component ciphertext that aims at threatening the confidentiality of another component intended to another recipient by gathering information on the sender's coin. In [5], they showed that several so-called 'reproducible' (in fact, they showed that a sufficient condition for a scheme to allow a secure randomness re-use was its reproducibility, see [5] for details) discrete logarithm based cryptosystems such as El Gamal or Cramer-Shoup allow such a secure randomness re-use.

A question that naturally arises is whether such a secure randomness re-use is also possible for public key authenticated encryption. We positively answer this question by showing that SOK-IBSC allows to use a single coin $r \in \mathbb{Z}_q$ to sign N messages and encrypt them under N distinct identities without exposing the sender's private key. The overall multi-component ciphertext has thus the form $\langle U, (W_1, Z_1), \ldots, (W_N, Z_N) \rangle$ and the sender then spares $O(N)$ multiplications in $\mathbb{G}_1$ as well as a $O(N)$ communication cost when sending N signcrypted messages to N receivers. In appendix, we show a formal security model (that takes into account the threat of inside attackers as in [5]) for randomness re-using multi-recipient identity based signcryption schemes and security proofs will be provided in the full paper. In the model of appendix D, we found that the reduction's quality decreases as the size l of the vectors of plaintexts in the challenge step increases (a similar phenomenon is encountered when proving the chosen-plaintext security of the multi-recipient randomness re-using adaptation of the basic version of the Boneh-Franklin IBE).

This application of SOK-IBSC is impossible with any other known provably secure IBSC or IBSE ([42],[39],[19],[20],etc.) since all of these schemes are built on top of signatures for which signing two different messages with a same coin exposes the private key. Except an identity based signature recently proposed in [44] (that still has no security proof), this feature is only met in SOK-IBS among all existing IBS. This may be one of the properties that explain its improved reductions.

5 Conclusions

In this paper, we showed that optimal security reductions are also achievable for identity based signatures. We showed that the variant of the Sakai-Ogishi-Kasahara IBS proposed by Bellare et al. ([6]) has a sub-optimal reduction from the Diffie-Hellman problem. Our proof does not make use of the forking lemma ([52],[53]) and bypasses the BNN framework ([6]) to directly achieve a reduction from scratch.

Interestingly, Bellare et al.'s modified SOK-IBS, that can be viewed as derived from the Gentry-Silverberg second hierarchical scheme, has an efficiency comparable to other existing IBS proposals such as those described in [17] or [34] and much stronger security guarantees under the Diffie-Hellman assumption for similar security parameters. It is furthermore shown to be as secure as the one more Diffie-Hellman problem. As far as we know, the latter reduction is the first optimal one for an identity based signature. We have to mention that another IBS can be shown to be sub-optimally related to a commonly assumed hard problem. Indeed, the technique proposed by Katz and Wang ([36]) to achieve a sub-optimal reduction for the Fiat-Shamir signature ([29]) under the quadratic residuosity assumption can be extended to a modification of the identity based Fiat-Shamir scheme described in [48] (the details for this will be given in the full paper). We think that, beside their interest for identity based cryptography, these results can also find application to key evolving signatures that were shown to be somewhat related to IBS schemes ([25],[57]).

As an application of our results, we showed that SOK-IBS can be turned into an identity based signature encryption (IBSE) similar to Boyen's one ([15]) but providing much better security guarantees or into a strongly secure monolithic identity based signcryption scheme (IBSC) depicted in appendix C. The obtained reductions are not optimal for SOK-IBSE: recall that we still have a loss of q_E in our bound, where q_E is a bound on the number of corrupted identities. Nevertheless, we think it is a real breakthrough in terms of provable security for identity based cryptosystems. Finally, one of our scheme's advantage over the first IBSE is that it can be scaled up into a multi-authority signature encryption protocol.

References

1. J.-H. An, Y. Dodis, and T. Rabin. On the security of joint signature and encryption. In *Advances in Cryptology - Eurocrypt'02*, LNCS 2332, pp. 83–107. Springer, 2002.
2. J. Baek and Y. Zheng. Identity-Based Threshold Decryption. proceedings of PKC'04, LNCS 2947, pp. 262-276, Springer, 2004.
3. P.-S.-L.-M. Barreto, *The Pairing Based Crypto Lounge*, web page located at http://planeta.terra.com.br/informatica/paulobarreto/pblounge.html
4. M. Bellare, A. Boldyreva and A. Palacio, *An Uninstantiable Random-Oracle-Model Scheme for a Hybrid-Encryption Problem*. To appear In *Advances in Cryptology - Eurocrypt'04*, LNCS series, Springer, 2004.
5. M. Bellare, A. Boldyreva and J. Staddon, *Randomness Re-use in Multi-recipient Encryption Schemes*, In *Proceedings of PKC'03*, LNCS 2567, Springer, pp. 85-99, 2003.
6. M. Bellare, C. Namprempre and G. Neven *Security Proofs for Identity-Based Identification and Signature Schemes*, to appear In *Advances in Cryptology - Eurocrypt'04*, LNCS series, Springer, 2004.
7. M. Bellare, C. Namprempre, D. Pointcheval and M. Semanko, *The One-More-RSA-Inversion Problems and the Security of Chaum's Blind Signature Scheme*, Journal of Cryptology, Volume 16 - Number 3. Pages 185–215, Springer-Verlag, 2003.
8. M. Bellare, P. Rogaway, *Random oracles are practical: A paradigm for designing efficient protocols*, Proc. of the 1^{st} ACM Conference on Computer and Communications Security, pp. 62-73, 1993.
9. M. Bellare and P. Rogaway. The Exact Security of Digital Signatures - How to Sign with RSA and Rabin. In *Advances in Cryptology - Eurocrypt'96*, LNCS 1070, pp. 399–416. Springer, 1996.
10. A. Boldyreva. Efficient threshold signature, multisignature and blind signature schemes based on the Gap-Diffie-Hellman-group signature scheme. In *Proceedings of PKC'03*, LNCS 2567, Springer, pp. 31-46, 2003.
11. D. Boneh and X. Boyen. Short Signatures Without Random Oracles. To appear In *Advances in Cryptology - Eurocrypt'04*, LNCS series, Springer, 2004.
12. D. Boneh and X. Boyen. Efficient Selective-ID Secure Identity Based Encryption Without Random Oracles. To appear In *Advances in Cryptology - Eurocrypt'04*, LNCS series, Springer, 2004.
13. D. Boneh and M. Franklin. Identity Based Encryption From the Weil Pairing. In *Advances in Cryptology - Proceedings of Crypto'01*, LNCS 2139, pp. 213–229. Springer, 2001.
14. D. Boneh, B. Lynn, and H. Shacham. Short signatures from the Weil pairing. In *Advances in Cryptology - Proceedings of Asiacrypt'01*, LNCS 2248, pp. 514–532. Springer, 2001.
15. X. Boyen. Multipurpose identity-based signcryption: A swiss army knife for identity-based cryptography. In *Advances in Cryptology (CRYPTO '03)*, LNCS 2729, pp. 382–398. Springer, 2003.
16. R. Canetti, O. Goldreich and, S. Halevi, *The Random Oracle Methodology, Revisited*, proceeding of STOC'98, pp. 209-218, ACM Press, 1998.
17. J.C. Cha, J.H. Cheon, *An Identity-Based Signature from Gap Diffie-Hellman Groups*, proceedings of PKC 2003. Springer-Verlag, LNCS 2567, pp. 18-30, Springer, 2003.
18. D. Chaum, *Blind signatures for untraceable payments*, Advances in Cryptology - Crypto'82, LNCS, pp. 199-204, Springer 1982.
19. L. Chen, J. Malone-Lee, *Improved Identity-Based Signcryption*, available at http://eprint.iacr.org/2004/114/.
20. S.S.M. Chow, S.M. Yiu, L.C.K. Hui, and K.P. Chow, *Efficient Forward and Provably Secure ID-Based Signcryption Scheme with Public Verifiability and Public Ciphertext Authenticity*. In Proceedings of the 6th Annual International Conference on Information Security and Cryptology (ICISC 2003), LNCS 2971, Springer, 2004.
21. C. Cocks, *An Identity Based Encryption Scheme Based on Quadratic Residues*, In *8th IMA International Conference on Cryptography and coding*, LNCS 2260, Springer-Verlag, pp. 360-363, 2001.
22. J.-S. Coron. On the Exact Security of Full Domain Hash. In *Advances in Cryptology - Crypto'00*, LNCS 1880, pp. 229–235, 2000.
23. J.-S. Coron. Optimal Security Proofs for PSS and Other Signature Schemes. In *Advances in Cryptology - Eurocrypt'02*, LNCS 2332, pp. 272–287. Springer, 2002.
24. R. Cramer and V. Shoup. A Practical Public Key Cryptosystem Provably Secure against Adaptive Chosen Ciphertext Attack. In *Advances in Cryptology - Crypto'98*, LNCS 1462, pp. 13–25. Springer, 1998.
25. Y. Dodis, J. Katz, S. Xu and M. Yung, *Strong Key-Insulated Signature Schemes*. In *Proceeding of PKC'03*, LNCS 2567, pp. 130-144, 2003.
26. Y. Dodis, M.-J. Freedman, and S. Walfish. Parallel Signcryption with OAEP, PSS-R and other Feistel Paddings. 2003. Available at http://eprint.iacr.org/2003/043/.

27. Y. Dodis, M. Yung, *Exposure-Resilience for Free: the Case of Hierarchical ID-based Encryption* , IEEE International Security In Storage Workshop (SISW'02), December 2002.
28. T. El Gamal, *A Public Key Cryptosystem and Signature Scheme Based on Discrete Logarithms*, IEEE Trans. on Information Theory, vol. 31, 1985.
29. A. Fiat, A. Shamir, *How to Prove Yourself: Practical Solutions to Identification and Signature Problems* , Advances in Cryptology - Crypto'86, LNCS 0263, Springer, pp. 186-194, 1986.
30. C. Gentry, A. Silverberg, *Hierarchical ID-based cryptography*, Advances in Cryptology - Asiacrypt'02, LNCS 2501, Springer-Verlag, pp. 548-566, 2002.
31. L. Guillou, J-J. Quisquater, *A "Paradoxical" Identity-Based Signature Scheme Resulting From Zero-Knowledge* , Advances in Cryptology - Crypto'88, LNCS 0403, Springer, pp. 216-231, 1988.
32. E.-J. Goh and S. Jarecki. A signature scheme as secure as the diffie-hellman problem. In *Advances in Cryptology - Eurocrypt'03*, LNCS 2656, pp. 401–415. Springer, 2003.
33. S. Goldwasser, S. Micali and R. Riverst. A digital signature scheme secure against adaptive chosen message attacks. SIAM J. of Computing 17(2). pp. 281-308, 1988.
34. F. Hess, *Efficient identity based signature schemes based on pairings*, proceedings of SAC'02. LNCS 2595, pp. 310-324, Springer-Verlag, 2002.
35. A. Joux and K. Nguyen. Separating Decision Diffie-Hellman from Diffie-Hellman in cryptographic groups. In *Journal of Cryptology*, volume 16-Number 4, pp. 239–247. Springer, 2003.
36. J. Katz and N. Wang. Efficiency improvements for signature schemes with tight security reductions, Proceedings of the 10^{th} ACM Conference on Computer and Communications Security, pp. 155–164, 2003.
37. K. Kurosawa, *Multi-Recipient Public Key Encryption with Shortened Ciphertext*, proceedings of PKC'02, LNCS 2274, pp. 48-63, Springer, 2002.
38. K. Kurosawa and S.-H. Heng, *From Digital Signature to ID-based Identification/Signature*, proceedings of PKC'04, LNCS 2947, pp. 248-261, Springer, 2004.
39. B. Libert and J.-J. Quisquater, *New identity based signcryption schemes from pairings*, available at http://eprint.iacr.org/2003/023.
40. B. Libert and J.-J. Quisquater, *Identity Based Undeniable Signatures*, Topics in Cryptology CT-RSA'04, LNCS 2964, pp. 112-125, Springer, 2004.
41. B. Libert and J.-J. Quisquater, *Efficient Signcryption with Key Privacy from Gap Diffie-Hellman Groups*, proceedings of PKC'04, LNCS 2947, pp. 187-200, Springer, 2004.
42. J. Malone-Lee. *Identity Based Signcryption*, available at http://eprint.iacr.org/2002/098/.
43. J. Malone-Lee, D. Pointcheval, N. Smart and J. Stern, *Flaws in Applying Proof Methodologies to Signature Schemes*, Advances in Cryptology - Crypto'02, LNCS 2442, pp. 93-110, Springer, 2002.
44. N. McCullagh, P.-S.-L.-M. Barreto, *Efficient and Forward-Secure Identity-Based Signcryption*, available at http://eprint.iacr.org/2004/117/.
45. A.-J. Menezes. *Elliptic curve public key cryptosystems.* Kluwer Academic Publishers, 1995.
46. S. Micali and L. Reyzin, *Improving the Exact Security of Digital Signature Schemes.* In *Journal of Cryptology*, Volume 15, Number 1, Winter 2002
47. D. Nalla, K.C. Reddy, *Signcryption scheme for Identity-based Cryptosystems*, available at http://eprint.iacr.org/2003/066/.
48. G. Neven, *Provably Secure Identity-Based Identification Schemes and Transitive Signatures*, PhD thesis, K.U. Leuven, Department Computerwetenschappen, 2004.
49. T. Okamoto and D. Pointcheval. The Gap-Problems: A New Class of Problems for the Security of Cryptographic Schemes. In *Proceedings of PKC'01*, LNCS 1992. Springer, 2001.
50. K.G. Paterson, *ID-based signatures from pairings on elliptic curves*, available at http://eprint.iacr.org/2002/004/.
51. D. Pointcheval. *Practical Security in Public-Key Cryptography.* In Proceedings of the 4th International Conference on Information Security and Cryptology (ICISC '01), LNCS 2288, pp. 1–17. Springer, 2002.
52. D. Pointcheval and J. Stern. Security proofs for signature schemes. In *Advances in Cryptology - Eurocrypt'96*, LNCS 1992, pp. 387–398. Springer, 1996.
53. D. Pointcheval and J. Stern. Security arguments for digital signatures and blind signatures. In *Journal of Cryptology*, volume 13-Number 3, pp. 361–396. Springer, 2000.
54. R. Sakai, K. Ohgishi, M. Kasahara, *Cryptosystems based on pairing*, In The 2000 Sympoium on Cryptography and Information Security, Okinawa, Japan, January 2000.
55. A. Shamir, *Identity Based Cryptosystems and Signature Schemes*, Advances in Cryptology - Crypto' 84, LNCS 196, pp. 47-53, Springer, 1984.
56. X. Yi, *An identity-based signature scheme from the Weil pairing.* IEEE Communications Letters, 7(2):76–78, 2003.
57. D.-H. Yum and P.-J. Lee, *Efficient Key Updating Signature Schemes Based on IBS.* In *9th IMA International Conference on Cryptography and coding*, LNCS 2898, pp. 167–182, Springer, 2003.
58. Y. Zheng. Digital signcryption or how to achieve cost (signature & encryption) << cost(signature) + cost(encryption). In *Advances in Cryptology - Crypto'97*, LNCS 1294, pp. 165–179, Springer, 1997.

Appendix

A. Formal security model for identity based signature/encryption (IBSE)

In [15], Boyen gives formal definitions for the security of identity based signature/encryption schemes. For the chosen-ciphertext security of his scheme, he considers "insider attacks" (i.e. attacks led by attackers that learn the private key of the challenge ciphertext's sender at some moment of the game). We recall this definition as well as the one of "encrypted signature non-repudiation". These definitions can be skipped by the reader who is familiar with the model of [15].

Definition 4. *An identity based signature/encryption scheme (IBSE) is said to be* ***adaptively chosen-ciphertext secure*** *(IND-IBSE-CCA) if no PPT adversary has a non-negligeable advantage in the following game.*

1. *The challenger runs the Setup algorithm on input of a security parameter k and sends the domain-wide parameters to the cca-adversary $\mathcal{A}$.*
2. *In a find stage, $\mathcal{A}$ starts probing query the following oracles:*
   - *Key extraction oracle: given an identity ID, it returns the extracted private key associated to it.*
   - *Signature/encryption oracle: given a sender and a receiver's identities ID_S, ID_R and a plaintext M, it returns an encryption under the receiver's identity ID_R of the message M signed in the name of the sender ID_S.*
   - *Decryption/verification oracle: given a receiver's identity ID_R and a ciphertext σ, it generates the private key $d_{ID_R} = \texttt{Keygen}(ID_R)$ and returns either a triple (M, s, ID_S) made of a valid message-signature pair (M, s) for the sender's identity ID_S or the $\perp$ symbol if σ does not decrypt into a valid message-signature pair.*
3. *After the find stage, $\mathcal{A}$ produces two equal-length plaintexts $M_0, M_1 \in \mathcal{M}$ and two identities ID_S^* and ID_R^* on which it wishes to be challenged. It may not have corrupted the private key corresponding to ID_R^* in the find stage.*
4. *The challenger picks a bit $b \xleftarrow{R} \{0,1\}$ and computes $C = \texttt{Sign/Encrypt}(M_b, d_{ID_S^*}, ID_R^*)$ which is sent to $\mathcal{A}$.*
5. *In the guess stage, $\mathcal{A}$ asks new queries as in the find stage. This time, it may not issue a key extraction request on ID_R^* and it cannot submit C to the decryption/verification oracle for the target identity ID_R^*.*
6. *Finally, $\mathcal{A}$ outputs a bit b' and wins if $b' = b$.*

$\mathcal{A}$'s advantage is defined as $Adv(\mathcal{A}) := 2 \times Pr[b' = b] - 1$.

In the following definition, the forger is allowed to corrupt the receiver's identity ID_R. The motivation for this is to really achieve the non-repudiation property: a dishonest recipient cannot send a ciphertext to himself on behalf of Alice and to try and convince a third party that Alice was the author of the signed-message embedded in it.

Definition 5. *An identity based signature/encryption scheme (IBSE) is said to be* ***strongly existentially signature-unforgeable*** *against adaptive chosen messages and ciphertexts attacks (ESUF-IBSE-CMCA) if no PPT adversary can succeed in the following game with a non-negligible advantage:*

1. *the challenger runs the Setup algorithm on input k and gives the system-wide public key to the adversary $\mathcal{F}$.*
2. *$\mathcal{F}$ starts querying the oracles as in the previous definition.*
3. *Finally, $\mathcal{F}$ outputs a pair (σ^*, ID_R^*) and wins the game if $\texttt{Decrypt/verify}(\sigma^*, d^*_{ID_R})$ (i.e. the result of the decryption/verification oracle on the ciphertext σ^* under the private key associated to ID_R^*) is a valid message-signature (M^*, s^*) pair for an uncorrupted sender's identity ID_S^* such that no signature/encrytion query involved M^*, ID_S^* and some receiver ID'_R (possibly different from ID_R^*) and resulted in a ciphertext σ' whose decryption under the private key $d_{ID'_R}$ is the alleged forgery (M^*, s^*, ID_S^*).*

The adversary's advantage is its probability of victory.

This notion is called "strong unforgeability" because the forger is allowed to produce a ciphertext decrypting into a triple (M^*, s^*, ID_S^*) such that a signature/encryption query involved M^*, ID_S^* and $ID_R' \neq ID_R^*$ but resulted into a ciphertext σ' such that the oracle call $\texttt{Decrypt/verify}(\sigma', d_{ID_R'})$ produces a triple $(M^*, s, ID_S) \neq (M^*, s^*, ID_S^*)$.

B. Security proofs for SOK-IBSE

B.1 Proof of theorem 3

Let (P, aP, bP, cP) be a random instance of the GBDH problem taken as input by a simulator $\mathcal{B}$ and let $\mathcal{O}_P^{DBDH}(.)$ be an oracle deciding within a short and constant time whether a given tuple $(a'P, b'P, c'P, h') \in {\mathbb{G}_1}^3 \times \mathbb{G}_2$ is a correct BDH one or not. We show how $\mathcal{B}$ uses a chosen-ciphertext attacker $\mathcal{A}$ against SOK-IBSE to solve this problem. To achieve this, it first pre-computes $\alpha = \hat{e}(cP, bP)$ and $\beta = \hat{e}(cP, P)$ and initializes $\mathcal{A}$ under the system-wide public key $P_{pub} = cP \in \mathbb{G}_1$. In order to coherently answer $\mathcal{A}$'s requests, $\mathcal{B}$ maintains lists L_i to keep track of answers given to queries on random oracles H_i (for $i = 1, \ldots, 4$). Without loss of generality, we may assume that any key extraction query, signature/encryption or decryption/verification query involving an identity is preceded by a H_1 query on the same identity. We also assume that $\mathcal{A}$ never submits to the decryption/verification oracle a ciphertext-identity pair obtained from the signature/encryption oracle. Indeed, since the decryption/verification algorithm is deterministic, $\mathcal{B}$ would be able to answer such a request by simply keeping in memory the plaintexts and the random coins used when simulating any signature/encryption query.

In the simulation, the H_3 oracle needs to be simulated in a somewhat special fashion. Namely, an auxiliary list L_3' is also used and the decision oracle $\mathcal{O}_P^{DBDH}(.)$ is called upon each H_3 query. In more details,

- H_1 queries are dealt with as in the proof of theorem 1. Namely, when an identity ID is submitted to the H_1 oracle, $\mathcal{B}$ flips a coin T taking the value 0 with probability ξ and the value 1 with probability $1-\xi$ where $\xi = q_E/(q_E+1)$. $\mathcal{B}$ then picks $u \stackrel{R}{\leftarrow} \mathbb{Z}_q^*$ before returning $uP \in \mathbb{G}_1$ if $T = 0$ and $u(bP) \in \mathbb{G}_1$ otherwise. The entry (ID, u, T) is then inserted into the list L_1.
- H_2 queries on input $(ID_{A,i}, M_i, U_i)$: $\mathcal{B}$ returns a randomly sampled element $h_{2,i} \stackrel{R}{\leftarrow} \mathbb{G}_1$. The whole tuple $(ID_{A,i}, M_i, U_i, h_{2,i})$ is then inserted into L_2. $\mathcal{B}$ must of course answer the previously stored $h_{2,i}$ if an entry of this form was already present in L_2.
- H_3 queries on an input $(X_i, Q_i, g_i) \in {\mathbb{G}_1}^2 \times \mathbb{G}_2$: if the list L_3 already contains an entry $(X_i, Q_i, g_i, h_{3,i}, .)$, then $\mathcal{B}$ returns $h_{3,i}$. Otherwise, $\mathcal{B}$ checks whether L_3' contains an entry of the form $(X_i, Q_i, ., h_{3,i})$ such that, when queried on $(P, P_{pub}, X_i, Q_i, g_i)$, the oracle $\mathcal{O}_P^{DBDH}(.)$ returns 1. If it does, $\mathcal{B}$ returns $h_{3,i}$ and stores the pair $(X_i, Q_i, g_i, h_{3,i}, 1)$ in L_3. If no entry of the above form is found in L_3', a random $h_{3,i} \stackrel{R}{\leftarrow} \{0,1\}^\ell$ is returned to $\mathcal{A}$ and the information $(X_i, Q_i, g_i, h_{3,i}, t_i)$, where $t_i = \mathcal{O}_P^{DBDH}(P, P_{pub}, X_i, Q_i, g_i)$, is stored in L_3.
- H_4 and H_5 queries are handled in the usual fashion by returning a randomly sampled element in the appropriate range ($\{0,1\}^\lambda$ or $\mathbb{Z}_q$) and updating the corresponding list.

The other kinds of queries are tackled with as follows:

- Key extraction query: when $\mathcal{A}$ asks for the private key associated to an identity ID, $\mathcal{B}$ looks at the corresponding entry (ID, u, T) that must exist in L_1. If $T = 1$, it aborts and outputs "failure". Otherwise, it returns $uP_{pub} = u(cP) \in \mathbb{G}_1$.
- Signature/encryption query: for a plaintext M and a pair of identities (ID_A, ID_B) submitted to the signature/encryption oracle, $\mathcal{B}$ first recovers $Q_{ID_A} = H_1(ID_A)$ and $Q_{ID_B} = H_1(ID_B)$ from L_1. It then picks $t, \nu \stackrel{R}{\leftarrow} \mathbb{Z}_q$, computes $U = \nu P_{pub}$, $V = tP_{pub}$ before setting the hash value $H_2(ID_A, M, U)$ to $\nu^{-1}(tP - Q_{ID_A})$. It aborts and outputs "failure" if H_2 is

already defined at the point (ID_A, M, U). Otherwise, it runs the H_5 simulation algorithm to obtain $x = H_5(ID_A, ID_B, \tau) \in \mathbb{Z}_q$, for a random $\tau \xleftarrow{R} \{0,1\}^\delta$, and then calculates $X = xU \in \mathbb{G}_1$. At this point, depending on whether L_3 contains an entry $(X, Q_{ID_B}, g, h_3, 1)$, two cases can be distinguished: if it does, $\mathcal{B}$ sets $W = V \oplus h_3$ for the corresponding h_3 and returns the ciphertext $\langle X, W, \mathcal{E}_{H_4(V)}(M||ID_A||\tau)\rangle$ where the value of H_4 is obtained through simulation. If no entry of the aforementioned form exists in L_3, $\mathcal{B}$ picks a random string $h_3 \xleftarrow{R} \{0,1\}^\ell$ and stores the tuple $(X, Q_{ID_B}, ., h_3)$ in L_3' before defining $W = V \oplus h_3 \in \{0,1\}^\ell$, $\kappa = H_4(V) \in \{0,1\}^\lambda$ (obtained by simulation) and $Z = \mathcal{E}_\kappa(M||ID_A||\tau)$. The ciphertext $\langle X, W, Z\rangle$ is finally returned to $\mathcal{A}$.

- Decryption/verification query: for a ciphertext $\sigma = \langle X, W, Z\rangle$ and a recipient's identity ID_B submitted to the decryption/verification oracle, $\mathcal{B}$ starts by recovering the corresponding Q_{ID_B} from the list L_1. With overwhelming probability, a unique entry $(X, Q_{ID_B}, g, h_3, 1)$ must exist in L_3, for some $h_3 \in \{0,1\}^\ell$, if σ was correctly formed. $\mathcal{B}$ then looks into L_4 for the unique entry (V, h_4), for some string $V \in \{0,1\}^\ell$ representing a point on the curve, such that $h_3 = V \oplus W \in \{0,1\}^\ell$ (h_3 being the output of $H_3(X, Q_{ID_B}, g)$ where $\mathcal{O}_P^{DBDH}(P, P_{pub}, X, Q_{ID_B}, g) = 1$). The only possible pair of such entries in L_3 and L_4 is further examined as follows: if $(M||ID_A||\tau) = \mathcal{D}_{h_4}(Z)$ is such that the hash value $h_2 = H_2(M, ID_A, U) \in \mathbb{G}_1$ satisfies $\hat{e}(P, V) = \hat{e}(P_{pub}, Q_{ID_A})\hat{e}(U, h_2)$, where $U = x^{-1}X \in \mathbb{G}_1$ for an $x = H_5(ID_A, ID_B, \tau)$ obtained by H_5 simulation (note that this condition can be checked by only computing two pairings and one exponentiation since $\hat{e}(P_{pub}, Q_{ID_A})$ can be obtained by raising α or β to a known power), $\mathcal{B}$ returns the message-identity pair $(M||ID_A)$ together with the appended signature (U, V). The $\perp$ symbol is returned if the appropriate H_3 query was not asked or if no entry of L_4 satisfies the aforementioned conditions (i.e. $V \oplus W$ matches the right output of H_3 and $\hat{e}(P, V) = \hat{e}(P_{pub}, Q_{ID_A})\hat{e}(U, h_2)$ with $h_2 = H_2(\mathcal{D}_{h_4}(Z), ID_B, U)$).

Once $\mathcal{A}$ decides that the find stage is over, it outputs a pair (M_0, M_1) of plaintexts that will be ignored by the simulator $\mathcal{B}$ and an uncorrupted recipient's identity ID_B. If $T = 0$ for the corresponding entry (ID_B, u, T) of L_1, then $\mathcal{B}$ fails. Otherwise, it sends the challenge $\sigma = \langle aP, W, Z\rangle$, where $W \xleftarrow{R} \{0,1\}^\ell$ and $Z \xleftarrow{R} \{0,1\}^{n_0+n_1+\delta}$ are random strings, to $\mathcal{A}$. The latter will be unable to realize that σ is not an actual ciphertext unless it asks the hash value $H_3(aP, u(bP), \hat{e}(P,P)^{(abc)u})$ during the simulation. This would provide $\mathcal{B}$ with the solution to the GBDH problem since it knows u and the $\mathcal{O}_P^{DBDH}$ oracle can decide whether a candidate $(P, aP, u(bP), cP, g)$ is a valid BDH tuple or not.

During the guess stage, $\mathcal{A}$ keeps on issuing new queries that are handled as in the find stage but, this time, for any query $(aP, u(bP), g)$ with $g \in \mathbb{G}_2$ made to the H_3 oracle, $\mathcal{B}$ submits $(P, aP, u(bP), cP, g)$ to the $\mathcal{O}_P^{DBDH}(.)$ oracle. If the latter returns 1, $\mathcal{B}$ then stops and outputs $g^{1/u} \in \mathbb{G}_2$ as a result. Since $\mathcal{A}$ is assumed to have a non-negligible advantage ϵ in the IND-IDSC-CCA game, one can easily show (as done in many papers in the literature, see [13] for example) that it must ask the hash value $H_3(aP, u(bP), \hat{e}(P,P)^{(abc)u})$ with a probability at least ϵ during the game provided the simulation is perfect.

The simulation can be readily checked to only be non perfect if a ciphertext is wrongly rejected by the simulator $\mathcal{B}$ or if the latter reaches a state of "failure" during the game. The probability for the second undesirable event to occur because of a 'bad' key extraction query or because of a 'bad' chosen target identity ID_B is bounded by $1/e(q_E + 1)$. On the other hand, $\mathcal{B}$'s probability to fail in answering a signature/encryption query because of a hash collison is not greater than $q_{SE}(q_{SE} + q_{H_2})/2^k$.

Finally, the probability to incorrectly reject a ciphertext at some moment of the simulation is bounded by $q_{DV}(q_{H_4}/2^\ell + 1/2^k)$. Indeed, for a given decryption/verification query on a pair $(\sigma = \langle X, W, Z\rangle, ID_B)$, the $\perp$ symbol is returned if no hash query $H_3(X, Q_{ID}, g)$ was made for the only valid BDH tuple $(P, P_{pub}, X, Q_{ID_B}, g)$. In this case, if this query is subsequently asked by $\mathcal{B}$, there is a probability of $q_{H_4}/2^\ell$ that the answer hits $W \oplus V_i$ $(0 \leq i \leq q_{H_4})$ for some entry $(V_i, h_{4,i})$ of L_4. On the other hand, if the appropriate $H_3(X, Q_{ID_B}, g) = W \oplus V$ was asked for some (V, h_4) in L_4 but the corresponding $H_2(ID_A, M, U)$, with $(M||ID_A||\tau) = D_{h_4}(Z)$ and

$U = x^{-1}X$ for $x = H_5(ID_A, ID_B, \tau)$, was not asked by $\mathcal{B}$, the probability for this query to be subsequently answered in the only way that renders σ valid from $\mathcal{A}$'s view is at most $1/2^k$ (indeed, there is a single value of $h_2 \in \mathbb{G}_1$ such that $\hat{e}(P, V)\hat{e}(P_{pub}, Q_{ID_A})^{-1} = \hat{e}(U, h_2)$).

The bound on $\mathcal{B}$'s running time easily comes by noting that each H_1 or key extraction query requires one scalar multiplication in $\mathbb{G}_1$ while answering a signature/encryption query implies to perform two multiplications and one computation of the type $aP + bQ \in \mathbb{G}_1$. Finally, decryption/verification queries all require two pairing evaluations as well as an exponentiation in $\mathbb{G}_2$ and every H_3 query implies a query to the $\mathcal{O}_P^{DBDH}$ oracle that takes a constant time.

□

B.2 Proof of theorem 4

Let (P, aP, bP, cP) be a random instance of the GBDH problem given as input to $\mathcal{B}$ and let $\mathcal{O}_P^{DBDH}(.)$ be an oracle solving DBDH instances in short and constant time. The simulator $\mathcal{B}$ runs the forger $\mathcal{F}$ on the system-wide key $P_{pub} = cP \in \mathbb{G}_1$ and answers all oracles requests exactly as in the proof of IND-IBSE-CCA security: the $\mathcal{O}_P^{DBDH}(.)$ oracle is again necessary to handle decryption/verification and H_3 queries (this is why we have a reduction from the GBDH problem instead of the CDH or the one more CDH ones).

At the end of the game, the forger $\mathcal{F}$ halts and comes up with a ciphertext $\sigma^* = \langle X, W, Z\rangle$ and a possibly corrupted receiver's identity ID_B^*. If ID_B^* is an identity for which $\mathcal{B}$ defined $H_1(ID_B^*)$ as a known power of P, then the private key $d_{ID_B^*}$ is computable for $\mathcal{B}$ that can then open σ^* into a valid message-signature pair (M^*, s^*), with $s^* = \langle U, V\rangle$, for some uncorrupted sender's identity ID_A^*. If $H_1(ID_B^*)$ was rather defined as a power of bP, $\mathcal{B}$ can extract the embedded message-signature pair (M^*, s^*) as well as the sender's identity ID_A^* by submitting (σ^*, ID_B^*) to its own decryption/verification oracle. As we will see, it can successfully extract the clear triple (M^*, s^*, ID_A^*) with overwhelming probability (i.e. if the appropriate H_2, H_3 and H_4 queries were made by $\mathcal{F}$ regarding the intended forgery). If the hash value $Q_{ID_A^*}$ was defined to be $u^*(bP) \in \mathbb{G}_1$ for a known $u^* \in \mathbb{Z}_q^*$ in L_1, then $\mathcal{B}$ can extract the solution abP to the CDH problem (P, aP, bP) as in the proofs of strong unforgeability of SOK-IBS (and then obtain the solution to the easier initial GBDH problem). If the hash value $Q_{ID_A^*}$ was rather defined to be $uP \in \mathbb{G}_1$, then $\mathcal{B}$ ends and fails because the produced forgery is useless.

Let us now assess $\mathcal{F}$'s probability to successfully output an encrypted existential forgery for an uncorrupted identity without having asked the appropriate queries to H_2, H_3 and H_4. First, its probability to create a pair $\langle U^*, V^*\rangle$ such that $\hat{e}(P, V^*) = \hat{e}(P_{pub}, Q_{ID_A^*})\hat{e}(U^*, H^*)$ with $H^* = H_2(ID_A^*, M^*, U^*)$ is at most $1/2^k$ if it never asks the latter hash value. On the other hand, given a valid fake signature $\langle U^*, V^*\rangle$ on M^* for an identity ID_A^*, its probability to properly encrypt it into a valid ciphertext $\langle X^*, W^*, Z^*\rangle$ intended to a recipient ID_B^* without querying H_3 on the input $(X^*, Q_{ID_B^*}, g)$ such that $(P, P_{pub}, X^*, Q_{ID_R^*}, g)$ is a valid BDH tuple is not greater than $1/2^\ell$. Finally, if the appropriate H_2 and H_3 queries are made but no entry of L_4 allows the decryption/verification simulator to complete the extraction, $\mathcal{F}$ cannot turn a given fake signature $\langle U^*, V^*\rangle$ for a pair (M^*, ID_A^*) into a correctly encrypted message-signature pair $\langle X^*, W^*, Z^*\rangle$ with a probability greater than $2^{n_0+n_1-2\lambda}$.

□

C. The monolithic SOK-IBSC scheme

The protocol consists of the four algorithms on figure 3. As done in [15] and [41], we assume the receiver of a ciphertext has no a priori knowledge about the sender's identity that is recovered during the designcryption operation.

Setup: given a security parameter k, the PKG chooses groups $\mathbb{G}_1$ and $\mathbb{G}_2$ of prime order $q > 2^k$, a generator P of $\mathbb{G}_1$, a randomly chosen master key $s \in \mathbb{Z}_q^*$ and the associated public key $P_{pub} = sP$. It also chooses a symmetric encryption scheme $(\mathcal{E}, \mathcal{D})$ and cryptographic hash functions $H_1, H_2 : \{0,1\}^{n_0+2n_1+\ell} \rightarrow \mathbb{G}_1^*$, $H_3 : \mathbb{G}_1{}^2 \times \mathbb{G}_2 \rightarrow \{0,1\}^\ell$ and $H_4 : \mathbb{G}_1 \rightarrow \{0,1\}^\lambda$ where ℓ denotes the length of the representation of $\mathbb{G}_1$'s elements, λ is the length of symmetric keys for $(\mathcal{E}, \mathcal{D})$ while n_0 and n_1 are respectively the size of plaintexts and the bitlength of identifiers. The system-wide parameters are now

$$\texttt{params} = (\mathbb{G}_1, \mathbb{G}_2, \hat{e}, P, P_{pub}, \ell, n_0, n_1, \lambda, \mathcal{E}, \mathcal{D}, H_1, H_2, H_3, H_4).$$

Keygen: as in SOK-IBS, given an identity ID, the PKG computes $Q_{ID} = H_1(ID) \in \mathbb{G}_1$ and the associated private key $d_{ID} = sQ_{ID} \in \mathbb{G}_1$ that is securely sent to the user.

Signcrypt: given a message, the parameters `params`, the sender's private key d_{ID_A} and the recipient's public key $Q_{ID_B} = H_1(ID_B)$, this algorithm

1. picks $r \stackrel{R}{\leftarrow} \mathbb{Z}_q$ and computes $U = rP \in \mathbb{G}_1$ and then $H = H_2(M, ID_A, ID_B, U) \in \mathbb{G}_1$,
2. sets $V = d_{ID_A} + rH \in \mathbb{G}_1$
3. hides the signature $W = V \oplus H_3(U, Q_{ID_B}, g^r) \in \{0,1\}^\ell$ where $g = \hat{e}(P_{pub}, Q_{ID_B})$,
4. encrypts the message and the sender's identity with a hash value of V together with the other components of the ciphertext as a symmetric key: $Z = \mathcal{E}_\kappa(M||ID_A) \in \{0,1\}^{n_0+n_1}$ with $\kappa = H_4(V) \in \{0,1\}^\lambda$.

The ciphertext is $\sigma = \langle U, W, Z\rangle \in \mathbb{G}_1 \times \{0,1\}^{\ell+n_0+n_1}$

Designcrypt: upon receiving a ciphertext $\sigma = \langle U, W, Z\rangle$ and given the recipient's private key d_{ID_B}, this algorithm

1. first checks that U is a point on the curve on which $\mathbb{G}_1$ is defined and rejects σ if it is not,
2. computes $V = W \oplus H_3(U, Q_{ID_B}, g') \in \{0,1\}^\ell$ where $g' = \hat{e}(U, d_{ID_B})$ and rejects σ if V is not a point on the curve,
3. computes $(M||ID_A) = \mathcal{D}_\kappa(Z)$ where $\kappa = H_4(V) \in \{0,1\}^\lambda$ and then $Q_{ID_A} = H_1(ID_A) \in \mathbb{G}_1$.
4. The message-signature pair $(M, \langle U, V\rangle)$ is accepted for the sender's identity ID_A if and only if $\hat{e}(P, V) = \hat{e}(P_{pub}, Q_{ID_A})\hat{e}(U, H)$ where $H = H_2(M, ID_A, ID_B, U) \in \mathbb{G}_1$.

Fig. 3. The SOK-IBSC signcryption scheme

D. Security notions for randomness re-using multi-recipient IBSC

Such a scheme is made of four algorithms. The first two ones are usual setup and key generation algorithms as in any identity based cryptosystem. The third one, called `Multi-signcypt` takes as input a vector of messages $(M_1, \ldots, M_N)$ and a vector of identities $(ID_1, \ldots, ID_N)$, picks a coin r from a coin set and returns a vector of ciphertexts $(C_1, \ldots, C_N)$ that all share a common component. The last one, called `Multi-designcrypt` takes a vector of ciphertexts and a recipient's identity to return a plaintext or the distinguished symbol $\perp$.

Definition 6. *We say that a multi-recipient IBSC is secure against chosen-ciphertext attacks (IND-MRIBSC-CCA) if no PPT attacker has a non-negligible advantage in the game below.*

1. *The adversary $\mathcal{A}$ receives the domain-wide parameters from the challenger and returns two natural numbers (l, N) with $1 \leq l < N$ such that $N - l$ is the number of corrupted identities in the challenge step.*
2. *In a find stage, $\mathcal{A}$ is given access to the following oracles:*
   - *`Keygen`(.): takes as input an identity and returns the associated private key.*
   - *`Multi-signcrypt`(.): given a natural integer $N \in \mathbb{N}$, a sender's identity ID_A, a vector of message $(M_1, \ldots, M_N)$ and a vector of recipient-identities $(ID_1, \ldots, ID_N)$, it picks a coin r and outputs a vector of ciphertexts $(C_1, \ldots, C_N)$ where, for all $i = 1, \ldots, N$, $C_i =$ `Signcrypt`(r, M_i, ID_i).*

- `Multi-designcrypt(.)`: *given a natural number* $N \in \mathbb{N}$*, an index* $i \in \{1, \ldots, N\}$*, a vector of ciphertexts* $(C_1, \ldots, C_N)$ *and a recipient's identity* $ID_{B,i}$*, it returns either a signed plaintext* (M_i, s_i) *together with a sender's identity* ID_A *or the distinguished symbol* $\perp$ *to indicate that the component* C_i *was not properly formed.*

3. *Once the find stage is over,* $\mathcal{A}$ *enters in the challenge step: it produces two* $l-$*vectors of plaintexts* $\boldsymbol{M}_0 = (M_{1,0}, \ldots, M_{l,0})$ *and* $\boldsymbol{M}_1 = (M_{1,1}, \ldots, M_{1,l})$*, a (possibly corrupted) sender's identity* ID_A*, a vector of identities* $(ID_{B,1}, \ldots, ID_{B,N})$ *among which* $N - l$ *are corrupted and* l *are uncorrupted (w.l.o.g. we may assume that* $ID_{B,1}, \ldots, ID_{B,l}$ *are not corrupted) and another vector of* $N - l$ *plaintexts* $(M_{l+1}, \ldots, M_N)$*. The challenger flips a coin* $b \overset{R}{\leftarrow} \{0, 1\}$ *and sends* $\mathcal{A}$ *a vector* $\boldsymbol{C}$ *which is the output of*

$$\texttt{Multi-signcrypt}(r, ID_A, (\boldsymbol{M}_b || M_{l+1}, \ldots, M_N), (ID_{B,1}, \ldots, ID_{B,N}))$$

   for a concealed random coin r*.*
4. *In a guess stage,* $\mathcal{A}$ *keeps on probing the same oracles as in the find stage but it is forbidden to ask for the private key of any uncorrupted identity* $ID_{B,1}, \ldots, ID_{B,l}$ *and to query the* `Multi-designcrypt` *oracle on the challenge* $\boldsymbol{C}$ *for one* $ID_{B,1}, \ldots, ID_{B,l}$*.*
5. $\mathcal{A}$ *eventually outputs a bit* $d \in \{0, 1\}$ *and wins if* $d = b$*. As usual, its advantage is defined as* $Adv(\mathcal{A}) := 2 \times Pr[d = b] - 1$*.*

The definition of unforgeabilitiy in the multi-recipient setting is a very natural extension of the definition of strong unforgeability of monolithic signcryption schemes in the single recipient setting (see [41]). Informally, the adversary is given access to the same oracles as in the above definition but runs in a single stage and aims at producing a $N-$vector of ciphertexts $\mathbf{C}$ and a $N-$vector of (possibly corrupted) recipient-identities $(ID_{B,1}, \ldots, ID_{B,N})$ such that for all $i \in \{1, \ldots, N\}$ the result of $\texttt{Multi-designcrypt}(\mathbf{C}, i, ID_{B,i})$ is a valid message-signature-identity triple (M_i, s_i, ID_A) such that ID_A was not corrupted and $\mathbf{C}[i]$ was not obtained as the output of a query

$$\texttt{Multi-signcrypt}(ID_A, (M'_1, \ldots, M_i, \ldots, M'_N), (ID'_{B,1}, \ldots, ID_{B,i}, \ldots, ID'_{B,N}))$$

during the game.