Efficient and Provably-Secure Identity-Based Signatures and Signcryption from Bilinear Maps

Paulo S.L.M. Barreto[2], Benoît Libert[3,⋆],
Noel McCullagh[1,⋆⋆], and Jean-Jacques Quisquater[3]

[1] School of Computer Applications,
Dublin City University,
Ballymun, Dublin 9, Ireland
noel.mccullagh@computing.dcu.ie
[2] PCS, Escola Politécnica, Universidade de São Paulo,
Av. Prof. Luciano Gualberto, tr. 3, n. 158, s. C1-46,
BR 05508-900, São Paulo(SP), Brazil
pbarreto@larc.usp.br
[3] UCL, Microelectronics Laboratory, Crypto Group,
Place du Levant, 3, B-1348, Louvain-La-Neuve, Belgium
Telephone : +32(0)10 47.80.62, Fax : +32(0)10 47.25.98
libert,quisquater@dice.ucl.ac.be

Abstract. In this paper we describe a new identity-based signcryption (IBSC) scheme built upon bilinear maps. This scheme turns out to be more efficient than all others proposed so far. We prove its security in a formal model under recently studied computational assumptions and in the random oracle model. As a result of independent interest, we propose a new provably secure identity-based signature (IBS) scheme that is also faster than all known pairing-based IBS methods.

1 Introduction

Two fundamental services of public key cryptography are privacy and authentication. Public key encryption schemes aim at providing confidentiality whereas digital signatures must provide authentication and non-repudiation. Nowadays, noticeably, many real-world cryptographic application require those distinct goals to be simultaneously achieved. This motivated Zheng [39] to provide the cryptographer's toolbox with a novel cryptographic primitive which he called 'signcryption.' The purpose of this kind of cryptosystem is to encrypt and sign data in a single operation which has a computational cost less than that of doing both operations sequentially. Proper signcryption schemes should provide confidentiality as well as authentication and non-repudiation. As in conventional encryption schemes, recovering the plaintext from a signcrypted message must be

⋆ This author's work was supported the DGTRE's First Europe Program of the Walloon Region in Belgium.

⋆⋆ This author wishes to thank Enterprise Ireland for their support with this research under grant IF/2002/0312/N.

B. Roy (Ed.): ASIACRYPT 2005, LNCS 3788, pp. 515–532, 2005.

computationally infeasible without the recipients private key; as in conventional digital signatures, it must be computationally infeasible to create signcrypted texts without the senders private key.

Identity based cryptography has become a very fashionable area of research for the last couple of years. The concept was originally introduced in 1984 by Shamir [34] whose idea was that users within a system could use their online identifiers (combined with certain system-wide information) as their public keys. This greatly reduces the problems with key management that have hampered the mass uptake of public key cryptography on a per individual basis. While identity-based signature schemes (IBS) rapidly emerged [20,23] after 1984 (see [5] for a thorough study of them), and despite another bandwidth-consuming proposal [18], it is only in 2001 that bilinear mappings over elliptic curve were found to yield the first fully practical identity-based encryption (IBE) solution [10]. Those bilinear maps, or pairings, subsequently turned out to yield a plenty of cryptographic applications [2] among which several recent outstanding results on identity-based encryption [7,8,21,36].

Several identity-based signcryption algorithms have been proposed so far, e.g. [11,14,16,17,26,27,30,33,37]. Within this handful of results, only [11,14,16,17,26,37] consider schemes supported by formal models and security proofs in the random oracle model [6]. Among them, Chen and Malone-Lee's proposal [14] happens to yield the most efficient construction.

The main contribution of this paper is to propose a new identity-based signcryption scheme that even supersedes [14] from an efficiency point of view at the expense of a security resting on stronger assumptions. The new construction can benefit from the most efficient pairing calculation techniques for a larger variety of elliptic curves than previous schemes. Indeed, recent observations [35] pinpointed problems arising when many provably secure pairing based protocols are implemented using asymmetric pairings and ordinary curves. Our proposal avoids those problems thanks to the fact that it does not require to hash onto an elliptic curve cyclic subgroup. As a result of independent interest, we discovered a new identity-based signature that happens to be faster at verification than previously known IBS schemes.

This paper is organized as follows. Section 2 presents the basic security theoretic concepts of bilinear map groups and the hard problems underlying our proposed algorithms. We describe our identity-based signature scheme and prove its security in section 3. We propose a new identity-based signcryption scheme in section 4, and compare its efficiency to various schemes in section 5. We draw our conclusions in section 6.

2 Preliminaries

2.1 Bilinear Map Groups and Related Computational Problems

Let k be a security parameter and p be a k-bit prime number. Let us consider groups $\mathbb{G}_1$, $\mathbb{G}_2$ and $\mathbb{G}_T$ of the same prime order p and let P, Q be generators of respectively $\mathbb{G}_1$ and $\mathbb{G}_2$. We say that $(\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T)$ are *bilinear map groups* if there exists a bilinear map $e : \mathbb{G}_1 \times \mathbb{G}_2 \rightarrow \mathbb{G}_T$ satisfying the following properties:

1. Bilinearity: $\forall\ (S, T) \in \mathbb{G}_1 \times \mathbb{G}_2,\ \forall\ a, b \in \mathbb{Z},\ e(aS, bT) = e(S, T)^{ab}$.
2. Non-degeneracy: $\forall\ S \in \mathbb{G}_1,\ e(S, T) = 1$ for all $T \in \mathbb{G}_2$ iff $S = \mathcal{O}$.
3. Computability: $\forall\ (S, T) \in \mathbb{G}_1 \times \mathbb{G}_2,\ \ e(S, T)$ is efficiently computable.
4. There exists an efficient, publicly computable (but not necessarily invertible) isomorphism $\psi : \mathbb{G}_2 \rightarrow \mathbb{G}_1$ such that $\psi(Q) = P$.

Such bilinear map groups are known to be instantiable with ordinary elliptic curves such as those suggested in [29] or [4]. In this case, the trace map can be used as an efficient isomorphism ψ as long as $\mathbb{G}_2$ is properly chosen [35]. With supersingular curves, symmetric pairings (i.e. $\mathbb{G}_1 = \mathbb{G}_2$) can be obtained and ψ is the identity.

The computational assumptions for the security of our schemes were previously formalized by Boneh and Boyen [9,7] and are recalled in the following definition.

Definition 1 ([9,7]). *Let us consider bilinear map groups $(\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T)$ and generators $P \in \mathbb{G}_1$ and $Q \in \mathbb{G}_2$.*

> *The **q-Strong Diffie-Hellman** problem (q-SDHP) in the groups $(\mathbb{G}_1, \mathbb{G}_2)$ consists in, given a $(q+2)$-tuple $(P, Q, \alpha Q, \alpha^2 Q, \ldots, \alpha^q Q)$ as input, finding a pair $\left(c, \frac{1}{c+\alpha} P\right)$ with $c \in \mathbb{Z}_p^*$.*
>
> *The **q-Bilinear Diffie-Hellman Inversion** problem (q-BDHIP) in the groups $(\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T)$ consists in, given $(P, Q, \alpha Q, \alpha^2 Q, \ldots, \alpha^q Q)$, computing $e(P, Q)^{1/\alpha} \in \mathbb{G}_T$.*

3 A New Identity-Based Signature

We here present a new identity-based signature that is significantly more efficient all known pairing based IBS schemes as its verification algorithm requires a single pairing calculation. This efficiency gain is obtained at the expense of letting the security rely on a stronger assumption than other provably secure pairing based IBS [12,15,24].

Setup: given a security parameter k, the PKG chooses bilinear map groups $(\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T)$ of prime order $p > 2^k$ and generators $Q \in \mathbb{G}_2$, $P = \psi(Q) \in \mathbb{G}_1$, $g = e(P, Q)$. It then selects a master key $s \xleftarrow{R} \mathbb{Z}_p^*$, a system-wide public key $Q_{pub} = sQ \in \mathbb{G}_2$ and hash functions $H_1 : \{0,1\}^* \rightarrow \mathbb{Z}_p^*$, $H_2 : \{0,1\}^* \times \mathbb{G}_T \rightarrow \mathbb{Z}_p^*$. The public parameters are

$$\mathsf{params} := \{\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, P, Q, g, Q_{pub}, e, \psi, H_1, H_2\}$$

Keygen: for an identity ID, the private key is $S_{\mathsf{ID}} = \frac{1}{H_1(\mathsf{ID})+s} P$.

Sign: in order to sign a message $M \in \{0,1\}^*$, the signer

1. picks a random $x \overset{R}{\leftarrow} \mathbb{Z}_p^*$ and computes $r = g^x$,
2. sets $h = H_2(M, r) \in \mathbb{Z}_p^*$,
3. computes $S = (x + h)S_{\mathsf{ID}}$.

The signature on M is $\sigma = (h, S) \in \mathbb{Z}_p^* \times \mathbb{G}_1$.

Verify: a signature $\sigma = (h, S)$ on a message M is accepted iff

$$h = H_2(M, e(S, H_1(\mathsf{ID})Q + Q_{pub})g^{-h}).$$

The scheme can be thought of as an identity-based extension of a digital signature discussed in two independent papers [9,38]. More precisely, the method for obtaining private keys from identities is a simplification of a method suggested by Sakai and Kasahara ([33]).

In [25], Kurosawa and Heng described an identity-based identification (IBI) protocol that implicitly suggests an IBS described in appendix E and which can be proven secure under the same assumption as our proposal. It turns out that ours is slightly faster than the Kurosawa-Heng IBS in the signature generation.

At Eurocrypt'04, Bellare, Namprempre and Neven established a framework [5] for proving the security of a large family of identity-based signatures and they only found two schemes to which their framework does not apply. The present one does not either fall into the category of schemes to which it applies. Indeed, it can be showed that our IBS does not result from the transformation of any convertible standard identification or signature scheme (in the sense of [5]) unless the q-SDHP is easy. A direct security proof is thus needed.

3.1 Security Results

We recall here the usual model [5,12,15,19,24] of security for identity-based signatures which is an extension of the usual notion of existential unforgeability under chosen-message attacks [22].

Definition 2 ([12]). *An IBS scheme is* ***existentially unforgeable*** *under adaptive chosen message and identity 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 to the following oracles:*
   - *Key extraction oracle: returns private keys for arbitrary identities.*
   - *Signature oracle: produces signatures on arbitrary messages using the private key corresponding to arbitrary identities.*
3. *$\mathcal{F}$ produces a triple $(\mathsf{ID}^*, M^*, \sigma^*)$ made of an identity ID^*, whose private key was never extracted, and a message-signature pair (M^*, σ^*) such that (M^*, ID^*) was not submitted to the signature oracle. She wins if the verification algorithm accepts the triple $(\mathsf{ID}^*, M^*, \sigma^*)$.*

The next lemmas establish the security of the scheme under the q-SDH assumption. Lemma 1 [12] allows to only consider a weaker attack where a forger is challenged on a given identity chosen by the challenger. The proof of lemma 2 relies on the forking lemma [31,32].

Lemma 1 ([12]). *If there is a forger $\mathcal{F}_0$ for an adaptively chosen message and identity attack having advantage ϵ_0 against our scheme when running in a time t_0 and making q_{h_1} queries to random oracle h_1, then there exists an algorithm $\mathcal{F}_1$ for an adaptively chosen message and given identity attack which has advantage $\epsilon_1 \leq \epsilon_0\big(1 - \frac{1}{2^k}\big)/q_{h_1}$ within a running time $t_1 \leq t_0$. Moreover, $\mathcal{F}_1$ asks the same number key extraction queries, signature queries and H_2-queries as $\mathcal{F}_0$ does.*

Lemma 2. *Let us assume that there is an adaptively chosen message and given identity attacker $\mathcal{F}$ that makes q_{h_i} queries to random oracles H_i $(i = 1, 2)$ and q_s queries to the signing oracle. Assume that, within a time t, $\mathcal{F}$ produces a forgery with probability $\epsilon \geq 10(q_s + 1)(q_s + q_{h_2})/2^k$. Then, there exists an algorithm $\mathcal{B}$ that is able to solve the q-SDHP for $q = q_{h_1}$ in an expected time*

$$t' \leq 120686 q_{h_2}(t + O(q_s\tau_p))/(\epsilon(1 - q/2^k)) + O(q^2\tau_{mult})$$

where τ_{mult} denotes the cost of a scalar multiplication in $\mathbb{G}_2$ and τ_p is the cost of a pairing evaluation.

Proof. See appendix A. □

The combination of the above lemmas yields the following theorem.

Theorem 1. *Let us assume that there exists an adaptively chosen message and identity attacker $\mathcal{F}$ making q_{h_i} queries to random oracles H_i $(i = 1, 2)$ and q_s queries to the signing oracle. Assume that, within a time t, $\mathcal{F}$ produces a forgery with probability $\epsilon \geq 10(q_s + 1)(q_s + q_{h_2})/2^k$. Then, there exists an algorithm $\mathcal{B}$ that is able to solve the q-SDHP for $q = q_{h_1}$ in an expected time*

$$t' \leq 120686 q_{h_1} q_{h_2}(t + O(q_s\tau_p))/(\epsilon(1 - q/2^k)) + O(q^2\tau_{mult})$$

where τ_{mult} and τ_p respectively denote the cost of a scalar multiplication in $\mathbb{G}_2$ and the required time for a pairing evaluation.

4 Fast Identity-Based Signcryption

4.1 Formal Model of Identity-Based Signcryption

The formal structure that we shall use for identity-based signcryption schemes is the following.

Setup: is a probabilistic algorithm run by a private key generator (PKG) that takes as input a security parameter to output public parameters params and a master key mk that is kept secret.

Keygen: is a key generation algorithm run by the PKG on input of params and the master key mk to return the private key S_{ID} associated to the identity ID.

Sign/Encrypt: is a probabilistic algorithm that takes as input public parameters params, a plaintext message M, the recipient's identity ID_R, and the sender's private key S_{ID_S}, and outputs a ciphertext $\sigma = \mathsf{Sign/Encrypt}(M, S_{\mathsf{ID}_S}, \mathsf{ID}_R)$.

Decrypt/Verify: is a deterministic decryption algorithm that takes as input a ciphertext σ, public parameters params, the receiver's private key S_{ID_R} and (optionally) a sender's identity ID_S before returning a valid message-signature pair (M, s) or a distinguished symbol $\perp$ if σ does not decrypt into a message bearing signer ID_S's signature.

Unlike recent works of [11,14] that present two-layer designs of probabilistic signature followed by a deterministic encryption, our construction is a single-layer construction jointly achieving signature and encryption on one side and decryption and verification on the other side. Although the description of our scheme could be modified to fit a two-layer formalism, we kept the monolithic presentation without hampering the non-repudiation property as, similarly to [11,14], our construction enables an ordinary signature on the plaintext to be extracted from any properly formed ciphertext using the recipient's private key. The extracted message-signature pair can be forwarded to any third party in such a way that a sender remains committed to the content of the plaintext.

Unlike models of [11,14] that consider anonymous ciphertexts, the above one assumes that senders' identities are sent in the clear along with ciphertexts. Actually, receivers do not need to have any a priori knowledge on whom the ciphertext emanates from in our scheme but this simply allows more efficient reductions in the security proofs. A simple modification of our scheme yields anonymous ciphertexts and enables senders' identities to be recovered by the Decrypt/Verify algorithm (which only takes a ciphertext and the recipient's private key as input).

Definition 3. *An identity-based signcryption scheme (IBSC) satisfies the* ***message confidentiality*** *property (or adaptive chosen-ciphertext security: IND-IBSC-CCA) if no PPT adversary has a non-negligible 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* params *to the $\mathcal{A}$.*
2. *In a find stage, $\mathcal{A}$ starts probing the following oracles:*
   - *Keygen: returns private keys associated to arbitrary identities.*
   - *Sign/Encrypt: given a pair of 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.*
   - *Decrypt/Verify: given a pair of identities $(\mathsf{ID}_S, \mathsf{ID}_R)$ and a ciphertext σ, it generates the receiver's private key $S_{\mathsf{ID}_R} = \mathsf{Keygen}(\mathsf{ID}_R)$ and returns either a valid message-signature pair (M, s) for the sender's identity ID_S or the $\perp$ symbol if, under the private key S_{ID_R}, σ does not decrypt into a valid message-signature pair.*
3. *$\mathcal{A}$ produces two plaintexts $M_0, M_1 \in \mathcal{M}$ and identities ID_S^* and ID_R^*. She may not have extracted the private key of ID_R^* and she obtains $C = \mathsf{Sign/Encrypt}(M_b, S_{\mathsf{ID}_S^*}, \mathsf{ID}_R^*, \mathsf{params})$, for a random a bit $b \xleftarrow{R} \{0, 1\}$.*
4. *In the guess stage, $\mathcal{A}$ asks new queries as in the find stage. This time, she may not issue a key extraction request on ID_R^* and she cannot submit C to the Decrypt/Verify oracle for the target identity ID_R^*.*
5. *Finally, $\mathcal{A}$ outputs a bit b' and wins if $b' = b$.*

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

The next definition, given in [11], considers non-repudiation w.r.t. signatures embedded in ciphertexts rather than w.r.t. ciphertexts themselves.

Definition 4. *An identity-based signcryption scheme (IBSC) is said to be* ***existentially signature-unforgeable*** *against adaptive chosen messages and ciphertexts attacks (ESUF-IBSC-CMA) 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. *$\mathcal{F}$ issues a number of queries as in the previous definition.*
3. *Finally, $\mathcal{F}$ outputs a triple $(\sigma^*, \mathsf{ID}_S^*, \mathsf{ID}_R^*)$ and wins the game if the sender's identity ID_S^* was not corrupted and if the result of the* $\mathsf{Decrypt/Verify}$ *oracle on the ciphertext σ^* under the private key associated to ID_R^* is a valid message-signature pair (M^*, s^*) such that no* $\mathsf{Sign/Encrypt}$ *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 $S_{\mathsf{ID}_R'}$ is the alleged forgery $(M^*, s^*, \mathsf{ID}_S^*)$.*

The adversary's advantage is its probability of victory.

In both of these definitions, we consider insider attacks [1]. Namely, in the definition of message confidentiality, the adversary is allowed to be challenged on a ciphertext created using a corrupted sender's private key whereas, in the notion of signature non-repudiation, the forger may output a ciphertext computed under a corrupted receiving identity.

4.2 The Scheme

Our scheme is obtained from an optimized combination of our IBS scheme with the most basic version of the Sakai-Kasahara IBE ([33,13]) which is only secure against chosen-plaintext attacks when used as an encryption-only system. This allows performing the signature-encryption operation without computing a pairing whereas only two pairings have to be computed upon decryption/verification.

Setup: given k, the PKG chooses bilinear map groups $(\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T)$ of prime order $p > 2^k$ and generators $Q \in \mathbb{G}_2$, $P = \psi(Q) \in \mathbb{G}_1$, $g = e(P, Q) \in \mathbb{G}_T$. It then chooses a master key $s \xleftarrow{R} \mathbb{Z}_p^*$, a system-wide public key $Q_{pub} = sQ \in \mathbb{G}_2$ and hash functions $H_1 : \{0,1\}^* \rightarrow \mathbb{Z}_p^*$, $H_2 : \{0,1\}^* \times \mathbb{G}_T \rightarrow \mathbb{Z}_p^*$ and $H_3 : \mathbb{G}_T \rightarrow \{0,1\}^n$. The public parameters are

$$\mathsf{params} := \{\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, P, Q, g, Q_{pub}, e, \psi, H_1, H_2, H_3\}$$

Keygen: for an identity ID, the private key is $S_{\mathsf{ID}} = \frac{1}{H_1(\mathsf{ID})+s} Q \in \mathbb{G}_2$.

Sign/Encrypt: given a message $M \in \{0,1\}^*$, a receiver's identity ID_B and a sender's private key S_{ID_A},

1. Pick $x \xleftarrow{R} \mathbb{Z}_p^*$, compute $r = g^x$ and $c = M \oplus H_3(r) \in \{0,1\}^n$.
2. Set $h = H_2(M, r) \in \mathbb{Z}_p^*$.
3. Compute $S = (x+h)\psi(S_{\mathsf{ID}_A})$.
4. Compute $T = x(H_1(\mathsf{ID}_B)P + \psi(Q_{pub}))$.

The ciphertext is $\sigma = \langle c, S, T\rangle \in \{0,1\}^n \times \mathbb{G}_1 \times \mathbb{G}_1$.

Decrypt/Verify: given $\sigma = \langle c, S, T\rangle$, and some sender's identity ID_A,

1. Compute $r = e(T, S_{\mathsf{ID}_B})$, $M = c \oplus H_3(r)$, and $h = H_2(M, r)$.
2. Accept the message iff $r = e(S, H_1(\mathsf{ID}_A)Q + Q_{pub})g^{-h}$. If this condition holds, return the message M and the signature $(h, S) \in \mathbb{Z}_p^* \times \mathbb{G}_1$.

If required, the anonymity property is obtained by scrambling the sender's identity ID_A together with the message at step 1 of Sign/Encrypt in such a way that the recipient retrieves it at the first step of the reverse operation. This change does not imply any computational penalty in practice but induces more expensive security reductions. In order for the proof to hold, ID_A must be appended to the inputs of H_2.

4.3 Security Results

The following theorems claim the security of the scheme in the random oracle model under the same irreflexivity assumption as Boyen's scheme [11]: the signature/encryption algorithm is assumed to always take distinct identities as inputs (in other words, a principal never encrypts a message bearing his signature using his own identity).

Theorem 2. *Assume that an IND-IBSC-CCA adversary $\mathcal{A}$ has an advantage ϵ against our scheme when running in time τ, asking q_{h_i} queries to random oracles H_i $(i = 1, 2, 3)$, q_{se} signature/encryption queries and q_{dv} queries to the decryption/verification oracle. Then there is an algorithm $\mathcal{B}$ to solve the q-BDHIP for $q = q_{h_1}$ with probability*

$$\epsilon' > \frac{\epsilon}{q_{h_1}(2q_{h_2} + q_{h_3})}\left(1 - q_{se}\frac{q_{se} + q_{h_2}}{2^k}\right)\left(1 - \frac{q_{dv}}{2^k}\right)$$

within a time $\tau' < \tau + O(q_{se} + q_{dv})\tau_p + O(q_{h_1}^2)\tau_{mult} + O(q_{dv}q_{h_2})\tau_{exp}$ where τ_{exp} and τ_{mult} are respectively the costs of an exponentiation in $\mathbb{G}_T$ and a multiplication in $\mathbb{G}_2$ whereas τ_p is the complexity of a pairing computation.

Proof. See appendix B. □

Theorem 3. *Assume there exists an ESUF-IBSC-CMA attacker $\mathcal{A}$ that makes q_{h_i} queries to random oracles H_i $(i = 1, 2, 3)$, q_{se} signature/ encryption queries and q_{dv} queries to the decryption/verification oracle. Assume also that, within a time τ, $\mathcal{A}$ produces a forgery with probability $\epsilon \geq 10(q_{se}+1)(q_{se}+q_{h_2})/2^k$. Then, there is an algorithm $\mathcal{B}$ that is able to solve the q-SDHP for $q = q_{h_1}$ in expected time*

$$\tau' \leq 120686 q_{h_1} q_{h_2} \frac{\tau + O((q_{se} + q_{dv})\tau_p) + q_{dv}q_{h_2}\tau_{exp}}{\epsilon(1 - 1/2^k)(1 - q/2^k)} + O(q^2\tau_{mult})$$

where τ_{mult}, τ_{exp} and τ_p denote the same quantities as in theorem 2.

Proof. See appendix C. □

We now restate theorem 2 for the variant of our scheme with anonymous ciphertexts. The simulator's worst-case running time is affected by the fact that, when handling Decrypt/Verify requests, senders'identities are not known in advance. The reduction involves a number of pairing calculations which is quadratic in the number of adversarial queries.

Theorem 4. *Assume that an IND-IBSC-CCA adversary $\mathcal{A}$ has an advantage ϵ against our scheme when running in time τ, asking q_{h_i} queries to random oracles H_i $(i = 1, 2, 3)$, q_{se} signature/encryption queries and q_{dv} queries to the decryption/verification oracle. Then there is an algorithm $\mathcal{B}$ to solve the q-BDHIP for $q = q_{h_1}$ with probability*

$$\epsilon' > \frac{\epsilon}{q_{h_1}(2q_{h_2} + q_{h_3})} \left(1 - q_{se}\frac{q_{se} + q_{h_2}}{2^k}\right)\left(1 - \frac{q_{dv}}{2^k}\right)$$

within a time $\tau' < \tau + O(q_{se} + q_{dv}q_{h_2})\tau_p + O(q_{h_1}^2)\tau_{mult} + O(q_{dv}q_{h_2})\tau_{exp}$ where τ_{exp}, τ_{mult} and τ_p denote the same quantities as in previous theorems.

Proof. See appendix D. □

Theorem 3 can be similarly restated as its reduction cost is affected in the same way.

A formal proof of *ciphertext anonymity* in the model of [11] will be given in the full version of this paper for the anonymous version of the scheme.

We concede that even the latter variant does not feature all the properties of the systems of Boyen ([11]) or Chen-Malone-Lee ([14]). For example, it does not have the *ciphertext unlinkability* property ([11,14]): it seems infeasible for anyone to use his private key to embed a given message-signature pair into a proper ciphertext intended to himself. We were also unable to formally establish the *ciphertext authentication property* according to which a ciphertext is always signed and encrypted by the same person and cannot be subject to a kind of 'man-in-the-middle' attack. Nevertheless, the scheme does seem to have this property because of the same reason that precludes the ciphertext unlinkability property.

Overall, we believe that the scheme does satisfy the main requirements that might be desired in practice. In our opinion, it suffices to implement most practical applications and its great efficiency renders it more than interesting for identity-based cryptography.

5 Efficiency Discussions and Comparisons

In [35], Smart and Vercauteren pointed out problems that arise when several pairing based protocols are implemented with asymmetric pairings. They showed the difficulty of finding groups $\mathbb{G}_2$ allowing the use of the most efficient pairing calculation techniques for ordinary curves [3] if arbitrary strings should be *efficiently* hashed onto them and efficient isomorphism $\psi : \mathbb{G}_2 \rightarrow \mathbb{G}_1$ must be available at the same time. As a consequence, several protocols have to be implemented with groups for which no efficient isomorphism $\psi : \mathbb{G}_2 \rightarrow \mathbb{G}_1$ is computable and their security eventually has to rely on somewhat unnatural assumptions.

Except [33] that has no security proof (and actually has several known security problems [28]), all known identity-based signcryption schemes would require to hash onto $\mathbb{G}_2$ if they were instantiated with asymmetric pairings. Our scheme avoids this problem since it does not require to hash onto a cyclic group. It thus more easily benefits from optimized pairing calculation algorithms. For example, section 4 of [35] yields an example of group $\mathbb{G}_2$ for which techniques of [3] can be used and where efficient isomorphisms are available.

We now assess the comparative efficiency of several identity-based signcryption schemes, implemented according to their original descriptions. Table 1 summarises the number of relevant basic operations underlying several identity-based signcryption and signature schemes, namely, $\mathbb{G}_T$ exponentiations, scalar point

Table 1. Efficiency comparison

	Sign/Encrypt				Decrypt/Verify			
signcryption scheme	exp	mul	pairings	time (ms)	exp	mul	pairings	time (ms)
Boyen ([11])	1	3	$1^{\dagger}$	9.37		2	$4^{\dagger}$	12.66
Chow-Yiu-Hui-Chow¶ ([16])		2	$2^{\star}$	7.24		1	$4^{\star}$	11.88
Libert-Quisquater$^{\P\spadesuit}$ ([26])		2	$2^{\star}$	7.24		1	$4^{\star}$	11.88
Nalla-Reddy$^{\diamondsuit\bowtie}$ ([30])	1	2	$1^{\dagger}$	8.43	1		$3^{\dagger}$	9.06
Malone-Lee$^{\clubsuit}$ ([27])		3	$1^{\ddagger}$	5.47		1	3	9.06
Chen-Malone-Lee ([14])		3	$1^{\ddagger}$	5.47		1	3	9.06
Sakai-Kasahara$^{\clubsuit}$ ([33])	2	$1{+}1^{\S}$		6.41	1		2	9.37
Libert-Quisquater$^{\bowtie}$ ([26])		3	$1^{\ddagger}$	5.47		1	2	6.41
ours	1	2		2.65	1		2	6.09
	Sign				Verify			
signature scheme	exp	mul	pairings	time (ms)	exp	mul	pairings	time (ms)
Chow-Yiu-Hui-Chow ([16])		2	$1^{\ddagger}$	3.60			$2^{\dagger}$	6.41
Heß([24])	1	2		2.50	1		$2^{\dagger}$	6.41
Cha-Cheon ([12])		2		1.87		1	2	6.41
ours		2		1.56		1	1	3.60

(†) One pairing is precomputable, incurring for each user a storage cost of one $\mathbb{G}_T$ element for each other user in the system.

(‡) One pairing is precomputable, incurring for each user a storage cost of one $\mathbb{G}_T$ element for each other user in the system, plus one $\mathbb{G}_T$ exponentiation.

($\star$) Two pairings are precomputable, incurring for each user a storage cost of one $\mathbb{G}_T$ element for each user in the system, plus two $\mathbb{G}_T$ exponentiations.

(§) One of the scalar multiplications is done in $\langle Q \rangle$ rather than $\langle P \rangle$ where (P, Q) generates $E[p]$.

(¶) Universally verifiable scheme (i.e. supports public ciphertext validation).

(♣) These schemes suffer from security problems as mentioned in [26,28].

(♠) This scheme does not provide insider-security for the message-confidentiality criterion.

(◇) This scheme has no security proof.

(⋈) This construction can only authenticate messages from the receiver's point of view.

multiplications, and pairing evaluations, and compares the observed processing times (in milliseconds) for a supersingular curve of embedding degree $k = 6$ over $\mathbb{F}_{3^{97}}$, using implementations written in C++ and run on an Athlon XP 2 GHz. Subtleties in the algorithms determine somewhat different running times even when the operation counts for those algorithms are equal. We see from these results that our proposed algorithms rank among the fastest schemes.

6 Conclusion

We have described efficient and provably secure signature and signcryption schemes that are faster than any pairing-based scheme previously proposed in the literature. The latter can be instantiated with either named or anonymous ciphertexts and is more convenient than previous proposals for implementations with asymmetric pairings.

References

1. J.-H. An, Y. Dodis, and T. Rabin. On the security of joint signature and encryption. In *Eurocrypt'02*, volume 2332 of *LNCS*, pages 83–107. Springer, 2002.
2. P. S. L. M. Barreto. The pairing based crypto lounge. http://planeta.terra.com.br/informatica/paulobarreto/pblounge.html.
3. P. S. L. M. Barreto, B. Lynn, and M. Scott. On the selection of pairing-friendly groups. In *SAC'03*, volume 3006 of *LNCS*, pages 17–25. Springer, 2003.
4. P. S. L. M. Barreto and M. Naehrig. Pairing-friendly elliptic curves of prime order. Cryptology ePrint Archive, Report 2005/133, 2005. http://eprint.iacr.org/2005/133.
5. M. Bellare, C. Namprempre, and G. Neven. Security proofs for identity-based identification and signature schemes. In *Eurocrypt'04*, volume 3027 of *LNCS*, pages 268–286. Springer, 2004.
6. M. Bellare and P. Rogaway. Random oracles are practical: A paradigm for designing efficient protocols. In *1st ACM Conference on Computer and Communications Security*, pages 62–73, Fairfax, USA, 1993. ACM Press.
7. D. Boneh and X. Boyen. Efficient selective-ID secure identity based encryption without random oracles. In *Eurocrypt'04*, volume 3027 of *LNCS*, pages 223–238. Springer, 2004.
8. D. Boneh and X. Boyen. Secure identity based encryption without random oracles. In *Crypto'04*, volume 3152 of *LNCS*, pages 443–459. Springer, 2004.
9. D. Boneh and X. Boyen. Short signatures without random oracles. In *Eurocrypt'04*, volume 3027 of *LNCS*, pages 56–73. Springer, 2004.
10. D. Boneh and M. Franklin. Identity-based encryption from the Weil pairing. In *Crypto'01*, volume 2139 of *LNCS*, pages 213–229. Springer, 2001.
11. X. Boyen. Multipurpose identity-based signcryption: A swiss army knife for identity-based cryptography. In *Crypto'03*, volume 2729 of *LNCS*, pages 383–399. Springer, 2003.
12. J. C. Cha and J. H. Cheon. An identity-based signature from gap Diffie-Hellman groups. In *PKC'03*, volume 2567 of *LNCS*, pages 18–30. Springer, 2003.

13. L. Chen and Z. Cheng. Security proof of Sakai-Kasahara's identity-based encryption scheme. Cryptology ePrint Archive, Report 2005/226, 2005. `http://eprint.iacr.org/2005/226`.
14. L. Chen and J. Malone-Lee. Improved identity-based signcryption. In *PKC'05*, volume 3386 of *LNCS*, pages 362–379. Springer, 2005.
15. J. H. Cheon, Y. Kim, and H. J. Yoon. A new id-based signature with batch verification. Cryptology ePrint Archive, Report 2004/131, 2004. `http://eprint.iacr.org/2004/131`.
16. 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 *6th International Conference on Information Security and Cryptology – ICISC'03*, volume 2971 of *LNCS*, pages 352–369. Springer, 2003.
17. S. S. M. Chow, T. H. Yuen, L. C. K. Hui, and S. M. Yiu. Signcryption in hierarchical identity based cryptosystem. In *20th International Conference on Information Security (SEC'05)*. IFIP TC11, 2005.
18. C. Cocks. An identity based encryption scheme based on quadratic residues. In *8th IMA International Conference*, volume 2260 of *LNCS*, pages 360–363. Springer, 2001.
19. Y. Dodis, J. Katz, S. Xu, and M. Yung. Strong key-insulated signature schemes. In *PKC'03*, volume 2567 of *LNCS*, pages 130–144. Springer, 2003.
20. A. Fiat and A. Shamir. How to prove yourself: Practical solutions to identification and signature problems. In *Crypto'86*, volume 0263 of *LNCS*, pages 186–194. Springer, 1986.
21. C. Gentry and A. Silverberg. Hierarchical ID-based cryptography. In *Asiacrypt'02*, volume 2501 of *LNCS*, pages 548–566. Springer, 2002.
22. S. Goldwasser, S. Micali, and R. Riverst. A digital signature scheme secure against adaptive chosen message attacks. *SIAM Journal of Computing*, 17(2):281–308, 1988.
23. L. Guillou and J.-J. Quisquater. A "paradoxical" identity-based signature scheme resulting from zero-knowledge. In *Crypto'88*, volume 0403 of *LNCS*, pages 216–231. Springer, 1988.
24. F. Heß. Efficient identity based signature schemes based on pairings. In *SAC'02*, volume 2595 of *LNCS*, pages 310–324. Springer, 2003.
25. K. Kurosawa and S.-H. Heng. Identity-based identification without random oracles. In *ISH'05*, LNCS. Springer, 2005. To appear.
26. B. Libert and J.-J. Quisquater. New identity based signcryption schemes from pairings. In *IEEE Information Theory Workshop*, Paris, France, 2003. `http://eprint.iacr.org/2003/023`.
27. J. Malone-Lee. Identity-based signcryption. Cryptology ePrint Archive, Report 2002/098, 2002. `http://eprint.iacr.org/2002/098`.
28. N. McCullagh and P. S. L. M. Barreto. Efficient and forward-secure identity-based signcryption. Cryptology ePrint Archive, Report 2004/117, 2004. `http://eprint.iacr.org/2004/117`.
29. A. Miyaji, M. Nakabayashi, and S. Takano. New explicit conditions of elliptic curve traces for FR-reduction. *IEICE Transactions on Fundamentals*, E84-A(5):1234–1243, 2001.
30. D. Nalla and K. C. Reddy. Signcryption scheme for identity-based cryptosystems. Cryptology ePrint Archive, Report 2003/066, 2003. `http://eprint.iacr.org/2003/066`.
31. D. Pointcheval and J. Stern. Security proofs for signature schemes. In *Eurocrypt'96*, volume 1992 of *LNCS*, pages 387–398. Springer, 1996.

32. D. Pointcheval and J. Stern. Security arguments for digital signatures and blind signatures. *Journal of Cryptology*, 13(3):361–396, 2000.
33. R. Sakai and M. Kasahara. ID based cryptosystems with pairing on elliptic curve. In *SCIS'03*, Hamamatsu, Japan, 2003. `http://eprint.iacr.org/2003/054`.
34. A. Shamir. Identity based cryptosystems and signature schemes. In *Crypto'84*, volume 0196 of *LNCS*, pages 47–53. Springer, 1984.
35. N. P. Smart and F. Vercauteren. On computable isomorphisms in efficient pairing based systems. Cryptology ePrint Archive, Report 2005/116, 2005. `http://eprint.iacr.org/2005/116`.
36. B. Waters. Efficient identity-based encryption without random oracles. In *Eurocrypt'05*, volume 3494 of *LNCS*, pages 114–127. Springer, 2005.
37. T. H. Yuen and V. K. Wei. Fast and proven secure blind identity-based signcryption from pairings. In *CT-RSA'05*, volume 3376 of *LNCS*, pages 305–322. Springer, 2003.
38. F. Zhang, R. Safavi-Naini, and W. Susilo. An efficient signature scheme from bilinear pairings and its applications. In *PKC'04*, volume 2947 of *LNCS*, pages 277–290. Springer, 2004.
39. Y. Zheng. Digital signcryption or how to achieve cost (signature & encryption) << cost(signature) + cost(encryption). In *Crypto'97*, volume 1294 of *LNCS*, pages 165–179. Springer, 1997.

A Proof of Lemma 2

Proof. We first show how to provide the adversary with a consistent view and we then explain how to apply the forking lemma.

Algorithm $\mathcal{B}$ takes as input $(P, Q, \alpha Q, \alpha^2 Q, \ldots, \alpha^q Q)$ and aims to find a pair $(c, \frac{1}{c+\alpha}P)$. In a setup phase, it builds a generator $G \in \mathbb{G}_1$ such that it knows $q-1$ pairs $(w_i, \frac{1}{w_i+\alpha}G)$ for $w_1, \ldots, w_{q-1} \in_R \mathbb{Z}_p^*$. To do so,

1. It picks $w_1, w_2, \ldots, w_{q-1} \stackrel{R}{\leftarrow} \mathbb{Z}_p^*$ and expands $f(z) = \prod_{i=1}^{q-1}(z + w_i)$ to obtain $c_0, \ldots, c_{q-1} \in \mathbb{Z}_p^*$ so that $f(z) = \sum_{i=0}^{q-1} c_i z^i$.
2. It sets generators $H = \sum_{i=0}^{q-1} c_i(\alpha^i Q) = f(\alpha)Q \in \mathbb{G}_2$ and $G = \psi(H) = f(\alpha)P \in \mathbb{G}_1$. The public key $H_{pub} \in \mathbb{G}_2$ is fixed to $H_{pub} = \sum_{i=1}^{q} c_{i-1}(\alpha^i Q)$ so that $H_{pub} = \alpha H$ although $\mathcal{B}$ does not know α.
3. For $1 \leq i \leq q-1$, $\mathcal{B}$ expands $f_i(z) = f(z)/(z + w_i) = \sum_{i=0}^{q-2} d_i z^i$ and
$$\sum_{i=0}^{q-2} d_i \psi(\alpha^i Q) = f_i(\alpha)P = \frac{f(\alpha)}{\alpha + w_i}P = \frac{1}{\alpha + w_i}G. \qquad (1)$$
The pairs $(w_i, \frac{1}{\alpha+w_i}G)$ are computed using the left member of (1).

$\mathcal{B}$ is then ready to answer $\mathcal{F}$'s queries along the course of the game. It first initializes a counter ℓ to 1 and launches $\mathcal{F}$ on the input (H_{pub}, ID^*) for a randomly chosen challenge identity $\mathsf{ID}^* \stackrel{R}{\leftarrow} \{0,1\}^*$. For simplicity, we assume that queries to H_1 are distinct, and that any query involving an identifier ID is preceded by the random oracle query $H_1(\mathsf{ID})$.

- H_1-queries on an identity $\mathsf{ID} \in \{0,1\}^*$: $\mathcal{B}$ returns a random $w^* \overset{R}{\leftarrow} \mathbb{Z}_p^*$ if $\mathsf{ID} = \mathsf{ID}^*$. Otherwise, $\mathcal{B}$ answers $w = w_\ell \in \mathbb{Z}_p^*$ and increments ℓ. In both cases, $\mathcal{B}$ stores (ID, w) (where $w^* = w$ or w_ℓ) in a list L_1.
- Key extraction queries on $\mathsf{ID} \neq \mathsf{ID}^*$: $\mathcal{B}$ recovers the matching pair (ID, w) from L_1 and returns the previously computed $(1/(\alpha + w))G$.
- Signature query on a message-identity pair (M, ID): $\mathcal{B}$ picks $S \overset{R}{\leftarrow} \mathbb{G}_1$, $h \overset{R}{\leftarrow} \mathbb{Z}_p^*$, computes $r = e(S, Q_{\mathsf{ID}})e(G, H)^{-h}$, where $Q_{\mathsf{ID}} = H_1(\mathsf{ID})H + H_{pub}$, and backpatches to define the value $H_2(M, r)$ as $h \in \mathbb{Z}_p^*$ ($\mathcal{B}$ aborts in the unlikely event that $H_2(M, r)$ is already defined).

We have explained how to simulate $\mathcal{F}$'s environment in a chosen-message and given identity attack. We are ready to apply the forking lemma that essentially says the following: consider a scheme producing signatures of the form (M, r, h, S), where each of r, h, S corresponds to one of the three moves of a honest-verifier zero-knowledge protocol. Let us assume that a chosen-message attacker $\mathcal{F}$ forges a signature (M, r, h, S) in a time t with probability $\epsilon \geq 10(q_s + 1)(q_s + q_h)/2^k$ (k being a security parameter so that h is uniformly taken from a set of 2^k elements) when making q_s signature queries and q_h random oracle calls. If the triples (r, h, S) can be simulated without knowing the private key, then there exists a Turing machine $\mathcal{F}'$ that uses $\mathcal{F}$ to produce two valid signatures (m, r, h_1, S_1), (m, r, h_2, S_2), with $h_1 \neq h_2$, in expected time $t' \leq 120686 q_h t/\epsilon$.

In our setting, from a forger $\mathcal{F}$, we build an algorithm $\mathcal{F}'$ that replays $\mathcal{F}$ a sufficient number of times on the input (H_{pub}, ID^*) to obtain two suitable forgeries $\langle M^*, r, h_1, S_1\rangle$, $\langle M^*, r, h_2, S_2\rangle$ with $h_1 \neq h_2$.

The reduction then works as follows. The simulator $\mathcal{B}$ runs $\mathcal{F}'$ to obtain two forgeries $\langle M^*, r, h_1, S_1\rangle$, $\langle M^*, r, h_2, S_2\rangle$ for the same message M^* and commitment r. At this stage, $\mathcal{B}$ recovers the pair (ID^*, w^*) from list L_1. We note that $w^* \neq w_1, \ldots, w_{q-1}$ with probability at least $1 - q/2^k$. If both forgeries satisfy the verification equation, we obtain the relations

$$e(S_1, Q_{\mathsf{ID}^*})e(G, H)^{-h_1} = e(S_2, Q_{\mathsf{ID}^*})e(G, H)^{-h_2},$$

with $Q_{\mathsf{ID}^*} = H_1(\mathsf{ID}^*)H + H_{pub} = (w^* + \alpha)H$. Then, it comes that

$$e((h_1 - h_2)^{-1}(S_1 - S_2), Q_{\mathsf{ID}^*}) = e(G, H),$$

and hence $T^* = (h_1 - h_2)^{-1}(S_1 - S_2) = \frac{1}{w^* + \alpha}G$. From T^*, $\mathcal{B}$ can proceed as in [9] to extract $\sigma^* = \frac{1}{w^* + \alpha}P$: it first obtains $\gamma_{-1}, \gamma_0, \ldots, \gamma_{q-2} \in \mathbb{Z}_p^*$ for which $f(z)/(z + w^*) = \gamma_{-1}/(z + w^*) + \sum_{i=0}^{q-2} \gamma_i z^i$ and eventually computes

$$\sigma^* = \frac{1}{\gamma_{-1}}\left[T^* - \sum_{i=0}^{q-2} \gamma_i \psi(\alpha^i Q)\right] = \frac{1}{w^* + \alpha}P$$

before returning the pair (w^*, σ^*) as a result.

It finally comes that, if $\mathcal{F}$ forges a signature in a time t with probability $\epsilon \geq 10(q_s + 1)(q_s + q_{h_2})/2^k$, $\mathcal{B}$ solves the q-SDHP in expected time

$$t' \leq 120686 q_{h_2}(t + O(q_s \tau_p))/(\epsilon(1 - q/2^k)) + O(q^2 \tau_{mult})$$

where the last term accounts for the cost of the preparation phase. □

B Proof of Theorem 2

Proof. Algorithm $\mathcal{B}$ takes as input $\langle P, Q, \alpha Q, \alpha^2 Q, \ldots, \alpha^q Q \rangle$ and attempts to extract $e(P,Q)^{1/\alpha}$ from its interaction with $\mathcal{A}$.

In a preparation phase, $\mathcal{B}$ selects $\ell \overset{R}{\leftarrow} \{1, \ldots, q_{h_1}\}$, elements $I_\ell \overset{R}{\leftarrow} \mathbb{Z}_p^*$ and $w_1, \ldots, w_{\ell-1}, w_{\ell+1} \ldots, w_q \overset{R}{\leftarrow} \mathbb{Z}_p^*$. For $i = 1, \ldots, \ell-1, \ell+1, \ldots, q$, it computes $I_i = I_\ell - w_i$. As in the technique of [9] and in lemma 2, it sets up generators $G_2 \in \mathbb{G}_2$, $G_1 = \psi(G_2) \in \mathbb{G}_1$ and another $\mathbb{G}_2$ element $U = \alpha G_2$ such that it knows $q-1$ pairs $(w_i, H_i = (1/(w_i + \alpha))G_2)$ for $i \in \{1, \ldots, q\} \backslash \{\ell\}$. The system-wide public key Q_{pub} is chosen as

$$Q_{pub} = -U - I_\ell G_2 = (-\alpha - I_\ell)G_2$$

so that its (unknown) private key is implicitly set to $x = -\alpha - I_\ell \in \mathbb{Z}_p^*$. For all $i \in \{1, \ldots, q\} \backslash \{\ell\}$, we have $(I_i, -H_i) = (I_i, (1/(I_i + x))G_2)$.

$\mathcal{B}$ then initializes a counter ν to 1 and starts $\mathcal{A}$ on input of (G_1, G_2, Q_{pub}). Throughout the game, we assume that H_1-queries are distinct, that the target identity ID_R^* is submitted to H_1 at some point and that any query involving an identity ID comes after a H_1-query on ID:

- H_1-queries (let us call ID_ν the input of the ν^{th} one of such queries): $\mathcal{B}$ answers I_ν and increments ν.
- H_2-queries on input (M, r): $\mathcal{B}$ returns the defined value if it exists and a random $h_2 \overset{R}{\leftarrow} \mathbb{Z}_p^*$ otherwise. To anticipate possible subsequent Decrypt/Verify requests, $\mathcal{B}$ additionally simulates random oracle H_3 on its own to obtain $h_3 = H_3(r) \in \{0,1\}^n$ and stores the information $(M, r, h_2, c = M \oplus h_3, \gamma = r \cdot e(G_1, G_2)^{h_2})$ in L_2.
- H_3-queries for an input $r \in \mathbb{G}_T$: $\mathcal{B}$ returns the previously assigned value if it exists and a random $h_3 \overset{R}{\leftarrow} \{0,1\}^n$ otherwise. In the latter case, the input r and the response h_3 are stored in a list L_3.
- Keygen queries on an input ID_ν: if $\nu = \ell$, then $\mathcal{B}$ fails. Otherwise, it knows that $H_1(\mathsf{ID}_\nu) = I_\nu$ and returns $-H_\nu = (1/(I_\nu + x))\, G_2 \in \mathbb{G}_2$.
- Sign/Encrypt queries for a plaintext M and identities $(\mathsf{ID}_S, \mathsf{ID}_R) = (\mathsf{ID}_\mu, \mathsf{ID}_\nu)$ for $\mu, \nu \in \{1, \ldots, q_{h_1}\}$: we observe that, if $\mu \neq \ell$, $\mathcal{B}$ knows the sender's private key $S_{\mathsf{ID}_\mu} = -H_\mu$ and can answer the query according to the specification of Sign/Encrypt. We thus assume $\mu = \ell$ and hence $\nu \neq \ell$ by the irreflexivity assumption. Observe that $\mathcal{B}$ knows the receiver's private key $S_{\mathsf{ID}_\nu} = -H_\nu$ by construction. The difficulty is to find a random triple $(S, T, h) \in \mathbb{G}_1 \times \mathbb{G}_1 \times \mathbb{Z}_p^*$ for which
$$e(T, S_{\mathsf{ID}_\nu}) = e(S, Q_{\mathsf{ID}_\ell}) e(G_1, G_2)^{-h} \qquad (2)$$
where $Q_{\mathsf{ID}_\ell} = I_\ell G_2 + Q_{pub}$. To do so, $\mathcal{B}$ randomly chooses $t, h \overset{R}{\leftarrow} \mathbb{Z}_p^*$ and computes $S = t\psi(S_{\mathsf{ID}_\nu}) = -t\psi(H_\nu)$, $T = t\psi(Q_{\mathsf{ID}_\ell}) - h\psi(Q_{\mathsf{ID}_\nu})$ where $Q_{\mathsf{ID}_\nu} = I_\nu G_2 + Q_{pub}$ in order to obtain the desired equality $r = e(T, S_{\mathsf{ID}_\nu}) = e(S, Q_{\mathsf{ID}_\ell}) e(G_1, G_2)^{-h} = e(\psi(S_{\mathsf{ID}_\nu}), Q_{\mathsf{ID}_\ell})^t e(G_1, G_2)^{-h}$ before patching the hash value $H_2(M, r)$ to h ($\mathcal{B}$ fails if H_2 is already defined but this only happens with probability $(q_{se} + q_{h_2})/2^k$). The ciphertext $\sigma = \langle M \oplus H_3(r), S, T \rangle$ is returned.

- Decrypt/Verify queries on a ciphertext $\sigma = \langle c, S, T \rangle$ for identities $(\mathsf{ID}_S, \mathsf{ID}_R) = (\mathsf{ID}_\mu, \mathsf{ID}_\nu)$: we assume that $\nu = \ell$ (and hence $\mu \neq \ell$ by the irreflexivity assumption), because otherwise $\mathcal{B}$ knows the receiver's private key $S_{\mathsf{ID}_\nu} = -H_\nu$ and can normally run the Decrypt/Verify algorithm. Since $\mu \neq \ell$, $\mathcal{B}$ has the sender's private key S_{ID_μ} and also knows that, for all valid ciphertexts, $\log_{S_{\mathsf{ID}_\mu}}(\psi^{-1}(S) - hS_{\mathsf{ID}_\mu}) = \log_{\psi(Q_{\mathsf{ID}_\nu})}(T)$, where $h = H_2(M, r)$ is the hash value obtained in the Sign/Encrypt algorithm and $Q_{\mathsf{ID}_\nu} = I_\nu G_2 + Q_{pub}$. Hence, we have the relation
$$e(T, S_{\mathsf{ID}_\mu}) = e(\psi(Q_{\mathsf{ID}_\nu}), \psi^{-1}(S) - hS_{\mathsf{ID}_\mu}) \tag{3}$$
which yields $e(T, S_{\mathsf{ID}_\mu}) = e(\psi(Q_{\mathsf{ID}_\nu}), \psi^{-1}(S))e(\psi(Q_{\mathsf{ID}_\nu}), S_{\mathsf{ID}_\mu})^{-h}$. We observe that the latter equality can be tested without inverting ψ as $e(\psi(Q_{\mathsf{ID}_\nu}), \psi^{-1}(S)) = e(S, Q_{\mathsf{ID}_\nu})$. The query is thus handled by computing $\gamma = e(S, Q_{\mathsf{ID}_\mu})$, where $Q_{\mathsf{ID}_\mu} = I_\mu G_2 + Q_{pub}$, and searching through list L_2 for entries of the form $(M_i, r_i, h_{2,i}, c, \gamma)$ indexed by $i \in \{1, \ldots, q_{h_2}\}$. If none is found, σ is rejected. Otherwise, each one of them is further examined: for the corresponding indexes, $\mathcal{B}$ checks if
$$e(T, S_{\mathsf{ID}_\mu})/e(S, Q_{\mathsf{ID}_\nu}) = e(\psi(Q_{\mathsf{ID}_\nu}), S_{\mathsf{ID}_\mu})^{-h_{2,i}} \tag{4}$$
(the pairings are computed only once and at most q_{h_2} exponentiations are needed), meaning that (3) is satisfied. If the unique $i \in \{1, \ldots, q_{h_2}\}$ satisfying (4) is detected, the matching pair $(M_i, \langle h_{2,i}, S \rangle)$ is returned. Otherwise, σ is rejected. Overall, an inappropriate rejection occurs with probability smaller than $q_{dv}/2^k$ across the whole game.

At the challenge phase, $\mathcal{A}$ outputs messages (M_0, M_1) and identities $(\mathsf{ID}_S, \mathsf{ID}_R)$ for which she never obtained ID_R's private key. If $\mathsf{ID}_R \neq \mathsf{ID}_\ell$, $\mathcal{B}$ aborts. Otherwise, it picks $\xi \overset{R}{\leftarrow} \mathbb{Z}_p^*$, $c \overset{R}{\leftarrow} \{0,1\}^n$ and $S \overset{R}{\leftarrow} \mathbb{G}_1$ to return the challenge $\sigma^* = \langle c, S, T \rangle$ where $T = -\xi G_1 \in \mathbb{G}_1$. If we define $\rho = \xi/\alpha$ and since $x = -\alpha - I_\ell$, we can check that
$$T = -\xi G_1 = -\alpha\rho G_1 = (I_\ell + x)\rho G_1 = \rho I_\ell G_1 + \rho\psi(Q_{pub}).$$

$\mathcal{A}$ cannot recognize that σ^* is not a proper ciphertext unless she queries H_2 or H_3 on $e(G_1, G_2)^\rho$. Along the guess stage, her view is simulated as before and her eventual output is ignored. Standard arguments can show that a successful $\mathcal{A}$ is very likely to query H_2 or H_3 on the input $e(G_1, G_2)^\rho$ if the simulation is indistinguishable from a real attack environment.

To produce a result, $\mathcal{B}$ fetches a random entry (M, r, h_2, c, γ) or $\langle r, . \rangle$ from the lists L_2 or L_3. With probability $1/(2q_{h_2} + q_{h_3})$ (as L_3 contains no more than $q_{h_2} + q_{h_3}$ records by construction), the chosen entry will contain the right element $r = e(G_1, G_2)^\rho = e(P, Q)^{f(\alpha)^2\xi/\alpha}$, where $f(z) = \sum_{i=0}^{q-1} c_i z^i$ is the polynomial for which $G_2 = f(\alpha)Q$. The q-BDHIP solution can be extracted by noting that, if $\gamma^* = e(P, Q)^{1/\alpha}$, then
$$e(G_1, G_2)^{1/\alpha} = \gamma^{*(c_0^2)} e\big(\sum_{i=0}^{q-2} c_{i+1}(\alpha^i P), c_0 Q\big) e\big(G_1, \sum_{j=0}^{q-2} c_{j+1}(\alpha^j) Q\big).$$

In an analysis of $\mathcal{B}$'s advantage, we note that it only fails in providing a consistent simulation because one of the following independent events:

E_1: $\mathcal{A}$ does not choose to be challenged on ID_ℓ.
E_2: a key extraction query is made on ID_ℓ.
E_3: $\mathcal{B}$ aborts in a $\mathsf{Sign/Encrypt}$ query because of a collision on H_2.
E_4: $\mathcal{B}$ rejects a valid ciphertext at some point of the game.

We clearly have $\Pr[\neg E_1] = 1/q_{h_1}$ and we know that $\neg E_1$ implies $\neg E_2$. We also already observed that $\Pr[E_3] \leq q_{se}(q_{se} + q_{h_2})/2^k$ and $\Pr[E_4] \leq q_{dv}/2^k$. We thus find that

$$\Pr[\neg E_1 \wedge \neg E_3 \wedge \neg E_4] \geq \frac{1}{q_{h_1}} \left(1 - q_{se}\frac{q_{se} + q_{h_2}}{2^k}\right) \left(1 - \frac{q_{dv}}{2^k}\right).$$

We obtain the announced bound by noting that $\mathcal{B}$ selects the correct element from L_2 or L_3 with probability $1/(2q_{h_2} + q_{h_3})$. Its workload is dominated by $O(q_{h_1}^2)$ multiplications in the preparation phase, $O(q_{se}+q_{dv})$ pairing calculations and $O(q_{dv}q_{h_2})$ exponentiations in $\mathbb{G}_T$ in its emulation of the $\mathsf{Sign/Encrypt}$ and $\mathsf{Decrypt/Verify}$ oracles. □

C Proof of Theorem 3

Proof. The proof is almost similar to the one of theorem 1. Namely, it shows that a forger in the ESUF-IBSC-CMA game implies a forger in a chosen-message and *given* identity attack. Using the forking lemma [31,32], the latter is in turn shown to imply an algorithm to solve the q-Strong Diffie-Hellman problem. More precisely, queries to the $\mathsf{Sign/Encrypt}$ and $\mathsf{Decrypt/Verify}$ oracles are answered as in the proof of theorem 2 and, at the outset of the game, the simulator chooses public parameters in such a way that it can extract private keys associated to any identity but the one which is given as a challenge to the adversary. By doing so, thanks to the irreflexivity assumption, it is able to extract clear message-signature pairs from ciphertexts produced by the forger (as it knows the private key of the receiving identity ID_R^*). □

D Proof of Theorem 4

Proof. The simulator is the same as in theorem 2 with the following differences (recall that senders' identities are provided as inputs to H_2).

- H_2-queries on input (ID_S, M, r): $\mathcal{B}$ returns the previously defined value if it exists and a random $h_2 \overset{R}{\leftarrow} \mathbb{Z}_p^*$ otherwise. To anticipate subsequent $\mathsf{Decrypt/Verify}$ requests, $\mathcal{B}$ simulates oracle H_3 to obtain $h_3 = H_3(r) \in \{0,1\}^{n+n_0}$ (where n_0 is the maximum length of identity strings) and stores $(\mathsf{ID}_S, M, r, h_2, c = (M\|\mathsf{ID}_S) \oplus h_3, \gamma = r \cdot e(G_1, G_2)^{h_2})$ in list L_2.

- Decrypt/Verify queries: given a ciphertext $\sigma = \langle c, S, T\rangle$ and a receiver's identity $\mathsf{ID}_R = \mathsf{ID}_\nu$, we assume that $\nu = \ell$ because otherwise $\mathcal{B}$ knows the receiver's private key. The simulator $\mathcal{B}$ does not know the sender's identity ID_S but knows that $\mathsf{ID}_S \neq \mathsf{ID}_\nu$. It also knows that, for the private key S_{ID_S}, $\log_{S_{\mathsf{ID}_S}}(\psi^{-1}(S) - hS_{\mathsf{ID}_S}) = \log_{\psi(Q_{\mathsf{ID}_\nu})}(T)$, and hence
$$e(T, S_{\mathsf{ID}_S}) = e(\psi(Q_{\mathsf{ID}_\nu}), \psi^{-1}(S) - hS_{\mathsf{ID}_S}), \tag{5}$$
where $h = H_2(\mathsf{ID}_S, M, r)$ is the hash value obtained in the Sign/ Encrypt algorithm and $Q_{\mathsf{ID}_\nu} = I_\nu G_2 + Q_{pub}$. The query is handled by searching through list L_2 for entries of the form $(\mathsf{ID}_{S,i}, M_i, r_i, h_{2,i}, c, \gamma_i)$ indexed by $i \in \{1, \ldots, q_{h_2}\}$. If none is found, the ciphertext is rejected. Otherwise, each one of these entries for which $\mathsf{ID}_{S,i} \neq \mathsf{ID}_\nu$ is further examined by checking whether $\gamma_i = e(S, H_1(\mathsf{ID}_{S,i})Q + Q_{pub})$ and
$$e(T, S_{\mathsf{ID}_{S,i}})/e(S, Q_{\mathsf{ID}_\nu}) = e(\psi(Q_{\mathsf{ID}_\nu}), S_{\mathsf{ID}_{S,i}})^{-h_{2,i}}. \tag{6}$$
(at most $3q_{h_2} + 1$ pairings and q_{h_2} exponentiations must be computed), meaning that equation (5) is satisfied and that the ciphertext contains a valid message signature pair if both relations hold. If $\mathcal{B}$ detects an index $i \in \{1, \ldots, q_{h_2}\}$ satisfying them, the matching pair $(M_i, \langle h_{2,i}, S\rangle)$ is returned. Otherwise, σ is rejected and such a wrong rejection again occurs with an overall probability smaller than $q_{dv}/2^k$. □

E The Kurosawa-Heng Identity-Based Signature

We describe here the IBS scheme that can be derived from a modification of the Kurosawa-Heng [25] identity-based identification scheme using the Fiat-Shamir heuristic [20].

Setup and **Keygen** are the same as in our scheme described in section 3. The public parameters are
$$\mathsf{params} := \{\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, P, Q, g, Q_{pub}, e, \psi, H_1, H_2\}.$$
We also define $Q_{\mathsf{ID}} = H_1(\mathsf{ID})Q + Q_{pub}$.

Sign: to sign a message $M \in \{0,1\}^*$, the signer does the following:

1. picks $x \overset{R}{\leftarrow} \mathbb{Z}_p^*$ and computes $r = e(P, Q_{\mathsf{ID}})^x \in \mathbb{G}_T$,
2. sets $h = H_2(M, r) \in \mathbb{Z}_p^*$,
3. computes $S = xP + hS_{\mathsf{ID}}$.

The signature on M is $\sigma = (h, S) \in \mathbb{Z}_p^* \times \mathbb{G}_1$.

Verify: a signature $\sigma = (h, S)$ on a message M is accepted iff
$$h = H_2(M, e(S, Q_{\mathsf{ID}})g^{-h}).$$

The above IBS can be proven secure under the q-Strong Diffie-Hellman assumption. Even in its optimized version where $e(P, H_1(\mathsf{ID})Q + Q_{pub})$ is pre-computed by the signer, its signature generation algorithm happens to be slightly more expensive than our scheme's one which requires a simple scalar multiplication at step 3.