Boneh-Gentry-Hamburg's Identity-based Encryption Schemes Revisited

Ferucio Laurenţiu Ţiplea, George Teşeleanu, Sorin Iftene, and Anca-Maria Nica

Abstract

We revise Boneh-Gentry-Hamburg's identity-based encryption schemes and we show that we can renounce to the use of pseudo-random functions. We then prove IND-ID-CPA and ANON-IND-ID-CPA security of these schemes by showing that the advantage of any efficient adversary against these schemes is less than or equal to the quadratic residuosity advantage of some efficient adversary against the RSA generator. This greatly improves the existing upper bounds (being probably the tightest upper bound).

I. Introduction and Preliminaries

Identity-based cryptography was proposed in 1984 by Adi Shamir [5] who formulated its basic principles. The first *identity-based encryption* (IBE) scheme was proposed by Boneh and Franklin [2], being based on bilinear maps. Shortly, Cocks proposed another IBE scheme based on the standard quadratic residuosity (QR) problem modulo an RSA composite n. Cocks' scheme encrypts a bit by two integers modulo n such that the bit is recovered as the Jacobi symbol of one of these two integers together with the private key. Although the scheme is very elegant and quite fast, its main disadvantage is the ciphertext expansion: a bit of message requires $2\log n$ bits of ciphertext. In [3], Boneh, Gentry, and Hamburg proposed another two IBE schemes related to Cocks' scheme, with short ciphertexts. The first scheme, named $BasicIBE$, is IND-ID-CPA secure in the random oracle model under the QR assumption, while the second one, named $AnonIBE$, is ANON-IND-ID-CPA secure in the standard model under the interactive QR assumption. Both security results are obtained by providing upper bounds on the advantage of an efficient adversary against the corresponding IBE scheme.

Ferucio Laurenţiu Ţiplea and George Teşeleanu and Sorin Iftene and Anca-Maria Nica are with the Department of Computer Science, "Alexandru Ioan Cuza" University of Iaşi, Iaşi, Romania (e-mail: ferucio.tiplea@uaic.ro, george.teseleanu@info.uaic.ro, siftene@info.uaic.ro, anca.nica@info.uaic.ro).

In this paper we revise the $BasicIBE$ and $AnonIBE$ schemes and show that we can avoid the use of pseudo-random functions. We then prove IND-ID-CPA and ANON-IND-ID-CPA security of these schemes in the random oracle model, by showing that the advantage of any efficient adversary against these schemes is less than or equal to the QR advantage of some efficient adversary against the RSA generator. This also greatly improve the upper bounds on the security results in [3]. In fact, the upper bound we establish is probably the tightest one.

The rest of this section recalls basic concepts and notations that will be used in our paper (for details, the reader is referred to [3]).

a) Identity-based encryption: An IBE scheme consists of four probabilistic polynomial-time (PPT) algorithms: $Setup$, $Extract$, $Encrypt$, and $Decrypt$. The first one takes as input a security parameter and outputs the system public parameters together with a master key. The $Extract$ algorithm takes as input an identity ID together with the public parameters and the master key and outputs a private key associated to ID. The $Encrypt$ algorithm, starting with a message m, an identity ID, and the public parameters, encrypts m into some ciphertext c (the encryption key is ID or some binary string derived from ID). The last algorithm decrypts c into m by using the private key associated to ID.

The ANON-IND-ID-CPA security of an IBE scheme $\mathcal{S}$ is formulated by means of the following *Game* between a challenger and an adversary $\mathcal{A}$:

$Setup$**:** The challenger takes a security parameter λ and runs $Setup(\lambda)$. It gives the adversary $\mathcal{A}$ the resulting system parameters PP, while keeping the master key msk to itself;

$Phase\ 1$**:** The adversary $\mathcal{A}$ issues a finite number of adaptive *extraction queries* by sending for each query an identity ID. In response, the challenger runs the $KeyGen$ algorithm to generate the private key corresponding to ID and sends it to $\mathcal{A}$;

$Challenge$**:** Once the adversary decided that Phase 1 is over, it outputs two pairs (ID_0, m_0) and (ID_1, m_1) consisting of two equal length plain-texts m_0 and m_1 and two identities ID_0 and ID_1 that did not appear in any query in Phase 1. The challenger picks a random bit $b \in \{0, 1\}$ and computes and sends $c^* = Encrypt(PP, ID_b, m_b)$ as a challenge to $\mathcal{A}$;

$Phase\ 2$**:** The adversary issues more adaptive queries like in Phase 1, but with the constraint that each queried ID must be different than ID_0 and ID_1;

$Guess$: The adversary outputs a guess $b' \in \{0, 1\}$ and wins the game if $b = b'$.

The *advantage* of $\mathcal{A}$ in attacking $\mathcal{S}$ is $Adv_{\mathcal{A},\mathcal{S}}(\lambda) = |P(b = b') - 1/2|$, where $P(b = b')$ is the probability that $b = b'$ (computed over the random bits used by the challenger and the adversary

$\mathcal{A}$). An IBE scheme $\mathcal{S}$ is *ANON-IND-ID-CPA secure* if for any PPT adversary $\mathcal{A}$, $Adv_{\mathcal{A},\mathcal{S}}(\lambda)$ is negligible. If we consider $ID_0 = ID_1$ in the above game, we obtain the concept of *IND-ID-CPA security*.

b) Jacobi symbols and the QR assumption: The Jacobi symbol of an integer a modulo an integer n is denoted by $\left(\frac{a}{n}\right)$. J_n stands for the set of integers in $\mathbb{Z}_n^*$ whose Jacobi symbol is 1, QR_n denotes the set of quadratic residues in $\mathbb{Z}_n^*$, and $SQRT_n(a)$ is the set of square roots modulo n of a. $\mathbb{Z}_n[x]$ is the ring of polynomials over $\mathbb{Z}_n$.

The *QR advantage* of an adversary $\mathcal{A}$ against an RSA generator $RSAgen(\lambda)$ is denoted by $QRAdv_{\mathcal{A},RSAgen}(\lambda)$ (λ is a security parameter). If this advantage is negligible for all adversaries $\mathcal{A}$, we say that the *QR assumption holds for* $RSAgen$. Given a pseudorandom function (PRF) F, $PRFAdv_{\mathcal{A},F}$ stands for the *PRF advantage of* $\mathcal{A}$ *against* F. F is *secure* if $PRFAdv_{\mathcal{A},F}$ is negligible for all $\mathcal{A}$.

II. Boneh-Gentry-Hamburg's IBE Schemes

Cocks' IBE scheme [4] encrypts a bit $m \in \{-1, 1\}$ by two integers c_1 and c_2 such that either the Jacobi symbol of $(c_1 + 2r)$ or the Jacobi symbol of $(c_2 + 2r)$ modulo an RSA composite n is m, where r is the private key computed from the identity of the decryptor. The scheme is IND-ID-CPA secure in the random oracle model under the QR assumption.

Despite its elegance, Cocks' scheme produces large ciphertexts: $2 \log n$ bits are used to encrypt just one bit. Moreover, it is not anonymous [1]. In 2007, Boneh, Gentry and Hamburg proposed two space efficient IBE schemes related to Cocks' scheme, whose security is similarly based on the QR problem in the random oracle model, one of them being additionally anonymous [3]. These schemes are based on associated polynomials as defined below (our approach tries to capture the essence of using such polynomials and, therefore, it is slightly different than the one in [3]).

Definition 2.1: Let n be a positive integer, $a, S \in \mathbb{Z}_n^*$, and $f, g \in \mathbb{Z}_n[x]$.

1) We say that (f, g) is a pair of (a, S)-*associated polynomials* if
$$\left(\frac{f(r)}{n}\right) = \left(\frac{g(s)}{n}\right)$$
for all $r \in SQRT_n(a)$ and $s \in SQRT_n(S)$, whenever $a, S \in QR_n$.
2) We say that f is a-*secure* if $\left(\frac{f(r)}{n}\right)$ is uniformly distributed in $\{-1, 1\}$ when r is uniformly chosen from $SQRT_n(a)$, whenever $a \in QR_n$.

3) We say that (f, g) is a pair of *a-secure* (*S-secure*) *(a, S)-associated polynomials* if (f, g) is a pair of (a, S)-associated polynomials and f (g) is a-secure (S-secure).

We are now able to introduce the first scheme proposed in [3] and called $BasicIBE$.

The $BasicIBE$ scheme [3]

% In this scheme, $\mathcal{D}$ is an unspecified deterministic algorithm that on input (n, a, S) outputs a % pair (f, g) of (a, S)-associated polynomials, where n is a positive integer and $a, S \in \mathbb{Z}_n^*$.

% Moreover, (f, g) is a-secure when $a \in QR_n$ and $S \in J_n \setminus QR_n$.

$Setup(\lambda)$**:** Generate $(p, q) \leftarrow RSAgen(\lambda)$, compute $n = pq$, generate $e \in J_n \setminus QR_n$, and choose a hash function $h : \{0,1\}^* \times \{1, \ldots, \ell\} \to J_n$ for some integer $\ell \geq 1$. Output the public parameters $PP = (n, e, h)$; the master key $msk = (p, q, K)$ is the factorization of n together with a random key K of some pseudo-random function $F_K : \{0,1\}^* \times \{1, \ldots, \ell\} \to \{0, 1, 2, 3\}$ (F_K chooses one of the four square roots of $h(ID, i)$ or $eh(ID, i)$, depending on which of them is a quadratic residue);

$Extract(msk, ID)$**:** For each $j \in \{1, \ldots, \ell\}$, let $a_j = h(ID, j)$ and $i_j = F_K(ID, j)$. If r_0, r_1, r_2, r_3 is a fixed total ordering of the square roots of a_j or ea_j (depending on which of them is a quadratic residue), then the private key is $r = (r_{i_1}, \ldots, r_{i_\ell})$;

$Encrypt(PP, ID, m)$**:** Assume $m = m_1 \cdots m_\ell \in \{-1, 1\}^\ell$ is the ℓ-bit sequence to be encrypted. The encryption process is as follows:

- Generate at random $s \in \mathbb{Z}_n^*$ and set $S = s^2 \bmod n$;
- For $j := 1$ to ℓ do
  - Compute $a_j = h(ID, j)$;
  - Compute $(f_j, g_j) = \mathcal{D}(n, a_j, S)$ and $(\bar{f}_j, \bar{g}_j) = \mathcal{D}(n, ea_j, S)$;
  - Compute $c_j = m_j \cdot \left(\frac{g_j(s)}{n}\right)$ and $\bar{c}_j = m_j \cdot \left(\frac{\bar{g}_j(s)}{n}\right)$;
- Return $(c, \bar{c}, S)$, where $c = c_1 \cdots c_\ell$ and $\bar{c} = \bar{c}_1 \cdots \bar{c}_\ell$;

$Decrypt((c, \bar{c}, S), r)$**:** The decryption process is as follows:

- For $j := 1$ to ℓ do
  - Compute $a_j = h(ID, j)$;
  - If $a_j \in QR_n$ then $a'_j = a_j$ else $a'_j = ea_j$;
  - Compute $(f'_j, g'_j) = \mathcal{D}(n, a'_j, S)$;
  - Compute $m_j = c_j \cdot \left(\frac{f'_j(r_{i_j})}{n}\right)$;
- Return $m = m_1 \cdots m_\ell$.

Definition 2.1(1) guarantees the soundness of decryption. As with respect to security, the following result proved in [3] shows that $BasicIBE$ is IND-ID-CPA secure in the random oracle model under the QR assumption.

Theorem 2.1 ([3]): For any efficient IND-ID-CPA adversary $\mathcal{A}$ against BasicIBE, there exist efficient algorithms $\mathcal{B}_1$ and $\mathcal{B}_2$, whose running time is about the same as that of $\mathcal{A}$, such that

$$IBEAdv_{\mathcal{A},BasicIBE}(\lambda) \leq PRFAdv_{\mathcal{B}_1,F}(\lambda) + 2 \cdot QRAdv_{\mathcal{B}_2,RSAgen}(\lambda),$$

provided that h is modeled as a random oracle, the QR assumption holds for $RSAgen$, and F is a secure pseudo-random function.

Remark 2.1: A few words about the inequality in Theorem 2.1 are in order. The proof of this theorem as it is in [3] exploits the fact that $\mathcal{D}$ outputs pairs of (a_j, S)-associated polynomials that are a_j-secure when $a_j \in QR_n$ and $S \in J_n \setminus QR_n$, for all $1 \leq j \leq \ell$ (we use the notations in the $BasicIBE$ scheme). According to this, the initial IND-ID-CPA game is successively changed into another game where the challenge ciphertext is created by decrypting the message (that is, by encrypting it by f's instead of g's). In order to have $a_j \in QR_n$ and $S \in J_n \setminus QR_n$, the QR assumption is used two times, which gives rise to the factor $2 \cdot QRAdv_{\mathcal{B}_2,RSAgen}(\lambda)$. Moreover, to ensure that f_j (and $\bar{f}_j$) is a_j-secure (ea_j-secure), the PRF F_k is replaced by a truly random function, and this gives rise to the factor $PRFAdv_{\mathcal{B}_1,F}(\lambda)$.

Remark 2.2: We emphasize that the $BasicIBE$ scheme is an abstract IBE scheme because no concrete algorithm $\mathcal{D}$ to compute (a, S)-associated polynomials is presented. In [3], the method proposed to construct such polynomials is based on the congruence $QC_n(a, S)$ given by

$$ax^2 + Sy^2 \equiv 1 \bmod n, \tag{1}$$

where $n = pq$ is an RSA modulus and $a, S \in \mathbb{Z}_n^*$.

Any solution (x_0, y_0) to $QC_n(a, S)$ gives rise to two polynomials f and g

$$\begin{aligned} f(r) &= x_0 r + 1 \bmod n \\ g(s) &= 2(y_0 s + 1) \bmod n \end{aligned}$$

that are (a, S)-associated. Moreover, (f, g) is a-secure when $a \in QR_n$ and $S \in J_n \setminus QR_n$ (see Lemma 3.3 in [3]).

The proof of Theorem 2.1 in [3] exploits the fact that $\mathcal{D}$ outputs pairs of (a, S)-associated polynomials that are a-secure when $a \in QR_n$ and $S \in J_n \setminus QR_n$.

If we assume that $\mathcal{D}$ outputs pairs of (a, S)-associated polynomials that are S-secure when $S \in QR_n$ and $a \in J_n \setminus QR_n$, then we are able to improve the upper bound in Theorem 2.1. Moreover, there will be no need for a pseudo-random function because in the security game the challenger encrypts the message by g's as it is specified in the scheme and not by f's as in the proof of Theorem 2.1 (see [3] and Remark 2.1 for details). Therefore, we slightly change $BasicIBE$ into $BasicIBE'$ as follows.

The $BasicIBE'$ scheme

% In this scheme, $\mathcal{D}$ is an unspecified deterministic algorithm that on input (n, a, S) outputs a % pair (f, g) of (a, S)-associated polynomials, where n is a positive integer and $a, S \in \mathbb{Z}_n^*$.
% Moreover, (f, g) is S-secure when $S \in QR_n$ and $a \in J_n \setminus QR_n$.

1) $Setup(\lambda)$ outputs the public parameters $PP = (n, e, h)$ and the master key $msk = (p, q)$ exactly as in the $BasicIBE$ scheme, except that the PRF F is not required;
2) $Extract(msk, ID)$ chooses a square root r_{i_j} of $h(ID, j)$ or $eh(ID, j)$ depending on which of them is a quadratic residue, for all $1 \leq j \leq \ell$, and returns $r = (r_{i_1}, \ldots, r_{i_\ell})$;
3) $Encrypt(PP, ID, m)$ and $Decrypt((c, \bar{c}, S), r)$ are unchanged.

The security of this new IBE scheme is settled by the following theorem, which shows that the scheme is IND-ID-CPA secure in the random oracle model under the QR assumption.

Theorem 2.2: For any efficient IND-ID-CPA adversary $\mathcal{A}$ against the $BasicIBE'$ scheme, there exists an efficient algorithm $\mathcal{B}$, whose running time is about the same as that of $\mathcal{A}$, such that

$$IBEAdv_{\mathcal{A},BasicIBE'}(\lambda) \leq QRAdv_{\mathcal{B},RSAgen}(\lambda),$$

provided that h is modeled as a random oracle and the QR assumption holds for $RSAgen$.

Proof. Let $\mathcal{A}$ be an adversary against the $BasicIBE'$ scheme. We present the proof as a sequence of games, *Game 0,...,Game 3*, and we let $P(G_i)$ denote the probability that the adversary wins *Game i*, for all $0 \leq i \leq 3$.

Game 0. This is the IND-ID-CPA game defined in Section I, between $\mathcal{A}$ and a challenger implementing the $BasicIBE'$ scheme. Moreover, it is assumed that $h : \{0,1\}^* \times \{1, \ldots, \ell\} \to J_n$ is a random oracle chosen at random by the challenger from the set of all such functions, and $\mathcal{A}$ is allowed to query it at arbitrary points. Therefore,

$$IBEAdv_{\mathcal{A},BasicIBE'}(\lambda) = |P(G_0) - 1/2| \tag{2}$$

Recall that the challenger keeps the master key msk.

Game 1. In this game, the random oracle h is changed as follows. For an arbitrary identity ID and a point $1 \leq j \leq \ell$, the challenger does the following:

- randomly choose $b_j \in \{0,1\}$ and $v_j \in J_n \setminus QR_n$ (recall that the challenger knows msk and, therefore, he can choose v_j in this way by using the factorization of n);
- set $h(ID, j) = e^{b_j} \cdot v_j \bmod n$.

It is clear that $e^{b_j} \cdot v_j \bmod n$ is a random integer in J_n and, therefore, the challenger implements a random function h as in *Game 0*.

When the adversary queries a private key for ID, the challenger answers with a square root of ev_j, for all $1 \leq j \leq \ell$, as in *Game 0* (remark that $h(ID, j)$ is either v_j or ev_j and, in the former case, $eh(ID, j) = ev_j$). Thus we have

$$P(G_0) = P(G_1) \tag{3}$$

Game 2. In this game, the challenger chooses e in QR_n instead of $J_n \setminus QR_n$. Since this is the only change between *Game 1* and *Game 2*, there exists an algorithm $\mathcal{B}$, whose running time is about the same as that of $\mathcal{A}$, such that

$$|P(G_1) - P(G_2)| = QRAdv_{\mathcal{B},RSAgen}(\lambda) \tag{4}$$

We notice that, according to the way h is implemented by challenger, we have both $h(ID, j)$ and $eh(ID, j)$ in $J_n \setminus QR_n$, for all ID and j. As an effect, each of the Jacobi symbols $\left(\frac{g_j(s)}{n}\right)$ and $\left(\frac{\bar{g}_j(s)}{n}\right)$ used to encrypt the bit m_j is uniformly distributed in $\{-1, 1\}$, for all $1 \leq j \leq \ell$ ((f_j, g_j) is a pair of S-secure (a_j, S)-associated polynomials and $(\bar{f}_j, \bar{g}_j)$ is a pair of S-secure (ea_j, S)-associated polynomials).

Game 3. We change *Game 2* in order to make the challenge ciphertext independent of the challenge bit b. Thus, the challenger randomly chooses $s \in \mathbb{Z}_n^*$. Then, for each $1 \leq j \leq \ell$ the challenger randomly generates a bit z_j and computes c_j and $\bar{c}_j$ by

$$c_j = z_j \cdot \left(\frac{g_j(s)}{n}\right) \quad \text{and} \quad \bar{c}_j = z_j \cdot \left(\frac{\bar{g}_j(s)}{n}\right)$$

Clearly,

$$P(G_2) = P(G_3) \tag{5}$$

Moreover, $P(G_3) = 1/2$ because we encrypt a random message that is independent of the challenge bit. Combining this with (2), (3), (4), and (5), we obtain the theorem. ■

Remark 2.3: The algorithm $\mathcal{D}$ in the $BasicIBE'$ scheme can be instantiated exactly as in Remark 2.1 because the pairs (f, g) obtained from solutions to $QC_n(a, S)$ are S-secure as well when $a \in J_n \setminus QR_n$ and $S \in QR_n$ (the proof is similar to the one in Lemma 3.3 in [3]: just replace the role of a by the role of S, and vice versa).

Remark 2.4: If we instantiate the algorithm $\mathcal{D}$ in $BasicIBE$ (or $BasicIBE'$) as in Remark 2.1, then the encryptor must find solutions to 2ℓ congruences of the form $QC_n(a, S)$, while the decryptor needs solutions to ℓ of these congruences. Boneh, Gentry, and Hamburg [3] have proposed the following *Combining Lemma* in order to reduce the number of congruences to be solved by the encryptor:

- If $(x_1, y_1) \in \mathbb{Z}_n^2$ is a solution to the congruence $QC_n(a_1, S)$ and $(x_2, y_2) \in \mathbb{Z}_n^2$ is a solution to the congruence $QC_n(a_2, S)$, then $(x_{1,2}, y_{1,2}) \in \mathbb{Z}_n^2$ is a solution to the congruence $QC_n(a_1 a_2, S)$, where

$$x_{1,2} = \frac{x_1 x_2}{S y_1 y_2 + 1} \bmod n \quad \text{and} \quad y_{1,2} = \frac{y_1 + y_2}{S y_1 y_2 + 1} \bmod n, \tag{6}$$

  provided that $(S y_1 y_2 + 1, n) = 1$.

By this result, the encryptor first finds solutions to $QC_n(e, S)$ and $QC_n(a_j, S)$, for all $1 \leq j \leq \ell$, and then combines these solutions to obtain solutions to $QC_n(e a_j, S)$, for all $1 \leq j \leq \ell$. Therefore, the encryptor needs to find solutions to only $\ell + 1$ congruences.

It is to be remarked that, by reducing the number of congruences to be solved in this way, the inequality in Theorem 2.2 still holds.

The $BasicIBE$ ($BasicIBE'$) scheme is IND-ID-CPA secure but it is not anonymous [3] because there are instances of the algorithm $\mathcal{D}$ for which anyone can test which identity created a given ciphertext. This scheme can be transformed into an anonymous one if the associated polynomials are chosen in a different way.

Definition 2.2: Let n be a positive integer, $a, e, S \in \mathbb{Z}_n^*$, and $f, \bar{f}, g, \tau \in \mathbb{Z}_n[x]$.

1) We say that $(f, \bar{f}, g, \tau)$ is a 4-tuple of (a, ea, S)-*associated polynomials* if (f, g) are (a, S)-associated, $(\bar{f}, g \cdot \tau)$ are (ea, S)-associated, and τ is independent of a.
2) We say that $(f, \bar{f}, g, \tau)$ is a 4-tuple of (a, ea)-*secure* (a, ea, S)-*associated polynomials* if it is a tuple of (a, ea, S)-associated polynomials, and f is a-secure or $\bar{f}$ is ea-secure.
3) We say that $(f, \bar{f}, g, \tau)$ is a 4-tuple of S-*secure* (a, ea, S)-*associated polynomials* if it is a tuple of (a, ea, S)-associated polynomials and g and τ are S-secure.

A few remarks are in order about (a, ea, S)-associated polynomials. The polynomial g is used for encryption, and f for decryption, exactly as in the $BasicIBE$ scheme. The encryption by $\bar{g}$ in the $BasicIBE$ scheme is replaced now by sending the Jacobi symbol of $\tau(s)$. This is because $g(s)\tau(s)$ will play the role of $\bar{g}(s)$ and the decryption will be performed by $\bar{f}$ (remark that $(\bar{f}, g \cdot \tau)$ are (ea, S)-associated). The anonymity of the scheme is obtained due to the fact that τ is independent of a.

In the scheme below, $\mathcal{D}'$ is a deterministic algorithm that on input (n, a, e, S), where n is a positive integer and $a, e, S \in \mathbb{Z}_n^*$, outputs a 4-tuple $(f, \bar{f}, g, \tau)$ of (a, ea, S)-associated polynomials. Moreover, $(f, \bar{f}, g, \tau)$ is (a, ea)-secure when $e, S \in J_n \setminus QR_n$.

The $AnonIBE$ scheme [3]

% In this scheme, $\mathcal{D}'$ is an unspecified deterministic algorithm that on input (n, a, e, S), where
% n is a positive integer and $a, e, S \in \mathbb{Z}_n^*$, outputs a 4-tuple $(f, \bar{f}, g, \tau)$ of (a, ea, S)-associated
% polynomials. Moreover, $(f, \bar{f}, g, \tau)$ is (a, ea)-secure when $e, S \in J_n \setminus QR_n$.

$Setup(\lambda)$**:** the same as in the $BasicIBE$ scheme;

$Extract(msk, ID)$**:** the same as in the $BasicIBE$ scheme;

$Encrypt(PP, ID, m)$**:** Assume $m = m_1 \cdots m_\ell$ is the ℓ-bit sequence to be encrypted. Generate at random $s \in \mathbb{Z}_n^*$, set $S = s^2 \bmod n$, and let $a_j = h(ID, j)$ and $(f_j, \bar{f}_j, g_j, \tau) = \mathcal{D}'(n, a_j, e, S)$, for all $1 \leq j \leq \ell$. Encrypt then m by $(c, \bar{c}, S)$, where $\bar{c} = \left(\frac{\tau(s)}{n}\right)$, $c = c_1 \cdots c_\ell$, and $c_j = m_j \cdot \left(\frac{g_j(s)}{n}\right)$ for all $1 \leq j \leq \ell$;

$Decrypt((c, \bar{c}, S), r)$**:** For each $1 \leq j \leq \ell$, let $a_j = h(ID, j)$ and $(f_j, \bar{f}_j, g_j, \tau) = \mathcal{D}'(n, a_j, e, S)$. If a_j is a quadratic residue modulo n, then compute $m_j = c_j \cdot \left(\frac{f_j(r_j)}{n}\right)$; otherwise, compute $m_j = c_j \cdot \bar{c} \cdot \left(\frac{\bar{f}_j(r_j)}{n}\right)$. Output $m = m_1 \cdots m_\ell$.

The correctness of the scheme follows easily from Definition 2.2. As with respect to security, it was shown in [3] that the scheme is ANON-IND-ID-CPA secure in the standard model under the *interactive QR* (IQR) assumption, namely that the QR problem is hard in the presence of a hash square root oracle.

Theorem 2.3 ([3]): For any efficient ANON-IND-ID-CPA adversary $\mathcal{A}$ against the $AnonIBE$ scheme, there exist efficient algorithms $\mathcal{B}_1$ and $\mathcal{B}_2$, whose running time is about the same as that of $\mathcal{A}$, such that

$$IBEAdv_{\mathcal{A}, AnonIBE}(\lambda) \leq PRFAdv_{\mathcal{B}_1, F}(\lambda) + IQRAdv_{\mathcal{B}_2, (RSAgen, h)}(\lambda),$$

provided that the IQR assumption holds for $(RSAgen, h)$ and F is a secure pseudo-random function.

Remark 2.5: The inequality in Theorem 2.3 is obtained in a similar way to the one in Theorem 2.1. IQR is necessary both to answer to adversary's queries by means of its square root oracle, and to allow choosing S in $J_n \setminus QR_n$ instead of QR_n with the price of one $IQRAdv_{\mathcal{B}_2,(RSAgen,h)}(\lambda)$. The change from PRF to a truly random function in order to have $\left(\frac{f_j(r_j)}{n}\right)$ or $\left(\frac{\bar{f}_j(r_j)}{n}\right)$ uniform in $\{-1, 1\}$, induces the factor $PRFAdv_{\mathcal{B}_1,F}(\lambda)$.

Remark 2.6: The $AnonIBE$ scheme is abstract because no concrete algorithm $\mathcal{D}'$ is specified. However, it turns out that the quadratic congruences used to instantiate $BasicIBE$ can also be used to obtain tuples of (a, ea, S)-associated polynomials. More precisely, given n a positive integer and $a, e, S \in \mathbb{Z}_n^*$, and given (x_0, y_0) a solution to $QC_n(a, S)$ and (α, β) a solution to $QC_n(e, S)$, four polynomials f, $\bar{f}$, g, and τ are defined:

$$\begin{aligned} f(r) &= x_0 r + 1 \bmod n \\ g(s) &= 2(y_0 s + 1) \bmod n \\ \bar{f}(\bar{r}) &= \alpha x_0 \bar{r} + \beta y_0 S + 1 \bmod n \\ \tau(s) &= \beta s + 1 \bmod n. \end{aligned}$$

These polynomials are (a, ea, S)-associated. Moreover, they are (a, ea)-secure when $e, S \in J_n \setminus QR_n$ (see [3] for details).

If we assume that the deterministic algorithm $\mathcal{D}'$ in $AnonIBE$ outputs tuples of S-secure (a, ea, S)-associated polynomials when $S \in QR_n$ and $e \in J_n \setminus QR_n$, then we may use the same idea as in the case of $BasicIBE'$ to obtain a new scheme $AnonIBE'$ with a better security upper bound.

The $AnonIBE'$ scheme

% In this scheme, $\mathcal{D}'$ is an unspecified deterministic algorithm that on input (n, a, e, S), where % n is a positive integer and $a, e, S \in \mathbb{Z}_n^*$, outputs a 4-tuple $(f, \bar{f}, g, \tau)$ of (a, ea, S)-associated % polynomials. Moreover, $(f, \bar{f}, g, \tau)$ is S-secure when $S \in QR_n$ and $e \in J_n \setminus QR_n$.

1) *Setup* and *Extract* are as in the $BasicIBE'$ scheme;
2) *Encrypt* and *Decrypt* are as in the $AnonIBE$ scheme.

As with respect to the security of $AnonIBE'$, the following result shows that the scheme is ANON-IND-ID-CPA secure in the random oracle model under the QR assumption.

Theorem 2.4: For any efficient ANON-IND-ID-CPA adversary $\mathcal{A}$ against the $AnonIBE'$ scheme, there exists an efficient algorithm $\mathcal{B}$, whose running time is about the same as that of $\mathcal{A}$, such that

$$IBEAdv_{\mathcal{A},AnonIBE'}(\lambda) \leq QRAdv_{\mathcal{B},RSAgen}(\lambda),$$

provided that h is modeled as a random oracle and the QR assumption holds for $RSAgen$.

Proof. The proof follows the same line as the proof of Theorem 2.2, except that *Game 2* is now split into two games *Game 2.1* and *Game 2.2* in order to see more clearly how the challenge ciphertext is created. So, let $\mathcal{A}$ be an adversary against the $AnonIBE'$ scheme.

Game 0. This is the ANON-IND-ID-CPA game defined in Section I, between $\mathcal{A}$ and a challenger implementing the $AnonIBE'$ scheme. Similar to *Game 0* in the proof of Theorem 2.2 we obtain

$$IBEAdv_{\mathcal{A},AnonIBE'}(\lambda) = |P(G_0) - 1/2| \tag{7}$$

Game 1. In this game, the random oracle h is changed exactly as in *Game 1* in the proof of Theorem 2.2. Therefore,

$$P(G_0) = P(G_1) \tag{8}$$

Game 2.1. In this game, the challenger computes the ciphertext $(c, \bar{c}, S)$ as in the previous game, except that $\bar{c}$ is chosen uniformly at random from $\{-1, 1\}$ instead of computing it by $\bar{c} = \left(\frac{\tau(s)}{n}\right)$. The adversary $\mathcal{A}$ does not see any difference between this game and *Game 1* because τ is S-secure (remark that $S \in QR_n$ and $e \in J_n \setminus QR_n$). Therefore,

$$P(G_1) = P(G_{2.1}) \tag{9}$$

Game 2.2. This is similar to *Game 2* in the proof of Theorem 2.2. The challenger chooses e in QR_n instead of $J_n \setminus QR_n$. Since this is the only change between *Game 2.1* and *Game 2.2*, there exists an algorithm $\mathcal{B}$, whose running time is about the same as that of $\mathcal{A}$, such that

$$|P(G_{2.1}) - P(G_{2.2})| = QRAdv_{\mathcal{B},RSAgen}(\lambda) \tag{10}$$

According to the way h is implemented by challenger, we have $h(ID, j) \in J_n \setminus QR_n$, for all ID and j. As an effect, the Jacobi symbol $\left(\frac{g_j(s)}{n}\right)$ used to encrypt the bit m_j is uniformly distributed in $\{-1, 1\}$, for all $1 \leq j \leq \ell$ (g_j is S-secure). As $e \in QR_n$, we may not say that $\left(\frac{\tau(s)}{n}\right)$ is S-secure. However, this is not important now because starting with *Game 2.1* $\tau(s)$ is no longer used (its Jacobi symbol is replaced by a randomly chosen bit $\bar{c}$).

Game 3. This game is similar to *Game 3* in the proof of Theorem 2.2. The challenger randomly generates $\bar{c} \in \{-1, 1\}$ and $s \in \mathbb{Z}_n^*$. Then, for each $1 \leq j \leq \ell$, the challenger randomly generates a bit z_j and computes $c_j = z_j \cdot \left(\frac{g_j(s)}{n}\right)$. The ciphertext is $(c, \bar{c}, S)$. Clearly,

$$P(G_{2.2}) = P(G_3) \tag{11}$$

Moreover, $P(G_3) = 1/2$. Combining this with (7), (8), (9), (10), and (11), we obtain the theorem. ■

Remark 2.7: It is mentioned in [3] that the IQR assumption follows from the QR assumption where h is a full-domain hash function modeled as a random oracle. As a conclusion, $AnonIBE$ is secure in the standard model under the IQR assumption, and in the random oracle model under the QR assumption. However, if we follow the same proof line as in the proof of Theorem 2.1, the same upper bound is obtained for $IBEAdv_{\mathcal{A},AnonIBE}(\lambda)$ when we work in the random oracle model under the QR assumption. Therefore, the result in Theorem 2.4 is a consistent improvement of the result in Theorem 2.3.

Remark 2.8: The algorithm $\mathcal{D}'$ in the $AnonIBE'$ scheme can be instantiated exactly as in Remark 2.6 because the tuples $(f, \bar{f}, g, \tau)$ obtained from solutions to $QC_n(a, S)$ and $QC_n(e, S)$ are S-secure as well when $e \in J_n \setminus QR_n$ and $S \in QR_n$ (the proof is similar to the one in Lemma 3.3 in [3]: just replace the role of a by the role of S, and vice versa).

III. Conclusions

Boneh, Gentry, and Hamburg have proposed in [3] two IBE schemes related to Cocks' IBE scheme, called $BasicIBE$ and $AnonIBE$. The later one provides anonymity of identity in addition (Cocks' IBE scheme is not anonymous [1]). These two schemes are more space efficient than Cocks' IBE scheme (but less time efficient). Both of them use pseudo-random functions to choose the private key.

In this paper we have revisited the $BasicIBE$ and $AnonIBE$ schemes and we have shown that the pseudo-random functions can be removed from their description. Moreover, using a different proof approach, we proved that the scheme are secure by showing that the advantage of any efficient adversary against these schemes, in the random oracle model, is bounded from above by the QR advantage of some efficient adversary against the RSA generator. This greatly improves the existing upper bounds established in [3], being probably the tightest upper bound.

REFERENCES

[1] G. Ateniese and P. Gasti, "Universally anonymous IBE based on the quadratic residuosity assumption," in *Proceedings of the The Cryptographers' Track at the RSA Conference 2009 on Topics in Cryptology*, ser. CT-RSA '09. Berlin, Heidelberg: Springer-Verlag, 2009, pp. 32–47.

[2] D. Boneh and M. K. Franklin, "Identity-based encryption from the Weil pairing," in *Proceedings of the 21st Annual International Cryptology Conference on Advances in Cryptology*, ser. CRYPTO '01. London, UK, UK: Springer-Verlag, Aug. 2001, pp. 213–229.

[3] D. Boneh, C. Gentry, and M. Hamburg, "Space-efficient identity based encryption without pairings," in *Proceedings of the 48th Annual IEEE Symposium on Foundations of Computer Science*, ser. FOCS '07. Washington, DC, USA: IEEE Computer Society, 2007, pp. 647–657.

[4] C. Cocks, "An identity based encryption scheme based on quadratic residues," in *Proceedings of the 8th IMA International Conference on Cryptography and Coding*. London, UK, UK: Springer-Verlag, Dec. 2001, pp. 360–363.

[5] A. Shamir, "Identity-based cryptosystems and signature schemes," in *Proceedings of CRYPTO 84 on Advances in cryptology*. New York, NY, USA: Springer-Verlag New York, 1985, pp. 47–53.