In C.-M. Cheng *et al.*, Eds., *Public Key Cryptography – PKC 2016*, Part I, vol. 9614 of *Lecture Notes in Computer Science*, pp. 225-254, Springer, 2016.

Identity-Based Cryptosystems and Quadratic Residuosity

Marc Joye

Technicolor
175 S. San Antonio Road, Los Altos, CA 94022, USA
marc.joye@technicolor.com

Abstract. Three approaches are currently used for devising identity-based encryption schemes. They respectively build on pairings, quadratic residues (QR), and lattices. Among them, the QR-based scheme proposed by Cocks in 2001 is notable in that it works in standard RSA groups: its security relies on the standard quadratic residuosity assumption. But it has also a number of deficiencies, some of them have been subsequently addressed in follow-up works. Currently, one of the main limitations of Cocks' scheme resides in its apparent lack of structure. This considerably restricts the range of possible applications. For example, given two Cocks ciphertexts, it is unknown how to evaluate of a function thereof.
Cocks' scheme is believed to be non-homomorphic. This paper disproves this conjecture and proposes a constructive method for computing over Cocks ciphertexts. The discovery of the hidden algebraic structure behind Cocks encryption is at the core of the method. It offers a better understanding of Cocks' scheme. As a further illustration of the importance of the knowledge of the underlying structure, this paper shows how to anonymize Cocks ciphertexts without increasing their size or sacrificing the security.
Finally and of independent interest, this paper presents a simplified version of the abstract identity-based cryptosystem with short ciphertexts of Boneh, Gentry, and Hamburg.

Keywords: Public-key cryptography; identity-based encryption; Cocks' scheme; homomorphic encryption; anonymous encryption; public-key encryption with keyword search; quadratic residuosity.

1 Introduction

Identity-based cryptography is an extension of the public-key paradigm which was first put forward by Shamir [25]. As discussed in [20, Chapter 1], a major issue with public-key cryptography is the management of trust. Another issue to be dealt with is to recover the public key and accompanying certificate, verify it, and then only encrypt and send messages. Identity-based cryptography aims at solving these practical issues by simplifying the key management.

While identity-based signature schemes were quickly proposed (already in his 1984 paper, Shamir presented such a scheme), identity-based encryption (IBE)

schemes seem harder to develop and only came later. The first implementation of an IBE scheme was proposed by Desmedt and Quisquater [13] in 1986. It is however non-standard in the sense that it requires tamper-proof hardware for its security. The realization of a truly practical IBE scheme remained elusive until a breakthrough paper by Boneh and Franklin [7] in 2001, and concurrently by Sakai *et al.* [24]. The Boneh-Franklin IBE scheme makes use of bilinear maps. Its publication was quickly followed up by a large number of works. More recently, lattices were considered as a building block for constructing IBE schemes [16]. Again this gave rise to a number of follow-up works.

A totally different approach was described back in 2001 by Cocks in a short 4-page paper [12]. Cocks' IBE scheme only requires elementary mathematics. Encryption merely involves a couple of operations modulo an RSA modulus and the evaluation of Jacobi symbols. Its security rests on the standard quadratic residuosity assumption in the random oracle model. Despite its simplicity, Cocks' scheme received less attention from the research community, compared to the pairing-based or lattice-based constructions. We believe that this is mainly due to the apparent lack of structure behind Cocks' scheme. In this paper, we identify Cocks ciphertexts as elements of a certain algebraic group. This makes Cocks' scheme amenable to applications that were previously not possible. In particular, it can now be used in applications where computing over ciphertexts is required. Typical applications include electronic voting, auction systems, private information retrieval, or cloud computing.

Related work Since it appeared in 2001, a handful of variants of Cocks' IBE scheme have been proposed in the literature, aiming at enhancing certain features of the original scheme or offering extra properties.

Cocks' scheme is known not to be anonymous. The ciphertexts leak information about their recipient's identity. The anonymity aspect in Cocks-like cryptosystems was first considered by Di Crescenzo and Saraswat in [14] (and incidentally in [8]). Subsequently, Ateniese and Gasti [2] and, more recently, Clear *et al.* [11] proposed concurrent anonymous cryptosystems derived from Cocks' scheme. Table 1 in [11] gives a comparison of these two latter schemes. The scheme by Clear *et al.* features the best encryption and decryption times (i.e., 79 ms and 27 ms for a 128-bit message with a key-size of 1024 bits in their setting). The Ateniese-Gasti scheme is slower but has a smaller ciphertext expansion.

Cocks' scheme is mostly attractive when used with the hybrid encryption paradigm encrypting a short session key. Indeed messages are encrypted in a bit-by-bit fashion with Cocks' IBE scheme. It therefore looses its practicability when long messages need to be encrypted. The ciphertext expansion issue was addressed by Boneh, Gentry, and Hamburg [8]. They propose a space-efficient, anonymous IBE scheme based on the quadratic residuosity. However, the encryption in their scheme is time-consuming. Encryption time is quartic in the security parameter per message bit. Several possible trade-offs are discussed in [8, Section 5.3] and [19].

The work closest to ours is a recent paper by Clear, Hughes, and Tewari [10]. They develop a XOR-homomorphic variant of Cocks' scheme. This is elegantly achieved by seeing ciphertexts as elements (of the multiplicative group) in a certain quotient ring. The homomorphism property then naturally pops up as an application of the corresponding multiplication operation. A similar scheme was independently found —and generalized to higher power residue symbols— by Boneh, LaVigne, and Sabin [9]. As explained in [9], these schemes are however less efficient, bandwidth-wise, than the original Cocks' scheme.

Our contributions Motivated by the work of [10], we were interested in finding the exact algebraic structure behind Cocks' scheme. We quote from [23]:

> *"We believe that studying and understanding the mathematics that underlies the associated cryptosystems is a useful aid to better understand their properties and their security."*

Interestingly, the results of Rubin and Silverberg [23] (appearing earlier in [22]) were instrumental in our work. In a nutshell, we consider the torus $\mathbb{T}_1(\mathbb{F}_p) = \mathbb{F}_p^\times$ viewed as a 'degenerate' representation of the torus $\mathbb{T}_2(\mathbb{F}_p)$ where $\mathbb{F}_{p^2}$ is replaced with $\mathbb{F}_p(\delta)$ where $\delta \in \mathbb{F}_p^\times$. We then extend the setting modulo an RSA composite through Chinese remaindering. We show that the Cocks ciphertexts are squares in the so-obtained algebraic structure and form a quasi-group. The underlying group law yields the sought-after homomorphism. Compared to the approach in [10] there is no ciphertext expansion —the ciphertexts in [10] are twice longer. More importantly, it directly applies to Cocks' scheme. This is somewhat surprising. It points out that the original Cocks' IBE scheme is *inherently* homomorphic. In this regard, it shares similarities with the Goldwasser-Micali [public-key] encryption scheme [17]. We note that both schemes are semantically secure under the quadratic residuosity assumption and have comparable performance.

Another contribution is an anonymous variant of Cocks's scheme. Anonymous identity-based schemes are important cryptographic tools as they constitute the central building block for public-key encryption with keyword search (PEKS). The companion PEKS scheme derived from our anonymous IBE scheme is detailed in Appendix D. Compared to the earlier QR-based[1] scheme by Di Crescenzo and Saraswat [14], it reduces the size of the searchable ciphertexts by a typical factor of 2 without sacrificing the security.

Of independent interest, we present in Appendix E a simplified version of the abstract IBE system with short ciphertexts of Boneh, Gentry, and Hamburg [8].

2 Definitions and Notation

In this section, we review the classical notions of semantic security and of anonymity for identity-based encryption. We also formally present the quadratic residuosity assumption and a variant thereof.

[1] We note that the new security assumption introduced in [14] was later shown in [2] to be equivalent to the QR assumption.

2.1 Identity-based encryption

An *identity-based encryption scheme* [7] (or IBE in short) is defined as a tuple of four polynomial-time algorithms ($\texttt{SETUP}, \texttt{EXTRACT}, \texttt{ENCRYPT}, \texttt{DECRYPT}$):

Setup The setup algorithm $\texttt{SETUP}$ is a randomized algorithm that, taking a security parameter 1^κ as input, outputs the system parameters mpk together with the master secret key msk: $(\mathsf{mpk}, \mathsf{msk}) \xleftarrow{R} \texttt{SETUP}(1^\kappa)$. The message space is denoted by $\mathcal{M}$.

Key derivation The key derivation algorithm $\texttt{EXTRACT}$ takes as input an identity id and, using the master secret key msk, returns a secret key for the user with identity id: $\mathsf{usk} \leftarrow \texttt{EXTRACT}_{\mathsf{msk}}(\mathsf{id})$.

Encryption The encryption algorithm $\texttt{ENCRYPT}$ is a randomized algorithm that takes as input an identity id and a plaintext $m \in \mathcal{M}$, and returns a ciphertext C. We write $C \leftarrow \texttt{ENCRYPT}_{\mathsf{mpk}}(\mathsf{id}, m)$.

Decryption The decryption algorithm $\texttt{DECRYPT}$ takes as input secret key usk (corresponding to identity id) and a ciphertext C and returns the corresponding plaintext m or a special symbol $\perp$ indicating that the ciphertext is invalid. We write $m \leftarrow \texttt{DECRYPT}_{\mathsf{usk}}(C)$ if C is a valid ciphertext and $\perp \leftarrow \texttt{DECRYPT}_{\mathsf{usk}}(C)$ if it is not.

It is required that $\texttt{DECRYPT}_{\mathsf{usk}}(\texttt{ENCRYPT}_{\mathsf{mpk}}(\mathsf{id}, m)) = m$ for any identity id and all messages $m \in \mathcal{M}$, where $(\mathsf{mpk}, \mathsf{msk}) \xleftarrow{R} \texttt{SETUP}(1^\kappa)$ and $\mathsf{usk} \leftarrow \texttt{EXTRACT}_{\mathsf{msk}}(\mathsf{id})$.

2.2 Security notions

Semantic security The notion of *indistinguishability of encryptions* [17] captures a strong notion of data-privacy: The adversary should not learn anything about a plaintext given its encryption, beyond the length of the plaintext. The definitions for the public-key setting naturally extend to the identity-based paradigm. The standard definition is strengthened by allowing the adversary to issue chosen private-key extraction queries [7].

We view an adversary $\mathcal{A}$ as a pair $(\mathcal{A}_1, \mathcal{A}_2)$ of probabilistic algorithms. This corresponds to adversary $\mathcal{A}$ running in two stages. Upon receiving the system parameters mpk, in the "find" stage, algorithm $\mathcal{A}_1$ issues private-key extraction queries $\mathsf{id}_1, \ldots, \mathsf{id}_{n_1}$ and receives back the private key usk_i corresponding to identity id_i: $\mathsf{usk}_i \leftarrow \texttt{EXTRACT}_{\mathsf{msk}}(\mathsf{id}_i)$. The queries may be asked adaptively. Once the adversary decides not to make further oracle queries, it outputs a challenge identity id^* (with $\mathsf{id}^* \neq \mathsf{id}_i$, $1 \leq i \leq n_1$), two (different) equal-size messages m_0 and $m_1 \in \mathcal{M}$ (where $\mathcal{M}$ denotes the message space), and some state information s. In the "guess" stage, algorithm $\mathcal{A}_2$ receives a challenge ciphertext C which is the encryption of m_b for identity id^* where b is chosen uniformly at random in $\{0, 1\}$. Algorithm $\mathcal{A}_2$ can issue more private-key extraction queries $\mathsf{id}_{n_1+1}, \ldots, \mathsf{id}_{n_2}$; the only restriction is that $\mathsf{id}_i \neq \mathsf{id}^*$, $n_1 < i \leq n_2$. The goal of $\mathcal{A}_2$ is to guess the value of b from s and C. Formally, a public-key encryption

scheme is said *indistinguishable* (or *semantically secure*) if

$$\Pr\left[\begin{array}{l}(\mathsf{mpk},\mathsf{msk}) \xleftarrow{R} \mathtt{SETUP}(1^\kappa),\\ (\mathsf{id}^*, m_0, m_1, s) \leftarrow \mathcal{A}_1^{\mathtt{EXTRACT}_{\mathsf{msk}}(\cdot)}(\mathsf{mpk}),\ : \mathcal{A}_2^{\mathtt{EXTRACT}_{\mathsf{msk}}(\cdot)}(s, C) = b\\ b \xleftarrow{R} \{0,1\}, C \leftarrow \mathtt{ENCRYPT}_{\mathsf{mpk}}(\mathsf{id}^*, m_b)\end{array}\right] - \frac{1}{2}$$

is negligible in the security parameter for any polynomial-time adversary $\mathcal{A}$; the probability is taken over the random coins of the experiment according to the distribution induced by SETUP and over the random coins of the adversary.

Adversary $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ can encrypt any message of its choice, for any identity of its choice. In other words, the adversary can mount chosen-identity, chosen-plaintext attacks (ID-CPA). Hence, we write IND-ID-CPA the security notion achieved by a semantically secure identity-based encryption scheme.

Remark 1. When the message space is $\mathcal{M} = \{0,1\}$, the previous probability simplifies to

$$\Pr\left[\begin{array}{l}(\mathsf{mpk},\mathsf{msk}) \xleftarrow{R} \mathtt{SETUP}(1^\kappa),\\ (\mathsf{id}^*, s) \leftarrow \mathcal{A}_1^{\mathtt{EXTRACT}_{\mathsf{msk}}(\cdot)}(\mathsf{mpk}), \qquad : \mathcal{A}_2^{\mathtt{EXTRACT}_{\mathsf{msk}}(\cdot)}(s, C) = b\\ b \xleftarrow{R} \{0,1\}, C \leftarrow \mathtt{ENCRYPT}_{\mathsf{mpk}}(\mathsf{id}^*, b)\end{array}\right] - \frac{1}{2}\ .$$

Anonymity Analogously, the notion of *anonymity* captures a strong requirement about privacy: a ciphertext should not reveal the identity of the recipient. More formally, it is defined as a straightforward adaptation of key privacy [4] to the identity-based paradigm [1].

As before, we view an adversary $\mathcal{A}$ as a pair $(\mathcal{A}_1, \mathcal{A}_2)$ of probabilistic algorithms. In the "find" stage, algorithm $\mathcal{A}_1$ issues private-key extraction queries $\mathsf{id}_1, \ldots, \mathsf{id}_{n_1}$ and receives back the private key usk_i corresponding to identity id_i: $\mathsf{usk}_i \leftarrow \mathtt{EXTRACT}_{\mathsf{msk}}(\mathsf{id}_i)$. The queries may be asked adaptively. Once the adversary decides not to make further oracle queries, it outputs two (different) challenge identities id_0^* and id_1^* (with $\mathsf{id}_0^*, \mathsf{id}_1^* \neq \mathsf{id}_i$, $1 \leq i \leq n_1$), a message $m \in \mathcal{M}$, and some state information s. In the "guess" stage, algorithm $\mathcal{A}_2$ receives a challenge ciphertext C which is the encryption of m for identity id_b^* where b is chosen uniformly at random in $\{0,1\}$. Algorithm $\mathcal{A}_2$ can issue more private-key extraction queries $\mathsf{id}_{n_1+1}, \ldots, \mathsf{id}_{n_2}$; the only restriction is that $\mathsf{id}_i \neq \mathsf{id}_0^*, \mathsf{id}_1^*$, $n_1 < i \leq n_2$. The goal of $\mathcal{A}_2$ is to recover the value of b from s and C.

An IBE scheme is said to be *anonymous* if

$$\Pr\left[\begin{array}{l}(\mathsf{mpk},\mathsf{msk}) \xleftarrow{R} \mathtt{SETUP}(1^\kappa),\\ (\mathsf{id}_0^*, \mathsf{id}_1^*, m, s) \leftarrow \mathcal{A}_1^{\mathtt{EXTRACT}_{\mathsf{msk}}(\cdot)}(\mathsf{mpk}),\ : \mathcal{A}_2^{\mathtt{EXTRACT}_{\mathsf{msk}}(\cdot)}(s, C) = b\\ b \xleftarrow{R} \{0,1\}, C \leftarrow \mathtt{ENCRYPT}_{\mathsf{mpk}}(\mathsf{id}_b^*, m)\end{array}\right] - \frac{1}{2}$$

is negligible in the security parameter for any polynomial-time adversary $\mathcal{A}$; the probability is taken over the random coins of the experiment according to the distribution induced by SETUP and over the random coins of the adversary. We write ANO-ID-CPA the corresponding security notion achieved by an anonymous IBE scheme.

Semantic security and anonymity Of course, the goals of indistinguishability and anonymity can be combined to give rise to the ANO-IND-ID-CPA security notion. Halevi's sufficient condition [18] was extended to IBE schemes in [1]. Namely, an IBE scheme is ANO-IND-ID-CPA if it is IND-ID-CPA and if

$$\Pr\left[\begin{array}{l}(\mathsf{mpk},\mathsf{msk}) \xleftarrow{R} \mathtt{SETUP}(1^\kappa),\\ (\mathsf{id}_0^*,\mathsf{id}_1^*,m,s) \leftarrow \mathcal{A}_1^{\mathtt{EXTRACT}_{\mathsf{msk}}(\cdot)}(\mathsf{mpk}),\\ b \xleftarrow{R} \{0,1\}, r \xleftarrow{R} \mathcal{M} \text{ and } |r|_2 = |m|_2,\\ C \leftarrow \mathtt{ENCRYPT}_{\mathsf{mpk}}(\mathsf{id}_b^*, r)\end{array} : \mathcal{A}_2^{\mathtt{EXTRACT}_{\mathsf{msk}}(\cdot)}(s,C) = b\right] - \frac{1}{2}$$

is negligible in the security parameter for any polynomial-time adversary $\mathcal{A}$; the probability is taken over the random coins of the experiment according to the distribution induced by SETUP and over the random coins of the adversary. The difference is that a *random* message r is encrypted as opposed to the message m chosen by $\mathcal{A}$; the only restriction being that r and m must be of equal length.

2.3 Complexity assumptions

It is useful to introduce some notation. Let $N = pq$ be the product of two primes p and q. The set of integers whose Jacobi symbol is 1 is denoted by $\mathbb{J}_N$, $\mathbb{J}_N = \left\{a \in \mathbb{Z}_N^* \mid \left(\frac{a}{N}\right) = 1\right\}$; the set of quadratic residues is denoted by $\mathbb{QR}_N$, $\mathbb{QR}_N = \left\{a \in \mathbb{Z}_N^* \mid \left(\frac{a}{p}\right) = \left(\frac{a}{q}\right) = 1\right\}$. Notice that $\mathbb{QR}_N$ is a subset of $\mathbb{J}_N$.

This leads to the following computational assumption [17]. Basically, it says that quadratic residues cannot be distinguished from quadratic non-residues modulo an RSA composite $N = pq$.

Definition 1 (Quadratic Residuosity Assumption). *Let* RSAgen *be a probabilistic algorithm which, given a security parameter κ, outputs primes p and q and their product $N = pq$. The* Quadratic Residuosity (QR) assumption *relative to* RSAgen *asserts that the success probability defined as the distance*

$$\left|\Pr[\mathcal{D}(x,N) = 1 \mid x \xleftarrow{R} \mathbb{QR}_N] - \Pr[\mathcal{D}(x,N) = 1 \mid x \xleftarrow{R} \mathbb{J}_N \setminus \mathbb{QR}_N]\right|$$

is negligible for any probabilistic polynomial-time distinguisher $\mathcal{D}$; the probabilities are taken over the experiment of running $(N,p,q) \leftarrow \mathsf{RSAgen}(1^\kappa)$ and choosing at random $x \in \mathbb{QR}_N$ and $x \in \mathbb{J}_N \setminus \mathbb{QR}_N$.

A stronger assumption is introduced in [8]. It says that the QR assumption holds in the presence of a hash square-root oracle. More formally, the assumption is defined as follows.

Definition 2 (Interactive Quadratic Residuosity Assumption). *Again let* RSAgen *be a probabilistic algorithm which, given a security parameter κ, outputs primes p and q and their product $N = pq$. Let also $\mathcal{H}$ be a hash function that on input an arbitrary bit-string returns an element in $\mathbb{J}_N$ and let $\mathcal{O}$ be a hash square-root oracle that maps an input pair (N,s) to one of $\mathcal{H}(s)^{1/2} \bmod N$*

or $(u\mathcal{H}(x))^{1/2} \bmod N$*, for some quadratic non-residue* $u \in \mathbb{J}_N$*. The* Interactive Quadratic Residuosity (IQR) assumption *asserts that the success probability defined as the distance*

$$\left|\Pr[\mathcal{D}^{\mathcal{O}}(x, N) = 1 \mid x \xleftarrow{R} \mathbb{QR}_N] - \Pr[\mathcal{D}^{\mathcal{O}}(x, N) = 1 \mid x \xleftarrow{R} \mathbb{J}_N \setminus \mathbb{QR}_N]\right|$$

is negligible for any probabilistic polynomial-time distinguisher $\mathcal{D}$*; the probabilities are taken over the experiment of running* $(N, p, q) \leftarrow \mathsf{RSAgen}(1^\kappa)$*, choosing at random oracle* $\mathcal{O}$*, and choosing at random* $x \in \mathbb{QR}_N$ *and* $x \in \mathbb{J}_N \setminus \mathbb{QR}_N$*.*

Remark 2. As noted in [8] the IQR assumption is equivalent to the QR assumption in the random oracle model [5].

3 Review of Cocks' Scheme

In 2001, Cocks published an identity-based encryption scheme that does not rely on pairings over elliptic curves [12]. Cocks' scheme works in standard RSA groups and its security relies on the quadratic residuosity assumption (in the random oracle model). The encryption processes one bit at a time. To simplify the presentation, we assume that messages being encrypted are in the set $\{-1, 1\}$. For example, the map $\mu\colon \{0,1\} \to \{-1,1\},\ b \mapsto m = (-1)^b$ maps a bit b to a message $m \in \mathcal{M} = \{\pm 1\}$. The inverse map is given by $\mu^{-1}(m) = (1-m)/2$.

3.1 Description

Cocks' scheme proceeds as follows.

$\texttt{SETUP}(1^\kappa)$ Given a security parameter κ, SETUP generates an RSA modulus $N = pq$ where p and q are prime. It also selects an element $u \in \mathbb{J}_N \setminus \mathbb{QR}_N$. The system parameters are $\mathsf{mpk} = \{N, u, \mathcal{H}\}$ where $\mathcal{H}$ is a cryptographic hash function mapping bit-strings to $\mathbb{J}_N$. The master secret key is $\mathsf{msk} = \{p, q\}$.

$\texttt{EXTRACT}_{\mathsf{msk}}(\mathsf{id})$ Using hash function $\mathcal{H}$, EXTRACT sets $R_{\mathsf{id}} = \mathcal{H}(\mathsf{id})$. If $R_{\mathsf{id}} \in \mathbb{QR}_N$ it computes $r_{\mathsf{id}} = {R_{\mathsf{id}}}^{1/2} \bmod N$; otherwise it computes $r_{\mathsf{id}} = (uR_{\mathsf{id}})^{1/2} \bmod N$. EXTRACT returns user's private key $\mathsf{usk} = \{r_{\mathsf{id}}\}$.

$\texttt{ENCRYPT}(\mathsf{id}, m)$ To encrypt a message $m \in \{\pm 1\}$ for user with identity id, ENCRYPT chooses at random $t, \bar{t} \in \mathbb{Z}/N\mathbb{Z}$ such that $\left(\frac{t}{N}\right) = \left(\frac{\bar{t}}{N}\right) = m$. It then computes

$$c = t + \frac{R_{\mathsf{id}}}{t} \bmod N \quad \text{and} \quad \bar{c} = \bar{t} + \frac{uR_{\mathsf{id}}}{\bar{t}} \bmod N$$

where $R_{\mathsf{id}} = \mathcal{H}(\mathsf{id})$. The returned ciphertext is $C = (c, \bar{c})$.

$\texttt{DECRYPT}_{\mathsf{usk}}(C)$ From $\mathsf{usk} = \{r_{\mathsf{id}}\}$ and $C = (c, \bar{c})$, if ${r_{\mathsf{id}}}^2 \equiv \mathcal{H}(\mathsf{id}) \pmod N$, DECRYPT sets $\gamma = c$; otherwise it sets $\gamma = \bar{c}$. Plaintext m is then recovered as

$$m = \left(\frac{\gamma + 2r_{\mathsf{id}}}{N}\right) .$$

Remark 3. The above description is a generalization of the original scheme. In [12], Cocks considers Blum integers; namely, RSA moduli $N = pq$ with $p, q \equiv 3 \pmod 4$. Doing so, it follows that $\left(\frac{-1}{p}\right) = \left(\frac{-1}{q}\right) = -1$ and therefore $-1 \in \mathbb{J}_N \setminus \mathbb{QR}_N$. The original scheme corresponds to the choice $u = -1$. The above description also slightly generalizes the one offered in [8, Appendix A] in that parameter u is not necessarily chosen as a *random* quadratic non-residue in $\mathbb{J}_N$.

Remark 4. Alternatively, the decryption algorithm can recover plaintext m as $m = \left(\frac{\gamma - 2r_{\mathsf{id}}}{N}\right)$. The correctness of the decryption follows by remarking that when ${r_{\mathsf{id}}}^2 \equiv \mathcal{H}(\mathsf{id}) \pmod N$, $\gamma \pm 2r_{\mathsf{id}} \equiv t(1 \pm \frac{r_{\mathsf{id}}}{t})^2 \pmod N$ yielding $\left(\frac{\gamma \pm 2r_{\mathsf{id}}}{N}\right) = \left(\frac{t}{N}\right) = m$. Likewise, when ${r_{\mathsf{id}}}^2 \equiv u\mathcal{H}(\mathsf{id}) \pmod N$, $\gamma \pm 2r_{\mathsf{id}} \equiv \bar{t}(1 \pm \frac{r_{\mathsf{id}}}{\bar{t}})^2 \pmod N$ and thus $\left(\frac{\gamma \pm 2r_{\mathsf{id}}}{N}\right) = \left(\frac{\bar{t}}{N}\right) = m$.

3.2 Security analysis

The next proposition shows that the generalized Cocks' scheme is semantically secure under the QR assumption in the random oracle model. Equivalently, as mentioned in Remark 2, the scheme is semantically secure under the IQR assumption in the standard model.

Proposition 1. *The scheme of § 3.1 is* $\mathsf{IND\text{-}ID\text{-}CPA}$ *under the quadratic residuosity assumption in the random oracle model.*

Proof. The proof can be found in Appendix A. □

4 A Useful Representation

Let $\mathbb{F}_q$ denote the finite field with q elements, where $q = p^r$ is a prime power. The order of the multiplicative group $\mathbb{F}_{p^r}^\times = \mathbb{F}_{p^r} \setminus \{0\}$ is $p^r - 1$. Note that $p^r - 1 = \prod_{d|r} \Phi_d(p)$ where $\Phi_d(x)$ represents the d-th cyclotomic polynomial. We let $\mathsf{G}_{p,r} \subseteq \mathbb{F}_{p^r}^\times$ denote the cyclic subgroup of order $\Phi_r(p)$. In [22], Rubin and Silverberg identify $\mathsf{G}_{p,r}$ with the $\mathbb{F}_p$-points of an algebraic torus. Namely, they consider

$$\mathbb{T}_r(\mathbb{F}_p) = \{\alpha \in \mathbb{F}_{p^r}^\times \mid N_{\mathbb{F}_{p^r}/F}(\alpha) = 1 \text{ whenever } \mathbb{F}_p \subseteq F \subsetneq \mathbb{F}_{p^r}\},$$

that is, the elements of $\mathbb{F}_{p^r}^\times$ whose norm is one down to every intermediate subfield F. Their key observation is that $\mathbb{T}_r(\mathbb{F}_p)$ forms a group whose elements can be represented with only $\phi(r)$ elements of $\mathbb{F}_p$, where ϕ denotes Euler's totient function. The compression factor is thus of $r/\phi(r)$ over the field representation [22,15].

4.1 Parametrization of $\mathbb{T}_2(\mathbb{F}_p)$

This corresponds to the case $r = 2$. We review the explicit representation for $\mathbb{T}_2(\mathbb{F}_p)$ presented in [22, Section 5.2].

For simplicity, we assume that p is an *odd* prime. Let $\Delta = \delta^2 \in \mathbb{F}_p^\times$ with $\delta \notin \mathbb{F}_p$. Then $\mathbb{T}_2(\mathbb{F}_p)$ is the multiplicative group given by

$$\mathbb{T}_2(\mathbb{F}_p) = \left\{x + \delta y \mid x, y \in \mathbb{F}_p \text{ and } x^2 - \Delta y^2 = 1\right\} .$$

Define the map $\psi\colon \mathbb{F}_p \to \mathbb{T}_2(\mathbb{F}_p),\ u \mapsto \frac{u+\delta}{u-\delta} = \frac{u^2+\Delta}{u^2-\Delta} + \delta\,\frac{2u}{u^2-\Delta}$. The inverse map is given by $\psi^{-1}\colon \mathbb{T}_2(\mathbb{F}_p) \setminus \{1\} \to \mathbb{F}_p,\ v \mapsto \frac{\delta(v+1)}{v-1}$. By augmenting $\mathbb{F}_p$ with a special symbol ∞ and defining $\psi(\infty) = 1$, maps ψ and ψ^{-1} extend naturally to give an isomorphism $T_2(\mathbb{F}_p) \xrightarrow{\sim} \mathbb{F}_p \cup \{\infty\}$.

4.2 An alternative representation for $(\mathbb{Z}/N\mathbb{Z})^\times$

One may wonder what happens if Δ is chosen as a quadratic residue in the Rubin-Silverberg representation for $\mathbb{T}_2(\mathbb{F}_p)$. As will become apparent in § 4.3, this seemingly useless setting has practical consequences. We start with the multiplicative group $\mathbb{F}_p^\times$ and then extend our results to $(\mathbb{Z}/N\mathbb{Z})^\times$ through Chinese remaindering.

The group $\mathcal{F}_{p,\Delta}$ Let p be an odd prime. We henceforth assume that $\delta \in \mathbb{F}_p^\times$ and thus that $\Delta = \delta^2 \in \mathbb{QR}_p$. In this case, the torus $\mathbb{T}_2(\mathbb{F}_p)$ becomes isomorphic to $\mathbb{T}_1(\mathbb{F}_p) = \mathbb{F}_p^\times$. Note also that map ψ as given in § 4.1 is no longer defined at $u = \delta$. Moreover, when $\delta \in \mathbb{F}_p^\times$, $\psi(-\delta) = 0$ cannot be expressed as $\psi(-\delta) = x + \delta y$ for some $x, y \in \mathbb{F}_p$ with $x^2 - \Delta y^2 = 1$. So, we define the set

$$\mathcal{F}_{p,\Delta} = (\mathbb{F}_p \setminus \{\pm\delta\}) \cup \{\infty\} = \{u \in \mathbb{F}_p \mid u^2 \neq \Delta\} \cup \{\infty\}$$

and restrict map ψ to $\mathcal{F}_{p,\Delta}$:

$$\psi\colon \mathcal{F}_{p,\Delta} \to \mathbb{F}_p^\times,\ u \mapsto \begin{cases} \frac{u+\delta}{u-\delta} & \text{if } u \neq \infty\,, \\ 1 & \text{otherwise}\,. \end{cases}$$

For completeness, we show that $\mathcal{F}_{p,\Delta}$ equipped with the group law $\circledast$ (defined hereafter) and $\mathbb{F}_p^\times$ are isomorphic. Clearly, the map $\psi\colon \mathcal{F}_{p,\Delta} \to \mathbb{F}_p^\times$ is injective and thus defines a bijection. Indeed, suppose $\psi(u_1) = \psi(u_2)$ for some $u_1, u_2 \in \mathcal{F}_{p,\Delta}$. If $\psi(u_1) \neq 1$, this implies $(u_1+\delta)(u_2-\delta) = (u_2+\delta)(u_1-\delta)$ and in turn $u_1 = u_2$; it $\psi(u_1) = 1$ then again this implies $u_1 = u_2\ (= \infty)$ since $(u+\delta)/(u-\delta) \neq 1$ for every $u \in \mathcal{F}_{p,\Delta}$. The inverse map is given by

$$\psi^{-1}\colon \mathbb{F}_p^\times \to \mathcal{F}_{p,\Delta},\ v \mapsto \begin{cases} \frac{\delta(v+1)}{v-1} & \text{if } v \neq 1\,, \\ \infty & \text{otherwise}\,. \end{cases}$$

Furthermore, map ψ yields a homomorphism from $\mathcal{F}_{p,\Delta}$ to $\mathbb{F}_p^\times$. Let $u_1, u_2 \in \mathcal{F}_{p,\Delta} \setminus \{\infty\}$ with $u_1 \neq -u_2$. Then we have

$$\psi(u_1) \cdot \psi(u_2) = \frac{(u_1+\delta)(u_2+\delta)}{(u_1-\delta)(u_2-\delta)} = \frac{u_1u_2+\Delta+\delta(u_1+u_2)}{u_1u_2+\Delta-\delta(u_1+u_2)} = \frac{\frac{u_1u_2+\Delta}{u_1+u_2}+\delta}{\frac{u_1u_2+\Delta}{u_1+u_2}-\delta}$$
$$= \psi(u_3) \quad \text{where } u_3 = \tfrac{u_1u_2+\Delta}{u_1+u_2}.$$

Note also that $\psi(\infty) = 1$ and that $\frac{1}{\psi(u_1)} = \frac{u_1-\delta}{u_1+\delta} = \frac{-u_1+\delta}{-u_1-\delta} = \psi(-u_1)$.

We write $\mathcal{F}_{p,\Delta}$ multiplicatively and use $\circledast$ to denote its group law. In more detail, we have:

- the neutral element is ∞: $u \circledast \infty = \infty \circledast u = u$ for all $u \in \mathcal{F}_{p,\Delta}$;
- the inverse of $u \in \mathcal{F}_{p,\Delta} \setminus \{\infty\}$ is $-u$: $u \circledast (-u) = (-u) \circledast u = \infty$;
- given $u_1, u_2 \in \mathcal{F}_{p,\Delta} \setminus \{\infty\}$, their product is given by:

$$u_1 \circledast u_2 = \begin{cases} \frac{u_1u_2+\Delta}{u_1+u_2} & \text{if } u_1 \neq -u_2\,, \\ \infty & \text{otherwise}\,. \end{cases}$$

The group $\mathcal{Z}_{N,\Delta}$ The previous setting naturally extends through Chinese remaindering. Let $N = pq$ be an RSA modulus. Then

$$\mathcal{Z}_{N,\Delta} := \mathcal{F}_{p,\Delta} \times \mathcal{F}_{q,\Delta} \cong (\mathbb{Z}/N\mathbb{Z})^\times \tag{1}$$

is a group w.r.t. $\circledast$ and has order $\phi(N)$.

For each element $u \in \mathcal{Z}_{N,\Delta}$, there exists a unique pair of elements $u_p \in \mathcal{F}_{p,\Delta}$ and $u_q \in \mathcal{F}_{q,\Delta}$ such that $u \bmod p = u_p$ and $u \bmod q = u_q$. We denote this equivalence by $u = [u_p, u_q]$ and let $\infty = [\infty_p, \infty_q]$ represent the neutral element. We also define the subset $\widetilde{\mathcal{Z}}_{N,\Delta} := \big((\mathcal{F}_{p,\Delta} \setminus \{\infty_p\}) \times (\mathcal{F}_{q,\Delta} \setminus \{\infty_q\})\big) \cup \{\infty\}$.

Efficient methods for working in $\mathcal{Z}_{N,\Delta}$ are discussed in Appendix B.

Remark 5. Please note that 0 is of order 2 as an element of $\mathcal{Z}_{N,\Delta}$, namely $0 \circledast 0 = \infty$. Note also that for any $u \in \mathcal{Z}_{N,\Delta}$, $u \neq 0, \infty$, we have $u \circledast 0 = \Delta/u$.

4.3 The subset $\mathcal{S}_{N,\Delta}$ of squares in $\widetilde{\mathcal{Z}}_{N,\Delta}$

We now have all ingredients to introduce the useful quasi-group $\mathcal{S}_{N,\delta}$. Let $N = pq$ be an RSA modulus and let $\Delta = \delta^2 \in \mathbb{QR}_N$. Consider the set of squares in $\widetilde{\mathcal{Z}}_{N,\Delta}$, namely $(\widetilde{\mathcal{Z}}_{N,\Delta})^2 = \{u \circledast u \mid u \in \widetilde{\mathcal{Z}}_{N,\Delta}\} \subset (\mathcal{Z}_{N,\Delta})^2$, or more exactly, the subset

$$\mathcal{S}_{N,\Delta} \stackrel{\text{def}}{=} \left\{ \tfrac{u^2+\Delta}{2u} \mid u \in (\mathbb{Z}/N\mathbb{Z})^\times \text{ and } \gcd(u^2-\Delta, N) = 1 \right\} \tag{2}$$
$$= (\widetilde{\mathcal{Z}}_{N,\Delta})^2 \setminus \{\infty\}\ .$$

The set $\mathcal{S}_{N,\Delta}$ almost defines a group: it contains all elements s of the group $(\mathcal{Z}_{N,\Delta})^2 = \{s = u \circledast u \mid u \in \mathcal{Z}_{N,\Delta}\}$ minus elements of the form $[s_p, \infty_q]$ or

$[\infty_p, s_q]$ (and $\infty = [\infty_p, \infty_q]$). Since it is a subset of $(\mathcal{Z}_{N,\Delta})^2$, $\mathcal{S}_{N,\Delta}$ is endowed with the $\circledast$-law. In practice, for cryptographic applications, working in $(\widetilde{\mathcal{Z}}_{N,\Delta})^2 = \mathcal{S}_{N,\Delta} \cup \{\infty\}$ rather than in $(\mathcal{Z}_{N,\Delta})^2$ does not really matter since the probability that operation $\circledast$ is not defined on $(\widetilde{\mathcal{Z}}_{N,\Delta})^2$ is negligible.

What makes the quasi-group $\mathcal{S}_{N,\Delta}$ special is that, up to a scaling factor of two, it represents the set of all valid [components of] Cocks ciphertexts. Before making this statement clear, we need to explain what is meant by 'valid component'. This follows from the next lemma, adapted from [12, Section 5]. Basically, it shows that, given a (generalized) Cocks ciphertext $C = (c, \bar{c})$, among its two components c and $\bar{c}$, one carries no information whatsoever about the corresponding plaintext. The component that yields the plaintext is called *valid component*.

Lemma 1. *Using the notations of § 3.1, let $C = (c, \bar{c})$ be a (generalized) Cocks ciphertext. If $\mathcal{H}(\mathsf{id}) \notin \mathbb{QR}_N$ then the component c corresponds with the same probability to the encryption of message $m = 1$ or $m = -1$. Conversely, if $u\,\mathcal{H}(\mathsf{id}) \notin \mathbb{QR}_N$ then the component $\bar{c}$ corresponds with the same probability to the encryption of message $m = 1$ or $m = -1$.*

Proof. Suppose that $\mathcal{H}(\mathsf{id}) \notin \mathbb{QR}_N$ (i.e., $\mathcal{H}(\mathsf{id}) \in \mathbb{J}_N \setminus \mathbb{QR}_N$). Let $R_{\mathsf{id}} = \mathcal{H}(\mathsf{id})$. We have $c = t + \frac{R_{\mathsf{id}}}{t} \bmod N$ for some random $t \in (\mathbb{Z}/N\mathbb{Z})^\times$ such $\left(\frac{t}{N}\right) = m$. Consider also $t_1, t_2, t_3 \in (\mathbb{Z}/N\mathbb{Z})^\times$ such that

- $t_1 \equiv t \pmod p$, $t_1 \equiv R_{\mathsf{id}}/t \pmod q$;
- $t_2 \equiv R_{\mathsf{id}}/t \pmod p$, $t_2 \equiv t \pmod q$;
- $t_3 \equiv R_{\mathsf{id}}/t \pmod p$, $t_3 \equiv R_{\mathsf{id}}/t \pmod q$.

Note that the condition $\left(\frac{t}{N}\right) = m$ implies $\left(\frac{t_1}{N}\right) = \left(\frac{t_2}{N}\right) = \left(\frac{t_3}{N}\right) = m$. The four possible values t, t_1, t_2, and t_3 are equally likely since $c \equiv t + R_{\mathsf{id}}/t \equiv t_1 + R_{\mathsf{id}}/t_1 \equiv t_2 + R_{\mathsf{id}}/t_2 \equiv t_3 + R_{\mathsf{id}}/t_3 \pmod N$. At the same time, since $R_{\mathsf{id}} \in \mathbb{J}_N \setminus \mathbb{QR}_N$ we also have $\left(\frac{t}{N}\right) = \left(\frac{t_3}{N}\right) \neq \left(\frac{t_1}{N}\right) = \left(\frac{t_2}{N}\right)$. Hence, component c leaks no information about $\left(\frac{t}{N}\right)$; it has the same probability to be 1 or -1.

The case $u\,\mathcal{H}(\mathsf{id}) \notin \mathbb{QR}_N$ is proved similarly. □

With the notations of § 3.1, from a ciphertext $C = (c, \bar{c})$, letting $\Delta = \mathcal{H}(\mathsf{id}) \in \mathbb{QR}_N$ (resp. $\Delta = u\,\mathcal{H}(\mathsf{id}) \in \mathbb{QR}_N$) and $\gamma = c$ (resp. $\bar{c}$), we have

$$\frac{\gamma}{2} \in \mathcal{S}_{N,\Delta} \implies \exists \tau \in (\mathbb{Z}/N\mathbb{Z})^\times \text{ with } \gcd(\tau^2 - \Delta, N) = 1 \text{ such that } \gamma = \tfrac{\tau^2+\Delta}{\tau}\ .$$

In other words, up to a factor of two, the valid component of C is an element of $\mathcal{S}_{N,\Delta}$; i.e., $\frac{\gamma}{2} \in \mathcal{S}_{N,\Delta}$. By Lemma 1, the other component does not matter. Note also that the condition $\gcd(\tau^2 - \Delta, N) = 1$ is implicit in the (generalized) Cocks encryption since $\left(\frac{\tau}{N}\right) = m \in \{\pm 1\}$ where $\tau = t$ (resp. $\tau = \bar{t}$) and decryption is obtained as

$$\left(\frac{\gamma \pm 2\delta}{N}\right)$$

which is 0 when $\gcd(\tau^2 - \Delta, N) \neq 1 \iff \tau \equiv \pm\delta \pmod{\{p, q\}}$. The condition $\tau \in (\mathbb{Z}/N\mathbb{Z})^\times$ (instead of $t, \bar{t} \in \mathbb{Z}/N\mathbb{Z}$) is trivially satisfied since $\left(\frac{\tau}{N}\right) \in \{\pm 1\}$.

5 Computing over Cocks Ciphertexts

Homomorphic encryption is a form of encryption which allows combining two ciphertexts through a non-private operation that results in a third ciphertext which, when decrypted, yields a plaintext that is the combination of the corresponding two plaintexts through a specific operation. Mathematically, for two ciphertexts $C_1 = \texttt{ENCRYPT}(m_1)$ and $C_2 = \texttt{ENCRYPT}(m_2)$, there exists a non-private operation ∂ such that $C_1 \,\partial\, C_2 = \texttt{ENCRYPT}(m_1 \star m_2)$ for some specific operation $\star$. Examples of known operations $\star$ include (modular) addition and (modular) multiplication.

5.1 HOM procedure

Consider the following procedure **HOM**. It takes as input two elements $x_1, x_2 \in \mathbb{Z}/N\mathbb{Z}$ and an element $\Gamma \in \mathbb{J}_N$, and outputs an element $z \in \mathbb{Z}/N\mathbb{Z}$. We write $z = \mathbf{HOM}(x_1, x_2, \Gamma)$.

1: **procedure HOM**(x_1, x_2, Γ)
2: Define $D = x_1 x_2 + 4\Gamma \bmod N$ and $U = x_1 + x_2 \bmod N$;
3: Select $t \in \mathbb{Z}/N\mathbb{Z}$ such that $\left(\frac{\theta}{N}\right) = 1$ where

$$\theta = t\,D + (t^2 + \Gamma)U \bmod N\,;$$

4: Evaluate

$$z = \frac{(t^2 + \Gamma)D + 4\Gamma t U}{\theta} \bmod N\,;$$

5: Return z.
6: **end procedure**

Remark 6. Note that when $\left(\frac{U}{N}\right) = \left(\frac{x_1+x_2}{N}\right) = 1$, we can take $t = 0$ (in Procedure **HOM**, Line 3). This yields $\theta = \Gamma U \bmod N$ $(\in \mathbb{J}_N)$ and in turn $z = D/U \bmod N$.

5.2 Application

The **HOM** procedure allows for computing over two Cocks ciphertexts.

Specifically, suppose we are given two ciphertexts $C_1 = \{c_1, \bar{c}_1\}$ and $C_2 = \{c_2, \bar{c}_2\}$ that are the respective encryption of two messages m_1 and m_2 for a same identity, say id. The system parameters are $\mathsf{mpk} = \{N, u, \mathcal{H}\}$ where $N = pq$ is an RSA modulus, u is a quadratic non-residue in $\mathbb{J}_N$, and $\mathcal{H}$ is a hash function mapping bit-strings to $\mathbb{J}_N$. As in the description given in § 3.1, we assume that the message space is $\mathcal{M} = \{\pm 1\}$.

Let $R_{\mathsf{id}} = \mathcal{H}(\mathsf{id})$. Two applications of the **HOM** procedure yields a third ciphertext $C_3 = (c_3, \bar{c}_3)$ obtained as

$$c_3 = \mathbf{HOM}(c_1, c_2, R_{\mathsf{id}}) \quad \text{and} \quad \bar{c}_3 = \mathbf{HOM}(\bar{c}_1, \bar{c}_2, uR_{\mathsf{id}})\ .$$

A simple calculation shows that C_3 is the encryption of $m_3 = m_1 \cdot m_2$.

Proof. Suppose first that $R_{\mathsf{id}} \in \mathbb{QR}_N$. Then, letting $r_{\mathsf{id}} = {R_{\mathsf{id}}}^{1/2} \bmod N$, we have:

$$\begin{aligned}\left(\tfrac{c_3+2r_{\mathsf{id}}}{N}\right) &= \left(\tfrac{\theta}{N}\right)\left(\tfrac{c_3+2r_{\mathsf{id}}}{N}\right) = \left(\tfrac{(t^2+R_{\mathsf{id}})D+4R_{\mathsf{id}}tU+2r_{\mathsf{id}}(tD+(t^2+R_{\mathsf{id}})U)}{N}\right)\\ &= \left(\tfrac{(t^2+R_{\mathsf{id}}+2r_{\mathsf{id}}t)D+(2r_{\mathsf{id}}t+t^2+R_{\mathsf{id}})(2r_{\mathsf{id}}U)}{N}\right) = \left(\tfrac{(t+r_{\mathsf{id}})^2(D+2r_{\mathsf{id}}U)}{N}\right)\\ &= \left(\tfrac{D+2r_{\mathsf{id}}U}{N}\right) = \left(\tfrac{(c_1c_2+4R_{\mathsf{id}})+2r_{\mathsf{id}}(c_1+c_2)}{N}\right)\\ &= \left(\tfrac{c_1+2r_{\mathsf{id}}}{N}\right)\left(\tfrac{c_2+2r_{\mathsf{id}}}{N}\right)\end{aligned}$$

as desired.

The case $R_{\mathsf{id}} \in \mathbb{J}_N \setminus \mathbb{QR}_N$ is similar. Letting $r_{\mathsf{id}} = (uR_{\mathsf{id}})^{1/2} \bmod N$, we then have $\left(\tfrac{\bar{c}_3+2r_{\mathsf{id}}}{N}\right) = \left(\tfrac{\bar{c}_1+2r_{\mathsf{id}}}{N}\right)\left(\tfrac{\bar{c}_2+2r_{\mathsf{id}}}{N}\right)$. □

Why does it work? At first sight, the **HOM** procedure may appear cumbersome. Actually it is not. Without loss of generality, assume that input $\Gamma \leftarrow R_{\mathsf{id}} = {r_{\mathsf{id}}}^2 \in \mathbb{QR}_N$. A straightforward application of the homomorphism induced by the underlying $\circledast$ law will not get the correct result. Clearly, if we call $c_3 = \mathbf{HOM}(c_1, c_2, R_{\mathsf{id}})$ and let

$$\frac{c_3'}{2} = \frac{c_1}{2} \circledast \frac{c_2}{2} \iff c_3' = \frac{c_1c_2 + 4R_{\mathsf{id}}}{c_1 + c_2}$$

then $\left(\tfrac{c_3+2r_{\mathsf{id}}}{N}\right) = \left(\tfrac{c_3'+2r_{\mathsf{id}}}{N}\right)$ if and only if $\left(\tfrac{c_1+c_2}{N}\right) = 1$. Indeed, the above definition of c_3' immediately yields

$$\left(\frac{c_3 + 2r_{\mathsf{id}}}{N}\right) := \left(\frac{c_1 + 2r_{\mathsf{id}}}{N}\right)\left(\frac{c_2 + 2r_{\mathsf{id}}}{N}\right) = \left(\frac{c_1 + c_2}{N}\right)\left(\frac{c_3' + 2r_{\mathsf{id}}}{N}\right) .$$

In other words, c_3' is the encryption of $m_1 \cdot m_2$ if and only if $\left(\tfrac{c_1+c_2}{N}\right) = 1$. This problem is resolved by randomizing one of the input ciphertexts using the homomorphism. One way to achieve this is to replace ciphertext c_1 with an equivalent randomized ciphertext,

$$\frac{c_1}{2} \leftarrow \frac{c_1}{2} \circledast \frac{\mathbb{1}}{2}$$

until $\left(\tfrac{c_1+c_2}{N}\right) = 1$, where $\mathbb{1} = \mathtt{ENCRYPT}_{\mathsf{msk}}(\mathsf{id}, 1)$ is a random Cocks encryption (w.r.t. R_{id}) of message $m = 1$. Note that no secret is involved in the randomization of c_1. This is the design strategy behind the **HOM** procedure.

We have seen that the **HOM** procedure can be used to randomize ciphertexts. Likewise, it can be used to flip the value of a plaintext message $m_1 \in \{\pm 1\}$ corresponding to a given Cocks ciphertext $C_1 = (c_1, \bar{c}_1)$ by taking the encryption of $m_2 = -1$ for ciphertext $C_2 = (c_2, \bar{c}_2)$.

A variant of Cocks' scheme that makes easier the computation over ciphertexts can be found in Appendix C.

It is also worth noting that when the message space is $\{0,1\}$ rather than $\{\pm 1\}$ then the scheme is homomorphic with respect the XOR operator. Indeed, letting $b_1 = \mu^{-1}(m_1)$ and $b_2 = \mu^{-1}(m_2)$, we have $\mu^{-1}(m_1 \cdot m_2) = b_1 \oplus b_2$, where $\mu^{-1}(m_i) = (1 - m_i)/2$ —see Section 3. We so get $\mu(b_1 \oplus b_2) = (-1)^{b_1 \oplus b_2} = (-1)^{b_1 + b_2} = m_1 \cdot m_2$.

6 An Anonymous IBE Scheme

In numerous scenarios, the recipient's identity in a transmission needs to be kept *anonymous*. This allows users to maintain some privacy. Protecting communication content may be not enough, as already observed in, e.g., [3,4,21]. For example, by analyzing the traffic between an antenna and a mobile device, one can recover some information about [at least] user's position and some details about the use of her mobile device. This information leaks easily during all day: it is a common habit, indeed, to use a mobile phone every day and to keep it (almost) always switched on.

As pointed out by Galbraith (see [6, Section 4], Cocks' scheme is *not* anonymous. A detailed discussion on the so-called Galbraith's test can be found in [2, Section 2.3]). In the same paper, Ateniese and Gasti also show that Galbraith's is the "best test" possible against the anonymity of Cocks' scheme.

In this section, we rephrase Galbraith's test using our representation. We then build on an original technique developed in [11] to get anonymized Cocks ciphertexts. However, unlike [11], there is no ciphertext expansion. Anonymized Cocks ciphertexts have the same size as non-anonymized ciphertexts. The decryption algorithm is modified accordingly by first de-anonymizing the ciphertext and then applying the regular decryption process. The resulting scheme is shown to meet the ANO-IND-ID-CPA security notion under the QR assumption in the random oracle model or, equivalently, under the IQR assumption in the standard model (cf. Remark 2).

6.1 Making Cocks ciphertexts anonymous

As Equation (6) indicates, the subset $\widetilde{\mathcal{Z}}_{N,\Delta} \subset \mathcal{Z}_{N,\Delta}$ can be defined as

$$\widetilde{\mathcal{Z}}_{N,\Delta} = \{u \in \mathbb{Z}/N\mathbb{Z} \mid \gcd(u^2 - \Delta, N) = 1\} \cup \{\infty\}$$

where $N = pq$ and $\Delta \in \mathbb{QR}_N$. The following subsets of $\widetilde{\mathcal{Z}}_{N,\Delta}$ will be useful:

- $\widehat{\mathcal{Z}}_{N,\Delta} := \{u \in \mathbb{Z}/N\mathbb{Z} \mid \gcd(u^2 - \Delta, N) = 1\} \subset \widetilde{\mathcal{Z}}_{N,\Delta}$;
- $\widehat{\mathcal{Z}}^{[-1]}_{N,\Delta} := \{u \in \mathbb{Z}/N\mathbb{Z} \mid \left(\frac{u^2-\Delta}{N}\right) = -1\} \subset \widehat{\mathcal{Z}}_{N,\Delta}$;
- $\widehat{\mathcal{Z}}^{[+1]}_{N,\Delta} := \{u \in \mathbb{Z}/N\mathbb{Z} \mid \left(\frac{u^2-\Delta}{N}\right) = 1\} \subset \widehat{\mathcal{Z}}_{N,\Delta}$;
- $(\widehat{\mathcal{Z}}_{N,\Delta})^2 := \{u \in \mathbb{Z}/N\mathbb{Z} \mid \left(\frac{u^2-\Delta}{p}\right) = \left(\frac{u^2-\Delta}{q}\right) = 1\} \subset \widehat{\mathcal{Z}}^{[+1]}_{N,\Delta}$.

We have

$$\widehat{\mathcal{Z}}_{N,\Delta} = \widetilde{\mathcal{Z}}_{N,\Delta} \setminus \{\infty\} = (\mathcal{F}_{p,\Delta} \setminus \{\infty_p\}) \times (\mathcal{F}_{q,\Delta} \setminus \{\infty_q\})$$

from Eq. (5). By definition (see § 4.2), for $u \in \mathcal{F}_{p,\Delta} \setminus \{\infty_p\}$, we have $\psi(u) = \frac{u+\delta}{u-\delta}$, and $\psi(\infty_p) = 1$. Multiplying both sides by $(u-\delta)^2$, the latter identity yields

$$\psi(u) \cdot (u-\delta)^2 = u^2 - \Delta \ . \tag{3}$$

The group $(\mathcal{F}_{p,\Delta})^2 = \{u \circledast u \mid u \in \mathcal{F}_{p,\Delta}\}$ is the subgroup of squares in $\mathcal{F}_{p,\Delta}$. Therefore, if $u \in (\mathcal{F}_{p,\Delta})^2$ then $\psi(u)$ is a quadratic residue modulo p; i.e., $\left(\frac{\psi(u)}{p}\right) = 1$. Together with Eq. (3) it follows that $u^2 - \Delta$ is a quadratic residue modulo p when $u \in (\mathcal{F}_{p,\Delta})^2$, $u \neq \infty_p$. This gives an alternative definition for the group $(\mathcal{F}_{p,\Delta})^2$, namely $(\mathcal{F}_{p,\Delta})^2 = \left\{u \in \mathcal{F}_{p,\Delta} \mid \left(\frac{u^2-\Delta}{p}\right) = 1\right\} \cup \{\infty_p\} = \left\{u \in \mathbb{F}_p \mid \left(\frac{u^2-\Delta}{p}\right) = 1\right\} \cup \{\infty_p\}$. Hence, using Chinese remaindering, we get

$$(\widehat{\mathcal{Z}}_{N,\Delta})^2 = \left((\mathcal{F}_{p,\Delta})^2 \setminus \{\infty_p\}\right) \times \left((\mathcal{F}_{q,\Delta})^2 \setminus \{\infty_q\}\right) \ .$$

This means that $(\widehat{\mathcal{Z}}_{N,\Delta})^2$ is an alternative representation for the subset $\mathcal{S}_{N,\Delta}$ of squares in $\widetilde{\mathcal{Z}}_{N,\Delta}$. Likewise, $\widehat{\mathcal{Z}}^{[-1]}_{N,\Delta}$ denotes the subset of elements that are squares modulo p and non-squares modulo q, or vice-versa; and $\widehat{\mathcal{Z}}^{[+1]}_{N,\Delta}$ denotes the subset of elements that are either both squares modulo p and modulo q, or both non-squares modulo p and modulo q.

We have shown that:

Proposition 2. *Let $N = pq$ be an RSA modulus and let $w \in \widehat{\mathcal{Z}}_{N,\Delta}$. If*

$$\left(\frac{w^2 - \Delta}{N}\right) = -1$$

then $w \notin \mathcal{S}_{N,\Delta}$. □

This is nothing but Galbraith's test. Back to the anonymity problem, letting $u \in \mathbb{J}_N \setminus \mathbb{QR}_N$, it implies that elements of the form $\frac{c}{2} = \frac{t^2+R_{\mathsf{id}}}{2t} \bmod N$ (respectively, $\frac{\bar{c}}{2} = \frac{\bar{t}^2+uR_{\mathsf{id}}}{2\bar{t}} \bmod N$) —where $R_{\mathsf{id}} = \mathcal{H}(\mathsf{id})$ is derived from some user's identity id— cannot be used as part of a ciphertext for user with identity id' because if $\left(\frac{(c/2)^2-R_{\mathsf{id}'}}{N}\right) = -1$ (respectively, if $\left(\frac{(\bar{c}/2)^2-uR_{\mathsf{id}'}}{N}\right) = -1$) —where $R_{\mathsf{id}'} = \mathcal{H}(\mathsf{id}')$ for some other identity id'— then one can conclude that the identity of the recipient of the ciphertext is <u>not</u> id'. This clearly violates the anonymity requirement.

This issue is easily solved by $\circledast$-multiplying with probability $1/2$ the value of $\frac{c}{2}$ (resp. $\frac{\bar{c}}{2}$) by an element $\frac{d}{2}$ satisfying $\left(\frac{(d/2)^2-\Delta}{N}\right) = -1$ (resp. $\left(\frac{(d/2)^2-u\Delta}{N}\right) = -1$). The decryption algorithm, assuming it is the legitimate recipient of the ciphertext, can then $\circledast$-divide by $\frac{d}{2}$ the ciphertext in the case it were $\circledast$-multiplied by $\frac{d}{2}$; letting $\frac{e}{2}$ (resp. $\frac{\bar{e}}{2}$) the received part of the ciphertext, it is easy for the decryption algorithm to know if $\frac{e}{2} = \frac{c}{2}$ or $\frac{e}{2} = \frac{c}{2} \circledast \frac{d}{2}$ (resp. $\frac{\bar{e}}{2} = \frac{\bar{c}}{2}$ or $\frac{\bar{e}}{2} = \frac{\bar{c}}{2} \circledast \frac{d}{2}$) by checking if $\left(\frac{(e/2)^2-\Delta}{N}\right) = 1$ or -1 (resp. $\left(\frac{(\bar{e}/2)^2-u\Delta}{N}\right) = 1$ or -1), respectively.

6.2 Anonymous IBE without ciphertext expansion

There is a variety of possible instantiations of the above methodology. An efficient implementation can be achieved by specializing hash function $\mathcal{H}$. Instead of considering a function mapping bit-strings to any element of $\mathbb{J}_N$, we require that, in addition, on input id, the output must satisfy the extra condition $\left(\frac{d^2-4R_{\mathsf{id}}}{N}\right) = \left(\frac{d^2-4uR_{\mathsf{id}}}{N}\right) = -1$ for some given d:

$$\mathcal{H}_d \colon \{0,1\}^* \to \mathbb{J}_N,\ \mathsf{id} \mapsto \mathcal{H}_d(\mathsf{id}) \ \text{ s.t. } \left(\frac{d^2-4\mathcal{H}_d(\mathsf{id})}{N}\right) = \left(\frac{d^2-4u\mathcal{H}_d(\mathsf{id})}{N}\right) = -1 \ . \quad (4)$$

Here is the resulting scheme.

$\texttt{SETUP}(1^\kappa)$ Given a security parameter κ, $\texttt{SETUP}$ generates an RSA modulus $N = pq$ where p and q are prime. It also selects an element $u \in \mathbb{J}_N \setminus \mathbb{QR}_N$ and a global integer d. The public system parameters are $\mathsf{mpk} = \{N, u, d, \mathcal{H}_d\}$ where $\mathcal{H}_d$ is a cryptographic hash function as per Eq. (4). The master secret key is $\mathsf{msk} = \{p, q\}$.

$\texttt{EXTRACT}_{\mathsf{msk}}(\mathsf{id})$ Given identity id, algorithm $\texttt{EXTRACT}$ sets $R_{\mathsf{id}} = \mathcal{H}_d(\mathsf{id})$. Then if $R_{\mathsf{id}} \in \mathbb{QR}_N$ it computes $r_{\mathsf{id}} = {R_{\mathsf{id}}}^{1/2} \bmod N$; otherwise it computes $r_{\mathsf{id}} = (uR_{\mathsf{id}})^{1/2} \bmod N$. $\texttt{EXTRACT}$ returns user's private key $\mathsf{usk} = \{r_{\mathsf{id}}\}$.

$\texttt{ENCRYPT}_{\mathsf{mpk}}(\mathsf{id}, m)$ To encrypt a message $m \in \{\pm 1\}$ for a user with identity id, $\texttt{ENCRYPT}$ defines $R_{\mathsf{id}} = \mathcal{H}_d(\mathsf{id})$. It chooses at random $t, \bar{t} \in \mathbb{Z}/N\mathbb{Z}$ such that $\left(\frac{t}{N}\right) = \left(\frac{\bar{t}}{N}\right) = m$ and lets

$$c^{(0)} = t + \frac{R_{\mathsf{id}}}{t} \bmod N\,, \quad c^{(1)} = \frac{c^{(0)}d + 4R_{\mathsf{id}}}{c^{(0)} + d} \bmod N\,,$$
$$\bar{c}^{(0)} = \bar{t} + \frac{uR_{\mathsf{id}}}{\bar{t}} \bmod N\,, \quad \bar{c}^{(1)} = \frac{\bar{c}^{(0)}d + 4uR_{\mathsf{id}}}{\bar{c}^{(0)} + d} \bmod N \ .$$

It chooses random bits $\beta_1, \beta_2 \in \{0,1\}$ and sets $c = c^{(\beta_1)}$ and $\bar{c} = \bar{c}^{(\beta_2)}$. The returned ciphertext is $C = (c, \bar{c})$.

$\texttt{DECRYPT}_{\mathsf{usk}}(C)$ Let $R_{\mathsf{id}} = \mathcal{H}_d(\mathsf{id})$. From $\mathsf{usk} = \{r_{\mathsf{id}}\}$ and $C = (c, \bar{c})$, if ${r_{\mathsf{id}}}^2 \equiv R_{\mathsf{id}} \pmod N$, $\texttt{DECRYPT}$ sets $\gamma = c$ and $\Delta = R_{\mathsf{id}}$; otherwise it sets $\gamma = \bar{c}$ and $\Delta = uR_{\mathsf{id}}$. Next, it computes $\sigma = \left(\frac{\gamma^2 - 4\Delta}{N}\right)$. Finally, it returns plaintext m as

$$m = \begin{cases} \left(\frac{\gamma + 2r_{\mathsf{id}}}{N}\right) & \text{if } \sigma = 1\,, \\ \left(\frac{(\gamma + 2r_{\mathsf{id}})(d - 2r_{\mathsf{id}})(d - \gamma)}{N}\right) & \text{if } \sigma = -1\,. \end{cases}$$

Remark 7. The choice $p \equiv -q \pmod 4$ (or equivalently $N \equiv 3 \pmod 4$) simplifies the setting. In this case, we know that $\left(\frac{-1}{p}\right) = -\left(\frac{-1}{q}\right)$ and therefore $\left(\frac{-1}{N}\right) = -1$. A nice observation is that $d = 0$ is a valid parameter when $N \equiv 3 \pmod 4$ since then $\left(\frac{d^2 - 4\mathcal{H}(\mathsf{id})}{N}\right) = \left(\frac{d^2 - 4u\mathcal{H}(\mathsf{id})}{N}\right) = \left(\frac{-1}{N}\right) = -1$ as desired. Any cryptographic hash function $\mathcal{H}$ mapping bit-strings to $\mathbb{J}_N$ can be used.

6.3 Security analysis

The two next propositions assess the security of the scheme under the quadratic residuosity assumption.

Proposition 3. *The scheme of § 6.2 is* IND-ID-CPA *under the quadratic residuosity assumption in the random oracle model.*

Proof. Assume there exists an IND-ID-CPA adversary $\mathcal{A}$ against the previous scheme (§ 6.2). We can then use $\mathcal{A}$ to break the semantic security of the generalized Cocks' scheme (§ 3.1), which in turn will contradict the quadratic residuosity assumption. This is readily verified by observing that if $C = (c, \bar{c})$ is a valid ciphertext for the scheme of § 3.1 for some user i with identity id then $C' = (c', \bar{c}')$ is a valid ciphertext for the scheme of § 6.2, where $c' = c$ with probability $1/2$ and $c' = \frac{cd+4R_{\mathsf{id}}}{c+d} \bmod N$ with probability $1/2$ and, likewise, $\bar{c}' = \bar{c}$ with probability $1/2$ and $\bar{c}' = \frac{\bar{c}d+4R_{\mathsf{id}}}{\bar{c}+d} \bmod N$ with probability $1/2$. □

Before proving that the scheme is anonymous, we need the following lemma.

Lemma 2. *Let* RSAgen *be a probabilistic algorithm which, given a security parameter κ, outputs primes p and q and their product $N = pq$. Let also δ be a random element in $(\mathbb{Z}/N\mathbb{Z})^\times$ and $\Delta = \delta^2 \bmod N$. Then, under the quadratic residuosity assumption,*

$$\left|\Pr\left[\mathcal{D}(x, \Delta, N) = 1 \mid x \xleftarrow{R} (\widehat{\mathcal{Z}}_{N,\Delta})^2\right] - \Pr\left[\mathcal{D}(x, \Delta, N) = 1 \mid x \xleftarrow{R} \widehat{\mathcal{Z}}^{[+1]}_{N,\Delta} \setminus (\widehat{\mathcal{Z}}_{N,\Delta})^2\right]\right|$$

is negligible for any probabilistic polynomial-time distinguisher $\mathcal{D}$; the probabilities are taken over the experiment of running $(N, p, q) \leftarrow \mathsf{RSAgen}(1^\kappa)$, sampling $\delta \xleftarrow{R} (\mathbb{Z}/N\mathbb{Z})^\times$, and choosing at random $x \in (\widehat{\mathcal{Z}}_{N,\Delta})^2$ and $x \in \widehat{\mathcal{Z}}^{[+1]}_{N,\Delta} \setminus (\widehat{\mathcal{Z}}_{N,\Delta})^2$.

Proof. As previously shown in § 6.1, we have that $(\widehat{\mathcal{Z}}_{N,\Delta})^2 = \mathcal{S}_{N,\Delta}$ (i.e., the set of all valid components of Cocks ciphertexts). The lemma now follows as an immediate application of [2, Lemma 2]. □

Proposition 4. *The scheme § 6.2 is* ANO-ID-CPA *under the quadratic residuosity assumption in the random oracle model.*

Proof. As mentioned in § 2.2, since the scheme is already known to be IND-ID-CPA, it suffices to prove that the statistical distance between the two distributions

$$D_0 = \left\{(\mathsf{id}_0^*, \mathsf{id}_1^*, \texttt{ENCRYPT}_{\mathsf{mpk}}(\mathsf{id}_0^*, m)) \mid m \xleftarrow{R} \{\pm 1\}\right\}$$

and

$$D_1 = \left\{(\mathsf{id}_0^*, \mathsf{id}_1^*, \texttt{ENCRYPT}_{\mathsf{mpk}}(\mathsf{id}_1^*, m)) \mid m \xleftarrow{R} \{\pm 1\}\right\}$$

is negligible. In our case, a ciphertext encrypted for identity id_b^* (with $b \in \{0, 1\}$) is of the form $C_b = (c_b, \bar{c}_b)$. From Lemma 1, only the valid component can

help in distinguishing D_0 from D_1. Without loss of generality, we assume that $\mathcal{H}(\mathsf{id}_0^*) = \mathcal{H}(\mathsf{id}_1^*) \in \mathbb{QR}_N$. Letting $\Delta_b = \mathcal{H}(\mathsf{id}_b^*)$, the valid component is then

$$c_b := c_b^{(0)} = t + \frac{\Delta_b}{t} \bmod N \quad \text{or} \quad c_b := c_b^{(1)} = \frac{c_b^{(0)} d + 4\Delta_b}{c_b^{(0)} + d} \bmod N$$

where $\left(\frac{t}{N}\right) = m$.

Omitting $\mathsf{id}_0^*, \mathsf{id}_1^*$ to ease the reading, the above criterion requires that the distributions $D_0 = \{c_0^{(\beta)} \mid \beta \xleftarrow{R} \{0,1\}\}$ and $D_1 = \{c_1^{(\beta)} \mid \beta \xleftarrow{R} \{0,1\}\}$ —or equivalently rescaling by a factor of two, that the distributions

$$D_0^* = \left\{\tfrac{c_0^{(\beta)}}{2} \mid \beta \xleftarrow{R} \{0,1\}\right\} \quad \text{and} \quad D_1^* = \left\{\tfrac{c_1^{(\beta)}}{2} \mid \beta \xleftarrow{R} \{0,1\}\right\}$$

must be indistinguishable with overwhelming probability. Using the $\circledast$ operator, the elements of D_b^* (for $b \in \{0,1\}$) are

$$\tfrac{c_b^{(\beta)}}{2} = \begin{cases} \frac{1}{2}\left(t + \frac{\Delta_b}{t}\right) = \frac{t^2+\Delta_b}{2t} = t \circledast t & \text{when } \beta = 0 \\ \frac{1}{2}\left(\frac{c_b^{(0)} d + 4\Delta_b}{c_b^{(0)} + d}\right) = \frac{\frac{c_b^{(0)}}{2}\frac{d}{2} + \Delta_b}{\frac{c_b^{(0)}}{2} + \frac{d}{2}} = \frac{c_b^{(0)}}{2} \circledast \frac{d}{2} = (t \circledast t) \circledast \frac{d}{2} & \text{when } \beta = 1 \end{cases}$$

where $t \xleftarrow{R} \{t \in (\mathbb{Z}/N\mathbb{Z})^\times \mid \gcd(t^2 - \Delta_b, N) = 1\}$. Hence, we can see that

$$D_b^* = \begin{cases} \left\{u \mid u \xleftarrow{R} (\widehat{\mathcal{Z}}_{N,\Delta_b})^2\right\} \overset{c}{\equiv} \left\{u \mid u \xleftarrow{R} \widehat{\mathcal{Z}}_{N,\Delta_b}^{[+1]}\right\} & \text{when } \beta = 0 \\ \left\{u \mid u \xleftarrow{R} \widehat{\mathcal{Z}}_{N,\Delta_b}^{[-1]}\right\} & \text{when } \beta = 1 \end{cases}.$$

The first assertion (when $\beta = 0$) follows from Lemma 2 (the notation $\overset{c}{\equiv}$ means computationally equivalent —under the QR assumption in this case). The second assertion (when $\beta = 1$) follows by noting that the Jacobi symbol $\left(\frac{\frac{d^2}{4} - \Delta_b}{N}\right) = -1$ and thus $\frac{d}{2} \in \widehat{\mathcal{Z}}_{N,\Delta_b}^{[-1]}$.

As a consequence, under the QR assumption, the distribution D_b^* appears indistinguishable from the uniform distribution over $\widehat{\mathcal{Z}}_{N,\Delta_b}^{[+1]} \cup \widehat{\mathcal{Z}}_{N,\Delta_b}^{[-1]} = \widehat{\mathcal{Z}}_{N,\Delta_b}$. This concludes the proof by noting that D_0^* and D_1^* are essentially the same sets: any random element is D_0^* is also an element in D_1^*, and vice-versa. □

Altogether, this proves that the scheme achieves ANO-IND-ID-CPA under the quadratic residuosity assumption in the random oracle model.

7 Conclusion

Somewhat surprisingly, we identified and detailed the algebraic group structure underlying Cocks encryption. The knowledge of this structure gives a better understanding of Cocks' scheme and allows one to see it differently. In particular, the hidden homomorphism opens the way to applications that were before not readily available or possible with Cocks' scheme, including homomorphic computations or anonymous encryption.

Acknowledgments I am grateful to Michael Clear for sending a copy of [11] and to Dan Boneh for useful discussions.

References

1. Abdalla, M., Bellare, M., Catalano, D., Kiltz, E., Kohno, T., Lange, T., Malone-Lee, J., Neven, G., Paillier, P., Shi, H.: Searchable encryption revisited: Consistency properties, relation to anonymous IBE, and extensions. J. Cryptology 21(3), 350–391 (2008)
2. Ateniese, G., Gasti, P.: Universally anonymous IBE based on the quadratic residuosity assumption. In: Fischlin, M. (ed.) Topics in Cryptology – CT-RSA 2009. LNCS, vol. 5473, pp. 32–47. Springer (2009)
3. Barth, A., Boneh, D., Waters, B.: Privacy in encrypted content distribution using private broadcast encryption. In: Di Crescenzo, G., Rubin, A. (eds.) Financial Cryptography and Data Security. LNCS, vol. 4107, pp. 52–64. Springer (2006)
4. Bellare, M., Boldyreva, A., Desai, A., Pointcheval, D.: Key-privacy in public-key encryption. In: Boyd, C. (ed.) Advances in Cryptology – ASIACRYPT 2001. LNCS, vol. 2248, pp. 566–582. Springer (2001)
5. Bellare, M., Rogaway, P.: Random oracles are practical: A paradigm for designing efficient protocols. In: 1st ACM Conference on Computer and Communications Security. pp. 62–73. ACM Press (1993)
6. Boneh, D., Di Crescenzo, G., Ostrovsky, R., Persiano, G.: Public key encryption with keyword search. In: Cachin, C., Camenisch, J. (eds.) Advances in Cryptology – EUROCRYPT 2004. LNCS, vol. 3027, pp. 506–522. Springer (2004)
7. Boneh, D., Franklin, M.K.: Identity-based encryption from the Weil pairing. SIAM J. Comput. 32(3), 586–615 (2003)
8. Boneh, D., Gentry, C., Hamburg, M.: Space-efficient identity based encryption without pairings. In: 48th Annual IEEE Symposium on Foundations of Computer Science (FOCS 2007). pp. 647–657. IEEE Computer Society (2007)
9. Boneh, D., LaVigne, R., Sabin, M.: Identity-based encryption with e^{th} residuosity and its incompressibility. In: Autumn 2013 TRUST Conference. Washington DC (Oct 9-10, 2013), poster presentation
10. Clear, M., Hughes, A., Tewari, H.: Homomorphic encryption with access policies: Characterization and new constructions. In: Youssef, A., Nitaj, A., Hassanien, A.E. (eds.) Progress in Cryptology – AFRICACRYPT 2013. LNCS, vol. 7918, pp. 61–87. Springer (2013)
11. Clear, M., Tewari, H., McGoldrick, C.: Anonymous IBE from quadratic residuosity with improved performance. In: Pointcheval, D., Vergnaud, D. (eds.) Progress in Cryptology – AFRICACRYPT 2014. LNCS, vol. 8469, pp. 377–397. Springer (2014)
12. Cocks, C.: An identity based encryption scheme based on quadratic residues. In: Honary, B. (ed.) Cryptography and Coding. LNCS, vol. 2260, pp. 360–363. Springer (2001)
13. Desmedt, Y., Quisquater, J.J.: Public-key systems based on the difficulty of tampering (Is there a difference between DES and RSA?). In: Odlyzko, A.M. (ed.) Advances in Cryptology – CRYPTO '86. LNCS, vol. 263, pp. 111–117. Springer (1987)
14. Di Crescenzo, G., Saraswat, V.: Public key encryption with searchable keywords based on Jacobi symbols. In: Srinathan, K., Rangan, C., Yung, M. (eds.) Progress in Cryptology – INDOCRYPT 2007. LNCS, vol. 4859, pp. 282–296. Springer (2007)

15. van Dijk, M., Woodruff, D.: Asymptotically optimal communication for torus-based cryptography. In: Franklin, M. (ed.) Advances in Cryptology – CRYPTO 2004. LNCS, vol. 3152, pp. 157–178. Springer-Verlag (2004)
16. Gentry, C., Peikert, C., Vaikuntanathan, V.: Trapdoors from hard lattices and new cryptographic constructions. In: Dwork, C. (ed.) Proc. of the 40th Annua ACM Symposium on Theory of Computing (STOC '08). pp. 197–206. ACM Press (2008)
17. Goldwasser, S., Micali, S.: Probabilistic encryption. J. Comput. Syst. Sci. 28(2), 270–299 (1984)
18. Halevi, S.: A sufficient condition for key-privacy. IACR Cryptology ePrint Archive, Report 2005/005 (2005)
19. Jhanwar, M.P., Barua, R.: A variant of Boneh-Gentry-Hamburg's pairing-free identity based encryption scheme. In: Yung, M., Liu, P., Lin, D. (eds.) Information Security and Cryptology (Inscrypt 2008). LNCS, vol. 5487, pp. 314–331. Springer (2009)
20. Joye, M., Neven, G. (eds.): Identity-Based Cryptography, Cryptology and Information Security Series, vol. 2. IOS Press (2009)
21. Kiayias, A., Tsiounis, Y., Yung, M.: Group encryption. In: Kurosawa, K. (ed.) Advances in Cryptology – ASIACRYPT 2007. LNCS, vol. 4833, pp. 181–199. Springer (2007)
22. Rubin, K., Silverberg, A.: Torus-based cryptography. In: Boneh, D. (ed.) Advances in Cryptology – CRYPTO 2003. LNCS, vol. 2729, pp. 349–365. Springer (2003)
23. Rubin, K., Silverberg, A.: Compression in finite fields and torus-based cryptography. SIAM J. Comput. 37(5), 1401–1428 (2008)
24. Sakai, R., Kasahara, M.: ID based cryptosystems with pairing on elliptic curve. IACR Cryptology ePrint Archive, Report 2003/054 (2003)
25. Shamir, A.: Identity-based cryptosystems and signature schemes. In: Blakley, G.R., Chaum, D. (eds.) Advances in Cryptology, Proceedings of CRYPTO '84. LNCS, vol. 196, pp. 47–53. Springer (1985)

A Proof of Proposition 1

Let $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ be adversary that can break the IND-ID-CPA security of the generalized scheme described in § 3.1 with probability ϵ. We will use $\mathcal{A}$ to decide whether a random element w in $\mathbb{J}_N$ is quadratic residue modulo N or not.

A.1 First case: u is universally fixed

The firt case assumes that system parameter u is a universally fixed parameter. It covers the original Cocks' scheme wherein $p, q \equiv 3 \pmod 4$ and $u = -1 \in \mathbb{J}_N \setminus \mathbb{QR}_N$.

Let $\mathcal{H}\colon \{0,1\}^* \to \mathbb{J}_N$ be a hash function viewed as a random oracle. Consider the following distinguisher[2] $\mathcal{D}(w, u, N)$ for solving the QR problem. The goal of $\mathcal{D}$ is to distinguish a random element $w \in \mathbb{QR}_N$ from a random element $w \in \mathbb{J}_N \setminus \mathbb{QR}_N$.

[2] Note that $\mathcal{D}$ is given $u \in \mathbb{J}_N \setminus \mathbb{QR}_N$ as it is universally fixed.

1. Set $\mathsf{mpk} = \{N, u, \mathcal{H}\}$, and give mpk to $\mathcal{A}_1^{\mathtt{EXTRACT}_{\mathsf{msk}}(\cdot),\mathcal{H}(\cdot)}$ —$\mathcal{A}_1$ has oracle access to $\mathtt{EXTRACT}_{\mathsf{msk}}(\cdot)$ and $\mathcal{H}(\cdot)$, it may issue a number of extraction and hash queries, after what it selects a target identity id^*;
2. Depending on id^*:
   (a) If $\mathcal{H}(\mathsf{id}^*) = w$ then
      i. Choose a random bit $b \in \{0,1\}$, let $R_{\mathsf{id}^*} = \mathcal{H}(\mathsf{id}^*)$, and compute the encryption of $(-1)^b$ as $C_b = (c_b, \bar{c}_b)$ where
      $$c_b = t + \frac{R_{\mathsf{id}^*}}{t} \bmod N\,, \quad \bar{c}_b = \bar{t} + \frac{uR_{\mathsf{id}^*}}{\bar{t}} \bmod N\,,$$
      for some random elements $t, \bar{t} \in \mathbb{Z}/N\mathbb{Z}$ such that $\left(\frac{t}{N}\right) = (-1)^b$ and $\left(\frac{\bar{t}}{N}\right) = (-1)^{1-b}$;
      ii. Give s and $C_b = (c_b, \bar{c}_b)$ to $\mathcal{A}_2^{\mathtt{EXTRACT}_{\mathsf{msk}}(\cdot),\mathcal{H}(\cdot)}$ —$\mathcal{A}_2$ may issue more extraction and hash queries, after what it returns its guess b';
      iii. If $b' = b$ return 1; otherwise return 0.
   (b) If $\mathcal{H}(\mathsf{id}^*) \neq w$ then
      i. Choose a random bit $b' \in \{0,1\}$;
      ii. Return b'.

It remains to detail how $\mathcal{D}$ simulates answers to oracle queries. $\mathcal{D}$ maintains a history list $\mathsf{Hist}[\mathcal{H}]$ composed of triplets. The list is initialized to $\emptyset$. It also maintains a counter k initialized to 0. Let q_{H_1} denote the number of hash queries that are not followed by extract queries and let q_{E_1} denote the number of extract queries, made by $\mathcal{A}_1$. Without loss of generality, we assume that $\mathcal{A}_1$ issues a hash query on id^*. Finally, we let k_1 denote a random integer in $\{1, \ldots, q_{H_1} + q_{E_1}\}$ chosen by $\mathcal{D}$.

Hash queries When $\mathcal{A}$ queries oracle $\mathcal{H}$ on some id, $\mathcal{D}$ checks whether there is an entry of the form (id, h, r) in $\mathsf{Hist}[\mathcal{H}]$; i.e., a triplet with id as the first component. If so, it returns h. Otherwise, it does the following:
1. Increment k;
2. Depending on k:
   (a) If $k = k_1$, define $h = w$ and append $(\mathsf{id}, h, \perp)$ to $\mathsf{Hist}[\mathcal{H}]$;
   (b) Else (if $k \neq k_1$), define $h = u^{-j} r^2 \bmod N$ with $r \xleftarrow{R} (\mathbb{Z}/N\mathbb{Z})^\times$ and $j \xleftarrow{R} \{0,1\}$ and append (id, h, r) to $\mathsf{Hist}[\mathcal{H}]$;
3. Return h.

Extraction queries When $\mathcal{A}$ queries oracle $\mathtt{EXTRACT}$ on some id, $\mathcal{D}$ checks whether there is an entry of the form (id, h, r) in $\mathsf{Hist}[\mathcal{H}]$. If not, it calls $\mathcal{H}(\mathsf{id})$ so that there is an entry. Let (id, h, r) denote the entry in $\mathsf{Hist}[\mathcal{H}]$ corresponding to id. Depending on it, $\mathcal{D}$ does the following:
1. If $r \neq \perp$ then return r;
2. If $r = \perp$ then abort.

We now analyze the success probability of $\mathcal{D}$ in solving the QR challenge. Since u is an element in $\mathbb{J}_N \setminus \mathbb{QR}_N$, the resulting mpk appear as valid system parameters. Three subcases can be distinguished.

Subcase i The first subcase supposes $w = \mathcal{H}(\mathsf{id}^*) \in \mathbb{QR}_N$. The condition $w = \mathcal{H}(\mathsf{id}^*)$ requires that id^* is the k_1-th query to $\mathcal{H}$. Further, since $\mathcal{H}(\mathsf{id}^*) \in \mathbb{QR}_N$, Lemma 1 teaches that $C_b = (c_b, \bar{c}_b)$ is a valid ciphertext for b. Namely, component c_b correctly decrypts to $(-1)^b$ and component $\bar{c}_b$ is of no use. Hence, $\mathcal{D}$ returns 1 exactly when $\mathcal{A}$ wins in the IND-ID-CPA game, provided that there is no abort. But since $\mathcal{A}$ is not allowed to submit id^* to `EXTRACT` (and so there is no abort when id^* is the k_1-th query), we get $\Pr[\mathcal{D}(w,u,N) = 1 \mid w \in \mathbb{QR}_N \wedge w = \mathcal{H}(\mathsf{id}^*)] = \epsilon$.

Subcase ii The second subcase supposes $w = \mathcal{H}(\mathsf{id}^*) \in \mathbb{J}_N \setminus \mathbb{QR}_N$. Since $\mathcal{H}(\mathsf{id}^*) \in \mathbb{J}_N \setminus \mathbb{QR}_N$, Lemma 1 teaches that $C_b = (c_b, \bar{c}_b)$ is a valid ciphertext for $(-1)^{(1-b)}$ —it is worth noticing that $\left(\frac{\bar{t}}{N}\right) = (-1)^{1-b}$. Hence, $\mathcal{D}$ returns 1 exactly when $\mathcal{A}$ looses in the IND-ID-CPA game. We therefore get $\Pr[\mathcal{D}(w,u,N) = 1 \mid w \in \mathbb{J}_N \setminus \mathbb{QR}_N \wedge w = \mathcal{H}(\mathsf{id}^*)] = 1 - \epsilon$.

Subcase iii The last subcase supposes $w \neq \mathcal{H}(\mathsf{id}^*)$. In this case $\mathcal{D}$ returns a random bit, regardless of w. Therefore, we have $\Pr[\mathcal{D}(w,u,N) = 1 \mid w \in \mathbb{QR}_N \wedge w \neq \mathcal{H}(\mathsf{id}^*)] = \Pr[\mathcal{D}(w,u,N) = 1 \mid w \in \mathbb{J}_N \setminus \mathbb{QR}_N \wedge w \neq \mathcal{H}(\mathsf{id}^*)] = 1/2$.

We so obtain:

$$\begin{aligned}
&\Pr\big[\mathcal{D}(w,u,N) = 1 \mid w \in \mathbb{QR}_N\big] \\
&\quad = \Pr\big[w = \mathcal{H}(\mathsf{id}^*)\big] \cdot \Pr\big[\mathcal{D}(w,u,N) = 1 \mid w \in \mathbb{QR}_N \wedge w = \mathcal{H}(\mathsf{id}^*)\big] \\
&\qquad + \Pr\big[w \neq \mathcal{H}(\mathsf{id}^*)\big] \cdot \Pr\big[\mathcal{D}(w,u,N) = 1 \mid w \in \mathbb{QR}_N \wedge w \neq \mathcal{H}(\mathsf{id}^*)\big] \\
&\quad = \frac{1}{q_{H_1}} \cdot \epsilon + \left(1 - \frac{1}{q_{H_1}}\right) \cdot \frac{1}{2} = \frac{1}{2} + \frac{\epsilon - \frac{1}{2}}{q_{H_1}}
\end{aligned}$$

and similarly,

$$\begin{aligned}
\Pr\big[\mathcal{D}(w,u,N) = 1 \mid w \in \mathbb{J}_N \setminus \mathbb{QR}_N\big] &= \frac{1}{q_{H_1}} \cdot (1 - \epsilon) + \left(1 - \frac{1}{q_{H_1}}\right) \cdot \frac{1}{2} \\
&= \frac{1}{2} + \frac{\frac{1}{2} - \epsilon}{q_{H_1}} \ .
\end{aligned}$$

Putting all together, we get:

$$\begin{aligned}
&\big|\Pr\big[\mathcal{D}(w,u,N) = 1 \mid w \in \mathbb{QR}_N\big] - \Pr\big[\mathcal{D}(w,u,N) = 1 \mid w \in \mathbb{J}_N \setminus \mathbb{QR}_N\big]\big| \\
&\qquad = \frac{2}{q_{H_1}} \left|\epsilon - \frac{1}{2}\right|
\end{aligned}$$

which must be negligible by the QR assumption. As a consequence, $|\epsilon - \frac{1}{2}|$ must be negligible, which means that the scheme is IND-ID-CPA secure under the QR assumption.

A.2 Second case: u is random

In this case, the proof can be obtained along the lines of the proof offered in [8, Appendix B.2] for the Boneh-Gentry-Hamburg scheme. The proof features a

tight reduction. It however crucially requires that parameter u is defined as a random element in $\mathbb{J}_N \setminus \mathbb{QR}_N$.

B Arithmetic in $\mathcal{Z}_{N,\Delta}$

As mentioned in § 4.2, each element u of the group $\mathcal{Z}_{N,\Delta} = \mathcal{F}_{p,\Delta} \times \mathcal{F}_{q,\Delta}$ can be uniquely represented by a pair $[u_p, u_q]$ with $u_p \in \mathcal{F}_{p,\Delta}$ and $u_q \in \mathcal{F}_{q,\Delta}$, and $\infty = [\infty_p, \infty_q]$. There is a slight complication when doing arithmetic in $\mathcal{Z}_{N,\Delta}$ as we need to deal with the elements of the form $[u_p, \infty_q]$ or $[\infty_p, u_q]$. This can be circumvented by adopting a projective representation. An element $u \in \mathcal{Z}_{N,\Delta}$ can be written as a pair $(U : Z)$. We say that two elements $u = (U : Z)$ and $u' = (U' : Z')$ are equivalent if there exists some $\lambda \in (\mathbb{Z}/N\mathbb{Z})^\times$ such that $U' = \lambda U$ and $Z' = \lambda Z$. Hence, from the definition of ψ^{-1}, we can represent $\mathcal{Z}_{N,\Delta}$ as

$$\mathcal{Z}_{N,\Delta} = \left\{ (\delta(v+1) : v-1) \mid v \in (\mathbb{Z}/N\mathbb{Z})^\times \right\} \ .$$

The neutral element is $\infty = (1 : 0)$. The inverse of an element $(U : Z)$ is $(-U : Z)$. The product of two elements $(U_1 : Z_1), (U_2, Z_2) \in \mathcal{Z}_{N,\Delta}$ is given by

$$(U_1 : Z_1) \circledast (U_2, Z_2) = (U_1U_2 + \Delta Z_1 Z_2 : U_1 Z_2 + U_2 Z_1) \ .$$

Observe that the group law is complete with the projective representation: it works for all inputs.

Another way to deal with the elements of the form $[u_p, \infty_q]$ or $[\infty_p, u_q]$ is simply to ignore them and to work in the subset

$$\widetilde{\mathcal{Z}}_{N,\Delta} = \left((\mathcal{F}_{p,\Delta} \setminus \{\infty_p\}) \times (\mathcal{F}_{q,\Delta} \setminus \{\infty_q\}) \right) \cup \{\infty\} \tag{5}$$

$$= \{ u \in \mathbb{Z}/N\mathbb{Z} \mid \gcd(u^2 - \Delta, N) = 1 \} \cup \{\infty\} \ . \tag{6}$$

Wherever it is defined, the $\circledast$-law in $\widetilde{\mathcal{Z}}_{N,\Delta}$ coincides with the group law on $\mathcal{Z}_{N,\Delta}$:

$$u_1 \circledast u_2 = \begin{cases} \frac{u_1 u_2 + \Delta}{u_1 + u_2} \pmod N & \text{if } u_1 \neq -u_2 \\ \infty & \text{otherwise} \end{cases} \ .$$

(If $u_1 = \infty$ then $u_1 \circledast u_2 = u_2$; if $u_2 = \infty$ then $u_1 \circledast u_2 = u_1$.)

C Some Variants of Cocks' Scheme

The **HOM** is dependent of the cryptosystem. We propose below some variants of Cocks' scheme that leads to better efficiency. In particular, obtaining the encryption of the complementary value is almost free.

C.1 Basic scheme

$\texttt{SETUP}(1^\kappa)$ Given a security parameter κ, SETUP generates an RSA modulus $N = pq$ where p and q are prime. It also selects an element $u \in \mathbb{J}_N \setminus \mathbb{QR}_N$. The public system parameters are $\mathsf{mpk} = \{N, u, \mathcal{H}\}$ where $\mathcal{H}$ is a cryptographic hash function mapping bit-strings to $\mathbb{J}_N$; i.e., $\mathcal{H}\colon \{0,1\}^* \to \mathbb{J}_N$. The master secret key is $\mathsf{msk} = \{p, q\}$.

$\texttt{EXTRACT}_{\mathsf{msk}}(\mathsf{id})$ Given identity id, key derivation algorithm EXTRACT sets $R_{\mathsf{id}} = \mathcal{H}(\mathsf{id})$. If $R_{\mathsf{id}} \in \mathbb{QR}_N$ it computes $r_{\mathsf{id}} = {R_{\mathsf{id}}}^{1/2} \bmod N$; otherwise it computes $r_{\mathsf{id}} = (uR_{\mathsf{id}})^{1/2} \bmod N$. EXTRACT returns user's private key $\mathsf{usk} = \{r_{\mathsf{id}}\}$.

$\texttt{ENCRYPT}_{\mathsf{mpk}}(\mathsf{id}, m)$ To encrypt a message $m \in \{\pm 1\}$ for a user with identity id, ENCRYPT defines $R_{\mathsf{id}} = \mathcal{H}(\mathsf{id})$. It chooses at random $t, \bar{t} \in \mathbb{Z}/N\mathbb{Z}$ and computes

$$\varepsilon = m \cdot \left(\frac{t}{N}\right), \quad c = t + \frac{R_{\mathsf{id}}}{t} \bmod N, \quad \bar{\varepsilon} = m \cdot \left(\frac{\bar{t}}{N}\right), \quad \bar{c} = \bar{t} + \frac{uR_{\mathsf{id}}}{\bar{t}} \bmod N .$$

The returned ciphertext is $C = (\varepsilon, c, \bar{\varepsilon}, \bar{c})$.

$\texttt{DECRYPT}_{\mathsf{usk}}(C)$ From $\mathsf{usk} = \{r_{\mathsf{id}}\}$ and $C = (\varepsilon, c, \bar{\varepsilon}, \bar{c})$, if ${r_{\mathsf{id}}}^2 \equiv \mathcal{H}(\mathsf{id}) \pmod N$, DECRYPT sets $\nu = \varepsilon$ and $\gamma = c$; otherwise it sets $\nu = \bar{\varepsilon}$ and $\gamma = \bar{c}$. Next it computes $\tau = \left(\frac{\gamma + 2r_{\mathsf{id}}}{N}\right)$ using secret key r_{id} and returns plaintext $m = \nu \cdot \tau$.

Homomorphic computation Let $C_1 = (\varepsilon_1, c_1, \bar{\varepsilon}_1, \bar{c}_1)$ and $C_2 = (\varepsilon_2, c_2, \bar{\varepsilon}_2, \bar{c}_2)$ be the respective encryption of messages m_1 and m_2 for a user with identity id. Then, letting $R_{\mathsf{id}} = \mathcal{H}(\mathsf{id})$ and $\bar{R}_{\mathsf{id}} = u \cdot \mathcal{H}(\mathsf{id}) \bmod N$, we get that $C_3 = (\varepsilon_3, c_3, \bar{\varepsilon}_3, \bar{c}_3)$ with

$$\varepsilon_3 = \varepsilon_1 \cdot \varepsilon_2 \cdot \left(\frac{c_1 + c_2}{N}\right), \quad c_3 = \frac{c_1 c_2 + 4R_{\mathsf{id}}}{c_1 + c_2} \bmod N,$$
$$\bar{\varepsilon}_3 = \bar{\varepsilon}_1 \cdot \bar{\varepsilon}_2 \cdot \left(\frac{\bar{c}_1 + \bar{c}_2}{N}\right), \quad \text{and} \quad \bar{c}_3 = \frac{\bar{c}_1 \bar{c}_2 + 4\bar{R}_{\mathsf{id}}}{\bar{c}_1 + \bar{c}_2} \bmod N$$

is the encryption of message $m_3 = m_1 \cdot m_2$ (for the user with identity id).

Complementary encryption Given the encryption of a message $m \in \{\pm 1\}$, it is easy to get the encryption of the complementary value. If $C = (\varepsilon, c, \bar{\varepsilon}, \bar{c})$ is the encryption of $m \in \{\pm 1\}$ then $C' = (-\varepsilon, c, -\bar{\varepsilon}, \bar{c})$ is the encryption of $-m$.

C.2 Compact variant

As an illustration, suppose that $R_{\mathsf{id}} = \mathcal{H}(\mathsf{id}) \in \mathbb{QR}_N$ in the previous scheme. If $C = (\varepsilon, c)$ with $\varepsilon = m \cdot \left(\frac{t}{N}\right)$ and $c = t + R_{\mathsf{id}}/t \bmod N$ is a valid encryption for message $m \in \{\pm 1\}$ then so is $C' := (\varepsilon', c')$ where $\varepsilon' = \varepsilon \cdot \left(\frac{-1}{N}\right)$ and $c' = N - c$.

Indeed, letting $t' = -t \bmod N$, we have

$$c' \equiv -c \equiv -\left(t + \frac{R_i}{t}\right) \equiv t' + \frac{R_i}{t'} \pmod{N} \quad \text{and}$$
$$\varepsilon' = m \cdot \left(\frac{t'}{N}\right) = m \cdot \left(\frac{-t}{N}\right) = \varepsilon \cdot \left(\frac{-1}{N}\right) .$$

One of the two equivalent ciphertexts $C = (\varepsilon, c)$ and $C' = (\varepsilon', c')$ is (at least) one bit shorter than the other one.

As a result, if ℓ represents the bit-length of RSA modulus N this allows reducing the size a whole ciphertext to at most 2ℓ bits, as in Cocks' scheme.

$\texttt{SETUP}(1^\kappa)$ Given a security parameter κ, SETUP generates an RSA modulus $N = pq$ where p and q are prime. It also selects an element $u \in \mathbb{J}_N \setminus \mathbb{QR}_N$. The public system parameters are $\mathsf{mpk} = \{N, u, \mathcal{H}\}$ where $\mathcal{H}$ is a cryptographic hash function mapping bit-strings to $\mathbb{J}_N$; i.e., $\mathcal{H}\colon \{0,1\}^* \to \mathbb{J}_N$. The master secret key is $\mathsf{msk} = \{p, q\}$.

$\texttt{EXTRACT}_{\mathsf{msk}}(\mathsf{id})$ Given identity id, key derivation algorithm EXTRACT sets $R_{\mathsf{id}} = \mathcal{H}(\mathsf{id})$. If $R_{\mathsf{id}} \in \mathbb{QR}_N$ it computes $r_{\mathsf{id}} = {R_{\mathsf{id}}}^{1/2} \bmod N$; otherwise it computes $r_{\mathsf{id}} = (uR_{\mathsf{id}})^{1/2} \bmod N$. EXTRACT returns user's private key $\mathsf{usk} = \{r_{\mathsf{id}}\}$.

$\texttt{ENCRYPT}_{\mathsf{mpk}}(\mathsf{id}, m)$ To encrypt a message $m \in \{\pm 1\}$ for a user with identity id, ENCRYPT defines $R_{\mathsf{id}} = \mathcal{H}(\mathsf{id})$. It chooses at random $t, \bar{t} \in \mathbb{Z}/N\mathbb{Z}$ and computes

$$\varepsilon' = m \cdot \left(\frac{t}{N}\right), \;\; c' = t + \frac{R_{\mathsf{id}}}{t} \bmod N, \;\; \bar{\varepsilon}' = m \cdot \left(\frac{\bar{t}}{N}\right), \;\; \bar{c}' = \bar{t} + \frac{uR_{\mathsf{id}}}{\bar{t}} \bmod N .$$

Define $c = \min(c', N - c')$ and $\bar{c} = \min(\bar{c}', N - \bar{c}')$. If $c = c'$ then define $\varepsilon = \varepsilon'$; otherwise define $\varepsilon = \varepsilon' \cdot \left(\frac{-1}{N}\right)$. Similarly, if $\bar{c} = \bar{c}'$ then define $\bar{\varepsilon} = \bar{\varepsilon}'$; otherwise define $\bar{\varepsilon} = \bar{\varepsilon}' \cdot \left(\frac{-1}{N}\right)$. The returned ciphertext is $C = (\varepsilon, c, \bar{\varepsilon}, \bar{c})$.

$\texttt{DECRYPT}_{\mathsf{usk}}(C)$ From $\mathsf{usk} = \{r_{\mathsf{id}}\}$ and $C = (\varepsilon, c, \bar{\varepsilon}, \bar{c})$, if ${r_{\mathsf{id}}}^2 \equiv \mathcal{H}(\mathsf{id}) \pmod{N}$, DECRYPT sets $\nu = \varepsilon$ and $\gamma = c$; otherwise it sets $\nu = \bar{\varepsilon}$ and $\gamma = \bar{c}$. Next it computes $\tau = \left(\frac{\gamma + 2r_{\mathsf{id}}}{N}\right)$ using secret key r_{id} and returns plaintext $m = \nu \cdot \tau$.

Remark 8. Assuming that primes p and q satisfy the extra condition $p \equiv q \pmod 4$ (for example if $N = pq$ is a Blum integer) —in which case $\left(\frac{-1}{N}\right) = 1$— the encryption algorithm can then form the ciphertext $C = (\varepsilon, c, \bar{\varepsilon}, \bar{c})$ more simply by defining $\varepsilon = m \cdot \left(\frac{t}{N}\right)$ and $\bar{\varepsilon} = m \cdot \left(\frac{\bar{t}}{N}\right)$.

D Public-Key Encryption with Keyword Search

A prominent application of anonymous IBE scheme resides in public-key encryption with keyword search (or PEKS) [6]. Basically, PEKS is a form of encryption

that allows searching on data that is encrypted using a public-key system. A typical application is for an email gateway to test whether or not the keyword "urgent" is present in an email. The gateway then routes the email if it is the case. Of course the gateway should only learn whether the word "urgent" is present but nothing else about the email. In the email use-case, another practical application is to test the sender's name of the email and to route the emails accordingly. Further applications for PEKS can be found in [6] and [1]. Of particular interest is the concept of temporarily searchable encryption [1, Section 6].

D.1 Definition

A *public-key encryption with keyword search scheme* [7] is defined as a tuple of four algorithms $(\texttt{KEYGEN}, \texttt{PEKS}, \texttt{TRAPDOOR}, \texttt{TEST})$:

Key generation The key generation algorithm KEYGEN is a randomized algorithm that takes as input some security parameter 1^κ and outputs a matching pair $(\mathsf{upk}, \mathsf{usk})$ of public key and private key: $(\mathsf{upk}, \mathsf{usk}) \xleftarrow{R} \texttt{KEYGEN}(1^\kappa)$.

Public-key encryption with keyword search (PEKS) Let $\mathcal{W}$ denote the keyword space. The PEKS algorithm PEKS takes as input a public key upk and a keyword $w \in \mathcal{W}$, and returns a searchable ciphertext S. We write $S \leftarrow \texttt{PEKS}_{\mathsf{upk}}(w)$.

Trapdoor The trapdoor algorithm TRAPDOOR takes as input the private key usk (corresponding to upk) and a keyword w, and returns a trapdoor T_w for keyword w. We write $T_w \leftarrow \texttt{TRAPDOOR}_{\mathsf{usk}}(w)$.

Test The test algorithm TEST takes as input a searchable ciphertext S and a trapdoor T_w, and returns a bit b. A bit b with 1 means "accept" or "yes", and a bit b with 0 means "reject" or "no". We write $b \leftarrow \texttt{TEST}(S, T_w)$.

It is required that $\texttt{TEST}(\texttt{PEKS}_{\mathsf{upk}}(w), \texttt{TRAPDOOR}_{\mathsf{usk}}(w)) = 1$ for all keywords $w \in \mathcal{W}$.

D.2 Public-key encryption with keyword search from quadratic residuosity

In a PEKS scheme, a sender can send messages in encrypted form to a receiver so that the receiver can allow a designated proxy to search keywords in the encrypted messages without incurring any (additional) loss of privacy. In [6], Boneh *et al.* suggest the following methodology:

- The sender encrypts the message being sent with a (regular) public-key cryptosystem;
- She appends to the resulting ciphertext a PEKS for each keyword.

In more detail, to encrypt a message m with searchable keywords $w_1, \ldots, w_n$ for the receiver with public key upk, the sender computes and sends

$$c = \texttt{ENCRYPT}_{\mathsf{upk}}(m)\,, \quad S_1 = \texttt{PEKS}_{\mathsf{upk}}(w_1)\,, \quad \ldots\,, \quad S_n = \texttt{PEKS}_{\mathsf{upk}}(w_n)\ .$$

The whole ciphertext is $C = \{c, S_1, \ldots, S_n\}$. Now if the receiver has given a proxy a trapdoor T_{w_j} for keyword w_j then this proxy can test whether the corresponding plaintext m contains the keyword w_j, but nothing more.

A conversion to turn an anonymous identity-based scheme (under certain conditions) into a PEKS scheme is developed in [6]. Some subsequent refinements are described in [1]. Applied to the scheme of § 6.2 as a building block, we so obtain a PEKS scheme based on the quadratic residuosity. For slightly better efficiency, instead of verifying whether $x_i = \mu^{-1}(\nu_i \cdot \tau_i)$ $(\in \{0,1\})$, for $0 \le i \le k-1$, the `TEST` algorithm equivalently verifies whether $\tau_i = \nu_i \cdot (1 - 2x_i)$. In detail, the scheme is as follows.

`KEYGEN`(1^κ) Given a security parameter κ, `KEYGEN` generates an RSA modulus $N = pq$ where p and q are prime. It defines a security parameter k depending on κ. It also selects an element $u \in \mathbb{J}_N \setminus \mathbb{QR}_N$ and a global integer d. The user's public key is $\mathsf{upk} = \{N, k, u, d, \mathcal{H}_d\}$ where $\mathcal{H}_d$ is a cryptographic hash function mapping bit-strings to $\mathbb{J}_N$ as per Eq. (4). The user's private key is $\mathsf{usk} = \{p, q\}$.

`PEKS`$_{\mathsf{upk}}(w)$ To encrypt a keyword $w \in \{0,1\}^*$, `PEKS` selects a k-bit integer $x = \sum_{i=0}^{k-1} x_i 2^i$ (with $x_i \in \{0,1\}$). It defines $R = \mathcal{H}_d(w)$.
For $i = 0, \ldots, k-1$, it does the following:

1. choose at random $t_i, \bar{t}_i \in \mathbb{Z}/N\mathbb{Z}$;
2. let

$$\varepsilon_i = (-1)^{x_i}\left(\frac{t_i}{N}\right), \quad c_i{}^{(0)} = t_i + \frac{R}{t_i} \bmod N, \quad c_i{}^{(1)} = \frac{c_i{}^{(0)} d + 4R}{c_i{}^{(0)} + d} \bmod N,$$
$$\bar{\varepsilon}_i = (-1)^{x_i}\left(\frac{\bar{t}_i}{N}\right), \quad \bar{c}_i{}^{(0)} = \bar{t}_i + \frac{uR}{\bar{t}_i} \bmod N, \quad \bar{c}_i{}^{(1)} = \frac{\bar{c}_i{}^{(0)} d + 4uR}{\bar{c}_i{}^{(0)} + d} \bmod N;$$

3. choose random bits $\beta_{1,i}, \beta_{2,i} \in \{0,1\}$ and set $c_i = c_i{}^{(\beta_{1,i})}$ and $\bar{c}_i = \bar{c}_i{}^{(\beta_{2,i})}$.

`PEKS` returns the searchable ciphertext $S = \{x, \varepsilon_0, c_0, \bar{\varepsilon}_0, \bar{c}_0, \ldots, \varepsilon_{k-1}, c_{k-1}, \bar{\varepsilon}_{k-1}, \bar{c}_{k-1}\}$.

`TRAPDOOR`$_{\mathsf{usk}}(w)$ Given keyword w, trapdoor algorithm `TRAPDOOR` sets $R = \mathcal{H}(w)$. If $R \in \mathbb{QR}_N$ it computes $T_w = R^{1/2} \bmod N$; otherwise it computes $T_w = (uR)^{1/2} \bmod N$. `TRAPDOOR` returns T_w.

`TEST`(S, T_w) For keyword w, `TEST` uses the trapdoor T_w. Let $R = \mathcal{H}_d(w)$. Given a searchable ciphertext $S = \{x, \varepsilon_0, c_0, \bar{\varepsilon}_0, \bar{c}_0, \ldots, \varepsilon_{k-1}, c_{k-1}, \bar{\varepsilon}_{k-1}, \bar{c}_{k-1}\}$, if ${T_w}^2 \equiv R \pmod N$, `TEST` sets $\nu_i = \varepsilon_i$, $\gamma_i = c_i$ for $0 \le i \le k-1$, and $\Delta = R$; otherwise it sets $\nu_i = \bar{\varepsilon}_i$, $\gamma_i = \bar{c}_i$ for $0 \le i \le k-1$, and $\Delta = uR$.
Next, for $i = 0, \ldots, k-1$, it does the following:

1. set $\sigma_i = \left(\frac{{\gamma_i}^2 - 4\Delta}{N}\right)$;
2. set

$$\tau_i = \begin{cases} \left(\frac{\gamma_i + 2T_w}{N}\right) & \text{if } \sigma_i = 1 \\ \left(\frac{(\gamma_i + 2T_w)(d - 2T_w)(d - \gamma_i)}{N}\right) & \text{if } \sigma_i = -1 \end{cases};$$

3. set $b_i = 1$ if $\tau_i = \nu_i \cdot (1 - 2x_i)$; set $b_i = 0$ otherwise;

`TEST` returns 1 if and only if $b_i = 1$ for all $0 \le i \le k-1$; and 0 otherwise.

E A Remark on Boneh-Gentry-Hamburg Abstract IBE System

Cocks' scheme was subsequently revisited by Boneh, Gentry, and Hamburg [8]. The advantage of their scheme resides in the length of the ciphertexts. While the encryption of an ℓ-bit message requires $2\ell \cdot \log_2 N$ bits with Cocks' scheme, ciphertext size in Boneh-Gentry-Hamburg scheme is about $\ell + \log_2 N$ bits.

This section simplifies the abstract IBE system with short ciphertexts as presented in [8, Section 3].

E.1 Description

`SETUP` and `EXTRACT` are similar to Cocks' scheme. `ENCRYPT` and `DECRYPT` require a deterministic algorithm $\mathcal{Q}$ taking as input an RSA modulus N and three elements $u, R, S \in \mathbb{Z}/N\mathbb{Z}$ and returning four *IBE-compatible polynomials* $f, g, \bar{f}, \tau \in \mathbb{Z}/N\mathbb{Z}[X]$. Polynomials $f, \bar{f}, g, \tau$ are said IBE-compatible if and only if the following conditions are met:

C1. If $R, S \in \mathbb{QR}_N$ then $f(r)g(s) \in \mathbb{QR}_N$ for all square roots r of R and s of S;
C2. If $R \in \mathbb{QR}_N$ then $f(r)f(-r)S \in \mathbb{QR}_N$ for all square roots r of R;
C3. If $uR, S \in \mathbb{QR}_N$ then $\bar{f}(\bar{r})g(s)\tau(s) \in \mathbb{QR}_N$ for all square roots $\bar{r}$ of uR and s of S;
C4. If $uR \in \mathbb{QR}_N$ then $\bar{f}(\bar{r})\bar{f}(-\bar{r})S \in \mathbb{QR}_N$ for all square roots $\bar{r}$ of uR;
C5. If $S \in \mathbb{QR}_N$ then $\tau(s)\tau(-s)u \in \mathbb{QR}_N$ for all square roots s of S;
C6. Polynomial τ is independent of R.

In more detail, the Boneh-Gentry-Hamburg scheme goes as follows.

`SETUP`(1^κ) Given a security parameter κ, `SETUP` generates an RSA modulus $N = pq$ where p and q are prime. It also generates a random element $u \in \mathbb{J}_N \backslash \mathbb{QR}_N$. The public system parameters are $\{N, u, \mathcal{H}, \mathcal{Q}\}$ where $\mathcal{H}$ is a cryptographic hash function mapping bitstrings to $\mathbb{J}_N$. The master secret key is $\mathsf{msk} = \{p, q\}$.

`EXTRACT`$_{\mathsf{msk}}(\mathsf{id})$ Using hash function $\mathcal{H}$, `EXTRACT` sets $R_{\mathsf{id}} = \mathcal{H}(\mathsf{id})$. If $R_{\mathsf{id}} \in \mathbb{QR}_N$ it computes $r_{\mathsf{id}} = {R_{\mathsf{id}}}^{1/2} \bmod N$; otherwise it computes $r_{\mathsf{id}} = (uR_{\mathsf{id}})^{1/2} \bmod N$. `EXTRACT` returns user's private key $\mathsf{usk} = \{r_{\mathsf{id}}\}$.

`ENCRYPT`(id, m) To encrypt a message $m \in \{\pm 1\}$ for user with identity id, `ENCRYPT`

- chooses at random $s \in \mathbb{Z}/N\mathbb{Z}$ and computes $S = s^2 \bmod N$;
- runs $\mathcal{Q}(N, u, R_{\mathsf{id}}, S)$ where $R_{\mathsf{id}} = \mathcal{H}(\mathsf{id})$ to obtain g and τ;
- computes $k = \left(\frac{\tau(s)}{N}\right)$;
- forms $w = m \cdot \left(\frac{g(s)}{N}\right)$;
- returns ciphertext $C = (S, k, w)$.

DECRYPT(usk, C) To decrypt $C = (S, k, w)$, intended for user with identity id, DECRYPT runs $\mathcal{Q}(N, u, R_{\mathsf{id}}, S)$ to obtain f and $\bar{f}$. Plaintext m is then recovered as

$$m = \begin{cases} w \cdot \left(\frac{f(r_{\mathsf{id}})}{N}\right) & \text{if } {r_{\mathsf{id}}}^2 \equiv R_{\mathsf{id}} \pmod N \\ w \cdot k \cdot \left(\frac{\bar{f}(r_{\mathsf{id}})}{N}\right) & \text{otherwise} \end{cases},$$

using the user's private key $\mathsf{usk} = \{r_{\mathsf{id}}\}$.

The correctness of the decryption follows from conditions C1–C6 imposed to polynomials $f, \bar{f}, g, \tau$.

The encryption of an ℓ-bit message $m = (m_1, m_2, \ldots, m_\ell) \in \{\pm 1\}^\ell$ proceeds broadly in the same way except that the user's private key is now $\mathsf{usk} = \{r_{\mathsf{id},1}, r_{\mathsf{id},2}, \ldots, r_{\mathsf{id},\ell}\}$ where ${r_{\mathsf{id},j}}^2 \equiv R_{\mathsf{id},j} \pmod N$ if $R_{\mathsf{id},j} \in \mathbb{QR}_N$ and ${r_{\mathsf{id},j}}^2 \equiv uR_{\mathsf{id},j} \pmod N$ if $R_{\mathsf{id},j} \in \mathbb{J}_N \setminus \mathbb{QR}_N$, and where $R_{\mathsf{id},j} = \mathcal{H}(\mathsf{id}, j)$ for $1 \leq j \leq \ell$. In other words, identity id is hashed ℓ times so as to produce ℓ values $R_{\mathsf{id},j}$ for $1 \leq j \leq \ell$. Now, each pair $(S, R_{\mathsf{id},j})$ (with the *same* S) is used to encrypt one message bit m_j; namely, $w_j = m_j \cdot \left(\frac{g_j(s)}{N}\right)$ where g_j is obtained by running $\mathcal{Q}(N, u, R_{\mathsf{id},j}, S)$. By the last condition, polynomial τ is always the same for all values of $R_{\mathsf{id},j}$. The ciphertext corresponding to message $m = (m_1, m_2, \ldots, m_\ell)$ is therefore given by $C = \{S, k, (w_1, w_2, \ldots, w_\ell)\}$. Plaintext message m is recovered from C bit-by-bit using private key $\mathsf{usk} = \{r_{\mathsf{id},1}, r_{\mathsf{id},2}, \ldots, r_{\mathsf{id},\ell}\}$ as $m_j = w_j \cdot \left(\frac{f(r_{\mathsf{id},j})}{N}\right)$ if ${r_{\mathsf{id},j}}^2 \equiv R_{\mathsf{id},j} \pmod N$ and as $m_j = w_j \cdot k \cdot \left(\frac{\bar{f}(r_{\mathsf{id},j})}{N}\right)$ otherwise, for $1 \leq j \leq \ell$.

E.2 A simplified abstract IBE

As described in the previous section, the abstract Boneh-Gentry-Hamburg system makes use of polynomials $f, \bar{f}, g, \tau \in \mathbb{Z}/N\mathbb{Z}[X]$. To simplify the notation, we consider one-bit messages but the discussion readily extends to ℓ-bit messages, $\ell > 1$.

We observe that polynomials f and $\bar{f}$ are evaluated at r_{id} and that polynomials g and τ are evaluated at s. Furthermore, we note that the values of R_{id} and of S are publicly known. So, letting δ denote the degree of polynomial f and $f(X) = \sum_{k=0}^{\delta} f_k X^k$ with $f_k \in \mathbb{Z}/N\mathbb{Z}$, we can write

$$f(r_{\mathsf{id}}) = \sum_{k=0}^{\delta} f_k {r_{\mathsf{id}}}^k = A\, r_{\mathsf{id}} + B \pmod N \tag{7}$$

where

$$A = \sum_{\substack{0 \leq k \leq \delta \\ k \text{ odd}}} f_k {r_{\mathsf{id}}}^{(k-1)} = \begin{cases} \displaystyle\sum_{\substack{0 \leq k \leq \delta \\ k \text{ odd}}} f_k {R_{\mathsf{id}}}^{(k-1)/2} \pmod N & \text{if } {r_{\mathsf{id}}}^2 \equiv R_{\mathsf{id}} \pmod N \\ \displaystyle\sum_{\substack{0 \leq k \leq \delta \\ k \text{ odd}}} f_k (uR_{\mathsf{id}})^{(k-1)/2} \pmod N & \text{otherwise} \end{cases}$$

and

$$B = \sum_{\substack{0\le k\le \delta \\ k \text{ even}}} f_k \, {r_{\mathsf{id}}}^k = \begin{cases} \displaystyle\sum_{\substack{0\le k\le \delta \\ k \text{ even}}} f_k \, {R_{\mathsf{id}}}^{k/2} \pmod N & \text{if } {r_{\mathsf{id}}}^2 \equiv R_{\mathsf{id}} \pmod N \\ \displaystyle\sum_{\substack{0\le k\le \delta \\ k \text{ even}}} f_k \, (uR_{\mathsf{id}})^{k/2} \pmod N & \text{otherwise} \end{cases} .$$

There is therefore no loss of generality to consider *degree-1* polynomials for f. The same conclusion holds for polynomials $\bar{f}$ (evaluated at r_{id}), and for polynomials g and τ (evaluated at s).

As a result, we define $f(X) = f_1\,X + f_0$, $\bar{f}(X) = \bar{f}_1\,X + \bar{f}_0$, $g(X) = g_1\,X + g_0$, and $\tau(X) = \tau_1\,X + \tau_0$. These four polynomials returned by public algorithm $\mathcal{Q}$ must be IBE-compatible.

For example, given R_{id} and S, one can select parameters $f_0, f_1, g_0, g_1 \in \mathbb{Z}/N\mathbb{Z}$ such that

$$2f_0g_0 \in \mathbb{QR}(N) \quad \text{and} \quad R_{\mathsf{id}} \left(\tfrac{f_1}{f_0}\right)^2 + S \left(\tfrac{g_1}{g_0}\right)^2 = 1 \pmod N . \tag{8}$$

This ensures that compatibility conditions C1 and C2 are satisfied.

Proof. Multiplying the second equation through ${f_0}^2$ yields $S{f_0}^2(\frac{g_1}{g_0})^2 = {f_0}^2 - R_{\mathsf{id}}{f_1}^2 = f(r_{\mathsf{id}})f(-r_{\mathsf{id}})$ for any square root r_{id} of R_{id}. Consequently, we have $f(r_{\mathsf{id}})f(-r_{\mathsf{id}})S = \left(Sf_0\,\frac{g_1}{g_0}\right)^2 \in \mathbb{QR}(N)$. We also have $\left(r_{\mathsf{id}}\,\frac{f_1}{f_0} + s\,\frac{g_1}{g_0} + 1\right)^2 = R_{\mathsf{id}}\left(\frac{f_1}{f_0}\right)^2 + S\left(\frac{g_1}{g_0}\right)^2 + 1 + 2r_{\mathsf{id}}s\,\frac{f_1}{f_0}\,\frac{g_1}{g_0} + 2r_{\mathsf{id}}\,\frac{f_1}{f_0} + 2s\,\frac{g_1}{g_0} = \frac{2}{f_0g_0}(f_0g_0 + r_{\mathsf{id}}sf_1g_1 + r_{\mathsf{id}}f_1g_0 + sg_1f_0) = \frac{2}{f_0g_0}\,f(r_{\mathsf{id}})g(s)$ for any square root r_{id} of R_{id} and any square root s of S. Since $2f_0g_0 \in \mathbb{QR}(N)$, it thus follows that $f(r_{\mathsf{id}})g(s) = \frac{2f_0g_0}{4}\left(r_{\mathsf{id}}\,\frac{f_1}{f_0} + s\,\frac{g_1}{g_0} + 1\right)^2 \in \mathbb{QR}(N)$, as required. □

We also define polynomial $\bar{g} \in \mathbb{Z}/N\mathbb{Z}[X]$ given by $\bar{g}(X) = \bar{g}_1\,X + \bar{g}_0$ where $\bar{g}_1 = g_1\tau_0 + g_0\tau_1 \bmod N$ and $\bar{g}_0 = g_1\tau_1 S + g_0\tau_0 \bmod N$. It is worth noticing that evaluated at s we have $\bar{g}(s) \equiv g(s)\tau(s) \pmod N$. Analogously, we require

$$2\bar{f}_0\bar{g}_0 \in \mathbb{QR}(N) \quad \text{and} \quad uR_{\mathsf{id}}\left(\tfrac{\bar{f}_1}{\bar{f}_0}\right)^2 + S\left(\tfrac{\bar{g}_1}{\bar{g}_0}\right)^2 = 1 \pmod N \tag{9}$$

so as to fulfill compatibility conditions C3 and C4. Compatibility conditions C5 and C6 are automatically satisfied from the product formula in [8, Lemma 5.1]. If (f_0, f_1, g_0, g_1) is a solution to Eq. (8) and if (α, β) is a solution to $u\alpha^2 + S\beta^2 = 1$ then $(\bar{f}_0, \bar{f}_1, \bar{g}_0, \bar{g}_1)$ is a solution to Eq. (9) provided that

$$\frac{\bar{f}_1}{\bar{f}_0} = \frac{\frac{f_1}{f_0}\alpha}{S\frac{g_1}{g_0}\beta + 1} \pmod N \quad \text{and} \quad \frac{\bar{g}_1}{\bar{g}_0} \stackrel{\text{def}}{\equiv} \frac{g_1\tau_0 + g_0\tau_1}{g_1\tau_1 S + g_0\tau_0} \equiv \frac{\frac{g_1}{g_0} + \beta}{S\frac{g_1}{g_0}\beta + 1} \pmod N \tag{10}$$

with $2\bar{f}_0\bar{g}_0 \in \mathbb{QR}(N)$.

The instantiation presented in [8, Section 4] corresponds to the choice $f_0 = 1$, $f_1 = x$, $g_0 = 2$, $g_1 = 2y$, $\bar{f}_0 = Sy\beta + 1$, $\bar{f}_1 = x\alpha$, $\tau_0 = 1$ and $\tau_1 = \beta$, for some (x, y) satisfying $R_{\mathsf{id}}x^2 + Sy^2 = 1 \pmod N$ and (α, β) satisfying $u\alpha^2 + S\beta^2 = 1 \pmod N$.