Identity-Based Encryption from e-th Power Residue Symbols

Xiaopeng Zhao[1], Jinwen Zheng[1], Zhenfu Cao *[1], Xiaolei Dong[1] and Nanyuan Cao[2]

[1]School of Computer Science and Software Engineering,
East China Normal University
[2]Department of Computer Science and Engineering,
Shanghai Jiao Tong University

Abstract

Boneh, LaVigne and Sabin (BLS)'s scheme naturally generalizes Cocks' scheme to higher power residue symbols, but it is less efficient, bandwidth-wise as computing the e-th power residue symbols is really time-consuming and ciphertexts are expressed in the form of polynomials. This paper, we improve the efficiency of BLS's scheme through taking off the part of computing e-th power residue symbols in its encryption phase. This modification makes the encryption much more efficient than that in Cocks' scheme. Our construction also widens BLS's scheme to the case e is square-free. Furthermore, we generalize the notable Galbraith's test by introducing the general reciprocity law on function fields. With the help of the extended Galbraith's test, we show BLS's scheme is not anonymous in general. We also provide some methods for computing the e-th power residue symbols.

Keywords: identity-based encryption; e-th power residue symbol; the general reciprocity law on function fields; anonymity.

1 Introduction

Identity-based encryption (IBE) is a public key encryption system, originally proposed by Shamir in 1984 [26], while first achieved in 2001 [5, 15]. The motivation of IBE is to solve some existing but unavoidable problems with classic public key encryption systems. For details, it substitutes the Public Key Infrastructure (PKI) for the Public Key Generator (PKG), and thus removes the overhead of certificate management and manages the keys centrally. As a result, IBE systems are more lightweight and scalable than classic ones. If an adversary learns nothing about the identity of a ciphertext, we call this IBE system anonymous. The notion of *anonymity* was put forward by Bellare *et al.* [3] as an additional security requirement of an encryption scheme.

Constructing cryptosystems from higher power residue symbols has been explored in several studies by researchers. Cao [9] proposed a type of extension of the Goldwasser and Micali's QR-based cryptosystem [18]. His scheme is based on k^{th}-power residues and enables segment encryption instead of bit encryption of Goldwasser and Micali's cryptosystem. In 2013, Joye and Libert [20] revisited the Goldwasser and Micali's QR-based cryptosystem using 2^k-th power residue symbols and described

*Corresponding author: zfcao@sei.ecnu.edu.cn

the most efficient lossy trapdoor function based on quadratic residuosity. Subsequently, Cao [10] proposed a type of extension of Joye and Libert's cryptosystem based on k^{th}-power residues. The extended scheme is more efficient than Joye and Libert's cryptosystem on decryption speed. Recently, Brier *et al.* [8] introduced new $p^r q$-based one-way functions and companion signature schemes based on higher-power residue symbols.

1.1 Related Work

Cocks' IBE scheme [15] is simple and a totally different approach. Encryption only needs several operations modulo an RSA modulus N and the evaluation of Jacobi symbols. Its security is based on the standard quadratic residuosity assumption. Cocks' scheme encrypts one bit plaintext into a ciphertext composed of a pair of two large integers, so it is not space efficient and used to encrypt short session keys in practice. Cocks' scheme is known not anonymous due to the Galbraith's test. In 2007, Boneh, Gentry and Hamburg [6] addressed the ciphertext expansion issue, they presented an anonymous IBE system which merely expands an ℓ-bit plaintext to a ciphertext about a size of $\ell + \log_2 N$. However, the encryption in their scheme is not efficient.
In 2013, Clear, Hughes, and Tewari [13] considered Cocks' scheme over the polynomial quotient ring $\mathbb{Z}_N[x]/(x^2 - R_{id})$ because it is natural and convenient to view ciphertexts as elements in it. With the help of this sharp observation, they constructed a strongly XOR-homomorphic IBE scheme. In the same year, Boneh, LaVigne and Sabin [7] generalized Cocks' scheme to e^{th} residuosity so that it can encrypt more than one bit in a message. The downside of this generalization is that the ciphertext expansion is massive, which is intractable to optimize yet because any intuitive attempt of compression fails to be secure due to the attack found by Boneh, LaVigne and Sabin [13]. Recently, Clear and McGoldrick [14] extended BLS's scheme so that it can use a hash function which can be securely instantiated.

1.2 Our Contributions

In this work, we investigate the IBE scheme of Boneh, LaVigne and Sabin (BLS's scheme) [7] and make the following contributions.
Our first contribution is to improve BLS's scheme in the following two aspects:

1. We omit the spare calculation of e-th power residue symbols, a very time-consuming part in the encryption phase of BLS's scheme. Also, this modification does not influence the security.

2. In BLS's scheme, e must be a prime number. We leverage knowledge of classical number theory and extend BLS's scheme to the case e is a square-free number, which strengthens the flexibility and is more efficient in the case that it is converted into a public-key scheme.

Moreover, we give a proof that our proposed IBE scheme is semantically secure.
Our first contribution is to make the discussion about the incompressibility of BLS's scheme in the original paper rigorous by introducing the general reciprocity law on function fields. Applying this technique, we successfully generalize the famous Galbraith's test and show that BLS's scheme is not anonymous when e is small.
Our third contribution is to provide methods for computing e-th power residue symbols for BLS's scheme and ours. Furthermore, we correct a theorem proposed in [17] and give an analogous conclusion which has the same effect.

2 Preliminaries

2.1 Notation

If X is a finite set, the notation $\#X$ means the cardinality of X, writing $x \leftarrow_{\$} X$ to indicate that x is an element sampled from the uniform distribution over X. If $\mathcal{A}$ is an algorithm, then we write $x \leftarrow \mathcal{A}(y)$ to mean: "run $\mathcal{A}$ on input y and the output is assigned to x". PPT is short for "probabilistic polynomial time".
For a group $\mathbb{G}$, the subgroup of $\mathbb{G}$ generated by the set X is denoted by $\langle X \rangle$. If R is a ring, $a, b \in R$ and I is an ideal of R, the relation $a - b \in I$ is written $a \equiv b \ (I)$. A finite field of size q is denoted by $\mathbb{F}_q$. For a polynomial f, we denote as $deg(f) = n$ to say f has degree n. lg stands for the binary logarithm. $\left(\frac{\cdot}{\cdot}\right)$ stands for Jacobi symbol. Let φ be the Euler's totient function.

2.2 Identity-Based Encryption

An identity-based encryption is defined as a tuple of four PPT algorithms $(\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$:
$\mathsf{Setup}\left(1^\lambda\right)$ The setup algorithm Setup is a randomized algorithm that takes a security parameter 1^λ as input, and outputs a tuple $(\mathsf{mpk}, \mathsf{msk})$, where the mpk denotes the public parameters and msk denotes the master secret key.
$\mathsf{KeyGen}(\mathsf{mpk}, \mathsf{msk}, id)$ The key generation algorithm KeyGen is a deterministic algorithm that takes msk and an identity id as inputs, and outputs a decryption key sk_{id} associated with the identity id.
$\mathsf{Enc}(\mathsf{mpk}, id, m)$ The encryption algorithm Enc is a randomized algorithm that takes mpk, id and a plaintext m as inputs, and outputs a ciphertext c. That is, we encrypt plaintext m with identity id and achieve ciphertext c.
$\mathsf{Dec}(\mathsf{mpk}, sk_{id}, c)$ The key generation algorithm Dec is a deterministic algorithm that takes mpk, sk_{id}, c as inputs, and outputs the corresponding plaintext m if c is a valid ciphertext, and $\perp$ otherwise.

2.3 Security Notions

2.3.1 Correctness

The correctness property issues the fact that any valid ciphertext can be decrypted to recover the corresponding plaintext. For a formal definition, we denotes $\mathsf{M}, \mathsf{ID}, \mathsf{C}$ as the plaintext space, the identity space and the ciphertext space respectively. An identity-based encryption $(\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ is said correct if $\forall m \in \mathsf{M}$, $\forall id \in \mathsf{ID}$, and $\mathsf{mpk}, id, sk_{id}$ obtained from Setup and KeyGen, it satisfies:

$$\Pr[\mathsf{Dec}(\mathsf{mpk}, sk_{id}, \mathsf{Enc}(\mathsf{mpk}, id, m)) = m] = 1.$$

2.3.2 Semantic Security

The semantic security property issues the fact that it is infeasible for any adversary with the limited computation ability to get any information of plaintext given the corresponding ciphertext. In another word, the behaviors of the adversary can be simulated by PPT algorithms.
A game played by the adversary and the challenger (the system or the designer) can describe the attack. The game has five phases: initialization phase, the first query phase, challenge phase, the

second query phase, guess phase. We describe each phase to capture the attack.
Initialization phase: The challenger runs the algorithm Setup, keeps the master secret key msk and gives the public parameters mpk to the adversary.
The first query phase: The adversary receives mpk. Thus it knows the plaintext space M, the identity space ID and the ciphertext space C. It then chooses a subset $\mathsf{ID}_1 \subseteq \mathsf{ID}$, issues key generation queries and receives back the the private keys corresponding to each identity in ID_1. The queries can be asked adaptively so the adversary can update and rich its knowledge of the scheme, which denotes by the state s.
Challenge phase: The adversary chooses a challenge identity $id^* \notin \mathsf{ID}_1$ and two different plaintexts m_0, m_1 of the same length . It sends them to the challenger.
The second query phase: This phase is the same to the first query phase excepts that the query identity subset ID_2 cannot contain id^*.
Guess phase: The challenger chooses a random bit b and encrypts m_b received from the adversary with msk, id^*. It then sends the corresponding ciphertext c to the adversary. The adversary tries to guess the bit b. It wins the game(carries a successful attack) when guessing right.
Formally, an identity-based encryption is said semantically secure if

$$\Pr\left[\begin{matrix}(\mathsf{mpk},\mathsf{msk}) \leftarrow\$ \mathsf{Setup}(1^\lambda) \\ (id^*,m_0,m_1,s)\leftarrow\mathcal{A}_1^{\mathsf{KeyGen}(\mathsf{mpk},\mathsf{msk},\cdot)} : \quad \mathcal{A}_2^{\mathsf{KeyGen}(\mathsf{mpk},\mathsf{msk},\cdot)}(s,c)=b \\ b \leftarrow\$ \{0,1\},\ c \leftarrow \mathsf{Enc}(\mathsf{mpk},id^*,m_b)\end{matrix}\right] - \frac{1}{2}$$

is negligible, where $\mathcal{A}_1$ denotes the behaviors of the adversary in two query phases and challenge phase, $\mathcal{A}_2$ denotes the behaviors of the adversary in guess phase.
Because the adversary can choose the challenge identity and plaintexts as it likes and tries to distinguish the challenge ciphertext, the semantic security can be also called indistinguishable chosen-identity chosen-plaintext security (IND-ID-CPA).

2.4 e-th Power Residue Symbol

Let K be a number field, and $\mathcal{O}_K$ be the ring of integers in K, and $e \geq 1$ be an integer. We say a prime ideal $\mathfrak{p}$ in $\mathcal{O}_K$ is prime to e if $\mathfrak{p} \nmid e\mathcal{O}_K$. It is easy to see that $\mathfrak{p}$ is relatively prime to e if and only if $\gcd(q, e) = 1$, where $q = p^f = \mathrm{Norm}(\mathfrak{p})$ for some $f \in \mathbb{N}$. For every $\alpha \in \mathcal{O}_K,\ \alpha \notin \mathfrak{p}$, we have

$$\alpha^{q-1} \equiv 1 \ \ (\mathfrak{p})$$

Let $\zeta_e = \exp^{2\pi i/e}$ be an e-th root of unity. If $\zeta_e \in K$ and $\mathfrak{p}$ is relatively prime to e, the order of the subgroup of $\#\left(\mathcal{O}_K/\mathfrak{p}\right)^\times$ generated by ζ_e mod $\mathfrak{p}$ is e. This indicates that e divides $q-1$, hence we can define the e-th power residue symbol $\left(\frac{\alpha}{\mathfrak{p}}\right)_e$ as follows:

1. $\left(\frac{\alpha}{\mathfrak{p}}\right)_e = 0$ if $\alpha \in \mathfrak{p}$.

2. If $\alpha \in \mathfrak{p}$, $\left(\frac{\alpha}{\mathfrak{p}}\right)_e$ is the unique e-th root of unity such that $\alpha^{\frac{\mathrm{Norm}(\mathfrak{p})-1}{e}} \equiv \left(\frac{\alpha}{\mathfrak{p}}\right)_e \ \ (\mathfrak{p})$.

Next, we extend the symbol multiplicatively to all ideals. Suppose $\mathfrak{a} \subset \mathcal{O}_K$ is an ideal prime to e. Let $\mathfrak{a} = \mathfrak{p}_1\mathfrak{p}_2\cdots\mathfrak{p}_m$ be the prime decomposition of $\mathfrak{a}$. For $\alpha \in \mathcal{O}_K$ define $\left(\frac{\alpha}{\mathfrak{a}}\right)_e = \prod_{i=1}^m \left(\frac{\alpha}{\mathfrak{p}_i}\right)_e$. If $\beta \in \mathcal{O}_K$ and β is prime to e, we define $\left(\frac{\alpha}{\beta}\right)_e = \left(\frac{\alpha}{(\beta)}\right)_e$. See [19, 22, 23] for more properties about

the e-th power residue symbol.
In the following, we only consider the case $K = \mathbb{Q}(\zeta_e)$. It's well-known that $\mathcal{O}_K = \mathbb{Z}[\zeta_e]$. Let $N = pq$ be a product of two primes satisfying $p \equiv 1 \pmod{n}$, $q \equiv 1 \pmod{n}$, then both p and q split completely in K. If $\mu \in \mathbb{Z}_N^*$ is a primitive e-th root of unity modulo p and modulo q, we say it a *non-degenerate* primitive e-th root of unity modulo N.

Lemma 2.1 (**Freeman** *et al.* [17])**.** *Let e be a positive integer, $N = pq$ be a product of two primes p, q with $p \equiv q \equiv 1$ mod e. Let $\mu \in \mathbb{Z}_N^*$ be a* non-degenerate *primitive e-th root of unity modulo N. For each i in $\{1, 2, \ldots, e\}$ with $\gcd(i, e) = 1$, let $\mathfrak{a}_i = N\mathcal{O}_K + (\zeta_e - \mu^i)\mathcal{O}_K$, $\mathfrak{p}_i = p\mathcal{O}_K + (\zeta_e - \mu^i)\mathcal{O}_K$ and $\mathfrak{q}_i = q\mathcal{O}_K + (\zeta_e - \mu^i)\mathcal{O}_K$. Then, we have $\mathrm{Norm}(\mathfrak{a}_i) = N$, $\mathfrak{a}_i = \mathfrak{p}_i\mathfrak{q}_i$ for all i, and*

$$p\mathcal{O}_K = \prod_{\gcd(i,e)=1} \mathfrak{p}_i \qquad q\mathcal{O}_K = \prod_{\gcd(i,e)=1} \mathfrak{q}_i \qquad N\mathcal{O}_K = \prod_{\gcd(i,e)=1} \mathfrak{a}_i$$

We define a function $\mathcal{J}_{N,e} : \mathbb{Z}_N \mapsto \{0, \ldots, e-1\}$ as follows:

$$\mathcal{J}_{N,e}(x) = \begin{cases} 0, & \text{if } \gcd(x, N) \neq 1, \\ i, & \text{if } \gcd(x, N) = 1 \text{ and } \left(\frac{x}{\mathfrak{a}_1}\right)_e = \zeta_e^i. \end{cases}$$

Obviously, if $x, y \in \mathbb{Z}_N^*$, then $\mathcal{J}_{N,e}(xy) = \mathcal{J}_{N,e}(x)\,\mathcal{J}_{N,e}(y)$. Squirrel [29] gave a polynomial algorithm computing e-th power residue symbols, which requires expensive precomputations. Boer [16] proposed an improved algorithm that does not rely on heavy precomputations and runs fast in experiments. However, he could not give a rigorous proof that it runs in polynomial time.
Assuming $e \geq 2$, we say an integer $x \in \mathbb{Z}_N^*$ is an e-th residue modulo N if there exists an integer $y \in \mathbb{Z}_N^*$ such that $y^e \equiv x \pmod{N}$. Note that if x is an e-th residue, then $\left(\frac{x}{\mathfrak{p}_i}\right)_e = \left(\frac{x}{\mathfrak{q}_i}\right)_e = 1$ holds for every i relatively prime to e. We denote the set of all e-th residues in $\mathbb{Z}_N^*$ by $\mathbb{ER}_{N,e}$. $\mathbb{PR}_{N,e}^i$ is defined by

$$\mathbb{PR}_{N,e}^i = \begin{cases} \left\{ x \in \mathbb{Z}_N^* | \left(\frac{x}{\mathfrak{a}_1}\right)_e = 1 \right\} & i = 0, \\ \left\{ x \in \mathbb{Z}_N^* | \left(\frac{x}{\mathfrak{a}_1}\right)_e = 1,\ \left(\frac{x}{\mathfrak{p}_1}\right)_e \text{ and } \left(\frac{x}{\mathfrak{q}_1}\right)_e \text{ are primitive} \right\} \cup \mathbb{ER}_{N,e} & i = 1. \end{cases}$$

We alter the MER assumption defined in [7] as follows.

Definition 2.2 (**Modified e-th Residue (MER_i, $i \in \{0, 1\}$) Assumption**)**.** For a PPT algorithm $\mathsf{RSAgen}(\lambda)$ that generates two equally sized primes p, q and a square-free integer e such that $p \equiv q \equiv 1$ mod e, $\gcd(\frac{p+q-2}{e}, e) = 1$, and picks $\mu \in \mathbb{Z}_N^*$ a *non-degenerate* primitive e-th root of unity to $N = pq$. We define the following two distributions relative to $\mathsf{RSAgen}(\lambda)$ as:

$$\mathbb{D}_{ER}^i : \left\{ (N, v, e, \mu) : (p, q, e, \mu) \leftarrow \mathsf{RSAgen}(\lambda),\ v \leftarrow\!\!\$\ \mathbb{PR}_{N,e}^i \right\}$$
$$\mathbb{D}_{ENR}^i : \left\{ (N, v, e, \mu) : (p, q, e, \mu) \leftarrow \mathsf{RSAgen}(\lambda),\ v \leftarrow\!\!\$\ \mathbb{PR}_{N,e}^i \setminus \mathbb{ER}_{N,e} \right\}$$

The MER_i assumption relative to $\mathsf{RSAgen}(\lambda)$ asserts that the advantage $\mathsf{Adv}_{\mathcal{A},\mathsf{RSAgen}}^{MER_i}(\lambda)$ defined as

$$\left| \Pr\left[\mathcal{A}(N, v, e, \mu) = 1 \mid (N, v, e, \mu) \leftarrow\!\!\$\ \mathbb{D}_{ER}(\lambda)\right] - \Pr\left[\mathcal{A}(N, v, e, \mu) = 1 \mid (N, v, e, \mu) \leftarrow\!\!\$\ \mathbb{D}_{ENR}^i(\lambda)\right] \right|$$

is negligible for any PPT adversary $\mathcal{A}$.

Both of two assumptions are natural extensions of the standard quadratic residue assumption ($\zeta_2 = -1$ and $\mu = -1$ when $e = 2$). Therefore, we believe that it is intractable to break both of them. Obviously, MER_1 implies MER_0. The following lemma further elaborates the relation between the two assumptions.

Lemma 2.3. *If there exists a* PPT *distinguisher* $\mathcal{A}$ *has* ϵ *advantage against* MER_1 *assumption where* $\epsilon > \frac{e}{\varphi(e)+1} - 1$, *then there exists a* PPT *distinguisher* $\mathcal{B}$ *has at least* $\frac{\varphi(e)+1}{e}(1+\epsilon) - 1$ *advantage against* MER_0 *assumption with comparable running time.*

Proof. Let $\mathbb{U} = \left\{1, \zeta_e, \ldots, \zeta_e^{e-1}\right\}$ denotes the subgroup of roots of unity in $\mathcal{O}_K$. The map $\theta : \mathbb{Z}_p^* \to \mathbb{U}$ given by $x \mapsto \left(\frac{x}{\mathfrak{p}_1}\right)_e$ is an homomorphism. Let $\mathbb{ER}_{p,e} = \left\{y \in \mathbb{Z}_p^* \mid y \equiv x^e \pmod p \text{ for some } x \in \mathbb{Z}_p^*\right\}$ be the subgroup composed of e-th residues in $\mathbb{Z}_p^*$. It's easy to verify that the cardinality of $\mathbb{ER}_{p,e}$ is $\frac{p-1}{e}$. Therefore, an integer $z \in \mathbb{Z}_p^*$ satisfying $\left(\frac{z}{\mathfrak{p}_1}\right)_e = 1$ must be in $\mathbb{ER}_{p,e}$. Hence the kernel of θ is $\mathbb{ER}_{p,e}$ and we have the following isomorphic

$$\mathbb{Z}_p^* \;/\; \mathbb{ER}_{p,e} \;\cong\; \mathbb{U}$$

due to the equality of cardinality. Of course, elements in different cosets of $\mathbb{ER}_{p,e}$ in $\mathbb{Z}_p^*$ have different e-th power residue symbols, whence there is a one to one correspondence between cosets of $\mathbb{ER}_{p,e}$ in $\mathbb{Z}_p^*$ and e-th roots of unity via the e-th power residue symbol. The above arguments are also valid in $\mathbb{Z}_q^*$. Hence, we have $\frac{\#\mathbb{PR}_{N,e}^1}{\#\mathbb{PR}_{N,e}^0} = \frac{(\varphi(e)+1)\frac{\varphi(N)}{e^2}}{e\frac{\varphi(N)}{e^2}} = \frac{\varphi(e)+1}{e}$,

When $\mathcal{B}$ is given the tuple $\{N, v, e, \mu\}$ as input, it invokes $\mathcal{A}$ and acts as follow: $\mathcal{B}(N, v, e, \mu) = 1$ if and only if $\mathcal{A}(N, v, e, \mu) = 1$. We have

$$\begin{aligned}
\mathsf{Adv}_{\mathcal{B},\mathsf{RSAgen}}^{MER_0} = &\left|\Pr\left[{}^{\mathcal{A}(N,v,e,\mu)=1:}_{(N,v,e,\mu)\,\leftarrow\$\,\mathbb{D}_{ER}^1(\lambda)}\right] \times \frac{\#\mathbb{PR}_{N,e}^1}{\#\mathbb{PR}_{N,e}^0} + \Pr\left[{}^{\mathcal{A}(N,v,e,\mu)=1:}_{(N,v,e,\mu)\,\leftarrow\$\,\mathbb{D}_{ER}^0(\lambda)\,\setminus\,\mathbb{D}_{ER}^1(\lambda)}\right] \times \left(1 - \frac{\#\mathbb{PR}_{N,e}^1}{\#\mathbb{PR}_{N,e}^0}\right)\right.\\
&- \Pr\left[{}^{\mathcal{A}(N,v,e,\mu)=1:}_{(N,v,e,\mu)\,\leftarrow\$\,\mathbb{D}_{ENR}^1(\lambda)}\right] \times \frac{\#\mathbb{PR}_{N,e}^1 - \#\mathbb{ER}_{N,e}}{\#\mathbb{PR}_{N,e}^0 - \#\mathbb{ER}_{N,e}} - \left.\Pr\left[{}^{\mathcal{A}(N,v,e,\mu)=1:}_{(N,v,e,\mu)\,\leftarrow\$\,\mathbb{D}_{ENR}^0(\lambda)\,\setminus\,\mathbb{D}_{ENR}^1(\lambda)}\right] \times \left(1 - \frac{\#\mathbb{PR}_{N,e}^1 - \#\mathbb{ER}_{N,e}}{\#\mathbb{PR}_{N,e}^0 - \#\mathbb{ER}_{N,e}}\right)\right|\\
\geq &\frac{\#\mathbb{PR}_{N,e}^1}{\#\mathbb{PR}_{N,e}^0} \times (1+\epsilon) - 1 = \frac{\varphi(e)+1}{e}(1+\epsilon) - 1
\end{aligned}$$

When the inequality $\epsilon > \frac{e}{\varphi(e)+1} - 1$ holds, $\mathcal{B}$ has at least $\frac{\varphi(e)+1}{e}(1+\epsilon) - 1$ advantage against MER_0 assumption, as desired. □

We close this section by showing that the precondition of Proposition 4.3 proposed in [17] can be relaxed as follows.

Proposition 2.4. *Let* e *be an integer. Let* $N = pq$ *where* $p \equiv q \equiv 1 \pmod e$. *Suppose that* $\gcd(\frac{p-1}{e}, e) = \gcd(\frac{q-1}{e}, e)$. *Then there is a* ν *such that*

1. *ν is a* non-degenerate *primitive e-th root of unity modulo N.*
2. *$\left(\frac{\nu}{\mathfrak{a}_i}\right)_e = 1$ for every ideal $\mathfrak{a}_i \subset \mathcal{O}_K$ as in Lemma 2.1.*

Proof. The condition $\gcd(\frac{p-1}{e}, e) = \gcd(\frac{q-1}{e}, e)$ implies that there exist integers s_p, t_p, s_q, t_q such that $s_p\frac{p-1}{e} + t_p e = s_q\frac{p-1}{e} + t_q e$. Let $\mu_p = \mu \bmod p$ and $\mu_q = \mu \bmod q$. We observe that every primitive e-th root of unity in $\mathbb{Z}_p$ has the form μ_p^i with $0 \le i < e$ and $\gcd(i, e) = 1$. It follows that

$$\left(\frac{\mu_p^{s_p}}{\mathfrak{p}_1}\right)_e = \left(\frac{\zeta_e^{s_p}}{\mathfrak{p}_1}\right)_e = \zeta_e^{\frac{p-1}{e}s_p}$$

Similarly,

$$\left(\frac{\mu_q^{-s_q}}{\mathfrak{q}_1}\right)_e = \left(\frac{\zeta_e^{-s_q}}{\mathfrak{q}_1}\right)_e = \zeta_e^{-\frac{q-1}{e}s_q}$$

Hence, letting ν be the integer that is congruent to $\mu_p^{s_p}$ modulo p and $\mu_q^{-s_q}$ modulo q. Then,

$$\left(\frac{\nu}{\mathfrak{a}_1}\right)_e = \left(\frac{\nu}{\mathfrak{p}_1}\right)_e \left(\frac{\nu}{\mathfrak{q}_1}\right)_e = \zeta_e^{s_p\frac{p-1}{e} - s_q\frac{q-1}{e}} = 1$$

Since $\nu \in \mathbb{Z}$, the result $\left(\frac{\nu}{\mathfrak{a}_i}\right)_e = 1$ follows from Galois-equivalence of the e-th power residue symbol. $\square$

3 Identity-Based Encryption from e-th Power Residue Symbols

3.1 Review of Boneh-LaVigne-Sabin(BLS)'s Scheme

We now describe the IBE scheme presented by Boneh, LaVigne and Sabin [7]. The scheme encrypts multiple bits at a time.

-$\mathsf{Setup}(1^\lambda)$ Given a security parameter λ, Setup selects a prime e, then generates an RSA modulus $N = pq$ a product of two large primes p and q such that $e \mid p-1, e \mid q-1$. The public parameters are $\mathsf{mpk} = \{N, e, \mu, \mathsf{H}\}$ where μ is a *non-degenerate* primitive e-th root of unity in $\mathbb{Z}_N$, H is a publicly available hash maps an arbitrary binary string to an e-th residue in $\mathbb{Z}_N^*$. The master secret key is $\mathsf{msk} = \{p, q\}$.

-$\mathsf{KeyGen}(\mathsf{mpk}, \mathsf{msk}, id)$ Using the hash function H and p, q, KeyGen sets $R_{id} = \mathsf{H}(id)$, then calculates $r_{id} = \mathsf{H}(id)^{\frac{1}{e}} \bmod N$. Finally, KeyGen returns $\mathsf{usk} = \{r_{id}\}$ as user's private key.

-$\mathsf{Enc}(\mathsf{mpk}, id, m)$ To encrypt a message $m \in \{0, \dots, e-1\}$ for a user with identity id, Enc derives the hash value $R_{id} = \mathsf{H}(id)$. It then chooses a random polynomial f of degree $e-1$ from $\mathbb{Z}_N[x]$ and calculates $g(x) = f(x)^e \bmod (x^e - R_{id}) = \sum_{i=0}^{e-1} a_i x^i$. Next, it chooses a transport key $t \leftarrow_\$ \mathbb{Z}_N^*$. The returned ciphertext is

$$C = \left\{\frac{a_0}{t}, \frac{a_1}{t}, \dots, \frac{a_{e-1}}{t}, (m + \mathcal{J}_{N,e}(t)) \bmod e\right\}.$$

-$\mathsf{Dec}(\mathsf{mpk}, \mathsf{usk}, C)$ When a user with $\mathsf{usk} = \{r_{id}\}$ receives a ciphertext set C, it parses C as $\{c_0, c_1, \dots, c_{e-1}, c\}$, Dec recovers the plaintext m as

$$m = \left(\mathcal{J}_{N,e}\left(\sum_{i=0}^{e-1} c_i r_{id}^i\right) + c\right) \pmod{e}$$

Remark 3.1. BLS's scheme extends Cocks' scheme to higher residue case. To see this, pick $e = 2$ and $f(x) = t + x$, there is $\frac{g(x)}{t} = t + \frac{R_{id}}{t} + 2x$, which behaves like that in Cocks' scheme.

Remark 3.2. In Cocks' scheme, the PKG can easily derive a user's secret key by several efficient probabilistic algorithms taking square roots in finite fields such as Cipolla-Lehmer [21], Tonelli-Shanks [27] and Adleman-Manders-Miller [1]. These methods can also be applied to general situations, e.g. [11], the extended Adleman-Manders-Miller's algorithm can extract an e-th root modulo a prime p in $\mathcal{O}\left(\log^4 p + e\log^3 p\right)$ time complexity.

Since it's unclear how to implement such a hash function H, BLS's scheme was not given a formally security proof in the original paper. In [14], the authors proposed an approach which is akin to Cocks' scheme to circumventing the issue, but at the cost of a lower efficiency and a larger ciphertext extension.

3.2 Our IBE Scheme

We find that it is redundant to compute e-th power residue symbols in the encryption phase of BLS's scheme. We presume that there exists a hash function mapping an arbitrary binary string to an e-th residue in $\mathbb{Z}_N^*$, e.g., The PKG selects a large number of e-th residues beforehand, and constructs it by the Lagrange's interpolation formula. Our improved IBE scheme for a square-free integer e is defined as follows:

-$\mathsf{Setup}\left(1^\lambda\right)$ Given a security parameter λ, Setup generates an RSA modulus $N = pq$ a product of two large primes p and q, and selects a square-free integer e with the prime decomposition $e = \prod_{i=1}^{\ell} e_i$ such that $e \mid p-1,\ e \mid q-1$ and $\gcd(\frac{p+q-2}{e}, e) = 1$. The settings of μ and H are the same as for in BLS's scheme. The public parameters are $\mathsf{mpk} = \{N, e, \mu, \mathcal{J}_{N,e}(\mu), \mathsf{H}\}$. The master secret key is $\mathsf{msk} = \{p, q\}$.

-$\mathsf{KeyGen}(\mathsf{mpk}, \mathsf{msk}, id)$ Using the hash function H and p, q, KeyGen sets $R_{id} = \mathsf{H}\left(id\right)$, then calculates $r_{id} = \mathsf{H}\left(id\right)^{\frac{1}{e}} \bmod N$. Finally, KeyGen returns $\mathsf{usk} = \{r_{id}\}$ as user's private key.

-$\mathsf{Enc}\left(\mathsf{mpk}, id, m\right)$ To encrypt a message $m \in \{0, \ldots, e-1\}$ for a user with identity id, Enc first derives the hash value $R_{id} = \mathsf{H}\left(id\right)$. Then, it generates a transport key $t = \mu^k$ where $k \leftarrow_{\$} \{0, \ldots, e-1\}$. We define the following sub-algorithm $\mathcal{E}$ which takes a prime number $\mathcal{P}$ and a public key R_{id} as inputs:

Algorithm 1: $\mathcal{E}$

Input: a prime number $\mathcal{P}$, a public key R_{id}
Output: a polynomial
1 Generate a uniform random polynomial $f(x) \leftarrow_{\$} \mathbb{Z}_N^*[x]$ of degree $\mathcal{P}-1$
2 Compute $g(x) \leftarrow f(x)^{\mathcal{P}} \bmod x^{\mathcal{P}} - R_{id}$
3 Output the polynomial $c(x) = \frac{g(x)}{\mu^{k \bmod \mathcal{P}}}$

The returned ciphertext is

$$C = \left\{\mathcal{E}\left(e_1\right), \ldots, \mathcal{E}\left(e_\ell\right), \left(m + \mathcal{J}_{N,e}\left(t\right)\right) \bmod e\right\}.$$

-$\mathsf{Dec}(\mathsf{mpk}, \mathsf{usk}, C)$ When a user with $\mathsf{usk} = \{r_{id}\}$ receives a ciphertext set C, it parses C as $\{c_1(x), \ldots, c_\ell(x), c\}$, Dec recovers the plaintext m as

$$m = \left(\mathcal{J}_{N,e}\left(\prod_{i=1}^{\ell} c_i(r_{id}^{\frac{e}{e_i}})^{\frac{e}{e_i}} \bmod N\right) + c\right) \bmod e$$

Remark 3.3. The condition $\gcd(\frac{p+q-2}{e}, e) = 1$ ensures that $\mathcal{J}_{N,e}(\mu)$ is primitive by the proof of Proposition 2.4. In the encryption phase, computing $\mathcal{J}_{N,e}(t) = (k\mathcal{J}_{N,e}(\mu) \bmod e)$ can be very convenient.

Remark 3.4. It is worth mentioning that in the case $e = 2$, the encryption is much more efficient than Cocks' scheme's, as a result of eliminating the operations about computing the inverses modulo $\mathbb{Z}_N^*$ and computing the Jacobi symbols.

Remark 3.5. Unfortunately, we could not omit to compute e-th power residue symbols in the decryption phase. One method is to utilize existing algorithms to compute power residue symbols with respect to each prime factor of e and to apply the Chinese remainder theorem (see Appendix B). The other is to keep the PKG online for answering queries about evaluating $\left(\frac{\cdot}{\mathfrak{a}_1}\right)_e$ from each user. In this case, a secure channel between the PKG and each user should be established.

Correctness Correctness can be verified directly as follows.

$$\begin{aligned}
\mathsf{Dec}(\mathsf{mpk}, sk_{id}, (\mathsf{Enc}(id, m))) &\equiv \sum_{i=1}^{\ell} \mathcal{J}_{N,e}\left(\left(\frac{c_i(r_{id}^{\frac{e}{e_i}})}{\mu^{k \bmod e_i}}\right)^{\frac{e}{e_i}}\right) + m + \mathcal{J}_{N,e}(\mu^k) \\
&\equiv \sum_{i=1}^{\ell} \mathcal{J}_{N,e}\left(\frac{1}{\mu^{(k \bmod e_i)\frac{e}{e_i}}}\right) + m + \mathcal{J}_{N,e}(\mu^k) \\
&\equiv \mathcal{J}_{N,e}\left(\frac{\mu^k}{\mu^{\sum_{i=1}^{\ell}(k \bmod e_i)\frac{e}{e_i}}}\right) + m \equiv m \pmod{e}
\end{aligned}$$

Security To simplify the security proof, we first prove the following theorem.

Theorem 3.6. *Let e be a prime number, $t \in \mathbb{Z}_N^*$ a transport key, R an element of $\mathbb{Z}_N^*$ such that $\left(\frac{R}{\mathfrak{p}_1}\right)_e = \zeta_e^{i_R}$, $\left(\frac{R}{\mathfrak{q}_1}\right)_e = \zeta_e^{j_R}$ where i_R, j_R are relatively prime to e. If $c(x) = \frac{f(x)^e}{t}$ mod $(x^e - R)$ where $f(x) \leftarrow_{\$} \mathbb{Z}_N^*[x]$ is a polynomial of degree $e-1$, then*

$$\Omega_{\bar{t}} = \left\{ g(x) \in \mathbb{Z}_N^*[x] \,\middle|\, \deg g(x) = e-1,\ \frac{g(x)^e}{\bar{t}} \textit{ mod } (x^e - R) = c(x) \right\}$$

has the same cardinality for each transport key $\bar{t} \in \mathbb{Z}_N^*$.

Proof. Suppose $\left(\frac{t\bar{t}^{-1}}{\mathfrak{p}_1}\right)_e = \zeta_e^{i_t}$, $\left(\frac{t\bar{t}^{-1}}{\mathfrak{q}_1}\right)_e = \zeta_e^{j_t}$. Since

$$\left(\frac{R^{i_R^{-1}i_t}}{\mathfrak{p}_1}\right)_e = \left(\frac{t\bar{t}^{-1}}{\mathfrak{p}_1}\right)_e, \quad \left(\frac{R^{j_R^{-1}j_t}}{\mathfrak{q}_1}\right)_e = \left(\frac{t\bar{t}^{-1}}{\mathfrak{q}_1}\right)_e,$$

by Lemma 2.3, there exist $W_p \in \mathbb{Z}_p^*$ and $W_q \in \mathbb{Z}_q^*$ such that

$$W_p^e R^{i_R^{-1}i_t} \equiv t\bar{t}^{-1} \pmod{p}, \quad W_q^e R^{j_R^{-1}j_t} \equiv t\bar{t}^{-1} \pmod{q}.$$

Since $\mathbb{Z}_N[x]/(x^e - R) \cong \mathbb{Z}_p[x]/(x^e - R) \oplus \mathbb{Z}_q[x]/(x^e - R)$, the map $\phi : \Omega_t \to \Omega_{\bar{t}}$ given by $h(x) \mapsto g(x)$ where

$$\begin{aligned}
g(x) &\equiv W_p x^{i_R^{-1}i_t} h(x) \pmod{p} \\
g(x) &\equiv W_q x^{j_R^{-1}j_t} h(x) \pmod{q}
\end{aligned}$$

is well defined for a fixed $\bar{t}$. Likewise, the inverse map $\psi : \Omega_{\bar{t}} \to \Omega_t$ is given by $g(x) \mapsto h(x)$ where

$$h(x) \equiv W_p^{-1} \left(R^{-1} x^{e-1}\right)^{i_R^{-1} i_t} g(x) \pmod p$$
$$h(x) \equiv W_q^{-1} \left(R^{-1} x^{e-1}\right)^{j_R^{-1} j_t} g(x) \pmod q$$

It is straightforward to verify $\psi\phi = 1_{\Omega_t}$ and $\phi\psi = 1_{\Omega_{\bar{t}}}$ where 1_{Ω_t} and $1_{\Omega_{\bar{t}}}$ denote the identity maps on Ω_t and on $\Omega_{\bar{t}}$ respectively. The proof is completed. □

We are now in a position to investigate the security of our IBE scheme.

Theorem 3.7. *Let $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ be an adversary against IND-ID-CPA security of our IBE scheme, making at most q_{H} queries to the random oracle H and a single query to the* ***Challenge*** *phase. Then, there exists an adversary $\mathcal{B}$ against the MER_1 assumption such that*

$$\mathsf{Adv}_{\mathcal{A}}^{\textit{ind-cpa}}(\lambda) = q_{\mathsf{H}} \cdot \mathsf{Adv}_{\mathcal{B},RSAgen}^{MER_1}(\lambda)$$

Proof. We prove it by defining a sequence of three games. For simplicity, we omit the procedure of Enc in the **Challenge** phase.

$\mathsf{Game}_1^{\mathcal{A}}(\lambda)$**:** This game is the real attack against our IBE scheme.

$\mathsf{Game}_2^{\mathcal{A}}(\lambda)$**:** In this game, we guess the number of the challenge identity and abort the game if the guess is wrong.

$\mathsf{Game}_3^{\mathcal{A}}(\lambda)$**:** We change the simulation of the **H** phase so that it returns a random element in $\mathbb{PR}_{N,e}^1$ for the i^*-th query.

We claim:

Claim 1: $\mathsf{Adv}_{\mathcal{A}}^{\text{ind-cpa}}(\lambda) = \left| \Pr\left[\mathsf{Game}_1^{\mathcal{A}}(\lambda) = true\right] - \frac{1}{2} \right|$.

Claim 2: $\Pr\left[\mathsf{Game}_2^{\mathcal{A}}(\lambda) = true\right] = \frac{1}{2}(1 - \frac{1}{q_{\mathsf{H}}}) + \frac{1}{q_{\mathsf{H}}} \Pr\left[\mathsf{Game}_1^{\mathcal{A}}(\lambda) = true\right]$.

Claim 3: $\left| \Pr\left[\mathsf{Game}_3^{\mathcal{A}}(\lambda) = true\right] - \Pr\left[\mathsf{Game}_2^{\mathcal{A}}(\lambda) = true\right] \right| \leq \mathsf{Adv}_{\mathcal{B},\text{RSAgen}}^{MER_1}(\lambda)$.

Claim 4: $\Pr\left[\mathsf{Game}_3^{\mathcal{A}}(\lambda) = true\right] = \frac{1}{2}$.

Proof. *Claim 1* follows immediately by the definition of semantic security. *Claim 2* is derived from Bayes' theorem. *Claim 3* follows from Difference Lemma [28]. If the public key Y of the challenge identity id^* is chosen from $\mathbb{PR}_{N,e}^1$, then $\left(\frac{Y}{\mathfrak{p}_1}\right)_{e_j}$ and $\left(\frac{Y}{\mathfrak{q}_1}\right)_{e_j}$ are both primitive e_j-th root of unity for every $1 \leq j \leq \ell$. Since $\left(\frac{\mu}{\mathfrak{a}_1}\right)_e$ is primitive e-th root of unity, the set

$$\left\{ \left\{ \left(\frac{\mu^{k \bmod e_1}}{\mathfrak{a}_1}\right)_{e_1}, \ldots, \left(\frac{\mu^{k \bmod e_\ell}}{\mathfrak{a}_1}\right)_{e_\ell} \right\} \,\middle|\, 0 \leq k < e \right\}$$

takes over combinations of all e_j-th roots of unity for each $1 \leq j \leq \ell$. Hence, by Theorem 3.6, ciphertexts are statistically indistinguishable to an adversary, which completes the proof of *Claim 4*. □

Combining all claims above gives this theorem.

$\mathsf{Game}_1^{\mathcal{A}}(\lambda)$

phase Setup(λ)

$b \leftarrow_\$ \{0,1\}$

$\mathcal{S}_\mathsf{H} \leftarrow \varnothing$; $ctr \leftarrow 0$

$\mathsf{msk} \leftarrow \{p, q\}$

$\mathsf{mpk} \leftarrow \{N, e, \mu, \mathcal{J}_{N,e}(\mu), \ell, e_1, \ldots, e_\ell\}$

return mpk

phase KeyGen(id)

if $(ctr, id, Y, \cdot) \notin \mathcal{S}_\mathsf{H}$ $\mathbf{H}(id)$

read $(ctr, id, Y, \cdot) \in \mathcal{S}_\mathsf{H}$

$\mathsf{usk} \leftarrow Y^{\frac{1}{e}} \bmod N$

return usk

phase H(id)

if $(ctr, id, Y, \cdot) \in \mathcal{S}_\mathsf{H}$ **return** Y

$ctr \leftarrow ctr + 1$

$Y \leftarrow_\$ \mathbb{ER}_N$

$\mathcal{S}_\mathsf{H} \leftarrow \mathcal{S}_\mathsf{H} \cup \{(ctr, id, Y, \perp)\}$

return Y

phase Challenge(id^*, m_0, m_1)

$C \leftarrow \mathsf{Enc}(\mathsf{mpk}, id^*, m_b)$

return C

phase Guess(b')

return $b' = b$

$\mathsf{Game}_2^{\mathcal{A}}(\lambda)$ $\boxed{\mathsf{Game}_3^{\mathcal{A}}(\lambda)}$

phase Setup(λ)

$b \leftarrow_\$ \{0,1\}$

$i^* \leftarrow_\$ \{1, \ldots, q_\mathsf{H}\}$

$\mathcal{S}_\mathsf{H} \leftarrow \varnothing$; $ctr \leftarrow 0$

$\mathsf{msk} \leftarrow \{p, q\}$

$\mathsf{mpk} \leftarrow \{N, e, \mu, \mathcal{J}_{N,e}(\mu), \ell, e_1, \ldots, e_\ell\}$

return mpk

phase KeyGen(id)

if $(ctr, id, Y, y) \notin \mathcal{S}_\mathsf{H}$ $\mathbf{H}(id)$

read $(ctr, id, Y, y) \in \mathcal{S}_\mathsf{H}$

if $y = \perp$ **abort**

$\mathsf{usk} \leftarrow y$

return usk

phase H(id)

if $(ctr, id, Y, y) \in \mathcal{S}_\mathsf{H}$ **return** Y

$ctr \leftarrow ctr + 1$

if $ctr = i^*$

  $y \leftarrow_\$ \mathbb{Z}_N^*$; $Y = y^e \bmod N$ $\boxed{Y \leftarrow_\$ \mathbb{PR}_N^1}$

  $\mathcal{S}_\mathsf{H} \leftarrow \mathcal{S}_\mathsf{H} \cup \{(ctr, id, Y, \perp)\}$

else

  $y \leftarrow_\$ \mathbb{Z}_N^*$; $Y = y^e \bmod N$

  $\mathcal{S}_\mathsf{H} \leftarrow \mathcal{S}_\mathsf{H} \cup \{(ctr, id, Y, y)\}$

return Y

phase Challenge(id^*, m_0, m_1)

if $(i^*, id, Y, y) \in \mathcal{S}_\mathsf{H}$ and $id = id^*$

  $C \leftarrow \mathsf{Enc}(\mathsf{mpk}, id^*, m_b)$

else abort

return C

phase Guess(b')

return $b' = b$

$\square$

3.3 Anonymity

In this section, we will generalize the famous Galbraith's test ([4]) to e-th power residue situation in order to prove that neither BLS's scheme nor our IBE scheme is anonymous when e is small. Let $a = H(id), N, c$ be the public key of user id, the modulus and the ciphertext as in Cocks' scheme respectively. Galbraith constructed the following elegant test:

$$GT(a, c) = \left(\frac{c^2 - 4a}{N}\right)$$

to distinguish the identity of a ciphertext. The reason it can be successful is: if the ciphertext c is generated by the user id with public key a, then $c^2 - 4a$ must be a square, but not necessarily if the public key a is replaced by another one. In [2], Ateniese and Gasti proved that Galbraith's test is the best test against the anonymity of Cocks' scheme. Following this method, we find if $g(x) = f(x)^e \bmod (x^e - R_{id})$ is a ciphertext polynomial encrypted by the user id in BLS's scheme, it is uncertain whether $g(x)$ can be encrypted by another user id' if the modulus $x^e - R_{id}$ is replaced by $x^e - R_{id'}$. With the notation and the technique as in Appendix A, an adversary can obtain c_N and γ by continuously applying Theorem A.4. Therefore, we define the *e-th Galbraith's test* as

$$GT(R_{id}, C)_e = \left(\frac{c_N\gamma}{\mathfrak{a}_1}\right)_e = \left(\frac{\left(\frac{t^{-1}g(x)}{x^e - R_{id}}\right)_{e,\mathbb{F}_p}^{\frac{e}{p-1}}}{\mathfrak{p}_1}\right)_e \left(\frac{\left(\frac{t^{-1}g(x)}{x^e - R_{id}}\right)_{e,\mathbb{F}_q}^{\frac{e}{q-1}}}{\mathfrak{q}_1}\right)_e .$$

Now that the ciphertext C is generated by the user id, the equation $GT(R_{id}, C)_e = 1$ must hold with all but negligible probability. While for another user id', we believe the value $GT(R_{id'}, C)_e$ is statistically close to the uniform distribution on $\left\{\zeta_e^i \mid i \in \{0, 1, \ldots, e-1\}\right\}$. We also naturally conjecture that the *e-th Galbraith's test* is the most effective test against the anonymity of BLS's scheme.

Remark 3.8. When $e = 2$, let $c_0, c_1 \in \mathbb{Z}_N^*$ and $c(x) = c_1x + c_0$ be the ciphertext polynomial, then

$$x^2 - R_{id} \equiv (c_1^{-1}c_0)^2 - R_{id} \pmod{c_1x + c_0}.$$

By Theorem A.4, we have $c_N = c_1^2$ and $\gamma = (c_1^{-1}c_0)^2 - R_{id}$. Hence, the 2*-th Galbraith's test* is

$$GT(R_{id}, C)_2 = \left(\frac{c_N\gamma}{\mathfrak{a}_1}\right)_2 = \left(\frac{c_0^2 - c_1^2R_{id}}{N}\right),$$

as mentioned in [13].

Example 3.9. Assume that all parameters of BLS's scheme are set as follows:

Parameter	Value	Parameter	Value
N	4331	r_{id}	67
p	61	$R_{id'}$	467
q	71	$r_{id'}$	51
e	5	t	7
μ	1900	$f(x)$	$x^4 + 2x^3 + 3x^2 + 4x + 6$
R_{id}	822	$\frac{g(x)}{t}$	$3184x^4 + 3485x^3 + 1183x^2 + 3757x + 1193$

Here, the ciphertext polynomial $\frac{g(x)}{t}$ is generated by the user id. To distinguish the identity of $\frac{g(x)}{t}$ between id and id', an adversary performs the following calculations first.

$\underline{id}$

$x^5 - 822 \equiv \mathsf{3855x^3 + 649x^2 + 1331x + 1525} \pmod{\mathsf{3184x^4 + 3485x^3 + 1183x^2 + 3757x + 1193}}$.
$\mathsf{3184x^4 + 3485x^3 + 1183x^2 + 3757x + 1193 \equiv 29x^2 + 460x + 1742} \pmod{\mathsf{3855x^3 + 649x^2 + 1331x + 1525}}$.
$\mathsf{3855x^3 + 649x^2 + 1331x + 1525 \equiv 3938x + 951} \pmod{\mathsf{29x^2 + 460x + 1742}}$.
$\mathsf{29x^2 + 460x + 1742 \equiv 55} \pmod{\mathsf{3938x + 951}}$.

$\underline{id'}$

$x^5 - 467 \equiv \mathsf{3855x^3 + 649x^2 + 1331x + 1880} \pmod{\mathsf{3184x^4 + 3485x^3 + 1183x^2 + 3757x + 1193}}$.
$\mathsf{3184x^4 + 3485x^3 + 1183x^2 + 3757x + 1193 \equiv 29x^2 + 105x + 3020} \pmod{\mathsf{3855x^3 + 649x^2 + 1331x + 1880}}$.
$\mathsf{3855x^3 + 649x^2 + 1331x + 1880 \equiv 3512x + 99} \pmod{\mathsf{29x^2 + 105x + 3020}}$.
$\mathsf{29x^2 + 105x + 3020 \equiv 4315} \pmod{\mathsf{3512x + 99}}$.

Next, it derives

$$c_N = (3184 \times 3855 \times 29 \times 3938)^2 \bmod 4331 \qquad c'_N = (3184 \times 3855 \times 29 \times 3512)^2 \bmod 4331$$
$$\gamma = 55 \qquad \gamma' = 4315$$

and computes $\left(\frac{c_N\gamma}{\mathfrak{a}_1}\right)_5 = 1$ and $\left(\frac{c'_N\gamma'}{\mathfrak{a}_1}\right)_5 = \zeta_5^3 \neq 1$. Finally, it can determine that the identity of $\frac{g(x)}{t}$ is id. Actually, there is $\left(\frac{c_N\gamma}{\mathfrak{p}_1}\right)_5 = \left(\frac{c_N\gamma}{\mathfrak{q}_1}\right)_5 = 1$.

4 Computing $\left(\frac{\cdot}{\mathfrak{p}_1}\right)_e$ and $\left(\frac{\cdot}{\mathfrak{q}_1}\right)_e$

In this section, we consider the public-key scheme converted from our IBE scheme. Computing e-th residue symbols seems to be easier in the decryption phase, for the factorization $\mathfrak{a}_1 = \mathfrak{p}_1\mathfrak{q}_1$ is known. The following simple theorem demonstrates that computing $\left(\frac{y}{\mathfrak{p}_1}\right)_e$ for an integer y is somewhat related to solving the discrete logarithm problem in a certain cyclic group. Recall that the discrete logarithm problem (DLP) is defined as: given a finite cyclic group $\mathbb{G}$ of order n with a generator α and an element $\beta \in \mathbb{G}$, find the integer $x \in \mathbb{Z}_n$ such that $\alpha^x = \beta$.

Theorem 4.1. $\left(\frac{y}{\mathfrak{p}_1}\right)_e = \zeta_e^x$ *if and only if* $\mu^x = y^{\frac{p-1}{e}}$ *in* $\mathbb{F}_p^*$. *Therefore, the solution of DLP in the finite cyclic subgroup* $\langle\mu\rangle$ *of order* e *means the computation of* $\left(\frac{y}{\mathfrak{p}_1}\right)_e$.

Proof. $\Leftarrow$ If $\mu^x = y^{\frac{p-1}{e}}$, then $y^{\frac{p-1}{e}} - \zeta_e^x = \mu^x - \zeta_e^x \in \mathfrak{p}_1$. Thus $\left(\frac{y}{\mathfrak{p}_1}\right)_e = \zeta_e^x$.
$\Rightarrow$ If $\left(\frac{y}{\mathfrak{p}_1}\right)_e = \zeta_e^x$ for some $x \in \mathbb{Z}_e$, that is $y^{\frac{p-1}{e}} - \zeta_e^x \in \mathfrak{p}_1$. Since the order of $y^{\frac{p-1}{e}}$ divides e, $y^{\frac{p-1}{e}}$ can be expressed as μ^z with an integer $z \in \mathbb{Z}_e$, which implies $\mu^x - \mu^z \in \mathfrak{p}_1$. The fact that the order of μ is e forces $x = y$. □

Although DLP is considered to be intractable in general, it can be quickly solved in a few special cases, e.g., if the order of $\mathbb{G}$ is smooth, Pohlig-Hellman algorithm [24] is much more efficient. Note that computing $\left(\frac{y}{\mathfrak{p}_1}\right)_e$ can be very fast in the case e is small (we can generate a lookup table or use baby-step giant-step algorithm). When e is medium large and smooth, using Pohlig-Hellman algorithm is appropriate. Specifically, if $\prod_i^\ell e_i$ is the prime factorization of e, the running time of computing $\left(\frac{y}{\mathfrak{p}_1}\right)_e$ is $\mathcal{O}\left(\sum_i^\ell(\lg e + \sqrt{e_i})\right)$ group multiplications.

Since the encryption only involves one evaluation of e-th power residue symbols, computing modular exponentiations of polynomials becomes the most time-consuming part. Therefore, pre-computing modular exponentiations for different random polynomials and saving the results are necessary.

5 Some thoughts and Problems

All known algorithms for computing e-th power residue symbols are suitable for small e. In the previous section, we have given an algorithm by solving DLP in the finite cyclic subgroup of order e if the factorization $\mathfrak{a}_1 = \mathfrak{p}_1\mathfrak{q}_1$ is already known. Without the knowledge of the factorization $\mathfrak{a}_1 = \mathfrak{p}_1\mathfrak{q}_1$, there is likely no algorithm running in low degree polynomial time in e, so is the problem of computing e-th power residue symbols hard for large e? And, if yes, is the public-key scheme converted from the space-efficient variation of BLS's scheme (see Appendix A) secure for sufficiently large e?

Acknowledgments

We are grateful to Jun Shao for useful discussions.
This work is supported in part by the National Natural Science Foundation of China (Grant No. 61632012, 61672239, 61602180), in part by Natural Science Foundation of Shanghai (Grant No. 16ZR1409200) and in part by China Postdoctoral Science Foundation (Grant No. 2017M611502).

References

[1] Leonard Adleman, Kenneth Manders, and Gary Miller. On taking roots in finite fields. In *18th Annual Symposium on Foundations of Computer Science (SFCS 1977)*, pages 175–178. IEEE, 1977.

[2] Giuseppe Ateniese and Paolo Gasti. Universally anonymous ibe based on the quadratic residuosity assumption. In *Cryptographers' Track at the RSA Conference*, pages 32–47. Springer, 2009.

[3] Mihir Bellare, Alexandra Boldyreva, Anand Desai, and David Pointcheval. Key-privacy in public-key encryption. In *International Conference on the Theory and Application of Cryptology and Information Security*, pages 566–582. Springer, 2001.

[4] Dan Boneh, Giovanni Di Crescenzo, Rafail Ostrovsky, and Giuseppe Persiano. Public key encryption with keyword search. In *International conference on the theory and applications of cryptographic techniques*, pages 506–522. Springer, 2004.

[5] Dan Boneh and Matt Franklin. Identity-based encryption from the weil pairing. In *Annual international cryptology conference*, pages 213–229. Springer, 2001.

[6] Dan Boneh, Craig Gentry, and Michael Hamburg. Space-efficient identity based encryption-without pairings. In *48th Annual IEEE Symposium on Foundations of Computer Science (FOCS'07)*, pages 647–657. IEEE, 2007.

[7] Dan Boneh, Rio LaVigne, and Manuel Sabin. Identity-based encryption with e^{th} residuosity and its incompressibility. In *Autumn 2013 TRUST Conference. Washington DC (Oct 9-10, 2013), poster presentation*, 2013.

[8] Eric Brier, Houda Ferradi, Marc Joye, and David Naccache. New number-theoretic cryptographic primitives. Cryptology ePrint Archive, Report 2019/484, 2019. `https://eprint.iacr.org/2019/484`.

[9] Zhenfu Cao. A new public-key cryptosystem based on k^{th}-power residues (full version). *Journal of the China Institute of Communications*, 11(2):80–83, 1990. `https://wenku.baidu.com/view/3ec12f0d7cd184254b3535dc`.

[10] Zhenfu Cao, Xiaolei Dong, Licheng Wang, and Jun Shao. More efficient cryptosystems from k-th power residues. *IACR Cryptology ePrint Archive*, 2013:569, 2013.

[11] Zhengjun Cao, Qian Sha, and Xiao Fan. Adleman-manders-miller root extraction method revisited. In *International Conference on Information Security and Cryptology*, pages 77–85. Springer, 2011.

[12] Leonard Carlitz et al. On certain functions connected with polynomials in a galois field. *Duke Mathematical Journal*, 1(2):137–168, 1935.

[13] Michael Clear, Arthur Hughes, and Hitesh Tewari. Homomorphic encryption with access policies: Characterization and new constructions. In *International Conference on Cryptology in Africa*, pages 61–87. Springer, 2013.

[14] Michael Clear and Ciaran McGoldrick. Additively homomorphic ibe from higher residuosity. In *IACR International Workshop on Public Key Cryptography*, pages 496–515. Springer, 2019.

[15] Clifford Cocks. An identity based encryption scheme based on quadratic residues. In *IMA International Conference on Cryptography and Coding*, pages 360–363. Springer, 2001.

[16] Koen de Boer. *Computing the power residue symbol*. PhD thesis, Master's thesis. Nijmegen, Radboud University. `www.koendeboer.com`, 2016.

[17] David Mandell Freeman, Oded Goldreich, Eike Kiltz, Alon Rosen, and Gil Segev. More constructions of lossy and correlation-secure trapdoor functions. *Journal of cryptology*, 26(1):39–74, 2013.

[18] Shafi Goldwasser and Silvio Micali. Probabilistic encryption. *Journal of computer and system sciences*, 28(2):270–299, 1984.

[19] Kenneth Ireland and Michael Rosen. *A classical introduction to modern number theory*, volume 84. Springer Science & Business Media, 2013.

[20] Marc Joye and Benoit Libert. Efficient cryptosystems from 2^k-th power residue symbols. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 76–92. Springer, 2013.

[21] Derrick H Lehmer. Computer technology applied to the theory of numbers. *Studies in number theory*, pages 117–151, 1969.

[22] Franz Lemmermeyer. *Reciprocity laws: from Euler to Eisenstein.* Springer Science & Business Media, 2013.

[23] Jürgen Neukirch. *Algebraic number theory*, volume 322. Springer Science & Business Media, 2013.

[24] Stephen Pohlig and Martin Hellman. An improved algorithm for computing logarithms over $\mathbb{GF}(p)$ and its cryptographic significance. *IEEE Transactions on information Theory*, 24(1):106–110, 1978.

[25] Michael Rosen. *Number theory in function fields*, volume 210. Springer Science & Business Media, 2013.

[26] Adi Shamir. Identity-based cryptosystems and signature schemes. In *Workshop on the theory and application of cryptographic techniques*, pages 47–53. Springer, 1984.

[27] Daniel Shanks. Five number-theoretic algorithms. In *Proceedings of the Second Manitoba Conference on Numerical Mathematics (Winnipeg), 1973*, 1973.

[28] Victor Shoup. Sequences of games: a tool for taming complexity in security proofs. *IACR Cryptology ePrint Archive*, 2004:332, 2004.

[29] Douglas Squirrel. Computing reciprocity symbols in number fields, 1997. *Undergraduate thesis, Reed College.*

A Incompressibility of BLS's Scheme

BLS's scheme has a space-efficient variation (see [7]) that seems to work. If a trusted PKG sets the user's secret key to the root of $x^\delta - R_{id}$ for some δ satisfying $2 \leq \delta < e$, and operations of polynomials are performed in the quotient ring $\mathbb{Z}_N[x]/\left(x^\delta - v\right)$ in the encryption phase, then the number of elements in the ciphertext can be reduced to $\delta + 1$. However, this ambitious method makes the scheme insecure. Moreover, there even exists an attack that recovers the decrypted messages from the reciprocity law on $\mathbb{F}_q[t]$, the polynomial ring over a finite field $\mathbb{F}_q$. This attack also shows that it is incompressible for any generalization of similar methods. We start by explaining notation to be used and give crucial definitions and results due to Carlitz [12]. We here refer to Chapter 3 in [25].

Every element in $\mathbb{F}_q[t]$ has the form $f(t) = \alpha_n t^n + \alpha_{n-1} t^{n-1} + \cdots + \alpha_0$. In this case we set $sgn(f) = \alpha_n$ and call it the sign of f. Let $P \in \mathbb{F}_q[t]$ of degree γ be an irreducible polynomial and e a divisor of $q-1$. Note that there is a unique $\alpha \in \mathbb{F}_q^*$ such that $a^{\frac{q^\gamma - 1}{e}} \equiv \alpha \pmod{P}$.

Definition A.1. If $a \in \mathbb{F}_q[t]$ and P does not divide a, let $\left(\frac{a}{P}\right)_e$ be the unique element of $\mathbb{F}_q^*$ such that

$$a^{\frac{q^\gamma - 1}{e}} \equiv \left(\frac{a}{P}\right)_e \pmod{P}.$$

If $P \mid a$ define $\left(\frac{a}{P}\right)_e = 0$. The symbol $\left(\frac{a}{P}\right)_e$ is called the e-th power residue symbol.

Proposition A.2. *The e-th power residue symbol has the following properties:*

1. $\left(\frac{a}{P}\right)_e = \left(\frac{b}{P}\right)_e$ *if* $a \equiv b \pmod{P}$.

2. $\left(\frac{ab}{P}\right)_e = \left(\frac{a}{P}\right)_e \left(\frac{b}{P}\right)_e$.

3. *Let* $\alpha \in \mathbb{F}_q$. *Then,* $\left(\frac{\alpha}{P}\right)_e = \alpha^{\frac{q-1}{e}\gamma}$.

Just as the Jacobi symbol, the definition of the e-th power residue symbol can be extended to the case that P is an arbitrary non-zero element $b \in \mathbb{F}_q[t]$ with the prime decomposition $b = sgn(b)Q_1^{f_1} \cdots Q_s^{f_s}$, and define

$$\left(\frac{a}{b}\right)_e = \prod_{j=1}^{s} \left(\frac{a}{Q_j}\right)_e^{f_j}.$$

Proposition A.3. *The symbol* $\left(\frac{a}{b}\right)_e$ *has the following properties:*

1. *If* $a_1 \equiv a_2 \pmod b$, *then* $\left(\frac{a_1}{b}\right)_e = \left(\frac{a_2}{b}\right)_e$.

2. $\left(\frac{a_1 a_2}{b}\right)_e = \left(\frac{a_1}{b}\right)_e \left(\frac{a_2}{b}\right)_e$.

3. $\left(\frac{a}{b_1 b_2}\right)_e = \left(\frac{a}{b_1}\right)_e \left(\frac{a}{b_2}\right)_e$.

4. $\left(\frac{a}{b}\right)_e \neq 0$ *if and only if* a *is relatively prime to* b.

5. *If* $x^e \equiv a \pmod b$ *is solvable, then* $\left(\frac{a}{b}\right)_e = 1$.

The following pretty theorem is the general reciprocity law for $\mathbb{F}_q[t]$.

Theorem A.4. *[**The general reciprocity law**] Let* $a, b \in \mathbb{F}_q[t]$ *be relatively prime, non-zero elements. Then,*

$$\left(\frac{a}{b}\right)_e = \left(\frac{b}{a}\right)_e \left((-1)^{deg(a)deg(b)} sgn(a)^{deg(b)} sgn(b)^{-deg(a)}\right)^{\frac{q-1}{e}}$$

An adversary who intercepts ciphertexts has the ability of recreating the polynomial $\frac{g(x)}{t} = \frac{f(x)^e}{t} \bmod \left(x^\delta - R_{id}\right)$. Let $x^\delta - R_{id} = \prod_{j=1}^{m} \eta_j^{p_j}$ be the prime decomposition of $x^\delta - R_{id}$ in $\mathbb{F}_p[x]$. We obtain

$$\left(\frac{t^{-1}g(x)}{x^\delta - R_{id}}\right)_{e,\mathbb{F}_p} = \left(\frac{t^{-1}f(x)^e}{x^\delta - R_{id}}\right)_{e,\mathbb{F}_p} = \prod_{j=1}^{m} \left(\frac{t^{-1}}{\eta_j}\right)_{e,\mathbb{F}_p}^{p_j} = \prod_{j=1}^{m} t^{-\frac{p-1}{e} p_j deg(\eta_j)} \tag{1}$$

$$= t^{-\frac{p-1}{e}\delta} \equiv \left(\frac{t^{-1}}{\mathfrak{p}_1}\right)_e^{\delta} \quad (\mathfrak{p}_1). \tag{2}$$

Similarly,

$$\left(\frac{t^{-1}g(x)}{x^\delta - R_{id}}\right)_{e,\mathbb{F}_q} \equiv \left(\frac{t^{-1}}{\mathfrak{q}_1}\right)_e^{\delta} \quad (\mathfrak{q}_1). \tag{3}$$

Notice that all the following three situations occur with overwhelming probability.

1. $\gcd(x^\delta - R_{id}, f(x)) = 1$.

2. The term x^{e-1} of $\frac{g(x)}{t}$ has non-zero coefficient.

3. Each term of each polynomial involved has coefficient relatively prime to N.

Therefore, we think all of them hold by default. By continuously applying Theorem A.4, we can get

$$\left(\frac{t^{-1}g(x)}{x^\delta - R_{id}}\right)_{e,\mathbb{F}_p} \equiv \left(\frac{c_p}{\mathfrak{p}_1}\right)_e \left(\frac{\alpha}{\Phi(x)}\right)_{e,\mathbb{F}_p} \quad (\mathfrak{p}_1), \quad \left(\frac{t^{-1}g(x)}{x^\delta - R_{id}}\right)_{e,\mathbb{F}_q} \equiv \left(\frac{c_q}{\mathfrak{q}_1}\right)_e \left(\frac{\beta}{\Psi(x)}\right)_{e,\mathbb{F}_q} \quad (\mathfrak{q}_1) \tag{4}$$

where $\alpha, c_p \in \mathbb{F}_p$, $\beta, c_q \in \mathbb{F}_q$ and $\Phi(x) \in \mathbb{F}_p[x]$, $\Psi(x) \in \mathbb{F}_q[x]$, $deg(\Phi(x)) = deg(\Psi(x)) = 1$. An adversary can also do the above steps, but in $\mathbb{Z}_N[x]$. That is, it can get c_N, γ and $\Theta(x)$ such that

$$\begin{aligned} c_N &\equiv c_p \pmod p & \gamma &\equiv \alpha \pmod p & \Theta(x) &\equiv \Phi(x) \pmod p \\ c_N &\equiv c_q \pmod q & \gamma &\equiv \beta \pmod q & \Theta(x) &\equiv \Psi(x) \pmod q \end{aligned}$$

Combining (1),(3),(4) yields

$$\left(\frac{t^{-1}}{\mathfrak{p}_1}\right)_e^\delta = \left(\frac{c_p}{\mathfrak{p}_1}\right)_e \left(\frac{\alpha}{\mathfrak{p}_1}\right)_e, \quad \left(\frac{t^{-1}}{\mathfrak{q}_1}\right)_e^\delta = \left(\frac{c_q}{\mathfrak{q}_1}\right)_e \left(\frac{\beta}{\mathfrak{q}_1}\right)_e. \tag{5}$$

Since $\delta < e$ and e is a prime number, an adversary gains $\left(\frac{t^{-1}}{\mathfrak{a}_1}\right)_e$ by calculating $\left(\frac{c_N\gamma}{\mathfrak{a}_1}\right)_e^{\delta^{e-2} \bmod e}$.

Example A.5. Finally, we give a toy example to show how an adversary attacks the space-efficient variation of BLS's scheme. Assume that all parameters are set as follows:

Parameter	Value	Parameter	Value
N	4331	R_{id}	158
p	61	r_{id}	67
q	71	$f(x)$	$x^4 + 2x^3 + 3x^2 + 4x + 6$
e	5	t	7
μ	1900	$\frac{g(x)}{t}$	$2102x + 3769$
δ	2		

By calculation, we learn $\left(\frac{7}{\mathfrak{p}_1}\right)_5 = \zeta_5^4$, $\left(\frac{7}{\mathfrak{q}_1}\right)_5 = \zeta_5$. An adversary first analyzes as

$$x^2 - 158 \equiv 2102^{-2}3769^2 - 158 = 1416 \pmod{2102x + 3769},$$

then gets $c_N = \big((-1)^2 2102^2\big)$, $\gamma = 1416$, and finally derives $\left(\frac{c_N\gamma}{\mathfrak{a}_1}\right)_5^3 = 1 = \left(\frac{7^{-1}}{\mathfrak{a}_1}\right)_5$. Actually, there is $\left(\frac{c_N\gamma}{\mathfrak{p}_1}\right)_5 = \left(\frac{7}{\mathfrak{p}_1}\right)_5^3$, $\left(\frac{c_N\gamma}{\mathfrak{q}_1}\right)_5 = \left(\frac{7}{\mathfrak{q}_1}\right)_5^3$.

B Computing $(\frac{\cdot}{\mathfrak{a}_1})_e$ for Large Values of e

In [17], to compute the e-th power residue symbol, the authors constructed a "compatibility" identity and stated that it holds for all ideals in $\mathbb{Z}[\zeta_e]$. But this is not correct, e.g., If $\mathfrak{U}$ is a prime ideal in $\mathbb{Z}[\zeta_e]$ and $\mathfrak{B} = \mathfrak{U} \cap \mathbb{Z}[\zeta_f]$ is a prime ideal in $\mathbb{Z}[\zeta_f]$ where $f \mid e$, the argument $\mathrm{Norm}_{\mathbb{Z}[\zeta_e]}(\mathfrak{U}) = \mathrm{Norm}_{\mathbb{Z}[\zeta_f]}(\mathfrak{B})$ is not always true. In fact, when $\mathfrak{B}$ is singular, the local-global principle makes the "compatibility" identity hold, see Chapter 1 in [16]. Furthermore, note that in the case $\mathrm{Norm}_{\mathbb{Z}[\zeta_e]}(\mathfrak{U}) = p-1$, it also holds due to the inclusion map $\iota : \mathbb{Z}[\zeta_e]/\mathfrak{U} \mapsto \mathbb{Z}[\zeta_f]/\mathfrak{B}$. Hence, we formalize the following revised theorem.

Theorem B.1. *Let e, f be integers with $f \mid e$. Let $\mathfrak{p}_1$ be as Lemma 2.1, and let $x \in \mathbb{Z}[\zeta_e]$. Then*

$$\left(\frac{x}{\mathfrak{p}_1 \cap \mathbb{Z}[\zeta_f]}\right)_f = \left(\frac{x}{\mathfrak{p}_1}\right)_e^{\frac{e}{f}}.$$

One can verify that $\mathfrak{p}_1 \cap \mathbb{Z}[\zeta_f] = p\mathbb{Z}[\zeta_e] + (\zeta_f - \mu^{e/f})\mathbb{Z}[\zeta_e]$ due to the fact that $\mu^{e/f}$ is a *non-degenerate* primitive f-th root of unity modulo N. Therefore, we are able to learn the value of $\left(\frac{x}{\mathfrak{a}_1}\right)_e$ by computing $\left(\frac{x}{N\mathbb{Z}[\zeta_f]+(\zeta_f-\mu^{e/f})\mathbb{Z}[\zeta_f]}\right)_f$ for each prime factor f of e and applying the Chinese remainder theorem.