New Assumptions and Efficient Cryptosystems from Power Residue Symbols

Xiaopeng Zhao[1], Zhenfu Cao[1] **, Xiaolei Dong[1], Jun Shao[2], and Zhusen Liu[1]

[1] School of Computer Science and Software Engineering, East China Normal University, Shanghai, China
52164500025@stu.ecnu.edu.cn,52184501023@stu.ecnu.edu.cn
zfcao@sei.ecnu.edu.cn,dongxiaolei@sei.ecnu.edu.cn
[2] School of Computer and Information Engineering, Zhejiang Gongshang University, Hangzhou, China
jshao@zjgsu.edu.cn

Abstract. At Eurocrypt 2013, Joye and Libert proposed a method for generalizing the famous Goldwasser-Micali cryptosystem to the cases of 2^k-th power residues. Their work basically addresses the issue of the ciphertext expansion, which is quite large in the Goldwasser-Micali cryptosystem. In this paper, we provide some new efficient methods for computing a type of power residue symbols, which presents a generic tool for improving cryptographic schemes based on the theory of quadratic residues. To illustrate this point, we utilize the tool to generalize and improve the Joye-Libert cryptosystem in terms of both decryption speed and ciphertext bandwidth. We also generalize some well-known results on quadratic residues and use them to instantiate the subgroup indistinguishability assumption, which can be utilized to construct key-dependent security and auxiliary-input security schemes.

Keywords: power residue symbols · Goldwasser-Micali cryptosystem · Joye-Libert cryptosystem · leakage resilient public-key encryption.

1 Introduction

At STOC 1982 [18], Goldwasser and Micali constructed the first probabilistic and homomorphic encryption scheme which is provably secure under the standard *quadratic residue assumption*. It plays a major role in the development of modern cryptography by introducing the concept of "semantic security" and "indistinguishability". The cryptosystem is quite elegant, simple and efficient in term of both encryption and decryption. However, it is inefficient in bandwidth utilization, which is a major concern for real-world applications. Since then many attempts [4,14,3,9,2,10,25,27,28,22,11] have been made to address this issue.

One intuitive approach to improve the bandwidth utilization is to introduce the power residue symbols as the Goldwasser-Micali cryptosystem is presented

** Corresponding author

based on the theory of quadratic residues. From 1985 to 1990, Benaloh and Fischer [2,14] put forward a more efficient, bandwidth-wise, scheme using a k-bit prime r such that $r|p-1$, $r^2 \nmid p-1$ and $r \nmid q-1$. However, the decryption is demanding and k is limited to 40, so the ciphertext expansion is still large; Cao [9,10] proposed two types of extensions of the Goldwasser-Micali cryptosystem. One scheme with faster decryption is based on the cubic residue in the ring $\mathbb{Z}[\omega]$. The other is based on the k^{th}-power residues and enables the segment encryption instead of encrypting bit-by-bit. In 1998, Naccache and Stern [25] observed that the decryption of the Benaloh-Fischer scheme can be made even faster by considering a smooth and square-free integer $R = \prod_i r_i$ such that $r_i \mid \varphi(N)$ but $r_i^2 \nmid \varphi(N)$ for each prime r_i. In 2013, Joye and Libert [22] revisited and extended the Goldwasser-Micali cryptosystem using 2^k-th power residue symbols. Their proposed cryptosystems inherit the homomorphic property of the original cryptosystem and are efficient in both bandwidth and speed. Subsequently, Cao [11] demonstrated that the work of Joye and Libert can be extended more generally. However, their security analysis is quite complicated and is not easy to understand since it closely follows the analysis of the Joye-Libert cryptosystem.

Another different approach, which was proposed by Okamoto and Uchiyama [27], is to change the group structure. They suggested using $\mathbb{G} = \mathbb{Z}_N^*$ with moduli of the form $N = p^2q$ instead of $N = pq$. One year later, Paillier [28] proposed an extension of the Okamoto-Uchiyama cryptosystem. This extension is working in a different group $\mathbb{G} = \mathbb{Z}_{N^2}^*$ where N is an usual RSA modulus and reduces the ciphertext expansion by one-third. An interesting question is whether the two methods described above can be combined or unified.

Recently, the theory of power residues has attracted the attention of some cryptographic researchers. For example, Clear and McGoldrick [13] constructed an *identity-based encryption* scheme supporting homomorphic addition modulo a polynomial-sized prime e. Brier *et al.* [6] introduced new $p^r q$-based one-way functions and companion signature schemes by means of replacing the Jacobi symbol with the power residue symbol. We summarize the main reason for introducing power residues into cryptography as: the public-key schemes based on the theory of quadratic residues will cause a very large ciphertext expansion and a small message space. However, designing an efficient algorithm to compute the power residue symbols is not an easy job. In this paper, we basically address this issue and provide a generic tool for improving the existing schemes based on the quadratic residues.

Our Contributions

Computing a Type of Power Residue Symbols The theory of quadratic residues (QR) has been a fundamental tool in a wide range of cryptographic applications. However, in some applications such as public-key encryption and digital signature, the bandwidth is wasteful if the scheme is constructed based on the QR theory. In this paper, we basically address this issue by introducing the theory of power residues, which is an important branch of algebraic number theory. Though several works [12,32,16,20,21,7] have been devoted to computing power residue symbols, as far as we know, the existing algorithms can only efficiently and deterministically deal with small powers. Therefore, their algorithms are not suitable for constructing cryptographic applications. Interestingly, we find that computing a type of power residue symbols is closely related to solving the discrete logarithm problem in a specific cyclic group given the factorization of the modulus. Based on this discovery, we construct practical and efficient power residue tools that can be used for making the bandwidths of schemes based on the QR theory much more efficient.

More Natural and Efficient Extension of the Goldwasser-Micali Cryptosystem The Goldwasser-Micali cryptosystem is presented based on the QR theory. Joye and Libert [22] extended the Goldwasser-Micali cryptosystem using the 2^k-th power residue symbols, but their method is not general. The $\mathsf{IND\text{-}CPA}$ secure of the their cryptosystem is equivalent to the $\mathsf{Gap\text{-}2^k\text{-}Res}$ assumption [Section 4.1, [22]]. However, there may exist an power residue attack on this assumption. Based on the tools we construct above, we propose a new reliable hardness assumption from power residue symbols; we improve the Joye-Libert cryptosystem in terms of both ciphertext expansion and decryption speed under the new assumption. Our constructions can work for any e-th power residue symbols if e only contains small prime factors. Naturally, the lossiness and the efficiency of the Joye-Libert LTDF are improved in the same way.

Besides, we instantiate the *subgroup indistinguishability* (SG) *assumption* introduced by Brakerski and Goldwasser [5] under another new hardness assumption, namely, the *power residue assumption*. The instantiation is supported by a new theorem which generalizes the following well-known result from quadratic residues to the case of algebraic number fields.

Proposition 1. *If N is a* Blum integer *and* $\mathbb{QR}_N = \left\{x^2 \mid x \in \mathbb{Z}_N^*\right\}$ *and* $\mathbb{J}_N = \left\{\left(\frac{x}{N}\right) = 1 \mid x \in \mathbb{Z}_N^*\right\}$*, then* $\mathbb{J}_N \cong \{\pm 1\} \otimes \mathbb{QR}_N$.

Brakerski and Goldwasser gave a generic construction of schemes achieving *key-dependent security* and *auxiliary-input security* based on the SG assumption, of which DCR and QR are special cases. Hence, the scheme based on our tools and new assumption is much more efficient in bandwidth exploitation than the scheme based on the QR assumption in [5].

Organization of the Paper

In section 2, we introduce some definitions and preliminaries about power residue symbols. In section 3, we show how to efficiently compute power residue symbols defined in section 2, and give more properties of them. In section 4, we introduce our reliable hardness assumption from the power residue symbols, which generalizes the standard *quadratic residue assumption* in a natural way. In section 5, we generalize the Goldwasser-Micali cryptosystem using our tools and discuss certain implementation aspects. In section 6, we give two more applications.

2 Notation and Basic Definitions

2.1 Notations

If X is a finite set, the notation $\#X$ means the cardinality of X, writing $x \xleftarrow{\$} X$ to indicate that x is an element sampled from the uniform distribution over X. If A is an algorithm, then we write $x \leftarrow \mathsf{A}(y)$ to mean: "run 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 $\mathfrak{I}$ is an ideal of R, the relation $a - b \in \mathfrak{I}$ is written $a \equiv b \ (\mathfrak{I})$. $\otimes$ represents the direct product of two algebraic structures. log stands for the binary logarithm. $\left(\frac{\cdot}{\cdot}\right)$ stands for Jacobi symbol. φ denotes the Euler's totient function.

2.2 Power Residue Symbols

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 relatively 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 $f \in \mathbb{N}$. Notably, 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 $(\mathcal{O}_K/\mathfrak{p})^\times$ generated by $\zeta_e \bmod \mathfrak{p}$ 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: if $\alpha \in \mathfrak{p}$, then $\left(\frac{\alpha}{\mathfrak{p}}\right)_e = 0$; otherwise, $\left(\frac{\alpha}{\mathfrak{p}}\right)_e$ 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 $\left(\frac{\alpha}{\beta}\right)_e = \left(\frac{\alpha}{(\beta)}\right)_e$. Since it is well-known that $\mathcal{O}_K = \mathbb{Z}[\zeta_e]$, we simply consider the case $K = \mathbb{Q}(\zeta_e)$ from here on. We suggest interested readers to refer to [19,24,26] for more details about e-th power residue symbols.

Let e_p, e_q be positive integers and $N = pq$ be a product of two distinct primes satisfying $p \equiv 1 \pmod{e_p}$, $q \equiv 1 \pmod{e_q}$, then both p and q split completely in $\mathbb{Q}(\zeta_{e_p})$ and $\mathbb{Q}(\zeta_{e_q})$ respectively. We define the *non-degenerate* primitive (e_p, e_q)-th root of unity modulo N as an integer in $\mathbb{Z}_N^*$ which is a primitive e_p-th and e_q-th root of unity modulo p and q respectively. For example, if r_p and r_q are primitive roots modulo p and q respectively and an integer $\mu \in \mathbb{Z}_N^*$ satisfies

$$\mu \equiv r_p^{\frac{p-1}{e_p}} \bmod p \quad \text{and} \quad \mu \equiv r_q^{\frac{q-1}{e_q}} \bmod q,$$

then μ is a *non-degenerate* primitive (e_p, e_q)-th root of unity modulo N. The following lemma might be crucial for instantiations of assumptions and schemes with respect to the power residue.

Lemma 1 (Freeman *et al.* [17]). *Let e_p, e_q be positive integers, $N = pq$ be a product of two distinct primes p, q with $p \equiv 1 \pmod{e_p}$ and $q \equiv 1 \pmod{e_q}$. Let $\mu \in \mathbb{Z}_N^*$ be a* non-degenerate *primitive (e_p, e_q)-th root of unity modulo N. For each $i \in \mathbb{Z}_{e_p}^*$ and $j \in \mathbb{Z}_{e_q}^*$, let $\mathfrak{p}_i = p\mathbb{Z}[\zeta_{e_p}] + (\zeta_{e_p} - \mu^i)\mathbb{Z}[\zeta_{e_p}]$ and $\mathfrak{q}_j = q\mathbb{Z}[\zeta_{e_q}] + (\zeta_{e_q} - \mu^j)\mathbb{Z}[\zeta_{e_q}]$, then we have $\mathrm{Norm}(\mathfrak{p}_i) = p$, $p\mathbb{Z}[\zeta_{e_p}] = \prod_{i \in \mathbb{Z}_{e_p}^*} \mathfrak{p}_i$ and $\mathrm{Norm}(\mathfrak{q}_j) = q$, $q\mathbb{Z}[\zeta_{e_q}] = \prod_{j \in \mathbb{Z}_{e_q}^*} \mathfrak{q}_j$. In particular, if $e_p = e_q = e$, we may define $\mathfrak{a}_i = N\mathbb{Z}[\zeta_e] + (\zeta_e - \mu^i)\mathbb{Z}[\zeta_e]$ and we furthermore have $\mathrm{Norm}(\mathfrak{a}_i) = N$, $\mathfrak{a}_i = \mathfrak{p}_i\mathfrak{q}_i$ for each $i \in \mathbb{Z}_e^*$ and $N\mathcal{O}_K = \prod_{i \in \mathbb{Z}_e^*} \mathfrak{a}_i$.*

Notations. In the rest of this paper, we will frequently use the notations in Lemma 1. The ideals $\mathfrak{p}_1$ and $\mathfrak{q}_1$ are mainly considered in the following discussion. As a matter of convenience, we denote $\mathfrak{p}_1$, $\mathfrak{q}_1$ and $\mathfrak{a}_1$ by $\mathfrak{p}$, $\mathfrak{q}$ and $\mathfrak{a}$ respectively.

3 Computation and Properties of Power Residue Symbols

In this section, we show how to compute the power residue symbols with respect to $\mathfrak{p}_1$ and $\mathfrak{q}_1$ efficiently, together with the factorization of N known. We also investigate more properties about power residue symbols.

3.1 Computing Power Residue Symbols

For a general integer e and an ideal in $\mathbb{Z}[\zeta_e]$, it is really tough to design an efficient and deterministic algorithm to compute e-th power residue symbols since we can hardly find a deterministic way to decrease the norm of ideals. In fact, efficient and deterministic algorithms are only known in the case of $e \in \{2, 3, 4, 5, 7[12], 8[20], 11[21], 13[7]\}$. The general case is tackled probabilistically by Squirrel [32] and Boer [16]. However, their algorithms are not quite efficient and no rigorous proof has been found to prove that they run in polynomial time. For a composite e, Freeman et al. [17] constructed the following "compatibility" identity [3] to decrease the size of the power e so as to reduce the amount of computation.

Proposition 2. *With notations as in Lemma 1. Let f be integers with $f \mid e_p$ and $x \in \mathbb{Z}[\zeta_{e_p}]$. Then*

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

[3] Freeman et al. claimed that the identity 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 ensures the identity held. See Chapter 1 in [16]. However, note that in the case of $\mathrm{Norm}_{\mathbb{Z}[\zeta_e]}(\mathfrak{U}) = p - 1$, the identity also holds due to the inclusion map $\iota : \mathbb{Z}[\zeta_e]/\mathfrak{U} \mapsto \mathbb{Z}[\zeta_f]/\mathfrak{B}$.

It follows readily that

$$\mathfrak{p} \cap \mathbb{Z}[\zeta_f] = p\mathbb{Z}[\zeta_f] + (\zeta_f - \mu^{\frac{e_p}{f}})\mathbb{Z}[\zeta_f]$$

due to the fact that $\mu^{\frac{e_p}{f}}$ is a *non-degenerate* primitive $(f,1)$-th root of unity modulo N. Therefore, we are able to learn the value of $\left(\frac{x}{\mathfrak{p}}\right)_{e_p}$ by computing $\left(\frac{x}{\mathfrak{p}\cap\mathbb{Z}[\zeta_f]}\right)_f$ for each coprime factor of e_p and applying the Chinese remainder theorem.

Quadratic and power residues are useful building-blocks of many cryptographic applications. In most applications, the factorization of N is transparent to participants who want to get the values of higher power residue symbols. Therefore, we don't necessarily consider the computation in the general case. We can actually do better with the factorization of N. The following simple theorem demonstrates that computing $\left(\frac{\cdot}{\mathfrak{p}}\right)_{e_p}$ (and hence also $\left(\frac{\cdot}{\mathfrak{q}}\right)_{e_q}$) is closely related to solving the discrete logarithm problem in a specific 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 1. *With notations as in Lemma 1, we deduce that* $\left(\frac{y}{\mathfrak{p}}\right)_{e_p} = \zeta_{e_p}^x$ *if and only if* $\mu^x = y^{\frac{p-1}{e_p}}$ *in* $\mathbb{Z}_p^*$. *Therefore, the solution to the DLP in the finite cyclic subgroup* $\langle\mu\rangle$ *of order* e_p *allows the computation of* $\left(\frac{\cdot}{\mathfrak{p}}\right)_{e_p}$.

Proof. If $\mu^x = y^{\frac{p-1}{e_p}}$ in $\mathbb{Z}_p^*$, then $y^{\frac{p-1}{e_p}} - \zeta_e^x = \mu^x - \zeta_{e_p}^x \in \mathfrak{p}$. It follows that $\left(\frac{y}{\mathfrak{p}}\right)_{e_p} = \zeta_{e_p}^x$. Conversely, If $\left(\frac{y}{\mathfrak{p}}\right)_{e_p} = \zeta_{e_p}^x$ for some $x \in \mathbb{Z}_{e_p}$, that is $y^{\frac{p-1}{e_p}} - \zeta_{e_p}^x \in \mathfrak{p}$. Since the order of $y^{\frac{p-1}{e_p}}$ divides e_p, it can be expressed as μ^z for $z \in \mathbb{Z}_{e_p}$, which implies $\mu^x - \mu^z \in \mathfrak{p}$ and hence $\mu^x = \mu^z$. The fact that the order of μ is e_p forces $x = z$. ∎

Although the DLP is considered in general to be intractable, it can be easily solved in a few particular cases, e.g., if the order of $\mathbb{G}$ is smooth, the Pohlig-Hellman algorithm [30] turns out to be quite efficient. In other words, if e_p is chosen with appropriate prime factors and given the factorization of N, we can get the value of $\left(\frac{\cdot}{\mathfrak{p}}\right)_{e_p}$ by using only the Pohlig-Hellman algorithm. In practice, e_p is usually set to be a prime power.

Algorithm 1 Pohlig-Hellman algorithm for prime powers

Input: an integer $\mu \in \mathbb{Z}_p$ of order e^k where e and p are primes with $e^k \mid p-1$ and $x \in \langle \mu \rangle$

Output: $\boldsymbol{y} = (y_{k-1}, \ldots, y_0)_e$ such that $\mu^{y} \equiv x \pmod{p}$

1: Compute the values $\mu^{e^{k-1}i}$ for each $0 \leq i < e$ and store them in a lookup table
2: Compute the values μ^{-e^i} for each $0 \leq i < k$ and store them in a lookup table
3: $x_0 \longleftarrow x$
4: Call the hash algorithm to find $y_0 \in \mathbb{Z}_e$ such that $\mu^{e^{k-1}y_0} \equiv x_0^{e^{k-1}} \bmod p$
5: **for** $1 \leq i \leq k-1$ **do**
6: $\quad x_i \longleftarrow x_{i-1}\mu^{-y_{i-1}e^{i-1}} \bmod p$
7: $\quad$ Call the hash algorithm to find $y_i \in \mathbb{Z}_e$ such that $\mu^{e^{k-1}y_i} \equiv x_i^{e^{k-i-1}} \bmod p$
8: **end for**
9: **return** $\boldsymbol{y} = (y_{k-1}, \ldots, y_0)_e$

Remark 1. The above algorithm can be made a little faster. According to the step 6 above, we have

$$x_i^{e^{k-i-1}} \equiv x_{i-1}^{e^{k-i-1}} \mu^{-y_{i-1}e^{k-2}} \bmod p$$

We can cut down the number of operations of computing $x_i^{e^{k-i-1}}$ by evaluating

$$x_{i-1}^{e^{k-i-1}} \quad \text{and} \quad x_{i-1}^{e^{k-(i-1)-1}} \equiv \left(x_{i-1}^{e^{k-i-1}}\right)^e \bmod p$$

successively in the previous step. So the way to optimize the algorithm is: we record the values of $x_{i-1}^{e^{k-i-1}}$ for odd indices i, then the number of operations of computing each even part is highly reduced.

3.2 Some Properties of Power Residue Symbols

In this section, we assume $e_p = e_q = e$. For an arbitrary $k \geq 2$, we say an integer $x \in \mathbb{Z}_N^*$ is a *k-th residue* modulo N if there exists an integer $y \in \mathbb{Z}_N^*$ such that $y^k \equiv x \pmod{N}$. Note that if x is an *e-th residue* modulo N, then we have $\left(\frac{x}{\mathfrak{p}_i}\right)_e = \left(\frac{x}{\mathfrak{q}_i}\right)_e = 1$ for each $i \in \mathbb{Z}_e^*$. Just as for quadratic residue, we denote the set of all *e-th residues* in $\mathbb{Z}_N^*$ by $\mathcal{ER}_N^e$. Correspondingly, the set $\left\{x \in \mathbb{Z}_N^* \;\middle|\; \left(\frac{x}{\mathfrak{a}}\right)_e = 1\right\}$ is denoted by $\mathbb{J}_N^e$.

Theorem 2. *With notations as in Lemma 1, assume that $e_p = e_q = e$ and let $\mathcal{ER}_m^e = \left\{x^e \text{ mod } m \;\middle|\; x \in \mathbb{Z}_m^*\right\}$ denote the set of all e-th residues in $\mathbb{Z}_m^*$ and let $\mathscr{U} = \left\{1, \zeta_e, \ldots, \zeta_e^{e-1}\right\}$ denote the multiplicative subgroup of roots of unity in $\mathbb{Z}[\zeta_e]$, then*

(1) $\mathbb{Z}_p^*/\mathcal{ER}_p^e \cong \mathscr{U}$ (and hence also $\mathbb{Z}_q^*/\mathcal{ER}_q^e \cong \mathscr{U}$)

(2) If we require that $\gcd(\frac{p-1}{e}, e) = \gcd(\frac{q-1}{e}, e) = 1$, then there is an integer ν such that

- ν *is a* non-degenerate *primitive* (e,e)*-th root of unity modulo* N*.*
- $\left(\frac{\nu}{\mathfrak{a}_i}\right)_e = 1$ *for every ideal* $\mathfrak{a}_i \subset \mathbb{Z}[\zeta_e]$*.*

Furthermore, we have

$$\mathbb{J}_N^e = \langle \nu \rangle \otimes \mathcal{ER}_N^e$$

Proof.

(1) Consider the homomorphism $\theta : \mathbb{Z}_p^* \to \mathscr{U}$ defined by $x \mapsto \left(\frac{x}{\mathfrak{p}}\right)_e$. Since the number of roots of the polynomial $f(x) = x^{\frac{p-1}{e}} - 1$ over the field $\mathbb{Z}[\zeta_e]/\mathfrak{p}$ is at most $\frac{p-1}{e}$ and the cardinality of $\mathcal{ER}_p^e$ is exactly $\frac{p-1}{e}$, an integer $z \in \mathbb{Z}_p^*$ satisfying $\left(\frac{z}{\mathfrak{p}}\right)_e = 1$ must be in $\mathcal{ER}_p^e$. Hence the kernel of θ is $\mathcal{ER}_p^e$ and we have the desired isomorphism due to the fact that the cardinality of left hand side is equal to the cardinality of right hand side. Of course, elements in different cosets of $\mathcal{ER}_p^e$ in $\mathbb{Z}_p^*$ have different e-th power residue symbols, and there is a one to one correspondence between the cosets of $\mathcal{ER}_p^e$ in $\mathbb{Z}_p^*$ and the e-th roots of unity via the e-th power residue symbols.

(2) The condition $\gcd(\frac{p-1}{e}, e) = \gcd(\frac{q-1}{e}, e) = 1$ implies that there exist integers $s_p \in \mathbb{Z}_e^*$, $t_p, s_q \in \mathbb{Z}_e^*$, t_q such that $s_p \frac{p-1}{e} + t_p e = s_q \frac{q-1}{e} + t_q e = 1$. Let μ_p be μ mod p and μ_q be μ mod q. Observe that every primitive e-th root of unity in $\mathbb{Z}_p$ has the form μ_p^i for some $i \in \mathbb{Z}_e^*$. It follows that

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

Similarly,

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

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

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

Since $\nu \in \mathbb{Z}$, the result $\left(\frac{\nu}{\mathfrak{a}_i}\right)_e = 1$ follows from the Galois equivalence. To prove the last statement we only need to prove that every element of $\mathbb{J}_N^e$ can be written as a product of two elements in $\langle \nu \rangle$ and $\mathcal{ER}_N^e$ respectively as $\langle \nu \rangle \cap \mathcal{ER}_N^e = \varnothing$. For any $x \in \mathbb{J}_N^e$, since there exists $j \in \mathbb{Z}_e$ such that $\left(\frac{\nu^j}{\mathfrak{p}}\right) = \left(\frac{x}{\mathfrak{p}}\right)$ and $\left(\frac{\nu^j}{\mathfrak{q}}\right) = \left(\frac{x}{\mathfrak{q}}\right)$, we have $x \equiv \nu^j y^e \bmod p$ and $x \equiv \nu^j z^e \bmod q$ for some $x \in \mathbb{Z}_p^*$ and $y \in \mathbb{Z}_q^*$ from (1). Take $w \equiv y \bmod p$ and $w \equiv z \bmod q$, then we have $x \equiv \nu^j w^e \bmod N$, as desired. ■

Remark 2. In particular, if $e = 2$, then we deduce from Theorem 2 the well-known theorem which says that $\mathbb{J}_N \cong \{\pm 1\} \otimes \mathbb{QR}_N$ where N is a *Blum integer* and $\mathbb{QR}_N = \left\{x^2 \mid x \in \mathbb{Z}_N^*\right\}$ and $\mathbb{J}_N = \left\{\left(\frac{x}{N}\right) = 1 \mid x \in \mathbb{Z}_N^*\right\}$.

4 A New Assumption from Power Residue Symbols

In this section, we shall give a formal definition of the assumption our cryptosystem relies on. We start with the following definition of a set which is contrary to $\mathcal{ER}_N^{\mathrm{lcm}(e_p,e_q)}$. Let e denote $\gcd(p-1,q-1)$ we define

$$\mathcal{J}_N^{(e_p,e_q)} = \left\{ x \in \mathbb{Z}_N^* \;\middle|\; \left(\frac{x}{\mathfrak{a}}\right)_e = 1,\ \left(\frac{x}{\mathfrak{p}}\right)_{e_p} \text{ and } \left(\frac{x}{\mathfrak{q}}\right)_{e_q} \text{ are primitive} \right\}.$$

Note that the condition $\left(\frac{x}{\mathfrak{a}}\right)_e = 1$ ensures that $\left(\frac{x}{\mathfrak{a}_i}\right)_e = 1$ for each $i \in \mathbb{Z}_e^*$ by the Galois equivalence, hence $\left(\frac{x}{N\mathbb{Z}[\zeta_e]}\right)_e = 1$ [4].

Definition 1 ((e_p,e_q)-th Residue ((e_p,e_q)-ER) Assumption). *Given a security parameter κ. A PPT algorithm $\mathsf{RSAgen}(\kappa)$ generates two integers e_p and e_q and a random RSA modulus $N = pq$ such that $p \equiv 1 \bmod e_p$ and $q \equiv 1 \bmod e_q$, and chooses at random $\mu \in \mathbb{Z}_N^*$ a* non-degenerate *primitive (e_p,e_q)-th root of unity modulo N. The (e_p,e_q)-ER assumption with respect to $\mathsf{RSAgen}(\kappa)$ asserts that the advantage $\mathsf{Adv}_{\mathcal{A},\mathsf{RSAgen}}^{(e_p,e_q)\text{-}\mathsf{ER}}(\kappa)$ defined as*

$$\left| \mathsf{Pr}\left(\mathcal{A}(N,x,\mathrm{lcm}(e_p,e_q)) = 1 \;\middle|\; x \xleftarrow{\$} \mathcal{ER}_N^{\mathrm{lcm}(e_p,e_q)} \right) - \mathsf{Pr}\left(\mathcal{A}(N,x,\mathrm{lcm}(e_p,e_q)) = 1 \;\middle|\; x \xleftarrow{\$} \mathcal{J}_N^{(e_p,e_q)} \right) \right|$$

is negligible for any PPT adversary $\mathcal{A}$; the probabilities are taken over the experiment of running $(N,(e_p,e_q),\mu) \leftarrow \mathsf{RSAgen}(\kappa)$ and choosing at random $x \in \mathcal{ER}_N^{\mathrm{lcm}(e_p,e_q)}$ and $x \in \mathcal{J}_N^{(e_p,e_q)}$.

Remark 3. The $(2,1)$-ER assumption is equivalent to the standard QR assumption. The $(2^k,1) - \mathsf{ER}$ assumption is equivalent to the Gap-2^k-Res assumption with $q \equiv 3 \pmod 4$ defined in [Definition 4, [22]] because $x \in \mathbb{J}_N \setminus \mathbb{QR}_N$ if and only if $x \in \mathbb{J}_N$ and $\left(\frac{x}{\mathfrak{p}}\right)_{2^k}$ is primitive (for an arbitrary μ).

5 A New Homomorphic Public-Key Cryptosystem

We generalize the Goldwasser-Micali cryptosystem as well as the Joye-Libert cryptosystem. Our new homomorphic cryptosystem can efficiently encrypt larger messages than both of them and the decryption is much faster than that of the Joye-Libert cryptosystem.

[4] The IND-CPA secure of the Joye-Libert cryptosystem is equivalent to the Gap-2^k-Res assumption [Section 4.1, [22]], which was considered in [1] by Abdalla, Ben Hamouda and Pointcheval. However, the hardness of this assumption depends on the choice of q in fact (recall that $p \equiv 1 \bmod 2^k$). In detail, if 2^ℓ $(2 \le \ell \le k)$ is a common divisor of $p-1$ and $q-1$, the symbol $\left(\frac{x}{N\mathbb{Z}[\zeta_{2^\ell}]}\right)_{2^\ell}$ must be equal to 1 for each $x \in \mathcal{ER}_N^{2^\ell}$, but it is incorrect when x is chosen from $\mathbb{J}_N \setminus \mathbb{QR}_N$. In this case, the generic algorithms introduced in the first paragraph of Section 3.1 may break this assumption. Even if $\ell = 2$, the Joye-Libert cryptosystem may leak 1-bit information of a plaintext.

5.1 Description

The setting of our new cryptosystem (denoted by Π) is essentially the same as the Goldwasser-Micali cryptosystem and the Joye-Libert cryptosystem. More precisely, the setting $e_p = e_q = 2$ corresponds to the Goldwasser-Micali cryptosystem and the setting $e_p = 2^k$, $e_q = 1$ corresponds to the Joye-Libert cryptosystem.

$\mathsf{KeyGen}\,(1^\kappa)$ Given a security parameter κ. KeyGen selects smooth integers e_p and e_q, then generates an RSA modulus $N = pq$ a product of two large and equally sized primes p and q such that $e_p \mid p-1, e_q \mid q-1$ and picks at random $\mu \in \mathbb{Z}_N^*$ a *non-degenerate* primitive (e_p, e_q)-th root of unity modulo N and $y \xleftarrow{\$} \mathcal{J}_N^{(e_p,e_q)}$. The public and private keys are $\mathsf{pk} = \{N, \mathrm{lcm}(e_p, e_q), y\}$ and $\mathsf{sk} = \{p, q, e_p, e_q, \mu\}$.

$\mathsf{Enc}\,(\mathsf{pk}, m)$ To encrypt a message $m \in \mathbb{Z}_{\mathrm{lcm}(e_p,e_q)}$, Enc picks a random $r \in \mathbb{Z}_N^*$ and returns the ciphertext

$$c = y^m r^{\mathrm{lcm}(e_p,e_q)} \bmod N.$$

$\mathsf{Dec}\,(\mathsf{sk}, c)$ Given the ciphertext c and the private key $\mathsf{sk} = \{p, q, e_p, e_q, \mu\}$, Dec first computes $\left(\frac{c}{\mathfrak{p}}\right)_{e_p} = \zeta_{e_p}^{z_p}$ and $\left(\frac{c}{\mathfrak{q}}\right)_{e_q} = \zeta_{e_q}^{z_q}$ by means of Theorem 1. Then, it recovers the message $m \in \mathbb{Z}_{\mathrm{lcm}(e_p,e_q)}$ from

$$m \equiv z_p k_p^{-1} \bmod e_p \quad \text{and} \quad m \equiv z_q k_q^{-1} \bmod e_q$$

by using the Chinese Remainder Theorem with non-pairwise coprime moduli, where $\left(\frac{y}{\mathfrak{p}}\right)_{e_p} = \zeta_{e_p}^{k_p}$ and $\left(\frac{y}{\mathfrak{q}}\right)_{e_q} = \zeta_{e_q}^{k_q}$ are pre-computed.

5.2 Security analysis

The cryptosystem Π also has the similar security analysis as for the Goldwasser-Micali cryptosystem.

Theorem 3. *The cryptosystem Π is* IND-CPA *secure under the (e_p, e_q)*-ER *assumption.*

Proof. Consider changing the distribution of the public key. Under the (e_p, e_q)-ER assumption, we may choose y uniformly in $\mathcal{ER}_N^{\mathrm{lcm}(e_p,e_q)}$ instead of choosing it from $\mathcal{J}_N^{(e_p,e_q)}$, while this is done without noticing the adversary. In this case, the ciphertext carries no information about the message and hence Π is IND-CPA secure.

5.3 Parameter Selection

The key generation requires two primes p and q such that $e_p \mid p-1$ and $e_q \mid p-1$, where e_p and e_q are better to be chosen so that they are powers of small primes in practice. The algorithm to produce p and q is similar in spirit to the algorithm described in [Section 5.1, [22]]. The major difference is that the size of $\log e_p + \log e_q$ is bounded by $\frac{1}{2}\log N$. The reason is provided by the following proposition [Lemma 8, [33]] related to Coppersmith's method for finding small roots of bivariate modular equations.

Proposition 3. *Let p and q be equally sized primes and $N = pq$. Let e be a divisor of $\varphi(N) = (p-1)(q-1)$. If there exists a positive constant c such that $e > N^{\frac{1}{2}+c}$ holds, then there exists a* PPT *algorithm that given N and e, it factorizes N.*

Note that taking $\log e_p + \log e_q$ to be $\frac{1}{2}\log N$ does not contradict the setting of *Φ-Hiding Assumption* [8] as the prime factors of $\varphi(N)$ are very small. However, $\log e_p + \log e_q$ shall not be close to $\frac{1}{2}\log N$ because we don't know whether there exists an attack of mixing together Coppersmith's attack and exhaustive searches. In particular, if we take $e_p = 2^k$, $e_q = 2$ and $k > \frac{1}{4}\log N$, the low-order $\frac{1}{4}\log N$ bits of p is revealed to an adversary, and hence it can factorize N by implementing Coppersmith's attack [15]. Therefore, if we choose e_p and e_q not to be a power of 2 and to be coprime, we may handle messages at least twice as long as the Joye-Libert cryptosystem does. The key generation also requires a random integer $y \in \mathbb{Z}_N^*$ in $\mathcal{J}_N^{(e_p,e_q)}$. We can use (2) in Theorem 2 for uniformly sampling integers in $\mathbb{J}_N^{\gcd(p-1,q-1)}$. A random integer modulo N has a probability of exactly $\frac{\varphi(e_p)\varphi(e_q)}{e_p e_q}$ of being in the set

$$\left\{ x \in \mathbb{Z}_N^* \;\middle|\; \left(\frac{x}{\mathfrak{p}}\right)_{e_p} \text{ and } \left(\frac{x}{\mathfrak{q}}\right)_{e_q} \text{ are primitive} \right\}.$$

If we take $e_p = e_1^{f_1}$ and $e_q = e_2^{f_2}$ where e_1 and e_2 are distinct primes, the above probability is equal to $\frac{(e_1-1)(e_2-1)}{e_1 e_2}$. Therefore, a suitable $y \in \mathcal{J}_N^{(e_p,e_q)}$ is likely to be obtained after several trials.

5.4 Performance and Comparisons

Now, we investigate the performance of our cryptosystem and make comparisons with the Paillier cryptosystem [28] and the Joye-Libert cryptosystem [22], two famous schemes in the literature on homomorphic encryption.

All the three cryptosystems require the generation of two large suitable primes. Though both the cryptosystem Π and Joye-Libert cryptosystem need to select other elements, it altogether takes a negligible amount of time compared with the selection of the primes.

It is easy to see that the Paillier cryptosystem takes about four times as long as the Π or the Joye-Libert cryptosystem to encrypt messages or perform homomorphic operations because the modular multiplications are computed over $\mathbb{Z}_{N^2}^*$.

One major drawback of the Joye-Libert cryptosystem is that its decryption [Algorithm 1, [22]] is slow. When decrypting a 128-bit message, it needs roughly

$$\log p - 128 + \frac{128(128-1)}{4} + \frac{128}{2} = \log p + 4000$$

modular multiplications over $\mathbb{Z}_p^*$ on average according to the remark following [Algorithm 1, [22]]. However, if we take $e_p = 929^{13} > 2^{128}$ and $e_q = 1$, the major time consuming part of Π's decryption is performing the Pohlig-Hellman algorithm to compute $\left(\frac{\cdot}{\mathfrak{p}}\right)_{e_p}$. If the storage is enough, in order to speed up, we may pre-evaluate the quantities $\mu^{929^{13}k} \bmod p$ for $k = 0, 1, \ldots, 928$ and $\mu^{-929^j} \bmod p$ for $j = 0, 1, \ldots, 12$ in a lookup table. If we ignore the constant time which it spends on the hash algorithm, then the decryption only requires

$$\log p - 128 + \sum_{\substack{k=0 \\ k \text{ is even}}}^{12} \log(929^k) + 128 \approx \log p + 414$$

modular multiplications over $\mathbb{Z}_p^*$ on average according to the remark following Algorithm 3.1. If N is taken as 1024 bits, the decryption of Π is approximately 5 times faster than that of the Joye-Libert cryptosystem. Also, it is easy to see that the larger the e_p is, the faster Π's encryption is and the larger the storage space Π will require. Even though we do not use the lookup table, Π's decryption still runs faster than that of Joye-Libert cryptosystem.

Comparatively, the advantage of the Paillier cryptosystem is that the ciphertext expansion is small and it supports homomorphic operations over larger messages. The Π and the Joye-Libert cryptosystem have better performance of performing smaller or specifically sized messages. For example, as mentioned in [22], they can be used to encrypt a 128- or 256-bit symmetric key in a KEM/DEM construction [31].

6 Applications

6.1 Circular and Leakage Resilient Public-Key Encryption

Brakerski and Goldwasser introduced the notion of *subgroup indistinguishability* (SG) *assumption* in [Section 3.1, [5]]. They instantiated the SG assumption based on the QR and the DCR assumptions and proposed a generic construction of schemes which achieved *key-dependent security* and *auxiliary-input security* based on the SG assumption. However, the scheme based on the QR assumption can only encrypt a 1-bit message at a time. In this section, we will show how to instantiate the SG assumption under the new hardness assumption called power residue assumption. In this way, the scheme becomes much more efficient in bandwidth exploitation.

Subgroup Indistinguishability Assumption Under the Power Residue Assumption Let e be an integer with small prime factors. We sample a random RSA modulus $N = pq$ such that $e \mid p-1$, $e \mid q-1$ and $\gcd(\frac{p-1}{e}, e) = \gcd(\frac{q-1}{e}, e) = 1$. Let $\mathcal{ER}_N^e$ and $\mathbb{J}_N^e$ be as in Section 3.2, then we have shown there exists a $\nu \in \mathbb{J}_N^e \setminus \mathcal{ER}_N^e$ such that $\mathbb{J}_N^e = \langle \nu \rangle \otimes \mathcal{ER}_N^e$ from (2) in Theorem 2. The groups $\mathbb{J}_N^e$, $\langle \nu \rangle$ and $\mathcal{ER}_N^e$ have orders $\frac{\varphi(N)}{e}$, e and $\frac{\varphi(N)}{e^2}$ respectively and we denote $\frac{\varphi(N)}{e}$ by N'. The condition $\gcd(\frac{p-1}{e}, e) = \gcd(\frac{q-1}{e}, e) = 1$ implicates that $\gcd(e, \frac{\varphi(N)}{e^2}) = 1$. We define the following power residue assumption which is similar to the (e_p, e_q)-ER assumption defined previously.

Definition 2 (Power Residue (PR) Assumption). *Given a security parameter κ. A PPT algorithm $\mathsf{RSAgen}(\kappa)$ generates an integer e with small prime factors and a random RSA modulus $N = pq$ such that $e \mid p-1$, $e \mid q-1$ and $\gcd(\frac{p-1}{e}, e) = \gcd(\frac{q-1}{e}, e) = 1$, and chooses at random $\mu \in \mathbb{Z}_N^*$ a* non-degenerate *primitive (e,e)-th root of unity modulo N. The PR assumption with respect to $\mathsf{RSAgen}(\kappa)$ asserts that the advantage $\mathsf{Adv}^{\mathsf{PR}}_{\mathcal{A},\mathsf{RSAgen}}(\kappa)$ defined as*

$$\left| \Pr\left(\mathcal{A}(N, x, e) = 1 \;\middle|\; x \xleftarrow{\$} \mathcal{ER}_N^e \right) - \Pr\left(\mathcal{A}(N, x, e) = 1 \;\middle|\; x \xleftarrow{\$} \mathbb{J}_N^e \right) \right|$$

is negligible for any PPT adversary $\mathcal{A}$; the probabilities are taken over the experiment of running $(N, e, \mu) \leftarrow \mathsf{RSAgen}(\kappa)$ and choosing at random $x \in \mathcal{ER}_N^e$ and $x \in \mathbb{J}_N^e$.

Since there exist efficient sampling algorithms that sample a random element from $\mathcal{ER}_N^e$ and $\mathbb{J}_N^e$ according to Theorem 2, the PR assumption leads immediately to the instantiation of the SG assumption by setting $\mathbb{G}_U = \mathbb{J}_N^e$, $\mathbb{G}_M = \langle \nu \rangle$, $\mathbb{G}_L = \mathcal{ER}_N^e$, $h = \nu$, and $T = N \geq eN'$.

6.2 Constructing Lossy Trapdoor Functions from the (e_p, e_q)-th Residue Assumption

Lossy Trapdoor Functions *Lossy trapdoor functions* (LTDF) were introduced by Peikert and Waters [29] and since then numerous applications emerge in cryptography. Informally speaking, LTDF consist of two families of functions. The functions in one family are injective trapdoor functions, while functions in the other family are lossy, that is, the image size is smaller than the domain size. It also requires that the functions sampled from the first and the second family are computationally indistinguishable. Using the constructions in [29], one can obtain CCA-secure public-key encryptions. So far, LTDF are mainly constructed from assumptions such as DDH [29], LWE [29], QR [17], DCR [17], *Φ-Hiding* [23], etc.

Joye and Libert constructed a LTDF with short outputs and keys based on the k-QR, k-SJS and DDH assumptions in [22]. Of course, it is an easy matter to generalize their constructions, using our techniques based on the power residue symbols. Hence, we only propose a new generic construction of the LTDF and the corresponding conclusions. We follow the definition of the LTDF in [22] and omit the security analysis since it proceeds in exactly the same way in [22].

$\mathsf{InjGen}(1^\kappa)$ Given a security parameter κ, let ℓ_N, k and n (n is a multiple of k) be parameters determined by κ. InjGen defines $m = \frac{n}{k}$ and performs the following steps.

1. Select smooth integers e_p and e_q such that $k < \log(e_p) + \log(e_q) < \frac{\ell_N}{2}$. Generate an ℓ_N-bit RSA modulus $N = pq$ such that $p-1 = e_p p', q-1 = e_q q'$ for large primes p, q, p', q'. Pick at random $\mu \in \mathbb{Z}_N^*$ a *non-degenerate* primitive (e_p, e_q)-th root of unity modulo N and $y \xleftarrow{\$} \mathcal{J}_N^{(e_p,e_q)}$.
2. For each $i \in \{1, \dots, m\}$, pick h_i in $\mathcal{ER}_N^{\mathrm{lcm}(e_p,e_q)}$ at random.
3. Choose $r_1, \dots, r_m \xleftarrow{\$} \mathbb{Z}_{p'q'}$ and compute a $m \times m$ matrix $Z = (Z_{i,j})$ with

$$Z_{i,j} = \begin{cases} y \cdot h_j^{r_i} \bmod N, & \text{if } i = j; \\ h_j^{r_i} \bmod N, & \text{otherwise.} \end{cases}$$

Output the evaluation key $\mathsf{ek} = \{N, Z\}$ and the secret key $\mathsf{sk} = \{p, q, e_p, e_q, \mu, y\}$.

$\mathsf{LossyGen}(1^\kappa)$ The process of $\mathsf{LossyGen}$ is identical to the process of InjGen, except that

- Set $Z_{i,j} = h_j^{r_i} \bmod N$ for each $1 \le i, j \le m$.
- $\mathsf{LossyGen}$ does not output the secret key sk.

$\mathsf{Evaluation}(\mathsf{ek}, x)$ Given $\mathsf{ek} = \left\{N, Z = (Z_{i,j})_{i,j \in \{1,\dots,m\}}\right\}$ and a message $x \in \{0,1\}^n$, $\mathsf{Evaluation}$ parses x as a k-adic string $\boldsymbol{x} = (x_1, \dots, x_m)$ with $x_i \in \mathbb{Z}_{2^k}$ for each i. Then, it computes and returns $\boldsymbol{y} = (y_1, \dots, y_m) \in \mathbb{Z}_N^m$ with $y_j = \prod_{i=1}^m Z_{i,j}^{x_i} \bmod N$.

$\mathsf{Inversion}(\mathsf{sk}, \boldsymbol{y})$ Given $\mathsf{sk} = \{p, q, e_p, e_q, \mu, y\}$ and $\boldsymbol{y} = (y_1, \dots, y_m) \in \mathbb{Z}_N^m$, $\mathsf{Inversion}$ applies the decryption algorithm $\mathsf{Dec}(\mathsf{sk}, y_j)$ of the Π for each y_j to recover x_j for $j = 1$ to m. It recovers and outputs the input $x \in \{0,1\}^n$ from the resulting vector $\boldsymbol{x} = (x_1, \dots, x_m) \in \mathbb{Z}_{2^k}^m$.

Proposition 4. *Let $\ell = n - \log(p'q')$. The above construction is a (n, ℓ)-LTDF if the (e_p, e_q)-th residue assumption holds and the* DDH *assumption holds in the subgroup $\mathcal{ER}_N^{\mathrm{lcm}(e_p,e_q)}$.*

Clearly, our new proposed LTDF outperforms the Joye-Libert LTDF in terms of its fast decryption and small ciphertext expansion. The lossiness may also be improved as there are no known attacks against the factorization of N when $\frac{\ell_N}{4} < \log(e_p) + \log(e_q) < \frac{\ell_N}{2}$.

References

1. Michel Abdalla, Fabrice Ben Hamouda, and David Pointcheval. Tighter reductions for forward-secure signature schemes. In *International Workshop on Public Key Cryptography*, pages 292–311. Springer, 2013.
2. Josh Daniel Cohen Benaloh. Verifiable secret-ballot elections. 1989.

3. Lenore Blum, Manuel Blum, and Mike Shub. A simple unpredictable pseudo-random number generator. *SIAM Journal on computing*, 15(2):364–383, 1986.
4. Manuel Blum and Shafi Goldwasser. An efficient probabilistic public-key encryption scheme which hides all partial information. In *Workshop on the Theory and Application of Cryptographic Techniques*, pages 289–299. Springer, 1984.
5. Zvika Brakerski and Shafi Goldwasser. Circular and leakage resilient public-key encryption under subgroup indistinguishability. In *Annual Cryptology Conference*, pages 1–20. Springer, 2010.
6. Éric Brier, Houda Ferradi, Marc Joye, and David Naccache. New number-theoretic cryptographic primitives. *Journal of Mathematical Cryptology*, 2019.
7. Eric Brier and David Naccache. The thirteenth power residue symbol. Cryptology ePrint Archive, Report 2019/1176, 2019. `https://eprint.iacr.org/2019/1176`.
8. Christian Cachin, Silvio Micali, and Markus Stadler. Computationally private information retrieval with polylogarithmic communication. In *International Conference on the Theory and Applications of Cryptographic Techniques*, pages 402–414. Springer, 1999.
9. Zhenfu Cao. A type of public key cryptosystem based on eisenstein ring $\mathbb{Z}[\omega]$. *Proceedings of the 3rd Chinese Conference of Source Coding, Channel Coding and Cryptography*, pages 178–186, 1988.
10. 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.
11. Zhenfu Cao, Xiaolei Dong, Licheng Wang, and Jun Shao. More efficient cryptosystems from k-th power residues. *IACR Cryptology ePrint Archive*, 2013:569, 2013.
12. Perlas C Caranay and Renate Scheidler. An efficient seventh power residue symbol algorithm. *International Journal of Number Theory*, 6(08):1831–1853, 2010.
13. Michael Clear and Ciaran McGoldrick. Additively homomorphic ibe from higher residuosity. In *IACR International Workshop on Public Key Cryptography*, pages 496–515. Springer, 2019.
14. Josh D Cohen and Michael J Fischer. *A robust and verifiable cryptographically secure election scheme*. Yale University. Department of Computer Science, 1985.
15. Don Coppersmith. Small solutions to polynomial equations, and low exponent rsa vulnerabilities. *Journal of cryptology*, 10(4):233–260, 1997.
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. Evaluating octic residue symbols.
21. Marc Joye, Oleksandra Lapiha, Ky Nguyen, and David Naccache. The eleventh power residue symbol. *Preprint available at URL `https://eprint.iacr.org/2019/870`*.
22. 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.
23. Eike Kiltz, Adam O'Neill, and Adam Smith. Instantiability of rsa-oaep under chosen-plaintext attack. *Journal of Cryptology*, 30(3):889–919, 2017.

24. Franz Lemmermeyer. *Reciprocity laws: from Euler to Eisenstein*. Springer Science & Business Media, 2013.
25. David Naccache and Jacques Stern. A new public key cryptosystem based on higher residues. In *Proceedings of the 5th ACM conference on Computer and communications security*, pages 59–66, 1998.
26. Jürgen Neukirch. *Algebraic number theory*, volume 322. Springer Science & Business Media, 2013.
27. Tatsuaki Okamoto and Shigenori Uchiyama. A new public-key cryptosystem as secure as factoring. In *International conference on the theory and applications of cryptographic techniques*, pages 308–318. Springer, 1998.
28. Pascal Paillier. Public-key cryptosystems based on composite degree residuosity classes. In *International conference on the theory and applications of cryptographic techniques*, pages 223–238. Springer, 1999.
29. Chris Peikert and Brent Waters. Lossy trapdoor functions and their applications. *SIAM Journal on Computing*, 40(6):1803–1844, 2011.
30. 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.
31. Victor Shoup. Using hash functions as a hedge against chosen ciphertext attack. In *International Conference on the Theory and Applications of Cryptographic Techniques*, pages 275–288. Springer, 2000.
32. Douglas Squirrel. Computing reciprocity symbols in number fields, 1997. *Undergraduate thesis, Reed College*.
33. Takashi Yamakawa, Shota Yamada, Goichiro Hanaoka, and Noboru Kunihiro. Adversary-dependent lossy trapdoor function from hardness of factoring semi-smooth rsa subgroup moduli. In *Annual International Cryptology Conference*, pages 3–32. Springer, 2016.