One-Wayness/KEM Equivalent to General Factoring *

Kaoru Kurosawa [1] Tsuyoshi Takagi [2]

[1] Ibaraki University,
4-12-1 Nakanarusawa, Hitachi, Ibaraki, 316-8511, Japan
kurosawa@cis.ibaraki.ac.jp

[2] Future University-Hakodate,
116-2 Kamedanakano-cho Hakodate Hokkaido, 041-8655, Japan
takagi@fun.ac.jp

Abstract

This paper shows the first practical semantically secure public-key encryption scheme such that its one-wayness is equivalent to *general* factoring in the *standard* model (in the sense of IND-CPA).

Next our proof technique is applied to Rabin-Parlier encryption scheme and a variant of RSA-Paillier encryption scheme to prove their exactly tight one-wayness.

We finally present the first KEM which is secure in the sense of IND-CCA under general factoring assumption in the random oracle model.

Keywords: one-wayness, factoring, semantic security, tight reduction, RSA-Paillier, Rabin-Paillier.

1 Introduction

1.1 Background

One-wayness and semantic-security are two important security notions of public-key encryption schemes. A public-key encryption scheme is called

*A preliminary version of this paper [21] was presented at ASIACRYPT 2003 and appeared in *Lecture Notes in Computer Science* **2894**, pp. 19–36, Springer-Verlag, 2003.

one-way if it is hard to find the plaintext m from a ciphertext c. It is called semantically-secure if c leaks no information on m in the computational sense. Especially, it is desirable to construct a semantically-secure encryption scheme under a reasonable assumption such that its one-wayness is equivalent to factoring $n = pq$.

Paillier showed an encryption scheme [28] such that it is one-way under Computational Composite Residuosity (CCR) assumption. It is semantically secure against chosen message attack (in the sense of IND-CPA) under Decisional Composite Residuosity (DCR) assumption.

RSA-Paillier encryption scheme given by [6] is one-way if RSA is one-way [7] [1] . It is secure in the sense of IND-CPA under RSA-Paillier assumption [6] [2] .

Rabin-Paillier encryption scheme is one-way if factoring Blum integers is hard [15], where $n(= pq)$ is called a Blum integer if $p = q = 3 \bmod 4$. It is secure in the sense of IND-CPA under Rabin-Paillier assumption [15] [3] .

However, no semantically-secure encryption scheme is known whose one-wayness is equivalent to factoring general $n = pq$ in the standard model.

On the other hand, we say that the security proof (reduction) is tight if the hardness of breaking the one-wayness and that of breaking the underlying computational problem are close. Otherwise, we say that the security proof is loose. Tight security proof is important because we would like to know how long n should be to protect the one-wayness. However, the previous security proofs on the one-wayness of RSA-Paillier and Rabin-Paillier encryption schemes are loose.

Remarks

- There are several provably secure constructions in the *random oracle* model, for example, OAEP+ [32], RSA-OAEP [14] and SAEP [3]. However, while the random oracle model is a useful heuristic, it does not rule out all possible attacks. Indeed, there exist cryptosystems that are secure in the random oracle model, but for which no secure implementation exists [4, 25, 1, 24, 5].

- The one-wayness of Okamoto-Uchiyama scheme [27] is equivalent to factoring $n = p^2q$, but not $n = pq$.

[1] This one-wayness problem was first raised by [30] and finally proved by [7] using LLL algorithm of lattice theory.

[2] In [6], the authors called the assumption Decisional Small e-Residues assumption.

[3] In [15], the authors called the assumption Decisional Small $2e$-Residues assumption.

- Cramer and Shoup showed the first practical encryption scheme which is semantically-secure against chosen ciphertext attacks (equivalently, secure in the sense of IND-CCA) under the decision Diffie-Hellman assumption [10]. They later showed a more general framework to construct IND-CCA schemes [11].

1.2 Our Contribution

This paper shows the first semantically-secure public-key encryption scheme such that its one-wayness is equivalent to *general* factoring of $n = pq$ in the *standard* model (in the sense of IND-CPA). The proposed scheme is based on the encryption scheme given by Kurosawa et al. [18, 19]. For comparison, see the following table on the one-wayness of Paillier, RSA-Paillier, Rabin-Paillier and the proposed scheme.

	One-wayness
Paillier	CCR assumption
RSA-Paillier	RSA assumption
Rabin-Paillier	factoring Blum integers
Proposed	factoring *general* integers

Also, our security proof is tight while the previous security proofs for RSA-Paillier and Rabin-Paillier encryption schemes are loose. This is because our proof is very simple and totally elemental except using Coppersmith algorithm [8]. The main differences from the previous proofs are:

1. Our proof technique makes only *one* decryption-oracle query while the previous proofs for RSA-Paillier/Rabin-Paillier encryption schemes make *two* oracle queries [7, 15].

2. No LLL algorithm is required, which was essentially used in the previous proofs for RSA-Paillier/Rabin-Paillier schemes [7, 15].

On semantic-security, the proposed scheme is secure in the sense of IND-CPA under a natural extension of RSA-Paillier assumption. In fact, we show a close relationship between our assumption and RSA-Paillier assumption.

We next apply our proof technique to Rabin-Paillier encryption scheme and a variant of RSA-Paillier encryption scheme to prove their exactly tight one-wayness (in the standard model):

- In Rabin-Paillier encryption scheme, suppose that the one-way is broken with probability ε. Then according to the previous security proof [15], Blum integers can be factored with probability $O(\varepsilon^2)$. On the other hand, we give a factoring algorithm with success probability just ε. This means that our security proof is exactly tight.

	[15]	Our proof
Factoring Blum integers	$O(\varepsilon^2)$	ε

- Suppose that the one-wayness of RSA-Paillier is broken with probability ε. Then RSA is not one-way with probability $O(\varepsilon^2)$ according to [7, 22]. Hence the security proof is loose.

  We now introduce RSA-Paillier+ encryption scheme. Suppose that the one-way is broken with probability ε. We then show that RSA with $e = 3$ is not one-way with probability ε.

	RSA-Paillier [7, 22]	RSA-Paillier+
Breaking RSA	$O(\varepsilon^2)$	ε

Finally, we consider hybrid encryption schemes. A hybrid encryption scheme uses public-key encryption techniques to derive a shared key that is then used to encrypt the actual messages using symmetric-key techniques. For hybrid encryption schemes, Cramer and Shoup formalized the notion of a key encapsulation mechanism (KEM), and an appropriate notion of security against adaptive chosen ciphertext attack [12]. A KEM works just like a public key encryption scheme, except that the encryption algorithm takes no input other than the recipient's public key. The encryption algorithm can only be used to generate and encrypt a key for a symmetric-key encryption scheme. (One can always use a public-key encryption scheme for this purpose. However, one can construct a KEM in other ways as well.) A secure KEM, combined with an appropriately secure symmetric-key encryption scheme, yields a hybrid encryption scheme which is secure in the sense of IND-CCA [12].

We can construct a CCA-secure KEM under the Blum integer factoring assumption from Rabin-SAEP [3] in the random oracle model. However, no KEM is known which is secure in the sense of IND-CCA under the general factoring assumption even in the random oracle model. For this problem, we show the first such KEM based on Kurosawa et al.'s encryption scheme

[18, 19] in the random oracle model. We should appreciate moving to the random oracle model when one is trying to achieve a goal that is difficult or impossible to achieve in the standard model.

1.3 Organization

This paper is organized as follows: In Section 2, we describe security notions discussed in this paper. In Section 3, the proposed scheme is presented. In Section 4, we prove that the tight one-wayness of the proposed scheme. In Section 5, we discuss its semantic security. In Section 6, the exactly tight security proof is given for the one-wayness of Rabin-Paillier encryption scheme. In Section 7, RSA-Paillier+ is introduced and the exactly tight one-wayness is proved. In Section 7.5, we discuss on the relationship between our results in the standard model and the previous results in the random oracle model. In Section 8, we present a KEM which is secure in the sense of IND-CCA under the general factoring assumption in the random oracle model. Section 9 includes some final comments.

2 Security of Encryption Schemes

PPT will denote a "probabilistic polynomial time". We say that a function $\mu(l)$ is negligible if $\mu(l)$ approaches to zero faster than the inverse of any polynomial in l, where l is the security parameter.

$|m|$ denotes the bit length of m if m is a string or a number. If $A(\cdot, \cdot, \cdots)$ is a probabilistic algorithm, then $x \stackrel{R}{\leftarrow} A(x_1, x_2, \cdots)$ denotes the experiment of running A on input $x_1, x_2, \cdots$ and letting x be the outcome. If S is a set, $x \stackrel{R}{\leftarrow} S$ denotes the experiment of choosing $x \in S$ at random.

2.1 Encryption Scheme

A public-key encryption scheme $\mathcal{PE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ consists of three algorithms.

- The key generation algorithm $\mathcal{K}$ outputs (pk, sk) on input 1^l, where pk is a public key, sk is the secret key and l is a security parameter. We write $(pk, sk) \stackrel{R}{\leftarrow} \mathcal{K}(1^\ell)$.

  The public key defines the space of plaintexts (messages) M and the space of ciphertexts C.

- The encryption algorithm $\mathcal{E}$ outputs a ciphertext c on input the public key pk and a plaintext (message) m; we write $c \stackrel{R}{\leftarrow} \mathcal{E}_{pk}(m)$.
- The decryption algorithm $\mathcal{D}$ outputs m on input the secret key sk and a ciphertext c (or a reserved symbol $\perp$ to denote that it has been given an invalid ciphertext c to decrypt); we write $m/\perp \leftarrow \mathcal{D}_{sk}(c)$.

We require that $\mathcal{D}_{sk}(\mathcal{E}_{pk}(m)) = m$ for each plaintext m. $\mathcal{K}$ and $\mathcal{E}$ are PPT algorithms, and $\mathcal{D}$ is a polynomial time algorithm.

2.2 One-Wayness

The one-wayness problem is as follows: given a public key pk and a ciphertext c, find the plaintext m such that $c \stackrel{R}{\leftarrow} \mathcal{E}_{pk}(m)$. Formally, for an adversary A, consider an experiment as follows.

$$(pk, sk) \stackrel{R}{\leftarrow} \mathcal{K},\ c \stackrel{R}{\leftarrow} \mathcal{E}_{pk}(m), \tilde{m} \stackrel{R}{\leftarrow} A(pk, c).$$

where m is chosen uniformly at random from the message space defined by pk. Let

$$Adv^{ow}_{\mathcal{PE}}(A) = \Pr(\tilde{m} = m).$$

For any $t > 0$, define

$$Adv^{ow}_{\mathcal{PE}}(t) = \max_{A} Adv^{ow}_{\mathcal{PE}}(A),$$

where the maximum is over all A who run in time t.

Definition 2.1 *We say that $\mathcal{PE}$ is (t, ε)-one-way if $Adv^{ow}_{\mathcal{PE}}(t) < \varepsilon$. We also say that $\mathcal{PE}$ is one-way if $Adv^{ow}_{\mathcal{PE}}(A)$ is negligible for any PPT adversary A on the security parameter ℓ.*

2.3 Semantic Security

Roughly speaking, we say that a public-key encryption scheme $\mathcal{PE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ is semantically secure against chosen plaintext attacks (SS-CPA) if it is hard to find any (partial) information on m from c [17]. This notion is equivalent to indistinguishability [17], which is denoted by IND-CPA.

Now IND-CPA is formalized as follows [2, 16]. Consider the model of an adversary $B = (B_1, B_2)$ as follows.

1. In the "find" stage, a challenger runs B_1 on input a public key pk. B_1 then outputs $(m_0, m_1, state)$, where m_0 and m_1 are two equal length plaintexts and $state$ is some state information.

2. In the "guess" stage, the challenger chooses a random bit b and computes a challenge ciphertext $c \stackrel{R}{\leftarrow} \mathcal{E}_{pk}(m_b)$. He then runs B_2 on input $(c, state)$. B_2 finally outputs a bit $\tilde{b}$.

We say that $\mathcal{PE}$ is secure in the sense of IND-CPA if $|\Pr(\tilde{b} = b) - 1/2|$ is negligible. Formally, let

$$(pk, sk) \stackrel{R}{\leftarrow} \mathcal{K},\ (m_0, m_1, state) \stackrel{R}{\leftarrow} B_1(pk), c \stackrel{R}{\leftarrow} \mathcal{E}_{pk}(m_b),\ \tilde{b} \stackrel{R}{\leftarrow} B_2(c, state).$$

Definition 2.2 *We say that $\mathcal{PE}$ is secure in the sense of indistinguishability against chosen-plaintext attack (IND-CPA) if*

$$\mathtt{Adv}^{ind}_{\mathcal{PE}}(B) \stackrel{\triangle}{=} |\Pr(\tilde{b} = b) - 1/2|$$

is negligible for any PPT adversary B on the security parameter ℓ.

If an adversary $B = (B_1, B_2)$ is allowed to access the decryption oracle $\mathcal{D}_{sk}(\cdot)$, we denote it by $B^{\mathcal{D}} = (B_1^{\mathcal{D}}, B_2^{\mathcal{D}})$. If $\mathtt{Adv}^{ind}_{\mathcal{PE}}(B^{\mathcal{D}})$ is negligible for any PPT adversary $B^{\mathcal{D}}$, we say that $\mathcal{PE}$ is secure in the sense of indistinguishability against adaptive chosen-ciphertext attack (IND-CCA).

2.4 Factoring Assumptions

The *general* factoring problem is to factor $n = pq$, where p and q are two primes such that $|p| = |q|$, where $|a|$ stands for the bit-length of integer a.

For an factoring algorithm B, consider the following experiment. Generate two primes p and q such that $|p| = |q|$ randomly. Run B on input $n = pq$. We say that B wins if B can output p or q.

Definition 2.3 *We say that the general factoring problem is (t, ε)-hard if $\Pr(B\ wins) < \varepsilon$ for any B who runs in time t.*

We also say that the general factoring problem is hard if $\Pr(B\ wins)$ is negligible for any PPT algorithm B on the security parameter ℓ, where $\ell = |p| = |q|$. The general factoring assumption claims that this problem is hard.

A *Blum* integer is $n(= pq)$ such that $p = q = 3 \bmod 4$, where p and q are prime numbers with $|p| = |q|$. For an factoring algorithm B, consider the following experiment. Generate a Blum integer $n = pq$ randomly. Run B on input $n = pq$. We say that B wins if B can output p or q.

Definition 2.4 *We say that the Blum-factoring problem is (t, ε)-hard if $\Pr(B \ wins) < \varepsilon$ for any B who runs in time t.*

We also say that the Blum-factoring problem is hard if $\Pr(B \ wins)$ is negligible for any PPT algorithm B. The Blum-factoring assumption claims that this problem is hard.

3 New Encryption Scheme

In this section, we propose a semantically secure (IND-CPA) encryption scheme such that its one-wayness is as hard as the *general* factoring problem. The proposed scheme is based on the encryption scheme given by Kurosawa et al. [18]. (Also, see [19].)

3.1 Kurosawa et al.'s Encryption Scheme

Kurosawa et al.'s showed an encryption scheme as follows [18].

(Secret key) Two prime numbers p and q such that $|p| = |q|$.

(Public key) $n(= pq)$ and α such that

$$(\alpha/p) = (\alpha/q) = -1, \tag{1}$$

where (α/p) denotes Legendre's symbol.

(Plaintext) $m \in Z_n^*$.

(Ciphertext) $c = (E, s, t)$ such that

$$E = m + \frac{\alpha}{m} \bmod n \tag{2}$$

$$s = \begin{cases} 0 & \text{if } (m/n) = 1; \\ 1 & \text{if } (m/n) = -1, \end{cases} \qquad t = \begin{cases} 0 & \text{if } (\alpha/m \bmod n) > m; \\ 1 & \text{if } (\alpha/m \bmod n) < m. \end{cases}$$

(Decryption) From eq.(2), it holds that

$$m^2 - Em + \alpha = 0 \bmod n. \tag{3}$$

The above equation has four roots. We can, however, decrypt m uniquely from (s, t) due to eq.(1). (See Appendix B.)

In [18], it is proved that this encryption scheme is one-way under the general factoring assumption. However, it is not secure in the sense of IND-CPA.

3.2 Proposed Encryption Scheme

We now show an encryption scheme which is secure in the sense of IND-CPA and its one-wayness is equivalent to the general factoring problem.

(Secret key) Two prime numbers p and q such that $|p| = |q|$.

(Public key) $n(= pq), e, \alpha$, where e is a prime such that $|n|/2 < |e| < |n|$ and $\alpha \in Z_n^*$ satisfies

$$(\alpha/p) = (\alpha/q) = -1. \tag{4}$$

(Plaintext) $m \in Z_n$.

(Ciphertext)

$$c = \left(r + \frac{\alpha}{r}\right)^e + mn \bmod n^2, \tag{5}$$

where $r \in Z_n^*$ is a random element such that $(r/n) = 1$ and $(\alpha/r \bmod n) > r$.

(Decryption) Compute $E = c^d \bmod n$, where $ed = 1 \bmod lcm(p-1, q-1)$. Then it is easy to see that

$$E = r + \frac{\alpha}{r} \bmod n.$$

Note that $(E, 0, 0)$ is the ciphertext of r of Kurosawa et al.'s encryption scheme. Therefore we can find r by decrypting $(E, 0, 0)$ with the decryption algorithm. Finally, by substituting r into eq.(5), we can obtain m.

Remark 1 *In RSA-Paillier encryption scheme,*

$$c = r^e(1 + mn) \bmod n^2,$$

where $m \in Z_n$ and $r \in Z_n$ [6].

3.3 How to Speed-Up Encryption

We need to compute $1/r \bmod n^2$ in our encryption algorithm. In this subsection, we show that it can be computed faster than computing it directly.

Lemma 3.1 *Let* $D_0 = 1/r \bmod n$. *Then*

$$1/r = D_0(2 - D_0 r) \bmod n^2.$$

Proof. We try to find D_1 such that $r^{-1} = D_0 + nD_1 \bmod n^2$. It is clear that $r(D_0 + nD_1) = 1 \bmod n^2$. On the other hand, $rD_0 = 1 + kn$ for some k because $rD_0 = 1 \bmod n$. Therefore, it holds that $1 + kn + nrD_1 = 1 \bmod n^2$. From this, we obtain that $D_1 = -kr^{-1} = -kD_0 \bmod n$. Therefore

$$r^{-1} = D_0 + n(-kD_0) = D_0 + D_0(1 - rD_0) = 2D_0 - D_0^2 r \bmod n^2.$$

□

Lemma 3.1 shows that we have only to compute $1/r \bmod n$ (not over mod n^2) and two multiplications over Z_{n^2} to obtain $r^{-1} \bmod n^2$. This method is faster than the direct computation. Consequently, a ciphertext c is computed as follows:

$$\begin{aligned} D_0 &= r^{-1} \bmod n, \\ c &= (r + \alpha D_0(2 - D_0 r))^e + mn \bmod n^2. \end{aligned}$$

4 One-Wayness of the Proposed Scheme

In this section, we prove that the one-wayness of the proposed scheme is equivalent to general factoring. Our security proof is tight in the sense that there is almost no gap between the one-wayness and the hardness of the general factoring problem.

4.1 Idea

Our proof is very simple and totally elemental except using Coppersmith algorithm [8]. In particular, our reduction algorithm makes only one query to the decryption oracle, and no LLL algorithm is required. They are the main differences from the previous proofs for RSA-Paillier/Rabin-Paillier encryption schemes [7, 15] which are loose.

To illustrate the underlying idea, consider a variant of our scheme such that $e = 1$ in eq.(5). That is,

$$c = r + \frac{\alpha}{r} + mn \bmod n^2.$$

We show that this scheme is one-way under the general factoring assumption. Suppose that it is not one-way. Then we can factor n as follows. For simplicity, we assume that α satisfies eq.(4).

1. We first choose a random $\bar{r} \in Z_n^*$ such that $(\bar{r}/n) = -1$ and compute
$$x = \bar{r} + \frac{\alpha}{\bar{r}} \bmod n^2.$$
2. Next for a randomly chosen $\tilde{m} \in Z_n^*$, let $c = x + \tilde{m}n \bmod n^2$.
3. Since this scheme is not one-way, we can find m from c such that
$$c = r + \frac{\alpha}{r} + mn \bmod n^2,$$
where $(r/n) = 1$.
4. Note that the above r is a solution of the following quadratic equation.
$$r^2 - (c - mn)r + \alpha = 0 \bmod n^2 \tag{6}$$
We can find such r by applying Coppersmith's algorithm [8] to eq.(6) because $r < n$.
5. Now we have obtained r and $\bar{r}$ such that
$$\bar{r} + \frac{\alpha}{\bar{r}} = x = c = r + \frac{\alpha}{r} \bmod n.$$
6. It is shown that we can factor n by computing $\gcd(r - \bar{r}, n)$.

Suppose that the one-wayness is broken with probability ε. Then we can factor n with the same probability because we make only one decryption query at step 3. Hence our proof is tight.

In what follows, it will be shown that the same approach can be applied for $1 < e < n$.

Remark Unfortunately, the above scheme is not IND-CPA because we can distinguish the ciphertexts for $m = m_0$ from those for $m = m_1$. Find r by solving eq.(6) by assuming that $m = m_0$. By recomputing c from (r, m_0), we can check if $m = m_0$ or not. To avoid this problem, $r + \alpha/r$ is raised by e in eq.(5).

4.2 Proof of One-Wayness

We say that

1. $r \in Z_n^*$ is *principal* if $(r/n) = 1$ and $(\alpha/r \bmod n) > r$.
2. $\bar{r} \in Z_n^*$ is *conjugate* if $(\bar{r}/n) = -1$.

Note that in terms of the parameters of Kurosawa et al's encryption scheme, $r \in Z_n^*$ is *principal* if $(s,t) = (0,0)$ and $\bar{r} \in Z_n^*$ is *conjugate* if $s = 1$.

Lemma 4.1 *For any conjugate $\bar{r}$, there exists a unique principal r such that*

$$E \stackrel{\triangle}{=} \bar{r} + \frac{\alpha}{\bar{r}} = r + \frac{\alpha}{r} \bmod n. \tag{7}$$

Further, $\gcd(r - \bar{r}, n) = p$ *or* q.

Proof. In Kurosawa et al's encryption scheme, E has four roots corresponding to $(s,t) = (0,0), (0,1), (1,0), (1,1)$ as shown in Appendix B. Hence, the former part of this Lemma holds.

Further $(r/n) = 1$ and $(\bar{r}/n) = -1$. Therefore, we can see that $\gcd(r - \bar{r}, n) = p$ or q from Appendix B. □

Lemma 4.2 Suppose that there exists an algorithm M_0 which on input (n, e, α) and $u = (r + \alpha/r)^e \bmod n$ with principal r, outputs

$$w = (r + \alpha/r)^e \bmod n^2$$

with probability ϵ in time t. Then there exists an algorithm M_1 which can factor n with probability $\epsilon/2$ in time $t + \mathcal{O}((\log n)^3)$.

Proof. On input n, M_1 first chooses a prime e such that $|n|/2 < |e| < |n|$. M_1 also chooses $\alpha \in Z_n^*$ such that $(\alpha/n) = 1$ randomly. It is easy to see that α satisfies eq.(4) with probability 1/2.

M_1 next chooses a conjugate $\bar{r} \in Z_n^*$ randomly and computes $u = (\bar{r} + \alpha/\bar{r})^e \bmod n$. From Lemma 4.1, we can see that $u = (r + \alpha/r)^e \bmod n$ for some principal $r \in QR_n$.

M_1 next runs M_0 on input (n, e, α, u). M_0 then outputs $w = (r + \alpha/r)^e \bmod n^2$ with probability ϵ in time t.

Now since

$$(x \stackrel{\triangle}{=})\bar{r} + \frac{\alpha}{\bar{r}} = r + \frac{\alpha}{r} \bmod n,$$

it holds that

$$r + \frac{\alpha}{r} = x + yn \bmod n^2 \tag{8}$$

for some $y \in Z_n$. We then obtain that

$$w = (r + \alpha/r)^e = (x + yn)^e = x^e + eynx^{e-1} \bmod n^2.$$

It is easy to see that

$$eyx^{e-1} = \frac{w - x^e}{n} \bmod n.$$

Therefore y is obtained as

$$y = \frac{w - x^e}{n}(ex^{e-1})^{-1} \bmod n.$$

Substitute y into eq.(8) and let $v = x + yn \bmod n^2$. Then we obtain that

$$r^2 - vr + \alpha = 0 \bmod n^2.$$

We can solve this quadratic equation in time $O((\log n)^3)$ by using the Coppersmith's algorithm [8] because of $0 < r < n$. Then we can factor n from $(\bar{r}, r)$ by using Lemma 4.1. The success probability of M_1 is $\epsilon/2$. □

Theorem 4.1 *The proposed encryption scheme is (t, ε) one-way if the general factoring problem is $(t', \varepsilon/2)$-hard, where $t' = t + poly(\log n)$.*

Proof. Suppose that there exists a PPT algorithm A that breaks the one-wayness of the proposed scheme with probability ε in time t. We will show an algorithm M_0 which computes $w = (r + \alpha/r)^e \bmod n^2$ from (n, e, α) and $u = (r + \alpha/r)^e \bmod n$ with probability ϵ in time t.

On input (n, e, α, u), M_0 chooses a plaintext $m \in Z_n$ randomly, and computes a ciphertext

$$c = u + mn \bmod n^2.$$

M_0 next runs A on input c and the public key (n, e, α, u). A then outputs a plaintext m with probability ε in time t. Then we have

$$w = c - mn = (r + \alpha/r)^e \bmod n^2.$$

Consequently, this Theorem holds from Lemma 4.2.

□

The total factoring algorithm for the proposed encryption scheme is summarized as follows:

OW_Reciprocal_Paillier

Input: n.
Output: p, q factoring of n

1. choose a prime e such that $|n|/2 < |e| < |n|$.
   Also choose $\alpha \in Z_n^*$ such that $(\alpha/n) = 1$ randomly.
2. choose a random $\bar{r} \in Z_n^*$ such that $(\bar{r}/n) = -1$.
3. compute $x = \bar{r} + \alpha/\bar{r} \bmod n^2$.
4. choose a random (fake) plaintext $\bar{m} \in Z_n^*$.
5. compute a ciphertext $c = x^e + \bar{m}n \bmod n^2$.
6. obtain a valid plaintext $m = A((n, e, \alpha), c)$, where A is the one-wayness adversary.
7. compute $w = c - mn = (r + \alpha/r)^e \bmod n^2$.
8. compute $u = (w - x^e)/n$.
9. compute $y = u(ex^{(e-1)})^{-1} \bmod n$.
10. compute $v = (\bar{r} + \alpha/\bar{r}) + ny \bmod n$.
11. solve $r^2 - vr + \alpha = 0 \bmod n^2$ using Coppersmith's algorithm [8].
12. return $gcd(\bar{r} - r, n)$.

4.3 Hensel Lifting and Large Message Space

Catalano et al. proved that Hensel-RSA problem is as hard as breaking RSA for any lifting index l [7, 22]. The Hensel-RSA problem with lifting index l is to compute $r^e \bmod n^l$ from $(n, e, r^e \bmod n)$ and l.

In this subsection, we define Hensel-Reciprocal problem and show that it is as hard as the general factoring problem for any lifting index l. This result implies that we can enlarge the message space of the proposed encryption scheme so that $m \in Z_{n^{l-1}}$.

The Hensel-Reciprocal problem is to compute

$$Y = \left(r + \frac{\alpha}{r}\right)^e \bmod n^l$$

from (n, e, α, y) and l, where

$$y = \left(r + \frac{\alpha}{r}\right)^e \bmod n$$

and $r \in Z_n^*$ is principal.

Theorem 4.2 *The Hensel-Reciprocal problem is as hard as the general factoring problem for any lifting index $l \geq 2$.*

Proof. It is easy to see that we can solve the Hensel-Reciprocal problem if we can factor n. We will prove the converse.

Suppose that there exists a PPT algorithm which can solve the Hensel-Reciprocal problem with probability ε for some $l \geq 2$. That is, the PPT algorithm can compute

$$Y = \left(r + \frac{\alpha}{r}\right)^e \bmod n^l$$

from (n, e, α, y). Then we obtain that

$$Y' = \left(r + \frac{\alpha}{r}\right)^e \bmod n^2.$$

Now Lemma 4.2, we can factor n with probability $\varepsilon/2$ in polynomial time. □

5 Semantic Security of the Proposed Scheme

In this section, we prove that the proposed encryption scheme is secure in the sense of IND-CPA under a natural extension of RSA-Paillier assumption which we call Reciprocal-Paillier assumption. We also show a close relationship between our assumption and RSA-Paillier assumption.

Definition 5.1 *Let S be a countable index set. An ensemble indexed by S is a sequence of finite sets indexed by S. Namely, any $X = \{X_w\}_{w \in S}$, where each X_w is a finite set, is an ensemble indexed by S.*

Definition 5.2 *Two ensembles, $X = \{X_w\}_{w \in S}$ and $Y = \{Y_w\}_{w \in S}$, are indistinguishable if for any PPT algorithm (called a distinguisher) D, every positive polynomial $p(\cdot)$, and all sufficiently long $w \in S$,*

$$|\Pr[D(x, w) = 1 \mid x \xleftarrow{R} X_w] - \Pr[D(x, w) = 1 \mid x \xleftarrow{R} Y_w]| < \frac{1}{p(|w|)}$$

5.1 Semantic security

It is known that RSA-Paillier encryption scheme is secure in the sense of IND-CPA if $SMALL_{RSAP}(n,e)$ and $LARGE_{RSAP}(n,e)$ are indistinguishable [6], where (n,e) is a public key of RSA and

$$\begin{aligned} SMALL_{RSAP}(n,e) &\stackrel{\triangle}{=} \{x \mid x = r^e \bmod n^2, r \in Z_n^*\} \\ LARGE_{RSAP}(n,e) &\stackrel{\triangle}{=} \{x \mid x = r^e \bmod n^2, r \in Z_{n^2}^*\} \end{aligned}$$

We call this indistinguishability **RSA-Paillier assumption**.

We next extend it to **Reciprocal-Paillier assumption** which claims that $SMALL_{RSAK}(n,e,\alpha)$ and $LARGE_{RSAK}(n,e,\alpha)$ are indistinguishable, where (n,e,α) is a public key of the proposed encryption scheme and

$$\begin{aligned} SMALL_{RSAK}(n,e,\alpha) &\stackrel{\triangle}{=} \{x \mid x = \left(r + \frac{\alpha}{r}\right)^e \bmod n^2, r \in Z_n^* \text{ is principal}\} \\ LARGE_{RSAK}(n,e,\alpha) &\stackrel{\triangle}{=} \{x \mid x = \left(r + \frac{\alpha}{r}\right)^e \bmod n^2, r \in Z_{n^2}^*\}. \end{aligned}$$

Then we prove the following theorem.

Theorem 5.1 *The proposed encryption scheme is secure in the sense of IND-CPA if and only if Reciprocal-Paillier assumption holds.*

A proof will be given in the next subsection.

5.2 Proof of Theorem 5.1

Let (n,e,α) be a public key of the proposed encryption scheme. Define

$$\begin{aligned} ZERO(n,e,\alpha) &\stackrel{\triangle}{=} \{\left(r + \frac{\alpha}{r}\right)^e \bmod n^2 \mid r \in Z_n^* \text{ is principal}\} \\ ALL(n,e,\alpha) &\stackrel{\triangle}{=} \{\left(r + \frac{\alpha}{r}\right)^e + mn \bmod n^2 \mid m \in Z_n \text{ and } r \in Z_n^* \text{ is principal}\}. \end{aligned}$$

Note that $ZERO(n,e,\alpha)$ is the set of ciphertexts for $m = 0$ and $ALL(n,e,\alpha)$ is the set of ciphertexts for all $m \in Z_n$.

We first show that the proposed encryption scheme is secure in the sense of IND-CPA if and only if $ZERO(n,e,\alpha)$ and $ALL(n,e,\alpha)$ are indistinguishable. We next prove that $ALL(n,e,\alpha) = LARGE_{RSAK}(n,e,\alpha)$. Then this implies Theorem 5.1 because $ZERO(n,e,\alpha) = SMALL_{RSAK}(n,e,\alpha)$.

Lemma 5.1 *The proposed encryption scheme is secure in the sense of IND-CPA if and only if* $ZERO(n, e, \alpha)$ *and* $ALL(n, e, \alpha)$ *are indistinguishable.*

Proof. Suppose that there exists an adversary $B = (B_1, B_2)$ which breaks our encryption scheme in the sense of IND-CPA. We will show a distinguisher D for $ZERO(n, e, \alpha)$ and $ALL(n, e, \alpha)$.

Let $(x, (n, e, \alpha))$ be the input to D. Then D works as follows.

1. In the find stage, D runs B_1 on input $pk = (n, e, \alpha)$. B_1 then outputs $(m_0, m_1, state)$.

2. In the guess stage, D chooses a bit b randomly and computes

$$c = x + m_b n \bmod n^2.$$

   D then runs B_2 on input $(c_b, state)$. B_2 finally outputs a bit $\tilde{b}$.

3. D outputs "1" if $\tilde{b} = b$. Otherwise, D outputs "0".

We consider two experiments.

(Experiment 0) x is randomly chosen from $ZERO(n, e, \alpha)$. In this case, c is a valid ciphertext of m_b. Therefore,

$$p_0 \stackrel{\triangle}{=} \Pr(D = 1) = Pr(\tilde{b} = b) = 1/2 + \epsilon \text{ or } 1/2 - \epsilon$$

for non-negligible ϵ from our assumption

(Experiment 1) x is randomly chosen from $ALL(n, e, \alpha)$. In this case, c is uniformly distributed over $ALL(n, e, \alpha)$. Therefore, it is clear that

$$p_1 = \Pr(D = 1) = Pr(\tilde{b} = b) = 1/2.$$

Hence $|p_0 - p_1| = \epsilon$. This means that D can distinguish between the two sets, $ZERO(n, e, \alpha)$ and $ALL(n, e, \alpha)$.

We next prove the converse. Suppose that there exists a distinguisher D for $ZERO(n, e, \alpha)$ and $ALL(n, e, \alpha)$. We will show an adversary $B = (B_1, B_2)$ which breaks our encryption scheme in the sense of IND-CPA. B works as follows.

1. In the find stage, on input $pk = (n, e, \alpha)$, B_1 outputs $m_0 = 0$ and $m_1 \in Z_n$, where m_1 is randomly chosen from Z_n.

2. In the guess stage, B_2 is given a challenge ciphertext c_b, where b is a random bit and c_b is a ciphertext of m_b. B_2 runs D on input (n, e, α, c_b). Finally, B_2 outputs whatever D does.

Note that c_0 is randomly chosen from $ZERO(n, e, \alpha)$ and c_1 is randomly chosen from $ALL(n, e, \alpha)$. Therefore, D can distinguish them from our assumption. Hence B_2 can distinguish them. □

Lemma 5.2 $LARGE_{RSAK}(n, e, \alpha) \subseteq ALL(n, e, \alpha)$.

Proof. Suppose that $(n, e, \alpha, c) \in LARGE_{RSAK}(n, e, \alpha)$. Then

$$c = \left(r + \frac{\alpha}{r}\right)^e \bmod n^2$$

for some $r \in Z^*_{n^2}$. Decrypt c by our decryption algorithm. Then we can find $m \in Z_n$ and a principal $r' \in Z^*_n$ such that

$$c = \left(r' + \frac{\alpha}{r'}\right)^e + mn \bmod n^2.$$

Therefore $(n, e, \alpha, c) \in ALL(n, e, \alpha)$. This means that

$$LARGE_{RSAK}(n, e, \alpha) \subseteq ALL(n, e, \alpha).$$

□

Lemma 5.3 $ALL(n, e, \alpha) \subseteq LARGE_{RSAK}(n, e, \alpha)$.

Proof. Suppose that $(n, e, \alpha, c) \in ALL(n, e, \alpha)$. Then

$$c = \left(r + \frac{\alpha}{r}\right)^e + mn \bmod n^2 \tag{9}$$

for some $m \in Z_n$ and a principal $r \in Z^*_n$. We want to find $u \in Z^*_{n^2}$ such that $c = (u + \frac{\alpha}{u})^e \bmod n^2$. For this purpose, we will compute $u_p = u \bmod p^2$ and $u_q = u \bmod q^2$.

Consider d such that $ed = 1 \bmod \phi(n)n$. Then from eq.(9), it is easy to see that

$$c^d = r + \frac{\alpha}{r} \bmod n.$$

In the following we want to find $u \in Z_{n^2}$ s.t. $c^d = u + a/u \bmod n^2$. At first we compute u modulo the primes p^2 and q^2.

From equation

$$c^d = r + \frac{\alpha}{r} \bmod p,$$

we have

$$r^2 - c^d r + \alpha = 0 \bmod p. \tag{10}$$

We claim that

$$2r - c^d \neq 0 \bmod p. \tag{11}$$

On the contrary, suppose that $2r - c^d = 0 \bmod p$. Then we obtain that

$$c^d = 2r = r + \frac{\alpha}{r} \bmod p.$$

Therefore,

$$r = \frac{\alpha}{r} \bmod p.$$

Then $r^2 = \alpha \bmod p$. Hence

$$\left(\frac{\alpha}{p}\right) = \left(\frac{r^2}{p}\right) = 1.$$

However, this contradicts to eq.(4).

From eq.(10) and eq.(11), we see that there exists $y_p \in Z_n$ such that

$$(r^2 - c^d r + \alpha) + p y_p (2r - c^d) \quad = \quad 0 \bmod p^2.$$

For this y_p, define

$$u_p \quad = \quad r + p y_p \bmod p^2.$$

Then it is easy to see that

$$u_p^2 - c^d u_p + \alpha = 0 \bmod p^2.$$

Similarly, there exists y_q such that

$$(r^2 - c^d r + \alpha) + q y_q (2r - c^d) \quad = \quad 0 \bmod q^2.$$

For this y_q, define

$$u_q \quad = \quad r + q y_q \bmod q^2.$$

Then

$$u_q^2 - c^d u_q + \alpha = 0 \bmod p^2.$$

Now consider u such that

$$u = u_p \bmod p^2,\ u = u_q \bmod q^2.$$

Then u satisfies

$$u^2 - c^d u + \alpha = 0 \bmod n^2.$$

Hence

$$c^d = u + \frac{\alpha}{u} \bmod n^2.$$

$$c = \left(u + \frac{\alpha}{u}\right)^e \bmod n^2$$

This means that $c \in LARGE_{RSAK}(n, e, \alpha)$. Hence

$$ALL(n, e, \alpha) \subseteq LARGE_{RSAK}(n, e, \alpha).$$

□

Now we are ready to prove Theorem 5.1. From Lemma 5.2 and Lemma 5.3, we obtain that

$$ALL(n, e, \alpha) = LARGE_{RSAK}(n, e, \alpha).$$

Further,

$$ZERO(n, e, \alpha) = SMALL_{RSAK}(n, e, \alpha)$$

from their definitions. Therefore, from Lemma 5.1, we see that the proposed encryption scheme is IND-CPA if and only if Reciprocal-Paillier assumption holds.

5.3 Relationship with RSA-Paillier Assumption

We investigate the relationship between RSA-Paillier assumption and Reciprocal-Paillier assumption. We first slightly modify RSA-Paillier assumption as follows.

Modified RSA-Paillier assumption:
$SMALL'_{RSAP}(n, e, \alpha)$ and $LARGE'_{RSAP}(n, e, \alpha)$ are indistinguishable, where (n, e, α) is a public key of the proposed encryption scheme and

$$\begin{aligned} SMALL'_{RSAP}(n, e, \alpha) &\stackrel{\triangle}{=} \{x \mid x = r^e \bmod n^2, r \in Z_n^*\} \\ LARGE'_{RSAP}(n, e, \alpha) &\stackrel{\triangle}{=} \{x \mid x = r^e \bmod n^2, r \in Z_{n^2}^*\}. \end{aligned}$$

The difference from RSA-Paillier assumption is that α is added to (n, e). However, the two assumptions are different because it is hard to find α satisfying eq.(4) from (n, e).

SMALL assumption:
$SMALL_{RSAK}(n, e, \alpha)$ and $SMALL'_{RSAP}(n, e, \alpha)$ are indistinguishable.

Then we have the relationship among assumptions as follows.

$$SMALL_{RSAK} \overset{(a)}{\approx} SMALL'_{RSAP} \overset{(b)}{\approx} LARGE'_{RSAP} \overset{(c)}{\approx} LARGE_{RSAK}, \quad (12)$$

where $\approx$ means indistinguishable. (a) is claimed by SMALL assumption. (b) is claimed by modified RSA-Paillier assumption.

If we assume (c) as well, then we see that the proposed encryption scheme is secure in the sense of IND-CPA from corollary of Theorem 5.1. We now prove that even if we do not assume (c), the proposed encryption scheme is secure in the sense of IND-CPA.

Corollary 5.1 *The proposed encryption scheme is secure in the sense of IND-CPA under modified RSA-Paillier assumption and SMALL assumption.*

Proof. From corollary of Theorem 5.1 and eq.(12), we have only to prove that $LARGE_{RSAK}(n, e, \alpha)$ and $LARGE'_{RSAP}(n, e, \alpha)$ are indistinguishable.

Suppose that there exists a distinguisher $\mathcal{A}$ for $LARGE_{RSAK}(n, e, \alpha)$ and $LARGE'_{RSAP}(n, e, \alpha)$. We construct a distinguisher D for $SMALL_{RSAK}(n, e, \alpha)$ and $SMALL'_{RSAP}(n, e, \alpha)$.

Let $X = (n, e, \alpha, c)$ be the input to D. Then D behaves as follows.

1. D chooses a random $m \in Z_n$, and computes $c' = c + mn \bmod n^2$.
2. Then D runs $\mathcal{A}$ on input $Y = (n, e, \alpha, c')$.
3. D outputs whatever $\mathcal{A}$ does.

It is easy to see that

- If X is randomly chosen from $SMALL_{RSAK}(n, e, \alpha)$, then Y is uniformly distributed over $LARGE_{RSAK}(n, e, \alpha)$.
- If X is randomly chosen from $SMALL'_{RSAP}(n, e, \alpha)$, then Y is uniformly distributed over $LARGE'_{RSAP}(n, e, \alpha)$.

Therefore, D can distinguish between $SMALL_{RSAK}(n, e, \alpha)$ and $SMALL'_{RSAP}(n, e, \alpha)$ if $\mathcal{A}$ can distinguish between $LARGE_{RSAK}(n, e, \alpha)$ and $LARGE'_{RSAP}(n, e, \alpha)$. However, this is against our assumption of this corollary. □

6 Exact One-Wayness of Rabin-Paillier Scheme

In this section, we show a tight security proof for the one-wyaness of Rabin-Paillier encryption scheme.

Suppose that the one-way is broken with probability ε. Then accoring to the previous proof [15], Blum integers can be factored with probability $O(\varepsilon^2)$. On the other hand, we give a factoring algorithm with success probability just ε.

	[15]	Our proof
Facoring Blum integers	$O(\varepsilon^2)$	ε

6.1 Rabin-Paillier Encryption Scheme

Let

$$QR_n \stackrel{\triangle}{=} \{r^2 \bmod n^2 \mid r \in Z_n^*\}.$$

We say that $\bar{r} \in Z_n^*$ is *conjugate* if $(\bar{r}/n) = -1$, where (m/n) denotes Jacobi's symbol.

Then Rabin-Paillier encryption scheme is described as follows.

(Secret key) Two prime numbers p and q such that $|p| = |q|$ and $p = q = 3 \bmod 4$.

(Public key) $n(= pq), e$, where e is a prime such that $|n|/2 < |e| < |n|$.

(Plaintext) $m \in Z_n$.

(Ciphertext)

$$c = r^{2e} + mn \bmod n^2, \tag{13}$$

where $r \in QR_n$ is randomly chosen.

(Decryption) Let $E = c^d \bmod n$, where $ed = 1 \bmod lcm(p-1, q-1)$. Then it is easy to see that

$$E = r^2 \bmod n.$$

We can find r such that $r \in QR_n$ uniquely because $p = q = 3 \bmod 4$. Finally, by substituting r into eq.(13), we can obtain m.

In [15]. the authors showed that Rabin-Paillier encryption scheme is secure in the sense of IND-CPA if $(n, e, \mathcal{E}(n, e; 0))$ and (n, e, Q_{n^2}) are indistinguishable, where

$$\mathcal{E}(n, e; 0) \stackrel{\triangle}{=} \{r^{2e} \bmod n^2 \mid r \in QR_n\}.$$

We call this indistinguishability Rabin-Paillier assumption.

Remark 2 *1. In [15], the condition on e is that $\gcd(e, \lambda(n)) = 1$, where λ is Carmichael's function. However, for this condition, we cannot prove that the claimed one-wayness because we cannot choose such e for a given n.*

2. In Appendix A, we also point out a flaw on their claim for the semantic security of Rabin-Paillier cryptosystem.

6.2 How to Prove Tight One-Wayness

We illustrate the underlying idea of our tight security proof. Consider a variant of Rabin-Paillier encryption scheme such that $e = 1$. That is,

$$c = r^2 + mn \bmod n^2.$$

Suppose that it is not one-way. Then we can factor n as follows.

1. We first choose a conjugate $\bar{r}$ randomly and compute $x = \bar{r}^2 \bmod n^2$.
2. Next for a randomly chosen $\tilde{m} \in Z_n^*$, let $c = x + \tilde{m}n \bmod n^2$.
3. Since this scheme is not one-way, we can find m from c such that

   $$c = r^2 + mn \bmod n^2,$$

   where $r \in QR_n$.
4. Note that r is a solution of the following quadratic equation.

   $$r^2 = c - mn \bmod n^2.$$

   We can find such r by computing a square root of $(c - mn \bmod n^2)$ without modulus because $r < n$.

Scheme	Factoring Probability
Galindo et al. [15]	ε^2
Our Proposed Proof	ε

Table 1: Factoring probability using OW-oracle with probability ε

5. Now we have obtained r and $\bar{r}$ such that

$$\bar{r}^2 = x = c = r^2 \bmod n^2.$$

6. It is easy to see that we can factor n by computing $\gcd(r - \bar{r}, n)$.

Suppose that the one-wayness is broken with probability ε. Then we can factor n with the same probability because we make only one decryption query at step 3. Hence our proof is tight.

In what follows, it will be shown that the same approach can be applied for $1 < e < n$.

6.3 Proof of Tight One-Wayness

Lemma 6.1 *Let n be a Blum integer. For any conjugate $\bar{r}$, there exists a unique $r \in QR_n$ such that*

$$r^2 = \bar{r}^2 \bmod n. \tag{14}$$

Further, $\gcd(r - \bar{r}, n) = p$ *or* q.

Proof. Note that $(-1/p) = -1$ and $(-1/q) = -1$ for a Blum integer $n = pq$. A conjugate $\bar{r} \in Z_n^*$ satisfies $(\bar{r}/n) = -1$, namely $(I) : (\bar{r}/p) = 1 \wedge (\bar{r}/q) = -1$ or $(II) : (\bar{r}/p) = -1 \wedge (\bar{r}/q) = 1$. In the case of (I), define $r = \bar{r} \bmod p$ and $r = -\bar{r} \bmod q$, then the statement of the lemma is obtained. Similarly in the case of (II) we assign $r = -\bar{r} \bmod p$ and $r = \bar{r} \bmod q$.

Lemma 6.2 Let n be a Blum integer. Suppose that there exists an algorithm M_0 which on input (n, e) and $v = r^{2e} \bmod n$ with $r \in QR_n$, outputs $w = r^{2e} \bmod n^2$ with probability ϵ in time t. Then there exists an algorithm M_1 which can factor n with probability ϵ in time $t + \mathcal{O}((\log n)^3)$.

Proof. On input n, M_1 first chooses a prime e such that $|n|/2 < |e| < |n|$. It also chooses a conjugate $\bar{r} \in Z_n^*$ randomly and computes $v = \bar{r}^{2e} \bmod n$. From Lemma 6.1, we can see that $v = r^{2e} \bmod n$ for some $r \in QR_n$.

M_1 next runs M_0 on input (n, e, v). M_0 then outputs $w = r^{2e} \bmod n^2$ with probability ϵ in time t.

Now since $r^2 = \bar{r}^2 \bmod n$, r^2 is written as

$$r^2 = \bar{r}^2 + yn \tag{15}$$

for some $-n < y < n$. By letting $x = \bar{r}^2$, we obtain that

$$w = r^{2e} = (x + yn)^e = x^e + eynx^{e-1} \bmod n^2. \tag{16}$$

It is easy to see that

$$eyx^{e-1} = \left(\frac{w - x^e \bmod n^2}{n} \right) \bmod n.$$

Therefore

$$y = (ex^{e-1})^{-1} \left(\frac{w - x^e \bmod n^2}{n} \right) \bmod n.$$

Let

$$A = (ex^{e-1})^{-1} \left(\frac{w - x^e \bmod n^2}{n} \right) \bmod n.$$

Then y is obtained as

$$y = A \text{ or } A - n \tag{17}$$

because $-n < y < n$. Substitute eq.(17) into eq.(15). Then we can obtain $r > 0$ by computing a square root of $\bar{r}^2 + yn$ because $r^2 < n^2$. Finally we can factor n by using $(r, \bar{r})$ from Lemma 6.1. This completes the proof.

□

Theorem 6.1 Rabin-Paillier encryption scheme is (t, ε)-one-way if Blum factoring problem is (t', ε)-hard, where $t' = t + \mathcal{O}((\log n)^3)$.

Proof. Suppose that there exists a PPT algorithm A which breaks the one-wayness of Rabin-Paillier encryption scheme with probability ε in time t. We will show an algorithm M_0 which computes $w = r^{2e} \bmod n^2$ from (n, e) and $v = r^{2e} \bmod n$ with probability ϵ in time $t + \mathcal{O}((\log n)^3)$.

On input (n, e, v), M_0 chooses a plaintext $m \in Z_n$ randomly, and computes a ciphertext

$$c = v + mn \bmod n^2.$$

M_0 next runs A on input (n, e, c). A then outputs a plaintext m with probability ε in time t. Then we have

$$w = c - mn = r^{2e}.$$

Consequently, this Theorem holds from Lemma 6.2.

□

The total algorithm of A which breaks the one-wayness of Rabin-Paillier scheme is summarized as follows.

Exact_OW_Rabin_Paillier
Input: n.
Output: p, q factoring of n
0. chooses a prime e such that $\|n\|/2 < \|e\| < \|n\|$ randomly.
1. choose a random $\bar{r} \in Z_n^*$ such that $(\bar{r}/n) = -1$.
2. compute $x = \bar{r}^2 \bmod n^2$.
3. choose a random (fake) plaintext $\bar{m} \in Z_n^*$.
4. compute a ciphertext $c = x^e + \bar{m}n \bmod n^2$.
5. obtain a valid plaintext $m = A(n, e, c)$
6. compute $w = c - mn = r^{2e} \bmod n^2$.
7. compute $u = (w - x^e)/n$.
8. compute $y = u(ex^{(e-1)})^{-1} \bmod n$.
9. compute $v = \bar{r}^2 + ny$.
10. Find $r > 0$ such that $r^2 = v$ in Z.
11. return $gcd(\bar{r} - r, n)$.

7 RSA-Paillier+

In this section, we introduce RSA-Paillier+, a variant of RSA-Pailier encryption scheme, and prove its exactly tight one-wayness by extending our proof technique.

Suppose that the one-wayness of RSA-Paillier encryption scheme is broken with probability ε. Then according to [7, 22], RSA is not one-way with probability $\varepsilon' = O(\varepsilon^2)$. On the other hand, suppose that the one-wayness of RSA-Pailier+ is broken with probability ε. Then we show that RSA with $e = 3$ is not one-way with probability ε.

	RSA-Paillier [7, 22]	RSA-Paillier+
Breaking RSA	$O(\varepsilon^2)$	ε

7.1 RSA-Paillier+

RSA-Paillier+ encryption scheme is described as follows.

Key generation: Generate two prime numbers p and q such that $|p| = |q|$ and $3 \not| p-1, 3 \not| q-1$. Choose a prime s such that $|n|/2 < |s| < |n|$. Let $n = pq$ and $e = 3s$. Compute d such that

$$ed = 1 \bmod lcm(p-1, q-1).$$

The public-key is (n, e) and the secret-key is d.

Encryption: To encrypt a message $m \in Z_{n^2}$, choose $r \in Z_n$ randomly and compute

$$c = r^e + mn \bmod n^3. \tag{18}$$

The ciphertext is c.

Decryption: To decrypt c, first compute r by $r = c^d \bmod n$. Next by substituting r into eq.(18), we can obtain m.

The message space is Z_{n^2} and the space of ciphertexts is Z_{n^3}. Hence the bandwidth is 2/3, where the bandwidth is the message bit-length divided by the ciphertext bit-length. Note that the bandwidth of RSA-Paillier encryption scheme is 1/2.

It is easy to see that RSA-Paillier+ encryption scheme is secure in the sense of IND-CPA if Z_{n^3} and $\{x \mid x = r^e \bmod n^3, r \in Z_n\}$ are indistinguishable.

7.2 How to Prove Tight One-Wayness

We illustrate the underlying idea of our reduction. Consider a variant of RSA-Paillier+ such that $e = 3$. That is,

$$c = r^3 + mn \bmod n^3.$$

Suppose that it is not one-way. Then we can find r such that $x = r^3 \bmod n$ from $(n, e = 3, x)$ as follows.

1. For a randomly chosen $\tilde{m} \in Z_{n^2}$, let $c = x + \tilde{m}n \bmod n^3$.

2. Since this scheme is not one-way, we can find m from c such that

$$c = r^3 + mn \bmod n^3,$$

   where $r \in Z_n$.

3. Note that r is a solution of the following equation.

$$r^3 = c - mn \bmod n^3.$$

   We can find such r by computing a cubic root of $(c - mn \bmod n^3)$ without modulus because $r < n$.

4. Now we have obtained r such that

$$x = c = r^3 \bmod n.$$

Suppose that the one-wayness is broken with probability ε. Then we can break RSA of $e = 3$ with the same probability because we make only one decryption query at step 2. Hence our proof is tight.

In what follows, it will be shown that the same approach can be applied for $e = 3s$.

7.3 Proof of One-Wayness

We denote RSA encryption scheme with a public-key (n, e) by RSA(n, e), and RSA-Paillier+ encryption scheme with a public-key (n, e) by RSA-Paillier+(n, e). We then prove the tight equivalence between the one-wayness of RSA-Paillier+(n, e) and the one-wayness of RSA$(n, 3)$.

Theorem 7.1 *RSA-Paillier+(n, e) is (t, ε) one-way if RSA$(n, 3)$ is (t', ε) one-way, where $t' = t + \mathcal{O}((\log n)^3)$.*

Proof. Suppose that there exists a PPT algorithm A which breaks the one-wayness of RSA-Paillier+(n, e) with probability ε in time t. We will show an algorithm M_0 which breaks the one-wayness of RSA$(n, 3)$ with probability ϵ in time $t + \mathcal{O}((\log n)^3)$.

Let $(n, 3, y)$ be the input to M_0, where M_0 wants to find $x \in Z_n$ such that $y = x^3 \bmod n$.

M_0 first chooses a prime s such that $|n|/2 < |s| < |n|$ randomly, and computes $e = 3s$. It next chooses a fake plaintext $m' \in Z_{n^2}$ randomly, and computes a ciphertext

$$c = y^s + m'n \bmod n^3. \tag{19}$$

M_0 then runs A on input (n, e, c). A outputs a plaintext $m \in Z_{n^2}$ with probability ε in time t. Then we have

$$c = r^e + mn \bmod n^3 \tag{20}$$

for some $r \in Z_n$. From eq.(19) and eq.(20), we obtain that

$$c = y^s + m'n = x^{3s} + m'n = r^{3s} + mn \bmod n^3.$$

It is now easy to see that $x^{3s} = r^{3s} \bmod n$. Hence $x = r$ because $x, r \in Z_n$ and $\gcd(3s, p-1) = \gcd(3s, q-1) = 1$. Therefore,

$$x^{3s} = x^e = r^e = c - mn \bmod n^3. \tag{21}$$

On the other hand, x^3 is written as

$$x^3 = y + t_0 n + t_1 n^2 \tag{22}$$

for some $t_0, t_1 \in Z_n$ because $x^3 < n^3$. First we show that we can compute t_0. From eq.(22), it holds that

$$\begin{aligned} x^{3s} &= (y + t_0 n + t_1 n^2)^s \bmod n^2 \\ &= y^s + s y^{s-1}(t_0 n + t_1 n^2) \bmod n^2 \\ &= y^s + s y^{s-1} t_0 n \bmod n^2 \end{aligned}$$

Therefore, it is easy to see that

$$\frac{x^{3s} - y^s \bmod n^2}{n} = s y^{s-1} t_0 \bmod n.$$

Hence we can obtain t_0 as

$$\begin{aligned} t_0 &= \left(\frac{x^{3s} - y^s \bmod n^2}{n} \right) / s y^{s-1} \bmod n \\ &= \left(\frac{c - mn - y^s \bmod n^2}{n} \right) / s y^{s-1} \bmod n, \end{aligned}$$

where we used eq.(21).

Similarly, we can compute t_1 as follows. From eq.(22), it holds that

$$\begin{aligned} x^{3s} &= (y + t_0 n + t_1 n^2)^s \bmod n^3 \\ &= (y + t_0 n)^s + s(y + t_0 n)^{s-1} t_1 n^2 \bmod n^3. \end{aligned}$$

Therefore, it is easy to see that

$$\frac{x^{3s} - (y + t_0 n)^s \bmod n^3}{n^2} = s(y + t_0 n)^{s-1} t_1 = s y^{s-1} t_1 \bmod n.$$

Hence we can obtain t_1 as

$$\begin{aligned} t_1 &= \left(\frac{x^{3s} - (y + t_0 n)^s \bmod n^3}{n^2} \right) / s y^{s-1} \bmod n \\ &= \left(\frac{c - mn - (y + t_0 n)^s \bmod n^3}{n^2} \right) / s y^{s-1} \bmod n, \end{aligned}$$

where we used eq.(21).

Finally, substitute the above t_0 and t_1 into eq.(22). Then we can find x. This means that M_0 can break the one-wayness of RSA$(n, 3)$ with probability ε in time $t' = t + \mathcal{O}((\log n)^3)$.

□

7.4 Generalization

We can generalize RSA-Paillier+ so that the message space is Z_{n^k} and the modulus is n^{k+1} for $k \geq 2$. The key generation algorithm and the decryption algorithm are the same as in RSA-Paillier+. The encryption algorithm is described as follows.

Encryption: To encrypt a message $m \in Z_{n^k}$, choose $r \in Z_n$ randomly and compute

$$c = r^e + mn \bmod n^{k+1}.$$

The ciphertext is c.

We call this scheme the *generalized* RSA-Paillier+ encryption scheme. It improves the bandwidth even more, and IND-CPA security holds under the assumption that distinguishing $\{x \mid x = r^e \bmod n^{k+1}, r \in Z_n\}$ from $Z_{n^{k+1}}$ is hard. Further, we can prove the one-wayness similarly to Theorem 7.1.

Corollary 7.1 *The generalized RSA-Paillier+(n, e) is (t, ε) one-way if RSA$(n, 3)$ is (t', ε) one-way, where $t' = t + \mathcal{O}((\log n)^3)$.*

The proof is almost the same as that of Theorem 7.1.

7.5 Discussion

In the random oracle model, the following encryption schemes are proved to be secure in the sense of IND-CCA.

1. Rabin-SAEP under Blum integer factoring assumption [3].
2. RSA-OAEP with $e = 3$ under RSA assumption [32].
3. RSA-OAEP with general e under RSA assumption [14].

The security proof of RSA-OAEP with general e uses LLL Algorithm, and its security reduction is not as tight as those of Rabin-SAEP and RSA-OAEP with $e = 3$ using the Coppersmith algorithm.

Now there is an interesting correspondence between the one-wayness of some variants of RSA-Paillier encryption schemes in the standard model and the above results in the random oracle model:

- Our result on Rabin-Paillier encryption scheme corresponds to Rabin-SAEP because both schemes are tightly reduced to factoring Blum integers.
- Our result on RSA-Paillier+ corresponds to RSA-OAEP with $e = 3$ because both schemes are tightly reduced to RSA with $e = 3$.
- RSA-Paillier corresponds to RSA-OAEP with general e because both schemes are loosely reduced to RSA with general e.

8 CCA-Secure KEM under General Factoring

In this section, we present the first CCA-secure KEM under the general factoring assumption in the random oracle model.

8.1 Definition of KEM [12, Sec.7.1]

It is known that by combining a KEM and a one-time symmetric encryption scheme which are both secure in the sense of IND-CCA, we can obtain a hybrid encryption scheme which is secure in the sense of IND-CCA. A key encapsulation mechanism KEM consists of the following algorithms.

- A key generation algorithm $\mathsf{KEM.Gen}$ that on input 1^l outputs a public/secret key pair (pk, sk).
- An encryption algorithm $\mathsf{KEM.Enc}$ that on input 1^l and a public key pk, outputs a pair (K, ψ), where K is a key and ψ is a ciphertext.

  A key K is a bit string of length $\mathsf{KEM.Len}(l)$, where $\mathsf{KEM.Len}(l)$ is another parameter of KEM.
- A decryption algorithm $\mathsf{KEM.Dec}$ that on input 1^l, a secret key sk, a string (in particular a ciphertext) ψ, outputs either a key K or the special symbol reject.

$\mathsf{KEM.Gen}$ and $\mathsf{KEM.Enc}$ are PPT algorithms and $\mathsf{KEM.Dec}$ is a deterministic polynomial time algorithm.

In the chosen ciphertext attack (IND-CCA) game, we imagine a PPT adversary A that runs in two stages. In the find stage, A takes a public key pk and queries an encryption oracle. The encryption oracle computes:

$$(K^*, \psi^*) \xleftarrow{R} \mathsf{KEM.Enc}(1^l); K^+ \xleftarrow{R} \{0,1\}^k; b \xleftarrow{R} \{0,1\};$$
$$\text{if } b = 0 \text{ then } K^\dagger \leftarrow K^* \text{ else } K^\dagger \leftarrow K^+ \tag{23}$$

where $k = \mathsf{KEM.Len}(l)$, and responds with the pair $(K^\dagger, \psi^*)$. In the guess stage, given $(K^\dagger, \psi^*)$, the adversary A outputs a bit $\tilde{b}$ and halts.

The adversary A is also given access to a decryption oracle. For each decryption oracle query, the adversary A submits a ciphertext ψ, and the decryption oracle responds with $\mathsf{KEM.Dec}(1^l, sk, \psi)$, where A cannot query the challenge ciphertext ψ^* itself in the guess stage.

Definition 8.1 *We say that* KEM *is secure in the sense of IND-CCA if* $|\Pr(\tilde{b} = b) - 1/2|$ *is negligible in the above game for any PPT adversary* A.

In particular, we define the IND-CCA advantage of A as follows.

$$\mathtt{Adv}^{cca}_{\mathsf{KEM}}(A) = |\Pr(\tilde{b} = b) - 1/2|. \tag{24}$$

In the random oracle model, define $\mathtt{Adv}^{cca}_{\mathsf{KEM}}(t, q_d, q_h) = \max_A \mathtt{Adv}^{cca}_{\mathsf{SKE}}(A)$, where the maximum is taken over all A which runs in time t, makes at most q_d queries to the decryption oracle and makes at most q_h queries to the random oracle H.

Definition 8.2 *We say that* KEM *is* $(t, q_d, q_h, \varepsilon)$*-secure if*

$$\mathtt{Adv}^{cca}_{\mathsf{KEM}}(t, q_d, q_h) < \varepsilon.$$

8.2 Proposed KEM

Let H be a random hash function.

(Secret key) Two prime numbers p and q such that $|p| = |q|$.

(Public key) $n(= pq)$ and $\alpha \in Z_n^*$ such that

$$(\alpha/p) = (\alpha/q) = -1. \tag{25}$$

(Remark) We say that $r \in Z_n^*$ is principal if $(r/n) = 1$ and $(\alpha/r \bmod n) > r$. Otherwise, we say that r is non-principal.

(Encryption) Choose a principal r randomly. Compute

$$\begin{aligned} \psi &= r + \frac{\alpha}{r} \bmod n \\ K &= H(r). \end{aligned} \tag{26}$$

Output (K, ψ).

(Decryption) For a given ψ, first compute a principal r which satisfies eq.(26) Then compute $K = H(r)$.

8.3 Security

We now prove that the proposed KEM is secure in the sense of IND-CCA if the general factoring problem is hard.

Lemma 8.1 *[12, Lemma 6.2] Let S_1, S_2 and F be events defined on some probability space. Suppose that the event $S_1 \wedge \neg F$ occurs if and only if $S_2 \wedge \neg F$ occurs. Then*

$$|\Pr(S_1) - \Pr(S_2)| \leq \Pr(F).$$

Theorem 8.1 *The proposed KEM is $(t, q_d, q_h, \varepsilon)$-secure if the general factoring problem is (t', ε')-hard, where*

$$\begin{aligned} \varepsilon &\leq 2\varepsilon' + 5q_d/n, \\ t' &= t + O(q_h + q_d). \end{aligned}$$

(Proof) Let $\mathbf{G}_0$ be the original attack game on the proposed KEM with an adversary A. That is,

step 1. A random public key (n, α) is given to the adversary A.

step 2. The adversary A queries to the encryption oracle, and the encryption oracle responds with $(K^\dagger, \psi^*)$ according to the hidden bit b of eq.(23).

step 3. The adversary A outputs a bit $\tilde{b}$.

During the game, A can query ciphertexts ψ to the decryption oracle adaptively, and the decryption oracle responds with the keys K.

Let T_0 be the event that $b = \tilde{b}$. We define a sequence of games $\mathbf{G}_1, \mathbf{G}_2$, and define T_i be the event that $b = \tilde{b}$ in game $\mathbf{G}_i$.

Game G_1. We modify game $\mathbf{G}_0$ as follows. The encryption oracle computes $(K^\dagger, \psi^*)$ at the beginning of the game. Second, if A submits a ciphertext $\psi = \psi^*$ to the decryption oracle before the encryption oracle is invoked, then we abort the game immediately.

Let F_1 be the event that game $\mathbf{G}_1$ is aborted as above. If n is large enough, then the size of the set of ψ is bounded by

$$|\{\psi\}| = |Z_n^*|/4 = (p-1)(q-1)/4 > n/5$$

because $r \in Z_n^*$ and the function $r \vdash \psi$ is a $4:1$ function. Therefore, it holds that

$$\Pr[F_1] \leq q_d/|\{\psi\}| \leq 5q_d/n.$$

It is clear that games $\mathbf{G}_0$ and $\mathbf{G}_1$ proceed identically until event F_1 occurs, and so by Lemma 8.1, we have $|\Pr[T_1] - \Pr[T_0]| \leq \Pr[F_1]$.

Game G_2. We next modify game $\mathbf{G}_1$ as follows. If A queries the principal r^* such that

$$\psi^* = r^* + \frac{\alpha}{r^*} \bmod n,$$

to the random oracle H, then we abort the game immediately. It is easy to see that $\Pr[T_2] = 1/2$ because A has no information on $K^* = H(r^*)$ in this case.

Let F_2 be the event that game $\mathbf{G}_1$ is aborted as above. It is clear that games $\mathbf{G}_1$ and $\mathbf{G}_2$ proceed identically until event F_2 occurs, and so by Lemma 8.1, we have $|\Pr[T_2] - \Pr[T_1]| \leq \Pr[F_2]$.

We will show that there exists a PPT algorithm B which can factor n with probability $\varepsilon' = \Pr[F_2]/2$. On input n, B simulates the environment of game G_2 for A as follows.

1. B chooses $\alpha \in Z_n^*$ such that $(\alpha/n) = 1$ randomly. It is easy to see that α satisfies eq.(25) with probability $1/2$.

2. B chooses $K^\dagger$ randomly. Note that $K^* = H(r^*)$ is random because H is a random oracle. Therefore, $K^\dagger$ is random from a view point of A regardless of the value of b.

3. B next chooses $\bar{r}$ such that $(\bar{r}/N) = -1$ randomly and computes

$$\psi^* = \bar{r} + \frac{\alpha}{\bar{r}} \bmod n.$$

   This has no problem because there exists a principal r^* such that

$$\psi^* = r^* + \frac{\alpha}{r^*} \bmod n$$

   from Lemma 4.1.

4. B then runs A on input $pk = (n, \alpha)$.

5. If A queries to the encryption oracle, then B returns $(K^\dagger, \psi^*)$ to A.

To simulate the decryption oracle, B maintains a list $\mathcal{L}_d$ of (ψ, K). To simulate the random oracle H, B maintains a list $\mathcal{L}_{H+}$ of (r, K) such that $H(r) = K$ for a principal r. Similarly, B maintains a list $\mathcal{L}_{H-}$ of (r, K) such that $H(r) = K$ for a non-principal r. Initially, $\mathcal{L}_H$ and $\mathcal{L}_d$ are empty.

Suppose that A queries ψ to the decryption oracle. Then B executes the following algorithm.

- If there exists $(r, K) \in \mathcal{L}_{H+}$ such that

$$\psi = r + \frac{\alpha}{r} \bmod n,$$

  then return K.

- Otherwise choose K randomly, add (ψ, K) to $\mathcal{L}_d$ and return K.

Suppose that A queries a non-principal r to H. Then B executes the following algorithm.

- If $(r, K) \in \mathcal{L}_{H-}$ for some K, then return K.

- Otherwise choose K randomly, add (ψ, K) to $\mathcal{L}_{H-}$ and return K.

Suppose that A queries a principal r to H. Then B executes the following algorithm.

- (The event F_2 occurs.) If $r = r^*$ such that

$$\psi^* = r^* + \frac{\alpha}{r^*} \bmod n, \tag{27}$$

  then B computes $y = \gcd(r^* - \bar{r})$, outputs y and halts.

- Else if $(r, K) \in \mathcal{L}_{H+}$ for some K, then return K.

- Else if there exists $(\psi, K) \in \mathcal{L}_d$ such that

$$\psi = r + \frac{\alpha}{r} \bmod n,$$

  then return K.

- Otherwise choose K randomly, add (ψ, K) to $\mathcal{L}_{H+}$ and return K.

We have finished the description of B. Now if F_2 occurs, then B obtains $\bar{r}$ and r^* such that

$$\bar{r} + \frac{\alpha}{\bar{r}} = \psi^* = r^* + \frac{\alpha}{r^*} \bmod n.$$

In this case, it holds that $y = \gcd(r^* - \bar{r}) = p$ or q from Lemma 4.1. Further, if α satisfies eq.(25), then B simulates the environment of game $\mathbf{G}_2$ for A

perfectly. Hence B can factor n with probability $\varepsilon' = \Pr[F_2]/2$ because α satisfies eq.(25) with probability 1/2.

Now we have

$$\begin{aligned}
\varepsilon &\stackrel{\triangle}{=} |\Pr[T_0] - (1/2)| \\
&= |\Pr[T_0] - \Pr[T_1] + \Pr[T_1] - \Pr[T_2] + \Pr[T_2] - (1/2)| \\
&\leq |\Pr[T_0] - \Pr[T_1]| + |\Pr[T_1] - \Pr[T_2]| + |\Pr[T_2] - (1/2)| \\
&\leq \Pr[F_1] + \Pr[F_2] + 0 \\
&\leq 5q_d/n + 2\varepsilon'.
\end{aligned}$$

It is easy to see that B runs in time $t' = t + O(q_h + q_d)$ if A runs in time t. This means that the theorem holds.

Q.E.D.

9 Conclusion

We showed the first practical semantically secure public-key encryption scheme such that its one-wayness is equivalent to *general* factoring in the *standard* model (in the sense of IND-CPA).

We next applied our proof technique to Rabin-Paillier encryption scheme and RSA-Paillier+ encryption scheme to prove their exactly tight one-wayness.

We finally presented the first KEM which is secure in the sense of IND-CCA under general factoring assumption in the random oracle model.

It will be a further work to develop a CCA-secure KEM under general factoring assumption in the standard model. We hope that our results provide us a good starting point to this challenging problem.

Acknowledgement

We thank Katja Schmidt-Samoa for her suggestion of Sec.7.4.

References

[1] M.Bellare, A.Boldyreva and A.Palacio. An Uninstantiable Random-Oracle-Model Scheme for a Hybrid-Encryption Problem. EUROCRYPT 2004, LNCS 3027, pp.171–188 (2004)

[2] M. Bellare, A. Desai, D. Pointcheval, and P. Rogaway, "Relations among Notions of Security for Public-Key Encryption Schemes," CRYPTO'98, LNCS 1462, pp.26-45, 1998.

[3] D. Boneh, "Simplified OAEP for RSA and Rabin Functions," CRYPTO 2001, LNCS 2139, pp.275-291, 2001.

[4] R.Canetti, O.Goldreich and S.Halevi. The Random Oracle Methodology, Revisited (Preliminary Version). STOC 1998, pp.209–218 (1998)

[5] R.Canetti, O.Goldreich and S.Halevi. On the Random-Oracle Methodology as Applied to Length-Restricted Signature Schemes. TCC 2004, LNCS 2951, pp.40–57 (2004)

[6] D. Catalano, R. Gennaro, N. Howgrave-Graham, and P. Nguyen; "Paillier's cryptosystem revisited," The 8th ACM conference on Computer and Communication Security, pp.206-214, 2001.

[7] D. Catalano, P. Nguyen, and J. Stern, "The Hardness of Hensel Lifting: The Case of RSA and Discrete Logarithm," ASIACRYPT 2002, LNCS 2501, pp.299-310, 2002.

[8] D. Coppersmith, "Finding a Small Root of a Univariate Modular Equation," EUROCRYPT '96, LNCS 1070, pp.155-165, 1996.

[9] J. -S. Coron, H. Handschuh, M. Joye, P. Paillier, D. Pointcheval, and C. Tymen, "GEM: A Generic Chosen-Ciphertext Secure Encryption Method," Topics in Cryptology - CT-RSA 2002, LNCS2271, pp.263-276, 2002.

[10] R. Cramer and V. Shoup, "A Practical Public-Key Cryptosystem Provably Secure against Adaptive Chosen Ciphertext Attacks," CRYPTO'98, LNCS 1462, pp.13-25, 1998.

[11] R. Cramer and V. Shoup, "Universal Hash Proofs and a Paradigm for Adaptive Chosen Ciphertext Secure Public-Key Encryption," EUROCRYPT 2002, LNCS 2332, pp.45-64, 2002.

[12] R. Cramer and V. Shoup, Design and analysis of practical public-key encryption schemes secure against adaptive chosen ciphertext attack, SIAM Journal on Computing, Volume 33, Number 1, pp. 167-226 (2003)

[13] E.Fujisaki and T.Okamoto, "Secure Integration of Asymmetric and Symmetric Encryption Schemes," CRYPTO 1999, LNCS 1666, pp.537–554, 1999

[14] E. Fujisaki, T. Okamoto, D. Pointcheval, and J. Stern, "RSA-OAEP Is Secure under the RSA Assumption," CRYPTO 2001, LNCS 2139, pp.260-274, 2001.

[15] D. Galindo, S. Molleví, P. Morillo, J. Villar, "A Practical Public Key Cryptosystem from Paillier and Rabin Schemes," PKC 2003, LNCS 2567, pp.279-291, 2003.

[16] O. Goldreich, *Foundations of Cryptography: Basic Tools*, Cambridge University Press, 2001.

[17] S.Goldwasser and S.Micali, *Probabilistic Encryption*, Journal of Computer and System Sciences, volume 28, pages 270–299 (1984)

[18] K. Kurosawa, T. Itoh, M. Takeuchi, "Public Key Cryptosystem using a Reciprocal Number with the Same Intractability as Factoring a Large Number," CRYPTOLOGIA, XII, pp.225-233, 1988.

[19] K. Kurosawa, W. Ogata, T. Matsuo, S. Makishima, "IND-CCA Public Key Schemes Equivalent to Factoring n=pq, PKC 2001, LNCS 1992, pp36-47, 2001.

[20] H.Krawczyk, LFSR-based Hashing and Authentication, CRYPTO 1994, pp.129-139 (1994)

[21] K. Kurosawa and T. Takagi, "Some RSA-Based Encryption Schemes with Tight Security Reduction," ASIACRYPT 2003, LNCS 2894, pp.19-36, 2003.

[22] K. Kurosawa, K. Schmidt-Samoa, T. Takagi, "A Complete and Explicit Security Reduction Algorithm for RSA-based Cryptosystems," Asiacrypt 2003, LNCS 2894, pp.474-491, 2003.

[23] M. Luby, *Pseudorandomness and Cryptographic Applications*, Princeton University Press, 1966.

[24] U.Maurer, R.Renner, C.Holenstein. Indifferentiability, Impossibility Results on Reductions, and Applications to the Random Oracle Methodology. TCC 2004, LNCS 2951, pp.21–39 (2004)

[25] J.B.Nielsen. Separating Random Oracle Proofs from Complexity Theoretic Proofs: The Non-committing Encryption Case. CRYPTO 2002, LNCS 2442, pp.111–126 (2002)

[26] T. Okamoto and D. Pointcheval, "REACT: Rapid Enhanced-security Asymmetric Cryptosystem Transform," In Proceedings of the Cryptographers' Track at RSA Conference '2001, LNCS 2020, (2001), pp.159-175.

[27] T.Okamoto and S.Uchiyama, "A New Public Key Cryptosystem as Secure as Factoring," Eurocrypt'98, LNCS 1403, pp.308–318, 1998

[28] P. Paillier, "Public-Key Cryptosystems Based on Composite Degree Residuosity Classes", Eurocrypt'99, LNCS 1592, pp.223–238, 1999

[29] D.Pointcheval, "New Public Key Cryptosystems based on the Dependent-RSA Problems," Eurocrypt'99, LNCS 1592, pp.239-254, 1999

[30] K. Sakurai, T. Takagi, "New Semantically Secure Public-Key Cryptosystems from the RSA-Primitive," PKC 2002, LNCS 2274, pp.1-16, 2002.

[31] V. Shoup, "Using Hash Functions as a Hedge against Chosen Ciphertext Attack," EUROCRYPT 2000, LNCS 1807, pp.275-288, 2000.

[32] V. Shoup, "OAEP Reconsidered," CRYPTO 2001, LNCS 2139, pp.239-259, 2001.

[33] D. Stinson, "Universal hash families and the leftover hash lemma, and applications to cryptography and computing," J. Combin. Math. Combin. Comput. vol.42, pp.3-31, 2002.

[34] T. Takagi, "Fast RSA-Type Cryptosystems using N-adic Expansion," CRYPTO '97, LNCS 1294, pp.372-384, 1997.

A Flaw on the Semantic Security of Rabin-Paillier

Let

$$\begin{aligned} SMALL_{QR}(n,e) &\stackrel{\triangle}{=} \{(n,e,x) \mid x = r^{2e} \bmod n^2, r \in QR_n\} \\ LARGE_{QR}(n,e) &\stackrel{\triangle}{=} \{(n,e,x) \mid x = r^{2e} \bmod n^2, r \in Q_{n^2}\} \end{aligned}$$

Rabin-Paillier encryption scheme is IND-CPA if and only if $SMALL_{QR}(n,e)$ and $LARGE_{QR}(n,e)$ are indistinguishable [15, Proposition 9].

Galindo et al. further claimed that $SMALL_{QR}(n,e)$ and $LARGE_{QR}(n,e)$ are indistinguishable if

- $SMALL_{RSA}(n,e)$ and $LARGE_{RSA}(n,e)$ are indistinguishable (RSA-Paillier is IND-CPA under this condition) and
- $QR(n)$ and $QNR(n,+)$ are indistinguishable, where

$$\begin{aligned} QR(n) &\stackrel{\triangle}{=} \{(n,x) \mid x \in QR_n\} \\ QNR(n,+) &\stackrel{\triangle}{=} \left\{(n,x) \mid x \in Z_n^*, \left(\frac{x}{n}\right) = 1\right\} \end{aligned}$$

in [15, Proposition 11].

However, this claim is wrong. In the proof, they say that D_1 and D_2 are indistinguishable, where

$$\begin{aligned} D_1 &\stackrel{\triangle}{=} \{x \mid x = r^e \bmod n^2, r \in QR_n\} \\ D_2 &\stackrel{\triangle}{=} \{x \mid x = r^e \bmod n^2, r \in Z_n^*\}. \end{aligned}$$

However, we can distinguish them easily by computing $(\frac{x}{n})$.

B Decryption of Kurosawa et al's Encryption Scheme

Let a_1 and a_2 be the roots of $eq.(3) \bmod p$ and b_1 and b_2 be the roots of $eq.(3) \bmod q$. Then, $eq.(3) \bmod n$ has the following four roots:

$$\begin{aligned} M_1 &= [a_1, b_1], \qquad M_2 = [a_2, b_2] \\ M_3 &= [a_1, b_2], \qquad M_4 = [a_2, b_1] \end{aligned}$$

where $M_1 = [a_1, b_1]$ means $M_1 = a_1 \bmod p$ and $M_1 = b_1 \bmod q$.

The plaintext m is one of the four roots. s and t tell the receiver which root the plaintext m is. From the relationship between the roots and the coefficients of eq.(3), we obtain

$$(a_1/p)(a_2/p) = (\alpha/p) = -1.$$

We set

$$(a_1/p) = 1, \qquad (a_2/p) = -1. \tag{28}$$

Similarly, we set

$$(b_1/q) = 1, \qquad (b_2/q) = -1. \tag{29}$$

Then, we obtain

$$(M_1/n) = (M_1/p)(M_1/q) = (a_1/p)(b_1/q) = 1.$$

Similarly, we get

$$\begin{aligned} (M_2/n) &= 1 \\ (M_3/n) &= (M_4/n) = -1. \end{aligned}$$

Therefore, the receiver sees that

$$m = \begin{cases} M_1 \text{ or } M_2 & \text{if } s = 0; \\ M_3 \text{ or } M_4 & \text{if } s = 1. \end{cases}$$

Now, suppose that $s = 0$. The relationship between the roots and the coefficients of eq.(3) gives us

$$M_1 M_2 = [a_1 a_2, b_1 b_2] = [\alpha, \alpha] = \alpha \bmod n.$$

Hence,

$$M_2 = \alpha / M_1 \bmod n.$$

Therefore, the receiver sees that

$$m = \begin{cases} \min(M_1, M_2) & \text{if } t = 0; \\ \max(M_1, M_2) & \text{if } t = 1. \end{cases}$$

When $s = 1$,

$$m = \begin{cases} \min(M_3, M_4) & \text{if } t = 0; \\ \max(M_3, M_4) & \text{if } t = 1. \end{cases}$$

Thus, a ciphertext is uniquely deciphered.