An Efficient Public Key Cryptosystem with a Privacy Enhanced Double Decryption Mechanism

Taek-Young Youn[1,⋆], Young-Ho Park[2], Chang Han Kim[3], and Jongin Lim[1]

[1] Graduate School of Information Security, Korea University, Seoul, Korea
{taekyoung, jilim}@cist.korea.ac.kr
[2] Dept. of Information Security, Sejong Cyber University, Seoul, Korea
youngho@cybersejong.ac.kr
[3] Dept. of Information Security, Semyung University, Jecheon, Korea
chkim@semyung.ac.kr

Abstract. A clue of double decryption mechanism was introduced at Eurocrypt '02 by Cramer and Shoup, and it was revisited at Asiacrypt '03 by Bresson, Catalano and Pointcheval. Previous double decryption schemes are designed based on $\mathbb{Z}_{n^2}$ where $n = pq$ for two primes, p and q. Note that, they use the Paillier's scheme as a primitive scheme to design a double decryption mechanism. In this paper, we propose an efficient public key scheme with double decryption mechanism based on $\mathbb{Z}_{p^2q}$. Our scheme is more efficient than the previous schemes. Moreover, we review the previous schemes in a privacy point of view and propose a privacy enhanced double decryption scheme.

Keywords: public key cryptosystem, double trapdoor decryption mechanism, semantic security, privacy.

1 Introduction

Public key cryptosystem (PKC) is regarded as a useful tool for secure communication, therefore designing a good PKC is an essential task for not only theoretic purpose but also practical purpose. When designing a PKC, we have to consider two conditions, security and efficiency.

In security point of view, two conditions are considered, one-wayness and semantic security. One-wayness is regarded as a basic condition for secure scheme, but it is not a sufficient condition to gain real security. In these days, the semantic security is also required as a fundamental condition for the security of PKC. In [2], some examples which illustrate the need of semantic security are listed. For example, to design an authenticated key exchange protocol in the public-key setting, the scheme has to meet IND-CCA2 security. However, even though a scheme is secure, we have to check whether the scheme is efficient or not. In general, the efficiency is regarded as an important condition as the security.

[⋆] This research was supported by the MIC (Ministry of Information and Communication), Korea, under the ITRC (Information Technology Research Center) support program supervised by the IITA(Institute of Information Technology Assessment).

B. Preneel and S. Tavares (Eds.): SAC 2005, LNCS 3897, pp. 144–158, 2006.

In [4], Cramer and Shoup introduced a clue of double decryption mechanism. After that, Bresson *et al.* revisited the mechanism and proposed a double decryption scheme [1]. The previous double decryption mechanisms are derived from Paillier's scheme [13], which is designed based on $\mathbb{Z}_{n^2}$ where $n = pq$ for two primes. Since the size of modulo of Paillier's scheme is twice than that of standard RSA or ElGamal in same security level, the efficiency of Paillier's scheme is not favorable. Hence the efficiency of the previous double decryption scheme is also not good. So, it is worth designing an efficient double decryption scheme.

In [1], Bresson *et al.* stated the necessity of double decryption scheme and proposed a scheme that solves the following two scenarios.

1. The head of a group may want to be able to read any message sent to the members of the group.
2. People may want to be able to recover the plaintexts even if they loose their private key.

To solve above two scenarios simultaneously, we need a kind of super-key. In [1], factoring information is given to authority as a super-key. But, in this scenario, we have some apprehensions about an invasion of privacy, since authority can decrypt any ciphertext without any consent of corresponding user. To prevent the abuse of super-key, i.e., to enhance the privacy of users, we need a way to restrict the excessive ability of authority.

In this paper, we propose an efficient double decryption scheme based on $\mathbb{Z}_{p^2q}$. Our scheme is efficient than the previous schemes [4, 1], since our scheme executes the cryptographic operations, such as modulo multiplication and exponentiation, on a smaller modulo than the previous schemes. Our basic scheme provides about 3 times faster encryption and decryption for ordinary user than the previous scheme. Moreover, the authority can decrypt a ciphertext about 4 times faster than the previous scheme. By modifying our basic scheme, we propose a scheme which can get rid of the apprehension of the invasion of privacy without losing the double decryption mechanism.

2 Preliminaries

From now, we review two previous schemes and describe the number-theoretic framework underlying our scheme.

2.1 Previous Schemes

In [1], Bresson, Catalano and Pointcheval proposed a modification of Paillier's scheme, which provides double trapdoor decryption mechanism. Let $n = pq$ be a safe-prime modulo, i.e. p and q are primes of the form $p = 2p' + 1$ and $q = 2q' + 1$, where p' and q' are also primes. Let $\mathbb{G}$ be the cyclic group of quadratic residues modulo n^2. Then $ord(\mathbb{G}) = \lambda(n^2)/2 = n\lambda(n)/2$. Note that every element of order n is of the form $\alpha = 1 + kn$ for some $k \in \mathbb{Z}_n$.

Key generation. Choose two primes p and q ($|p| = |q|$), and let $n = pq$. Choose a random $\alpha \in \mathbb{Z}_{n^2}$, a random number $a \in [1, ord(\mathbb{G})]$ and set $g = \alpha^2 \mod n^2$

and $h = g^a \mod n^2$. The public key is (n, g, h) while the corresponding secret key is a. Two prime factors, p and q, are superkey-like secret key.

Encryption. Given a message $m \in \mathbb{Z}_n$, choose a random $r \in \mathbb{Z}_{n^2}$. Then the ciphertext is computed as follows: $C = (A, B)$ where $A = g^r \mod n^2$ and $B = h^r(1 + mn) \mod n^2$.

Decryption1. First trapdoor is similar to that of ElGamal scheme. Knowing a, decryption operation is executed as follows: $m = (B/A^a - 1 \mod n^2)/n$.

Decryption2. Second decryption mechanism uses the factoring information of n as a secret key. If the prime factors are provided, $a \mod n$ and $r \mod n$ are easily recovered (see [1] for details). If we write $ar \mod ord(\mathbb{G})$ by $\gamma_1 + \gamma_2 n$ then $\gamma_1 = ar \mod n$. Firstly, compute

$$D = (\frac{B}{g^{\gamma_1}})^{\lambda(n)} = \frac{(g^{ar}(1+mn))^{\lambda(n)}}{g^{\gamma_1 \lambda(n)}} = 1 + m\lambda(n)n \mod n^2.$$

Let π be the inverse of $\lambda(n)$ in $\mathbb{Z}_n^*$, then m is recovered by the following simple computation: $m = ((D - 1 \mod n^2)/n)\pi \mod n$.

By analyzing the scheme in [1], we can see that if a group has a mechanism that solves the discrete logarithm problem, a double decryption scheme can be designed based on the group.

In [12], a novel scheme proposed by Okamoto and Uchiyama as OU scheme in short, is based on $\mathbb{Z}_{p^2q}$. The scheme uses such a mechanism that solves the discrete logarithm problem. Therefore we can design a double decryption scheme based on $\mathbb{Z}_{p^2q}$. So, it is meaningful to review the scheme [12].

OU scheme is based on a logarithmic function, L, defined over p-Sylow subgroup of $\mathbb{Z}_{p^2}^*$. Let $\Gamma = \{x \in \mathbb{Z}_{p^2}^* | x \equiv 1 \mod p\}$. Then, for $x \in \Gamma$, L is defined as $L(x) = \frac{x-1}{p}$. The function L is well-defined and has a homomorphic property from multiplication to addition (see [12] for details). Let $x_p = x^{p-1} \mod p^2$ for $x \in \mathbb{Z}_n$.

Key generation. Choose two primes p and q ($|p| = |q| = k$), and let $n = p^2q$. Choose a random g in $\mathbb{Z}_n$ such that the order of $g^{p-1} \mod p^2$ is p. Let $h = g^n \mod n$. The public key is (n, g, h, k) while the corresponding secret key is (p, q).

Encryption. Given a message $m \in [1, 2^{k-1}]$, choose a random $r \in \mathbb{Z}_n$. Then the ciphertext is computed as following: $C = g^m h^r \mod n$.

Decryption. Compute $C_p = C^{p-1} \mod p^2$ and $g_p = g^{p-1} \mod p^2$. Then the plaintext is computed as following: $m = L(C_p)/L(g_p) \mod p$.

2.2 Discrete Logarithm and Diffie-Hellman Problem over $\mathbb{Z}_{p^2q}$

The Diffie-Hellman problem [6] is a well-known cryptographic primitive. Until now, the Diffie-Hellman problem remains the most widely used cryptographic technique. Our scheme is also designed based on a kinds of Diffie-Hellman problem, denoted by p-DHP.

Let $\mathcal{P}(k)$ be the set of prime numbers of length k. Choose two primes p and q in $\mathcal{P}(k)$. From now, let $n = p^2q$. Let $\mathbb{G}_p = \{x \in \mathbb{Z}_n \mid \text{order of } x^{p-1} \mod p^2 \text{ is } p\}$. Formal definition of p-DHP is described below.

Definition 1. *(p-DHP) The p-DHP is defined as follows: Given a set $\mathbb{G}_p$, an element g of $\mathbb{G}_p$ and ($g^a \mod n$, $g^b \mod n$) for $a, b \in_R [1, p-1]$, find $g^{ab} \mod n$.*

Although it is not known whether DHP over $\mathbb{Z}^*_{p^2q}$ is more tractable than DHP over $\mathbb{Z}^*_{rs}$ (r and s are prime numbers such that $|r| = |s|$) or vice versa, the security of a prime order subgroup of $\mathbb{Z}^*_{rs}$ is studied in [10]. The attack described in [10] is valid only on the prime order subgroup of $\mathbb{Z}^*_{rs}$ rather than composite order subgroup. Note that, our scheme use a generator g whose order is composite number. Moreover, there is no known attack for breaking the DHP over $\mathbb{Z}^*_{p^2q}$. So, the hardness of p-DHP is based on the size of modulo. The size of exponent, k bit, is not too small to be broken, since 160 bit of exponent is sufficient to gain the desired security on DHP in these days.

Conjecture 1. For every probabilistic polynomial time algorithm $\mathcal{A}$, there exists a negligible function $negl(\cdot)$ such that for sufficiently large k,

$$\Pr\left[\mathcal{A}(n, A, B) = C \;\middle|\; \begin{array}{ll} p, q \leftarrow \mathcal{P}(k); & n = p^2q; \\ g \leftarrow \mathbb{G}_p; & a, b \leftarrow_R [1, p-1]; \\ A = g^a \mod n; & B = g^b \mod n; \\ C = g^{ab} \mod n; & \end{array}\right] \leq negl(k).$$

From now, we define a kind of DLP over $\mathbb{Z}^*_n$, denoted by p-DLP. After that, we will prove that the hardness of p-DLP is equivalent to factoring n.

Definition 2. *(p-DLP) The p-DLP is defined as follows: Given a set $\mathbb{G}_p$, an element g of $\mathbb{G}_p$ and $g^a \mod n$ for $a \in_R \mathbb{Z}_n$, find $a \mod p$.*

Theorem 1. *p-DLP over $\mathbb{Z}^*_n$ is hard to solve if and only if the factoring assumption holds.*

Proof. ($\Rightarrow$) Suppose that the factoring assumption does not hold. Let $A = g^a \mod n$ for some $a \in \mathbb{Z}_n$. Since we can find the factoring of n, $a \mod p$ is recovered as following: $a' = a \mod p = L(A_p)/L(g_p) \mod p$.

($\Leftarrow$) Suppose that there exist an algorithm $\mathcal{A}$ which solves p-DLP over $\mathbb{Z}_n$. Choose a random $k \in [2^{k+1}, n]$ and compute g^k. Then, for given g^k, $\mathcal{A}$ outputs $k' = k \mod p$. Since $k > p$, we have $k' \neq k$. So, we get $gcd(n, k - k') = p$, a factor of n. □

Remark 1. We proved that p-DLP over $\mathbb{Z}_n$ is hard to solve if and only if the factoring assumption holds by using the idea in [12]. In [12], it is proved that the one-wayness of OU scheme is intractable if and only if the factoring assumption holds. The hardness of p-DLP is equivalent to the one-wayness of OU scheme and so we have the following relations: p-DLP $\Leftrightarrow$ factoring assumption $\Leftrightarrow$ one-wayness of OU scheme.

2.3 Semantic Security

The notion of securities are firstly considered in [8, 5]. After the concept of semantic securities are announced, many general conversion methods that make a semantically secure scheme from a naive scheme are proposed in [7, 15, 3, 9].

From now, we describe one of the previous general conversion methods. By using the method, we can make a semantically secure double decryption scheme from our naive double decryption scheme. Notice that, this is just a summary of the general conversion method of Kiltz and Lee [9], so, any understanding reader who knows the method need not see this section.

General Conversion Method of Kiltz and Lee. Kiltz and Lee proposed a general construction for public key encryption schemes that are IND-CCA2 secure in the random oracle model [9]. The conversion method based on a general hard problem called as Y-computational problem (YCP). They point out that many of the most widely used cryptographic primitives, such as RSA and Diffie-Hellman, are YCP.

Definition 3. *An instance generator $\mathcal{I}_{YC}(1^k)$ for YC outputs a description of (S_1, S_2, f_1, f_2, t). Here S_1 and S_2 are sets with $|S_1| = k$, $f_1, f_2 : S_1 \rightarrow S_2$ are functions and $t : S_2 \rightarrow S_2$ is a (trapdoor) function such that for all $x \in S_1$, $t(f_1(x)) = f_2(x)$. The functions f_1, f_2 and t should be easy to evaluate and it should be possible to sample efficiently from S_1. Let $\mathcal{A}$ be an adversary and define*

$$Adv_{\mathcal{A},\mathcal{I}_{YC}}(1^k) = Pr\begin{bmatrix} (S_1, S_2, f_1, f_2, t) \leftarrow \mathcal{I}_{YC}(1^k); \quad x \in S_1; \\ f_2(x) \leftarrow \mathcal{A}(S_1, S_2, f_1, f_2, f_1(x)); \end{bmatrix}.$$

We define the advantage function $Adv_{\mathcal{I}_{YC}}(1^k, t) = max\{Adv_{\mathcal{A},\mathcal{I}_{YC}}(1^k)\}$ where the maximum is taken over all adversaries that run for time t. We say that YCP is hard for $\mathcal{I}_{YC}(1^k)$ if t being polynomial in k implies that the advantage function $Adv_{\mathcal{I}_{YC}}(1^k, t)$ is negligible in k.

Under YCP in the random oracle model, Kiltz and Lee propose a general construction of an IND-CPA secure cryptosystem. The conversion model is composed as following: $\mathcal{E}_{pk}(m, r) = (f_1(r), E_\kappa(m))$ where E is symmetric encryption function and $\kappa = G(f_2(r))$ where G is an hash function. Let the converted encryption scheme as Π_0.

By applying the conversion method in [7], they convert the cryptosystem Π_0 to a cryptosystem Π_1 that is IND-CCA2 secure in the random oracle model. The converted IND-CCA2 secure scheme is composed as following: $\mathcal{E}_{pk}(m, r) = (f_1(H(m||r)), E_\kappa(m||r))$. where E is a symmetric encryption function, H is an hash function and $\kappa = G(f_2(H(m||r)))$ where G is an hash function.

2.4 User's Privacy Against Authority

In general, a malicious authority of a system is not distinguished from other adversaries. However, in some cases, the authority has more information than other players. Hence, it needs to distinguish the malicious authority from other adversaries. The authority in the system of the previous double decryption scheme also has such an information, the factoring. So, the authority can decrypt any ciphertext without the consent of an user by using the factoring information. As a result, any user can not expect a privacy against the authority. For this reason, we define the privacy of an encryption scheme against a malicious authority as the one-wayness and semantic security against the authority.

Definition 4. *(One-Wayness against Authority) Suppose that there is a system with an authority. Let s_{sp} be a secret system parameter only known to the authority and p_{sp} be a public system parameter known to all users in the system. Let $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be an encryption scheme of an user. Let $\mathcal{A}$ be a malicious authority that breaks the one-wayness of the scheme, then the advantage of $\mathcal{A}$ is defined as following:*

$$Adv^{ow-atk}_{\mathcal{A},\Pi,s_{sp}}(1^k) = Pr\left[\begin{array}{c}\mathcal{A}(C, p_k, s_{sp}) \\ = M\end{array}\middle|\begin{array}{c}(s_k, p_k) \leftarrow \mathcal{K}(p_{sp});\ M \leftarrow \{0,1\}^n; \\ C \leftarrow \mathcal{E}_{sk}(M);\end{array}\right].$$

Then the scheme Π is one-way against the authority if and only if there exists a negligible function $negl(\cdot)$ such that for sufficiently large k,

$$Adv^{ow-atk}_{\mathcal{A},\Pi,s_{sp}}(1^k) \leq negl(k).$$

Definition 5. *(Semantic Security against Authority) Under the same condition of Definition 4, let $\mathcal{A}$ be the authority that breaks the semantic security of the scheme, then the advantage of $\mathcal{A}$ is defined as following:*

$$Adv^{ind-atk}_{\mathcal{A},\Pi,s_{sp}}(1^k) = 2Pr\left[\begin{array}{c}\mathcal{A}(c, p_k, s_{sp}) \\ = b\end{array}\middle|\begin{array}{c}(s_k, p_k) \leftarrow \mathcal{K}(p_{sp});\ M_0, M_1 \leftarrow \{0,1\}^n; \\ b \leftarrow \{0,1\}; \qquad C \leftarrow \mathcal{E}_{sk}(M_b);\end{array}\right] - 1.$$

Then the scheme Π is semantically secure against the authority if and only if there exists a negligible function $negl(\cdot)$ such that for sufficiently large k,

$$Adv^{ind-atk}_{\mathcal{A},\Pi,s_{sp}}(1^k) \leq negl(k).$$

It is easy to grasp the notions, since the notions are defined based on the existing notions of one-wayness and semantic security. A difference of these notions and the existing notions is the information given to the adversary. If we think the secret system parameters s_{sp} as a public information, one-wayness and semantic security against the authority are same as the previous notion of securities against the ordinary adversary. So, in this case, only the secret key of a user s_k is secret information which does not given to the authority. Since the notion of securities against the authority can be seen as the previous notions, the relation of notion of securities [2] are holds equivalently.

In the previous scheme [1], the modulo n and the generator g is public system parameters while the prime factors p and q are the secret system parameters. Sometimes, the public system parameters are duplicated with the public key information of an user. For example, the generator g is public system parameter and also the public key information of all users. However, the secret system parameters are not duplicated and remains secret to the public, so the duplication is not a matter for our definition.

3 A New Double Decryption Scheme

3.1 Description of the Proposed Scheme

Key generation. Choose two primes p and q ($|p| = |q| = k$), and let $n = p^2q$.

Choose a random g in $\mathbb{Z}_n^*$ such that the order of $g^{p-1} \mod p^2$ is p. Choose a random $k-1$ bit a and compute $h = g^a \mod n$. The public key is (n, g, h, k) while the corresponding secret key is a. Two prime factors, p and q, are superkey-like secret key. Only the authority knows the factoring of n.

Encryption. Given a message $m \in \mathbb{Z}_n$, choose a random $k-1$ bit r. Then the ciphertext is computed as $C = (A, B)$ where $A = g^r \mod n$ and $B = h^r m \mod n$.

Decryption 1. First trapdoor is similar to that of ElGamal scheme. With the knowledge of a, one can decrypt m as following: $m = B/A^a \mod n$.

Decryption 2. Second decryption mechanism depends on the factoring information of n. Firstly, compute $h_p = h^{p-1} \mod p^2$. Then the secret value a is computed as following: $a = L(h_p)/L(g_p) \mod p$. Compute $A^a \mod n$ with a, then m is recovered by the following computation: $m = B/A^a \mod n$.

3.2 Security Analysis of the Proposed Scheme

One-Wayness. Our scheme is broken if one can solves p-DHP or p-DLP. In general, DLP is hard to solve than DHP. Therefore, it is sufficient to show that the one-wayness of our scheme is equivalent to the hardness of p-DHP.

Theorem 2. *Our double decryption scheme is one-way if and only if the p-DHP is hard.*

Proof. Assume that the p-DHP is not hard. Then there exists a polynomial time algorithm $\mathcal{B}$ which can solve p-DHP with non-negligible probability. We will construct a polynomial time algorithm $\mathcal{A}$, with help of $\mathcal{B}$, which can break the one-wayness of our scheme. Let the challenge ciphertext and public key be $(A = g^r \mod n, B = g^{ar}m \mod n)$ and (n, g, g^a), respectively. Since $\mathcal{B}$ can compute $g^{ar} \mod n$ from $(g^r \mod n, g^a \mod n)$, the corresponding plaintext is computed as $m = B/g^{ar} \mod n$. So, the scheme is not one-way.

Conversely, suppose that the proposed scheme is not one-way. Then for given ciphertext, an adversary $\mathcal{A}$ can recover the plaintext with non-negligible probability. We can make a polynomial time adversary $\mathcal{B}$, with help of $\mathcal{A}$, which can solve p-DHP. Let $(g^a \mod n, g^b \mod n)$ be a challenge pair to compute $g^{ab} \mod n$. Set (n, g, g^a) and $(A = g^b \mod n, B = g^k \mod n)$ for some $k \in \mathbb{Z}_n$ as public key data and ciphertext, respectively. Note that $k = ab+k'$ for some k'. So $g^k = g^{ab+k'} = g^{ab}g^{k'} = g^{ab}m \mod n$. Note that $m = g^{k'} \mod n$. Since the proposed scheme is not one-way, $\mathcal{A}$ can recover the corresponding plaintext m from $(A = g^b \mod n, B = g^k \mod n)$. With help of $\mathcal{A}$, $\mathcal{B}$ can compute $g^{ab} = g^k/m \mod n$ from $(g^a \mod n, g^b \mod n)$. □

Remark 2. When m is an element of $\mathbb{Z}_n$ of order $q-1$, m has no component in $\mathbb{G}_p$. In this case, the problem of inverting the encryption function for such message is not reduced to the p-DHP. However, the probability that a randomly chosen massage m has of order $q-1$ is about $\frac{1}{2^k}$. So, the problem of inverting the encryption function is completely reduced to the p-DHP except for negligible probability $\frac{1}{2^k}$.

Semantic Security. To apply the general conversion method proposed in [9], the security of a scheme has to be based on a kind of YCP. As commented in [9], DHP is a YCP. So, we can apply the conversion method to our scheme and then our scheme is semantically secure against CCA2 adversary.

Let H, G be two hash functions, then the converted encryption function is given as following: $\mathcal{E}_{pk}(m, r) = (g^{H(m||r)} \mod n, E_\kappa(m||r))$ where E is symmetric encryption function and $\kappa = G(h^{H(m||r)} \mod n)$. As commented in [9], we can enhance the efficiency by using the one-time pad as the symmetric function E (i.e., $E_\kappa(m||r) = \kappa \oplus (m||r)$).

In [9], the enhanced security of ElGamal encryption scheme is proved. According to their proof, ElGamal encryption scheme is secure against CCA2 attacker if the corresponding computational DHP is intractable and the symmetric encryption scheme is OTE[1] secure.

Since our scheme is a ElGamal type encryption scheme which is based its security on the computational p-DHP. We omit the proof of the semantic security against CCA2 attack, since the security proof for ElGamal type is given in [9].

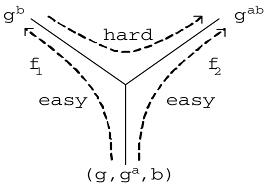

Fig. 1. Y-computational problem: the case of Diffie-Hellman problem

Theorem 3. *In the random oracle model, the converted our scheme is IND-CCA2 secure if the computational p-DHP is intractable and the symmetric encryption scheme E is OTE secure.*

3.3 Efficiency

We denote a modular exponentiation under modulo M with exponent e by $ME(|M|, |e|)$ where $|M|$ and $|e|$ are the bit length of modulo M and exponent e, respectively. Note that, the computational complexity of $ME(a, b)$ and $ME(\alpha a, \beta b)$ are in the ratio of 1 to $\alpha^2\beta$. For example, the calculation of $ME(a, b)$ is $12 = 2^2 3$ times faster than that of $ME(2a, 3b)$.

In [14], Rene Peralta claims that the hardness of factoring of 1600 bit integer of the form p^2q is equivalent to 1000 bit RSA integer. So, to compare the efficiency in same security level, we have to compare 1600 bit integer of the form p^2q with 1000 bit RSA modulo.

We compare our scheme with the previous double decryption scheme [1]. The previous scheme is semantically secure in the standard model. On the other

[1] Here, OTE means *one time encryption*. Since we use the term OTE to use the previous conversion model [9], we did not explain the detailed explanation about the OTE.

Table 1. Efficiency Comparison

Scheme	The scheme in [1]	Proposed Scheme
Plaintext	1000 bit	1600 bit
Ciphertext	4000 bit	3200 bit
Encryption	$2ME(2000, 1000)$	$2ME(1600, 533)$
Decrytion 1	$ME(2000, 1000)$	$ME(1600, 533)$
Decrytion 2	$2ME(2000, 1000)$	$ME(1066, 533) + ME(1600, 533)$

hand, our scheme is semantically secure in the random oracle model. So, we compare the efficiency of two double decryption schemes in the plain scheme point of view. Here, the term *plain scheme* means the basic model that is not transformed to CCA2 secure scheme.

Our scheme is more efficient than the previous double decryption schemes. The scheme in [1] needs 4000 bit ciphertext to guarantee the same security as 1000 bit RSA. However, our scheme needs only 3200 bit ciphertext. Moreover, the length of plaintext is larger than that of the scheme in [1]. From a computational complexity point of view, our scheme is more efficient that that of [1]. The encryption and decryption of our scheme is faster than the previous scheme about 3 times. Moreover, the cost of authority's decryption is about 4 times cheaper than the previous scheme.

4 Privacy Enhanced Double Decryption Scheme

Our double decryption scheme, proposed in the section 3, involves the same apprehension about the invasion of privacy as the previous double decryption schemes, i.e., authority can decrypt any ciphertext without a consent of an user. To solve the problem, we need some trick to restrict the ability of authority. In this section, we propose a double decryption scheme which provides a way to restrain the unlimited ability of authority.

From now, we give a detailed description of the scheme for a reader to gain a better understanding of our idea, though there are many repetition.

4.1 Description of the Privacy Enhanced Double Decryption Scheme

Key generation. Choose two primes p and q ($|p| = |q| = k$), and let $n = p^2q$. Choose a random $g \in \mathbb{Z}_n^*$ such that the order of $g^{p-1} \mod p^2$ is p. If an user allows the authority to decrypt his ciphertext, he chooses a random $k-1$ bit integer a and compute $h = g^a \mod n$. Otherwise, he choose a random t bit a where $t > k$, and compute $h = g^a \mod n$. Then the public key is (n, g, h, k) or (n, g, h, t) while the corresponding secret key is a. Two prime factors, p and q, are superkey-like secret key. Only the authority knows the factoring of n.

Encryption. Given a message $m \in \mathbb{Z}_n$, choose a random t bit r. Then the ciphertext is computed as follows: $C = (A, B)$ where $A = (g^r \mod n$ and $B = h^r m \mod n)$.

Decryption 1. With the knowledge of the secret value a, one can decrypt given a ciphertext as following: $m = B/A^a \mod n$.

Decryption 2. Second decryption mechanism depends on the factoring information of n. The authority can decrypt a ciphertext with a corresponding user's consent. If the user permits the authority to decrypt, the authority can recover the plaintext. Firstly, he computes $h_p = h^{p-1} \mod p^2$. Then the random $k-1$ bit a is computed as following: $a = L(h_p)/L(g_p) \mod p$. The authority computes $A^a \mod n$ by using a. Then m is recovered by the following simple computation: $m = B/A^a \mod n$.

Remark 3. If the user does not consent, the authority can not recovers the secret exponent a. The authority can find $a' = a \mod p$ since he knows the factoring of n. However, for sufficiently large k and t, it is hard to finding out a from a'. So the authority can not recover the plaintext for given ciphertext. The hardness of this problem will be discussed and proved in the next section.

Remark 4. When a sender want to permit the authority's ability of decryption, he choose $k-1$ bit r and compute $g^r \mod n$. Then the authority can decrypt the ciphertext generated by the sender by computing the secret exponent r though the corresponding receiver dose not consent the authority's decryption. At first, the authority computes $(g^r)_p = (g^r)^{p-1} \mod p^2$. Then the random $k-1$ bit r is computed as following: $r = L((g^r)_p)/L(g_p) \mod p$.

Remark 5. In the case of the encrypted information is not important to an user's privacy, the user will consent the authority to decrypt his ciphertext to enjoy the properties of double decryption mechanism.

4.2 Security Analysis of the Privacy Enhanced Double Decryption Scheme

Since the privacy enhanced double decryption scheme is not based its security on p-DHP, we introduce a variant of p-DHP, denoted by t-DHP. Formal definition of t-DHP is given below.

Definition 6. *(t-DHP) The t-DHP is defined as follows: Given a set $\mathbb{G}_p$, an element g of $\mathbb{G}_p$ and ($g^a \mod n$, $g^b \mod n$) for $a, b \in [1, 2^t - 1]$ where $t > k$, find $g^{ab} \mod n$ where $n = p^2q$.*

Conjecture 2. For every probabilistic polynomial time algorithm $\mathcal{A}$, there exists a negligible function $negl(\cdot)$ such that for sufficiently large t and k,

$$\Pr\left[\mathcal{A}(n, A, B) = C \;\middle|\; \begin{array}{cc} p, q \leftarrow \mathcal{P}(k); & n = p^2q; \\ g \leftarrow \mathbb{G}_p; & a, b \leftarrow_R [1, 2^t - 1]; \\ A = g^a \mod n; & B = g^b \mod n; \\ C = g^{ab} \mod n; & \end{array}\right] \leq negl(t, k).$$

Intuitively, we can say that the t-DHP is harder than p-DHP, since the exponent of t-DHP is larger than that of p-DHP. Under the hardness of t-DHP, we can prove that the privacy enhanced double decryption scheme is intractable.

Theorem 4. *The privacy enhanced double decryption scheme is one-way if and only if the t-DHP is hard to solve.*

Proof. If the t-DHP is not hard, there exists a polynomial time algorithm $\mathcal{B}$ which solves the t-DHP. By using $\mathcal{B}$, we can construct a polynomial time algorithm $\mathcal{A}$ that breaks the one-wayness of the privacy enhanced scheme. Let (n, g, g^a) be the public key parameters and let $(A = g^r \mod n, B = g^{ar}m \mod n)$ be a challenge ciphertext. Then, $\mathcal{B}$ can compute $g^{ar} \mod n$ from the pair $(g^a \mod n, g^r \mod n)$, so $\mathcal{A}$ can compute the corresponding plaintext as following: $m = B/g^{ar} \mod n$.

Suppose that the scheme is not one-way. Then there exists an adversary $\mathcal{A}$ that recovers the plaintext for given ciphertext. By using $\mathcal{A}$, we can solve the t-DHP in polynomial time. Let $(g^a \mod n, g^b \mod n)$ be a challenge. Set (n, g, g^a) as the public key parameters. Then, compute $(A = g^b \mod n, B = g^k \mod n)$ for some $k \in \mathbb{Z}_n$ and set the pair as the ciphertext which corresponds to the public key parameters. In this case, $k = ab + k'$ for some k'. Then, the algorithm $\mathcal{A}$ returns $m = g^{k'} \mod n$ as the plaintext of the ciphertext $(A = g^b \mod n, B = g^k \mod n)$ since $g^k = g^{ab+k'} = g^{ab}g^{k'} = g^{ab}m \mod n$. We can solve the t-DHP by computing $g^{ab} = g^k/m \mod n$ with help of $\mathcal{A}$. □

We have to consider the security of the privacy enhanced scheme against the authority since a malicious authority has more information than other adversary, the factoring of n. So, we define a variant of *t-DHP* named as *t_p-DHP* to formalize the security of privacy enhanced scheme against the authority.

Definition 7. *(t_p-DHP) Given a set $\mathbb{G}_p$, an element g of $\mathbb{G}_p$ and $(g^a \mod n, g^b \mod n)$ for $a, b \in [1, 2^t - 1]$ where $t > k$, compute $g^{ab} \mod n$ where $n = p^2q$ and the prime factors are known.*

We can simplify the t_p-DHP to show that the problem is sufficiently hard to solve against the authority. Consider a pair (g^a, g^b) where $g \in \mathbb{G}_p$ and $a, b \in [1, 2^t - 1]$. Let $a' = a \mod p$ and $b' = b \mod p$, then $a = a' + a''p$ and $b = b' + b''p$ for some integers a'', b''. Then, we can rewrite g^{ab} as following: $g^{ab} = g^{(a'+a''p)(b'+b''p)} = g^{a'b'+(a'b''+a''b')p+a''b''p^2} \mod n$. Note that the authority knows the factoring of n and so he can compute a' and b'. Then three values $g^{a'}$, $g^{b'}$ and $g^{a'b'}$ are easily computed. By using the values, $g^{a''p}$ and $g^{b''p}$ are computed as following: $g^a/g^{a'} = g^{a'+a''p}/g^{a'} = g^{a''p} \mod n$ and $g^b/g^{b'} = g^{b'+b''p}/g^{b'} = g^{b''p} \mod n$. Then $g^{(a'b''+a''b')p}$ is computed as following: $(g^{b''p})^{a'}(g^{a''p})^{b'} = g^{a'b''p}g^{a''b'p} = g^{(a'b''+a''b')p} \mod n$. The authority can computes $g^{a'b'}$ and $g^{(a'b''+a''b')p}$, so the following equation shows that the t_p-DHP is equal to the problem of solving the DHP for $(g^{a''p}, g^{b''p})$: $g^{ab}/g^{a'b'}g^{(a'b''+a''b')p} = g^{a''b''p^2} \mod n$. The factoring information permits the authority to compute partial information about the secret exponent when the modulo has the form of $n = p^2q$. However, if the exponent is a multiple of p, the authority can not find any information about the secret exponent. So, for the pair $(g^{a''p}, g^{b''p})$, he can not recover any information about the secret exponent, i.e., the authority can not solve the DHP for given (g^a, g^b) by computing the secret exponent, a and b.

Since the authority knows the factoring, he can reduce the problem on $\mathbb{Z}_n$ to the problem on a subgroup of $\mathbb{Z}_n$. As commented in [11], the hardness of DLP over $\mathbb{Z}_n$ is equals to the hardness of DLP over the subgroup of $\mathbb{Z}_n$. Similarly, the DHP over $\mathbb{Z}_n$ is equals to the problem of the DHP over $\mathbb{Z}_{p^2}$ and $\mathbb{Z}_q$ and we prove it in Theorem 5.

Lemma 1. *Let $n = p^2q$ where p, q are primes, then the following equation holds for some integer a: $(\alpha p^2 + \beta q)^a = (\alpha p^2)^a + (\beta q)^a \mod n$.*

Proof. Recall that, $(\alpha p^2 + \beta q)^a = \sum_{i=0}^{a} {}_aC_i(\alpha p^2)^i(\beta q)^{a-i} \mod n$. If $i \neq 0, a$ then $p^2|(\alpha p^2)^i$ and $q|(\beta q)^{a-i}$, and so $(\alpha p^2)^i(\beta q)^{a-i} = 0 \mod n$. So, we have $(\alpha p^2 + \beta q)^a = {}_aC_a(\alpha p^2)^a(\beta q)^0 + {}_aC_0(\alpha p^2)^0(\beta q)^a = (\alpha p^2)^a + (\beta q)^a \mod n$. □

Lemma 2. *Let $n = p^2q$ where p, q are primes, then the following equations hold for some integer a: $(p^2(p^{-2} \mod q))^a = p^2(p^{-2} \mod q) \mod n$ and $(q(q^{-1} \mod p^2))^a = q(q^{-1} \mod p^2) \mod n$.*

Proof. Let $l = p^2(p^{-2} \mod q) \mod n$. It suffices to show that $l^2 = l \mod n$. Note that, $l^2 = l \Leftrightarrow l^2 - l = 0 \Leftrightarrow l(l-1) = 0 \mod n$. Since $l = p^2(p^{-2} \mod q) \mod n$, $l(l-1)$ can be expressed as following; $l(l-1) = (p^2(p^{-2} \mod q))(p^2(p^{-2} \mod q) - 1) \mod n$. Then $l(l-1) = 0 \mod n$ holds since $p^2(p^{-2} \mod q) = 0 \mod p^2$ and $p^2(p^{-2} \mod q) - 1 = 0 \mod q$. □

Theorem 5. *Suppose that the factoring of $n = p^2q$ is known. Then the DHP over $\mathbb{Z}_n$ is intractable if and only if the DHP over $\mathbb{Z}_{p^2}$ and $\mathbb{Z}_q$ are intractable.*

Proof. If there exists an algorithm $\mathcal{A}$ that solves the DHP over both $\mathbb{Z}_{p^2}$ and $\mathbb{Z}_q$, then we can solve the DHP over $\mathbb{Z}_n$ by using the algorithm. Let $(g^a \mod n, g^b \mod n)$ be a challenge. Since the factoring of n is known, we can compute $(g^a \mod p^2, g^b \mod p^2)$ and $(g^a \mod q, g^b \mod q)$ from given challenge. Then, algorithm $\mathcal{A}$ computes $g^{ab} \mod p^2$ and $g^{ab} \mod q$. Since $gcd(p^2, q) = 1$, we can compute $g^{ab} \mod n$ by using the Chinese Remainder Theorem.

Conversely, if there exist an algorithm $\mathcal{B}$ that solves the DHP over $\mathbb{Z}_n$, we can solve the DHP over $\mathbb{Z}_{p^2}$ and the DHP over $\mathbb{Z}_q$ by using the algorithm. Without lose of generality, suppose that $(g^a \mod p^2, g^b \mod p^2)$ is given as a challenge where $g \in \mathbb{Z}_{p^2}^*$ is the generator. Then we can make z, x and y as following:

$$\begin{aligned} z &= (g \mod p^2)q(q^{-1} \mod p^2) + p^2(p^{-2} \mod q) \mod n, \\ x &= (g^a \mod p^2)q(q^{-1} \mod p^2) + p^2(p^{-2} \mod q) \mod n, \\ y &= (g^b \mod p^2)q(q^{-1} \mod p^2) + p^2(p^{-2} \mod q) \mod n. \end{aligned}$$

Then we compute $z^a \mod n$ and $z^b \mod n$ as following by Lemma 1 and Lemma 2:

$$\begin{aligned} z^a &= ((g \mod p^2)q(q^{-1} \mod p^2) + p^2(p^{-2} \mod q))^a \mod n \\ &= ((g \mod p^2)q(q^{-1} \mod p^2))^a + (p^2(p^{-2} \mod q))^a \mod n \\ &= (g \mod p^2)^a(q(q^{-1} \mod p^2))^a + (p^2(p^{-2} \mod q))^a \mod n \\ &= (g \mod p^2)^a q(q^{-1} \mod p^2) + p^2(p^{-2} \mod q) \mod n, \end{aligned}$$

$$\begin{aligned} z^b &= ((g \mod p^2)q(q^{-1} \mod p^2) + p^2(p^{-2} \mod q))^b \mod n \\ &= ((g \mod p^2)q(q^{-1} \mod p^2))^b + (p^2(p^{-2} \mod q))^b \mod n \\ &= (g \mod p^2)^b(q(q^{-1} \mod p^2))^b + (p^2(p^{-2} \mod q))^b \mod n \\ &= (g \mod p^2)^b q(q^{-1} \mod p^2) + p^2(p^{-2} \mod q) \mod n. \end{aligned}$$

Since $x = z^a \mod p^2$ and $x = z^a \mod q$, $x = z^a \mod n$. Similarly, $y = z^b \mod n$. We compute $z^{ab} \mod n$ by using algorithm $\mathcal{B}$ for $(z^a \mod n, z^b \mod n)$ where z is used as a generator. Then, $g^{ab} \mod p^2$ is computed as $(z^{ab} \mod n) \mod p^2 = g^{ab} \mod p^2$. We can solve the DHP over $\mathbb{Z}_q$ in the same way. □

The security of the t_p-DHP is equal to the security of the DHP over $\mathbb{Z}_{p^2}$ and $\mathbb{Z}_q$. However, if the difference between t and k is small, the t_p-DHP is not secure against the authority, since the authority can recover a' for given $g^a \mod n$ where $a = a' + a''p$ and so the remained secret information is $t - k$ bit integer a''. Hence, when $t - k$ is small, the authority can find a'' by brute-forcing. If we choose large t to make it hard to guessing a'' then the t_p-DHP is sufficiently hard to solve against the authority.

Conjecture 3. For every probabilistic polynomial time algorithm $\mathcal{A}$, there exists a negligible function $negl(\cdot)$ such that for sufficiently large t and k,

$$\Pr\left[\mathcal{A}(p,q,A,B) = C \;\middle|\; \begin{array}{ll} p, q \leftarrow \mathcal{P}(k); & n = p^2q; \\ g \leftarrow \mathbb{G}_p; & a, b \leftarrow_R [1, 2^t - 1]; \\ A = g^a \mod n; & B = g^b \mod n; \\ C = g^{ab} \mod n; & \end{array}\right] \leq negl(t,k).$$

Under the hardness of t_p-DHP, we can prove that the privacy enhanced double decryption scheme is secure against a malicious authority. Note that, to achieve sufficient security against the authority, we should use large k than the scheme proposed in Section 3. The proof of Theorem 6 is same to Theorem 4, except the hard problem, the t_p-DHP.

Theorem 6. *The privacy enhanced double decryption scheme is one-way against a malicious authority if and only if the t_p-DHP is hard to solve.*

The one-wayness of privacy enhanced scheme against the ordinary adversary and the authority is based on the t-DHP and the t_p-DHP, respectively. The t-DHP and the t_p-DHP are also YCP, so we can use the general conversion method proposed in [9] to achieve the semantic security against adaptive chosen ciphertext attack. The converted scheme of privacy enhanced scheme is as following: $\mathcal{E}_{pk}(m,r) = (g^{H(m||r)} \mod n, E_\kappa(m||r))$ where E is symmetric encryption function, H and G are two hash functions, and $\kappa = G(h^{H(m||r)} \mod n)$. Semantic security of the converted scheme is proved similar to Theorem 3.

Theorem 7. *In the random oracle model, the converted scheme is IND-CCA2 secure if the computational t-DHP is intractable and the symmetric encryption scheme E is OTE secure. Especially, the scheme is IND-CCA2 secure against the authority if the t_p-DHP is intractable.*

Remark 6. Since the authority has the factoring information, it looks like that the semantic security of the privacy enhanced scheme can be defeated by the authority. However, the previous conversion method guarantee the semantic security of a scheme if the one-wayness of the scheme is based on a kind of YCP. So, the privacy enhanced double decryption scheme is secure against IND-CCA2 adversary.

Since the privacy enhanced scheme achieves the one-wayness and the semantic security against the authority, the scheme is enhanced in the privacy point of view. If an user want to get rid of the apprehension of invasion of privacy, he will renounce the property, double decryption mechanism. However, he can choose whether to use the property or not. The property is not duty anymore in our scheme. So, we say that our scheme is enhanced in the privacy point of view rather perfectly secure against the authority.

Obviously, the security of the privacy enhanced doubled decryption scheme against the authority differs from the other adversaries. However, by choosing sufficiently large k and t, we can make the scheme achieve enough security against both the authority and ordinary adversaries.

5 Conclusion

In this paper, we have proposed an efficient public key cryptosystem with a double decryption mechanism and a modification that offers to an user more higher privacy than the previous double decryption scheme. Compared with [1], our schemes have the following advantages:

1. Efficiency: The length of ciphertext is shorter than that of the previous scheme in the same security level. Moreover, the encryption and decryption of our scheme is faster than those of the previous scheme about 3 times. Especially, the authority's decryption is faster about 4 times than that of the previous scheme.
2. Security (Privacy against the authority): The privacy enhanced double decryption scheme is secure against the authority who knows the factoring of n. In the previous scheme, the authority can recover any ciphertext by using the factoring information. However, in our scheme, the authority's excessive ability is restricted.

However, the efficiency of the basic double decryption scheme that we proposed is better than that of the previous scheme, but the privacy enhanced double decryption scheme is not efficient than that of the previous scheme. So, it is an open problem to design a double decryption scheme that raises the efficiency and enhances the privacy against the authority simultaneously.

References

1. Emmanuel Bresson, Dario Catalano, and David Pointcheval, *A Simple Public-Key Cryptosystem with a Double Trapdoor Decryption Mechanism and Its Applications*, ASIACRYPT 2003, LNCS 2894, pp. 37-54, Springer-Verlag, 2003.
2. Mihir Bellare, Anand Desai, David Pointcheval, and Phillip Rogaway, *Relations Among Notions of Security for Public-Key Encryption Schemes*, CRYPTO'98, LNCS 1462, pp. 26-46, Springer-Verlag, 1998.

3. Joonsang Baek, Byoungcheon Lee, and Kwangjo Kim, *Provably Secure Length-Saving Public-Key Encryption Scheme under the Computational Diffie-Hellman Assumption*, ETRI Journal, Volume 22, Number 4, December 2000.
4. Ronald Cramer, and Victor Shoup, *Universal Hash Proofs and a Paradigm for Adaptive Chosen Ciphertext Secure Public-Key Encryption*, EUROCRYPT 2002, LNCS 2332, pp. 45-64, Springer-Verlag, 2002.
5. D. Dolev, C. Dwork, and M. Naor, *Non-malleable cryptography*, Proceedings of the 23rd Annual Symposium on Theory of Computing, ACM, 1991.
6. W. Diffie, and M. E. Hellman, *New Directions in Cryptography*, IEEE Transactions on Informaion Theory, 22(6), 644-654, 1976.
7. Eiichiro Fujisaki, and Tatsuaki Okamoto, *How to Enhance the Security of Public-Key Encryption at Minimum Cost*, PKC'99, LNCS 1560, pp. 53-68, 1999.
8. S. Goldwasser, and S. Micali, *Probabilistic encryption*, Journal of Computer and System Science, Vol.28, No.2, pp.270-299, 1984.
9. Eike Kiltz and John Malone-Lee, *A General Construction of IND-CCA2 Secure Public Key Encryption*, Cryptography and Coding 2003, LNCS 2898, pp. 152-166, 2003.
10. Wenbo Mao, and Chae Hoon Lim, *Cryptanalysis in Prime Order Subgroups of Z_n^**, ASIACRYPT'98, LNCS 1514, pp. 214-226, 1998.
11. A.J. Menezes, P.C. Oorschot, and S.A. Vanstone, *Handbook of Applied Cryptography*, CRC Press, Inc, (1999).
12. Tatsuaki Okamoto, Shigenori Uchiyama, *A New Public-Key Cryptosystem as Secure as Factoring*, EUROCRYPT 98, LNCS 1403, pp. 308-318, Springer-Verlag, 1998.
13. Pascal Paillier, *Public-Key Cryptosystems Based on Composite Degree Residuosity Classes*, EUROCRYPT'99, LNCS 1592, pp. 223-238, Springer-Verlag, 1999.
14. Rene Peralta, *Report on Integer Factorization*, available at http://www.ipa.go.jp/security/enc/CRYPTREC/fy15/doc/1025_report.pdf, 2001.
15. David Pointcheval, *Chosen-Ciphertext Security for any One-Way Cryptosystem*, Proceedings of PKC'2000, LNCS 1751, pp. 129-146, 2000.