Analysis of Collusion-Attack Free ID-Based Non-interactive Key Sharing

Muxiang Zhang
Verizon Communications, Inc.
40 Sylvan Road, Waltham, MA 02451, USA
muxiang.zhang@verizon.com

Abstract

Recently, Tanaka proposed an identity based non-interactive key sharing scheme based on the intractability of integer factorization and discrete logarithm. The proposed identity based non-interactive key sharing scheme is similar to the well-known Maurer-Yacobi public key distribution scheme but the computational complexity for private key generation can be significantly reduced. It is also claimed that the proposed identity based non-interactive key sharing scheme is "collusion-attack free", i.e., secure against collusion attacks. In this paper, we analyze the security of the "collusion-attack free" identity based non-interactive key sharing scheme. First, we show that, without colluding with other users, a single user can recover some of the secret information of the private key generator. Then we show that a small group of users can collude to recover all of the secret information held by the private key generator. Thus, the "collusion-attack free" identity based non-interactive key sharing scheme can be completely compromised by collusion attacks.

Key words: identity based cryptosystem, non-interactive key sharing, integer factorization, discrete logarithm, collusion attack,

1 Introduction

Identity based public key cryptosystem is a paradigm proposed by Shamir [14] in 1984. In such a cryptosystem, a user's public key can be chosen as the user's identity, thus key management can be greatly simplified in comparison with certificates management in traditional public key infrastructure (PKI). Following Shamir's proposal, efficient solutions for the related notions of identity based signature and identification schemes were quickly found, e.g., [5, 6], however, identity based encryption remained to be a much more challenging problem until 2001 when Boneh and Franklin [2] proposed to use bilinear maps (the Weil or Tate pairing) over supersingular elliptic curves to achieve an elegant identity based encryption scheme.

Prior to the invention of the Boneh-Franklin identity based encryption scheme, many researchers had proposed a variety of solutions for identity based encryption, e.g., [3, 17, 15, 9, 7], most notably the Maurer-Yacobi public key distribution scheme [9]. Unfortunately, almost none of the proposed solutions were fully satisfactory. Some solutions require enormous computing power for private key generation. Other solutions require tamper resistant hardware, or they are vulnerable to user collusions. In the Maurer-Yacobi public key distribution scheme, a user's private key is the discrete logarithm of the user's identity (or a modification of user's identity) modulo a large composite number. To generate a private key, a trusted authority called private key generator (PKG) needs to solve more than two, say three discrete logarithms modulo large prime moduli p, q and r of which the product $n = pqr$ is intractable to factor, and then synthesizes the discrete logarithms using the Chinese Remainder Theorem. Hence, it is not only extremely difficult to generate each user's private key, but also the size of parameters is strongly restricted.

Recently, Tanaka proposed an identity based non-interactive key sharing scheme [16], which can be turned into an ElGamal-like [4] public-key cryptosystem. Tanaka's identity based non-interactive key sharing scheme is similar to the Maurer-Yacobi public key distribution scheme [9] including its modified version [8] and Murakami-Kasahara's scheme [11], but the private key generation is essentially different from that in the Maurer-Yacobi scheme. In Tanaka's scheme, each user's private key can be generated by solving two simple discrete logarithm problems for prime moduli p and q, respectively, and then synthesizing the discrete logarithms by a linear combination without using the Chinese Remainder Theorem, where p and q are selected so that the Pohlig-Hellman algorithm [12] assisted by the index calculus [1] can be applied easily but it is intractable to apply the Pollard's factoring algorithm [13] and its modified algorithm [18]. It is shown in [16] that the computational complexity for private key generation can be remarkably reduced in comparison with that in the Maurer-Yacobi public key distribution scheme. Moreover, it is also claimed in [16] that the proposed identity-based non-interactive key sharing scheme is "collusion attack free", i.e., secure against collusion attacks.

In this paper, we analyze the security of the "collusion-attack free" identity based non-interactive key sharing scheme proposed by Tanaka. First, we show that, without colluding with other users, a single user can recover some of the secret information of the private key generator. Then we show that a small group of users can collude to recover all of the secret information held by the private key generator. Thus, the "collusion-attack free" identity based non-interactive key sharing scheme can be completely compromised by collusion attacks. The rest of the paper is organized as follows. In Section 2, we provide an overview of Tanaka's identity based non-interactive key sharing scheme and the corresponding identity based encryption. We identify an error in the private key generation of Tanaka's scheme and provide a revision for the private key generation. In Section 3, we present collusion attacks on Tanaka's identity based non-interactive key sharing scheme and show that the identity based non-interactive key sharing scheme can be completely compromised. We conclude in section 4.

2 Non-interactive Key Sharing and Identity based Encryption

In this section, we provide a brief description of Tanaka's identity based non-interactive key sharing scheme and the corresponding identity based encryption. We also point out an error in the private key generation of Tanaka's scheme and provide a revision for the private key generation.

SET-UP: A trusted authority, called private key generator (PKG), selects two large primes p and q of about the same size such that $p-1=\alpha\gamma$ and $q-1=\beta\delta$, where α, β are large primes and γ, δ are *b-smooth* integers. The prime numbers p, q, α and β should also satisfy the condition that factoring pq and $\alpha\beta$ are computationally infeasible. Then the private key generator computes the product $n=pq$ of the selected primes p and q, determines an element g that is primitive in both multiplicative groups of integers modulo p and q, and publishes n as the system parameter.

KEY GENERATION: Given an identity ID_i of user i, the private key generator computes two integers x_i and y_i satisfying the following equations

$$ID_i^\alpha = g^{\alpha x_i} = g_\alpha^{x_i} \mod p, \tag{1}$$

$$ID_i^\beta = g^{\beta y_i} = g_\beta^{y_i} \mod q, \tag{2}$$

where $g_\alpha = g^\alpha \bmod p$ and $g_\beta = g^\beta \bmod q$. Equations (1) and (2) have unique solutions $x_i \in Z_{p-1}$ and $y_i \in Z_{q-1}$ for any ID_i because g is a primitive element in both Z_p^* and Z_q^*. Note that the order of g_α in Z_p^*, which is equal to γ, is b-smooth, the integer x_i can be computed using the Pohlig-Hellman algorithm [12]. Likewise, the integer y_i can be computed using the Pohlig-Hellman algorithm since the order of g_β in Z_q is also b-smooth. Let $\lambda(n)$ denote the order of g in Z_n^*, that is $\lambda(n)=\mathrm{lcm}(p-1,q-1)$. Also let

$$L_1 = \frac{\lambda(n)}{p-1} = \frac{\lambda(n)}{\alpha\gamma}, \tag{3}$$

and

$$L_2 = \frac{\lambda(n)}{q-1} = \frac{\lambda(n)}{\beta\delta}. \tag{4}$$

Then the private key generator computes the private key, denoted by d_i, of user i as follows

$$d_i = \alpha L_1 x_i + \beta L_2 y_i \mod \lambda(n). \tag{5}$$

The private key generator may deliver the private key d_i to user i through a secure out-of-band channel.

It should be remarked that d_i is not necessarily a discrete logarithm of ID_i (or a modification of ID_i) modulo n.

NON-INTERACTIVE KEY SHARING: Let ID_A and ID_B denote the identities of two users A and B respectively. It is stated in [16] that x_A, y_A and x_B, y_B satisfy the following equations

$$ID_B^{\alpha L_1 x_A} = ID_A^{\alpha L_1 x_B} \mod n \tag{6}$$

$$ID_B^{\beta L_2 y_A} = ID_A^{\beta L2 y_B} \mod n \tag{7}$$

Multiplying both sides of (6) and (7) results in the following equation

$$ID_B^{d_A} = ID_A^{d_B} \mod n \tag{8}$$

Equation (8) indicates that user A can share a common key, $K_{AB} = ID_B^{d_A} \bmod n$, with user B by using the identity ID_B of user B and the private key d_A of user A. Likewise, user B can share a common key, $K_{BA} = ID_B^{d_A} \bmod n = K_{AB}$, with user A by using the identity ID_A of user A and the private key d_B of user B. It is interesting to notice that the common key sharing $K_{AB} = ID_B^{d_A} \bmod n$ is similar to the decrypting process of RSA public key cryptosystem.

IDENTITY BASED ENCRYPTION: It is straightforward to turn the non-interactive key sharing scheme into identity based encryption. Assume that user A wants to send a message $m \in Z_n^*$ to user B. User A selects a random number $R, 0 < R < n$ and computes

$$\begin{aligned} C_1 &= ID_A^R \mod n \\ C_2 &= m \cdot ID_B^{R d_A} \mod n \end{aligned}$$

Then user A sends the ciphertext $C = (C_1, C_2)$ to user B. User B decrypts the ciphertext C using B's private key d_B as follows.

$$\begin{aligned} C_2 \cdot C_1^{-d_B} &= m \cdot ID_B^{d_A R} \cdot ID_A^{-d_B R} \mod n \\ &= m \cdot K_{AB}^R \cdot K_{BA}^{-R} \mod n \\ &= m. \end{aligned}$$

REMARKS: At the end of this section, we would like to point out an error in the private key generation of the non-interactive key sharing scheme as described above. In fact, equations (6) and (7) may not be satisfied. By (1) and (2), it is clear that the following two equations are satisfied.

$$ID_B^{\alpha L_1 x_A} = ID_A^{\alpha L_1 x_B} \mod p$$

$$ID_B^{\beta L_2 y_A} = ID_A^{\beta L2 y_B} \mod q$$

However, the following equations may not be satisfied.

$$ID_B^{\alpha L_1 x_A} = ID_A^{\alpha L_1 x_B} \mod q \tag{9}$$

$$ID_B^{\beta L_2 y_A} = ID_A^{\beta L2 y_B} \mod p \tag{10}$$

To explain the reason, let's assume that $\gcd(\beta, \delta) = 2$. Then $L_1 = (q-1)/2$ and $L_2 = (p-1)/2$. Also assume that the Legendre symbols of ID_A and ID_B satisfy the following conditions

$$\left(\frac{ID_A}{p}\right) = 1, \quad \left(\frac{ID_A}{q}\right) = -1, \quad \left(\frac{ID_B}{p}\right) = -1.$$

By (1),

$$\left(\frac{ID_A}{p}\right)^{\alpha} = \left(\frac{{ID_A}^{\alpha}}{p}\right) = \left(\frac{g}{p}\right)^{\alpha x_A} = 1$$

Since g is a primitive element in Z_p^*, $(\frac{g}{p}) = -1$, which implies that x_A must be even. Similar, it can be proved that x_B is an odd integer. By Euler's criterion,

$${ID_A}^{L_1} = {ID_A}^{\frac{(q-1)}{2}} = \left(\frac{ID_A}{q}\right) = -1 \mod q.$$

Hence,

$${ID_A}^{\alpha L_1 x_B} = \left(\frac{ID_A}{q}\right)^{\alpha x_B} = -1 \mod q$$

Other the other hand,

$${ID_B}^{\alpha L_1 x_A} = \left(\frac{ID_B}{q}\right)^{\alpha x_A} = 1 \mod q,$$

which indicates that equation (9) can not be satisfied. Similarly, it can be proved that equation (10) can not be satisfied, either.

It is easy to verify that equations (6) and (7) can be satisfied if $\gcd(\gamma, \delta) = 2$ and the Jocobi symbols $(\frac{ID_A}{n})$ and $(\frac{ID_B}{n})$ are equal to 1. Based on this observation, the private key generation can be revised as follows: 1) select two primes $p = 1 + \alpha\gamma$ and $q = 1 + \beta\delta$ such that $\gcd(\gamma, \delta) = 2$; and 2) replace the the identity ID_i of each user i by by $ID_i + a_i \mod n$, where a_i is the least positive integer such that $(\frac{ID_i + a_i}{n}) = 1$.

We also observe that, if L_1 and L_2 as defined by equations (3) and (4) are replaced by $L_1' = q - 1 = \beta\delta$ and $L_2' = p - 1 = \alpha\gamma$, then equations (6) and (7) are satisfied. So, the private key generation, i.e., equation (5), can be revised, alternatively, as follows

$$d_i' = \alpha\beta\delta x_i + \beta\alpha\gamma y_i \mod \lambda(n). \tag{11}$$

With the revision, user A and user B can share a common key as $ID_B^{d_A'} = ID_A^{d_B'} \mod n$.

In the following section, we show that the identity based non-interactive key sharing and encryption are vulnerable regardless of whether the private key generation is based on (5) or (11).

3 Analysis of Non-Interactive Key Sharing

In this section, we analyze the security of the non-interactive key sharing scheme described in Section 2. Dependent on whether γ is equal to δ, our analysis is carried out in two cases. In the case when $\gamma \neq \delta$, we show that a single user can factor the modulus n by exploiting the information embedded in his/her private key. In the case when s$\gamma = \delta$, we show that a small group of users can collude to factor the modulus n. After factoring modulus n, we show that two users can collude to recover the generator g. In the following, we focus our analysis on the private key generation defined by equation (5). It is straightforward to extend our analysis to the revised private key generation described by equation (11).

3.1 Factoring Modulus n

Let $\gamma' = \gamma / \gcd(\gamma, \delta)$ and $\delta' = \delta / \gcd(\gamma, \delta)$. By (5), the private key d_A of user A can be expressed as

$$d_A = \alpha\beta(\delta' x_A + \gamma' y_A) \bmod \lambda(n).$$

Let $\zeta_A = \delta' x_A + \gamma' y_A \bmod \lambda(n)$. If x_A and y_A are modeled as independent and identically distributed random variables in Z_{p-1} and Z_{q-1} respectively, then ζ_A can be treated as a random variable in $Z_{\lambda(n)}$. Thus, the probability that ζ_A is relatively prime to γ is approximately $\phi(\gamma)/\gamma$, which is greater than $1/(6 \log\log \gamma)$ (see page 65 of [10]). For simplicity, we assume that the private key of user A satisfies the conditions: $\gcd(\gamma, \zeta_A) = 1$ and $\gcd(\delta, \zeta_A) = 1$. Then for any integer a satisfying $\gcd(a, n) = 1$, Fermat's theorem implies

$$a^{\gamma d_A} = 1 \mod p, \tag{12}$$

and

$$a^{\delta d_A} = 1 \mod q. \tag{13}$$

Now, we show that modulus n can be factored based on equations (12) and (13). Depending on whether γ is equal to δ, we consider the following two cases.

Case 1: $\gamma \neq \delta$. In this case, there exist two b-smooth integers Q_1 and Q_2 such that $\gamma | Q_1$ and $\delta | Q_2$, but $\gamma \nmid Q_2$ and $\delta \nmid Q_1$, or equivalently,

$$a^{Q_1 d_A} = 1 \mod p, \quad a^{Q_2 d_A} = 1 \mod q,$$

but

$$a^{Q_1 d_A} \neq 1 \mod q, \quad a^{Q_2 d_A} \neq 1 \mod p.$$

Consequently, we have

$$p | (a^{Q_1 d_A} - 1), \quad q \nmid (a^{Q_1 d_A} - 1),$$

and

$$q | (a^{Q_2 d_A} - 1), \quad p \nmid (a^{Q_2 d_A} - 1).$$

Therefore, modulus n can be factored as

$$p = \gcd(a^{Q_1 d_A} - 1, n), \quad q = \gcd(a^{Q_2 d_A} - 1, n).$$

Next, we show that Q_1 or Q_2 can be derived using the idea of Pollard's $p-1$ factoring algorithm [13]. Let $q_1, q_2, \cdots, q_t$ denote all the primes less than or equal to b and let Q denote the least common multiple of all powers of $q_1, q_2, \cdots, q_t$ that are less than or equal to n, that is

$$Q = \prod_{i=1}^{t} q_i^{l_i},$$

where $l_i = \lfloor \log n / \log q_i \rfloor$. Then $\gamma | Q$ and $\delta | Q$, or equivalently,

$$a^{Q d_A} = 1 \mod n.$$

Since $\gamma \neq \delta$, there exists a prime number $q_i, 1 \leq i \leq t$ and an integer c_i, $1 \leq c_i \leq l_i$ such that $Q/q_i^{c_i}$ is divisible by either γ or δ, but not by both. Thus, $Q/q_i^{c_i}$ is equal to either Q_1 or Q_2. The prime number q_i and the integer c_i satisfy the following conditions

$$a^{q_i^{-c_i} Q d_A} \neq 1 \mod n, \tag{14}$$

$$\gcd(a^{Q q_i^{-c_i} d_A} - 1, n) > 1. \tag{15}$$

Based on (14) and (15), we can search for q_i and c_i for $1 \leq i \leq t$ and $1 \leq c_i \leq l_i$. we have the following algorithm for searching q_i and c_i, and consequently for factoring modulus n.

1. Select a smoothness bound b.
2. Select a random integer a, $2 \leq a \leq n-1$, and compute $z = \gcd(a, n)$. If $z \geq 2$, then return z and n/z.
3. For all prime numbers $q_1, q_2, \cdots, q_t$ less than or equal to b, compute $l_i = \lfloor \log n / \log q_i \rfloor, 1 \leq i \leq t$, and $Q = \prod_{i=1}^{t} q_i^{l_i}$.
4. For each integer i, $1 \leq i \leq t$ do the following:
   (a) $q \leftarrow q_i, l \leftarrow l_i$.
   (b) For each integer c, $1 \leq c \leq l$ do the following:
      i. $Q \leftarrow Q/q$.
      ii. Compute $z = a^{Q d_A} - 1 \bmod n$.
      iii. If $z \neq 0$ and $\gcd(z, n) > 1$, then return $\gcd(z, n)$ and $n/\gcd(z, n)$.
      iv. If $z \neq 0$ and $\gcd(z, n) = 1$, then $Q \leftarrow q \cdot Q$ and go to step 4.

Once n is factored, γ and δ can be computed by $\gamma = \gcd(Q, p-1)$ and $\delta = \gcd(Q, q-1)$. The running time of the above factoring algorithm is $O(b \ln n / \ln b)$ modular multiplications. The running time may be reduced if user A can collude with other users. Note that user A can obtain $\alpha\beta$ by computing the greatest common divisor of d_A and the private keys of other users. Once $\alpha\beta$ is known, user A can obtain $\gamma\delta$ by computing $\frac{n-1}{\alpha\beta}$, that is,

$$\begin{aligned} \frac{n-1}{\alpha\beta} &= \frac{(\alpha\gamma+1)(\beta\delta+1)-1}{\alpha\beta}, \\ &= \gamma\delta + \frac{\gamma}{\beta} + \frac{\delta}{\alpha}. \end{aligned}$$

Thus, $\gamma\delta = \lfloor \frac{n-1}{\alpha\beta} \rfloor$, under the condition that $\gamma < \beta/2$ and $\delta < \alpha/2$, which is usually true. From the prime-power factorization of $\gamma\delta$, the smoothness bound b can be precisely determined for γ and δ. Furthermore, Q can be replaced by $\gamma\delta$ in step 3 of the factoring algorithm described above. Since the prime factors of $\gamma\delta$ are a subset of those of Q, the searching time for a prime number q and an integer c satisfying $\gamma | q^{-c}\gamma\delta$ but $\delta \nmid q^{-c}\gamma\delta$ or vice versa can be reduced when Q and its prime powers are replaced by $\gamma\delta$ and the prime powers of $\gamma\delta$, respectively.

Case 2: $\gamma = \delta$. In this case, the factoring algorithm developed for Case 1 can not be used to factor modulus n since any integer divisible by γ is also divisible by δ. Consequently, for any integer v, $\gcd(a^{vd_A} - 1, n)$ is either equal to 1 or equal to n. Nevertheless, a single user, say user A again, can recover γ and δ without colluding with other users. As in Case 1, let $Q = \prod_{i=1}^{t} q_i^{l_i}$ denote the least common multiple of all powers of primes upper-bounded by b such that Q is less than or equal to n. If $q_i^{c_i}, 1 \le i \le t, 1 \le c_i \le l_i$, is a prime power in the prime-power factorization of γ, then for any integer a relatively prime to n, the following conditions are satisfied

$$a^{q_i^{c_i - l_i} Q d_A} = 1 \bmod n, \tag{16}$$

and

$$a^{q_i^{c_i - l_i - 1} Q d_A} \neq 1 \bmod n. \tag{17}$$

Based on (16) and (17), user A can search for all the prime powers in the prime-power factorization of γ and β.

To factor modulus n, however, user A needs to collude with a group of other users. First, user A obtains the product of α and β, denoted by $V = \alpha\beta$, by computing the greatest common divisor of his/her private key d_A and the private keys of other users. After $V = \alpha\beta$ is known, user A can recover γ and δ in an alternative way as $\gamma = \delta = \sqrt{\lfloor \frac{n-1}{\alpha\beta} \rfloor}$. Then user A factors $V = \alpha\beta$ by solving the following equations

$$\begin{aligned} (\alpha+\beta)\gamma &= n - \gamma^2 V - 1 \\ \alpha\beta &= V \end{aligned}$$

Once α and β are obtained, user A can factor modulus n by computing $p = \alpha\gamma + 1$ and $q = \beta\delta + 1$.

3.2 Recovering Generator g

After recovering p and q, user A can compute an element $h \in Z_n^*$ which is primitive in both Z_p^* and Z_q^*. Furthermore, user A can determine two integers $\mu_A \in Z_{p-1}$ and $\nu_A \in Z_{q-1}$ satisfying

$$ID_A^{\alpha} = h^{\alpha\mu_A} \mod p, \tag{18}$$

$$ID_A^{\beta} = h^{\beta\nu_A} \mod q. \tag{19}$$

Since g is also a primitive element in both Z_p^* and Z_q^*, there exists two integers $\sigma_1 \in Z_{p-1}$ and $\sigma_2 \in Z_{q-1}$ such that $\sigma_1 \nmid (p-1), \sigma_2 \nmid (q-1)$, and

$$g = h^{\sigma_1} \mod p, \tag{20}$$

$$g = h^{\sigma_2} \mod q. \tag{21}$$

Note that there may not exist an integer $\sigma \in Z_{\lambda(n)}$ such that $\sigma = \sigma_1 \bmod (p-1)$ and $\sigma = \sigma_2 \bmod (q-1)$. Substituting (20) and (21) into (1) and (2) and comparing with (18) and (19), we have

$$x_A = \sigma_1^{-1}\mu_A \mod (p-1),$$

and

$$y_A = \sigma_2^{-1}\nu_A \mod (q-1).$$

Thus, the private key of user A can be expressed as follows

$$d_A = \alpha L_1 \sigma_1^{-1}\mu_A + \beta L_2 \sigma_2^{-1}\nu_A \mod \lambda(n) \tag{22}$$

Likewise, the private key of user B can be described as

$$d_B = \alpha L_1 \sigma_1^{-1}\mu_B + \beta L_2 \sigma_2^{-1}\nu_B \mod \lambda(n) \tag{23}$$

Hence, user A can collude with user B to solve for σ_1 and σ_2 based on equations (22) and (23). Once σ_1 and σ_2 are obtained, the primitive element g can be recovered based on (20) and (21) by the Chinese Remainder Theorem.

After recovering all the secrets information of the private key generator, i.e., $p, q, \alpha, \beta, \gamma, \delta$ and g, user A can compute the private keys of all users. Therefore, the identity based non-interactive key sharing scheme proposed in [16] can be compromised completely.

4 Conclusion

In this paper, we analyze the security of the "collusion-attack free" identity based non-interactive key sharing scheme proposed in [16]. The proposed identity based non-interactive key sharing scheme is very similar to the the Maurer-Yacobi public key distribution scheme, but the private key generation is much more efficient than that in the Maurer-Yacobi public key distribution scheme. Unfortunately, we show in this paper that the identity based non-interactive key sharing scheme proposed in [16] is vulnerable to collusion attacks. Using the idea of Pollard's $p-1$ factoring algorithm, we show that a single user can recover some of the secret information of the private key generator without colluding with other users. We also show that a small group of users can collude to recover all of the secret information held by the private key generator. Thus, the "collusion-attack free" identity based non-interactive key sharing scheme can be completely compromised by collusion attacks.

References

[1] L. M. Adleman, "A subexponential algorithm for the discrete logarithm problem with application to cryptography", *Proceedings of the IEEE 20th Annual Symposium on Foundations of Computer Science (FOCS)*, pp. 55-60, 1979.

[2] D. Boneh and M. Franklin, "Identity-based encryption from the Weil pairing", in *Advances in Cryptology–Crypto'01 Proceedings*, Lecture Notes in Computer Science, vol. 2139, Springer-Verlag, pp. 213-229, 2001.

[3] Y. Desmedt and J. Quisquater, "Public-key systems based on the difficulty of tampering", in *Advances in Cryptology–Crypto '86 Proceedings*, Lecture Notes in Computer Science, vol. 263, Springer-Verlag, pp. 111-117, 1986.

[4] T. ElGamal, "A public key cryptosystem and signature scheme based on discrete logarithms", *IEEE Trans. Inf. Theory*, vol.IT-31, no.4, pp.469-472, July 1985.

[5] U. Feige, A. Fiat and A. Shamir, "Zero-knowledge proofs of identity", *J. Cryptology*, vol. 1, pp. 77-94, 1988.

[6] A. Fiat and A. Shamir, "How to prove yourself: Practical solutions to identification and signature problems", in *Advances in Cryptology–Crypto '86 Proceedings*, Lecture Notes in Computer Science, vol. 263, Springer-Verlag, pp. 186-194, 1986.

[7] D. Huhnlein, M. Jacobson, and D. Weber, "Towards practical non-interactive public key cryptosystems using non-maximal imaginary quadratic orders", in *Selected Areas in Cryptography*, Lecture Notes in Computer Science, vol. 2012, Springer-Verlag, pp. 275-287, 2000.

[8] C.H. Lim and P.J. Lee, "Modified Maurer-Yacobis scheme and its applications", in *Proc. Auscrypt92*, Lecture Notes in Computer Science, vol.718, pp.308-323, 1993.

[9] U. Maurer and Y. Yacobi, "Non-interactive public-key cryptography", in *Advances in Cryptology–Crypto '91 Proceedings*, Lecture Notes in Computer Science, vol. 547, Springer-Verlag, pp. 498-507, 1991.

[10] A. Menezes, P. C. van Oorschot, S. A. Vanstone, *Handbook of Applied Cryptography*, CRC Press, 1997.

[11] Y. Murakami and M. Kasahara, "An ID-based key distribution system", *IEICE Technical Report*, ISEC90-26, 1990.

[12] S. C. Pohlig and M. E. Hellman, "An improved algorithm for computing logarithms over $GF(p)$ and its cryptographic significance", *IEEE Trans. Inf. Theory*, vol.IT-24, no.1, pp.106-110, Jan. 1978.

[13] J. M. Pollard, "Theorems on factorization and primality testing", in *Proc. Cambridge Philosopher Society*, vol.76, pp.521-528, 1974.

[14] A. Shamir, "Identity-based cryptosystems and signature schemes", in *Advances in Cryptology–Crypto'84 Proceedings*, Lecture Notes in Computer Science, vol. 196, Springer-Verlag, pp. 47-53, 1984.

[15] S. Tsuji and T. Itoh, "An ID-based cryptosystem based on the discrete logarithm problem", *IEEE Journal on Selected Areas in Communication*, vol. 7, no. 4, pp. 467-473, 1989.

[16] H. Tanaka, "Collusion-attack free ID-based non-interactive key sharing", *IEICE Trans. Fundamentals*, vol. E89A, no.6, pp. 1820-1824, June 2006.

[17] H. Tanaka, "A realization scheme for the identity-based cryptosystem", in *Advances in Cryptology–Crypto '87 Proceedings*, Lecture Notes in Computer Science, vol. 293, Springer-Verlag, pp. 341-349, 1987.

[18] H. C. Williams, "A p + 1 method of factoring", *Math. Comput.*, vol.39, pp.225-234, July 1982.