How to Utilize the Randomness of Zero-Knowledge Proofs

(Extended Abstract)

Tatsuaki Okamoto *Kazuo Ohta*

NTT Communications and Information Processing Laboratories
Nippon Telegraph and Telephone Corporation
1-2356, Take, Yokosuka-shi, Kanagawa-ken, 238-03, Japan

Abstract

In zero-knowledge interactive proofs, a lot of randomized information is exchanged between the prover and the verifier, and the randomness of the prover is used in satisfying the zero-knowledge condition. In this paper, we show a new methodology that utilizes the randomness of the prover in a zero-knowledge proof for some positive objectives as well as for zero-knowledge condition. Based on this idea, we propose two types of applications; key distribution, and digital signature. We propose identity-based key distribution schemes that are provably secure against strong active attacks (*chosen-message-known-key active attacks*) assuming the difficulty of factoring a composite number. In addition, we show that *non-transitive* digital signature schemes can be constructed if and only if a one-way function exists. We also show some practical non-transitive digital signature schemes. A new general method of constructing identity-based cryptographic schemes is presented as an application of the identity-based non-transitive digital signature schemes. We also propose a new digital signature scheme based on the (extended) Fiat-Shamir identification scheme.

1. Introduction

In zero-knowledge proofs [GMRa], a lot of randomized information is exchanged between the prover and the verifier. To date, this information has been used just for the zero-knowledge interactive proof. However, many new security applications would become possible if the randomized information could be more effectively utilized.

This was first realized by Desmedt, Goutier and Bengio [DGB] who used the randomized information of the Fiat-Shamir scheme to create a subliminal channel while retaining the zero-knowledge interactive proof property. In another development, Okamoto and Ohta [OkO1] introduced the disposable zero-knowledge authentication protocol in which the provers' randomness (the number of coin flips) is restricted. The most important application of this protocol is an electronic cash system. The subliminal channel of [DGB] is used for *negative purposes* (i.e. abuse), while the protocol of [OkO1] uses randomness in a *negative manner* (i.e. restriction).

This paper propose a new methodology for utilizing randomness in a *positive manner* to achieve several *positive purposes*. Based on this methodology, we create four new cryptographic techniques: identity-based key distribution, non-transitive

A.J. Menezes and S.A. Vanstone (Eds.): Advances in Cryptology - CRYPTO '90, LNCS 537, pp. 456-475, 1991.

digital signatures, new digital signature construction using the Fiat-Shamir scheme, and a general technique for constructing identity-based schemes.

The key point of the proposed methodology is that $f(r,a)$ is used instead of the true random number R, where the distributions of $g(f(r,a))$ and $g(R)$ are indistinguishable when r is a true random number, a is a fixed parameter, and $g(R)$ is a message from a prover to a verifier using a zero-knowledge proof. Zero-knowledge proofs are still possible if the distributions of $g(f(r,a))$ and $g(R)$ are indistinguishable. The advantage of using $f(r,a)$ is that it leads to several useful functions. We show that one such function, $a^r \bmod n$, can be used to construct identity-based key distribution schemes, while other functions, $r^a \bmod n$, and bit-commitment functions [N] are appropriate for digital signature schemes.

First, we propose key distribution schemes provably secure against strong active attacks assuming the intractability of factoring. Although the recently advanced key distribution scheme of [YS] is also provably secure against active attacks under the same assumption, and is very simple, the proposed schemes are practically superior because they are identity-based schemes (or they do not need any public-key file), and, moreover, ours are provably secure against stronger active attacks (*chosen-message-known-key active attacks*) than theirs (*plain active attacks*). (Our considered active attacks, *chosen-message-known-key active attacks*, seem to be stronger than any attacks so far considered against key distribution schemes. For example, although Yacobi[Y] has also proposed a key distribution scheme provably secure against stronger passive attacks (*known-key passive attacks*) than primitive passive attacks (*plain passive attacks*), their attacks are still weaker than ours). On the other hand, although some identity-based key distribution schemes have been proposed [Ok, TI, KO, GP], after Shamir explicitly proposed identity-based schemes in 1984 [Sha], no previously published identity-based key distribution scheme has been proven secure against even weaker passive attacks. (Note that our identity-based key distribution schemes can be easily converted to regular (public-key file based) key distribution schemes with the same properties.)

Next, new *non-transitive* digital signature schemes are proposed that utilize the randomness of zero-knowledge proofs. The proposed schemes have the following properties: (we assume that user A sends a message M to user B.)

(1) Only user A can prove the validity of A's message M to any user B by using A's public key or A's identity (validity).

(2) User B cannot prove the origin of message M to another user C (non- transitivity).

With this digital signature scheme, A can validate his message to anyone, while leaving no proof of its origin. That is, the receiver cannot validate the origin of the message to anyone else. The scheme will be useful in many business and political discussions, because messages can be authenticated but they are unattributable. The concept of *non-transitive* digital signature scheme itself is not new, and the approach of using zero-knowledge proofs has been implied by Desmedt [D]. Note that the *undeniable* digital signature scheme has a similar property, but is distinctly different from the *non-transitive* signature scheme (see Section 3 in more detail). In this paper, we show that *non-transitive* digital signature scheme can be constructed

if and only if a one-way function exists. We also show some practical (identity-based) non-transitive digital signature schemes.

Using the same technique used in constructing the practical non-transitive digital signature, we construct a digital signature scheme that utilizes the randomness of the (extended) Fiat-Shamir identification scheme. Although Fiat and Shamir [FiS] have already proposed a digital signature scheme based on their identification scheme, we show another construction. The performance (data size, and processing speed) of our scheme is roughly comparable to that of Fiat and Shamir's.

Finally, we show a new general methodology for constructing identity-based cryptographic schemes, using the above-mentioned identity-based non-transitive digital signature scheme.

2. Identity-based key distribution schemes

In this section, we will show a new methodology of constructing provably secure identity-based key distribution schemes utilizing the randomness of zero-knowledge-based identification protocols such as the (extended) Fiat-Shamir scheme [FiS, FFS, GQ, Oh1, OhO].

2.1 Zero-knowledge identification protocols

Here, we introduce some of the typical zero-knowledge-based identification protocols that can be utilized to construct identity-based key distribution schemes.

(1) the Fiat-Shamir scheme [FiS, FFS]
(2) the extended Fiat-Shamir scheme 1 (higher degree version of (1)) [GQ, OhO]
(3) the extended Fiat-Shamir scheme 2 (symmetric version of (2)) [Oh1]
(4) the Beth scheme (discrete log version of (1)) [Be]

Each of the above schemes have three variations:

(a) Sequential version
(b) Parallel version (one round or three moves version)
(c) Non-interactive version

Among these variations, only sequential version (a) is zero-knowledge identification with schemes (1)-(4). The parallel version (b) of scheme (1) has been proven to be secure using no-transferable information [FFS], and the parallel versions (b) of schemes (2)-(3) have been partially proven to be secure by using no-transferable information [OhO, Oh2].

The non-interactive version (c) is constructed based on the parallel version (b) and a one-way function h as follows: Here, we assume that in the parallel version, the prover sends X to the verifier at first, then the verifier sends E to the prover, finally the prover sends Y to the verifier, then the verifier checks the validity of X, Y. In the non-constructive version, the prover generates $E = h(X)$ by himself, then, he generates Y. After generating X, E, Y, he sends them to the verifier. The check by the verifier is the same as in the parallel version. The security of this non-interactive version depends on both the property of the one-way function and the security of the parallel version. The Fiat-Shamir digital signature scheme has the same security as this non-interactive version. If we assume the function is an ideal random function [MS], the non-interactive versions (c) are provably secure when the

basic parallel versions are provably secure. Note that these non-interactive versions are different from the framework of non-interactive zero-knowledge proofs [BFM, DMP].

2.2 Identity-based key distribution schemes

In this subsection, we will introduce identity-based key distribution schemes that utilize the randomized information from the identification protocols shown in 2.1. Subsection 2.1 introduced a total of $4 \times 3 = 12$ identification protocols. Because it is tedious to write up 12 key distribution schemes (i.e. for all identification protocols), we will show 4 typical cases here; (1)(a), (2)(b), (2)(c), and (3)(c) (Here, (1)(a) means the sequential version (a) of the Fiat-Shamir scheme (1)).

2.2.1 Construction using the sequential version of the Fiat-Shamir scheme provably secure against active adversary

Key distribution scheme 2.2.1

(1) Preprocessing stage

The unique trusted center in the system generates the Fiat-Shamir scheme secret keys $s_{1,j}$ and $s_{2,j}$ $(j = 1, 2, \ldots, k)$ for user 1 and user 2, respectively. The center's secret key is (p, q), center's public key is (n, g), and $1/s_{i,j} = (f(I_i, j))^{1/2} \bmod n$ $(i = 1, 2, j = 1, 2, \ldots, k)$, where p, q are primes for which $p' = (p-1)/2$ and $q' = (q-1)/2$ are also primes, $n = pq$, the order of $g \in Z_n^*$ is $p'q'$, $|p| = c_1|n|$, $|q| = c_2|n|$ (c_1, c_2: constant). I_i is the identity of user i.

(2) Key distribution stage

Repeat steps (i) to (v) t times (for $l = 1, 2, \ldots, t$).

(i) User 1 picks a random number $r_1 \in Z_n$ and sends $x_1 = g^{2r_1} \bmod n$ to user 2.

(ii) User 2 sends a random binary vector $(e_{1,1}, \ldots, e_{1,k})$ to user 1. User 2 also picks a random number $r_2 \in Z_n$ and sends $x_2 = g^{2r_2} \bmod n$ to user 1.

(iii) User 1 sends to user 2 y_1 such that $y_1 = g^{r_1} \prod_j s_{1,j}^{e_{1,j}} \bmod n$. User 1 also sends a random binary vector $(e_{2,1}, \ldots, e_{2,k})$ to user 2.

(iv) User 2 checks that $x_1 = y_1^2 \prod_j f(I_1, j)^{e_{1,j}} \pmod n$. If the check is valid, he generates K_l such that $K_l = x_1^{r_2} \bmod n$. User 2 also sends to user 1 y_2 such that $y_2 = g^{r_2} \prod_j s_{2,j}^{e_{2,j}} \bmod n$.

(v) User 1 checks that $x_2 = y_2^2 \prod_j f(I_2, j)^{e_{2,j}} \pmod n$. If the check is valid, he generates K_l such that $K_l = x_2^{r_1} \bmod n$.

After all t procedure cycles are passed, users 1 and 2 calculate the common key K such that $K = K_1 + K_2 + \ldots + K_t \bmod n$.

Definition 1 Let $(m_1^1, m_2^1, \ldots, m_{k_1}^1)$ and $(m_1^2, m_2^2, \ldots, m_{k_2}^2)$ be the ordered set of messages sent by honest user 1 and user 2, respectively, who follow a key distribution protocol, and finally share a common key $K = h_1(p, s_1, m_1^2, \ldots, m_{k_2}^2, r_1^1, \ldots, r_{l_1}^1) = h_2(p, s_2, m_1^1, \ldots, m_{k_1}^1, r_1^2, \ldots, r_{l_2}^2)$, where h_i $(i = 1, 2)$ is the key generation function for user i, p is public information, s_i is a secret key for user i, r_j^i is a random value employed by user i to generate a message. A *plain active adversary* A interferes with the key distribution protocols between the two honest parties user

1 and user 2 in such a way that A sends a message $\widetilde{m}_j^1$ to user 2 after receiving m_j^1 from user 1, and sends $\widetilde{m}_j^2$ to user 1. Accordingly, honest user 1 computes $\widetilde{K}_1 = h_1(p, s_1, \widetilde{m}_1^2, \ldots, \widetilde{m}_{k_2}^2, r_1^1, \ldots, r_{l_1}^1)$ instead of K. The *plain active attack* is *successful* if A can finally compute $\widetilde{K}_1$.

Remark: Note that A utilizes user 2 to share a key with user 1. A may get two keys, each of which is shared with each honest user. However, this two directional attack is a specific case of the one-directional case in Definition 1.

Definition 2 A *chosen-message-known-key active adversary* A is allowed to play a role of user 2 (i.e., A knows s_2), and to know the value $K = h_1(p, s_1, m_1^2, \ldots, m_{k_2}^2, r_1^1, \ldots, r_{l_1}^1)$ after generating and sending $(m_1^2, m_2^2, \ldots, m_{k_2}^2)$ to user 1. After A is allowed to perform the above attack polynomially many times, A tries the plain active attack shown in Definition 1. The *chosen-message-known-key active attack* is *successful* if A can finally compute $\widetilde{K}_1$.

Remark: Definitions 1 and 2 correspond to *malicious adversary* and *amortized security* in [YS], respectively. We can also define two kinds of *passive* attacks; *plain passive attack* and *known-key passive attack*, which correspond to *ciphertext-only attack by a passive adversary* and *known-key attack by a passive adversary* in [Y], respectively. *Chosen-message-known-key active attack* is stronger than the other attacks including the plain active attack, and the two types of passive attacks.

Lemma 1 Let p, q be primes for which $p' = (p-1)/2$ and $q' = (q-1)/2$ are also primes, n be pq, and the order of $g \in Z_n^*$ be $p'q'$. If $r \in Z_{p'q'}$ and $R \in Z_n^*$ are randomly and uniformly selected, then $\{g^{2r} \bmod n\}$ and $\{R^2 \bmod n\}$ are perfectly indistinguishable.

Proof: In order to prove that $g^{2r} \bmod n$ and $R^2 \bmod n$ are perfectly indistinguishable, we will prove that $\{g^{2r} \bmod n\}$ is the set of quadratic residue numbers which are uniformly distributed, if $r \in Z_{p'q'}$ are randomly and uniformly selected.
First, we introduce some notations. Any x in Z_n^* can be uniquely expressed as (x_p, x_q), where $x_p = x \bmod p$ and $x_q = x \bmod q$. (Z_p^* is equivalent to $Z_p - 0$.) Any $g^r \bmod n$ can be uniquely expressed as $(g_p^{r_p} \bmod p,\ g_q^{r_q} \bmod q)$, where $g = (g_p, g_q)$, and $r_p = r \bmod p'$, and $r_q = r \bmod q'$, because the order of g is $p'q'$. We simply write $< r_p, r_q >$ for $g^r \bmod n = (g_p^{r_p} \bmod p,\ g_q^{r_q} \bmod q)$.
Then we show that there exist many g's whose orders are $p'q'$. Let $\overline{g}$ be $(\overline{g}_p, \overline{g}_q)$ such that the order of $\overline{g}_p$ in Z_p^* is $2p'$, and the order of $\overline{g}_q$ in Z_q^* is $2q'$. Then, the order of $\overline{g}$ is $2p'q'$. Any g whose order is $p'q'$ can be expressed as $\overline{g}^a \bmod n$, where $\gcd(a, 2p'q') = 2$. Therefore, the number of g whose order is $p'q'$ is $(p'q' - p' - q' + 1)$. Hence, roughly speaking, about $1/4$ of the elements in Z_n are g with the order of $p'q'$. Here note that g is quadratic residue. Next, any $g^r \bmod n$ can be represented as $< i, j >$ $(i = 0, 1, ..., p'-1;\ j = 0, 1, .., q'-1)$, because $r (r = 0, 1, .., p'q'-1)$ has a unique solution satisfying $r \equiv i \pmod{p'}$ and $r \equiv j \pmod{q'}$ by the Chinese remainder theorem. Similarly, $g^{2r} \bmod n$ can be represented as $< h, k >$ $(h = 0, 1, ..., p'-1;\ k = 0, 1, .., q'-1)$, because $r (r = 0, 1, .., p'q'-1)$ has a unique solution satisfying $2r \equiv h \pmod{p'}$ and $2r \equiv k \pmod{q'}$ (since $\gcd(2, p', q') = 1$). Therefore, the numbers of both $\{g^r \bmod n\}$ and $\{g^{2r} \bmod n\}$ are $p'q'$. The number

of quadratic residue numbers in Z_n^* is $p'q'$. Hence, any quadratic residue number (or $R^2 \bmod n$) can be expressed by $g^{2r} \bmod n$. Clearly, if $r \in Z_{p'q'}$ is randomly and uniformly selected, then $\{g^{2r} \bmod n\}$ is uniformly distributed. Thus, $\{g^{2r} \bmod n\}$ is the set of quadratic residue numbers which are uniformly distributed, if $r \in Z_{p'q'}$ are randomly and uniformly selected. **QED**

Lemma 2 Let p, q be primes for which $p' = (p-1)/2$ and $q' = (q-1)/2$ are also primes, n be pq, the order of $g \in Z_n^*$ be $p'q'$, and $|p| = c_1|n|$, $|q| = c_2|n|$ (c_1, c_2: constant). If $r \in Z_n$ and $R \in Z_n$ are randomly and uniformly selected, then $\{g^{2r} \bmod n\}$ and $\{R^2 \bmod n\}$ are statistically indistinguishable.

Proof: Here, we will prove that $\{g^{2r} \bmod n\}$ with $r \in_R Z_n$ is statistically indistinguishable from $\{g^{2r'} \bmod n\}$ with $r' \in_R Z_{p'q'}$, and that $\{R^2 \bmod n\}$ with $R \in_R Z_n$ is statistically indistinguishable from $\{R'^2 \bmod n\}$ with $R' \in_R Z_n^*$. Here, $r \in_R Z_n$ means that r is randomly and uniformly selected from Z_n. By combining the above result and Lemma 1, we can immediately obtain Lemma 2.

Because the order of g is $p'q'$, $g^{2r} \bmod n = g^{2r'} \bmod n$, where $r' = r \bmod p'q'$. The number of the elements of Z_n is $n = (2p'+1)(2q'+1) = 4p'q' + 2p' + 2q' + 1$. Then, $n/(p'q') = 4 + (2p' + 2q' + 1)/p'q'$. Therefore, when r' is a value such that $(0 \leq r' \leq 2p' + 2q')$, and $r \in Z_n$ is randomly and uniformly selected, then

$$Pr(g^{2r} \bmod n = g^{2r'} \bmod n) = 5/n.$$

When r' is a value such that $2p' + 2q' + 1 \leq r' \leq p'q' - 1$, and $r \in Z_n$ is randomly and uniformly selected, then

$$Pr(g^{2r} \bmod n = g^{2r'} \bmod n) = 4/n.$$

On the other hand, from Lemma 1, $\{g^{2r'} \bmod n\}$ is uniformly distributed on the quadratic residue set modulo n, therefore for a value $\alpha \in \{R^2 \bmod n\}$, if r' is randomly and uniformly selected, then

$$\Pr(g^{2r'} \bmod n = \alpha) = 1/(p'q')$$

Therefore, from the definition of statistical distinguishability [GMRa],

$$\sum_{\alpha \in \{0,1\}^*} |\Pr(g^{2r} \bmod n = \alpha) - \Pr(g^{2r'} \bmod n = \alpha)|$$

$$= \sum_{\alpha \in \{R^2 \bmod n\}} |\Pr(g^{2r} \bmod n = \alpha) - \Pr(g^{2r'} \bmod n = \alpha)|$$

$$= (2p' + 2q' + 1)(5/n - 1/(p'q')) + (p'q' - 2p' - 2q' - 1)(1/(p'q') - 4/n)$$

$$< 2(2p' + 2q' + 1)/n$$

Because $|p'| = c_1|n|$, $|q'| = c_2|n|$ (c_1, c_2: constant), for any constant c,

$$2(2p' + 2q' + 1)/n < 1/|n|^c.$$

Thus, $g^{2r} \bmod n$ is statistically indistinguishable from $g^{2r'} \bmod n$. Similarily, we can prove that $R^2 \bmod n$ is statistically indistinguishable from $R'^2 \bmod n$. **QED**

Theorem 1 If the factoring assumption is true, then there is no probabilistic polynomial-time *plain active attack* (Definition 1) which is successful with nonnegligible probability against the key distribution scheme 2.2.1.

(Factoring assumption) There exists no probabilistic polynomial-time algorithm F such that, given n, F computes p with non-negligible probability, where $n = pq$ (p and q are randomly and uniformly selected prime numbers).

Proof sketch: For simplicity, we assume that the Fiat-Shamir scheme's parameter k is 1. By using the technique similar to that in [FFS], we can easily extend our result to the general case that $k \neq 1$.

First, we assume that there exists a probabilistic polynomial-time plain active adversary A (Definition 1) that succeeds in sharing key K with user 1 with non-negligible probability. Here, A utilizes user 2 as a kind of oracle under the protocol condition in order to share key K with user 1. A^{U2} denotes A who utilizes user 2 under the protocol condition. A^{U2} must be verified as user 2 through the Fiat-Shamir Scheme identification t times to obtain final shared key $K = K_1 + .. + K_t \bmod n$. Since each K_i is independently determined, A and user 1 must share each K_i to share K.

First, we assume that A generates and sends a message $\tilde{x}_2$ instead of user 2's valid message x_2 *after* receiving user 1's message x_1. (This assumption is no problem because it is more advantageous than the other assumption such that A generates and sends $\tilde{x}_2$ *before* receiving x_1.) Because user 1 is a honest party, each K_i must be $\tilde{x}_2^{r_1} \bmod n$ at each round. Therefore, A^{U2} must have an algorithm H such that $H : (g, n, g^{2r_1} \bmod n) \rightarrow (\tilde{x}_2, \tilde{x}_2^{r_1} \bmod n)$. In addition, A^{U2} must have an algorithm of passing the Fiat-Shamir scheme identification as user 2 with using $\tilde{x}_2$ instead of x_2. Here, we can assume that A succeeds with nonnegligible probability. Therefore, H must output a correct answer with nonnegligible probability.

Then, we will show that A can construct an algorithm of calculating user 2's secret key by using H. First, A generates a random odd number $t \in Z_n$, and calculates $H : (g, n, g^t \bmod n) \rightarrow (\tilde{x}_2, \tilde{x}_2^{t/2} \bmod n)$ with nonnegligible probability, because $\{g^t \bmod n\}$ is statistically indistinguishable from $\{g^{2r_1} \bmod n\}$ (when t is odd and $1 \leq t \leq 2p'q' - 1$, then $t \bmod p'q'$ has all values from 0 through $p'q' - 1$). Then, A^{U2} follows the protocol with user 1. As mentioned above, A^{U2} has an algorithm to pass the identification protocol as user 2 with using $\tilde{x}_2$. Here, A can calculate $\tilde{x}_2^{1/2} \bmod n$ from $\tilde{x}_2$ and $\tilde{x}_2^{t/2} \bmod n$, because t is odd. (When we set $X = \tilde{x}_2^{1/2} \bmod n$, then $\tilde{x}_2 = X^2 \bmod n$, and $\tilde{x}_2^{t/2} \bmod n = X^t \bmod n$. Because $\gcd(t, 2) = 1$, A can calculate X from $X^2 \bmod n$ and $X^t \bmod n$.) Therefore, A can calculate one of the solutions $S = 1/f(I_2)^{1/2} \bmod n$ by $\tilde{y}_2 / \tilde{x}_2^{1/2} \bmod n$ with nonnegligible probability.

On the other hand, A interacts with two protocols with user 1 and 2. The interfaces of two protocols are the Fiat-Shamir scheme except that $x_i = \{g^{2r_i} \bmod n\}$ $(i = 1, 2)$ is used instead of $\{R^2 \bmod n\}$. As shown in Lemma 2, $x_i = \{g^{2r_i} \bmod n\}$ $(i = 1, 2)$ is statistically indistinguishable from $\{R^2 \bmod n\}$. Therefore, the protocols with user 1 and 2 are statistical zero-knowledge proofs, while the original Fiat-Shamir scheme is a perfect zero-knowledge proof. In addition, A interacts with

these two protocols in a parallel manner. We can easily prove that any parallel composition of two Fiat-Shamir scheme protocols holds the zero-knowledge property by constructing a simulator for any parallel composition of two protocols (here, note that this property is not guaranteed for general zero-knowledge protocols as shown in Theorem 3.2 in [FeS] and Theorem 7 in [GK]). Therefore, if factoring assumption is true, then A has no chance of obtaining $S = 1/f(I_2)^{1/2} \bmod n$ with nonnegligible probability from the zero-knowledge property unless A is user 2, because factoring is probabilistically polynomial-time reducible to computing S from $f(I)$ [Ra].
However, as above mentioned, if we assume that plain active adversary A succeeds in sharing key K with user 1 with nonnegligible probability, then we can show that A can obtain $S = f(I_2)^{1/2} \bmod n$ with nonnegligible probability. This is a contradiction. **QED**

Remarks: We can obtain the same result even if we replace the factoring assumption by assumption A used in [YS].
(Assumption A) Factorization of n is a one-way function [GL] with super-polynomial security.

2.2.2 Construction using the sequential version of the Fiat-Shamir scheme provably secure against stronger active adversary

In subsection 2.2.1, we have shown that key distribution scheme 2.2.1 is secure against *plain active attack*. However, it is not clear whether the scheme is secure against *chosen-message-known-key active attack* or not. In this subsection, we will propose a modified scheme which is provably secure against *chosen-message-known-key active attack*.

Key distribution scheme 2.2.2

(1) Preprocessing stage

Same as 2.2.1.(1).

(2) Key distribution stage

Repeat steps (i) to (v) t times (for $l = 1, 2, \ldots, t$).

(i) User 1 picks a random number $r_1 \in Z_n$ and sends $x_1 = g^{2r_1} \bmod n$ to user 2. *User 1 proves that s/he knows the value r_1 satisfying $x_1 = g^{2r_1} \bmod n$ using a zero-knowledge proof (see Subprotocol 2.2.2). If this zero-knowledge proof fails, user 2 halts.*

(ii) User 2 sends a random binary vector $(e_{1,1}, \ldots, e_{1,k})$ to user 1. User 2 also picks a random number $r_2 \in Z_n$ and sends $x_2 = g^{2r_2} \bmod n$ to user 1. *User 2 proves that s/he knows the value r_2 satisfying $x_2 = g^{2r_2} \bmod n$ using a zero-knowledge proof. If this zero-knowledge proof fails, user 1 halts.*

(iii) User 1 sends to user 2 y_1 such that $y_1 = g^{r_1} \prod_j s_{1,j}^{e_{1,j}} \bmod n$. User 1 also sends a random binary vector $(e_{2,1}, \ldots, e_{2,k})$ to user 2.

(iv) User 2 checks that $x_1 = y_1^2 \prod_j f(I_1, j)^{e_{1,j}} \pmod n$. If the check is valid, he generates K_l such that $K_l = x_1^{r_2} \bmod n$. User 2 also sends to user 1 y_2 such that $y_2 = g^{r_2} \prod_j s_{2,j}^{e_{2,j}} \bmod n$.

(v) User 1 checks that $x_2 = y_2^2 \prod_j f(I_2, j)^{e_{2,j}} \pmod n$. If the check is valid, he

generates K_l such that $K_l = x_2^{r_1} \bmod n$.

After all t procedure cycles are passed, users 1 and 2 calculate the common key K such that $K = K_1 + K_2 + \ldots + K_t \bmod n$.

Subprotocol 2.2.2 (Zero-knowledge proof of proving that the prover has r satisfying $x = g^{2r} \bmod n$)

(0) First, set $G = g^2 \bmod n$.

(1) Prover (P) selects a random number t in $[0, 2n-1]$, and sends $X = G^t \bmod n$ to verifier (V).

(2) V sends a random bit e in $\{0, 1\}$.

(3) If $e = 0$, P sends $Y = t$. Otherwise, P calculates $u = r + t$. Then, if u is in the interval $[n, 2n-1]$, then P sends $Y = u$ to V. If u is out of $[n, 2n-1]$, then P sends $Y = -1$.

(4) V checks the validity of P's message, if Y is not -1. Then, Y is not valid, P halts. If Y is valid or -1, P continues the procedure.

After repeating the above procedure k times, V accepts P's proof if the procedure does not halt and the number of the rounds in which $Y = -1$ is less than $\lceil (2/3)l \rceil$, where the number of the rounds in which $e = 1$ is l, and $|k| = c_1|n|$, $|l| = c_2|n|$ (C_1, c_2: constant).

Next we show that this protocol is the zero-knowledge proof of proving that the prover has r satisfying $x = G^r \bmod n$.

- (Completeness) The probability that valid P selects bad t such that u is not in $[n, 2n-1]$ is $1/2$. Therefore the probability that P selects bad t more than $\lceil (2/3)l \rceil$ times through l rounds in which $e = 1$ is l is less than $1/|n|^c$ for any constant c for sufficient large n. Therefore, valid P is accepted with overwhelming probability.
- (Soundness) If invalid P' has an algorithm A that passes the protocol, then P' can construct an algorithm M of computing r in a manner similar to Feige-Fiat-Shamir's algorithm [FFS].
- (Zero-knowledgeness) When M guesses $e = 0$, M generates $(G^Y \bmod n, 0, Y)$, where Y is in $[0, 2n-1]$. When M guesses $e = 1$, M fips a coin. If it is 0, M generates $(X = G^t \bmod n, e = 1, Y = -1)$. Otherwise, M generates $(G^Y / x \bmod n, 1, Y)$, where Y is in $[n, 2n-1]$. Then, M uses V as a black-box by checking which value of e is selected after sending X (or checking whether the guess is correct or not), and repeats the procedure.

Theorem 2 If the factoring assumption is true, then there is no probabilistic polynomial-time *chosen-message-known-key active attack* (Definition 2) which is successful with nonnegligible probability against key distribution scheme 2.2.2.

Proof sketch: Let A be a chosen-message-known-key active adversary. Suppose that A is allowed to play the role of user 2 and that A sends $\tilde{x}_2$ instead of x_2. From the soundness condition of the zero-knowledge proof of proving that A knows the value r_2 satisfying $\tilde{x}_2 = g^{2r_2} \bmod n$, A can construct a probabilistic polynomial time algorithm of calculating $K = \tilde{x}_2^{r_1} \bmod n$ by calculating $x_1^{r_2} \bmod n$ with overwhelming probability. On the other hand, we can construct a probabilistic polynomial

time algorithm of simulating the total interactive protocol between User 1 (prover) and A (verifier) through (i) to (iv) (This simulation is statistically indistinguishable from the true history of the interaction).
Therefore, in key distribution scheme 2.2.2, A can calculate any knowledge that is given through a chosen-message-known-key attack with overwhelming probability. In other words, in key distribution scheme 2.2.2, any chosen-message-known-key active adversary has the same power as a plain active adversary with overwhelming probability. Thus, because key distribution scheme 2.2.2 is secure against plain active attacks (Theorem 1), this protocol is also secure against chosen-message-known-key active attacks. **QED**

2.2.3 Construction using the parallel version of the extended Fiat-Shamir scheme 1

Key distribution scheme 2.2.3

(1) Preprocessing stage

The unique trusted center in the system generates the extended Fiat-Shamir scheme secret keys s_1 and s_2 for user 1 and user 2, respectively. Here, the center's secret key is p, q (primes), center's public key is ($n = pq$, L, g), and $1/s_i = f(I_i)^{1/L} \bmod n$ ($i = 1, 2$), where $p - 1 = Lp'$, $q - 1 = Lq'$, (p', q': prime), the order of g is $p'q'$, and I_i is an identity of user i.

(2) Key distribution stage

(i) User 1 picks a random number $r_1 \in Z_n$ and sends $x_1 = g^{Lr_1} \bmod n$ to user 2.

(ii) User 2 sends a random number $e_1 \in Z_L^*$ to user 1. User 2 also picks a random number $r_2 \in Z_n$ and sends $x_2 = g^{Lr_2} \bmod n$ to user 1.

(iii) User 1 sends to user 2 y_1 such that $y_1 = g^{r_1} \cdot s_1^{e_1} \bmod n$. User 1 also sends a random number $e_2 \in Z_L^*$ to user 2.

(iv) User 2 checks that $x_1 = y_1^L \cdot f(I_1)^{e_1} \pmod n$. If the check is valid, he generates the common key K such that $K = x_1^{r_2} \bmod n$. User 2 also sends to user 1 y_2 such that $y_2 = g^{r_2} \cdot s_2^{e_2} \bmod n$.

(v) User 1 checks that $x_2 = y_2^L \cdot f(I_2)^{e_2} \pmod n$. If the check is valid, he generates the common key K such that $K = x_2^{r_1} \bmod n$.

Remark: The security of the parallel version of the extended Fiat-Shamir scheme where $\gcd(L, p-1) \neq 1$ can be guaranteed by the *no-transferable information* technique [OhO]. Therefore, the security of this identity-based key distribution schemes can be proven in a manner similar to that used in theorem 1. Here, however, note that it is not easy to find $f(I_i)$ that is the L-th residue mod$n$. (If we do not use the identity-based system, we do not have this problem, because we can generate s_i randomly and publish $1/s_i^L \bmod n$ instead of $f(I_i)$.) On the other hand, if $\gcd(L, p-1) = \gcd(L, q-1) = 1$, we can always calculate $1/s_i = f(I_i)^{1/L} \bmod n$. However, in this case, the security cannot be guaranteed by the no-transferable information technique [OhO]. A compromise selection for the value L may be $\gcd(L, p-1) = \gcd(L, q-1) = 2$. Although, in this case, the security level for the scheme is 2 (ie., not so secure)[OhO], the security may be proven by the "wit-

ness indistinguishable" technique [FeS].
The modification of scheme 2.2.3 which is secure against chosen-message-known-key active attacks can be constructed in a manner similar to scheme 2.2.2. Here, parallel version of subprotocol 2.2.2 can be used.

2.2.4 Construction using the non-interactive version of the extended Fiat-Shamir scheme 1

Key distribution scheme 2.2.4

(1) Preprocessing stage

This stage is the same as the preprocessing stage of 2.2.3.

(2) Key distribution stage

(i) User 1 picks a random number $r_1 \in Z_n$ and generates $x_1 = g^{Lr_1} \bmod n$, $e_1 = h(x_1) \in Z_L$, $y_1 = g^{r_1} \cdot s_1^{e_1} \bmod n$. User 1 sends e_1, y_1 to user 2.

(ii) User 2 picks a random number $r_2 \in Z_n$ and generates $x_2 = g^{Lr_2} \bmod n$, $e_2 = h(x_2) \in Z_L$, $y_2 = g^{r_2} \cdot s_2^{e_2} \bmod n$. User 2 sends e_2, y_2 to user 1.
User 2 calculates $x_1 = y_1^L \cdot f(I_1)^{e_1} \pmod n$, and checks that $e_1 = h(x_1)$. If the check is valid, he generates the common key K such that $K = x_1^{r_2} \bmod n$.

(iii) User 1 calculates $x_2 = y_2^L \cdot f(I_2)^{e_2} \pmod n$, and checks that $e_2 = h(x_2)$. If the check is valid, he generates the common key K such that $K = x_2^{r_1} \bmod n$.

Remark: If we assume the non-interactive version of the extended Fiat-Shamir scheme 1 is secure, then we can prove the security of this key distribution scheme. The modification of scheme 2.2.4 which may be secure against chosen-message-known-key active attacks can be constructed in a manner similar to scheme 2.2.2. Here, non-interactive version of subprotocol 2.2.2 can be used.

2.2.5 Construction using the non-interactive version of the extended Fiat-Shamir scheme 2

Key distribution scheme 2.2.5

(1) Preprocessing stage

This stage is the same as the preprocessing stage of 2.2.3.

(2) Key distribution stage

(i) User 1 picks a random number $r_1 \in Z_n$ and generates $x_1 = g^{Lr_1} \bmod n$, $e_1 = h(x_1) \in Z_L^*$, $y_1 = g^{r_1 e_1} \cdot s_1 \bmod n$. User 1 sends x_1, y_1 to user 2.

(ii) User 2 picks a random number $r_2 \in Z_n$ and generates $x_2 = g^{Lr_2} \bmod n$, $e_2 = h(x_2) \in Z_L$, $y_2 = g^{r_2 e_2} \cdot s_2 \bmod n$. User 2 sends x_2, y_2 to user 1.
User 2 calculates $e_1 = h(x_1)$, and checks that $x_1^{e_1} = y_1^L \cdot f(I_1) \pmod n$. If the check is valid, he generates the common key K such that $K = x_1^{r_2} \bmod n$.

(iii) User 1 calculates $e_2 = h(x_2)$, and checks that $x_2^{e_2} = y_2^L \cdot f(I_2) \pmod n$. If the check is valid, he generates the common key K such that $K = x_2^{r_1} \bmod n$.

Remarks:

1. As in the remark given in 2.2.4, if we assume the non-interactive version of the extended Fiat-Shamir scheme 2 is secure, then we can prove the security of

this key distribution scheme. We can also construct the modification of scheme 2.2.4 which may be secure against chosen-message-known-key active attacks can be constructed in a manner similar to scheme 2.2.2.

2. Note that this scheme correponds to the second key distribution scheme proposed by Okamoto [Ok] (although, in his original scheme, e_i was a constant value, he later changed e_i into $h(x_i)$ thus duplicating the above-mentioned scheme). In practice, the scheme given in 2.2.4 is superior to that of 2.2.5 (the Okamoto scheme), because the transmission amount in 2.2.4 is almost half of that required in 2.2.5. That is, in 2.2.4, each user sends e_i and y_i, while in 2.2.5 each user sends x_i and y_i. The sizes of x_i and y_i are almost the size of n, while e_i is much shorter than n. For example, when the size of n is 512 bits and the size of e_i is 20 bits, in 2.2.4 each user sends 532 bits, while in 2.2.5 each user must send 1024 bits.

3. Non-transitive digital signature scheme

3.1 Definition

In this section, we propose new *non-transitive digital signature* schemes through the utilization of the randomness of zero-knowledge proofs. This non-transitive digital signature scheme has the following properties: (we assume that user A sends a message M to user B.)

(1) Only user A can prove the validity of A's message M to any user B (validity).
(2) User B cannot prove the origin of the messgage M to another user C (non-transitivity).

We will compare this non-transitive digital signature with a regular digital signature scheme, a message authentication scheme [I]. We will also show the difference from a *undeniable* digital signature scheme [CA, C] after explaining the application of a non-transitive digital signature.

In a digital signature scheme,

(1) (Same as non-transitive digital signature scheme)(validity)
(2) User B can also prove the origin of the message M to any user C (transitivity).

On the other hand, in a message authentication scheme,

(1) User B can prove the validity of message M to C while pretending to be A. (non-validity).
(2) (Same as non-transitive digital signature scheme) (this is from (1)) (non- transitivity).

Note that message authentication is aimed to prevent message interception and alternation between A and B, not to protect against user B.

We can consider the same security criterion for a non-transitive digital signature as that for a regular digital signature. Therefore, the most hopeful security criterion for a non-transitive digital signature is security against existential forgery under adaptive chosen message attack [GMRi, NY, Ro].

What are some of the applications of this non-transitive digital signature scheme? Many very sensitive negotiations, both political and business, are held every day.

The scheme allows the negotiations to proceed while protecting the privacy (irresponsibility) of all parties. Consider a government officer who is privy to information of public interest but who will unduly suffer if he is identified as the source of the information. The officer can pass the information on to the press unattributively through this non-transitive digital signature. The press can have confidence in the accuracy of the information and its source (from the validity) but the source retains his anonymity (from the non-transitivity). In addition, when we conduct sensitive negotiations for a contract, this kind of privacy (irresponsibility) is often required before concluding the contract. We will also show another application of the non-transitive digital signature in Section 5.

Then, can we use an undeniable digital signature scheme [C, CA] as a non-transitive digital signature? Although it seems that an undeniable digital signature scheme has the property of the non-transitive digital signature scheme, here, we will show that the answer to this question is negative.

In an undeniable signature scheme, the signer issues a signature z (e.g., $z = m^x \bmod n$, where x is his secret key), then he proves the validity of z interactively. Therefore, in this scheme, z can be used as evidence in some situations. The property that z is left with the related message m is an advantage of this scheme in some applications; for example, when a receiver of the undeniable signature wants to be able to confirm the signer's responsibility.

However, suppose that the undeniable digital signature [C, CA] is used for the above-mentioned example, where an officer sends confidential information to the press. After the information is published, the officer may be suspected as the source. In this case, the officer leaves z as well as m. If the goverment obtains z along with m, the goverment can force the officer to reveal his secret key x in order to clear his suspicion. Of course, the officer can refuse to reveal his secret key, but his refusal itself will become an implicit evidence that he is the source. Instead, if the officer uses a disavowal protocol [C], he can prevent his suspicion from falling on him without revealing his secret key. Therefore, the officer has no other reason for refusing a disavowal protocol except that he is the source. Thus, in the undeniable signature scheme, anyone with (m, z) can check whether a suspected signer is the true signer or not. That is, the undeniable signature is not non-transitive.

On the other hand, we can construct a non-transitive digital signature scheme by using the "symmetric public-key encryption" [GHY]. That is, a non-transitive signature signer sends a message encrypted by a symmetric public-key encryption. For the receiver to check the validity of the message, the signer must embed redundant information in the message such as an error correcting code. However, this scheme is an indirect solution of constructing a non-transitive signature scheme, because the security essentially depends on the property of the redundant information. (This scheme corresponds to a message authentication scheme based on the combination of a conventional encryption and redundant information.)

Here we will propose several implementations of the non-transitive digital signature scheme, utilizing the randomness of zero-knowledge proofs. First, we will show a general result about the non-transitive digital signature scheme. That is, we

will show that a non-transitive digital signature scheme secure against existential forgery under adaptive chosen message attack can be constructed if and only if a one-way function exists. Next, we will show some practical constructions of non-transitive signature schemes based on the zero-knowledge identification schemes shown in Subsection 2.1, although they have not been proven to be secure. For the same reason as described in Subsection 2.2, we will just show 2 typical cases here; (1)(a) and (2)(b). Here, note that the construction using the non-interactive version of these protocols ((1)-(4)(c)) cannot constitute a non-transitive digital signature; in fact, they are regular (transitive) digital signatures. We will discuss these digital signature protocols in Section 4.

3.2 Construction using one-way function

In this subsection, we show a general result about the existence of a non-transitive digital signature scheme. The key techniques for this result are zero-knowledge proof and bit-commitment.

Theorem 3 A non-transitive digital signature scheme that is secure against existential forgery under adaptive chosen message attack can be constructed if and only if a one-way function exists.

Proof sketch:

The proof of the "only if" part is almost trivial, and can be proven in the same way as shown in [Ro]. Then, we will prove the "if" part. We assume that a one-way function exists. Let signer's secret key be s and its public key be $p = f(s)$, f is a one-way function. Then, we can construct a zero-knowledge proof of proving that the signer has s satisfying $p = f(s)$ because we assume the existence of a one-way function [BCC, Blu, GMW, FFS, ILL, H, N]. In this zero-knowledge proof, we use Naor's construction of bit-commitment using a one-way function (pseudo-random generator) [N]. Here, we use $g(m_i, W)$ instead of S, where S is a random value used for the bit-commitment, g is the bit-commitment function, and W is a random value. If $g(m_i, W)$ is indistinguishable from S, we can use $g(m_i, W)$ instead of S without losing the property of the bit-commitment. Next, we will show the algorithm of the bit-commitment part in more detail. Let $g : b_{<1>}, S_{<n>}, R_{<3n>} \rightarrow d_{<3n>}$ be Naor's bit-commitment function (Section 3 in [N]), where the verifier (Bob) sends $3n$ random bits, $R_{<3n>}$, and the commiter (Alice) generates n random bits, $S_{<n>}$, and sends the bit-commit $d_{<3n>} = g(b_{<1>}, S_{<n>}, R_{<3n>})$ of a bit $b_{<1>}$ to the verifier, and at the reveal stage the commiter opens $S_{<n>}$ to the verifier. Here, the suffix of each parameter means the bit size of the parameter, and a parameter written by a capital letter is true random bits. n is the size of the key.

Let $m^{(l)}_{<1>}$ be the l-th bit of a message to be signed. Hereafter, we simply write $m^{(l)}_{<1>}$ by $m_{<1>}$. In our signature scheme, bit commitment is executed as follows: First, the verifier sends $9n$ random bits $R_{<9n>}$ and $3n$ random bits $T_{<3n>}$ to the signer. The signer generates n random bits, $U_{<n>}$. Then the signer calculates $v_{<3n>} = g((m_{<1>}, U_{<n>}, T_{<3n>})$, and $d_{<9n>} = g(b_{<1>}, v_{<3n>}, R_{<9n>})$. The signer sends $d_{<9n>}$ regarding a bit $b_{<1>}$ and $m_{<1>}$ to the verifier. At the reveal stage, the signer opens $U_{<n>}$ to the verifier.

In the zero-knowledge proof [BCC, Blu, GMW, FFS], the revealed messages are determined by the verifier's message (coin flips). Therefore, the signer embeds his messages in the bit-comitmment dupulicatedly. That is, first the prover sends committed messages X_1 and X_2. Then, the prover reveals either X_1 or X_2 depending on the verifier's message (0 or 1). Here, the prover embeds a message to be signed M into both X_1 and X_2.

This bit-commitment clearly satisfies the criterion of the bit-commitment, because if the verifier can distinguish between $g(b_{<1>}, v_{<3n>}, R_{<9n>})$ and $g(b_{<1>}, V_{<3n>}, R_{<9n>})$ ($V_{<3n>}$ is true random bits) with nonnegligible probability, he must be able to distinguish between true random bits and Naor's bit-commitment sequence. Moreover, from the property of the bit-commitment, message bit $m_{<1>}$ cannot be changed by anyone after sending $d_{<9n>}$.

By this protocol, the signer can sign a message whose size is almost half of the size of committed bits that are necessary for the zero-knowledge proof of proving that the signer has s satisfying $p = f(s)$. Therefore, when the key size is n, $O(n^c)$ bits can be signed as a message, where c is a constant.

From the property of zero-knowldge proof, the above scheme clearly satisfies the non-transitivity, because anyone can make the history of the non-transtive signature (or the interaction between the signer and the verifier). The above scheme also satisfies the security condition from the zero-knowledge property and Naor's bit-commitment's property. **QED**

Remark: More efficient bit-commitment scheme (Section 4 in [N]) can be also used instead of the bit-commitment scheme used in the proof. There are various alternative ways to embed a message to be signed in the bit-commitment scheme.

3.3 Construction using the sequential version of the Fiat-Shamir scheme

Non-transitive digital signature scheme 3.3

(1) Preprocessing stage

The unique trusted center in the system generates Fiat-Shamir scheme's secret key s_j $(j = 1, \ldots, k)$ for user A. Here, the center's secret key is p, q (primes), center's public key is $n = pq$, $1/s_j = (f(I_A, j))^{1/2} \bmod n$ $(j = 1, 2, \ldots, k)$, and I_A is the identity of user A.

(2) Authentication stage

(0) User A sends A's identity I_A and A's message M to user B.

Repeat steps (i) to (iv) t times.

(i) User A picks a random number $r \in Z_n$ and sends $x = r^{2g(M)} \bmod n$ to user B. Here, function g is a one-way hash function such that $g(m)$ distributes uniformly over Z_n when $m \in dom(h)$ is selected randomly.

(ii) User B sends a random binary vector $(e_1 \ldots, e_k)$ to user A.

(iii) User A sends to user B y such that $y = r \prod_j s_j^{e_j} \bmod n$.

(iv) User B checks that $x = y^{2g(M)} \prod_j f(I_A, j)^{e_j g(M)} \pmod n$. If the check is not valid, user B quits the procedure.

After all t rounds procedures are passed, user B recognizes that M is A's valid

message.

Remarks:

1. In the above protocol, we showed the *identity-based* version of non-transitive digital signature scheme, because it is well compatible with the original Fiat-Shamir schemes. If each user generates p, q in place of the trusted center, and publishes n, I_A, then the scheme becomes a regular (not identity-based) non-transitive digital signature.
2. In this protocol, $r^{g(M)} \bmod n$ is used in place of a true random number $R \in Z_n$ in the Fiat-Shamir scheme, where $r \in Z_n$ is a true random number. If $\gcd(g(M), p-1) = \gcd(g(M), q-1) = 1$, $r^{g(M)} \bmod n$ and R are perfectly indistinguishable. If $g(M)$ distributes uniformly, $g(M)$ satisfies the above condition with overwhelming probability.

3.4 Construction using the parallel version of the extended Fiat-Shamir scheme

Non-transitive digital signature scheme 3.4

(1) Preprocessing stage

The unique trusted center in the system generates a secret key s of the extended Fiat-Shamir scheme for user A. Here, the center's secret key is p, q (primes), center's public key is $n = pq$ and L ($\gcd(L, p-1) = \gcd(L, q-1) = 1$), $1/s = (f(I_A))^{1/L} \bmod n$, and I_A is the identity of user A.

(2) Authentication stage

(0) User A sends I_A and message M to user B.

(i) User A picks a random number $r \in Z_n$ and sends $x = r^{Lg(M)} \bmod n$ to user B. Here, function g is a one-way hash function such that $g(m)$ distributes uniformly over Z_n when $m \in dom(h)$ is selected randomly.

(ii) User B sends a random number $e \in Z_L$ to user A.

(iii) User A sends to user B y such that $y = rs^e \bmod n$.

(iv) User B checks that $x = y^{Lg(M)} f(I_A)^{eg(M)} \pmod n$. If the check is valid, user B recognizes that M is A's valid message.

Remark: Same as the remarks in Subsection 3.3.

4. New construction of digital signature schemes using the (extended) Fiat-Shamir scheme

In this section, we will show a construction of the digital signature based on the Fiat-Shamir identification scheme, which is different from the Fiat-Shamir digital signature scheme. This construction uses the technique similar to that used in Section 3. Although we can construct 4 digital signature schemes using non-interactive versions of identification schemes, (1)-(4)(c), shown in Subsection 2.1, we will just one typical case here; (2)(c).

Digital signature scheme 4.

(1) Preprocessing stage

Same as the preprocessing stage in Subsection 3.4.

(2) Authentication stage

(i) User A picks a random number $r \in Z_n$ and calculates $x = r^{Lg(M)} \bmod n$, $e = h(x) \in Z_L$, $y = rs^e \bmod n$. Here, M is a message, and (e, y) is A's signature of M. Function g is a one-way hash function such that $g(M)$ distributes uniformly over Z_n when $M \in dom(h)$ is selected randomly. User A sends I_A, M, and (e, y) to user B.

(ii) User B calculates $x = y^{Lg(M)} f(I_A)^{eg(M)} \pmod n$, and checks that $e = h(x)$. If the check is valid, user B recognizes that M is A's valid message.

5. New general method for constructing identity-based schemes

Identity-based cryptographic schemes were explicitly proposed by Shamir [Sha] in 1984 as variants of public-key cryptographic schemes (Okamoto and Shiraishi [OS] also proposed the same idea independently). In the new scheme, we use each user's identity in place of his/her public-key, therefore, we need no public-key file, instead we need a trusted center that generates and distributes each user's secret-key which is based on his/her identity. Preceding Shamir's proposal, Kohnfelder implicitly proposed the identity-based scheme in 1979 [Koh], but his construction is quite different from Shamir's. Thus, there are two types of methods for constructing identity-based cryptographic schemes; one is the *general method* [Koh], and the other is the *individual method* [Sha, OS]. In a general method, we can create an identity-based scheme from any traditional public-key cryptographic scheme, however, the overhead of key length and message length is relatively larger than a well-implemented individual method. On the other hand, in the individual method, each identity-based scheme must be constructed individually. Although only one general method [Koh] has been proposed, many individual identity-based schemes have been proposed such as key distribution schemes [Blo, Ok, MI, KO, TI], and identification and signature schemes [Sha, OS, FiS, GQ, OhO]. (The key distribution schemes shown in Section 2 are also individual identity-based schemes.)

In this section, we will show a new general method that is an application of the identity-based non-transitive digital signature scheme, although this method is similar to that of [Koh]. That is, in our scheme, we use the identity-based non-transitive digital signature scheme shown in Section 3 in place of the digital signature scheme in [Koh].

First, we will introduce the previous general method [Koh]. Trusted center T publishes its public key, P_T, of a public-key digital signature scheme, and holds the corresponding secret key S_T in secret. User U creates his public key E_U and secret key D_U of an arbitrary public-key cryptographic scheme. User U sends his public key E_U to center T with his identity I_U. After checking the validity of user U, T issues the digital signature C_T of U's public key E_U along with U's identity I_U as T's certificate to U's public key and identity. After that, U always uses T's certificate C_T with his public-key E_U and identity I_U. Because anyone can check the validity of T's certificate with T's public key P_T, in this system, without any public key file, anyone can match U's identity I_U with his public key E_U.

Our method, as mentioned above, uses the non-transitive digital signature scheme. The major difference between our method and that of [Koh] is that the certificate is not issued by the trusted center (T) but by each user (U). Our method proceeds as follows. First, the user' secret key S_U is generated as described in 3.3 or 3.4 in a preprocessing operation. The user can select any public key cryptographic scheme that best suits his purpose and create his own private and public keys, D_U and E_U. He can (interactively) generate certificates of his public key E_U at any time using the identity-based non-transitive digital signature scheme described in 3.3 or 3.4. The receiver can cofirm the combination of user's identity I_U and public key E_U by verifying the certificate with the trusted center's public key P_T.

Our method is practically superior to the previous one [Koh], because in our method the user can change or create his own private and public keys, D_U and E_U, without access to the trusted center T, while in the previous method [Koh] the user must always ask the trusted center to issue truted center's certificate C_T when the user change or create his own private and public keys. This property of our method stems from the *identity-based* property of the schemes in 3.3 and 3.4.

Another merit of our method is that the user can often change his own keys, D_U, E_U, and dispose the used keys, in order to prevent abuse of these used keys, while in [Koh] the used keys may be abused by an adversary since the used keys with the certificate can be used by anyone and at any time. This property of our method stems from the *non-transitive* property of the schemes in 3.3 and 3.4.

6. Concluding remarks

In this paper, we have presented a new methodology that utilizes the randomness of the zero-knowledge proof, and have proposed two types of applications: key distribution, and digital signature. It remains a further study to prove the security of the practical schemes shown in Subsections 2.2.3, 2.2.4, 2.2.5, 3.3, 3.4, and 4.

Acknowledgements

We would like to thank David Chaum, Shinichi Kawamura, Kazue Tanaka and anonymous referees for their valuable comments.

References

[Be] T.Beth, "Efficient Zero-Knowledge Identification Scheme For Smart Cards," Eurocrypt'88 (1988)

[BCC] G.Brassard, D.Chaum, and C.Crépeau, "Minimum Disclosure Proofs of Knowledge," Journal of Computer and System Sciences, Vol.37, pp.156-189 (1988)

[Blo] R.Blom, "Non-Public Key Distribution," Crypto'82, pp.231-236 (1982)

[Blu] M.Blum, "How to Prove a Theorem So No One Else Can Claim It," ISO/ TC97/ SC20/ WG2 N73 (1986)

[BFM] M.Blum, P.Feldman and S.Micali, "Non-Interactive Zero-Knowledge and Its Applications," STOC, pp.103-112 (1988)

[C] D.Chaum "Zero-Knowledge Undeniable Signatures," Eurocrypt'90 (1990)

[CA] D.Chaum, and H. van Antwerpen, "Undeniable Signatures," Crypto'89

(1989)

[D] Y.Desmedt, "Subliminal-Free Authentication and Signature," Eurocrypt'88, pp.23-34 (1988)

[DGB] Y.Desmedt, C.Goutier and S.Bengio, "Special Uses and Abuses of the Fiat-Shamir Passport Protocol," Crypto'87 (1987)

[DH] W.Diffie, and M.Hellman, "New Directions in Cryptography," IEEE Transactions on Information Theory, IT-22, 644-654 (1976)

[DMP] A.DeSantis, S.Micali and G.Persiano, "Non-Interactive Zero-Knowledge Proof Systems with Auxiliary Language," Crypto'88 (1988)

[FeS] U.Feige and A.Shamir, "Witness Indistinguishable and Witness Hiding Protocols," STOC, pp.416-426 (1990)

[FFS] U.Feige, A.Fiat and A.Shamir, "Zero Knowledge Proofs of Identity," STOC, pp.210-217 (1987)

[FiS] A.Fiat and A.Shamir, "How to Prove Yourself," Crypto'86 (1986)

[GHY] Z.Galil, S.Harber, and M.Yung "Symmetric Public-Key Encryption," Crypto'85 (1985)

[GK] O.Goldreich, and H.Krawczyk "On the Composition of Zero-Knowledge Proof Systems," Technical Report #570 of Technion (1989)

[GL] O.Goldreich, and A.L.Levin, "A Hard-Core Predicate for All One-Way Functions," STOC'89, pp.25-32 (1989)

[GMRa] S.Goldwasser, S.Micali, and C.Rackoff, "The Knowledge Complexity of Interactive Proofs," SIAM J. Comput., 18, 1, pp.186-208 (1989). Previous version, Proc. STOC, pp291-304 (1985)

[GMRi] S.Goldwasser, S.Micali, and C.Rivest, "A Secure Digital Signature Scheme," SIAM J. Comput., 17, 2, pp.281-308 (1988).

[GMW] O.Goldreich, S.Micali, and A.Wigderson, "Proofs that Yield Nothing But their Validity and a Methodology of Cryptographic Protocol Design," FOCS, pp.174-187 (1986)

[GP] M. Girault, and J.C. Pailles, "An Identity-Based Scheme Providing Zero-Knowledge Authentication and Authenticated Key-Exchange," ISO IEC/ JTC 1/ SC 27/ WG20.2 N200 (1990)

[GQ] L.C.Guillou, and J.J.Quisquater, "A Practical Zero-Knowledge Protocol Fitted to Security Microprocessors Minimizing Both Transmission and Memory," Eurocrypto'88 (1988)

[H] J.Håstad, "Pseudo-Random Generators under Uniform Assumptions," STOC, pp.395-404 (1990)

[I] ISO "Banking- Requirements for Message Authentication (Wholesale)" ISO/ TC68/ SC2/ WG2 N191 (1987 November)

[ILL] R.Impagliazzo, L.Levin, M.Luby "Pseudo-Random Number Generation from One-Way Functions," STOC, pp.12-24 (1989)

[Koh] L.Kohnfelder, "Towards a Practical Public-Key Cryptosystems," B.S.Thesis, MIT (1979)

[KO] K.Koyama, and K.Ohta, "Identity-based Conference Key Distribution Systems," Crypto'87 (1987)

[M] K.S.McCurley, "A Key Distribution System Equivalent Factoring," J. of Cryptology, 1, 2, pp.95-106 (1988)

[MI] T.Matsumoto and H.Imai, "On the Key Distribution Problem," Crypto'87, (1987)

[N] M.Naor, "Bit Commitment Using Pseudo-Randomness," Crypto'89, (1989)

[NY] M.Naor, and M.Yung, "Universal One-Way Hash Functions and Their Cryptographic Applications," STOC, pp.33-43 (1989)

[Oh1] K.Ohta, "Efficient Identification and Signature Scheme," Electronics Letters, 24, 2, pp.115-116 (1988)

[Oh2] K.Ohta, "Encryption and Authentication Techniques for Information Security," Dr.Sci Thesis, Waseda University (1990)

[OhO] K.Ohta, and T.Okamoto, "A Modification of the Fiat-Shamir Scheme," Crypto'88 (1988)

[Ok] E.Okamoto, "Proposal for Identity-based key distribution systems," Electronics Letters, 22, 24, pp.1283-1284 (1986)

[OkO1] T.Okamoto, and K.Ohta "Disposable Zero-Knowledge Authentications and Their Applications to Untraceable Electronic Cash," Crypto'89 (1989)

[OkO2] T.Okamoto, and K.Ohta "Divertible Zero-Knowledge Interactive Proofs and Commutative Random Self-Reducible," Eurocrypt'89 (1989)

[OS] T.Okamoto, and A.Shiraishi "A Single Public-Key Authentication Scheme for Multiple Users," *Systems and Computers in Japan, 18, 10,* pp.14-24 (1987) Previous version, Technical Report of IECE Japan, IN83-92 (1984)

[Ra] M.Rabin "Digitalized Signatures and Public-Key Cryptosystems," MIT/ LCS/ TR-212, MIT Technical Report (1979)

[Ro] J.Rompel, "One-Way Functions are Sufficient for Secure Signatures," STOC, pp.387-394 (1990)

[Sha] A.Shamir, "Identity-based Cryptosystems and Signature Schemes," Crypto'84 (1984)

[Shm] Z.Shmuely, "Composite Diffie-Hellman Public-Key Generating Systems Are Hard to Break," TR #356, Computer Science Dept. Technion, IIT (1985)

[TI] S.Tsujii, and T.Itoh, "An ID-Based Crypto-system Based on the Discrete Logarithm Problem," IEEE J. Selected Area in Communications, 7, 4 (1989)

[TW] M.Tompa and H.Woll, "Random Self-Reducibility and Zero Knowledge Interactive Proofs of Possession of Information," Proc. FOCS, pp472-482 (1987)

[Y] Y.Yacobi, "A Key Distribution "Paradox"," These Proceedings (1990)

[YS] Y.Yacobi, and Z.Shmuely, "On Key Distribution Systems," Crypto'89 (1989)