New Proxy Signature, Proxy Blind Signature and Proxy Ring Signature Schemes from Bilinear Pairings

Fangguo Zhang[1], Reihaneh Safavi-Naini[1] and Chih-Yin Lin[2]

[1] School of Information Technology and Computer Science
University of Wollongong, NSW 2522 Australia
{fangguo, rei}@uow.edu.au
[2] Institute of Information Management, National Chiao Tung University
1001 University Road, Hsinchu, 300, Taiwan
lincy@iim.nctu.edu.tw

Abstract. Proxy signatures are very useful tools when one needs to delegate his/her signing capability to other party. After Mambo *et al.*'s first scheme was announced, many proxy signature schemes and various types of proxy signature schemes have been proposed. Due to the various applications of the bilinear pairings in cryptography, there are many ID-based signature schemes have been proposed. In this paper, we address that it is easy to design proxy signature and proxy blind signature from the conventional ID-based signature schemes using bilinear pairings, and give some concrete schemes based on existed ID-based signature schemes. At the same time, we introduce a new type of proxy signature – proxy ring signature, and propose the first proxy ring signature scheme based on an existed ID-based ring signature scheme.

Key words: Proxy signature, ID-based cryptography, Proxy blind signature, Proxy ring signature, Bilinear pairings.

1 Introduction

The concept of proxy signature was first introduced by Mambo, Usuda, and Okamoto in 1996 [15]. The proxy signature schemes allow proxy signers to sign messages on behalf of an original signer. After Mambo *et al.*'s first scheme was announced, many proxy signature schemes have been proposed [10, 12, 16, 24]. Proxy signatures can combine other special signatures to obtain some new types of proxy signatures. Till now, there are various kinds of proxy signature schemes have been proposed [14, 22, 25].

Proxy blind signature is an important type of proxy signature, it plays an important role in the following scenario: In e-cash system, the user makes the bank blindly sign a coin using blind signature schemes. Whenever a user goes through a valid branch to withdraw a coin, he/she needs the branch to make proxy blind signature on behalf of the signee bank. The first proxy blind signature scheme was introduced by Lin and Jan in [14]. Recently, there are two new schemes have been proposed: Tan *et al.*'s scheme [21] which is based on Schnorr blind signature scheme and Lal *et al.*'s scheme [11] which is based on Mambo *et al.*'s proxy signature scheme.

In ID-based public key cryptosystem, everyone's public keys are predetermined by information that uniquely identifies them, such as name, social security number, email address, *etc.*, rather than an arbitrary string. This concept was first proposed by Shamir [20]. In the last couple of years, the bilinear pairings have been found various applications in cryptography, they can be used to realize some cryptographic primitives that were previously unknown or impractical [1–3, 9, 19]. More precisely, they are basic tools for construction of ID-based cryptographic schemes, many ID-based cryptographic schemes have been proposed using them [2, 4, 8, 17, 19, 23]. In this paper we address that it is easy to design proxy signature and proxy blind signature from ID-based signature schemes using bilinear pairings, and give some concrete schemes.

The concept of ring signatures was formalized in 2001 by Rivest, Shamir, and Tauman [18]. A ring signature is considered to be a simplified group signature which consists of only users without managers. It protects the anonymity of a signer since the verifier knows that the signature comes from a member of a ring, but doesn't know exactly who the signer is. There is also no way to revoke the anonymity of the signer. Ring signature can support *ad hoc* subset formation and in general does not require special setup.

Before introduce a new concept of a type of signature, we consider the following scenario: An entity delegate his signing capability to many proxies, called proxy signers set. Any proxy signer can perform the signing operation of the original entity. These proxy signers want to sign messages on behalf of the original entity while providing anonymity. Of course, this problem can be solved by group signature (Take the group manger as the original entity). But in some applications, it is necessary to protect the privacy of participants (we believe that unconditional anonymity is necessary in many occasions). If the proxies don't hope that some one (include the original signer) can open their identities, the group signature is not suitable in here (Because a group manager can open the signature to reveal the identity of the signer). In this paper, we introduce a new type signature – proxy ring signature to solve this problem (In fact, it can be regarded as an untraceable group signature). At the same time, we propose a proxy ring signature scheme based on Zhang-Kim's [23] ID-based ring signature scheme from bilinear pairing. It is easy to see that proxy ring signature is very similar to group signature, a difference is the unconditional anonymity of signer (more exactly, it should be unconditionally signer ambiguous).

The rest of the paper is organized as follows: The next section introduce the definitions and security requirements of proxy signature, proxy blind signature and proxy ring signature; Section 3 briefly explains some preliminaries. Section 4 gives a description of the general construction of various types of proxy signature from ID-based public key setting using bilinear pairing. In Section 5, 6 and 7, some concrete proxy signature schemes are presented. Section 8 concludes this paper.

2 Proxy Signature, Proxy Blind Signature and Proxy Ring Signature

A proxy signature scheme consists of three entities: original signer, proxy signer and verifier. One assumes that each participant has received (via a PKI or a certificate) a public-secret key pair (**Setup**). If an original signer wants to delegate the signing capability to a proxy signer, he/she uses the original signature key to create a proxy signature key, which will then be sent to the proxy signer (**Generation of the proxy key**). The proxy signer can use the proxy signature key to sign messages on behalf of the original signer (**Proxy signature generation**). Proxy signatures can be verified using a modified verification equation such that the verifier can be convinced that the signature is generated by the authorized proxy entity of the original signer (**Verification**).

Depending on whether the original signer can generate the same proxy signatures as the proxy signers do, there are two kinds of proxy signature schemes: (1) Proxy-unprotected (the original signer can also generate the proxy signatures.); (2) Proxy-protected (anyone except proxy signer, including the original signer, cannot generate the proxy signatures).

Because the original signer can create a valid proxy signature in proxy-unprotected proxy signature scheme, this is unfair for proxy signer. So, we will focus on the proxy-protected proxy signature.

The **Generation of the proxy key** in proxy signature is a delegation procedure. There are three types of delegation in Mambo *et al.*'s paper: full delegation, partial delegation and delegation by warrant. In [10], S. Kim *et al.* gave a new type of delegation called partial delegation with warrant, which can be considered as the combination of partial delegation and delegation by warrant.

Lee *et al.* [12] defined properties that a strong proxy signature scheme should provide:

1) **Distinguishability:** Proxy signatures are distinguishable from normal signatures by everyone.
2) **Verifiability:** From the proxy signature, the verifier can be convinced of the original signer's agreement on the signed message.
3) **Strong non-forgeability:** A designated proxy signer can create a valid proxy signature for the original signer. But the original signer and other third parties who are not designated as a proxy signer cannot create a valid proxy signature.
4) **Strong identifiability:** Anyone can determine the identity of the corresponding proxy signer from the proxy signature.
5) **Strong non-deniability:** Once a proxy signer creates a valid proxy signature of an original signer, he/she cannot repudiate the signature creation.

6) **Prevention of misuse:** The proxy signer cannot use the proxy key for other purposes than generating a valid proxy signature. That is, he/she cannot sign messages that have not been authorized by the original signer.

Proxy blind signature is considered be the combination of proxy signature and blind signature, so, beside above security requirements of proxy signature, it should satisfy the additional requirements: **Blindness**, *i.e.*, the signer does not know the content of the message. In general, a proxy blind signature scheme consists of four participants: an original signer, a proxy signer, a user and a verifier, and the following five algorithms, **Setup**, **Generation of the proxy key**, **Proxy blind signature generation**, and **Verification**.

Proxy Ring Signature can be viewed as the combination of proxy signature and ring signature. It consists of three participants: an original signer, a set of proxy signers, verifier, and the following four algorithms, **Setup**, **Generation of the proxy key**, **Proxy ring signature generation**, and **Verification**. Proxy ring signature is a type of proxy signature, so it should satisfy all the requirements of general proxy signature, beside these, it should satisfy the additional requirements: **Signer ambiguity**, *i.e.*, the adversary (include the original signer) cannot tell the identity of the signer with a probability larger than $1/r$, where r is the cardinality of the ring, even assuming that he/she has unlimited computing resources.

3 Preliminaries

3.1 Bilinear Pairings

Let $\mathbb{G}_1$ be a cyclic additive group generated by P, whose order is a prime q, and $\mathbb{G}_2$ be a cyclic multiplicative group of the same order q. A bilinear pairing is a map $e : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$ with the following properties:

P1 *Bilinear:* $e(aP, bQ) = e(P, Q)^{ab}$;
P2 *Non-degenerate:* There exists $P, Q \in \mathbb{G}_1$ such that $e(P, Q) \neq 1$;
P3 *Computable:* There is an efficient algorithm to compute $e(P, Q)$ for all $P, Q \in \mathbb{G}_1$.

When the DDHP (Decision Diffie-Hellman Problem) is easy but the CDHP (Computational Diffie-Hellman Problem) is hard on the group $\mathbb{G}$, we call $\mathbb{G}$ a *Gap Diffie-Hellman* (*GDH*) *group*. Such groups can be found on supersingular elliptic curves or hyperelliptic curves over finite field, and the bilinear parings can be derived from the Weil or Tate pairing. We can refer to $[2, 4, 8]$ for more details.

Through this paper, we define the system parameters in all schemes are as follows: Let P be a generator of $\mathbb{G}_1$, the bilinear pairing is given by $e : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$. Define two cryptographic hash functions $H_1 : \{0, 1\}^* \rightarrow Z_q$ and $H_2 : \{0, 1\}^* \rightarrow \mathbb{G}_1$.

3.2 Pairing-Based Short Signature Scheme

Now we are ready to introduce Boneh *et al.*'s pairing-based short signature scheme proposed in [3], we denote BLS scheme.

Key generation:
Secret key: a random number s chosen from Z_q^*;
Public key: $(\mathbb{G}_1, \mathbb{G}_2, q, P, P_{pub}, H_2)$, here $P_{pub} = sP$.
Signing:
A message $M \in \{0,1\}^*$, $P_M = H_2(M) \in \mathbb{G}_1$, $S_M = sP_M$.
The signature of M is S_M.
Verification: Check whether the following equation holds:

$$e(S_M, P) = e(H_2(M), P_{pub}).$$

This scheme is proven to be secure against existential forgery on adaptive chosen-message attacks (in the random oracle model) assuming the CDHP is hard [3].

3.3 General Process of Conventional ID-based Signature Scheme from Pairing

ID-based public key setting involves a KGC (Key Generator Center) and users. The basic operations consists of **Setup** and **Private Key Extraction** (simply **Extract**). When we use bilinear pairings to construct ID-based signature scheme, the general process will be as follows:

- **Setup:** KGC chooses a random number $s \in Z_q^*$ and sets $P_{pub} = sP$. The center publishes system parameters $params = \{\mathbb{G}_1, \mathbb{G}_2,\ e, q, P,\ P_{pub}, H_1, H_2\}$, and keeps s as the *master-key*, which is known only by itself.
- **Extract:** A user submits his/her identity information ID to KGC. KGC computes the user's public key as $Q_{ID} = H_2(ID)$, and returns $S_{ID} = sQ_{ID}$ to the user as his/her private key.
- **Signing:** is a probabilistic polynomial time (PPT) algorithm that takes $params$, a private key S_{ID}, and a message m. The algorithm outputs a signature $\sigma(m)$ for m.
- **Verification:** is a PPT algorithm that takes $(params, ID, m, \sigma(m))$ and outputs either accept or reject.

There is a relationship between the short signature schemes and the conventional ID-based public key setting from bilinear pairing, that is the signing process in the short signature scheme can be regarded as the private key extract process in the ID-based public key setting.

We address that ID-based signature scheme with a trusted KGC can be regarded as a proxy-unprotected proxy signature scheme with multiple proxies. This is obviously: we take the KGC as the original signer, user as the proxy signer. **Extract** can be considered the **Generation of the proxy key**, this is the delegation.

4 The General Construction

About the delegation function of pairing based cryptosystems, Boneh *et. al* [2] and Chen *et. al* [6] had noted it. If using their delegation to construct proxy signature schemes directly, they are proxy-unprotected proxy signature schemes. To obtain the proxy-protected delegation, we will require the user to make a signature on the same message using BLS short signature. Assume that there are two participants, one called original signer with public key PK_o and secret key s_o, another called proxy signer with public key PK_p and secret key s_p, they have the common system parameters: $\{\mathbb{G}_1, \mathbb{G}_2, e, q, P, H_1, H_2\}$. We describe the delegation in detail as follows:

- The original signer makes a warrant w. There is an explicit description of the delegation relation in the warrant w.
- The original signer computes $So_w = s_o H_2(w)$, and sends w and So_w to proxy signer.
- The proxy signer checks if $e(So_w, P) = e(H_2(w), PK_o)$, if it is right, then computes $S_w = So_w + s_p H_2(w)$.

In fact this is the partial delegation with warrant [10]. So, it is can be regarded as the **Generation of the proxy key** in proxy signature. The proxy secret key is S_w, and the proxy public key is PK_o+PK_p. Then the proxy signer can uses any ID-based signature schemes and ID-based blind signature schemes from pairings (takes the ID public key as $H_2(w)$) and secret key as S_w, the public key of KGC as $PK_o + PK_p$) to get proxy signature and proxy blind signature schemes.

Anyone cannot forge an $S_{w'}$ of a warrant w', since the original signer and proxy signer all use BLS short signature scheme to sign warrant, and BLS short signature scheme is proven to be secure. Like the discussion in [13], above delegation need not the secure channel for the delivery of the signed warrant by the original signer, *i.e.*, the original signer can publish w and So_w. More precisely, any adversary can get the original signer's signature on warrant w. Even this, the adversary cannot get the S_w of the proxy signer, because S_w satisfies $e(S_w, P) = e(H_2(w), PK_o + PK_p)$, and $e(So_w, P) = e(H_2(w), PK_o)$, so, $e(S_w - So_w, P) = e(H_2(w), PK_p)$. This means if the adversary can get the S_w of the proxy signer, then he can forge the BLS signature of the message w with the public key PK_p of proxy signer, this is impossible due to the security of BLS scheme.

In the next 3 sections, we give some concrete proxy signature schemes using above key idea. We use essentially the existed ID-based signature, blind signature or ring signature schemes, and hence borrow them from there almost unchanged.

Note: *Before verifying any proxy signature, the verifier will check the validity of the public keys of all participants via certificates. This is to against the public key substitution attack.*

5 New Proxy Signature Schemes from Pairings

In this section, we give a new proxy signature scheme based on Hess' [8] ID-based signature scheme.

[**Setup:**]

The system parameters $params = \{\mathbb{G}_1, \mathbb{G}_2, e, q, P, H_1, H_2\}$, the original signer has public-secret key pair (PK_o, s_o), the proxy signer has public-secret key pair (PK_p, s_p).

[**Generation of the proxy key:**]

After the original signer and the proxy signer finish the process in Section 4, the proxy signer gets a proxy key S_w.

[**Proxy signature generation:**]

For any delegated message m, the proxy signer uses Hess's ID-based signature scheme [8] (takes the signing key as S_w) and obtains a signature (c_p, U_p) as follows:

- $r_p = e(P, P)^{k_p}$, $k_p \in_R Z_q^*$.
- $c_p = H_1(m||r_p)$.
- $U_p = c_p S_w + k_p P$.

The valid proxy signature will be the tuple

$$< m, c_p, U_p, w > .$$

[**Verification:**]

A verifier can accept this proxy signature if and only if

$$c_p = H_1(m||e(U_p, P)e(H_2(w), PK_o + PK_p)^{-c_p}).$$

The verification of the signature is justified by the following equations:

$$\begin{aligned}
&e(U_P, P)(e(H_2(w), PK_o + PK_p))^{-c_P} \\
&= e(c_p S_w + k_p P, P)(e(H_2(w), PK_o + PK_p))^{-c_P} \\
&= e(c_p(So_w + s_p H_2(w), P)e(k_p P, P)(e(H_2(w), PK_o + PK_p))^{-c_P} \\
&= (e(H_2(w), PK_o + PK_p))^{c_P} e(k_p P, P)(e(H_2(w), PK_o + PK_p))^{-c_P} \\
&= e(P, P)^{k_p} = r_P
\end{aligned}$$

So, we have:

$$c_P = H_1(m||r_P) = H_1(m||e(U_p, P)e(H_2(w), PK_o + PK_p)^{-c_p}).$$

Due to using the warrant m_w, it is obvious that our new proxy signature scheme satisfies the requirements stated in Section 2. but **strong non-forgeability**. On the other hand, we use Hess's ID-based signature scheme to generate the proxy signature, and it is proven to be secure under the hardness assumption of CDHP and the random oracle model, so the new proxy signature is unforgeable.

Recently, many ID-based signature schemes have been proposed using the bilinear pairings [4, 8, 17, 19]. Like above construction of Hess version, it is easy to construct other proxy signature schemes based on Paterson scheme [17], Cha-Cheon scheme [4] and Sakai-Ohgishi-Kasahara scheme [19].

6 New Proxy Blind Signature Schemes from Pairings

The proxy blind signature satisfies the security properties of both the blind signature and the proxy signature, such signature is suitable for many applications where the users' privacy and proxy signature are required. From the ID-based blind signature scheme, we can construct proxy blind signature scheme. The first ID-based blind signature scheme was proposed by Zhang and Kim [23] in Asiacrypt2002. Recently, they gave another ID-based blind signature scheme [24]. Now, we give a new proxy blind signature scheme based on this ID-based blind signature scheme.

[**Setup:**]

The system parameters $params = \{\mathbb{G}_1, \mathbb{G}_2, e, q, P, H_1, H_2\}$, the original signer has public-secret key pair (PK_o, s_o), the proxy signer has public-secret key pair (PK_p, s_p).

[**Generation of the proxy key:**]

After the original signer and the proxy signer finish the process in Section 4, the proxy signer gets a proxy key S_w.

[**Proxy blind signature generation:**]
Suppose that m is the message to be signed.

- The proxy signer randomly chooses a number $r \in_R Z_q^*$, computes $U = rH_2(w)$, and sends U and the warrant w to the user.
- (Blinding) The user randomly chooses $\alpha, \beta \in_R Z_q^*$ as blinding factors. He/She computes $U' = \alpha U + \alpha\beta H_2(w)$ and $h = \alpha^{-1} H_1(m||U') + \beta$, sends h to the signer.
- (Signing) The signer sends back V, where $V = (r + h)S_w$.
- (Unblinding) The user computes $V' = \alpha V$. He/She outputs $\{m, U', V'\}$.

Then (U', V', w) is the proxy blind signature of the message m.

[**Verification:**]
A verifier can accept this proxy blind signature if and only if

$$e(V', P) = e(U' + H_1(m||U')H_2(w), PK_o + PK_p).$$

Like the discussion in [24], our new proxy blind signature scheme can provide the batch verification. This is very important when the number of verifications is considerably large (*e.g.*, when a branch bank issues a large number of electronic coins and the customer wishes to verify the correctness of the coins). Assuming that $(U_1', V_1'), (U_2', V_2'), \cdots, (U_n', V_n')$ are proxy blind signatures on messages $m_1, m_2, \cdots, m_n$ which issued by the proxy signer with the public key PK_p and the same warrant w form the original signer. The batch verification is then to test if the following equation holds:

$$e(\sum_{i=1}^{n} V_i', P) = e(\sum_{i=1}^{n} U_i' + (\sum_{i=1}^{n} H_1(m_i, U_i'))H_2(w), PK_o + PK_p).$$

The correctness of the verification is easy to check. A warrant made by the original signer is included in a valid proxy blind signature, so, the proxy blind signature is distinguishable, verifiable, identifiable and non-deniable. The blindness and the non-forgeability of this new proxy blind signature are similar to the discussion of [24].

7 A Proxy Ring Signature Scheme

In this section, we give the first proxy ring signature scheme based on Zhang-Kim's ID-based ring signature scheme [23].

[**Setup**]
The system parameters $params = \{\mathbb{G}_1, \mathbb{G}_2, e, q, P, H_1, H_2\}$. Let Alice be the original signer with public key $PK_o = s_oP$ and private key s_o, and $L = \{PS_i\}$ be the set of proxy signers with public key $\{PK_{p_i} = s_{p_i}P\}$ and private key $\{s_{p_i}\}$.

[**Generation of the proxy key:**]
To delegate the signing capacity to a set of proxy signers, the original signer uses BLS short signature scheme to make the signed warrant w. There is an explicit description of the delegation relation in the warrant w. Then sends $(w, s_oH_2(w))$ to the proxy group L. Each proxy signer uses his secret key s_{p_i} to sign the warrant w, and gets his proxy key $S_i = s_oH_2(w) + s_{p_i}H_2(w)$.

[**Proxy ring signature generation:**]
Let Bob be a proxy signer in L with the proxy key S_i. He wants to give a proxy ring signature on message m. Bob chooses a subset $L' \subseteq L$. Bob's public key is listed in L', we assume that n is the cardinality of L'. Bob performs the following procedure:

- (Initialization): Choose randomly an element $A \in \mathbb{G}_1$, compute $c_{k+1} = H_1(L' \parallel m \parallel e(A, P))$.
- (Generate forward ring sequence): For $i = k+1, \cdots, n-1, 0, 1, \cdots, k-1$ (*i.e.*, the value of i all modulo n), choose randomly $T_i \in \mathbb{G}_1$ and compute $c_{i+1} = H_1(L' \parallel m \parallel e(T_i, P)e(c_iH_2(w), PK_o + PK_{p_i}))$.
- (Forming the ring): Compute $T_k = A - c_kS_i$.
- (Output the ring signature): Select 0 (*i.e.*, n) as the glue value, the resulting signature for m and L' is the $(n+1)$-tuple: $(c_0, T_0, T_1, \cdots, T_{n-1})$.

[**Verification**]
Given $(c_0, T_0, T_1, \cdots, T_{n-1}), m, w$, and L', the verifier checks if L' is a valid subset of proxy group L at first. If so, compute

$$c_{i+1} = H_1(L' \parallel m \parallel e(T_i, P)e(c_iH_2(w), PK_o + PK_{p_i}))\ for\ i = 0, 1, \cdots, n-1.$$

Accept if $c_n = c_0$, and reject otherwise.

This proxy ring signature scheme is similar to Zhang-Kim's [23] ID-based ring signature scheme, the difference is taking place the ID public key as $H_2(w))$ and secret key as S_w, the public key of KGC as $PK_o + PK_p$, so, the correctness, signer ambiguity and non-forgeability are like the discussion of Zhang-Kim's [23] ID-based ring signature scheme. On the other hand, the warrant made by the original signer is public, we assume that there is a description for all proxy signers (not for individual, this ensures the signer ambiguity), so, any one can check if L' is valid. The warrant w and the public keys of the original signer and some proxy signers must occur in the verification equation of a proxy ring signature, the distinguishability, verifiability identifiability and non-deniability are satisfied.

8 Conclusion

Various type proxy signatures are important in many applications, such as secure e-commerce. Due to the various applications of the bilinear pairings in cryptography, there are many ID-based cryptographic schemes have been proposed. In this paper, we first have shown how can obtain the proxy-protected delegation using the short signature system of Boneh, Lynn and Shacham. Using this delegation, it is easy to design the proxy signature and proxy blind signature from the conventional ID-based signature schemes using bilinear pairings, we have given some concrete schemes based on existed ID-based signature schemes. At the same time, we introduced a new type of proxy signature – proxy ring signature, and proposed the first proxy ring signature scheme.

References

1. A. Boldyreva, *Efficient threshold signature, multisignature and blind signature schemes based on the Gap-Diffie-Hellman -group signature scheme*, Public Key Cryptography - PKC 2003, LNCS 2139, pp.31-46, Springer-Verlag, 2003.
2. D. Boneh and M. Franklin, *Identity-based encryption from the Weil pairing*, Advances in Cryptology-Crypto 2001, LNCS 2139, pp.213-229, Springer-Verlag, 2001.
3. D. Boneh, B. Lynn, and H. Shacham, *Short signatures from the Weil pairing*, In C. Boyd, editor, Advances in Cryptology-Asiacrypt 2001, LNCS 2248, pp.514-532, Springer-Verlag, 2001.
4. J.C. Cha and J.H. Cheon, *An identity-based signature from gap Diffie-Hellman groups*, Public Key Cryptography - PKC 2003, LNCS 2139, pp.18-30, Springer-Verlag, 2003.
5. D. Chaum, *Blind signatures for untraceable payments*, Advances in Cryptology-Crypto 82, Plenum, NY, pp.199-203, 1983.
6. L. Chen, K. Harrison, A. Moss, D. Soldera and N.P. Smart, *Certification of public keys within an identity based system*, ISC 2002, LNCS 2433, pp. 322-333, Springer-Verlag, 2002.
7. C. Gentry and A. Silverberg, *Hierarchical ID-based cryptography*, Proc. of Asiacrpt2002, LNCS 2501, pp. 548-566, Springer-Verlag, 2002.
8. F. Hess, *Efficient identity based signature schemes based on pairings*, SAC 2002, LNCS 2595, pp.310-324, Springer-Verlag, 2002.
9. A. Joux, *A one round protocol for tripartite Diffie-Hellman*, ANTS IV, LNCS 1838, pp.385-394, Springer-Verlag, 2000.

10. S. Kim, S. Park, and D. Won, *Proxy signatures, revisited*, In Pro. of ICICS 97, LNCS 1334, Springer-Verlag, pp. 223-232, 1997.
11. S. Lal and A.K. Awasthi, *Proxy blind signature scheme*, Cryptology ePrint Archive, Report 2003/072, available at http://eprint.iacr.org/2003/072/.
12. B. Lee, H. Kim and K. Kim, *Secure mobile agent using strong non-designated proxy signature*, Proc. of ACISP2001, LNCS 2119, pp.474-486, Springer Verlag, 2001.
13. J.Y. Lee, J.H. Cheon and S. Kim, *An analysis of proxy signatures: Is a secure channel necessary?*, CT-RSA 2003, LNCS 2612, pp. 68-79, Springer-Verlag, 2003.
14. W.D. Lin and J.K. Jan, *A security personal learning tools using a proxy blind signature scheme*, Proccedings of International Conference on Chinese Language Computing, Illinois, USA, July 2000, PP.273-277.
15. M. Mambo, K. Usuda, and E. Okamoto, *Proxy signature: Delegation of the power to sign messages*, In IEICE Trans. Fundamentals, Vol. E79-A, No. 9, Sep., pp. 1338-1353, 1996.
16. T. Okamoto, M. Tada and E. Okamoto, *Extended proxy signatures for smart cards*, ISW'99, LNCS 1729, Springer-Verlag, pp. 247-258, 1999.
17. K.G. Paterson, *ID-based signatures from pairings on elliptic curves*, Electron. Lett., Vol.38, No.18, pp.1025-1026, 2002.
18. R.L. Rivest, A. Shamir and Y. Tauman, *How to leak a secret*, Advances in Cryptology-Asiacrypt 2001, LNCS 2248, pp.552-565, Springer-Verlag, 2001.
19. R. Sakai, K. Ohgishi, M. Kasahara, *Cryptosystems based on pairing*, SCIS 2000-C20, Jan. 2000, Okinawa, Japan.
20. A. Shamir, *Identity-based cryptosystems and signature schemes*, Advances in Cryptology-Crypto 84, LNCS 196, pp.47-53, Springer-Verlag, 1984.
21. Z. Tan, Z. Liu and C. Tang, *Digital proxy blind signature schemes based on DLP and ECDLP*, MM Research Preprints, No. 21, December 2002, MMRC, AMSS, Academia, Sinica, Beijing, pp. 212-217.
22. L. Yi, G. Bai and G. Xiao, *Proxy multi-signature scheme: A new type of proxy signature scheme*, Electronics Letters, Vol. 36, No. 6, 2000, pp.527-528.
23. F. Zhang and K. Kim, *ID-based blind signature and ring signature from pairings*, Proc. of Asiacrpt2002, LNCS 2501, pp. 533-547, Springer-Verlag, 2002.
24. F. Zhang and K. Kim, *Efficent ID-based blind signature and proxy signature from pairings*, to appear at ACISP 2003, Springer-Verlag, 2003.
25. K. Zhang, *Threshold proxy signature schemes.* 1997 Information Security Workshop, Japan, Sep., 1997, pp.191-197.