Cryptanalysis of the Lee-Hwang Group-Oriented Undeniable Signature Schemes

Guilin Wang, Jianying Zhou, and Robert H. Deng

Laboratories for Information Technology
21 Heng Mui Keng Terrace, Singapore 119613
{glwang,jyzhou,deng}@lit.a-star.edu.sg
http://www.lit.a-star.edu.sg/

Abstract. Undeniable signature is an intriguing concept introduced by Chaum and Antwerpen at Crypto'89. In 1999, Lee and Hwang presented two group-oriented undeniable signature schemes with a trusted center. Their schemes are natural generalizations of Chaum's zero-knowledge undeniable signature scheme proposed in 1990. However, we find that the Lee-Hwang schemes are insecure. In this paper, we demonstrate five attacks on their schemes: four of them are *universal forgery*, in which one dishonest member (maybe collude with a verifier) can get a valid signature on any chosen massage, and another attack allows a dishonest member to prevent honest members from generating valid signatures but his cheating behavior is undetected. We also suggest heuristic improvements to overcome some of the problems involved in these attacks.

Keywords: digital signatures, undeniable signatures, group-oriented undeniable signatures, cryptanalysis.

1 Introduction

Undeniable signature is a special kind of digital signature in the sense that the validity of an alleged signature cannot be verified without the cooperation of the signer. The concept of undeniable signature was first proposed at Crypto'89 by Chaum and van Antwerpen [4]. Followed by this pioneering work, Chaum proposed a zero-knowledge undeniable signature scheme in [2]. Later, at Auscrypt'92, by combining the two concepts of undeniable signature and group-oriented signature [5, 6], Harn and Yang proposed the concept of *group-oriented undeniable signature* [10], in which *only* when all members in an authorized subset of a given group operate collectively, they can generate, confirm or deny a signature on behalf of the group. If the authorized subsets are all subsets of t or more members of a group with n members, then it is called a (t, n) *threshold undeniable signature scheme.*

In [10], Harn and Yang also designed two concrete threshold undeniable signature schemes: $(1, n)$ scheme and (n, n) scheme. However, Langford pointed out that their (n, n) scheme only has a security of 2-out-of-n, because any two adjacent members can generate a valid threshold signature on any message [11]. Lin et al. presented a general (t, n) threshold undeniable signature scheme [14], but it is also subjected to Langford's attack. To overcome the Langford-attack, Lee and Hwang constructed two group-oriented undeniable signature schemes with a trusted center [12] by naturally generalizing Chaum's zero-knowledge undeniable signature [2] to group-oriented environment.

In this paper, we analyze the security of the Lee-Hwang schemes [12] and demonstrate five attacks. Under reasonable assumptions, our attacks are simple, straightforward and efficient. In these attacks, four of them are *universal forgery*, in which one dishonest member (maybe collude with a verifier or the designated combiner) can get a valid signature on any chosen massage, and another attack allows a dishonest member to prevent honest members from generating valid signatures but his cheating behavior is undetected. We also suggest heuristic improvements to overcome some of the problems involved in these attacks.

The rest of this paper is organized as follows. Section 2 reviews two Lee-Hwang group-oriented undeniable signature schemes. Section 3 presents five attacks on their schemes. Section 4 addresses other two weaknesses of their schemes. Section 5 proposes some heuristic improvements to the Lee-Hwang schemes. Section 6 discusses related work. Finally, the conclusion is given in section 7.

2 Review of Two Lee-Hwang Schemes

The Lee-Hwang schemes [12] consist of a trusted center TC, a designed combiner DC [1] and a group of n members U_i $(i \in A = \{1, 2, \cdots, n\})$. The first is a (t, n) threshold undeniable signature scheme, and the other is a generalized group-oriented undeniable signature scheme.

2.1 The (t, n) Threshold Undeniable Signature Scheme

System Setup

The trusted center TC first determines the following public parameters:

- P, p : two large primes, such that $P = 2p + 1$.
- α: an element of order p in Z_P.
- $H(\cdot)$: a collision free one-way hash function [2].
- l: a security parameter (e.g. $l = 1023$).
- x_i: n public values, each x_i is associated with the member U_i such that $x_i \neq x_j$ if $i \neq j$ $(i, j \in A)$.

[1] DC is an untruthful entity [15, 13].

[2] In order to guarantee the order of overwhelming part of $H(m)$ is p, [LH99] required that if the order of $H(m)$ mod P is $P - 1$, then let $H(m)^2$ be the digest of message m. Because this processing does not affect the discussion here, we will simply use $H(m)$ as the digest of message m.

After this, the TC selects a secret random number S from Z_p as the group private key, and a random polynomial $f(x) \in Z_p[x]$ of degree $t-1$ such that $f(0) = S$. Then, the TC privately sends the share $s_i = f(x_i)$ to the member U_i as his secret key, and publishes $Y = \alpha^S \bmod P$ as the group public key.

Signing Protocol

Any t members U_i ($i \in B$, $|B| = t$ and $B \subseteq \{1, 2, \cdots, n\}$) can generate a threshold undeniable signature on any message m as follows.

(1-1) Each U_i calculates his partial undeniable signature $z_i = H(m)^{s_i C_{Bi}} \bmod P$, where the Lagrange coefficient C_{Bi} is determined by

$$C_{Bi} = \sum_{j \in B, j \neq i} x_j/(x_j - x_i) \bmod p.$$

Then, U_i sends z_i to the DC.

(1-2) Upon receiving t partial undeniable signatures, the DC computes the threshold undeniable signature Z on message m by

$$Z = \prod_{i \in B} z_i \bmod P \;(= H(m)^{\sum_{i \in B} s_i C_{Bi} \bmod p} \bmod P = H(m)^S \bmod P).$$

Confirmation Protocol

If t members U_i ($i \in B$) in the group agree to verify a undeniable signature pair (m, Z), then the verifier V and these t members run the following confirmation protocol cooperatively.

(2-1) V chooses two random numbers $a, b \in_R Z_p$, computes the value $W = H(m)^a \alpha^b \bmod P$ and sends W to each member in B.

(2-2) After receiving W, each U_i ($i \in B$) selects a number $k_i \in_R Z_p$, and computes the value $K_i = \alpha^{k_i} \bmod P$. Then, all K_i are broadcasted to let all members in B compute a value R_1 by the following equation:

$$R_1 = W \prod_{i \in B} K_i \bmod P \,(= W \prod_{i \in B} \alpha^{k_i} \bmod P = W\alpha^{\sum_{i \in B} k_i} \bmod P).$$

Moreover, U_i computes and broadcasts the following value $R_{2,i}$

$$R_{2,i} = {R_1}^{s_i C_{Bi}} \bmod P.$$

Up to this, the DC (and any member) calculates the following value R_2

$$R_2 = \prod_{i \in B} R_{2,i} \bmod P \,(= R_1^S \bmod P).$$

At last, R_1 and R_2 are sent to verifier V.

(2-3) V sends a and b to each U_i in B.

(2-4) Each member in B checks whether $W \equiv H(m)^a \alpha^b$. If it does hold, then the value $k = \sum_{i \in B} k_i$ is revealed to V.

(2-5) V accepts (m, Z) as a valid signature pair if and only if the following two equalities hold:

$$R_1 \equiv W\alpha^k \bmod P, \quad R_2 \equiv Z^a Y^{b+k} \bmod P.$$

Denial Protocol

Any t members, U_i $(i \in B)$, can convince a verifier V that an alleged signature pair (m, Z) is not generated by the group. For this goal, the following denial protocol is run between these t members and V [3].

(3-1) V randomly selects two numbers $q \in_R [0, l]$ and $c \in_R Z_p$, then computes and sends $E_1 = H(m)^q \alpha^c \bmod P$ and $E_2 = Z^q Y^c \bmod P$ to each member in B.

(3-2) All members in B cooperate to find the value of q by trial and error. Then, each U_i $(i \in B)$ chooses a random integer d_i and sends $blob(d_i, q)$ to V as his commitment to q [4].

(3-3) V sends c to each member in B.

(3-4) Each member U_i in B checks whether the following two equalities hold:

$$E_1 \equiv H(m)^q \alpha^c \bmod P, \quad E_2 \equiv Z^q Y^c \bmod P.$$

If they do hold, then each U_i reveals his d_i to the verifier V.

(3-5) V opens all $blob(d_i, q)$ to check whether all the committed values are equivalent to q. If yes, V believes that (m, Z) is not generated by the group.

2.2 The Generalized Group-Oriented Undeniable Signature Scheme

Let $\mathcal{L}$ be an access structure on A, i.e., the collection of all authorized sets B_r. To allow each authorized set can generate a valid signature, Lee and Hwang designed a simple and efficient generalized group-oriented undeniable signature scheme such that each member has only one secret key. We only overview the setup stage here because the signing, confirmation and denial protocols are almost the same as the above scheme.

System Setup

Similar to the threshold case, the TC chooses S as the group private key, and $\{P, p, \alpha, Y\}$ as the group public key. At the same time, the TC assigns a pair of numbers (x_i, s_i) to each member U_i in A, where x_i is a public value and s_i is the secret key of member U_i. Then, for each authorized set B_r $(|B_r| = t_r)$, the

[3] In practice, if $l = 1023$, then the denial protocol could be conducted ten times to reach the $1/2^{100}$ level security [2]. That is, the occurrence of the following event is no more than one in a million: V believes that S is not signed by the group, but in fact S is the group's signature on message m.

[4] $blob(d_i, q)$ means that the value of q is committed by d_i [2, 12].

TC constructs a polynomial $f_r(x)$ of degree t_r by interpolating (t_r+1) points, i.e. $(0, S)$ and (x_i, s_i) $(i \in B_r)$, as follows

$$f_r(x) \triangleq S \prod_{j \in B_r} \frac{(x - x_j)}{(0 - x_j)} + \sum_{i \in B_r} \Big[s_i \frac{(x-0)}{(x_i - 0)} \prod_{j \in B_r,\, j \neq i} \frac{(x - x_j)}{(x_i - x_j)} \Big] \bmod p.$$

After this, the TC chooses a public value x_c, then computes and publishes $f_r(x_c) \bmod p$ for each B_r. Obviously, all $f_r(x)$ have the properties that $f_r(0) = S$ and $f_r(x_i) = s_i$, $\forall\, i \in B_r$. So, by using Lagrange interpolating equation on $t_r + 1$ points $(x_c, f_r(c))$ and (x_i, s_i) $(i \in B_r)$, t_r members in each B_r can cooperatively generate, confirm or deny a group-oriented undeniable signature in a similar way as they did in the threshold scenario.

3 Five Attacks on Lee-Hwang Schemes

In [12], Lee and Hwang claimed that less than t members in their threshold scheme, or t_r members in the generalized scheme, cannot generate, confirm or deny the group-oriented undeniable signature. However, this is not true. We demonstrate five attacks on their schemes: In the fist attack, against signing protocols, one dishonest member can prevent honest members from generating valid signatures but his cheating behavior is undetected; in the other four attacks, against confirmation and denial protocols, one dishonest member (maybe collude with the DC or a verifier V) can generate a valid signature on any chosen message. So, these later four attacks are universal forgery, which should be avoided in a secure digital signature scheme.

In the following attacks, for convenience but without loss of generality, we always assume U_1 is dishonest and $B = \{1, 2, \cdots, t\}$. Sometimes, the success of an attack needs the help of the DC or a verifier V, i.e. in this case, the DC or V is also dishonest. This is reasonable because the DC and V are untruthful entities in the system. Here, we only describe attacks on Lee-Hwang threshold undeniable signature scheme. Similar attacks can be applied to their generalized group-oriented scheme.

3.1 One Attack on Signing Protocol

[**Attack 1**] In the signing protocol, no method is provided to verify the validity of each partial signature z_i. So dishonest member U_1 can cheat others by publishing a false $\bar{z}_1$ instead of true z_1. Then, using his valid partial signature z_1 and other published valid partial signature z_i (if only the DC knows these valid z_i, we assume the DC colludes with U_1 and reveals these values to him.), he can compute the valid threshold undeniable signature Z on message m by the following equation.

$$Z = \prod_{i=1}^{t} z_i \bmod P \; (= H(m)^S \bmod P).$$

But the DC, other group members and the expected receiver of the valid signature on message m can only get an invalid signature $\bar{Z}$ on message m by the following equation

$$\bar{Z} = \bar{z}_1 \cdot \prod_{i=2}^{t} z_i \bmod P.$$

Once U_1 gets the valid signature pair (m, Z), he keeps it secretly, and publishes it in a suitable time or reveals it to some relevant party for getting benefits. When this unexpected receiver provides (m, Z) to the group and require to verify the validity of this signature pair, group members cannot deny it because is indeed valid. The essence of this attack is that dishonest member U_1 (maybe collude with the DC) successfully prevents other members from generating valid signatures without any penalty, because his cheating behavior is undetected.

3.2 Two Attacks on Confirmation Protocol

[**Attack 2**] In this attack, U_1 colludes with a verifier V. Before attacking, they have chosen a message $\bar{m}$. Then V selects two numbers a and b, and computes W normally. Member U_1 sets $\bar{K}_1 = H(\bar{m})\alpha^{k_1} \bmod P$, although each other member $U_i\,(i = 2, \cdots, t)$ honestly chooses k_i and computes $K_i = \alpha^{k_i} \bmod P$. After $\bar{K}_1$ and all K_i are broadcasted, $\bar{R}_1$ and $\bar{R}_2$ are calculated as follows

$$\bar{R}_1 = W\bar{K}_1K_2\cdots K_t \bmod P \ (= W\alpha^{k_1+\cdots+k_t}H(\bar{m}) \bmod P),$$

$$\bar{R}_2 = \bar{R}_1^S \bmod P \ (= Z^aY^{b+k_1+\cdots+k_t}H(\bar{m})^S \bmod P).$$

Then, $\bar{R}_1$ and $\bar{R}_2$ are sent to V and V reveals a and b to each member in B. At last, $k = k_1 + k_2 + \cdots + k_t \bmod p$ is sent to V. Using the values of a, b and k, U_1 and V compute the signature $\bar{Z}$ on message $\bar{m}$ by the following equation

$$\bar{Z} = \bar{R}_2/(Z^aY^{b+k}) \bmod P \ (= H(\bar{m})^S \bmod P).$$

[**Attack 3**] In this attack, under the assumption that U_1 publishes the value K_1 last, he can get a valid threshold undeniable signature on any chosen message $\bar{m}$. The details are described as follows.

When U_1 has received W from V and all K_i from U_i $(i = 2, 3, \cdots, t)$, he computes and broadcasts his value $\bar{K}_1$ as follows

$$\bar{K}_1 = H(\bar{m})(WK_2\cdots K_t)^{-1} \bmod P.$$

Then, the following value $\bar{R}_1$, instead of R_1, will be calculated by

$$\bar{R}_1 = W\bar{K}_1K_2\cdots K_t \bmod P = H(\bar{m}) \bmod P.$$

Followed by this value $\bar{R}_1$, $\bar{R}_2 = \bar{R}_1^S \bmod P = H(\bar{m})^S \bmod P$ will be produced. Then, $\bar{R}_1$ and $\bar{R}_2$ are sent to V. As a response, V sends a and b back to each member in B. Up to this, U_1 gets a valid signature pair $(\bar{m}, \bar{R}_2)$.

In the step (2-4), U_1 has the following two choices: (1) He selects a random number $k_1 \in_R [0, p-1]$ and reveals it to other members; (2) He disrupts the confirmation protocol by telling other members that he lost the value of k_1 in a reasonable excuse (e.g., by a computer crash). In the first case, V will fail in step (2-5) with probability of $(p-1)/p$. But in the second case, possibly, the protocol will be conduct again and this time U_1 behave honestly. Anyway, from the above attack, U_1 gets a valid undeniable signature $\bar{R}_2$ on message $\bar{m}$ selected by himself.

If U_1 cannot access the value of $\bar{R}_2$, he will also succeed in this attack in collusion with the DC or a verifier V to get $\bar{R}_2$.

3.3 Two Attacks on Denial Protocol

[**Attack 4**] In [12], no details were given on how to find the value of q in the denial protocol by trial and error method. A straightforward method is to compute the values of E_1^S and $H(m)^S$ by using the signing protocol, then all members in B find the value of q from the following equation by trial and error:

$$E_1^S/E_2 = (H(m)^S/Z)^q \bmod P.$$

However, by exploring this method, the valid signature $H(m)^S$ on message m is generated, so each member (and the DC) knows its value. Therefore, any dishonest member of them can keep this signature privately or reveal it to a third party which has interest in it. In some scenarios, it is also possible that all members in B are unwilling to generate the signature on message m.

[**Attack 5**] U_1 colludes with a verifier V to get a valid signature on any chosen message $\bar{m}$. For this sake, V prepares $E_1 = H(\bar{m}) \bmod P$ and $E_2 = Z^q \alpha^c \bmod P$. When all members in B generated E_1^S by using signing protocol, U_1 knows that this value is the valid signature $\bar{Z}$ on message $\bar{m}$, i.e., $\bar{Z} = E_1^S \bmod P = H(\bar{m})^S \bmod P$. In step (3-3), verifier V disrupts denial protocol by claiming he lost the value of c because of a computer crash. Then, possibly, the denial protocol will be repeated and at this time V behaves honestly. Generally, it would be unreasonable to assume that the denial protocol will not be conducted again only because a verifier erroneously sends a wrong value.

4 Other Weaknesses

Besides the five attacks demonstrated in the previous section, there are two weaknesses in the Lee-Hwang schemes. Firstly, we need to add a requirement on choosing values of all s_i in their generalized scheme. Secondly, we address the limitation of their schemes on how to efficiently and securely delete members.

In the generalized undeniable signature scheme, Lee and Hwang didn't give details about how to choose the secret sharing key s_i for each member U_i. If the TC simply selects these values as random numbers, it is possible that some of $f_r(x)$ (see subsection 2.2) are not t_r-degree polynomials. Therefore, some requirements should be met when selecting these values. For each authorized subset $B_r = \{i_1, i_2, \cdots, i_{t_r}\}$, let $f_r(x) = \sum_{j=0}^{t_r} a_{rj} x^j$ (where, $a_{r0} = S$). It is obvious that the coefficients $(a_{r0}, a_{r1}, \cdots, a_{rt_r})$ of $f_r(x)$ is the solution of the following system of $t_r + 1$ linear equations:

$$s_{i_j} = \sum_{j=0}^{t_r} a_{rj} x_{i_j}^j, \quad j = 0, 1, \cdots, t_r.$$

We take $s_{i_0} = S$, $x_{i_0} = 0$ and $0^0 = 1$.

Let A_r be the $(t_r + 1) \times (t_r + 1)$ coefficient matrix of the above linear equations. According to Cramer's rule, each a_{rj} can be expressed as $a_{rj} = \det(A_{rj})/\det(A_r)$, for all $j = 0, 1, \cdots, t_r$. A_{rj} is the matrix obtained by replacing the j-th column of A_r with the column vector $(s_{i0}, s_{i1}, \cdots, s_{it_r})$. Since all x_{i_j} are different, so Vandermonde matrix A_r is nonsingular. Therefore, the necessary and sufficient condition for $f_r(x)$ to be a t_r-degree polynomial, i.e. $a_{rt_r} \neq 0$, is that

$$\det(A_{rt_r}) \neq 0, \quad \forall\, B_r \in \mathcal{L}.$$

Now, we discuss the problem about deleting members. In practical applications, members should be dynamically added or deleted according to the change of structure of the group. Due to the existence of the trusted center TC, member addition can be implemented easily in the Lee-Hwang schemes. However, how to dynamically delete members seems difficult in their schemes. If one or several members are deleted but the TC doesn't update the group private key S, then the information controlled by the deleted members can be used for generating valid partial signatures. On the other hand, if the TC always updates the group private key S after a member is deleted, each member's secret key s_i will be updated too. This almost has no difference from re-setting up the whole system. But in a large group, such an approach is inefficient due to expensive costs in computation and communications. In fact, dynamically deleting member is an open problem that stands in the way of practical applications of group-oriented signatures [1].

5 Improvements

In the attack 1, the problem is that when a signature Z is generated, *neither* the DC *nor* any member in B knows whether Z is a valid signature on message m, unless all of them cooperatively conduct the confirmation protocol. If so, to let all these t members in B believe that they have generated a valid Z, the confirmation protocol must be conducted t times: in each running instance, one different member of them plays the role of verifier. If the scheme is improved like this, other problem arises. For example, if all members in B are honest in the procedures of generating and publishing their partial signatures, a valid signature Z will be generated. But, in the verification procedure, one dishonest member can conduct the confirmation protocol for an illegal verifier. The result is that the illegal verifier is convinced of the validity of the undeniable signature pair (m, Z) just when it is generated.

Therefore, in the signing protocol, some mechanisms, e.g. discrete logarithm knowledge proofs [3, 18], should be provided such that any members in B can verify the validity of partial signatures generated by others. At the same time, each member should check whether Z is identical with the product of all z_i, $i \in B$. Otherwise, the DC also can cheat honest members by publishing a false Z which is not equivalent to the product of all z_i.

The kernel problem in attacks 2 and 3 is that a dishonest member can use a value of K_i without knowing the value of k_i such that $K_i = \alpha^{k_i} \bmod P$. To overcome this problem, some standard techniques, like knowledge commitments or discrete logarithm knowledge proofs [3, 21], could be employed in confirmation protocol.

The real reason for the success of attacks 4 and 5 is that values of the form X^S, i.e. E_1^S and $H(m)^S$, are generated in order to find the value of q by trial and error. A direct countermeasure is not to generate these values in the process of finding q. Unfortunately, we have no idea to solve this problem at the moment.

6 Related Work

In [15], Michels and Horster discovered some attacks against several multiparty signature schemes. Their attacks are in common that the attacker is an *insider*, i.e. a dishonest group member, and the protocol will be disrupted. In fact, our attack 4 is inspired by their attacks.

In [16], based on Schnorr's signature scheme [19], Michels and Stadler proposed an efficient convertible undeniable signature scheme in which confirmation protocol and denial protocol are combined together into a verification protocol, and furthermore, they extended their scheme to a (t, n) threshold undeniable signature scheme. Since they used techniques of verifiable secret sharing of discrete logarithms [17] and the particular form of values in verification protocol, the attacks presented here cannot be applied to their scheme.

Base on the first undeniable RSA signature scheme [9] and Shoup's threshold RSA signature [21], Wang et al presented a threshold undeniable RSA signature scheme in [23]. Our attacks presented here cannot be applied to this scheme either because discrete logarithm knowledge proofs [3, 21] are used to verify the validity of partial signatures and no values of the form X^d are calculated. Where d, similar to the value of S in Lee-Hwang schemes, is the signing key in [23].

7 Conclusion

In this paper, we demonstrated five effective attacks on the Lee-Hwang group-oriented undeniable schemes [12]. Four of these attacks are universal forgery, in which one dishonest member (maybe collude with a verifier or the designated combiner) can get a valid signature on any chosen massage, and the remainder attack allows a dishonest member to prevent honest members from generating valid signatures but his cheating behavior is undetected. To overcome some of the problems involved in these attacks, heuristic improvements were also suggested. But, how to solve the problem in the denial protocol is still open. At the same time, the Lee-Hwang schemes have two strong limitations: it needs a trusted center and cannot delete members efficiently. Furthermore, as pointed out in [15], heuristic improvements cannot guarantee the security of a repaired cryptosystem. So, threshold cryptosystems should be designed as provably secure [22, 8]. These problems would be considered in the future research.

References

1. G. Ateniese, and G. Tsudik. Some open issues and new directions in group signature schemes. In: *Financial Cryptography* (*FC'99*), *LNCS 1648*, pp. 196-211. Berlin: Springer-Verlag, 1999.
2. D. Chaum. Zero-knowledge undeniable signatures. In: *Eurocrypt'90, LNCS 473*, pp. 458-464. Springer-Verlag, 1991.
3. D. Chaum, and T.P. Pedersen. Wallet databases with observers. In *Crypto'92, LNCS 740*, pp. 89-105. Springer-Verlag, 1993.
4. D. Chaum, and H. van Antwerpen. Undeniable signatures. In *Crypto'89, LNCS 435*, pp. 212-216. Springer-Verlag, 1989.
5. Y. Desmedt. Society and group oriented cryptography: A New Concept. In *Crypto'87, LNCS 293*, pp. 120-127. Springer-Verlag, 1988.
6. Y. Desmedt, and Y. Frankel. Threshold cryptosystems. In *Crypto'89, LNCS 435*, pp. 307-315. Springer-Verlag, 1990.
7. I. Damgård, and T. Pedersen. New convertible undeniable signature schemes. In *Eurocrypt'96, LNCS 1070*, pp. 372-386. Springer-Verlag, 1996.
8. P.-A. Fouque, and D. Pointcheval. Threshold Cryptosystems Secure against Chosen-Ciphertext Attacks. In *Asiacrypt'01, LNCS 2248*, pp. 351368. Springer-Verlag, 2001.
9. R. Gennaro, H. Krawczyk, and T. Rabin. RSA-based undeniable signature. In *Crypto'97*, pp. 132-148. Springer-Verlag, 1997. 307-313.
10. L. Harn, and S. Yang. Group-oriented undeniable signature schemes without the assistance of a mutually trusted party. In *Auscrypt'92, LNCS 718*, pp. 133-142. Springer-Verlag, 1993.
11. S.K. Langford. Weakness in some threshold cryptosystems. In *Crypto' 96, LNCS 1109*, pp. 74-82. Springer-Verlag, 1996.
12. N.-Y. Lee, and T. Hwang. Group-oriented undeniable signature schemes with a trusted center. *Computer Communications*, 22(8): 730-734. Elsevier Science, May 1999.
13. C-M. Li, T. Hwang and N-Y. Lee. Threshold-multisignature schemes where suspected forgery implies traceability of adversarial shareholders. In: *Eurocrypt'94, LNCS 950*, pp. 194-204. Berlin: Springer-Verlag, 1995.
14. C.-H. Lin, C.-T. Wang, and C.-C. Chang. A group-oriented (t, n) undeniable signature scheme without trusted center. In: *Information Security and Privacy, ACISP'96, LNCS 1172*, pp. 266-274. Springer-Verlag, 1996.
15. M. Michels, and P. Horster. On the risk of discruption in several multiparty signature schemes. In *Asiacrypt'96, LNCS 1163*, pp.334-345. Springer-Verlag, 1996.

16. M. Michels, and M. Stadler. Efficient convertible signature schemes. In *Proc. 4th Workshop on Selected Areas in Cryptography* (*SAC'97*), pp. 231-244. Ottawa, Canada, 1997.
17. T.P. Pedersen. No-interactive and information-theoretic secure verifiable secret sharing. In *Crypto'91, LNCS 576*, pp. 129-140. Springer-Verlag, 1992.
18. T.P. Pedersen. Distributed provers with applications to undeniable signatures. In *Eurocrpt'96, LNCS 547*, pp. 221-242. Springer-Verlag, 1996.
19. C.P. Schnorr. Efficient signature generation for smart cards. *Journal of Cryptology*, 1991, 4(3): 161-174.
20. A. Shamir. How to share a secret. *Communications of the ACM*, 1979, 22(11): 612-613.
21. V. Shoup. Practical Threshold Signatures. In *Eurocrypt'2000, LNCS 1807*, pp. 207-220. Springer-Verlag, 2000.
22. D.R. Stinson, R. Strobl. Provably Secure Distributed Schnorr Signatures and a (t, n) Threshold Scheme for Implicit Certificates. In *ACISP'01, LNCS 2119*, pp. 417-434. Springer-Verlag, 2001.
23. G. Wang, S. Qing, M. Wang and Z. Zhou. Threshold undeniable RSA signature scheme. In *Information and Communications Security* (*ICICS 2001*), *LNCS 2229*, pp. 220-231. Berlin: Springer-Verlag, 2001.