Analysis and Improvement of Fair Certified E-Mail Delivery Protocol

Chih-Hua Lai and Ren-Junn Hwang
Department of Computer Science and Information Engineering, Tamkang University, Damsui, Taipei County, Taiwan, R.O.C.
892190140@s92.tku.edu.tw

Abstract*-Electronic mail delivery is more and more indispensable to the application of e-commerce over public Internet. Fair certified e-mail delivery (CEMD) provides fair exchange protocol for preserving non-repudiation of origin and receipt, simultaneously. CEMD ensures that the sender is capable of obtaining an irrefutable receipt if and only if the recipient gets the certified e-mail in a fair way. In this paper, a novel CEMD protocol based on familiar RSA signature is proposed with pre-computation for sustained e-mail communication. Our protocol reduces computational cost and communication overhead in sending the other mails to the same recipient.*

Keywords: Certified e-mail, Digital signature, Fair exchange, Security.

1. Introduction

Due to characteristics of rapidity and inexpensive, electronic letter (e-mail) has become more and more popular communication tool for business instead of traditional manuscript letter nowadays. Although we can use digital signature such as well-known RSA [20] or S/MIME [21, 25] appended into e-mail to ensure non-repudiation of origin, the non-repudiation of receipt still relies on the willingness of the recipient.

Certified e-mail delivery (CEMD) protocol [11, 13, 17, 23] is developed to establish reliable e-mail system. It allows two mistrusting parties to exchange the certified e-mail and its receipt in a fair way. Obviously, the CEMD protocol is a kind of fair exchange. Fair exchange protocols include the following different but related variants [1, 26]: non-repudiation protocols [9, 22], electronic contract signing protocols [5, 7, 8, 12], certified e-mail delivery protocols [11, 13, 17], and fair document exchange protocols [18, 27]. A non-repudiation and electronic contract signing protocols aimed for fairly exchanging irrefutable signatures. However, CEMD is an e-mail system which fairly exchanges messages and its receipt. The other kind of fair exchange is fair document exchange protocol that developed for fairly exchanging respective documents. Although fair document exchange protocol can be regarded as the generalization case of fair exchange protocol, it is not the efficient way to exchange only one e-mail message and its receipt. For more details of fair exchange protocols, please refer to [2, 10].

Formally, CEMD protocol provides following main security requirements [11, 23]:

(1) *non-repudiation of origin*: the recipient must have a way of proving that the e-mail indeed sent from the original sender.
(2) *non-repudiation of receipt*: the sender must have a way of proving that his/her e-mail has been successfully obtained by the designated recipient.
(3) *Strong fairness*: at the end of CEMD protocol, the recipient is able to obtain the e-mail if and only if the sender can obtain its receipt.

For fairness assurance in CEMD protocol, the help of a trusted third party between two mutually distrusting parties is necessary [6, 14]. Based on the extent of involvement of the trusted third party, certified e-mail delivery protocol can be classified into two main types including on-line TTP [17] and off-line TTP [11, 13, 14]. The on-line TTP actively involved during each transmission of exchange. However, on-line TTP could be expensive for maintenance, and usually will cause the communication bottleneck. The off-line TTP of CEMD protocol only needs to interact with the participators when dispute occurs for restoring fairness.

Generally, the verifiable encrypted signature (VES) [4] is used to construct CEMD. The concept of VES technology is verifiability and recoverability. The verifiability ensures that the e-mail sender can verify the VES without obtaining the real signature. The recoverability permits that the real receipt can be recovered with the assistance of an agreed off-line TTP to maintain the fairness if any party misbehaves or unexpectedly aborts. Hence, off-line TTP-based

CEMD protocol using VES technology is the state-of-the-art solution since it is efficiently solving the difficult problem of strong fairness.

In this article, the contributions contain twofold. At first, we present the critique that Ma et al.'s CEMD protocol [11] unfortunately still can not achieve the requirement of strong fairness. In this article, we revise the weakness of Ma et al.'s CEMD protocol. Secondly, we design a novel certified e-mail delivery protocol. The proposed protocol provides an efficient pre-computation function for continued transmission when to the same recipient. Therefore, our protocol is more efficiently suitable for the common e-mail delivery circumstance that the sender will send a number of different e-mails to the same recipient frequently. With pre-computation function of our protocol, the computational cost can be reduced about 30% than Ma et al.'s CEMD protocol [11].

The remainder of the paper is organized as follows. The notations and assumptions used throughout the paper are defined in Section 2. Next, we review and analysis the weakness for Ma et al.'s protocol [11] in Section 3. Afterward, we propose a novel CEMD with pre-computation in Section 4. The security analyses and the performance evaluations of our CEMD protocol are shown in Section 5 and Section 6. Eventually, we give the briefly conclusions in Section 7.

2. Notations and assumptions

Throughout the paper, the notations are defined in Section 2.1, and the assumptions are defined in Section 2.2.

2.1. Notations

- A, B, T: the unique identity of e-mail sender A, recipient B, and trusted third party T, respectively.
- H(.): collision-resistant one-way hash function such as SHA-1 [16].
- $x||y$: the concatenation of messages x and y.
- A→B: m denotes that the message m is sending from party A to party B.

2.2. Assumptions

- E-mail sender A and recipient B have both agreed to employ an off-line trusted third party T. The off-line TTP will not conspire with any participators.
- Every parties $i \in \{A, B, T\}$ have their own public and private RSA-based key pair, where the public key $pk_i=\{e_i, n_i\}$ and the private key $sk_i=\{d_i, n_i\}$ such that n_i is a product of two distinct large prime p_i and q_i and $(e_i \times d_i) \equiv 1 \pmod{(p_i-1)(q_i-1)}$. The public key pk_i is assumed that certified by the Certification Authority (CA) and known by all the other parties. The party i keeps his/her own private key sk_i in secret.
- Initially, recipient B has obtained a recovery certificate $C_{BT}=\{pk_{BT}, w_{BT}, s_{BT}\}$, issued from the party T. The values embedded in C_{BT} are defined as following. Note that, off-line TTP T has no need to store any temporary key x and C_{BT}. The temporary key $x=w_{BT} \times \mathrm{H}(sk_T||pk_{BT}) \bmod n_B$ can be recovered using the private key sk_T of party T.
  - $pk_{BT}=(g, y, n_B)$, where $g \in Z_{n_B}^*$ is selected prime integer with large order, and $y=g^x \bmod n_B$ such that x is the random integer;
  - $w_{BT}=x \times \mathrm{H}(sk_T||pk_{BT})^{-1} \bmod n_B$, such that sk_T is the private key of party T;
  - $s_{BT}=\mathrm{H}(pk_{BT}||w_{BT}||e_B||n_B)^{d_T} \bmod n_T$ is the RSA-based signature.

3. Review of Ma et al.'s CEMD protocol

Firstly, we review Ma et al.'s CEMD protocol [11] in Section 3.1. Afterward, we demonstrate the weakness of Ma et al.'s protocol in Section 3.2.

3.1. Ma et al.'s CEMD protocol

Ma et al.'s protocol consists exchange phase and receipt recovery phase, and describes below.

3.1.1. Exchange phase. We assumed that party A attempts to use e-mail m in exchange of its receipt $\sigma_B=\mathrm{H}(m)^{d_B} \bmod n_B$ from party B. The exchange phase contains Step (E1) to Step (E4) as following. The message flows are shown in Figure 1.

(E1): Party A sends $h=H(m)$ and the signature $\sigma_A=\mathrm{H}(m)^{d_A} \bmod n_A$ to party B.

(E2): After verifying the RSA-based signature σ_A for h, the party B sends the VES values (U, V, c, r) and C_{BT} to party A. The details of Step (E2) are described as following sub-steps:

(E2-1): selects random numbers α and $w \in Z_{n_B}^*$;

(E2-2): computes $\sigma_B=h^{d_B} \bmod n_B$;

(E2-3): computes the values $U=g^{\alpha} \bmod n_B$ and $V=\sigma_B \times y^{\alpha} \bmod n_B$;

(E2-4): computes the values $t_g=g^w \bmod n_B$ and $t_y=(y^{e_B})^w \bmod n_B$;

(E2-5): computes two values $c=\mathrm{H}(h||A||B||t_g||t_y)$ and $r=w-c \times \alpha$;

(E2-6): sends values $\{U, V, c, r\}$ and C_{BT} to A.

(E3): Party A performs the following sub-steps to verify the values $\{U, V, c, r, C_{BT}\}$ and then sends the real e-mail message m to party B.

(E3-1): checks RSA-based signature s_{BT} on C_{BT};
(E3-2): computes $t_g=g^r \times U^c \bmod n_B$;
(E3-3): computes $t_y=(y^{e_B})^r \times (V^{e_B}/\mathrm{H}(m))^c \bmod n_B$;
(E3-4): If equation $c=\mathrm{H}(\mathrm{H}(m)\|A\|B\|t_g\|t_y)$ holds, sends e-mail message m to party B.

(E4): After receiving e-mail m and verifying the equation $h=\mathrm{H}(m)$, party B sends back the real receipt σ_B to party A. Eventually, party A checks $\mathrm{H}(m)=\sigma_B^{e_B} \bmod n_B$. If it is valid, the certified e-mail delivery protocol is completed. Otherwise, party A initiates the receipt recovery phase.

$$\begin{aligned}&(\mathrm{E1}): A \to B: h = H(m), \sigma_A = H(m)^{d_A} \bmod n_A\\ &(\mathrm{E2}): B \to A: (U, V, c, r), C_{BT}\\ &(\mathrm{E3}): A \to B: m\\ &(\mathrm{E4}): B \to A: \sigma_B\end{aligned}$$

Figure 1. Exchange phase of Ma et al.'s protocol.

3.1.2. Receipt recovery phase. In the circumstance that party A fails to obtain the party B's receipt σ_B, party A may request for involving in the receipt recovery phase with the help of off-line TTP T. The steps including Step (R1) and Step (R2) of this phase are illustrated in Figure 2.

(R1): Party A sends $\{U, V, c, r, C_{BT}, m\}$ to the agreed off-line trusted third party T.

(R2): party T checks $\{U, V, c, r, C_{BT}\}$ for e-mail m by the same procedures shown in the step (E3) of exchange phase above. If verification passed, party T recovers the secret key $x=w_{BT} \times \mathrm{H}(sk_T\|pk_{BT}) \bmod n_B$. Afterward, party T recovers the real receipt $\sigma_B=V/U^x \bmod n_B$. Finally, party T securely sends e-mail m to party B and sends receipt σ_B to party A, respectively.

$$\begin{aligned}&(\mathrm{R1}): A \to T: (U, V, c, r), C_{BT}, m\\ &(\mathrm{R2}): T \to A: \sigma_B \text{ and } T \to B: m\end{aligned}$$

Figure 2. Receipt recovery phase of Ma et al.'s protocol.

3.2. Weakness in Ma et al.'s protocol

Unfortunately, the weakness of unfairness occurs since the party B always can easily forge the unrecoverable VES values (U', V', c', r') to pass all the party A's verifications in the exchange phase. Hence, it will cause erroneous decision for the party A to send back the real e-mail message m to party B in Step (E3) of exchange phase. In this moment, the party B gives up sending the receipt σ_B to party A in Step (E4) of exchange phase. Although party A can try to initiate the receipt recovery phase, party T will generate the wrong receipt $\sigma_B' \neq \sigma_B$ from the forged VES values (U', V', c', r'). Party B performs the following Step (E2') in place of Step (E2) of original exchange phase to forge the unrecoverable VES values (U', V', c', r'). The details of Step (E2') are described in follows and shown in Figure 3.

(E2'): After receiving these values and verifying the signature σ_A for h, party B sends the forged VES values $\{U', V', c', r', C_{BT}\}$ back to the party A. The details are described in following sub-steps:

(E2'-1): randomly selects three distinct integers r', β and $\lambda \in Z_{n_B}^*$;
(E2'-2): computes the values $t_g'=g^{r'+\beta} \bmod n_B$ and $t_y'=(y^{e_B})^{r'} \times h^{\lambda} \bmod n_B$;
(E2'-3): computes $c'=\mathrm{H}(h\|A\|B\|t_g'\|t_y')$;
(E2'-4): computes the value $U'=g^{\beta \times (c')^{-1}} \bmod n_B$, where $c' \times (c')^{-1} \equiv 1 \pmod{(p_B-1)(q_B-1)}$;
(E2'-5): computes the value $V'=h^{d_B \times (\lambda \times (c')^{-1}+1)} \bmod n_B$ using the private key $sk_B=\{d_B, n_B\}$ of B;
(E2'-6): sends the forged VES values $\{U', V', c', r'\}$ and C_{BT} to party A.

Therefore, party A will get the valid values $\{t_g', t_y'\}$ in the Step (E3) of exchange phase. The correctness for the values $\{t_g', t_y'\}$ are presented below:

- $t_g'=g^{r'} \times (U')^{c'} = g^{r'} \times (g^{\beta \times (c')^{-1}})^{c'} = g^{r'+\beta} \bmod n_B$.
- $t_y'=(y^{e_B})^{r'} \times ((V')^{e_B} / \mathrm{H}(m))^{c'} \bmod n_B$
  $=(y^{e_B})^{r'} \times ((h^{d_B \times (\lambda \times (c')^{-1}+1)})^{e_B} / h)^{c'} \bmod n_B$
  $=(y^{e_B})^{r'} \times ((h^{(\lambda \times (c')^{-1}+1)}) / h)^{c'} \bmod n_B$
  $=(y^{e_B})^{r'} \times (h^{(\lambda \times (c')^{-1})})^{c'} \bmod n_B$
  $=(y^{e_B})^{r'} \times h^{\lambda} \bmod n_B$

Therefore, the equation $c'=\mathrm{H}(\mathrm{H}(m)\|A\|B\|t_g'\|t_y')$ will always pass for the forged VES values $\{U', V', c', r'\}$. However, when dispute occurs, the party A uses the forged VES $\{U', V', c', r'\}$ and C_{BT} to request receipt recovery, the party T will recover the error receipt $\sigma_B' \neq \sigma_B$. As the demonstrated aforementioned, the error receipt would be

$$\begin{aligned}\sigma_B' &= (V')/(U')^x \bmod n_B\\ &= (h^{d_B \times (\lambda \times (c')^{-1}+1)}) / (g^{\beta \times (c')^{-1}})^x \bmod n_B\\ &\neq \mathrm{H}(m)^{d_B} \bmod n_B.\end{aligned}$$

It is obviously to find that $\sigma_B' \neq \sigma_B$. Hence, it is unable to provide evidence because of $\mathrm{H}(m) \neq (\sigma_B')^{e_B} \bmod n_B$. The main weakness of Ma et al.'s protocol [11] is that party B can try to forge the values U' and V'. Therefore, we just needs to use $c=H(h\|A\|B\|t_g\|t_y\|U\|V)$ in place of original $c=H(h\|A\|B\|t_g\|t_y)$ in the Ma et al.'s protocol to overcome the weakness of unfairness. Although Ma et al.'s protocol can be easily revised, it still wastes too much computational cost.

(E1): $A \rightarrow B: h = H(m), \sigma_A = H(m)^{d_A} \bmod n_A$
(E2'): $B \rightarrow A: (U', V', c', r'), C_{BT}$
(E3): $A \rightarrow B: m$
(E4'): $B \rightarrow A:$ nothing

Figure 3. The forgery attack on Ma et al.'s protocol.

4. Our CEMD protocol

Our proposed protocol consists two phases: the main exchange phase and the receipt recovery phase. The notations and assumptions are as defined in Section 2 above. We assumed that the RSA-based receipt σ_B is re-defined as $\sigma_B = \mathrm{H}(m||I)^{d_B} \bmod n_B$, where the notation $I=(A, B, T, TimeStamp, info)$ is the unique session identify for each exchange phase. The notation *TimeStamp* means that the timestamp of seeding the e-mail message to against replay attack. The *info* contains the abstract and simple titles of the e-mail message used for authenticity of originator. The details of our CEMD protocol are described as follows.

4.1. Main exchange phase

Without loss of generality, we assume that part A attempts to send e-mail message m in exchange of its receipt σ_B from party B. The main exchange phase contains four Steps (M1)-(M4) as shown in Figure 4 and describes in following.

(M1): Party A sends the values $I=(A, B, T, TimeStamp, info)$, $h=H(m||I)$ and the signature $\sigma_A=\mathrm{H}(m||I)^{d_A} \bmod n_A$ to the designated party B.

(M2): After verifying the unique identity I and the RSA-based signature σ_A for h, the party B performs the following sub-steps to send the VES values (U, V, c, r) and C_{BT} back to the party A.

- **(M2-1)**: selects a random integer $\alpha \in Z^*_{n_B}$;
- **(M2-2)**: computes $\sigma_B=h^{d_B} \bmod n_B$;
- **(M2-3)**: computes $U=g^{d_B} \bmod n_B$; (The value U is pre-computable.)
- **(M2-4)**: computes $V=\sigma_B \times y^{d_B} \bmod n_B$, where the value $y=g^x \bmod n_B$ is obtained from C_{BT};
- **(M2-5)**: computes $R=g^{\alpha} \bmod n_B$;
- **(M2-6)**: computes $c=\mathrm{H}(I||h||U||V||R||y)$;
- **(M2-7)**: computes $r=\alpha - c \times d_B$;
- **(M2-8)**: sends the VES values $\{U, V, c, r\}$ and C_{BT} to party A. Note that, the value U is needless in sending the other mails to the same recipient.

(M3): After receiving $\{U, V, c, r, C_{BT}\}$, party A performs the following sub-steps to verify the VES. If the VES is valid, party A will send the real e-mail message m to party B. Note that, it is easily to use the public key encryption such as RSA [20] under party B's public key to protect e-mail message for confidentiality.

- **(M3-1)**: checks the signature s_{BT} of C_{BT};
- **(M3-2)**: verifies the equation $U^{e_B} \equiv g \pmod{n_B}$; Note that, this sub-step can be omitted while pre-computation supported.
- **(M3-3)**: verifies the equation $V^{e_B} \equiv \mathrm{H}(m||I) \times y \pmod{n_B}$;
- **(M3-4)**: computes $R=g^r \times U^c \pmod{n_B}$;
- **(M3-5)**: verifies $c=\mathrm{H}(I||\mathrm{H}(m||I)||U||V||R||y)$.
- **(M3-6)**: If all verifications above are passed, party A sends e-mail message m to party B. Otherwise, party A aborts the protocol.

(M4): After receiving the e-mail message m and verifying $h=H(m||I)$, party B sends back the real receipt σ_B to party A. Eventually, party A checks $\mathrm{H}(m||I)=\sigma_B^{e_B} \bmod n_B$. If it is valid, the certified e-mail delivery protocol is completed. Otherwise, party A initiates the receipt recovery phase described in the following Section 4.2.

(M1): $A \rightarrow B: I, h = H(m \,||\, I), \sigma_A = H(m \,||\, I)^{d_A} \bmod n_A$
(M2): $B \rightarrow A: (U, V, c, r), C_{BT}$
(M3): $A \rightarrow B: m$
(M4): $B \rightarrow A: \sigma_B$

Figure 4. Main exchange phase of our CEMD.

4.2. Receipt recovery phase

In the circumstance that party A fails to obtain the party B's receipt σ_B, party A may request for receipt recovery with the help of off-line TTP T. The steps including Step (T1) and Step (T2) of this phase are illustrated as following.

(T1) Party A sends the VES values (U, V, c, r), recovery certificate C_{BT} and e-mail m to party T.

(T2) Party T runs the same procedures as Step (M3) of main exchange phase. If all verification passed, party T recovers secret key $x=w_{BT} \times \mathrm{H}(sk_T||pk_{BT}) \bmod n_B$, and the real receipt $\sigma_B=V/U^x \bmod n_B$. Finally, the party T securely sends e-mail m to party B and the receipt σ_B to party A, respectively.

5. Security analyses

In this section, we demonstrates that our CEMD protocol can prevent all known security attacks including replay attack, existential forgery attack, and satisfies strong fairness property.

5.1. Replay attack

The unique identify $I=(A, B, T, TimeStamp, info)$ is embedded in both digital signature σ_A and σ_B. Legitimate expired time will be checked using the

timestamp. Hence, our proposed protocol not only can authenticate the identity of all participators, but also can resist the replay attack.

5.2. Existential forgery attack

As being pointed out in [24], our verifiable encrypted signature (VES) is based on variant RSA-based signature to design the existentially unforgeable signature. It is proven to be semantic security [24] against existential forgery attack. Moreover, anyone except party T is computational infeasible to derive the real receipt σ_B from the values (U, V) under the well-known difficulty of RSA problem [19]. Hence, the adversary including party A is unable to forge the VES (U, V, c, r).

5.3. Strong fairness

The strong fairness of our proposed CEMD protocol is achieved with the consideration for the following two cases:

(1) We assume that sender A has obtained the receipt before revealing the e-mail to party B.
(2) We assume that recipient B has been received the e-mail before sending its receipt.

In the first case, it implicitly means that the sender A has been received the receipt σ_B from Step (M4) of main exchange phase or has been recovered the receipt σ_B from Step (T2) of receipt recovery phase. Obviously, the recipient B had received the e-mail from Step (M3) of main exchange phase or Step (T2) of receipt recovery phase. Hence, fairness is achieved.

In the second case, it implies that the recipient B has been obtained e-mail from Step (M3) of main exchange phase or from Step (T2) of receipt recovery phase. Because the VES values (U, V, c, r) of our protocol is secure against existential forgery as demonstrated above in Section 5.2, the sender A must obtain the valid receipt from Step (M4) of main exchange phase or has a way of receiving the receipt using the recoverable VES with the help of an agreed off-line TTP in the receipt recovery phase. In addition, if recipient B unexpectedly aborts Step (M4) of main exchange phase after receiving the e-mail m, the sender A can initiate the receipt recovery phase to recover the real receipt σ_B by using recoverable VES $\{U, V, c, r\}$ and recovery certificate C_{BT}. In summary of two cases above, our CEMD protocol can satisfy the strong fairness property.

6. Performance evaluations

In Nenadic et al.'s CEMD protocol [13], the recipient can cheat the e-mail sender by sending an unrecoverable VES to pass all verifications [11]. Hence, the e-mail sender can not obtain the irrefutable receipt after sending the real e-mail to the dishonest recipient. As the demonstrated above, Ma et al.'s CEMD protocol [11] also exists weakness of unfairness and wastes too much computational cost. However, our protocol can support the pre-computation function. This feature will greatly reduce the computational cost and save communication overhead for continued e-mail delivery. As shown in Table 1, our CEMD protocol can reduce the computational cost about 30% than Ma et al.'s protocol and the communication overhead of VES is only 1280 bits in the same security level while sending the other mails to the same recipient.

7. Conclusions

This paper proposes a novel CEMD protocol. The proposed protocol provides pre-computation function for continued e-mail communication. Our protocol efficiently reduces about 30% computational costs than Ma et al.'s scheme. In addition, we point out the weakness in Ma et al.'s CEMD protocol and revise it for fairness.

Acknowledgement

This work was partially supported by the National Science Council, Taiwan, under the grants no. 97-2221-E-032-019.

Table 1. Comparisons of our and related CEMD protocols.

	Ours	Nenadic et al. [13]	Ma et al. [11]	Faster than Ma et al.
#exp in VES generation	3 **(2 for pre.)**	3	4	25% **(50% for pre.)**
#exp in VES verification	4 **(3 for pre.)**	3	5	20% **(40% for pre.)**
#exp in exchange phase	11 **(9 for pre.)**	9	13	15.38% **(30.77% for pre.)**
#exp in recovery phase	2+4=6	2+3=5	3+5=8	25%
The overhead for VES[1]	2304 bits	3072 bits	2304 bits	**1280 bits for pre. in ours**
Strong fairness	Yes	No	No	-

VES[1]: we assume that the overhead of traditional RSA signature encrypted in VES is 1024 bits.
pre.: it means pre-computation used for sending the other mails to the same recipient in our CEMD.
#exp: it stands for exponentiation operation times.

References

[1] B.B. Anderson, J.V. Hansen, P.B. Lowry, and S.L. Summers, "Standards and verification for fair-exchange and atomicity in e-commerce transactions," *Information Sciences*, vol. 176, pp.1045-1066, 2006.

[2] N. Asokan, M. Schunter, and M. Waidner, "Optimistic fair exchange of digital signatures," *IEEE Journal on Selected Areas in Communications*, vol. 18, pp.593-610, 2000.

[3] G. Ateniese, "Verifiable encryption of digital signatures and applications," *ACM Transactions on Information and System Security (TISSEC'04)*, vol. 7, pp.1-20, 2004.

[4] G. Ateniese, B. Medeiros, and M.T. Goodrich, "TRICERT: A distributed certified E-mail scheme," *Symposium on Network and Distributed Systems Security*, pp.30-39, 2001.

[5] F. Bao, G. Wang, J. Zhou, and H. Zhu, "Analysis and improvement of Micali's fair contract signing protocol," *Information Security and Privacy (ACISP'04)*, LNCS 3108, pp.176-187, 2004.

[6] S. Even, and Y. Yacobi, "Relations among public-key signature systems," Technical Report 175, 1980.

[7] K.B. Frikken, and M.J. Atallah, "Achieving Fairness in Private Contract Negotiation," *9th Financial Cryptography and Data Security (FC'05)*, LNCS 3570, pp.270-284, 2005.

[8] J.A. Garay, and C. Pomerance, "Timed Fair Exchange of Standard Signatures: [Extended Abstract]," *Financial Cryptography and Data Security (FC'04)*, LNCS 2742, pp.190-207, 2004.

[9] S. Gurgens, C. Rudolph, and H. Vogt, "On the security of fair non-repudiation protocols," *Information Security Conference (ISC'03)*, LNCS 2851, pp.193-207, 2003.

[10] S. Kremer, and O. Markowitch, "Selective receipt in certified e-mail," *Indocrypt'01*, LNCS 2247, pp.136-148, 2001.

[11] C. Ma, S. Li, K. Chen, and S. Liu, "Analysis and improvement of fair certified e-mail delivery protocol," *Computer Standards & Interfaces*, vol. 28, pp.467-474, 2006.

[12] A. Mukhamedov, and M. Ryan, "Improved multi-party contract signing," *11th Financial Cryptography and Data Security (FC'07)*, LNCS 4535, 2007.

[13] A. Nenadic, N. Zhang, and S. Barton, "Fair certified e-mail delivery," *ACM Symposium on Applied Computing-Computer Security Track*, pp.391-396, 2004.

[14] A. Nenadic, N. Zhang, B. Cheetham, and C. Goble, "RSA-based certified delivery of E-Goods Using Verifiable and Recoverable Signature Encryption," *Journal of Universal Computer Science*, vol. 11,pp.175-192, 2005.

[15] National Institute of Standards and Technology (NIST), *Digital signature standard*, FIPS Publication 186, 1994.

[16] National Institute of Standards and Technology (NIST), *Secure Hash Standard (SHS)*, FIPS Publication 180-2, 2002.

[17] R. Oppliger, "Certified mail: The next challenge for secure messaging," *Communications of the ACM*, vol. 47, pp.75-79, 2004.

[18] I. Ray, I. Ray, and N. Natarajan, "An anonymous and failure resilient fair-exchange e-commerce protocol," *Decision Support Systems*, vol. 39, pp.267-292, 2005.

[19] R.L. Rivest, "RSA Problem," *Encyclopedia of cryptography and security*, New York, Springer, pp. 532-536, 2005.

[20] R.L. Rivest, A. Shamir, and L.M. Adleman, "A method for obtaining digital signatures and public-key cryptosystems," *Communications of the ACM*, vol. 21, pp.120-126, 1978.

[21] S/MIME, Secure multipurpose Internet mail extensions, Available at *http://www.rsasecurity.com/standards/smime/*

[22] A. Schmidt, N. Kuntze, and C. Hett, "Non-repudiation in Internet Telephony," *IFIP International Information Security Conference*, pp.361-372, 2007.

[23] M.-H. Shao, G. Wang, and J. Zhou, "Some common attacks against certified email protocols and the countermeasures," *Computer Communications*, vol. 29 pp.2759-2769, 2006.

[24] N. Smart, *Cryptography, An Introduction, Second Edition*, Mcgraw-Hill College, 2006.

[25] W. Stallings, *Cryptography and Network Security: Principles and Practice, Third Edition*, Prentice-Hall, 2003.

[26] G. Wang, F. Bao, and J. Zhou, "On the Security of a Certified E-Mail Scheme," *Indocrypt'04*, pp.48-60, 2004.

[27] N. Zhang, Q. Shi, M. Merabti, and R. Askwith, "Practical and efficient fair document exchange over networks," *Journal of Network and Computer Applications*, vol. 29, pp.46-61, 2006.