Efficient and Secure Protocol in Fair Certified E-Mail Delivery

Ren-Junn Hwang and Chih-Hua Lai
Department of Computer Science and Information Engineering
Tamkang University
151, Ying-Chuan Road, Tamsui, Taipei County, Taiwan 25137, R.O.C.
Taipei County, Taiwan, R.O.C.
junhwang@ms35.hinet.net

Abstract: - An efficient and secure protocol in certified e-mail delivery is proposed in this paper. With the widespread use of public Internet, communication via electronic mail (e-mail) becomes a convenience application instead of traditional manuscript letter. People can easily append his/her digital signature to the e-mail in order to achieve the goal of non-repudiation of origin. However, the evidence of receipt still relies on the willingness of the recipient in the standard e-mail service. Hence, the recipient has no responsible for the received e-mail. In this paper, we present an efficient and secure protocol in fair certified e-mail delivery (CEMD). Our protocol efficiently provides non-repudiation of origin and receipt in the fair manner. In other words, the sender can obtain the irrefutable receipt if and only if the recipient gets the certified e-mail from the sender, otherwise, neither of them. Moreover, the proposed CEMD is efficiently in sending the other mails to the same recipient by using the pre-computation function. As the evaluations of computational cost and communication overhead, our protocol is cost-effective and efficient than other relevant protocols.

Key-Words: - Certified e-mail, Digital signature, Fair exchange, RSA signature, Security.

1 Introduction

With the growth of open network such as Internet, the problem of secure electronic transactions becomes more and more important issues. One of the applications via Internet is the electronic mail (e-mail) delivery service. Communication via e-mail turns heavy message delivery of traditional post office into the convenient ways. Moreover, the advantages in modern e-mail system is able to rapidly and inexpensively send the valuable messages such as software, digital products, financial report or purchase order. However, even though e-mail service is an increasingly popular application for business communication, it doesn't yet provide a mutual reliable infrastructure for sender and recipient [34]. Although the digital signature [2, 3] such as well-known RSA [37], S/MIME [43] or key management [22] can be easily appended to the e-mail system for non-repudiation of origin and confidentiality, the irrefutable receipt that e-mail was actually delivered to and received by its intended recipient is still based on the willingness of the recipient. In other words, the lack of undeniable receipt in the basic e-mail system may cause the sender has no way of proving that the e-mail had sent to the designated recipient. Therefore, the recipient has no responsible to the sender and the sender can not do much to prove the opposite.

Certified e-mail delivery protocol (CEMD) [33] is the reliable service developed to fairly exchange the certified e-mail of the sender and its irrefutable receipt from the recipient. Briefly speaking, CEMD aims for solving the problem of how two mutually distrustful parties can fairly exchange a valuable message from the sender for a recipient's digital signature representing a proof of receipt. Thus, the main purpose of CEMD is to achieve the fairness property in the sense that either the sender obtains the receipt from the recipient and the recipient accesses the message of e-mail simultaneously, or neither party gets the expected item.

Obviously, CEMD is a kind of fair exchange [5]. The issues of fair exchange contains following different but relevant variants [4, 45]: fair non-repudiation protocols [17, 21, 32, 38, 44], electronic contract signing protocols [9, 10, 18, 19, 26], certified e-mail delivery protocols [1, 23, 25, 28, 29], and fair document exchange protocols [35, 41, 48]. The main purpose of fair non-repudiation and electronic contract signing protocols are aimed for fairly exchanging respective irrefutable evidence, i.e., digital signature. The digital signature is the specific and verifiable item. Unfortunately, the fair non-repudiation protocols always assume resilient channels and can not be run on unreliable channels [17]. Moreover, an electronic contract signing protocol only allows the participants to fairly exchange the respective signature on the pre-agreed contract text. Hence, these protocols can not be designed for e-mail delivery. However, CEMD is developed to fairly exchange any message such as e-mail or digital document from the sender to its irrefutable receipt from the recipient. Thus, while the recipient indeed received the e-mail or digital document, the sender is capable of proving the evidence to the third party. In addition, fair document exchange protocol is implemented for fairly exchanging respective message of any format. Although fair document exchange protocols can be regarded as the generalization case of fair exchange, it is not the most efficient way to exchange only one e-mail message from the sender and its irrefutable receipt from the recipient. For more details about fair exchange, please refer to [6, 24].

Formally, the significant purpose of certified e-mail delivery protocol is to develop the reliable on-line delivery system to provide following main security requirements [25, 40]:

(1) *Non-repudiation of origin*: the recipient is capable of proving the irrefutable evidence that the e-mail was indeed sent by the sender. In other words, the e-mail has been signed from the sender and the certified e-mail is verifiable by the recipient.

(2) *Non-repudiation of receipt*: the sender is capable of proving the irrefutable evidence that the e-mail was indeed received by the recipient. In other words, the recipient must sign the received e-mail back to the e-mail sender.

(3) *Strong fairness*: at the end of CEMD protocol, either both parties get their expected items or neither do. In other words, any participator has no more advantage over the opposite party while any party misbehavers or prematurely aborts.

The involvement of trusted third party (TTP) between mistrusting parties is necessary for fairness assurance in the CEMD protocol [29]. Therefore, CEMD can be classified into following types:

(1) *In-line TTP-based* [8, 16]: An in-line TTP acts as an intermediary between the sender and the recipient. In other words, the in-line TTP can be regard as the centralized marketplace, and the message flows are fully controlled by the in-line TTP. The in-line TTP mediates a fair exchange. It takes the message from the sender and forwards it to the recipient, and vice versa. The sender and the recipient have no direct interaction. In the in-line TTP-based CEMD, we must assume that the in-line TTP will always send both messages of the sender and the recipient via Internet, simultaneously. The advantages are simplicity of the concept. The disadvantages are related to computational or communication bottleneck in the in-line TTP. Furthermore, the in-line TTP is difficult to guarantee that both participators are able to receive the exchanged items at the same time especially in the public Internet.

(2) *On-line TTP-based* [1, 33, 47]: Similar to the in-line TTP-based, the on-line TTP is actively involved during every transaction of exchange. However, the on-line TTP does not have to process entire messages. In other words, the sender and the recipient have a direct interaction. Thus, the on-line TTP is more efficient than in-line TTP. Unfortunately, the on-line TTP could be expensive for maintenance and will cause the communication bottleneck. Involving the on-line TTP in each message of the fair exchange protocol remarkably decreases the performance especially for multi-user environment.

(3) *Off-line TTP-based* [25, 28, 29]: Contrary to on-line TTP, an off-line TTP only had to intervene in case of dispute after main exchange phase. The off-line TTP just needs to restore the strong fairness property in the circumstance when dispute occurs. The advantage is that the off-line TTP is out of the loop of the exchange phase. The disadvantage is that the message size to be exchanged will be increased in order to provide arbitration and to maintain the strong fairness property.

Furthermore, according to the signed type of the receipt, the CEMD protocols also can be classified into RSA-based CEMD [25, 28, 29], DSA-based CEMD [30], and ID-based CEMD [12, 20, 46], etc. The receipt of RSA-based CEMD is the familiar RSA signature [37]. The receipt of DSA-based CEMD is using the digital signature standard [31], which is recommended by NIST. In addition, the receipt of ID-based CEMD is the short signature [13, 14] by using the cryptographic primitive of bilinear pairing. That is the one of the direction of digital signature in the future. However, RSA signature is the most universal digital signature technology used in the common e-mail system. Hence, in this article, we only focus on the familiar RSA signature and discuss the relevant RSA-based certified e-mail delivery protocols.

Generally, the technology of the verifiable encryption of a signature (VES) [7, 39] is the efficient and state-of-the-art solution to construct certified e-mail delivery protocol. The concept of VES technology is to provide verifiability and recoverability on the encrypted receipt. The verifiability ensures that the encrypted receipt can be verifiable without revealing the real signature. The recoverability permits that the real signature can be recovered from the encrypted receipt with the assistance of an agreed off-line TTP. Hence, the sender can firstly send the real e-mail message at ease while obtaining the valid VES generated from the recipient. At last, after receiving the real e-mail, the recipient will send back the real receipt to the sender in order to complete the protocol. Due to the recoverability of the VES, the e-mail sender can ask for recovering the real receipt if the receipt is not sent back from the recipient in the last step of the main exchange phase.

In this paper, the contributions contain twofold. At first, a novel CEMD protocol is designed by using the efficient VES technology. Our VES is based on the familiar RSA signature. Moreover, our CEMD can support pre-computation function to reduce not only the computational cost, but also the communication overhead in sending the other mails to the same recipient. Hence, our CEMD is more efficiently suitable for the common e-mail circumstance that the sender will always send a number of different e-mails to the same recipient frequently. Secondly, we point out the weakness of unfairness in Ma et al.'s CEMD protocol [25]. Due to the carelessness of design, the recipient can easily forge an unrecoverable VES in Ma et al.'s CEMD protocol. Thus, if the dishonest recipient gives up sending the real receipt after receiving the e-mail message, the off-line TTP is unable to recover the irrefutable receipt. It is unfair to the e-mail sender. In this article, we also revise Ma et al.'s protocol to achieve the strong fairness property. As the evaluation of performance below, our novel CEMD protocol can greatly reduce the computational cost about 30% than Ma et al.'s protocol in the same security level. Furthermore, the communication overhead of VES is only 1280 bits in sending the other mails to the same recipient while using the RSA-based receipt with 1024 bits. Hence, the message size of the receipt is not extended very much.

The remainder of the paper is organized as following. Section 2 gives some notations and assumptions used throughout the paper. We point out and revise the weakness of Ma et al.'s protocol [25] in Section 3. Afterward, we propose a novel CEMD with pre-computation function in Section 4. The security analyses and performance evaluations of our CEMD protocol is demonstrated in Section 5 and Section 6. Finally, briefly conclusions are given in Section 7.

2 Notations and Assumptions

Throughout this paper, the notations are defined in Section 2.1. Next, Section 2.2 gives the assumptions used in our and Ma et al.'s CEMD protocol [25].

2.1 Notations

- A, B, T: the unique identity of e-mail sender A, recipient B, and trusted third party T, respectively.
- H(.): a collision-resistant one-way hash function such as SHA-1 [27] with following properties:
  (1) for any message m, it is easily to compute H(m);
  (2) given H(m), it is computational infeasible to derive the message m;
  (3) given m, it is computational infeasible to find another $m' \neq m$ such that $H(m')=H(m)$;
- $x||y$: 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 task of party T is to ensure the strong fairness if the sender and recipient can not reach a fair completion of exchange themselves. We assume that 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. The 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 $g \in [1, n_B-1]$ is selected prime integer with large order, and $y=g^x \bmod n_B$ such that $x \in [1, n_B-1]$ is the randomly selected temporary key;
  - $w_{BT}=x \times \mathrm{H}(sk_T||pk_{BT})^{-1} \bmod n_B$, 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 protocol

Firstly, we review Ma et al.'s protocol [25] in Section 3.1. Afterward, we point out the weakness of Ma et al.'s protocol and revise it to achieve the strong fairness property in Section 3.2.

3.1 Ma et al.'s protocol

Ma et al.'s protocol includes the exchange phase and receipt recovery phase. The details are described below.

3.1.1 Exchange phase

We assume that party A attempts to obtain the irrefutable receipt σ_B=H(m)^{d_B} mod n_B from the recipient B after sending the e-mail message m. The off-line TTP is needless to involve into this phase. The exchange phase includes following Step (E1) to Step (E4). The message flows are shown in Fig. 1.

(E1): Party A sends the value h=H(m) and the signature σ_A=H(m)^{dA} mod n_A to party B, firstly.

(E2): After receiving the values {h, σ_A}, party B runs the following sub-steps to send back the VES values {U, V, c, r} and recovery certificate C$_{BT}$ to party A:

- **(E2-1)**: verifies whether H(m)=$\sigma_A{}^{e_A}$ mod n_A; if the equation is invalid, aborts the protocol.
- **(E2-2)**: randomly selects two integers $\alpha\in$[1, n_B-1] and $w\in$[1, n_B-1];
- **(E2-3)**: computes the real receipt σ_B=h^{d_B} mod n_B;
- **(E2-4)**: computes the value U=g^{α} mod n_B
- **(E2-5)**: computes the value V=$\sigma_B\times y^{\alpha}$ mod n_B;
- **(E2-6)**: computes the value t_g=g^w mod n_B;
- **(E2-7)**: computes the value t_y=$(y^{e_B})^w$ mod n_B;
- **(E2-8)**: computes the value c=H(h||A||B||t_g||t_y);
- **(E2-9)**: computes the value r=w–$c\times\alpha$;
- **(E2-10)**: sends the VES values {U, V, c, r} and the certificate C_{BT} to party A.

(E3): Party A performs the following sub-steps to verify the received values, and then sends the e-mail m to party B:

- **(E3-1)**: verifies the signature s_{BT} of C_{BT}; if the equation H(pk_{BT}||w_{BT}||e_B||n_B) = $(s_{BT})^{e_T}$ mod n_T is not hold, aborts the protocol, where the values {pk_{BT}, w_{BT}} are obtained from the certificate C_{BT}={pk_{BT}, w_{BT}, s_{BT}};
- **(E3-2)**: computes two values t_g=$g^r\times U^c$ mod n_B and t_y=$(y^{e_B})^r\times(V^{e_B}$/H($m$)$)^c$ mod n_B;
- **(E3-3)**: If the equation c=H(H(m)||A||B||t_g||t_y) holds, sends e-mail m to party B.

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

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

Fig. 1. Exchange phase of Ma et al.'s CEMD.

3.1.2 Receipt recovery phase

While party A fails to obtain the valid receipt σ_B, party A can perform the following Step (R1) and Step (R2) of receipt recovery phase. The message flows are illustrated in Fig. 2.

(R1): Party A sends the values {U, V, c, r, C_{BT}, m} to party T.

(R2): Party T runs the following sub-steps to check the values {U, V, c, r, C_{BT}} and try to recover the real receipt σ_B to send to the party A. The procedures of the verifications in sub-steps (R2-1) to (R2-3) are the same as the sub-steps (E3-1) to (E3-3) of the main exchange phase, which is described in Section 3.1.1 above.

- **(R2-1)**: verifies the signature s_{BT} of C_{BT}; if the equation H(pk_{BT}||w_{BT}||e_B||n_B) = $(s_{BT})^{e_T}$ mod n_T is not hold, aborts the protocol, where the values {pk_{BT}, w_{BT}} are obtained from the certificate C_{BT}={pk_{BT}, w_{BT}, s_{BT}};
- **(R2-2)**: computes two values t_g=$g^r\times U^c$ mod n_B and t_y=$(y^{e_B})^r\times(V^{e_B}$/H($m$)$)^c$ mod n_B;
- **(R2-3)**: If the equation c=H(H(m)||A||B||t_g||t_y) holds, recovers the secret key x=$w_{BT}\times$H(sk_T||pk_{BT}) mod n_B;
- **(R2-4)**: recovers the receipt σ_B by computing σ_B=V/$(U)^x$ mod n_B;
- **(R2-5)**: Finally, party T securely sends e-mail m to party B and sends receipt σ_B to party A simultaneously or using the out-of-the-band method.

$$\boxed{\begin{array}{l}\text{(R1)}: A \rightarrow T: \{U, V, c, r\}, C_{BT}, m \\ \text{(R2)}: T \rightarrow A: \sigma_B \text{ and } T \rightarrow B: m\end{array}}$$

Fig. 2. Receipt recovery phase of Ma et al.'s CEMD.

3.2 Weakness in Ma et al.'s protocol

In this section, we will point out the weakness of Ma et al.'s protocol [25]. In Ma et al.'s CEMD, unfortunately, party B is capable of cheating party A to obtain the e-mail m without releasing the real receipt σ_B. In other words, party B is able to easily forge the unrecoverable but verifiable VES values {U', V', c', r'} to deceive the party A. However, party T is unable to recover the real receipt σ_B from the forged VES values {U', V', c', r'}. Hence, the strong fairness can not be achieved in Ma et al.'s protocol.

The detail of forgery attack on Ma et al.'s protocol is described below. In the Step (E2) of exchange phase, party B can easily forge the unrecoverable VES values {U', V', c', r'} to pass all verifications. Thus, it will cause erroneous decision for party A to send back the real e-mail m to party B in the Step (E3) of exchange phase. In this moment, party B is sending nothing in the Step (E4) of exchange phase. Although party A can 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'}. In other words, the wrong receipt σ_B' can not be proven to the third party for the specific e-mail m. Hence, party B can deny the fact that the e-mail m is indeed received. This situation is unfair to the e-mail sender A.

Party B performs the following Step (E2') in place of Step (E2) of exchange phase to forge the unrecoverable VES values {U', V', c', r'}. The details of Step (E2') are described below and shown in Fig. 3.

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

(E2'-1): selects three distinct integer r', β and $\lambda \in [1, n_B - 1]$;

(E2'-2): computes $t_g' = g^{r'+\beta} \bmod n_B$;

(E2'-3): computes $t_y' = (y^{e_B})^{r'} \times h^{\lambda} \bmod n_B$;

(E2'-4): computes $c' = \mathrm{H}(h \| A \| B \| t_g' \| t_y')$;

(E2'-5): computes $U' = g^{\beta \times (c')^{-1}} \bmod n_B$, where $c' \times (c')^{-1} \equiv 1 \pmod{(p_B - 1)(q_B - 1)}$;

(E2'-6): computes $V' = h^{d_B \times (\lambda \times (c')^{-1} + 1)} \bmod n_B$ using the private key $sk_B = \{d_B, n_B\}$ of B;

(E2'-7): sends the forged VES values {U', V', c', r'} and the certificate C_{BT} to party A.

Therefore, party A will get the valid values {t_g', t_y'} to pass all verifications in Step (E3) of exchange phase. The correctness for the values {t_g', t_y'} is presented below:

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

Obviously, the equation $c' = \mathrm{H}(\mathrm{H}(m) \| A \| B \| t_g' \| t_y')$ would be always passed. However, when dispute occurs, party A sends forged VES values {U', V', c', r'} and C_{BT} to ask for receipt recovery. Party T will recover the wrong receipt

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

Hence, it is unable to provide non-repudiation of receipt because of $\mathrm{H}(m) \neq (\sigma_B')^{e_B} \bmod n_B$.

The main flaw in Ma et al.'s protocol is that party B can easily forge the values U' and V' after computing the value c'. Thus, we just need to use the value $c = H(h \| A \| B \| t_g \| t_y \| U \| V)$ in place of original value $c = H(h \| A \| B \| t_g \| t_y)$ to overcome the weakness of Ma et al.'s protocol. Although Ma et al.'s protocol can be easily revised, it still wastes too much computational cost.

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

Fig. 3. The forgery attack on Ma et al.'s CEMD.

4 Our novel CEMD protocol

Our novel certified e-mail delivery (CEMD) protocol consists two phases: the main exchange phase and receipt recovery phase. The notations and assumptions is identical to the definitions in Section 2 except 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 identity for each exchange phase. The notation *TimeStamp* means that the timestamp of seeding the e-mail to against replay attack. The *info* contains the abstract and simple titles of the e-mail used for authenticity of originator. The details of main exchange phase are described in Section 4.1 and receipt recovery phase is shown in Section 4.2.

4.1 Main exchange phase

Without loss of generality, we assume that part A attempts to send e-mail *m* in exchange of its receipt σ_B from party B. The main exchange phase contains four Steps (M1)-(M4) as shown in Fig. 4. The details of main exchange phase are described below:

(M1): Firstly, party A sends the values I=(A, B, T, *TimeStamp*, *info*), h=H(m||I) and the signature σ_A=H(m||I)^{d_A} mod n_A to party B.

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

- **(M2-1)**: selects an integer $\alpha \in$[1, n_B –1];
- **(M2-2)**: computes σ_B=h^{d_B} mod n_B;
- **(M2-3)**: computes U=g^{d_B} mod n_B; (The value U is pre-computable.)
- **(M2-4)**: computes V=σ_B×y^{d_B} mod n_B, where the value y=g^x mod n_B is obtained from C_{BT};
- **(M2-5)**: computes R=g^α mod n_B;
- **(M2-6)**: computes c=H(I||h||U||V||R||y);
- **(M2-7)**: computes r=α – c×d_B;
- **(M2-8)**: sends VES values {U, V, c, r} and C_{BT} to party A. The value U is needless in sending the other mails to the same recipient.

(M3): Party A performs the following sub-steps to verify the VES. If the VES is valid, party A sends the real e-mail m to party B. Note that, it is easily to use the public key encryption such as RSA [37] under party B's public key to protect e-mail for confidentiality.

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

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

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

Fig. 4. Main exchange phase of our CEMD protocol.

4.2 Receipt recovery phase

While party A fails to obtain the receipt σ_B of party B, party T can help party A to recover the receipt σ_B. The details including Step (T1) and (T2) are shown in following:

(T1): Firstly, 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 verifications passed, party T recovers secret key x = w_{BT}×H(sk_T||pk_{BT}) mod n_B, and the real receipt σ_B=V/(U)^x mod n_B. Finally, the party T securely sends e-mail m to party B and sends receipt σ_B to party A, simultaneously. If the values {U, V} is valid, the party T will always recovers the correct receipt σ_B, since:

$$\begin{aligned} \sigma_B &= V/(U)^x \bmod n_B \\ &= (\sigma_B \times y^{d_B} \bmod n_B)/(g^{d_B} \bmod n_B)^x \bmod n_B \\ &= (\sigma_B \times g^{x \cdot d_B})/(g^{d_B})^x \bmod n_B \\ &= \sigma_B \bmod n_B \end{aligned}$$

5 Security Analyses

In this section, we prove that our CEMD protocol can resist replay attack and existential forgery attack. Afterward, the strong fairness is demonstrated for our CEMD protocol.

5.1 Replay attack

The unique identity I=(A, B, T, *TimeStamp*, *info*) is signed by party A in the step (M1) of the main exchange phase and signed back by the party B in the step (M2) of the main exchange phase. The expired time of timestamp will be checked by both participators. Hence, our protocol not only can authenticate the identity of all participators, but also can prevent the attacker re-sending the eavesdropped transcripts to impersonate the legal participator. Hence, our CEMD can resistant replay attack.

5.2 Existential forgery attack

In this section, we prove that our novel CEMD is resilience of the existential forgery attack [11, 15]. The definition of existential forgery attack for the digital signature is defined in the following Definition 1.

Definition 1 (*Existential forgery attack*): The adversary can create at least one pair of message and the related digital signature, such that the signature of the message is valid and the message was not signed by the original signer.

In our CEMD, the verifiable encryption of the signature (VES) is the values (U, V, c, r). Obviously, the value U is the traditional RSA-based signature on the message h=H(m||I) and the value V is also the RSA signature on the integer g. After receiving values (U, V) in the Step (M3) of the main exchange phase, party A will verify both RSA signatures. The message m and the data I are chosen by party A in the Step (M1) of the main exchange phase, and the integer g is certified by the trusted third party T. Hence, the values (U, V) are unforgeable.

Moreover, even though party A can easily compute V^{e_B} mod n_B to get the value (h×y) mod n_B, it is computational infeasible for the adversary including part A to obtain the real receipt σ_B. Furthermore, the value R can be derived by computing R=g^r×U^c (mod n_B). As being pointed out in [42], the values (R, c, r) are under the well-known difficulty of RSA problem [36], and provable secure to against the existential forgery attack. Hence, the adversary including party A is unable to forge the VES values (U, V, c, r). In other words, our novel CEMD is secure under the existential forgery attack.

5.3 Strong fairness

In our novel CEMD, anyone can unexpectedly abort the procedure of main exchange phase. We prove that the strong fairness can be achieved in any circumstances.

In the first circumstance, we assume that party A already obtains the receipt σ_B. Due to the unforgeability of VES values (U, V, c, r) proven above, party A must receive the receipt σ_B from Step (M4) of the main exchange phase or recover it from the receipt recovery phase. Hence, party B also can get e-mail m from Step (M3) of the main exchange phase or can receive e-mail m from the receipt recovery phase. Thus, it is fair to both participators.

In the second circumstance, we assume that party B already obtains the e-mail m firstly. In this moment, party A must obtain VES values (U, V, c, r) from Step (M2) of the main exchange phase. Although party B can unexpectedly abort Step (M4) of the main exchange phase without revealing the real receipt σ_B. Party A can initiate the receipt recovery phase by using the VES values (U, V, c, r) with its certificate C_{BT}. Thus, after verifying the VES values, party T will help party A to recover the real receipt σ_B=V/(U)^x mod n_B. Therefore, our protocol satisfies strong fairness property.

6 Performance evaluations

We now analyze the computational cost and communication overhead of our protocol by comparing it with previous relevant RSA-based CEMD protocols [25, 28, 29].

In Nenadic et al.'s RSA-based CEMD [28, 29], the recipient can cheat the e-mail sender by sending an unrecoverable VES to pass all verifications. Ma et al. [25] point out that the VES used in Nenadic et al.'s CEMD can not be correctly verified during the execution because of careless design. Moreover, the dishonest party B can easily forge the unrecoverable VES. Hence, party T is unable to recover the real receipt for party A in Nenadic et al.'s RSA-based CEMD [28, 29].

As the demonstrated above, Ma et al.'s CEMD protocol [25] still exists weakness of unfairness. Moreover, Ma et al.'s CEMD protocol wastes too much computational cost. However, our novel CEMD is efficiently designed especially in sending the other mails to the same recipient. Due to the pre-computation function used in our novel CEMD, the computational cost and the communication overhead can be greatly reduced in the sustained e-mail delivery. As shown by Table 1 below, our CEMD protocol is more efficient than the relevant RSA-based CEMD protocols [25, 28, 29]. Obviously, our novel CEMD protocol can decrease the computational cost about 30% than Ma et al.'s CEMD protocol in the same security level. Furthermore, the communication overhead of VES in our novel CEMD is only 1280 bits while sending the other mails to the same recipient. Hence, it is more suitable for the practical e-mail delivery circumstance.

7 Conclusions

This paper proposes an efficient and secure protocol in certified e-mail delivery. The proposed novel

CEMD supports a pre-computation function in sending the other mails to the same recipient. As performance evaluations, our novel CEMD is cost-effective and efficient than relevant RSA-based CEMD protocols. Besides that, we point out and revise the weakness of unfairness for Ma et al.'s CEMD protocol. The proposed novel CEMD efficiently reduce the computational cost about 30% than Ma et al.'s CEMD protocol and decrease 1024 bits overhead in sending the other mails to the same recipient.

Acknowledgement

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

Table 1. Performance comparisons of our CEMD and relevant protocols.

	Our CEMD	Nenadic et al. [28, 29]	Ma et al. [25]	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 in our CEMD for continued e-mail delivery to the same recipient.
#exp: it stands for exponentiations operation times.

References:

[1] M. Abadi, N. Glew, B. Horne, and B. Pinkas, Certified Email with a Light On-Line Trusted Third Party: Design and Implementation, *Proceedings of International World Wide Web Conference*, 2002, pp. 387-395.

[2] S.J. Aboud, and M.A. Al-Fayoumi, A Secure Signature Scheme with a Broadband Covert Channel, *WSEAS Transactions on Computers*, Vol.6, Issue 2, 2007, pp. 367-372.

[3] M.A. Al-Fayoumi, and S.J. Aboud, An Efficient Concept in Digital Signature Schemes using Computational Delegation, *WSEAS Transactions on Computers Research*, Vol.1, Issue 1, 2006, pp. 25-30.

[4] 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, 2006, pp. 1045-1066.

[5] M. Apiromvorakarn, and Y. Permpoontanalarp, Undeniable Fair Exchange, *Proceedings of the 6th WSEAS International Conference on Multimedia, Internet & Video Technologies*, 2006, pp. 78-83.

[6] N. Asokan, M. Schunter, and M. Waidner, Optimistic Fair Exchange of Digital Signatures, *IEEE Journal on Selected Areas in Communications*, Vol.18, 2000, pp.593-610.

[7] G. Ateniese, Verifiable Encryption of Digital Signatures and Applications, *ACM Transactions on Information and System Security*, Vol.7, No.1, 2004, pp. 1-20.

[8] A. Bahreman, and J.D. Tygar, Certified Electronic Mail, *IEEE Proceedings of Internet Society Symposium on Network and Distributed System Security*, 1994, pp. 3-19.

[9] 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)*, Lecture Notes in Computer Science, Vol.3108, 2004, pp. 176-187.

[10] M. Ben-Or, O. Goldreich, S. Micali, and R.L. Rivest. A Fair Protocol for Signing Contracts, *IEEE Transactions on Information Theory*, Vol.36, 1990, pp. 40-46.

[11] G. Bleumer, Existential Forgery, *Encyclopedia of Cryptography and Security*, Springer, 2005.

[12] D. Boneh, C. Gentry, B. Lynn, and H. Shacham, Aggregate and Verifiably Encrypted Signatures from Bilinear Maps, *Proceedings of Eurocrypt'03*, Lecture Notes in Computer Science, Vol.2656, 2003, pp. 416-432.

[13] D. Boneh, H. Shacham, and B. Lynn, Short Signatures from the Weil Pairing, *Journal of Cryptography*, Vol.17, No.4, 2004, pp. 297-319.

[14] H.F. Chiang, S.M. Yen, and H.C. Lin, Security Analysis of Batch Verification on Identity-

Based Signature Schemes, *Proceedings of the 11th WSEAS International Conference on Computers*, 2007, pp. 50-55.

[15] H.Y. Chien, Forgery attacks on digital signature schemes without using one-way hash and message redundancy, *Communications Letters*, Vol.10, No.5, 2006, pp. 324-325.

[16] T. Coffey, and P. Saidha, Nonrepudiation with Mandatory Proof of Receipt, *ACM SIGCOMM Computer Comm. Rev.*, Vol.26, No.1, 1996, pp. 6-17.

[17] A.M. Lopez, D.D. Sanchez, F. Almenarez, C.G. Rubio, and C. Campo, Fair Multi-Party Contract Signing Using Private Contract Signatures, *Computer Networks*, Vol.51, 2007, pp. 2288-2298.

[18] K.B. Frikken, and M.J. Atallah, Achieving Fairness in Private Contract Negotiation, *The 9th Financial Cryptography and Data Security (FC'05)*, Lecture Notes in Computer Science, Vol.3570, 2005, pp. 270-284.

[19] J.A. Garay, and C. Pomerance, Timed Fair Exchange of Standard Signatures: [Extended Abstract], *Financial Cryptography and Data Security (FC'04)*, Lecture Notes in Computer Science, Vol.2742, 2004, pp.190-207.

[20] C. Gu, and Y. Zhu, An Id-Based Verifiable Encrypted Signature Scheme Based on Hess's Scheme, *Conference on Information Security and Cryptology (CISC'05)*, Lecture Notes in Computer Science, Vol.3822, 2005, pp. 42-52.

[21] S. Gurgens, C. Rudolph, and H. Vogt, On the Security of Fair Non-repudiation Protocols, *Information Security Conference (ISC'03)*, Lecture Notes in Computer Science, Vol.2851, 2003, pp.193-207.

[22] R.J. Hwang, C.H. Lai, and Y.R. Chen, Secure Key Management Scheme for Wireless LAN, *WSEAS Transactions on Communications*, Vol.5, Issue 9, 2006, pp. 1800-1807.

[23] K. Imamoto, and K. Sakurai, A Certified Email System with Receiver's Selective Usage of Delivery Authority, *Progress in Cryptology-INDOCRYPT'02*, Lecture Notes in Computer Science, Vol.2551, 2002, pp. 326-338.

[24] S. Kremer, and O. Markowitch, Selective Receipt in Certified E-Mail, *Progress in Cryptology-INDOCRYPT'01*, Lecture Notes in Computer Science, Vol.2247, 2001, pp.136-148.

[25] C. Ma, S. Li, K. Chen, and S. Liu, Analysis and Improvement of Fair Certified E-Mail Delivery Protocol, *Computer Standards & Interfaces*, Vol.28, 2006, pp.467-474.

[26] A. Mukhamedov, and M. Ryan, Improved Multi-Party Contract Signing, *Financial Cryptography and Data Security (FC'07)*, Lecture Notes in Computer Science, Vol.4535, 2007.

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

[28] A. Nenadic, N. Zhang, and S. Barton, Fair Certified E-Mail Delivery, *ACM Symposium on Applied Computing-Computer Security Track*, 2004, pp.391-396.

[29] 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, 2005, pp.175-192.

[30] A. Nenadic, N. Zhang, Q. Shi, and C. Goble, Certified E-Mail Delivery with DSA Receipts, *Proceedings of 19th IEEE International Parallel and Distributed Processing Symposium (IPDPS'05)*, Vol.1, 2005, pp. 4-8.

[31] National Institute of Standards and Technology (NIST), *Federal Information Processing Standards Publication 186-2, Digital Signature Standard (DSS)*, January 2000.

[32] J. Onieva, J. Lopez, R. Roman, J. Zhou, and S. Gritzalis, Integration of Non-repudiation Services in Mobile DRM Scenarios, *Telecommunication Systems*, Vol.35, No.3-4, 2007, pp. 161-176.

[33] R. Oppliger, Certified Mail: The Next Challenge for Secure Messaging, *Communications of ACM*, Vol.47, No.8, 2004, pp. 75-79.

[34] R. Oppliger, Providing Certified Mail Services on the Internet, *IEEE Security and Privacy*, Vol.5, No.1, 2007, pp. 16-22.

[35] I. Ray, I. Ray, and N. Natarajan, An Anonymous and Failure Resilient Fair-Exchange E-Commerce Protocol, *Decision Support Systems*, Vol.39, 2005, pp.267-292.

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

[37] 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, 1978, pp. 120-126.

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

[39] Z. Shao, Certificate-Based Verifiably Encrypted Signatures from Pairings, *Information Sciences*, Vol.178, 2008, pp. 2360-2373.

[40] M.H. Shao, G. Wang, and J. Zhou, Some Common Attacks Against Certified Email Protocols and the Countermeasures, *Computer Communications*, Vol.29, 2006, pp.2759-2769.

[41] Q. Shi, N. Zhang, and M. Merabti, Signature-Based Approach to Fair Document Exchange, *IEE Proceedings of Communications*, Vol.150, 2003, pp. 21-27.

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

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

[44] M. Ventuneac, T. Coffey, and T. Newe, Reasoning on Properties of Non-Repudiation Security Protocols, *WSEAS Transactions on Information Science and Applications*, Vol.1, Issue 5, 2004, pp. 1262-1267.

[45] G. Wang, F. Bao, and J. Zhou, On the Security of a Certified E-Mail Scheme, *Progress in Cryptology-INDOCRYPT'04*, Lecture Notes in Computer Science, Vol.3348, 2004, pp. 48-60.

[46] F. Zhang, R. Safavi-Naini, and W. Susilo, Efficient Verifiably Encrypted Signature and Partially Blind Signature from Bilinear Pairings, *Progress in Cryptology-INDOCRYPT'03*, Lecture Notes in Computer Science, Vol.2904, 2003, pp. 191-204.

[47] N. Zhang and Q. Shi, Achieving Nonrepudiation of Receipt, *The Computer Journal*, Vol.39, No.10, 1996, pp. 844-853.

[48] 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, 2006, pp.46-61.