A Fair Non-transfer Exchange Protocol with Off-line TTP*

Cheng-Chi Lee[‡] Shu-Yin Hsiao[§] Min-Shiang Hwang[†]

Department of Management Information Systems[†]
National Chung Hsing University
250 Kuo Kuang Road, 402 Taichung, Taiwan, R.O.C.
Email: mshwang@nchu.edu.tw
Fax: 886-4-22857173

Department of Library and Information Science[‡]
Fu Jen Catholic University
510 Jhongjheng Road, Taipei 24205, Taiwan, R.O.C.
Email: cclee@mail.fju.edu.tw

Graduate Institute of Networking and Communication Engineering[§]
Chaoyang University of Technology
168 Gifeng E. Rd., Wufeng,
Taichung County, Taiwan 413, R.O.C.

October 31, 2012

*This research was partially supported by the National Science Council, Taiwan, R.O.C., under contract no.: NSC98-2221-E-005-050-MY3 and NSC99-2221-E-030-022.
[†]Responsible for correspondence: Prof. Min-Shiang Hwang.

A Fair Non-transfer Exchange Protocol with Off-line TTP

Abstract

Network exchange is now widely used. However, it still cannot avoid the problems evolving from network exchange. For example. A buyer may not receive the order even if he/she makes the payment. For another example, the seller possibly get nothing even when the merchandize is sent. Some studies about the fair exchange have proposed protocols for the design of efficiency and exploited the signature property to specify that two parties agree on the exchange. The information about purchased item and price are disclosed in this way. This paper proposes a new fair network payment protocol with off-line trusted third party. The proposed protocol can protect the buyers' purchase message from being traced. In addition, the proposed protocol can meet the proposed requirements. The most significant feature is *Non-transfer* property we achieved.

Keywords: E-commerce, digital signature, fair exchange, security.

1 Introduction

With the growth of open network and Internet in particular, many solutions have been proposed to solve the security related problems. There are still many risks on Internet, such as information loss happened in the communication of both sides or interrupted by one side maliciously. It hardly makes people to believe the security on Internet. It could lead to heavy losses for customers, especially E-commerce, if the above described conditions happen.

Recently, the speedy growth of the network encourages the study of security problems, in which fair exchange of electronic information is a vital one.

Fairness must be ensured in the procedure of exchange. In other words, one party in the protocol cannot take advantages of the other party if the protocol is halted for any sake. The electronic commerce is one of the practical applications. A payment protocol must ensure the fairness of exchange processes. A buyer may not receive the order even if he's/she's made the payment, or the seller possibly get nothing even when the merchandize is sent. This payment protocol must avoid the situation in which a party P_a obtain an expected item from another party P_b, but P_b doesn't get the expected item from P_a, and vice versa.

1.1 Related Work

Two approaches have been proposed to achieve fair exchange. The first one is that two parties exchange data simultaneously [10, 20]. It has two drawbacks. Firstly, it requires many steps of interactions for exchanging data. Secondly, one party will have chance to take advantage of the other if it maliciously aborts this deal in the middle of the protocol [23]. The second approach is that a trusted third party (TTP) is introduced in the exchange process. No matter in what way the TTP is involved in the protocol, its role is to resolve the problems may occur between the two parties. According to its involvement level in a protocol, a TTP can be seen either online or off-line. Online TTP are both involved in any instance of a protocol [6, 8, 13, 24]. However, online TTP would become a bottleneck during the communication of these two parties. Therefore, the off-line TTP has been proposed to improve the performance [1, 3, 4, 7, 11, 12, 25]. An off-line TTP is used when the participants in a protocol are supposed to be honest enough. And it's not made to ask for external help in order to achieve fairness; the TTP will only be involved if some problem emerges. Protocols with such a TTP are also known as *optimistic*.

However, the previous papers seem to achieve fair exchange only by using fair exchange on signatures. The information about purchased item and price are disclosed in this way. But they doesn't consider how to protect the buyer's purchase privacy. Therefore, Wang propose a complete solution for fair payment containing payment actions, such as electronic cash or network credit card method [23]. Their proposed protocol is the first work to provide a protection on buyer's privacy. It can be regarded as a process of fairly exchanging electronic coins (e-coins) and secret information. The four contributions are as follows.

1. Propose a generic model for real fair network payments.
2. Apply a subtle tool of Restrictive Confirmation Signature Scheme (RCSS) to achieve the property of untraceability.
3. Design a new technique of pseudo e-coin to achieve fairness of exchanging the electronic cash.
4. Demonstrate how to construct a practical and efficient fair network payment protocol based on the Brands' e-cash scheme [5].

However, Bao proposes a simple colluding attack to defy the Wang's protocol in 2004 [2]. He demonstrates that the fairness is breached under a simple colluding attack. That is to say, a dishonest merchant can obtain the digital money and a buyer cannot obtain their goods. Besides, Wang's protocol cannot achieve Non-transfer. Till to now, no researchers discuss and propose a solution of Non-transfer. What is Non-transfer? In Non-transfer, duplicating is a property of digital data. The proposed protocol should prevent a customer from giving away the merchandise to another customer while pretending to be a merchant.

1.2 Requirements

This paper proposes a new fair network payment protocol with off-line TTP based on the Brands's protocol [5]. The proposed fair exchange protocol can avoid Bao's colluding attack and implements the following requirements between the two exchanging parties:

1. *Effectiveness:* If no message is lost and none is a regular procedure, both parties can obtain the item they desired.

2. *Fairness:* One party in the protocol cannot take any advantages of the other, even if the protocol is halted for any reason.

3. *Timeliness:* One party delivers the item to the other in finite amount of time.

4. *Non-repudiability:* No one can repudiate once the participants start the exchanges.

5. *Verifiability:* When one party addresses the dispute to TTP, TTP must be able to verify the items are valid.

6. *Recoverability:* If one party does not receive the item or receive an invalid item, he/she can propose a dispute to TTP. TTP will be able to recover the factor.

7. *Non-transfer:* Duplicating is a property of digital data. The proposed protocol prevents a customer from giving away the merchandise to another customer while pretending to be a merchant. Non-transfer is required for legal constraint on the consumer's activities.

8. *Anonymous:* The consumer usually doesn't want the merchandise or the identity to be known on Internet. It needs to protect personal privacy by means of anonymous.

Note that the most significant feature of this paper is *Non-transfer* property we achieved. To achieve this only property, it can use a standard public-key cryptosystem to link the description of the merchandise's ID. However, in this paper, we do not only solve this property, but also achieve the all above properties. Therefore, we propose a fair non-transfer exchange protocol to meet the proposed requirements.

1.3 Attacks

Here, we define some attacks to help the readability of the paper. The detail of security analysis is shown in Section 3. Our proposed protocol should withstand the defined attacks. Each attack is an important and independent attack for a new fair exchange protocol.

A1. If some situations occur during information delivery (e.g. message loss or intercepting), that will lead to some loss of any parties of exchange or the third party reaping profits in the process.

A2. One party in the protocol can take advantage of the other party.

A3. The two parties of exchange cannot receive the items in finite and forfeit the advantage.

A4. The merchant cannot repudiate once the participants starts the exchange.

A5. The adversary can addresses the valid items to dispute with TTP.

A6. While the consumer proposes a dispute to TTP, TTP is incapable to recovery the key for consumer.

A7. The consumer can duplicate the digital data and transfer it to other after buying it from the merchant.

A8. The identity of the consumer can be detected and cannot protect personal privacy of the consumer by anonymity.

A9. The merchant M wants to collude with his/her conspirator C. After M receives the pseudo e-coins from the buyer U, M brings the pseudo e-coins to TTP but claims that the trade is between C and M. Then the

TTP will convert the e-coins to equivalent true ones for M and send the soft goods to C, but U will gain nothing [2].

1.4 Organization

The remainder of this paper wee be organized as follows. In next section, we shall propose a fair non-transfer exchange protocol for E-commerce. The security analysis of the proposed protocol is discussed in Section 3. Finally, the conclusions will be drawn in Section 5.

2 The Proposed Protocol

In this session, a new protocol based on Brands's protocol [5] is proposed to meet all the requirements in Introduction. The proposed protocol is divided into six parts to give explanation respectively: Setup, Account Opening, Registration, Withdrawal, Payment, and Dispute. The encryption/decryption and signature in the proposed protocol adopt the symmetric and public key cryptosystems and digital signature schemes [9, 14, 15, 16, 21].

2.1 Setup

Let p and q be two large primes and $q|p-1$. The bank B publishes a generator-tuple (g, g_1, g_2) in G_q and two collision-resistant hash function $H : G_q \times G_q \times G_q \times G_q \times G_q \times G_q \to Z_q^*$ and $H_0 : G_q \times G_q \times ID \times DATE/TIME \to Z_q^*$. B also generates a random number $x_B \in Z_q^*$ as a secret key and a public key $y_B = g^{x_B} \bmod p$. B's secret key and public key are (x_B) and $(p, q, g, g_1, g_2, H, H_0, y_B, g_1^{x_B} \bmod p, g_2^{x_B} \bmod p)$. Besides, a TTP is set up to resolve the problems that may occur between the parties.

2.2 Account Opening

The buyer U select a random value $u_1 \in Z_q^*$ and transmits $I = g_1^{u_1} \bmod p$ to B if $Ig_2 \neq 1$. The identifier I used to uniquely identify U can be regarded as the account number of U. U can compute $z = (Ig_2)^{x_B} = (g_1^{x_B})^{u_1} g_2^{x_B} \bmod p$.

2.3 Registration

Suppose that merchant M has a valuable item Goods and then registers *Goods* to TTP. TTP should check the validity of the received *Goods* at this stage. The registration procedure is described in the following steps and Figure 1.

Step 1. M chooses a symmetric key K for the encryption/decryption in the symmetric key cryptosystem and encrypts the K and *Goods* by using the TTP's public key y_T and encryption function PE through the public key cryptosystem. Then M sends it and the description of *Goods* to TTP. To avoid confusing the notations, we define the encryption and decryption function in symmetric key cryptosystem and public key cryptosystem as $(SE(\cdot), SD(\cdot))$ and $(PE(\cdot), PD(\cdot))$.

Step 2. When TTP receives the items sent by M, TTP can derive the *Goods* and K by using $PD(\cdot)$ and his/her secret key x_T to calculates the following equation:

$$\begin{aligned} ED &= SE_K(Goods), \\ RK &= PE_{y_T}(K), \\ hd &= H(ED), \quad \text{and} \\ Cert_G &= PD_{x_T}(desc, M, hd, RK, H(K)). \end{aligned}$$

Afterward TTP sends the certificate of Goods to M and publishes the certificate in the public directory.

2.4 Withdrawn

The following protocol is preformed while U wants to withdrawal e-cash from the bank.

$$M \xrightarrow{desc, PE_{y_T}(Goods, K)} TTP$$

$$M \xleftarrow{Cert_G} TTP$$

Figure 1: The procedure of Registration

Step 1. B selects a random value $\omega \in Z_q^*$ and sends $e_1 = (g)^\omega \bmod p$ and $e_2 = (Ig_2)^\omega \bmod p$ to U.

Step 2. U randomly chooses $s, x_1, x_2 \in Z_q^*$ and computes $A = (Ig_2)^s \bmod p$, $D = g_1^{x_1} g_2^{x_2} \bmod p$ and $z' = z^s \bmod p$. U also selects random values u, v, and $t_c \in Z_q^*$ and computes $e'_1 = e_1^u g^v \bmod p$, $e'_2 = e_2^{su} A^v \bmod p$, and $(a_c, b_c) = (g^{t_c} \bmod p, y_T^{t_c} \bmod p)$, where $y_T = g^{x_T} \bmod p$ is TTP's public key and x_T is TTP's secret key. Then U sends $c = c'/u \bmod q$ to B, where $c' = H(A, D, z', e'_1, e'_2, b_c) + a_c \bmod q$.

Step 3. B sends $r = cx_B + \omega \bmod q$ to U.

Step 4. U verifies whether $g^r = y_B^c e_1 \bmod p$ and $(Ig_2)^r = z^c e_2 \bmod p$. If verification result is true, U computes $r' = ru + v \bmod q$. Note $< A, D, (z', e'_1, e'_2, r', a_c, b_c) >$ represents a pseudo e-cash.

2.5 Payment

The buyer U and the merchant M exchange the electronic money and goods in this procedure. We assume U and M achieve to an order agreement that deal with merchandise items and price. And U can get the hd and $H(K)$ from the public directory of TTP. The payment procedure is described in Figure 2.

Step 1. U chooses a session key α and order number sn, then computes $H(sn)$. U encrypts $H(sn)$, pseudo e-coins $< A_i,\ D_i,\ (z'_i,\ e'_{1i},\ e'_{2_i},\ r'_i,\ a_{ci},\ b_{ci}) >$, for $i = 1, 2, \cdots, n$, and α using PK_M and $PE(\cdot)$. Note that U and M have a pair keys (secret key and public key) that are (SK_U, PK_U) and (SK_M, PK_M) to use public key cryptosystem. And note that n denotes the number of e-coins for the goods which U asks for. U sends the encryption item to M.

Step 2. M decrypts the received encryption item by using SK_M and $PD(\cdot)$. If M agrees to sell goods to U, and the following verifications hold of pseudo e-coins, M will generate a signature $S_\delta = PD_{SK_M}(RK, H(sn), H(A_1 \| A_2, \cdots, \| A_n), H(D_1 \| D_2, \cdots, \| D_n))$:

$$\begin{aligned} g^{r'_i} &= y_B^{H(A_i, D_i, z'_i, e'_{1i}, e'_{2i}, b_{ci}) + a_{ci}} e'_{1i}, \quad \text{and} \\ A_i^{r'_i} &= z'^{H(A_i, D_i, z'_i, e'_{1i}, e'_{2i}, b_{ci}) + a_{ci}}_i e'_{2i}. \end{aligned}$$

Then M encrypts the signature and ED by using the session key α and $SE(\cdot)$. After that, an encryption item is sent to U.

Step 3. U decrypts the S_δ, ED by using the session key α and $SD(\cdot)$. If $hd = H(ED)$, U computes the confirmation parameters $d_i = H_0(A_i,\ D_i,\ ID_M,\ date/time)$, $k_{1i} = d_i(u_i s_i) + x_{1i} \bmod q$, $k_{2i} = d_i s_i + x_{2i} \bmod q$, for $i = 1, 2, \cdots, n$, and then encrypts and sends them and t_{ci} to M. Here, we assume that the buyer knows the merchant identity ID_M before transaction.

Step 4. M decrypts the received encryption item and verifies the parameters by using the following equations:

$$\begin{aligned} g_1^{k_{1i}} g_2^{k_{2i}} &= A_i^{di} D_i, \\ a_{ci} &= g^{t_{ci}} \bmod p, \quad \text{and} \\ b_{ci} &= y_T^{t_{ci}} \bmod p. \end{aligned}$$

If they hold, M encrypts the K and sends it to U.

U decrypts the received encryption item and verifies the key K. U computes $H(K)$ and compares the two $H(K)$s, which is from $Cert_G$. If it holds, U

U $\qquad\qquad\qquad$ M

$$\xrightarrow{PE_{PK_M}(H(sn), < A_i, D_i, (z'_i, e'_{1i}, e'_{2_i}, r'_i, a_{ci}, b_{ci}) >, \alpha)}$$

$$\xleftarrow{SE_\alpha(S_\delta, ED)}$$

$$\xrightarrow{SE_\alpha(H(sn), d_i, k_{1i}, k_{2i}, t_{ci})}$$

$$\xleftarrow{SE_\alpha(K)}$$

Figure 2: The procedure of Payment

derives the Goods from $SD_K(ED) = SD_K(SE_K(Goods)) = Goods$ and ends the protocols. Otherwise U proposes the dispute to TTP. In Figure 2, we can see that the merchant can receive the money first, and then to deliver the goods(key) to the buyer. Under such condition, the buyer may not get the key. If this condition occurs, the dispute phase starts.

2.6 Dispute

This procedure is described in Figure 3.

Step 1. U encrypts the signature of that M agreement sale S_δ, order number sn, the true e-coins $< A_i, D_i, (z'_i, e'_{1i}, e'_{2i}, r'_i, a_{ci}, b_{ci}), (d_i, k_{1i}, k_{2i}, t_{ci}) >$ for $i = 1, 2, \cdots, n$, and session key β, and then sends it to TTP.

Step 2. When TTP receives U's request, it starts to verify whether the signature S_δ and e-coins are valid. If the result is true, TTP decrypts K from RK and sends $SE_\beta(K)$ to U.

Step 3. TTP also sends $PE_{PK_M}(< A_i, D_i, (z'_i, e'_{1i}, e'_{2i}, r'_i, a_{ci}, b_{ci}), (d_i, k_{1i}, k_{2i}, t_{ci}) >)$ to M that encrypts with M's public key PK_M.

This procedure is performed while U and M get nothing from the other. Since TTP has no way to tell the reall owner of the e-coins, how can TTP make sure that the merchant is the attacker. The reason is that when U do not get K to derive goods, U asks TTP to send it to him/her. This is why our scheme needs a registration phase.

$$\mathrm{U} \xrightarrow{PE_{y_T}(S_\delta, sn, < A_i, D_i, (z'_i, e'_{1i}, e'_{2i}, r'_i, a_{ci}, b_{ci}), (d_i, k_{1i}, k_{2i}, t_{ci}) >, \beta)} \mathrm{TTP}$$

$$\mathrm{U} \xleftarrow{SE_\beta(K)} \mathrm{TTP}$$

$$\mathrm{M} \xleftarrow{PE_{PK_M}(< A_i, D_i, (z'_i, e'_{1i}, e'_{2i}, r'_i, a_{ci}, b_{ci}), (d_i, k_{1i}, k_{2i}, t_{ci}) >)} \mathrm{TTP}$$

Figure 3: The procedure of Dispute

3 Security Analysis

In this section, some attacks will be analyzed to demonstrate the security of the proposed protocol.

Attack 1: If some situations occur during information delivery (e.g. message loss or intercepting), that will lead to some loss of any parties of exchange or the third party reaping profits in the process.

Analysis 1: Consumer sends the purchase information to merchant, and we suppose there exists loss in the transmission of information in Steps 1 $\sim$ 3 of the Payment phase: In case of message loss in Step 1, M cannot receive the request message, and we can assume M does not know anything. Under this circumstance, the Payment procedure cannot be called already finished. Moreover, if the adversary obtains message of Steps 1 $\sim$ 3, it is as difficult as breaking discrete logarithms [9, 17, 18] or factoring [21, 22] to obtain M's private key. In this way, the adversary does not know any information. If it is still in time duration period and the sender does not terminate the procedure, the sender can resend the message again.

Attack 2: One party in the protocol can take advantage of the other party.
Analysis 2: It most likely brings about the risk of exchange process, in which the consumer may not receive the goods after payment or the merchant may not receive the payment of goods after delivering the goods. If the merchant does not deliver the decrypt key of goods to the consumer after receiving the payment in payment procedure Step 3, the consumer can issue a certificate to TTP by demonstrating the mutual agreement on exchange with consumer identity, not adversary. TTP must verify the items addressed by the consumer, including the signature of merchant, sn, and e-cash. Suppose that the TTP approves the signature and e-cash while receives items, TTP can decrypt key for consumer. Besides, TTP cannot get any information when delivering message between consumer and merchant. Assume that an adversary wants to derive session key from message delivery procedure, the adversary must derive merchant's private key from the corresponding public key in advance. It is difficult as breaking discrete logarithms or factoring problem.

Attack 3: The two parties of exchange cannot receive the items in finite and forfeit the advantage.
Analysis 3: In the system, the tolerable time is set before the parties of exchanging deliver items to the other party.

Attack 4: The merchant cannot repudiate once the participants starts the exchange.

Analysis 4: In payment procedure, the merchant has to sign the signature S_δ and send it to the consumer once the merchant receives the exchange request from consumer. The fundamental of digital signature is no one can forge the signature of the merchant. So the merchant cannot repudiate by the same token.

Attack 5: The adversary can addresses the valid items to dispute with TTP.
Analysis 5: The items addressed by the consumer include the signature of the merchant, session number and e-cash. These items had been encrypted during transmission process antecedently. Therefore TTP cannot know the transmission contents. As regards forging these items, it is impossible according to the forward Analysis 4, TTP cannot forge the signature of merchant and e-cash since the e-cash is signed by the banker (TTP). Furthermore, the sn is unknown since the $H(sn)$ is the hash value of sn. Besides, the merchant just know the $H(sn)$, and no one can find another value $sn'(\neq sn)$ satisfying $H(sn') = H(sn)$ under one-way hash function [19] security requirement.

Attack 6: While the consumer proposes a dispute to TTP, TTP is incapable to recovery the key for consumer.
Analysis 6: If the items proposed by the consumer are verified by TTP, TTP can recover the key for the consumer. Since the signature of the merchant contains $RK(= PE_{y_T}(K))$, TTP surely can recover the key K.

Attack 7: The consumer can duplicate the digital data and transfer it to other after buying it from the merchant.
Analysis 7: In proposed protocol, the seller has to address ED which is element of $hd(= H(ED))$. Furthermore, the consumer must be able to sign the signature as the merchant M, because the identity M has been stated in the

certificate of Goods. Hence, the consumer cannot duplicate the digital data and transfer it to other after buying it from the merchant.

Attack 8: The identity of the consumer can be detected and cannot protect personal privacy of the consumer by anonymity.

Analysis 8: In proposed protocol, the consumer does not need to produce relation to the identity of itself. The e-cash of the items addressed by the consumer is an untraceable payment tool. No one knows the identity of the consumer even if the consumer engage in exchange repeatedly.

Attack 9: The merchant M wants to collude with his/her conspirator C. After M receives the pseudo e-coins from the buyer U, M brings the pseudo e-coins to TTP but claims that the trade is between C and M. Then the TTP will convert the e-coins to equivalent true ones for M and send the soft goods to C, but U will gain nothing [2].

Analysis 9: In our Payment protocol, when the merchant M obtains the valid e-coins $< A_i, D_i, (z'_i, e'_{1i}, e'_{2i}, r'_i, a_{ci}, b_{ci}), (d_i, k_{1i}, k_{2i}, t_{ci}) >$, the buyer U obtained the (S_δ, ED). Though M may collude with a conspirator, the buyer U can use our Dispute protocol to get K and then obtains his/her goods. In addition, S_δ is a signature signed by M. Once the dispute occurs, TTP can know which merchant is a attacker. Hence, the Bao's colluding attack cannot work successfully in this proposed protocol. In addition, we analyze the buyer-side colluding attack. That is, a dishonest buyer may collude with a counterfeit merchant. I think this attack cannot be successful, since the merchant's identity M has been put into goods.

4 Performance Analysis

In this section, the computational complexity of our scheme is analyzed. To analyze the computational complexity of our scheme, we first define the following notations.

T_{exp}: the time for computing a modular exponentiation operation;

T_{PKC}: the time for computing the public key cryptosystem;

T_{SKC}: the time for computing the symmetric key cryptosystem;

T_{mul}: the time for computing the multiplication of two numbers;

T_{inv}: the time for computing a modular inversion operation;

T_h: the time for computing one-way hash function.

We show the computational complexities of our scheme in Table 1. In the Setup Phase, B compute the public keys by requiring $3 \times T_{exp}$. In the Account Opening Phase, U compute I and z by requiring $2 \times T_{exp} + 1 \times T_{mul}$.

In the Registration Phase, M computes the $PE_{y_T}(Goods, K)$ by requiring $1 \times T_{PKC}$. Then, TTP decrypts the $Goods$ and K by requiring $1 \times T_{PKC}$. Finally, compute ED, RK, hd, and $Cert_G$ by requiring $2 \times T_{PKC} + 1 \times T_{SKC} + 1 \times T_h$.

In the Withdrawn Phase, B computes e_1, e_2, and r by requiring $2 \times T_{exp} + 2 \times T_{mul}$. U computes A, D, z', e'_1, e'_2, a_c, b_c, c', and c by requiring $10 \times T_{exp} + 5 \times T_{mul} + 1 \times T_h + 1 \times T_{inv}$. Finally, verify r and compute r' by requiring $4 \times T_{exp} + 4 \times T_{mul}$.

In the Payment Phase, U computes $H(sn)$ and encrypts it and pseudo e-coins requires $1 \times T_{PKC} + 1 \times T_h$ in Step 1. In Step 2 of the Payment Phase, M decrypts the received encryption item, verifies the pseudo e-coins, generates a signature, and encrypts a signature and ED by requiring $2 \times T_{PKC} + 1 \times T_{SKC} + 4 \times T_{exp} + 2 \times T_{mul} + 2 \times T_h$. In Step 3, U decrypts the received message and verifies it by requiring $1 \times T_{SKC} + 1 \times T_h$. Then, compute the confirmation parameters and encrypt them by requiring $1 \times T_{SKC} + 4 \times T_{mul} + 1 \times T_h$. In Step

4, M decrypted the received item, verifies them, and encrypts K by requiring $2 \times T_{SKC} + 5 \times T_{exp} + 2 \times T_{mul}$. Finally, U decrypted the received item, verifies K and derives the Goods by requiring $2 \times T_{SKC} + 1 \times T_h$.

Table 1: The computational complexities of our scheme

	Computational Complexity
Setup Phase	$3 \times T_{exp}$
Account Opening Phase	$2 \times T_{exp} + 1 \times T_{mul}$
Registration Phase	$4 \times T_{PKC} + 1 \times T_{SKC} + 1 \times T_h$
Withdrawn Phase	$16 \times T_{exp} + 11 \times T_{mul} + 1 \times T_h + 1 \times T_{inv}$
Payment Phase	$3 \times T_{PKC} + 7 \times T_{SKC} + 9 \times T_{exp} + 8 \times T_{mul} + 6 \times T_h$
Dispute Phase	$5 \times T_{PKC} + 1 \times T_{SKC} + 4 \times T_{exp} + 2 \times T_{mul}$

In the Dispute Phase, U encrypts a signature and some other information by requiring $1 \times T_{PKC}$ in Step 1. In Step 2, TTP decrypts the received message, and verifies the signature and e-coins by requiring $2 \times T_{PKC} + 4 \times T_{exp} + 2 \times T_{mul}$. Then, decrypt K and encrypt it requires $1 \times T_{PKC} + 1 \times T_{SKC}$. In Step 3, TTP encrypts some messages and sends it to M by requiring $1 \times T_{PKC}$.

5 Conclusions

This paper proposes a fair non-transfer network payment protocol with off-line TTP based on the Wang's protocol. In the proposed protocol, we have presented a general model in which two parties can fairly exchange the e-cash and soft goods. Our new protocol is also the first one that can provide the non-transfer property on fair payments.

References

[1] N. Asokan, Victor Shoup, and Michael Waidner, "Optimistic fair exchange of digital signatures," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 4, pp. 593–610, 2000.

[2] Feng Bao, "Collunding attacks to a payment protocol and two signature exchange schemes," in *Proc of Asiacrypt'2004, LNCS 3329, Springer-Verlag*, pp. 417–429, 2004.

[3] Feng Bao, Robert Deng, and Wenbo Mao, "Efficient and practical fair exchange protocols with off-line TTP," in *IEEE Symposium on Security and Privacy, Oakland, CA*, pp. 77–85, 1998.

[4] Colin Boyd and Ernest Foo, "Off-line fair payment protocols using convertible signatures," in *Advances in Cryptology –ASIACRYPT'98: International Conference on the Theory and Applications of Cryptology, Beijing, China*, vol. 1514, pp. 271–285. LNCS, Springer, 1998.

[5] S. Brands, "Untraceable off-line cash in wallets with observers," in *Advances in Cryptology - Crypto'93*, vol. 773, pp. 302–318. LNCS, Springer, 1993.

[6] H. Burk and A. Pfitzmann, "Value exchange systems enabling security and unobservability," *Computers & Security*, vol. 9, no. 9, pp. 715–721, 1990.

[7] Liqun Chen, "Efficient fair exchange with verifiable confirmation of signatures," in *Advances in Cryptology –ASIACRYPT'98: International Conference on the Theory and Applications of Cryptology, Beijing, China*, vol. 1514, pp. 286–299. LNCS, Springer, 1998.

[8] R. H. Deng, L. Gong, A. A. Lazar, and W. Wang, "Practical protocol for certified electronic mail," *Journal of Network and Systems Management*, vol. 4, no. 3, pp. 279–297, 1996.

[9] T. ElGamal, "A public-key cryptosystem and a signature scheme based on discrete logarithms," *IEEE Transactions on Information Theory*, vol. IT-31, pp. 469–472, July 1985.

[10] S. Even, O. Goldreich, and A. Lempel, “A randomized protocol for signing contracts,” *Comm. of the ACM*, vol. 28, no. 6, pp. 637–647, 1985.

[11] Chun-I Fan and Yu-Kuang Liang, “Anonymous fair transaction protocols based on electronic cash ,” *International Journal of Electronic Commerce*, vol. 13, no. 1, pp. 131–151, 2008.

[12] Chun-I Fan, Ming-Te Chen, and Wei-Zhe Sun , “Buyer-seller watermarking protocols with off-line trusted third parties ,” *International Journal of Ad Hoc and Ubiquitous Computing*, vol. 4, no. 1, pp. 36–43, 2009.

[13] Matthew K. Franklin and Michael K. Reiter, “Fair exchange with a semi-trusted third party (extended abstract),” in *ACM Conference on Computer and Communications Security, Zurich, Switzerland*, pp. 1–5, 1997.

[14] Cheng-Chi Lee, “Two attacks on the Wu-Hsu user identification scheme ,” *International Journal of Network Security*, vol. 1, no. 2, pp. 67–68, 2005.

[15] Min-Shiang Hwang, Cheng-Chi Lee, and Yan-Chi Lai, “Traceability on low-computation partially blind signatures for electronic cash,” *IEICE Transactions on Fundamentals on Electronics, Communications and Computer Sciences*, vol. E85-A, no. 5, pp. 1181–1182, 2002.

[16] Min-Shiang Hwang, Cheng-Chi Lee, and Eric Jui-Lin Lu, “Cryptanalysis of the batch verifying multiple DSA-type digital signatures,” *Pakistan Journal of Applied Sciences*, vol. 1, no. 3, pp. 287–288, 2001.

[17] Cheng-Chi Lee, Min-Shiang Hwang, and Li-Hua Li, “A new key authentication scheme based on discrete logarithms,” *Applied Mathematics and Computation*, vol. 139, no. 2, pp. 343–349, 2003.

[18] Li-Hua Li and Shiang-Feng Tzeng and Min-Shiang Hwang, “Generalization of proxy signature based on discrete logarithms,” *Computers & Security*, vol. 22, no. 3, pp. 245–255, 2003.

[19] R. C. Merkle, “One-way hash functions and DES,” in *Advances in Cryptology, CRYPTO’89*, pp. 428–446, Lecture Notes in Computer Science, Vol. 435, 1989.

[20] T. Okamoto and K. Ohta, “How to simultaneously exchange secrets by general assumption,” in *Proceedings of 2nd ACM conference on Computer and communications security*, pp. 184–192, 1994.

[21] R. L. Rivest, A. Shamir, and L. Adleman, “A method for obtaining digital signatures and public key cryptosystems,” *Communications of the ACM*, vol. 21, pp. 120–126, Feb. 1978.

[22] Shiang-Feng Tzeng, Cheng-Ying Yang, and Min-Shiang Hwang, “A new digital signature scheme based on factoring and discrete logarithms,” *International Journal of Computer Mathematics*, vol. 81, no. 1, pp. 9–14, 2004.

[23] Chih-Hung Wang, “Untraceable fair network payment protocols with off-line ttp,” in *Advances in Cryptology, ASIACRYPT’2003*, pp. 173 – 187, Lecture Notes in Computer Science, Vol. 2894, 2003.

[24] J. Zhou and D. Gollmann, “A fair non-repudiation protocol,” in *IEEE Symposium on Security and Privacy, Oakland, CA*, pp. 55–61, 1996.

[25] J. Zhou and D. Gollmann, “An efficient non-repudiation protocol,” in *Proceedings of the 1997 IEEE Computer Security Foundations Workshop*, pp. 126–132, 1997.