

Anonymity and Security in Continuous Double Auctions for Internet Retail Market

Changjie Wang and Ho-fung Leung
 Department of Computer Science & Engineering
 The Chinese University of Hong Kong
 Sha Tin, Hong Kong, P.R. China
 {cjwang, lhf}@cse.cuhk.edu.hk

Abstract

Electronic auctions have become an integral part of Electronic Commerce nowadays. Besides the popular single-sided auction protocols, i.e. English auction, Vickrey auction etc., Continuous Double Auction (CDA) is an important auction protocol that permits multiple buyers and sellers to trade goods. Internet CDAs have been widely used in financial and commodities markets. Although Internet provides an excellent infrastructure for CDAs, anonymity and security are important issues in the electronic CDA marketplaces. While most studies have been focusing on the strategies in agent-oriented CDAs [2][3][4][5], relatively little research has been done on the privacy and security issues. In this paper, we first discuss the privacy and security issues in electronic CDAs, and propose that the security requirements in electronic CDAs include anonymity, traceability, impossibility of impersonation, unforgeability, and verifiability. We then describe an anonymous and secure CDA protocol for electronic marketplaces. In the new protocol, both the anonymity of traders and the traceability of false offers are achieved. The identities of normal traders and their bidding behaviors are protected, while the identities of malicious traders can be revealed. With a simple analysis, we show that our CDA scheme satisfies all the required security properties.

1. Introduction

An integral part of Electronic Commerce [1], Internet auctions have become popular nowadays. Many commercial Internet auction sites have been very successful and they continue to expand.

Besides the popular single-sided auctions, such as English auction, Vickrey auction, etc., Continuous Double Auction (CDA) is also an important auction protocol that is widely used in financial and commodities markets around the world, including the Internet retail market. In

a CDA market [2], buyers and sellers are free to publicly announce at any time their bids/asks to buy and sell. The lowest ask submitted so far is called the *outstanding ask*, and the highest bid the *outstanding bid*. In many cases, only the outstanding bid and ask are maintained. A newly announced ask has to be lower than the outstanding ask, and a new bid higher than the outstanding bid. If the new ask is equal to or less than the outstanding bid, then a transaction occurs at the bid price, otherwise it becomes the current ask. Similarly, if the new bid is equal to or greater than the outstanding ask, then a transaction occurs at the ask price; otherwise it becomes the outstanding bid. After a transaction occurs, the outstanding bid and ask are removed, and a new round of CDA starts, in which the above procedure is repeated.

In the literature, most research regarding the CDA has been done on the investigation of strategies of bidders. A number of different CDA models have been constructed so far [2][17][18], and many bid-determination strategies have been proposed, such as [2][3][4][5][19][20]. However, relatively little research has been done on the privacy and security issues of agent-based CDAs. In this paper, we focus on the privacy and security issues in electronic CDA, and propose that the security requirements in electronic CDAs include *anonymity, traceability, impossibility of impersonation, unforgeability, and verifiability*. We then describe an anonymous and secure CDA protocol for electronic marketplaces, which is strategically equivalent to the traditional CDA protocol. However, in the new protocol both the anonymity of traders and the traceability of false offers are achieved. The identities of normal traders and their bidding behaviors are protected, while the identities of malicious traders can be revealed. With a simple analysis, we show that our CDA scheme satisfies the all the required security properties.

This paper is organized as follows. In section 2, we discuss the privacy and security issues in CDA markets. In section 3, we propose an anonymous and secure CDA protocol. We first present a brief introduction to the public key cryptosystem and its application of blind RSA signature, followed by a description of the anonymous

and secure CDA protocol. The properties of the protocol are discussed and analyzed in section 4. Section 5 concludes the paper.

2. Security issues in continuous double auction markets

A number of secure electronic auction schemes have been proposed [6][7][8][9], and most of them focus on the single-sided auctions, such as secure Vickrey auctions. The security of CDAs, however, has not attracted much attention, despite of the fact that it is a very important issue.

Anonymity of customers, that is, the protection of the privacy of identities of customers during electronic transactions, is a general security consideration in many Electronic Commerce applications over the Internet. Very often, people want to be anonymous when making transactions over the Internet, just like what they can do in the real world. Consider the following simple example. If Bob wants to buy a piece of chocolate, he can then just pay cash and take the piece of chocolate without having to provide his name, address and other information to the shopkeeper or anyone else. In an Internet CDA market (say, an Internet retails site), the anonymity of users is also a reasonable expected feature: people should be able to submit offers and conduct a transaction in an anonymous manner. However, in an Internet retails market, auctions are usually common value or correlated value auctions, in which the bidders' values of the auctioned item entirely or partially depend upon other bidders' preferences. If complete anonymity is provided, there could be some potential malicious behaviors. For instance, someone can play a prank and submit an unrealistic offer, and then just run away without actually completing the transaction. In a more complicated scenario, a dishonest trader may mislead the market and give wrong information to other traders by submitting false bids, so as to get an extra profit, as exemplified in the following. Suppose there is a seller Alice who wants to sell a video game at \$10 in a CDA market. As we know, one important feature of CDA is that traders can revise their bids/asks in response to the latest market situation. Therefore, Alice may request her friend, Bob, to submit false bids to mislead other traders. That is, Bob can always submit a bid (say, \$20) higher than the current outstanding ask (say, \$10) to generate a transaction record at \$20.¹ Such records then create a fake indication to other traders that there are many demand requests of the video game at high prices in the market, though Bob will never conduct those

¹ In the Internet retails market, the transaction is usually not done in real-time manner. Therefore, Bob's false offer may generate a trade record that affects the market even it does not lead to the final transaction.

transactions by buying the video game eventually, for sure. As a result, when Alice submits her ask at, say, \$15, it is very possible that a buyer will accept her ask. In this way, Alice obtains an extra profit of \$5. Due to the complete anonymity of traders, we have no way to identify Bob even if we notice that there are some false offers in the market, which never result in the successful eventual transactions. From this example, we see that there can be cheating behavior if complete anonymity of traders is provided in an Internet CDA market. Therefore, an anonymity revocation method under certain conditions is necessary.

We summarize some preferred security properties that a secure CDA market should satisfy.

(a) **Anonymity.** During the CDA, the identities of traders are not revealed to anyone. In other word, nobody can associate a trader with the ask/bid he submits. (An exception is that the authority can identify the winners under certain conditions, see *Traceability* below.)

(b) **Traceability (Non-Repudiation).** The authority (or authorities) of the CDA market can identify winners (*i.e.* a trader who makes a trade successfully) under certain conditions, and only the authority (or authorities) can do so. Therefore a winner cannot deny that he has submitted an ask/bid.

(c) **Impossibility of impersonation.** No one can impersonate any other traders.

(d) **Unforgeability.** No one can forge a valid ask/bid.

(e) **Verifiability.** Everybody can verify the validity of an ask/bid, and can confirm whether an ask/bid is submitted from a valid trader or not (although it is impossible to identify *which* trader submits the ask/bid).

3. An anonymous and secure scheme for continuous double auction

In this section, we propose an anonymous and secure CDA scheme that satisfies all the security properties presented in Section 2. The scheme is practical and therefore suitable for the Internet retails market.

3.1 Preliminaries

We first give a brief introduction to the cryptography technologies that we employ so that the paper is self-contained. These include *Digital Signature* for non-repudiation of the data, *Blind Signature* for anonymity and *Cut-and-Choose* technique for correctness.

(a) Public Key Cryptosystem and Digital Signature

The concept of public-key cryptography was due to Diffie and Hellman [10] in 1976. The main idea is that every one can have a pair of keys: a *public key*, which is open; and a *private key*, which is kept secret. To send a secret message m to, say, Alice, one should encrypt m

using the Alice's public key, so that only Alice can get the m by decrypting the encrypted message using her private key. No one else can decrypt without knowing the private key of Alice. A public key encryption system can also be used for digital signature [10]. To sign a message m , Alice first generates a signature S on the m as $S = \text{Sig}(m, s)$, where $\text{Sig}()$ is a signature function, and s is Alice's private key. Anybody can then verify the validity of the signature (S, m) by computing a verification function $v = V(m, S, p)$: if $v = \text{TRUE}$ then (S, m) is authenticated by Alice. In such a way, the non-reputation of the message m is achieved since no one else can possibly generate S without knowing the Alice's private key.

Named after Rivest, Shamir and Adleman, RSA [10] is the first public key cryptosystem, the security of which is based on the difficulty of factoring large numbers. The RSA algorithm can be described as follows.

Key generation. To generate the public/private key pairs, Alice chooses two large prime numbers p and q , computes their product $n = p \cdot q$, and randomly chooses a number e smaller than n , such that e and $\Phi(n)$ are relatively prime, where $\Phi(n) = (p-1) \cdot (q-1)$. After choosing the public key, Alice computes the private key d such that $e \cdot d = 1 \pmod{\Phi(n)}$. Then, (n, e) are announced as the public key of Alice and d is kept secret as the private key of Alice.

Encryption. To send an encrypted message m to Alice, Bob first divides m into a number of blocks (m_1, m_2, \dots) , such that each block has a unique representation modulo n (for binary data, choose the largest power of 2 less than n). The encrypted message c consists of similarly sized message blocks (c_1, c_2, \dots) of about the same length, such that $c_i = m_i^e \pmod{n}$.

Decryption. To decrypt the message c received, Alice takes each encrypted block c_i and recover the block m_i using $m_i = c_i^d \pmod{n}$.²

RSA can also be used for digital signature. By encrypting the message m using her private key, Alice generates a secure digital signature. Following is a simple basic protocol of digital signature:

(1) Alice generates a secure digital signature S on the message m , that is, $S = m^d \pmod{n}$. In other words, S is the encryption of m using the private key of Alice.

(2) Alice sends the message with her signature (S, m) to Bob.

(3) Bob uses Alice's public key to decrypt the signature and check whether m can be recovered. If $m = S^d \pmod{n}$, then it is evidenced that the message m has indeed been signed by Alice and has never been altered by others since then.

(b) Blind RSA Signature

In the general digital signature scheme, one always knows the contents of a message before signing on it. However, in some particular applications, we prefer to have someone sign on a message without letting him know the contents of the message. This scheme of blind signature [13] has been widely used in electronic cash systems, electronic election, etc. In this paper, we make use of Chaum's blind RSA signature [14], which can be illustrated by the following example.

Suppose Bob wants Alice to sign on a message m without knowing what m is, the basic protocol is as follows.

(1) Bob first randomly selects a number k , which is called a *blind factor*, and makes a blinding transformation on m to get message, a blinding $m' = m \cdot k^e \pmod{n}$, where (n, e) is the public key of Alice. Then Bob sends m' to Alice. Note that Alice cannot recover m from m' without knowing k .

(2) Alice generates a signature S' on m' using her private key, i.e. $S' = m'^d = (m \cdot k^e)^d = m^d \cdot k \pmod{n}$, and returns S' to Bob.

(3) Bob computes $S = S' / k = (m \cdot k^e)^d / k = m^d \pmod{n}$ and gets a signature of Alice on m . Note that Alice knows nothing about m , since she does not know k .

Usually, the Cut-and-Choose technique [15] is always applied together with blind signature to ensure that the signer can check the correctness of the message without knowing the actual message. The protocol of blind RSA signature employing the Cut-and-Choose technique is described below:

(1) Bob first randomly selects L numbers $k_i, i = \{1, 2, \dots, L\}$, as *blind factors*, and generates L blinding messages: $m'_i = m_i \cdot k_i^e \pmod{n}$, $i = \{1, 2, \dots, L\}$, where (n, e) is the public key of Alice. Then Bob sends $m'_i, i = \{1, 2, \dots, L\}$ to Alice.

(2) Alice randomly selects a t , such that $1 \leq t \leq L$, and requests that Bob submit the other $L-1$ k_i values, $i = \{1, 2, \dots, t-1, t+1, \dots, L\}$. Bob then submits the $L-1$ k_i values requested by Alice.

(3) On receiving the $L-1$ k_i values, $i = \{1, 2, \dots, t-1, t+1, \dots, L\}$, Alice opens the $m'_i, i = \{1, 2, \dots, t-1, t+1, \dots, L\}$ to get the m_i for correctness checking. If the $L-1$ m_i are correctly generated, Alice believes that the m_t is also correctly generated, and then

² Note that

$c_i^d = (m_i^e)^d = m_i^{k \cdot \Phi(n) + 1} = m_i^{k \cdot \Phi(n)} \cdot m_i = 1^k \cdot m_i = m_i \pmod{n}$, and $m_i^{k \cdot \Phi(n)} = 1 \pmod{n}$ due to the Fermat's Theorem [10].

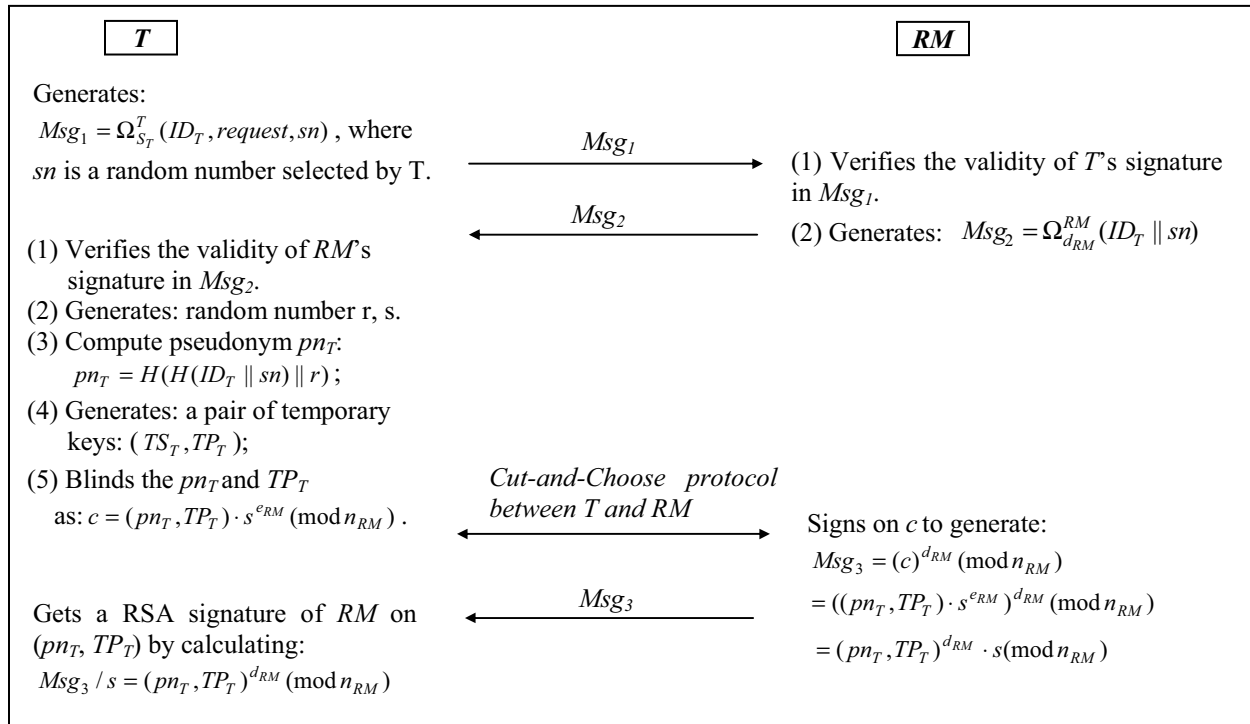


Fig. 2 Flow chart of protocol 1-registration between trader T and RM

the notation $(m1 \parallel m2)$ denotes the concatenation of two strings $m1$ and $m2$.

Protocol 1: Registration between trader T and RM (here, RM uses RSA system).

Step 1: The trader T first sends a registration request message $Msg_1 = \Omega_{S_T}^T (ID_T, request, sn)$ to RM , where ID_T is the identity information of T (say, ID card number), and sn is a random number selected by T .

Step 2: On receiving the request from T , RM verifies the validity of the signature of T in Msg_1 to make sure that the request is generated from T . Then RM signs on the $H(ID_T \parallel sn)$ to generate a message $Msg_2 = \Omega_{d_{RM}}^{RM} (ID_T \parallel sn)$ and sends Msg_2 back to T .

Step 3: After verifying the validity of RM 's signature on $H(ID_T \parallel sn)$, T generates L (L should be selected large enough) pairs of temporary keys (TS_T, TP_T) , and constructs L blinded data c , $c = (pn_T, TP_T) \cdot s^{e_{RM}} \pmod{n_{RM}}$, where $pn_T = H(H(ID_T \parallel sn) \parallel r)$, r and s are randomly selected numbers. T then sends L blinded data c to RM .

Step 4: RM randomly selects $L-1$ data c and request T to submit the corresponding $L-1$ TP_T , r and s . RM then unblinds the $L-1$ data c for verification of correctness. If the $L-1$ opened c are all formed correctly, RM signs on remain data c , as:

$$\begin{aligned} Msg_3 &= (c)^{d_{RM}} \pmod{n_{RM}} \\ &= ((pn_T, TP_T) \cdot s^{e_{RM}})^{d_{RM}} \pmod{n_{RM}} \\ &= (pn_T, TP_T)^{d_{RM}} \cdot s \pmod{n_{RM}} \end{aligned}$$

then returns Msg_3 to T .

Step 5: On receiving Msg_3 , T computes $Msg_3 / s = (pn_T, TP_T)^{d_{RM}} \pmod{n_{RM}}$ and thus gets the signature of RM on his pseudonym and temporary public key: (pn_T, TP_T) . Protocol ends.

Fig. 2 shows the flow chart of the registration protocol between T and RM . After registration, T gets a suit of data $\{Msg_2, r, TS_T, (pn_T, TP_T), (pn_T, TP_T)^{d_{RM}} \pmod{n_{RM}}\}$, where pn_T is a pseudonym of T and $\{TS_T, TP_T\}$ are the temporary key pairs of T . Note that both pn_T and TP_T are blindly authenticated by RM . That is, RM knowing nothing about p_T and TP_T , so that he cannot link them with the true identity of T . Furthermore, RM records the following data related to T : $\{ID_T, Msg_1, Msg_2\}$ to database after registration.

After registration with RM , the trader T should register with MM to generate an auction certificate.

Protocol 2: Registration between trader T and MM .

Step 1: The trader T only sends message $Msg_4: \{Msg_2, r, (pn_T, TP_T), (pn_T, TP_T)^{d_{RM}} \pmod{n_{RM}}\}$ to MM for registration, without showing his/her ID_T to MM .

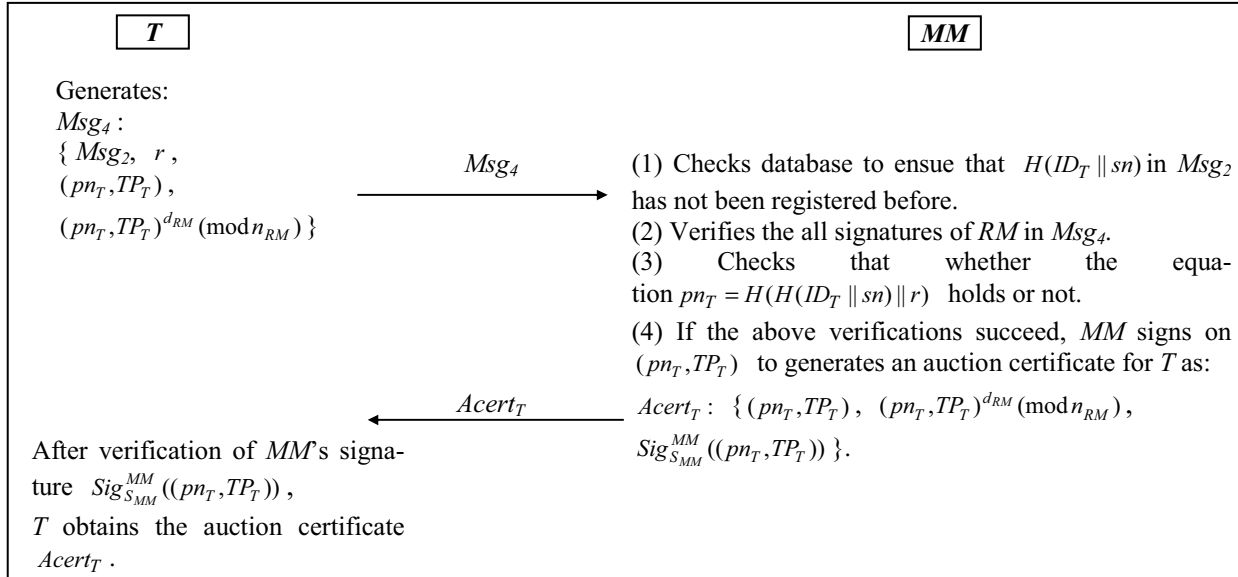


Fig. 3 Flow chart of protocol 2-registration between trader T and MM

Step 2: On receiving Msg_4 , MM first checks his database to ensure that $H(ID_T || sn)$ in Msg_2 has not been registered by others before, and then verifies the all signatures of RM in Msg_4 . MM also checks that $pn_T = H(H(ID_T || sn) || r)$ hold. If the above verifications succeed, it is convinced that pn_T of trader T is correctly generated, and has been authenticated by RM . MM then signs on the (pn_T, TP_T) to generate an auction certificate $Acert_T$ for T as

$Acert_T : \{ (pn_T, TP_T), (pn_T, TP_T)^{d_{RM}} \pmod{n_{RM}},$
 $Sig_{S_{MM}}^{MM}((pn_T, TP_T)) \}$. MM sends $Acert_T$ to T .

Step 3: After verifying the validity of MM 's signature $Sig_{S_{MM}}^{MM}((pn_T, TP_T))$, T obtains the auction certificate $Acert_T$. Protocol ends.

Fig. 3 shows the flow chart of the registration protocol between T and MM . After protocol 2, trader T possesses a pair of temporary keys (TS_T, TP_T) , in which TP_T is bound with his auction certificate $Acert_T$, authenticated by both RM and MM . Note that MM does not know the identity of T during protocol 2 since he receive nothing related ID_T . MM just verifies that the trader T has been registered with RM correctly, and then issue $Acert_T$ by signing on pseudonym (pn_T, TP_T) . MM also records Msg_4 in his database as non-repudiation evidence.

(b) Phase 2: Trading at the CDA Market

After registrations, trader T is eligible to enter a CDA market and submit offers for buy or sell. Our description of CDA market using messages is similar to [17]. A valid offer message m of trader T is the ordered tuple $m_T = \langle offer, Sig_{TS_T}^T(offer), Acert_T \rangle$, where

$offer = \{ pn_T, BUY/SELL, Commodity, Value, TimeStamp \}$

means that a trader, named pn_T wants to buy/sell a commodity with price $value$. Note that $TimeStamp$ here is the time stamp (sometime it can be substituted just by a random number) used to against the replay attack.

The flow chart of verification of an offer is shown in Fig. 4. Anybody in the CDA market can check the validity of an offer according to the following rules:

1. Verifies the validity of the signatures of both RM and MM in auction certificate $Acert_T$, respectively with the corresponding public key of RM and MM , which are publicly available.

2. Verifies the validity of the signature $Sig_{TS_T}^T(offer)$ using the corresponding public key TP_T , which is bound with the $Acert_T$.

If all above verification succeeds, the offer message is valid. Note that in above verification procedures, the identity of trader T is always kept secret. A trader's offer is bound with his pseudonym and authenticated with his temporary private key. The offer can only be verified using the temporary public key in his auction certificate, which only proves that the offer message is submitted by a registered trader, and no more information relate to the trader's identity is provided. No one, even RM or MM separately, can reveal the identity of trader from an offer message.

On receiving a valid offer message, a so-called market monitor [17] works according to the general CDA rules, such as updates the outstanding ask/bid, notifies the traders involved in the transaction, publishes all transaction history, and also records every valid offer message as non-repudiation evidence when there are disputes.

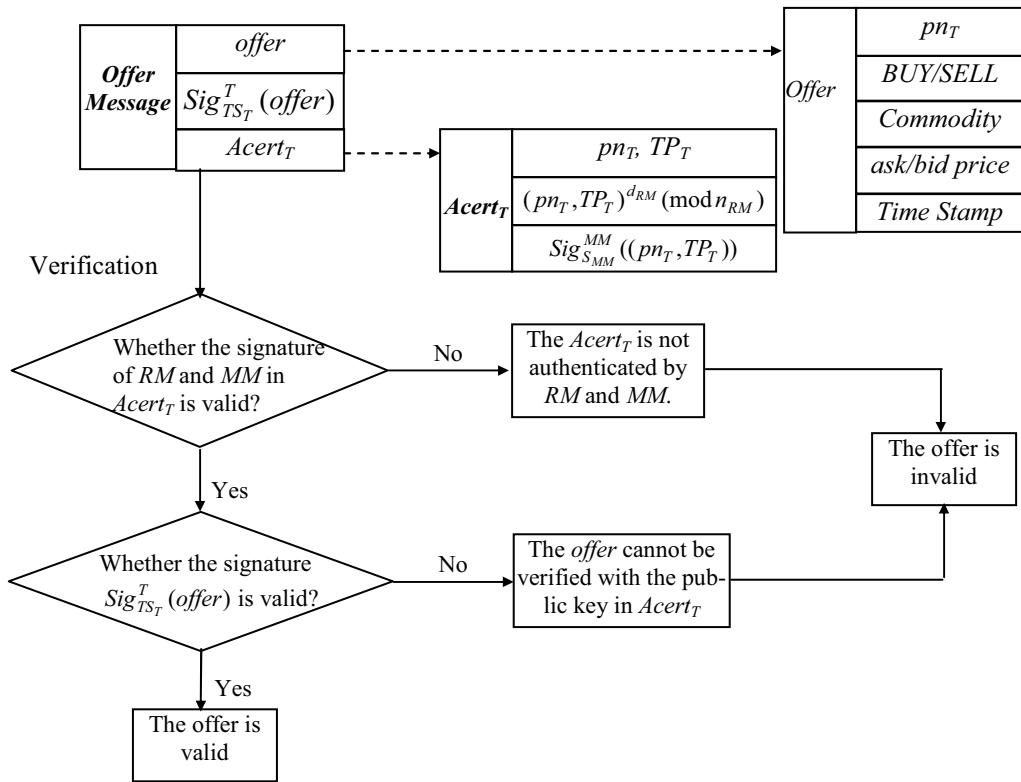


Fig. 4 Flow chart of offer verification

(c) Phase 3: Trader Tracing

When two valid offer messages result in a transaction, each involved trader should prove that he is eligible for the transaction by showing that he has the knowledge of temporary private key used to authenticate the winning offer. That is, the real eligible trader can sign on any message selected by market monitor to prove that he really know the corresponding temporary private key). Then, two traders make the trade anonymously.

In another case, as we mention in section 2, the anonymity of certain trader should be revoked in some cases, say, a trader runs away while his offer leads to a trade. To reveal the identity of the suspected trader is simple in our scheme. Suppose *A* is a suspected trader, who submitted an offer message $m_A = \langle offer, Sig_{TS_A}^A(offer), Acert_A \rangle$. To perform the tracing of *A*, the market monitor sends the m_A and the evidence of suspected behavior of *A* (say, not finish the expected trade) to *MM*. On receiving the request of tracing from Monitor, *MM* first verify the signatures in m_A , i.e. $Sig_{TS_A}^A(offer)$ and $Sig_{S_{MM}}^{MM}(pn_A, TP_A)$ in $Acert_A$. If verification succeeds, it is convinced that *A* is a registered trader. *MM* then checks his database to find the Msg_2 corresponding to the pn_A of *A*. Note that *MM* should record these information in protocol 2. Next, *MM* forwards all information related, including m_A , Msg_2 and tracing re-

quest of monitor to *RM*. What *RM* needs to do is only to check his database to find the true identity ID_A of *A* matched to the Msg_2 . Note that there is only one exact Msg_2 in *RM*'s database, which is bound to ID_A . Then, both *RM* and *MM* publish their records so that every one can verify that the trader *A* is just the one who submitted the suspected offer. In such way, the identity of a malicious trader can be revealed with the cooperation of *MM* and *RM*, while the normal traders can always act anonymously in the CDA market.

4. Analysis

In this section, we discuss the security of our scheme, and show that our scheme satisfies all security properties presented in Section 2. We do not make any special trust assumptions for the *RM* and the *MM* in our CDA scheme, that is, both authorities are not fully trusted, and it is possible that they misbehave. However, no one (including traders, *RM* and *MM*) can misbehave without being disclosed. In the following, we evaluate the security of our scheme with reference to the requirements presented in Section 2.

4.1 Anonymity

In the CDA market, a submitted offer by a trader T is in the form of $m_T = \langle offer, Sig_{TS_T}^T(offer), Acert_T \rangle$. Any trader can verify the validity of the *offer* using the public key embedded in $Acert_T$, while knowing nothing about the identity of the trader T since that there is no identity information in m_T . MM knows nothing about T 's identity, either, because he only verifies the *correctness* of the message submitted by T in protocol 2 while having no idea of who is submitting the message. On the other hand, as described in the protocol 1, RM checks the identity of a trader who submits a pseudonym and a temporary public key for registration. However, blind RSA signature is used in protocol 1 and both the pseudonym and the temporary public key are only blindly authenticated by RM . Consequently, RM cannot trace from an offer, authenticated using $Acert_T$ (includes the pseudonym and the temporary public key of T), back to the trader's real identity.

Therefore, while the validity of an offer can always be verified by anyone who wants to do so, no one (not even any one of the authorities MM and RM alone) can associate the offer with the trader who submitted this offer.

4.2 Traceability

In the proposed CDA scheme, a trader cannot repudiate his/her offer in any case. As described in above section, when the authorities RM and MM join hand, it is always feasible to trace from any offer back to the real identity of the trader who makes this offer. Therefore, no trader can repudiate offers he submitted.

4.3 Impossibility of impersonation

It is often a rational act that one submits fake bids in other bidders' names if this is possible. For example, a bidder may wish to incriminate Bob in case of offer withdrawal, or for high bogus offers. However, impersonation is technically impossible in our scheme, as shown in the following theorem.

Theorem 1 *Nobody, not even RM and MM , can impersonate a valid trader to submit a valid offer.*

Lemma 1 *One cannot impersonate another trader unless he can break the public key cryptosystem.*

For Alice to impersonate Bob, say, she has to do one of two things. One is that Alice gets the temporary private key of Bob and signs on a fake offer in Bob's pseudonym. The other is that she gets the long-time private key of Bob so that she can register as Bob with RM and MM to get an

auction certificate. It is clear that whichever methods Alice uses, she has to break the public key cryptosystem, which is generally believed to be impossible.

Lemma 2 *Neither RM nor MM can impersonate a valid trader.*

Suppose that RM can impersonate a valid trader T in the following way:

First, RM generates a fake \overline{Msg}_4 of T independently as follows.

$\overline{Msg}_4 : \{ \overline{Msg}_2, \bar{r}, (\overline{pn}_T, \overline{TP}_T), (\overline{pn}_T, \overline{TP}_T)^{d_{RM}} \pmod{n_{RM}} \}$, where \overline{Msg}_2 is $\{ ID_T \parallel \bar{sn}, Sig_{d_{RM}}^{RM}(H(ID_T \parallel \bar{sn})) \}$, and

$\overline{pn}_T = H(H(ID_T \parallel \bar{sn}) \parallel \bar{r})$. Note that \bar{r} , \overline{TP}_T and \bar{sn} are generated by RM , instead of that of T . Then RM sends

\overline{Msg}_4 to MM for registration as trader T . On receiving this fake \overline{Msg}_4 , MM can not be conscious of the fraud of RM , and will then sign on the $(\overline{pn}_T, \overline{TP}_T)$. In such way, RM

get a fake auction certificate of trader T as: $\overline{Acert}_T :$

$\{ (\overline{pn}_T, \overline{TP}_T), (\overline{pn}_T, \overline{TP}_T)^{d_{RM}} \pmod{n_{RM}} \}, Sig_{S_{MM}}^{MM}((\overline{pn}_T, \overline{TP}_T))$

and a corresponding private key \overline{TS}_T . Although such fraud cannot be found during registration, RM will be

proved a cheater when the fake auction certificate is used with a false offer. That is, at the end of tracing, both RM

and MM have to publish all related records they have for public verification. However, if \overline{Acert}_T is faked by

RM , RM cannot provide a $\overline{Msg}_1 : \{ ID_T, Request, \bar{sn}, Sig_{S_T}^T(H(ID_T, Request, \bar{sn})) \}$, as a evidence to prove

that the \overline{Acert}_T is generated on request of T , since he can

not generate the signature $Sig_{S_T}^T(H(ID_T, Request, \bar{sn}))$ unless he knows the private key S_T of T . As a result, it is

convinced that RM fakes the auction certificate \overline{Acert}_T .

Note that, RM may use the real sn of T and \bar{r} , \overline{TP}_T of

himself to construct a \overline{Msg}_4 and then register with MM .

however such misbehavior will be detected by MM , since MM can find that $H(ID_T \parallel sn)$ is used twice for registra-

tion with different $(\overline{pn}_T, \overline{TP}_T)$ in auction certificate.

Even RM and MM collude to generate the fake \overline{Acert}_T

of T , this cheating behavior also can be detected eventually. Note that there must be a true $Acert_T$ possessed

by T , and T can publish his data related to $Acert_T$ when necessary to prove his innocence. There is no other explains why two different $Acert_T$ are bound with the

same number sn except that the RM and MM has misbe-

havior.

haved to cheat. Therefore, both *RM* and *MM* cannot impersonate a valid trader without being detected.

Lemma 3 *No trader can frame RM and MM by registering two different Acert_T bound with the same sn.*

As mentioned in Lemma 2, if there are two different *Acert_T* bound with the same *sn*, we believe that *RM* and *MM* has misbehaved. Therefore, a malicious trader may try to frame *RM* and *MM* in this way. Suppose two colluding traders Alice and Bob who are malicious. Alice can give her $H(ID_A || sn_A)$ to Bob, and Bob generates a false pseudonym of his $\overline{pn_B} = H(H(ID_A || sn_A) || r_B)$ (instead of $pn_B = H(H(ID_B || sn_B) || r_B)$) and then submits the blinded $\overline{pn_B}$ and a temporary public key $\overline{TP_B}$ to *RM* to be authenticated. Note that we use *Cut-and-Choose* technique in protocol 1, so that *RM* can check the correctness of pn_B (i.e. whether the pn_B is formed as: $pn_B = H(H(ID_B || sn_B) || r_B)$), even if he does not know the final pn_B he signs on. It is clear that the chance that Alice and Bob frame the *RM* and *MM* is only $1/L$, where L is the security parameter in the *cut-and-choose* techniques. If L is large enough, the probability of successfully framing is very small. In practice, we can select a not very large L (say, $L > 20$) for higher efficiency, and a penalization is applied in case of *RM*'s finding a cheating attempt from traders.

4.4 Unforgeability

In the secure CDA scheme, only registered traders can submit valid offers, as each offer is signed by the temporary private key *TS* of trader and the *TS* is bound to an authenticated auction certificate. No one else can ever forge a valid offer without registration unless he can break the public key cryptosystem to get the private keys of *RM* and *MM*. Furthermore, each valid offer includes a *timestamp*, which is used as a freshness indicator. Therefore, any offer with an outdated *timestamp* will be rejected as an invalid ask/bid.

4.5 Verifiability

In the new scheme, any one can verify a submitted offer is valid or not by checking the signature on the offer. In addition, anyone can check whether the offer is submitted by a valid trader by checking the validity of the auction certificate in the offer message. In tracing a malicious trader, both *RM* and *MM* publish the all records at the end of tracing, so that every one can verify the result of tracing and is convinced that there is no trick.

5. Conclusion

To the best of our knowledge, this paper is the first one discussing the privacy and security issues in the CDA markets. We present the security requirements in a CDA market, and propose an anonymous and secure CDA scheme, which satisfies all the requirements.

The main idea of proposed scheme is that each trader generates a pair of temporary keys and a pseudonym to construct an auction certificate, which is blindly authenticated by *RM*. The auction certificate is then checked and authenticated by *MM*. During the CDA, traders always submit offers authenticated using their temporary key pairs and the auction certificate. In such way, no one, not even *RM* or *MM*, can link an offer back to the real identity of the trader who submits the offer. In case of fraud, both *RM* and *MM* can work together to perform the tracing of a malicious trader. As the discussion in Section 4 shows, we do not make any special trust assumption in our scheme, which means that neither *RM* nor *MM* is fully trusted, and they cannot misbehave without being detected.

Our scheme, while quite efficient, uses *Cut-and-Choose* techniques to ensure correctness of the auction certificate. This requires a certain number of interactions, which may be costly. However, note that the *Cut-and-Choose* technique is only used during the registrations of the traders with *RM* in the first phase, and does not effect the efficient of CDA market. Furthermore, both *RM* and *MM* are working offline, which means that they are not involved in the CDA until there is a dispute occurred. This also makes the proposed CDA scheme efficient for the Internet retails market.

6. Acknowledgement

The work described in this paper was supported by a grant from the Research Grants Council of the Hong Kong Special Administrative Region, China (Project No.CUHK4346/02E).

7. References

1. He, M., Jennings, N. and Leung, H. F. "On Agent-Mediated Electronic Commerce". *IEEE Trans. on Data and Knowledge Engineering*, 15(4), pp.985-1003, 2003..
2. Friedman, D. and Rust, J. "The Double auction Market: Institutions, Theories and Evidence". Addison-Wesley 1992.
3. He, M. and Leung, H. F. "An Agent Bidding Strategy Based on Fuzzy Logic in a Continuous Double Auction", *Proceedings of the 2001 IEEE International Conference on Systems, Man and Cybernetics. e-Systems and e-Man for Cybernetics in Cyberspace*, pp. 583-8, May 2001, Tucson, AZ, USA.
4. Cliff, D. and Bruten, J. "Less than Human: Simple adaptive trading agents for CDA markets". *Proceedings of the 1998 Sym-*

posium on Computation in Economics, Finance and Engineering: Economic Systems.

5. Preist, C. and van Tol, M. "Adaptive Agents in a Persistent Shout Double Auction". *Proceedings of the 1st International Conference on the Internet, Computing and Economics*, pp.11-18, ACM press, 1998.
6. Boyd C. and Mao W., "Security Issues fro Electronic Auctions", Hewlett Packard, HP Technical Report HPL-2000-90, 2000.
7. Naor M., Pinkas B. and Summer R., "Privacy Preserving Auctions and Mechanism Design", *Proceedings of ACM conference on E-commerce*, pp.129-139, 1999.
8. Baudron O. and Stern J., "Non-interactive Private Auctions", *Proceedings of Financial Cryptography'01*, LNCS 2339, pp.364-377, 2001.
9. Lipmaa H., Asokan N. and Niemi V., "Secure Vickrey Auctions without Threshold Trust", *Proceedings of Financial Cryptography'02*, LNCS 2357, 2001.
10. Diffie W. and Hellman M.E., "New direction in cryptography", *IEEE Trans. On Information Theory*, vol 22(6), pp.644-654, 1976.
11. Rivest R., Shamir A. and Adleman L., "A method for obtaining digital structures and public-key cryptosystem", *Communication of ACM*, 21(2), Feb 1978.
12. Rabin, M. O., "Digital signatures and Public-key Functions as Factorization", MIT Laboratory for Computer Science, Technical Report, MIT/LCS/TR – 212, Jan, 1979.
13. Merkle, R.C., "Secrecy, Authentication, and Public Key System", MUI Research Press, Michigan, 1979.
14. Chaum,D., "Blind signatures for untraceable payments", *Advances in Cryptology-CRYPTO'82 Proceedings*, pp199-203, Plenum Press, 1983.
15. Chaum, D. "Security without identification: transaction system to make big brother obsolete", *Communication of ACM*, Vol. 28, No. 10, pp.1030-1044, 1985.
16. Chaum, D., Fiat, A., Naor, M., "Untraceable Electronic Cash", *Advances in Cryptology-CRYPTO'88, LNCS*, Vol. 1440. pp.319-327, Springer-Verlag, 1988.
17. Gjerstad S. and Dickhaut J., "Price Formation in Double Auction", *Games and Economic Behavior*, vol.22, pp1-29, 1998.
18. Sadrieh A., *The Alternating Double Auction Market: A Game Theoretic and Experimental Investigation*. Springer, 1998.
19. Park S., Durfee E., and Birmingham W., "An Adaptive Agent Bidding Strategy Based on Stochastic Modeling", *Proc. Third Int'l Conf. Autonomous Agents*, pp. 147-153, 1999.
20. Tesauro C. and Bredin J., "Strategic Sequential Bidding in Auctions Using Dynamic Programming", *Proc. First Int'l Joint Conf. Autonomous Agents and Multi-Agent systems*, pp 591-598, July 2002.