Provable Secure Brand-new Multi-auction Mechanism with Dynamic Identity

Jung-San Lee*, Kuo-Jui Wei, Ying-Chin Chen, and Yun-Hsiang Sun
Department of Information Engineering and Computer Science,
Feng Chia University,
Taichung 407, Taiwan, ROC
leejs@fcu.edu.tw

Received July 2, 2016; revised September 25, 2016; accepted October 12, 2016; published December 31, 2016

Abstract

Different from traditional auctions, electronic auctions provide a platform to allow bidders and auctioneers merchandise to each other over network anytime and anywhere. Auctioneers can publish information of goods, and bidders can choose the interested targets through this bidding platform. To ensure the fairness and security of electronic auctions, Li et al. have proposed a practical electronic auction scheme which can confirm the requirement of strong anonymity, bidding privacy, and secret bidding price. However, we have found out that Li et al.'s scheme may lurk the risk of the denial-of-service attack during the bidding phase in a sealed-bid auction. Thus, we propose a brand-new sealed-bid auction mechanism, in which the essentials of e-auction can be firmly preserved. In particular, each bidder only needs to register at the center once and then can join to multiple plays launched by different auctioneers. Moreover, the correctness of mutual authentication is confirmed according to the BAN logic model.

Keywords: Electronic auction, sealed-bid auction, dynamic identity, BAN logic, anonymity

http://dx.doi.org/10.3837/tiis.2016.12.025 ISSN : 1976-7277

1. Introduction

With the rapid development of the Internet technology and electronic commerce, commonly known as E-commerce, electronic auction which is one of E-commerce business model becomes more and more popular in recent years. Originally, traditional auctions occur at the market when someone wants to sell the goods or services and invite lots of buyers to bid. Different from traditional auctions, electronic auctions allow bidders and sellers communicate each other over the network anytime and anywhere. For example, eBay, Yahoo! and Taobao are quite famous electronic auction websites.

Generally, electronic auctions can be divided into three types [1-18]: English auction, Dutch auction, and sealed-bid auction. As to the English auction, each bidder arbitrarily chooses a bidding price and then submits a public bid which must be higher than the previous round. The bid will be finished whenever no one bids the higher price. Dutch auction is similar to English auction, however, it starts at the top price, and the price will progressively go down until someone is willing to buy the goods with the first price. Both English auction and Dutch action are collectively known as an open auction. On the other hand, a sealed-bid auction allows each bidder submit a bid to the auctioneer with a concealed price. After receiving tender information from bidders, an auctioneer will reveal the contents of all the bids and further compare these bids to pick out the highest one. Later, the auctioneer publishes the highest bidding price as well as announces the winner at the end of the auction.

Nowadays, many protocols for electronic auction have been proposed. In 2003, Chang and Chang [2] presented an efficient anonymous auction protocol based on deniable authentication which can achieve the bidder anonymity. Later on, Jiang et al. [3] proposed an improvement on efficient anonymous auction which claimed that Chang et al.'s protocol would suffer from the replay attack in the initial phase. However, the computational cost was not taken into consideration in their scheme. Thus, Chang and Chang [4] presented an enhanced anonymous auction protocol with the alias. Afterwards, Liaw et al. [5] proposed an electronic online bidding auction protocol to meet certain security and make auction system more efficient. In 2008, Wu et al. [11] found that Liaw et al.'s protocol could not withstand the forge attacks. Hence, they provided a new electronic auction scheme to overcome these weaknesses. Lately, to reduce the heavy computational load of the entire auction in previous schemes [2, 3, 5], Chung et al. [6] proposed an English auction scheme with the bulletin board method which can raise the efficiency and verify bidding information. Owing to the increasing demand for mobile devices in electronic auctions recently, Chung et al. [7] proposed a mobile auction agent model (MoAAM) using elliptic curve cryptosystem in 2011. According to their scheme, it can permit bidders take part in electronic auctions via a mobile agent. Moreover, this scheme can make the auctions more efficient than others in terms of mitigation of computational cost on mobile devices. Later on, Li et al. [12] proposed a practical electronic auction scheme to achieve strong anonymity, bidding privacy, and secret bidding prices for sealed-bid auction which [2, 3, 4] could not confirm. Nevertheless, we find out that Li et al.'s scheme may suffer from the denial-of-service attack during the bidding phase in a sealed-bid auction.

Thus, we propose a brand-new sealed-bid auction mechanism based on the concept of multi-server verification [22], in which the essentials of e-auction can be firmly preserved. In particular, each bidder only needs to register at the center once and then can join to multiple plays launched by different auctioneers. The followings are the requirements the proposed new method can firmly confirmed [19-21].

(1) **Anonymity**
Bidders and auctioneers can authenticate each other and then establish a common session key to protect their subsequent communications.

(2) **Bidding Privacy**
In order to protect the privacy of losers, all tender information of bidders should not be revealed except for the winner.

(3) **Bidding Price**
All bids should not be public even if auctioneer announces a winner in the opening phase. Only the highest bidding price will be announced by the auctioneer.

(4) **Unforgeability**
Nobody can counterfeit a valid bid.

(5) **Public Verifiability**
Anyone can verify whether the winner is valid or not and confirm the authenticity of the bid at the end of the auction.

(6) **Non-repudiation**
A bidder cannot deny the bid even after the auction is over.

(7) **No Framing**
Nobody can impersonate as any bidder's identity to join the bidding.

(8) **One Time Registration**
Any bidder only needs to register at registration center once and then can participate in different auctions.

(9) **Unlinkability**
Nobody can find out the relationship of bidder's identity among various auctions.

The rest of this paper is organized as follows. In Section 2, we review Li et al.'s practical electronic auction scheme and analyze the security vulnerabilities of their scheme. Then, the proposed multi-auction mechanism with dynamic identity is presented in Section 3. Later on, the security analysis and performance discussion are shown in Section 4 and 5, respectively. Finally, we make conclusions in Section6.

2. Review of Li et al.'s Scheme

In this section, we review Li et al.'s scheme [12], called a practical electronic auction scheme with strong anonymity and bidding privacy. There are four participants in their scheme, i.e., registration manager (*RM*), auction manager (*AM*), auctioneer, and bidder ($Bidder_i$). In their scheme, registration manager is a role as government department which can revoke the anonymity of a given bidder while the auction manager requests, whereas auction manager is like a bidding platform such as eBay, Yahoo! or Taobao. In order to preserve strong anonymity for bidders, the authors combine registration manager with auction manager to disperse auction manager's power [2, 3, 4]. The cooperation of registration manager and auction manager is the only way to identify the bidder who wins the auction and becomes the winner. Their scheme contains three phases: registration phase, bidding phase, and decision-of-winner and announcement phase in an English auction and the sealed-bid auction, respectively. Nevertheless, we focus on the sealed-bid auction. The notations used throughout Li et al.'s scheme are summarized in **Table 1**.

Table 1. Notations used in Li et al.'s scheme.

Notation	Definition	Notation	Definition
N	Public system parameters	a_i	Random number generated by $Bidder_i$
g	Generator of multiplication group Z_N^*	b	Random number generated by RM
$Bidder_i$	The ith bidder	c	Random number generated by AM
ID_i	Identity of $Bidder_i$	T_{RM}	Timestamp generated by RM
RM	Registration manager	T_{AM}	Timestamp generated by AM
AM	Auction manager	p_i	Bidding price of $Bidder_i$
$Auctioneer$	Auctioneer	bid_i	Tender information of $Bidder_i$
BBS_{RM}	Bulletin board system of RM	GID	Identification of goods
BBS_{AM}	Bulletin board system of AM	y_i	Registration key of $Bidder_i$
BBS_i	Bulletin board system of $Bidder_i$	Y_i	Diffie-Hellman key tween $Bidder_i$ and RM
DB_{RM}	Database of RM	Y_{RA}	Diffie-Hellman key tween RM and AM
DB_{AM}	Database of AM	Y_{A_i}	Diffie-Hellman key tween AM and $Bidder_i$
SK_i	Asymmetric private key of $Bidder_i$	$SK_i\{\cdot\}$	Signing algorithm with asymmetric private key SK_i
PK_i	Asymmetric public key of $Bidder_i$	$PK_i\{\cdot\}$	Encryption algorithm with asymmetric public key PK_i

Then, we briefly described zero knowledge proof [23] that will be used in Li et al.'s scheme. Zero knowledge proof (ZKP) is a protocol to prove that someone knows the secret value of discrete logarithms. Here, we assume that a prover knows the discrete logarithm $x = \log_g y$ for the message m, where g is a generator, and he intends to show that he knows the secret value x to the verifier. $V = ZKP(x;\ g,\ y,\ m)$ is the confirmation which implies that the prover can compute a Schnorr signature $(r,\ s)$ on the message m by calculating $t = g^a \bmod N$, $r = h(g \,\|\, y \,\|\, t \,\|\, m)$, and $s = a - rx$, where $a \in \{1, \cdots, q\}$ [4] and h denotes an one-way hash function. Hence, the verifier can verify the validity of the ture $(r,\ s)$ by confirming $r = h(g \,\|\, y \,\|\, g^s y^r \,\|\, m)$.

2.1 Review of Li et al.'s Scheme for Sealed-bid Auction

In this section, we review Li et al.'s scheme for sealed-bid auction. There are three main phases: registration phase, bidding phase, and decision-of-winner and announcement phase. The details of these phases are introduced as follows.

2.1.1 Registration Phase

Firstly, each bidder $Bidder_i$ generates his registration key and identity ID_i through the following steps:

Step 1: $Bidder_i$ chooses a random number $a_i \in Z_{N\text{-}1}$, and calculates the registration key $y_i = g^{a_i} \bmod N$.

Step 2: $Bidder_i$ registers his identity ID_i at the registration center and generates a digital signature $\delta_i = SK_i\{ID_i, y_i\}$.

Step 3: $Bidder_i$ sends the registration information (ID_i, y_i, δ_i) with *RM*. After *RM* receives the message, it confirms the digital signature δ_i by PK_i . If δ_i is valid, *RM* accepts the registration request and chooses a random number *b*. Otherwise, *RM* rejects the session. Later, *RM* calculates y_{RM} and a Diffie-Hellman session key Y_i which is also the auction key as follows:

$$y_{RM} = g^b \bmod N,$$
$$Y_i = y_i^b = (y_{RM})^{a_i} \bmod N.$$

Step 4: *RM* stores ID_i, δ_i, and Y_i into the database DB_{RM} (see **Table 2**) and then publishes N, g, y_{RM}, T_{RM}, and Y_i on the bulletin board system BBS_{RM} (see **Table 3**), where the timestamp T_{RM} is generated by *RM*. Moreover, BBS_{RM} is write-only for *RM*.

Table 2. Bulletin board system of *RM*

BBS_{RM}
$N,\ g,\ y_{RM},\ T_{RM},\ SK_{RM}\{y_{RM} \parallel T_{RM}\}$
$y_1,\ y_2,\ \cdots,\ y_n$ $Y_1,\ Y_2,\ \cdots,\ Y_n$

Table 3. Database of *RM*

DB_{RM}		
Identity	Signature	Session Key
ID_1	α_1	Y_1
ID_2	α_2	Y_2
$\vdots$	$\vdots$	$\vdots$
ID_n	α_n	Y_n

Step 5: *RM* confirms $SK_{RM}\{y_{RM} \parallel T_{RM}\}$ by PK_{RM} . If it holds, *AM* reads a Diffie-Hellman session key Y_i from BBS_{RM} and chooses a random number *c*. Later on, *AM* calculates y_{AM} , a Diffie-Hellman session key Y_{RA}, and the auction key Y_{A_i} for each bidder as follows:

$$y_{AM} = g^c \bmod N,$$
$$Y_{RA} = (y_{RM})^c = (y_{AM})^b = g^{bc} \bmod N,$$
$$Y_{A_i} = (Y_i)^c = (Y_{RA})^{a_i} = g^{a_i bc} \bmod N.$$

Step 6: *AM* stores Y_{A_i} and Y_i into the database DB_{AM} (see **Table 4**) and publishes N, g, y_{AM}, T_{AM}, Y_{RA}, and Y_{A_i} on the bulletin board system BBS_{AM} (See **Table 5**), where the timestamp T_{AM} is generated by *AM*. Moreover, BBS_{AM} is write-only for *AM*.

Table 4. Bulletin board system of *AM*

BBS_{AM}
$N,\ g,\ y_{AM},\ T_{AM},\ Y_{RA},\ SK_{AM}\{Y_{RA} \parallel T_{AM}\}$
$Y_{A_1},\ Y_{A_2},\ \cdots,\ Y_{A_n}$

Table 5. Database of *AM*

DB_{AM}	
Auction Key	Session Key
Y_{A_1}	Y_1
Y_{A_2}	Y_2
$\vdots$	$\vdots$
Y_{A_n}	Y_n

Step 7: $Bidder_i$ checks $SK_{RM}\{y_{RM} \parallel T_{RM}\}$ and $SK_{AM}\{Y_{RA} \parallel T_{AM}\}$ by PK_{RM} and PK_{AM}. If it holds, $Bidder_i$ can obtain y_{RM} and Y_{RA} to confirm a Diffie-Hellman session key Y_i from BBS_{RM} and a Diffie-Hellman session key Y_{A_i} from BBS_{AM} as follows:

$$Y_i = (y_{RM})^{a_i} = g^{a_i b} \bmod N,$$

$$Y_{A_i} \equiv (Y_{RA})^{a_i} = g^{a_i bc} \bmod N.$$

If they hold, it implies that the auction key Y_{A_i} is believed to be valid. Otherwise, $Bidder_i$ blames to *RM* or *AM*.

2.1.2 Bidding Phase

In this phase, each bidder attends the auction to bid with his own bid bid_i via the following steps:

Step 1: $Bidder_i$ selects a random number e_i and computes $F_i = PK_{Y_{A_i}}\{e_i \parallel p_i' \parallel h(e_i, p_i')\}$, where p_i' is the bidding price and $h(e_i, p_i')$ is a two-variable one-way hash function with e_i and p_i'.

Step 2: Then, $Bidder_i$ sends $bid_i' = (F_i,\ GID,\ Y_{A_i},\ V1_i')$ to *AM*, where *GID* is the identification of the good, $V1_i' = ZKP(a_i;\ Y_{RA},\ Y_{A_i},\ m_i')$ is the signature, and the message $m_i' = (F_i \parallel GID)$. If the signature $V1_i'$ is valid for the message m_i', $Bidder_i$ will certainly know the secure value a_i.

Step 3: Let T_i' be the time at which *AM* receives the bid from $Bidder_i$ and T_0' be the deadline of the auction. Subsequently, *AM* confirms the validity of the timestamp T_i', and if $T_i' < T_0'$, *AM* accepts the bid request. Otherwise, *AM* rejects it.

Step 4: *AM* or anyone can confirm whether the signature $V1_i'$ is valid. If it holds, that implies the bid is believed to be valid by the verifier. Otherwise, *AM* rejects the request.

Step 5: *AM* confirms the information $(GID,\ Y_{A_i})$ from DB_{AM}. If this information can be found in the database, the request is rejected. On the contrary, *AM* stores the information $(GID,\ Y_{A_i})$ in DB_{AM} to avoid double bidding.

Finally, *AM* publishes all values of $bid_i' = (F_i,\ GID,\ Y_{A_i},\ V1_i')$ on BBS_{AM}.

2.1.3 Decision-of-winner and Announcement Phase

While the sealed bidding is finished, each bidder posts the information $(e_i,\ p'_i,\ Y_{A_i})$ on BBS_i, where BBS_i is write-only for each bidder. Thus, anyone can verify whether the published value F_i on BBS_i is the same as the computed one. Afterwards, *AM* announces the winner and *RM* revokes the anonymity of the bidder and identifies the winner at the same time.

Step 1: Each bidder posts the information $(e_i,\ p'_i,\ Y_{A_i})$ on BBS_i.

Step 2: We assume that p'_j is the highest price at the end of bidding phase. In such a case, *AM* checks whether $F_j = PK_{Y_{A_j}}\{e_j \parallel p'_j \parallel h(e_j, p'_j)\}$ by $(e_j,\ p'_j,\ Y_{A_j})$ and a two-variable one-way hash function. If it holds, *AM* announces the winner with his winning bid $bid'_j = (F_j,\ GID,\ Y_{A_j},\ V1'_j)$. Otherwise, *AM* decides the second highest bidding price. Subsequently, anyone can confirm whether the bidding price is valid. If it holds, that means the bidding price is believed to be valid by the verifier.

Step 3: According to the relationship of Y_{A_j} and Y_j, *AM* can find Y_j through DB_{AM}.

Step 4: *AM* posts the information $(Y_j,\ V2'_j)$ on BBS_{AM} so that anyone can confirm the relationship Y_{A_j} and Y_j by the signature $V2'_j$, where $V2'_j = ZKP(c;\ Y_j,\ Y_{A_j},\ m_{RM})$.

Step 5: *RM* confirms the signature $V2'_j$ and identifies $Bidder_j$ as the winner by consulting DB_{RM} with the help of Y_j.

Step 6: Next, *RM* generates the signature $V3_j' = ZKP(b;\ y_j,\ Y_j,\ bid_j')$ and informs *Auctioneer* of $W' = PK_{Auctioneer}\{\delta_j,\ PK_j,\ V3_j'\}$, where $PK_{Auctioneer}$ is the public key of *Auctioneer* and p'_j is the bidding price of the winner $Bidder_j$.

Finally, *Auctioneer* decrypts W' to obtain $\delta_j, PK_j,$ and the signature $V3'_j$. Later on, *Auctioneer* confirms the relationship between y_j and Y_j by $V3'_j$ and verifies the winner utilizing δ_j and PK_j. Here, the winner $Bidder_j$ cannot deny his own bid since δ_j is the digital signature of $Bidder_j$.

2.2 Cryptanalysis of Li et al.'s Scheme

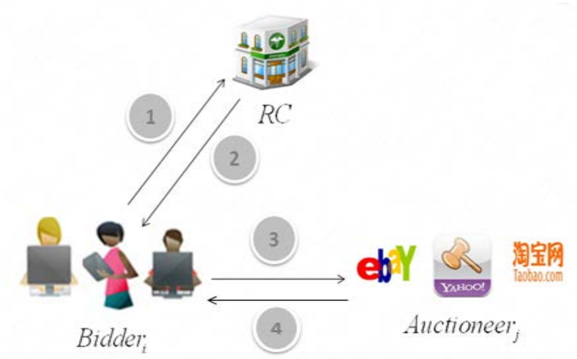
Li at al. have claimed that their scheme can withstand various attacks, however, we find out that this scheme may suffer from the denial-of-service attack during the bidding phase in a sealed-bid auction. Hence, we demonstrate the security flaw of their scheme outlined below.

If a malicious attacker *Eve* intends to launch the attack, she needs to intercept the bid request $bid'_i = (F_i,\ GID,\ Y_{A_i},\ V1'_i)$ sent from the bidder $Bidder_i$ in the bidding phase. Later on, she changes F_i to another one F_{Eve} and transmits this bid request $bid_i^* = \{F_{Eve},\ GID,\ Y_{A_i},\ V1'_i\}$ to *AM* in the valid time. Afterwards, *AM* will verify the validity of the signature $V1_i$. If the signature is not valid, *AM* will deny the request. Even though the bid request will be accepted by *AM*, a legitimate bidder $Bidder_i$ must be fail to *AM*. Since F_{Eve} is not a legal one, F_{Eve} will not pass the verification to *AM*. Thus, $Bidder_i$ cannot successfully tender a bid to *AM*. If

Eve repeatedly intercepts the bid request, replaces F_i to a new one F_{Eve}, and then sends the forged bid requests to *AM*, this will make *AM* overload by computing the Schnorr signatures. Consequently, their scheme is unable to withstand the denial-of-service attack.

3. Proposed Scheme

In this section, we propose an efficient and secure multi-auction mechanism with dynamic identity using smart card. It includes three participants in our mechanism: bidder $(Bidder_i)$, auctioneer $(Auctioneer_j)$, and registration center (*RC*). In the beginning, *RC* chooses x as the master secret key and y as the secret number. These two parameters are only known to the registration center which is considered to be trustworthy. Later, the registration center calculates and shares the hash value $h(x)$, $h(x \| y)$, and $h(AID_j \| h(y))$ for each auctioneer via a secure channel while $Auctioneer_j$ registers himself to the registration center with his identity AID_j. Most important of all, we have assumed that the information stored in the smart card cannot be extracted to be utilized in our proposed mechanism. The proposed mechanism contains five phases: registration phase, login phase, mutual authentication and key agreement phase, sealed bidding phase, and winner announcement phase. The notations used throughout the proposed mechanism are shown in **Table 6**.



1. Bidders register at *RC*.

2. *RC* issues the smart card to each bidder.

3. Bidders join an action.

4. Auctioneer announces the winner.

Fig. 1. The flowchart of multi-auction mechanism

The flowchart of multi-auction mechanism is described in **Fig. 1**. If someone wants to participate in an auction, he must register at the registration center first. Later on, the registration center will issue the smart card so that each bidder can join in any auction using smart card. Afterwards, each bidder negotiates a common session key to ensure that the following communications are secure. After receiving the bids from bidders, the auctioneer will compare all bidding prices and pick out the highest one as the winner.

Table 6. Notations used in multi-auction mechanism.

Notation	Definition	Notation	Definition
$Bidder_i$	The ith bidder	r_i	Random number chosen by RC for each bidder
UID_i	Identity of $Bidder_i$	N_i	Random nonce generated by $Bidder_i$'s smart card
PW_i	Password of $Bidder_i$	N_j	Random nonce generated by $Auctioneer_j$
CID_i	Dynamic identity of $Bidder_i$	K_i	Session key shared between $Bidder_i$ and $Auctioneer_j$
bid_i	Bidding price of $Bidder_i$	$[]_{K_i}$	Symmetric encryption using the common session key K_i
$Auctioneer_j$	The jth auctioneer	$h(.)$	A one-way hash function
AID_j	Unique identity of $Auctioneer_j$	$\oplus$	Exclusive-or operation
RC	Registration center	$\|\|$	Message concatenation operation
x	Master secret key maintained by RC	$\Rightarrow$	A secure channel
y	Secret number generated by RC	$\rightarrow$	A common channel

3.1 Registration Phase

The bidder $Bidder_i$ must register at the registration center before participating in an auction. The flowchart of registration phase is illustrated in **Fig. 2.**

Step 1: $Bidder_i$ arbitrarily chooses his identity UID_i, password PW_i and calculates $P_i = h(PW_i)$. After that, $Bidder_i$ submits UID_i and P_i to RC via a secure channel.

Step 2: After receiving the message UID_i and P_i from $Bidder_i$, RC calculates parameters A_i, B_i, C_i, D_i, E_i as follows:

$$\begin{aligned} A_i &= h(UID_i \,\|\, x) \\ B_i &= h(UID_i \,\|\, P_i) \\ C_i &= h(A_i \,\|\, B_i) \oplus h(h(x) \,\|\, r_i) \\ D_i &= r_i \oplus h(x \,\|\, y) \\ E_i &= P_i \oplus A_i. \end{aligned}$$

Finally, RC stores these security parameters $\{B_i, C_i, D_i, E_i, h(.), h(y)\}$ into the smart card and sends back to $Bidder_i$ through a secure channel.

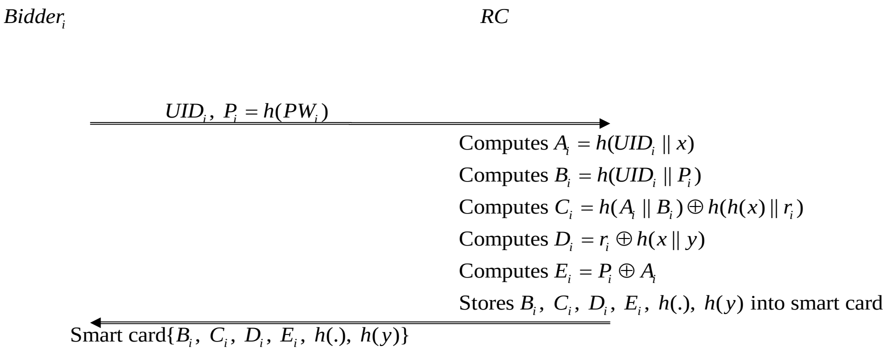

$Bidder_i$
RC
UID_i, $P_i = h(PW_i)$
Computes $A_i = h(UID_i \| x)$
Computes $B_i = h(UID_i \| P_i)$
Computes $C_i = h(A_i \| B_i) \oplus h(h(x) \| r_i)$
Computes $D_i = r_i \oplus h(x \| y)$
Computes $E_i = P_i \oplus A_i$
Stores B_i, C_i, D_i, E_i, $h(.)$, $h(y)$ into smart card
Smart card$\{B_i, C_i, D_i, E_i, h(.), h(y)\}$

Fig. 2. The flowchart of registration phase

3.2 Login Phase

When $Bidder_i$ wants to join in the auction, he needs to perform the following steps to generate the request.

Step 1: He has to insert his smart card into the card reader and enters his identity UID_i and password PW_i.

Step 2: The smart card calculates $B_i^* = h(UID_i \| h(PW_i)) = h(UID_i \| P_i)$ and confirms whether B_i^* is equal to B_i, where the value B_i is stored in the smart card. If they are equal, it implies that $Bidder_i$ is a legitimate user. $Bidder_i$ carries on the following steps. Otherwise, the smart card rejects the request.

Step 3: After confirming $Bidder_i$ is a legitimate user, the smart card generates a random nonce N_i and calculates the following steps:

$$\begin{aligned}
&A_i = P_i \oplus E_i \\
&h(h(x) \| r_i) = h(A_i \| B_i) \oplus C_i \\
&CID_i = C_i \oplus h(h(h(x) \| r_i) \| N_i \| AID_j) \\
&P_{ij} = D_i \oplus AID_j \oplus N_i \\
&\alpha_i = h(AID_j \| h(y)) \oplus N_i \\
&\beta_i = h(h(A_i \| B_i) \| N_i \| P_{ij} \| CID_i).
\end{aligned}$$

Finally, $Bidder_i$ submits $m_1 = \{CID_i,\ P_{ij},\ \alpha_i,\ \beta_i\}$ as the login request message to $Auctioneer_j$ over a public channel.

3.3 Mutual Authentication and Key Agreement Phase

In this phase, $Auctioneer_j$ will authenticate each bidder after receiving the login message. Later on, both of them accomplish the common key while the authentication is over. Shortly afterwards, $Bidder_i$ can utilize the session key to communicate with $Auctioneer_j$. The flowchart of mutual authentication and key agreement phase is illustrated in **Fig. 3**, and the detailed steps are described as follows.

Step 1: Upon receiving the login request message $m_1 = \{CID_i,\ P_{ij},\ \alpha_i,\ \beta_i\}$ from $Bidder_i$, $Auctioneer_j$ utilizes the pre-shared hash value $h(x)$, $h(x \| y)$ and $h(AID_j \| h(y))$ to calculate the following:

$$N_i = h(AID_j \| h(y)) \oplus \alpha_i$$
$$D_i = P_{ij} \oplus AID_j \oplus N_i$$
$$r_i = D_i \oplus h(x \| y)$$
$$C_i = CID_i \oplus h(h(h(x) \| r_i) \| N_i \| AID_j)$$
$$h(A_i \| B_i) = C_i \oplus h(h(x) \| r_i)$$
$$\beta_i^* = h(h(A_i \| B_i) \| N_i \| P_{ij} \| CID_i).$$

Step 2: Later on, $Auctioneer_j$ confirms whether the computed value β_i^* is equal to the received β_i . If they hold, $Auctioneer_j$ authenticates $Bidder_i$ successfully. Thus, it implies that $Bidder_i$ is an authorized user, and then $Auctioneer_j$ further chooses a random nonce N_j . Otherwise, $Auctioneer_j$ rejects the login request in this session.

Step 3: $Auctioneer_j$ calculates and sends $\gamma_i = h(A_i \| B_i) \oplus N_i \oplus N_j$, $\delta_i = h(h(A_i \| B_i) \| N_j \| AID_j)$ to $Bidder_i$.

Step 4: Upon receiving the message $m_2 = \{\gamma_i, \delta_i\}$ from $Auctioneer_j$, $Bidder_i$ calculates $N_j = h(A_i \| B_i) \oplus N_i \oplus \gamma_i$ and $\delta_i^* = h(h(A_i \| B_i) \| N_j \| AID_j)$. Later, $Bidder_i$ confirms whether the computed value δ_i^* is equal to δ_i . If they are equal, it means that $Bidder_i$ successfully authenticates $Auctioneer_j$ which is an authorized service provider. Shortly afterwards, $Bidder_i$ calculates the mutual authentication message $\theta_i = h(h(A_i \| B_i) \| N_j \| AID_j)$ and submits the message $m_3 = \{\theta_i\}$ to $Auctioneer_j$. Otherwise, $Bidder_i$ rejects the message and terminates the session.

Step 5: Upon receiving the message $m_3 = \{\theta_i\}$ from $Bidder_i$, $Auctioneer_j$ calculates $\theta_i^* = h(h(A_i \| B_i) \| N_j \| AID_j)$ and confirms whether the computed value θ_i^* is equal to θ_i . If they are the same, it implies that $Auctioneer_j$ successfully authenticates $Bidder_i$. The mutual authentication is completed. After the mutual authentication phase, $Bidder_i$ can negotiate the session key $K_i = h(h(A_i \| B_i) \| N_i \| N_j \| AID_j)$ with $Auctioneer_j$.

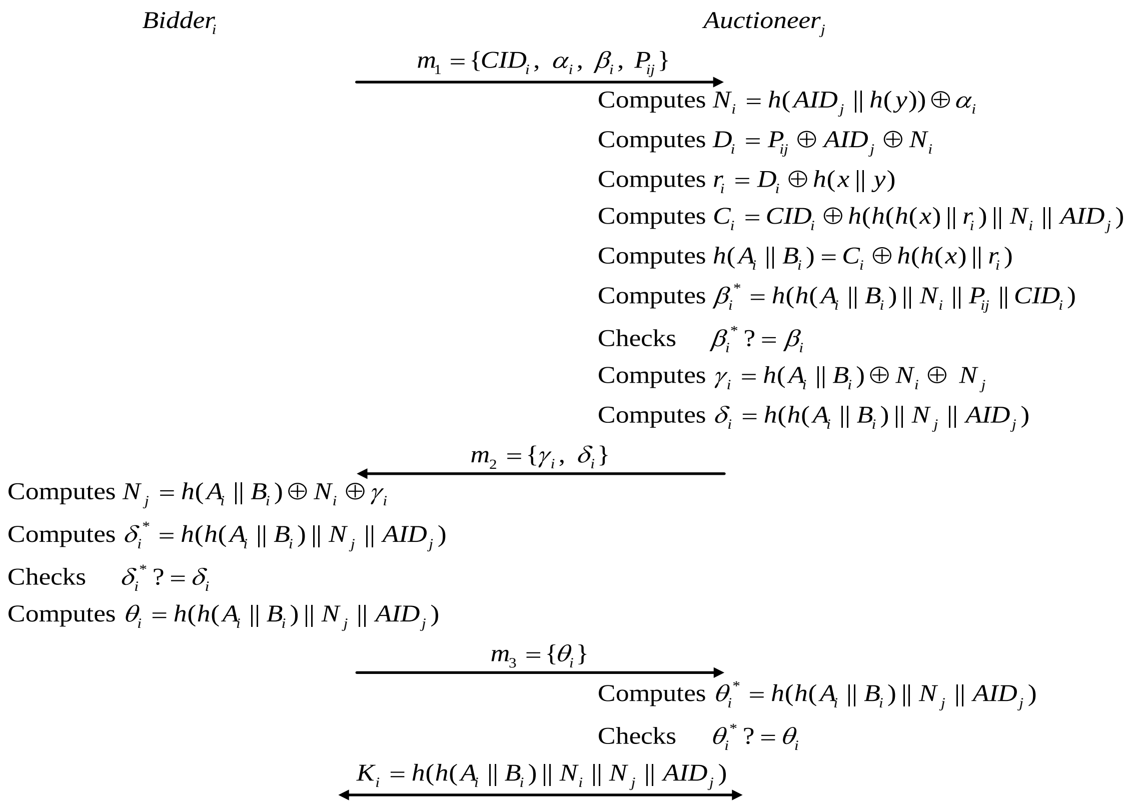

$Bidder_i$
$Auctioneer_j$
$m_1 = \{CID_i,\ \alpha_i,\ \beta_i,\ P_{ij}\}$
Computes $N_i = h(AID_j \| h(y)) \oplus \alpha_i$
Computes $D_i = P_{ij} \oplus AID_j \oplus N_i$
Computes $r_i = D_i \oplus h(x \| y)$
Computes $C_i = CID_i \oplus h(h(h(x) \| r_i) \| N_i \| AID_j)$
Computes $h(A_i \| B_i) = C_i \oplus h(h(x) \| r_i)$
Computes $\beta_i^* = h(h(A_i \| B_i) \| N_i \| P_{ij} \| CID_i)$
Checks $\beta_i^* ?= \beta_i$
Computes $\gamma_i = h(A_i \| B_i) \oplus N_i \oplus N_j$
Computes $\delta_i = h(h(A_i \| B_i) \| N_j \| AID_j)$
$m_2 = \{\gamma_i,\ \delta_i\}$
Computes $N_j = h(A_i \| B_i) \oplus N_i \oplus \gamma_i$
Computes $\delta_i^* = h(h(A_i \| B_i) \| N_j \| AID_j)$
Checks $\delta_i^* ?= \delta_i$
Computes $\theta_i = h(h(A_i \| B_i) \| N_j \| AID_j)$
$m_3 = \{\theta_i\}$
Computes $\theta_i^* = h(h(A_i \| B_i) \| N_j \| AID_j)$
Checks $\theta_i^* ?= \theta_i$
$K_i = h(h(A_i \| B_i) \| N_i \| N_j \| AID_j)$

Fig. 3. The flowchart of mutual authentication and key agreement phase

3.4 Sealed Bidding Phase

After the key agreement phase, $Bidder_i$ can utilize the session key K_i to participate in bidding. The flowchart of sealed bidding phase is illustrated in **Fig. 4**.

Step 1: $Bidder_i$ arbitrarily chooses the bidding price bid_i . Then, he calculates $w_i = h(CID_i \| bid_i \| K_i \| AID_j)$ with his dynamic identity CID_i and $msg_i = [CID_i,\ bid_i,\ w_i]_{K_i}$ by the session key K_i . Here, it not only protects the tender information of the bidder, but also confirms integrity and confidentiality of the tender information.

Step 2: $Bidder_i$ submits his tender information $m_4 = \{CID_i,\ msg_i\}$ to $Auctioneer_j$.

Step 3: Upon receiving the tender information $m_4 = \{CID_i,\ msg_i\}$ from $Bidder_i$. $Auctioneer_j$ decrypts $msg_i = [CID_i,\ bid_i,\ w_i]_{K_i}$ to obtain CID_i, bid_i and w_i by the common session key K_i . Later on, it confirms whether CID_i and W_i is valid or not. If it does not hold, the bidding price bid_i can be treated as nullity. Otherwise, $Bidder_i$ and the bidding price bid_i will be considered as legal. Finally, $Bidder_i$ successfully bids in this sealed-bid auction.

$Bidder_i$ $Auctioneer_j$

Computes $w_i = h(CID_i \| bid_i \| K_i \| AID_j)$

Computes $msg_i = [CID_i,\ bid_i,\ w_i]_{K_i}$

$$\xrightarrow{m_4 = \{CID_i,\ msg_i\}}$$

Decrypts $msg_i = [CID_i, bid_i, w_i]_{K_i}$ to obtain CID_i, bid_i, w_i

Computes $w_i^* = h(CID_i \| bid_i \| K_i \| AID_j)$

Confirms $w_i^* ?= w_i$

Fig. 4. The flowchart of sealed bidding phase

3.5 Winner Announcement Phase

Once $Auctioneer_j$ receives all bids from bidders, it compares all bidding prices and chooses the highest bidding one among bidders as the winner. Finally, the auctioneer announces the winner with his tender information CID_i, msg_i, session key K_i, and the bidding price bid_i.

4. Analysis

Here, we analyze the security of our proposed mechanism which is based on symmetric encryption function, one-way hash function, and exclusive-or operation in Subsection 4.1. Furthermore, the assumptions of symmetric encryption function, one-way hash function, and exclusive-or operation are depicted as below. Later on, the requirements of the sealed-bid auction can be achieved in our proposed mechanism are shown in Subsection 4.2.

Symmetric encryption function $[.]_K$

According to [25], it is computationally infeasible to obtain the plaintext M from the encrypted message $[M]_K$ in the case of unknown encryption key K.

One-way hash function $h(.)$

Variable size message is input and fixed size result returned. One-way hash function $h(.)$ is defined as follows [26].

Pre-image resistance: Given the message digest $y = h(M)$, it is computationally infeasible to derive the message M from y.

Second pre-image resistance: Given the message M and the message digest $y = h(M)$, it is computationally infeasible to find another message M' to satisfy that $M' \neq M$ and $h(M') = y$.

Collision resistance: It is computationally infeasible to pick out two arbitrary plaintexts M, M' such that $M' \neq M$ and $h(M') = h(M)$.

Exclusive-or operation $\oplus$

The exclusive-or operation shall not be compromised in polynomial time. Given the message M_1, M_2, it is computationally infeasible to derive the message M_1 from the ciphertext C without knowing the message M_2 in polynomial time. However, it is easily to calculate $C = M_1 \oplus M_2$.

4.1 Security Analysis

4.1.1 Perfect Forward Secrecy

In our proposed mechanism, both the bidder $Bidder_i$ and the auctioneer $Auctioneer_j$ can utilize the shared information $h(A_i \| B_i)$ to calculate the common session key $K_i = h(h(A_i \| B_i) \| N_i \| N_j \| AID_j)$ with the random nonce N_i and N_j. Even if a malicious attacker *Eve* can compromise $K_i = h(h(A_i \| B_i) \| N_i \| N_j \| AID_j)$, she still cannot recover the previous session keys or speculate the session keys for the future. Since each session key contains an individual random nonce, this makes all session keys be distinct. However, she cannot learn of the random nonce under the assumption of pre-image resistance. Even though *Eve* can obtain a valid random nonce, she still cannot construct the common session key without the knowledge of $h(A_i \| B_i)$ under the assumption of pre-image resistance. As a result, the proposed mechanism can provide perfect forward secrecy.

4.1.2 Replay Attack

We assume that a malicious attacker *Eve* intends to intercept the login request message $m_1 = \{CID_i, P_{ij}, \alpha_i, \beta_i\}$ from the bidder $Bidder_i$ to launch the replay attack. Then, *Eve* will replay the request message to $Auctioneer_j$. However, this login request message must fail since $Auctioneer_j$ can confirm whether the random nonce is fresh or not. As a result, the proposed mechanism is secure against the replay attack.

4.1.3 Impersonation Attack

Impersonation attack means that a malicious attacker *Eve* intends to masquerade as the legal bidder $Bidder_i$ to login to the auctioneer $Auctioneer_j$. First, she needs to forge the login request message $\alpha_i, P_{ij}, CID_i, \beta_i$ sequentially and send them to $Auctioneer_j$. However, it is computationally infeasible for *Eve* to generate the first parameter $\alpha_i = h(AID_j \| h(y)) \oplus N_i$ without knowing $h(AID_j \| h(y))$ and N_i under the assumption of the exclusive-or function. Owing to unknown nonce N_i, she cannot separately generate other parameters P_{ij}, CID_i, β_i which are also associated with the random nonce N_i under the assumption of the exclusive-

or function and the pre-image resistance. Despite she is able to obtain a valid nonce, it is useless. The reason is that *Eve* cannot construct the first parameter α_i without $h(AID_j \,||\, h(y))$ under the assumption of the exclusive-or function. For the above-mentioned reasons, *Eve* has no way to launch this attack so that the proposed mechanism can prevent the impersonation attack.

4.1.4 Server Spoofing Attack

Server spoofing attack means that if a legal bidder $Bidder_k$ (*Eve*) intends to fabricate as the auctioneer $Auctioneer_j$ to fool and response $m_2 = \{\gamma_i,\ \delta_i\}$ to the bidder $Bidder_i$, she must fail. The reason is that it is impossible for $Bidder_k$ to calculate $\gamma_i = h(A_i \,||\, B_i) \oplus N_i \oplus N_j$ without the knowledge of $h(A_i \,||\, B_i)$ and N_i under the assumption of the exclusive-or function as well as $\delta_i = h(h(A_i \,||\, B_i) \,||\, N_i \,||\, N_j \,||\, AID_j)$ without the knowledge of $h(A_i \,||\, B_i)$ and N_i under the assumption of the pre-image resistance. Furthermore, N_i is protected by $\alpha_i = h(AID_j \,||\, h(y)) \oplus N_i$. Without the hash value $h(AID_j \,||\, h(y))$, it is infeasible for $Bidder_k$ to retrieve N_i from $\alpha_i = h(AID_j \,||\, h(y)) \oplus N_i$ under the assumption of the exclusive-or function. Hence, $Bidder_k$ cannot transfer a valid response $m_2 = \{\gamma_i,\ \delta_i\}$ to $Bidder_i$.

Similarly, a legal auctioneer $Auctioneer_k$ (*Eve*) may try to imitate as the auctioneer $Auctioneer_j$ to cheat the play. However, it is computationally infeasible to construct $h(AID_j \,||\, h(y))$ without secret hash value $h(y)$ under the assumption of pre-image resistance. Here, no one but RC can know secret y and $h(y)$. Moreover, the pre-shared hash value $h(AID_k \,||\, h(y))$ from $Auctioneer_k$ will not be equal to $h(AID_j \,||\, h(y))$ from $Auctioneer_j$. Therefore, $\alpha_i = h(AID_j \,||\, h(y)) \oplus N_i$ will not be valid under the assumption of exclusive-or function. For the above-mentioned reasons, no one can impersonate a legal auctioneer to spoof bidders in the proposed mechanism.

4.1.5 Man-in-the-middle Attack

In this type of attack, we assumed that a malicious attacker *Eve* can intercept the login request messages sent from $Bidder_i$ to the auctioneer $Auctioneer_j$ and then transfer the modified messages to $Bidder_i$. In the beginning, *Eve* intercepts the login request message $m_1 = \{CID_i,\ \alpha_i,\ \beta_i,\ P_{ij}\}$ from $Bidder_i$ and the response message $m_2 = \{\gamma_i,\ \delta_i\}$ from $Auctioneer_j$. Later on, *Eve* starts a new session with $Bidder_i$ by transmitting the modified response $m_2' = \{\gamma_i',\ \delta_i'\}$. However, *Eve* cannot alter any message such as $\gamma_i = h(A_i \,||\, B_i) \oplus N_i \oplus N_j$ without knowing $h(A_i \,||\, B_i)$ and N_i under the assumption of exclusive-or function and $\delta_i = h(h(A_i \,||\, B_i) \,||\, N_j \,||\, AID_j)$ without the knowledge of $h(A_i \,||\, B_i)$ under the assumption of pre-image resistance. Despite she can obtain CID_i and P_{ij} from intercepting the login request message m_1, she still cannot retrieve $h(A_i \,||\, B_i)$ from $\beta_i = h(h(A_i \,||\, B_i) \,||\, N_i \,||\, P_{ij} \,||\, CID_i)$ under the assumption of pre-image resistance.

Therefore, *Eve* is unable to launch the attack and the proposed mechanism is secure against man-in-the-middle attack.

4.1.6 Denial-of-service Attack

The resistance to denial-of-service attack means that it can prevent legitimate bidders from failing to auctioneers, namely auctioneers cannot provide bidding service to bidders properly. Here, we demonstrate that denial-of-service attack cannot be launched by *Eve* in our proposed mechanism. If *Eve* intends to launch this attack, she needs to modify the login request message m_1 to pass the verification with her nonce N_i^*. However, it must be fail since it is computationally infeasible for *Eve* to generate a valid nonce without hash value $h(AID_j \| h(y))$ under the assumption of exclusive-or function, where $N_i^* = h(AID_j \| h(y)) \oplus \alpha_i^*$. *Eve* cannot successfully generate these parameters α_i, CID_i, β_i, P_{ij} to pass the verification, which are associated with the random nonce N_i. Thus, it is difficult for malicious attackers to launch an effective denial-of-service attack in our proposed mechanism.

4.1.7 Proper Mutual Authentication and Key Agreement

Upon receiving the login request message $m_1 = \{CID_i,\ \alpha_i,\ \beta_i,\ P_{ij}\}$ from $Bidder_i$, $Auctioneer_j$ will calculate the parameters N_i, D_i, r_i, C_i, $h(A_i \| B_i)$ sequentially and confirm whether the computed β_i^* is equal to β_i. If so, $Bidder_i$ is successfully authenticated by $Auctioneer_j$. Here, only the proper $Auctioneer_j$ can filter out N_i using $h(AID_j \| h(y)) \oplus \alpha_i$ under the assumption of exclusive-or function since $h(AID_j \| h(y))$ is only known to $Auctioneer_j$. Hence, it can continually calculate $D_i = P_{ij} \oplus AID_j \oplus N_i$ with P_{ij}, AID_j, and the valid nonce N_i. After obtaining D_i, it uses the pre-shared hash value $h(x \| y)$ to compute r_i , where $r_i = D_i \oplus h(x \| y)$. In the case of owning these parameters CID_i, $h(x)$, r_i, N_i and AID_j, $Auctioneer_j$ can easily calculate $C_i = CID_i \oplus h(h(h(x) \| r_i) \|\ N_i \| AID_j)$, where $h(x)$ is pre-shared by *RC* in the registration phase. Thanks to these two values C_i and $h(h(x) \| r_i)$, $Auctioneer_j$ can further compute $h(A_i \| B_i) = C_i \oplus h(h(x) \| r_i)$. Thus, $Auctioneer_j$ can authenticate $Bidder_i$ at this moment by verifying whether the computed $\beta_i^* = h(h(A_i \| B_i) \| N_i \| P_{ij} \| CID_i)$ is equal to β_i. Obviously, if it holds, it implies that $Bidder_i$ is a legitimate user, and the login request message is accepted by $Auctioneer_j$.

$Bidder_i$ checks that $Auctioneer_j$ is a legitimate service provider by confirming whether the computed δ_i^* is equal to the received one δ_i under the assumption of second pre-image resistance. If they are the same, $Auctioneer_j$ is successfully authenticated by $Bidder_i$. Here, $Bidder_i$ can obtain N_j from $\gamma_i = h(A_i \| B_i) \oplus N_i \oplus N_j$ under the assumption of exclusive-or function using the hash value $h(A_i \| B_i)$ which is only known to himself and N_i which is generated by the smart card. Thus, no one but $Bidder_i$ can acquire the valid nonce N_j sent from $Auctioneer_j$. Afterwards, $Bidder_i$ calculates the mutual message $\theta_i = h(h(A_i \| B_i) \| N_i \| AID_j)$ and submits it to $Auctioneer_j$. Later on, it will verify whether the computed θ_i^* is equal to the received one θ_i under the assumption of second pre-image resistance. It is clear that $m_3 = \{\theta_i\}$ will pass the verification. Hence, $Bidder_i$ and $Auctioneer_j$ can mutually authenticate and accomplish the key $K_i = h(h(A_i \| B_i) \| N_i \| N_j \| AID_j)$ in our proposed mechanism.

4.1.8 Formal Mutual Authentication according to BAN Logic Model

In order to confirm the identity for communicating participants, we demonstrate the achievement of mutual authentication between bidders and auctioneers based on BAN logic [27, 28]. The notations of BAN logic are described in Table 7.

Table 7. Notations used in BAN logic.

Notation	Definition	Notation	Definition
X	Statement	$P \mid\Rightarrow X$	P has *jurisdiction* over X .
$P,\ Q$	Parties	$\#(X)$	The formula X is *fresh.*
$P \mid\equiv X$	P *believes* X	$P \xleftrightarrow{K} Q$	P communicates to Q by utilizing the *shared key* K .
$P \lhd X$	P *receives* X	$P \overset{K}{\rightleftharpoons} Q$	The formula X is a *secret* only known between P and Q .
$P \mid\sim X$	P once *said* X	$\langle X \rangle_Y$	X combined with the formula Y ; it is implied that Y be a secret.

Here, we describe the logical assumptions of BAN logic that we use in the proof as below.

$$(1)\ Message-meaning: \frac{P \mid\equiv P \overset{Y}{\rightleftharpoons} Q, P \lhd \langle X \rangle_Y}{P \mid\equiv Q \mid\sim X} \qquad (4)\ Receiving: \frac{P \lhd \langle X \rangle_Y}{P \lhd X}$$

$$(2)\ Nonce-verification: \frac{P \mid\equiv \#(X), P \mid\equiv Q \mid\sim X}{P \mid\equiv Q \mid\equiv X} \qquad (5)\ Freshness\ Propagation: \frac{P \mid\equiv \#(X)}{P \mid\equiv \#(X,Y)}$$

$$(3)\ Jurisdiction: \frac{P \mid\equiv Q \mid\Rightarrow X, P \mid\equiv Q \mid\equiv X}{P \mid\equiv X} \qquad (6)\ Session\ Key: \frac{P \mid\equiv \#(K), P \mid\equiv Q \mid\equiv X}{P \mid\equiv P \xleftrightarrow{K} Q}$$

The rule of *Session Key* is an extension rule for the combination key [29] in BAN logic. Note that X is a critical element of the combination key K.

For simplicity, we transfer our proposed mechanism into the generic form **M1, M2, M3** as follows.

$$\textbf{M1}.\ Bidder_i \rightarrow Auctioneer_j : h(AID_j \| h(y)),\ r_i \oplus h(x \| y) \oplus AID_j \oplus N_i,\ C_i \oplus h(h(h(x) \| r_i) \| N_i \| AID_j),$$
$$h(h(A_i \| B_i) \| N_i \| P_{ij} \| CID_i)$$

$$\textbf{M2}.\ Auctioneer_j \rightarrow Bidder_i : h(A_i \| B_i) \oplus N_i \oplus N_j,\ h(h(A_i \| B_i) \| N_j \| AID_j)$$

$$\textbf{M3}.\ Bidder_i \rightarrow Auctioneer_j : h(h(A_i \| B_i) \| N_j \| AID_j)$$

Later on, we further convert the generic form **M1, M2, M3** into the idealized from **I1**, **I2**, **I3** and list the corresponding goals as below.

I1. $Bidder_i \rightarrow Auctioneer_j : \langle N_i \rangle_{h(AID_j \| h(y))},\ \left\langle r_i \right\rangle_{h(x\|y),\, N_i}, \left\langle \left\langle h(A_i \parallel B_i) \right\rangle_{h(x),\, r_i} \right\rangle_{N_i},\ \langle N_i \rangle_{h(A_i\|B_i)}$

I2. $Auctioneer_j \rightarrow Bidder_i : \langle N_i,\ N_j,\ Auctioneer_j \mid\equiv h(A_i \parallel B_i) \rangle_{h(A_i\|B_i)},\ \left\langle N_j \right\rangle_{h(A_i\|B_i)}$

I3. $Bidder_i \rightarrow Auctioneer_j : \left\langle N_j \right\rangle_{h(A_i\|B_i)}$

G1. $Auctioneer_j \mid\equiv N_i$

G2. $Bidder_i \mid\equiv N_j$

G3. $Auctioneer_j \mid\equiv Bidder_i \mid\equiv N_j$

G4. $Auctioneer_j \mid\equiv Bidder_i \xleftrightarrow{K_i} Auctioneer_j$

G5. $Bidder_i \mid\equiv Bidder_i \xleftrightarrow{K_i} Auctioneer_j$

Here, we use $\left\langle X \right\rangle_Y$ to denote the result of exclusive-oring X with the secret Y for exclusive-or operation representation. Moreover, the hash function can be processed similarly as the above-mentioned in our proposed mechanism. To complete the proof, we define essential assumptions to prove that the communications can be established in our proposed mechanism.

A1. $Auctioneer_j \mid\equiv RC \overset{h(AID_j \| h(y))}{\rightleftarrows} Auctioneer_j$

A2. $Auctioneer_j \mid\equiv RC \overset{h(x\|y)}{\rightleftarrows} Auctioneer_j$

A3. $Auctioneer_j \mid\equiv RC \overset{h(x)}{\rightleftarrows} Auctioneer_j$

A4. $Auctioneer_j \mid\equiv \#(N_i)$

A5. $Auctioneer_j \mid\equiv Bidder_i \mid\Rightarrow N_i$

A6. $Bidder_i \mid\equiv N_i$

A7. $Bidder_i \mid\equiv \#(N_j)$

A8. $Bidder_i \mid\equiv Auctioneer_j \mid\Rightarrow N_j$

Theorem 4.1 Bidders and auctioneers can authenticate each other in our proposed mechanism.

Proof: We infer the goals **G1**, **G2**, **G3** to show that both bidders and auctioneers can mutually authenticate.

For the first goal **G1**, we can derive following formulas.

$\boxed{\textbf{R1}.\ Auctioneer_j \lhd N_i}$ (using **I1** and **A1** based on *Receiving* rule)

$\boxed{\textbf{R2}.\ Auctioneer_j \lhd r_i}$

$\boxed{\textbf{R3}.\ Auctioneer_j \lhd h(A_i \parallel B_i)}$

$\boxed{\textbf{R4}.\ Auctioneer_j \mid\equiv Bidder_i \overset{h(A_i\|B_i)}{\rightleftarrows} Auctioneer_j}$

$\boxed{\textbf{R5}.\ Auctioneer_j \mid\equiv Bidder_i \mid\sim N_i}$ (using **R3** and **R4** based on *Message-meaning* rule)

The formula **R1** implies that $Auctioneer_j$ can use **I1** and **A1** to retrieve the random nonce N_i. Hence, we assume that $Auctioneer_j$ will temporarily believe N_i once he retrieves it. Afterward, we derive that $Auctioneer_j$ can further retrieve r_i from $\left\langle r_i \right\rangle_{h(x\|y),\, N_i}$ using **A2** and **R1** (i.e., the formula **R2**).

Similarly, we can derive the formula **R3** according to the derivation of the formula **R2** which is derived from the formula **R1**. Note that only the correct $Auctioneer_j$ can successfully retrieve N_i from $Bidder_i$ using **A1** and further retrieve the secret value $h(A_i \parallel B_i)$ of $Bidder_i$ using **R2**, **A2**, and **A3**. Obviously, we can deduce that no one but $Bidder_i$ and the correct auctioneer $Auctioneer_j$ can know the hash value $h(A_i \| B_i)$. That is, once the $Auctioneer_j$ retrieves $h(A_i \| B_i)$ (i.e., the formula **R3**), he will believe that it is only shared with the $Bidder_i$ (i.e., the formula **R4**).

Subsequently, we can proceed to the proof of **G1** as follows.

$\boxed{\textbf{R6}.\ Auctioneer_j \mid\equiv Bidder_i \mid\equiv N_i}$ (using **R5** and **A4** based on *Nonce-verification* rule)

$\boxed{\textbf{R7}.\ Auctioneer_j \mid\equiv N_i}$ (using **R6** and **A5** based on *Jurisdiction* rule)

Later on, the second goal **G2** is inferred in the same manner as below.

$\boxed{\textbf{R8}.\ Bidder_i \lhd (N_i,\ N_j,\ Auctioneer_j \mid\equiv h(A_i \| B_i))}$ (using **I2** based on *Receiving* rule)

$\boxed{\textbf{R9}.\ Bidder_i \mid\equiv Bidder_i \xleftrightarrow{h(A_i \| B_i)} Auctioneer_j}$

The formula **R8** implies that $Bidder_i$ can retrieve the random nonce N_j using **I2**. Hence, we assume that $Bidder_i$ will temporarily believe N_j once he retrieves it. Note that only the correct $Bidder_i$ can retrieve N_j from $Auctioneer_j$ using **I2**. Obviously, we can deduce that no one but the correct $Bidder_i$ and the correct auctioneer $Auctioneer_j$ can know the hash value $h(A_i \| B_i)$. Thus, $Bidder_i$ will believe that the hash value $h(A_i \| B_i)$ is only shared with $Auctioneer_j$ (i.e., the formula **R9**).

$\boxed{\textbf{R10}.\ Bidder_i \mid\equiv Auctioneer_j \mid\sim N_j}$ (using **R8** and **R9** based on *Message-meaning* rule)

$\boxed{\textbf{R11}.\ Bidder_i \mid\equiv Auctioneer_j \mid\equiv N_j}$ (using **R**10 and **A**7 based on *Nonce-verification* rule)

$\boxed{\textbf{R12}.\ Bidder_i \mid\equiv N_j}$ (using **R11** and **A8** based on *Jurisdiction* rule)

Finally, we can obtain the third goal **G3**, inferred as below.

$\boxed{\textbf{R13}.\ Auctioneer_j \mid\equiv Bidder_i \mid\sim N_j}$(using **I2** and **R9** based on *Message-meaning* rule)

$\boxed{\textbf{R14}.\ Auctioneer_j \mid\equiv Bidder_i \mid\equiv N_j}$ (using **R13** based on *Nonce-verification* rule)

As the above-mentioned rules and assumptions, we can infer that the goals **G1**, **G2**, **G3** are achieved.

Theorem 4.2 Bidders and auctioneers can generate a common session key in our proposed mechanism. Namely, both of them have authenticated each other.

Proof: We infer the goals **G4**, **G5** to show that both bidders and auctioneers can negotiate a common session key. Besides, we expand the session key before beginning the proof as below:

$$K_i = h(h(A_i \| B_i) \| N_i \| N_j \| AID_j)$$

Clearly, we can find out that if both $Bidder_i$ and $Auctioneer_j$ intend to negotiate this session key K_i, they must have a common secret value $h(A_i \| B_i)$.

As demonstrated in **Theorem 4.1**, we can deduce the formula **R4** and **R9**. Later on, we can further derive following formulas for the goal **G4**.

$\boxed{\textbf{R15}.\ Auctioneer_j \mid\equiv \#(K_i)}$ (using **A4** and **R4** based on *Freshness Propagation* rule)

$\boxed{\textbf{R16}.\ Auctioneer_j \mid\equiv Bidder_i \xleftrightarrow{K_i} Auctioneer_j}$ (using **R15** based on *Session Key* rule)

Afterwards, the goal **G5** can be deduced similarly.

With the formula **R16**, we can deduce that $Auctioneer_j$ believes that it has the common session key K_i shared with $Bidder_i$.

$\boxed{\textbf{R17}.\ Bidder_i \mid\equiv \#(K_i)}$ (using **A7** and **R9** based on *Freshness Propagation* rule)

$\boxed{\textbf{R18}.\ Bidder_i \mid\equiv Bidder_i \xleftrightarrow{K_i} Auctioneer_j}$ (using **R17** based on *Session Key* rule)

According to the formula **R18**, we can summarize that $Bidder_i$ believes that it has the common session key K_i shared with $Auctioneer_j$.

Corollary 4.1 Bidders and auctioneers that run in the proposed mechanism can not only authenticate each other, but also share a common session key.

Proof: From **Theorem 4.1**, bidders and auctioneers can provide mutual authentication to each other. On the basis of **Theorem 4.2**, bidders and auctioneers can negotiate a common session key in our proposed mechanism. In other words, both of them can be authenticated each other and communicate with a common session key in our proposed mechanism. Hence, we can provide the formal proof of the correctness of mutual authentication.

4.2 Essential Requirements

4.2.1 Anonymity

Each bidder will generate the dynamic identity CID_i which is similar to the pseudonym before the sealed bidding phase and then submit the tender information msg_i with his dynamic identity CID_i to the auctioneer. Even though the bidding is over, nobody can reveal the real identity of the winner either auctioneers or other bidders. Owing to the random nonce N_i , the dynamic identity CID_i will be different in each auction, where $CID_i = C_i \oplus h(h(h(x) \| r_i) \| N_i \| AID_j)$. Furthermore, no one can learn of the real identity of $Bidder_i$ even CID_i is public.

4.2.2 Bidding Privacy

To protect the privacy of losers, all tender information of bidders has been encrypted by the session key in our proposed mechanism. Here, we assume that $Bidder_j$ is the winner. The auctioneer will only publish the information of winner about the tender information msg_j, the session key K_j, the highest bidding price bid_j, and the dynamic identity CID_j in the winner announcement phase. If someone intends to get more information of $Bidder_i$, he must have the common session key of $Bidder_i$. However, the session key $K_i = h(h(A_i \| B_i) \| N_i \| N_j \| AID_j)$ will be distinct in each auction with two random nonces. Thus, it is computationally infeasible to construct K_i without N_i, N_j and $h(A_i \| B_i)$ under the assumption of pre-image resistance. Hence, nobody can learn more secret information about losers.

4.2.3 Secret Bidding Price

The bidding price bid_i is protected by the tender information $msg_i = [CID_i,\ bid_i,\ w_i]_{K_i}$ which is encrypted by the common session key K_i under the assumption of the symmetric encryption in our proposed mechanism. Only the correct auctioneer $Auctioneer_j$ can decrypt and learn the bidding price and publish. Here, it is computationally infeasible to obtain the plaintext CID_i, bid_i, and w_i from the encrypted message msg_i without session key K_i .

4.2.4 Unforgeability

The bidding price bid_i is encrypted by the common session key K_i in the tender information $msg_i = [CID_i, bid_i, w_i]_{K_i}$. If someone intends to fabricate a valid bid, he must construct the session key K_i to decrypt the tender information in the first place. However, the session key is only known to $Bidder_i$ and $Auctioneer_j$. Hence, nobody can create a valid biding price of others in our proposed mechanism.

4.2.5 Non-repudiation

If bidders intend to deny their bids at the end of an auction in our proposed mechanism, the auctioneer can prove that the tender information $msg_i = [CID_i, bid_i, w_i]_{K_i}$ is the bid of $Bidder_i$ with the session key K_i . Here, only the correct auctioneer can obtain CID_i, bid_i, and w_i from $Bidder_i$. After that, it carries on computing $w_i^* = h(CID_i \| bid_i \| K_i \| AID_j)$ whether w_i is equal to the computed one w_i^*. If they are the same, it implies that the bidding price is bid by $Bidder_i$.

4.2.6 No Framing

If a legal but malicious bidder $Bidder_k$ wants to impersonate a valid bidder $Bidder_i$ to participate in the auction, he needs to know the session key of $Bidder_i$. However, it is computationally infeasible for $Bidder_k$ to impersonate and submit valid tender information to the auctioneer without knowing the session key under the assumption of the symmetric encryption. Thus, our proposed mechanism can withstand $Bidder_k$ to impersonate a valid bidder.

4.2.7 One Time Registration

In our proposed mechanism, each bidder just registers once with the registration center and then can join multiple auctions. For each play, he randomly generates the nonce N_i to negotiate a common session key with the auctioneer. Later, each bidder can use this common session key to participant in the auction which he is interested in.

4.2.8 Unlinkability

The bidder attends different auctions by the distinct nonce. Due to the random nonce, the bidder can arbitrarily change his dynamic identity and the session key to a new one among plural auctions. Thus, no one can link another auction to know the relationship of bidders.

4.3 Comparisons

Here, we compare the confirmed requirement of our mechanism with other related works in **Table 8**. Most of related works can achieve parts of the requirement. However, [2] and [4] have been pointed out that they cannot achieve bidding privacy and secret bidding price for sealed-bid auctions by Li et al. [12]. Furthermore, [2, 5, 12] have been demonstrated that there exist some security problems, respectively. Also, it has been shown that Chang et al.'s protocol [2] may suffer from the replay attack in the initial phase [3]. Later on, Wu et al. [11] found that Liaw et al.'s protocol [5] could not resist the forge attacks. Recently, Li et al. [12] proposed a practical electronic auction to meet strong anonymity, bidding privacy, and secret bidding prices. Unfortunately, we have found that the denial-of-service attack may be launched in Li et al.'s scheme during the bidding phase in a sealed-bid auction. Consequently, we propose mechanism to satisfy the requirements of the previously mentioned auction.

Table 8. Achieved requirements

Requirements	[2]	[4]	[5]	[6]	[11]	[12]	[13]	Ours
Anonymity	O	O	O	O	O	O	O	O
Bidding Privacy	X	X	-	-	-	O	-	O
Secret Bidding Price	X	X	-	-	-	O	-	O
Security Problem	X	-	X	O	O	X	O	O
Public Verifiability	O	O	-	O	O	O	O	O
Non-repudiation	O	O	O	O	O	O	O	O
No Framing	-	-	-	O	-	-	O	O
Fairness	O	O	-	O	O	-	O	O
One Time Registration	-	-	O	O	-	O	-	O
Unlinkability	-	-	O	O	-	-	O	O

O：It indicates that the essential can achieve.
X：It indicates that the essential cannot achieve.
-：It means that the essential is not mentioned.

In **Table 9**, we list security comparisons of our proposed mechanism with other related works. Here, our proposed mechanism can provide proper mutual authentication, secure session key agreement, and robustness to resist various attacks. The detail of security analysis is shown in Subsection 4.1. In addition, we present a formal proof based on BAN logic model which can provide mutual authentication between bidders and auctioneers to assure the legitimacy of two participants. In 2003, Chang and Chang presented an efficient anonymous auction protocol which can achieve the bidder anonymity. However, Jiang et al. claimed that their protocol would suffer from the replay attack. Later on, Chang and Chang presented an enhanced anonymous auction protocol to decrease the computation cost which Jiang et al. was not taken into consideration. Afterwards, Liaw et al. proposed an electronic auction protocol which was found that their protocol could not resist the impersonation attack by Wu et al. Lately, in order to reduce the computation load of the total auction in [2, 3, 5], Chung et al. presented an English auction protocol with the bulletin board method which can raise the efficiency. Later, Li et al. proposed a practical electronic auction protocol to achieve strong anonymity, bidding privacy, and secret bidding prices which cannot be preserved in [2, 3, 4].

As a result, we propose a multi-auction mechanism which can satisfy all the security and fundamental requirements.

Table 9. Security comparisons

Requirements	[2]	[4]	[5]	[6]	[11]	[12]	[13]	Ours
Session key agreement	-	-	-	-	-	-	-	O
Proper mutual authentication	X	X	X	X	O	X	X	O
Replay attack	X	O	-	-	-	-	-	O
Impersonation attack	O	O	X	-	-	-	-	O
Server spoofing attack	-	-	-	-	-	-	-	O
Resist man-in-the-middle attack	O	O	-	-	-	-	-	O
Resist denial-of-service attack	-	-	-	-	-	X	-	O

5. Performance Discussions

In this section, we discuss the computational cost of the proposed mechanism and make comparisons with related schemes. In our proposed mechanism, we demonstrate the computational overhead of each participant in verification phase, sealed bidding phase, and winner announcement phase. The detail is shown as **Table 10**. In order to reduce the burden of computational overhead, we have applied the one-way hash function, exclusive-or function, and symmetric key function, instead if public cryptosystem. Exclusive-or operations should be ignored because the computational load of exclusive-or operation is very low. All of these functions can help our proposed mechanism reduce the computational cost and be more efficient than other related schemes.

Table 10. Computational cost of proposed mechanism

Phase	Bidder	Auctioneer
Verification Phase	$5T_h + 2T_{Xor}$	$6T_h + 7T_{Xor}$
Sealed Bidding Phase	$1T_{Sym} + 1T_h$	0
Winner Announcement Phase	0	$1T_{Sym} + 1T_h$
T_h : Hash cost, T_{Xor} : Exclusive-or cost, T_{Sym} : Symmetric key encryption cost		

In order to analyze the computational complexity with related schemes, we define the notation and present the details in **Table 10**. Since almost all related schemes include the bidding phase, thus, we consider the computational overhead of this phase as the principal part in a sealed-bid auction. Based on [30, 31], we can learn that $1T_{Asym} \approx 100T_{Sym}$; $1T_{Sym} \approx 5/3T_{Exp}$; $1T_{Exp} \approx 600T_h$ for software consideration. For brevity, we just discuss software consideration in our paper and find that one-way function is more efficient than symmetric key encryption. From **Table 11**, the computational load of our proposed mechanism is $1T_{Sym} + 1T_h + 9T_{Xor}$ in sealed bidding phase. In this phase, we only use one symmetric key encryption, one hash function, and nine exclusive-or operations. Here, the pervious schemes use one public key encryption, or one signature, or three exponentiation operations at least. Obviously, the new scheme requires less computational overhead than the previous auction schemes. Thus, our proposed mechanism is more efficient and applicable for the mobile device.

Table 11. Comparisons of computational cost

	Sealed Bidding Phase
[2]	$3T_{PKE} + 3T_h$
[4]	$1T_{PKE} + 2T_{Sym}$
[5]	$5T_{Exp}$
[6]	$3T_{Exp} + 6T_{Mul} + 1T_h$
[11]	$4T_{Exp} + 1T_{Mul}$
[12]	$1T_{Sig} + 1T_{Sym} + 1T_h$
[13]	$1T_{PKE} + 1T_{Sym} + 4T_h$
Ours	$1T_{Sym} + 1T_h + 9T_{Xor}$

T_{PKE} : Public key encryption cost
T_{Exp} : Modular exponentiation cost
T_{Sig} : Signature cost
T_{Xor} : Exclusive-or cost
T_{Sym} : Symmetric key encryption cost
T_{Mul} : Modular multiplication cost
T_h : Hash cost

6. Conclusions

In this paper, we have pointed out that Li et al.'s practical electronic auction scheme is vulnerable to the denial-of-service attack. Aside from essential of electronic auction, we have developed a brand-new version with dynamic identity property to prevent from malicious tracing. The correctness of mutual authentication between bidder and auctioneer has been proved according to the BAN logic model. Moreover, the efficiency of the proposed mechanism has been demonstrated to be superior to related works. Specifically, a bidder only needs to register at the center once then he can join multiple plays of different auctioneers, which is defined as multi-auction.

References

[1] D. Hirakiuchi and K. Sakurait, "English vs. Sealed Bid in Anonymous Electronic Auction Protocols," in *Proc. of IEEE International Conference on Enabling Technologies: Infrastructure for Collaborative Enterprises*, pp. 171-176, 2001. Article (CrossRef Link)

[2] C.C. Chang and Y.F. Chang, "Efficient Anonymous Auction Protocols with Freewheeling Bids," *Computers & Security*, Vol. 22, No. 8, pp. 728-734, 2003. Article (CrossRef Link)

[3] R. Jiang, L. Pan and J. H. Li, "An Improvement on Efficient Anonymous Auction Protocols," *Computers & Security*, Vol. 24, No. 2, pp. 169-174, 2005. Article (CrossRef Link)

[4] Y.F. Chang and C.C. Chang, "Enhanced Anonymous Auction Protocols with Freewheeling Bids," in *Proc. of 20th International Conference on Advanced Information Networking and Application*, Vol. 1, pp. 353-358, 2006. Article (CrossRef Link)

[5] H.T. Liaw, W.S. Juang and C.K. Lin, "An Electronic Online Bidding Auction Protocol with both Security and Efficiency," *Applied Mathematics and Computation*, Vol. 174, No. 2, pp. 1487-1497, 2006. Article (CrossRef Link)

[6] Y.F. Chung, K.H. Huang, H.H. Lee, F.P. Lai and T.S. Chen, "Bidder-anonymous English Auction Scheme with Privacy and Public Verifiability," *Journal of Systems and Software*, Vol. 81, No. 1, pp. 113-119, 2008. Article (CrossRef Link)

[7] Y.F. Chung, Y.T. Chen, T.L. Chen and T.S. Chen, "An Agent-based English Auction Protocol using Elliptic Curve Cryptosystem for Mobile Commerce," *Expert Systems with Applications*, Vol. 38, No. 8, pp. 9900-9907, 2011. Article (CrossRef Link)

[8] H. Xiong, Z. Chen and F. Li, "Bidder-anonymous English Auction Protocol based on Revocable Ring Signature," *Expert Systems with Applications*, Vol. 39, No. 8, pp. 7062-7066, 2012. Article (CrossRef Link)

[9] J.Heezen and W. Beats, "The Impact of Electronic Markets: The Case of the Dutch Flower Auction," *Journal of Strategic Information System*, Vol. 5, No. 4, pp. 317-333, 1996. Article (CrossRef Link)

[10] W. Standaert, S. Muylle and I. Amelinckx, "An Empirical Study of Electronic Reverse Auction Project Outcomes," *Electronic Commerce Research and Applications*, Vol.14, No. 2, pp. 81-94, 2015. Article (CrossRef Link)

[11] C.C. Wu, C.C. Chang and I.C. Lin, "New Sealed-bid Electronic Auction with Fairness, Security and Efficiency," *Journal of Computer Science and Technology*, Vol. 23, No. 2, pp. 253-264, 2008. Article (CrossRef Link)

[12] M.J. Li, J. S.T. Juan and J. H.C. Tsai, "Practical Electronic Auction Scheme with Strong Anonymity and Bidding Privacy," *Information Sciences*, Vol. 181, No. 12, pp. 2576-2586, 2011. Article (CrossRef Link)

[13] W. Shi, "An Efficient Sealed-bid Auction Protocol with Bid Privacy and Bidder Privacy," *International Journal of Innovative Computing, Information and Control*, Vol. 8, No. 11, pp. 7943-7953, 2012. Article (CrossRef Link)

[14] W.S. Juang, H.T. Liaw, P.C. Lin and C.K. Lin, "The Design of a Secure and Fair Sealed-bid Auction Service," *Mathematical and Computer Modelling*, Vol. 41, No. 8-9, pp. 973-985, 2005. Article (CrossRef Link)

[15] K. Miyashita, "Online Double Auction Mechanism for Perishable Goods," *Electronic Commerce Research and Applications*, Vol.13, No. 5, pp. 355-367, 2015. Article (CrossRef Link)

[16] L. I. de Castro and D. H. Karney, "Equilibria Existence and Characterization in Auctions: Achievements and Open Questions," *Journal of Economic Surveys*, Vol. 26, No. 5, pp. 911-932, 2012. Article (CrossRef Link)

[17] C.C. Lina, S.C. Chenb and Y.M. Chu, "Automatic Price Negotiation on The Web: An Agent-based Web Application using Fuzzy Expert System," *Expert Systems with Applications*, Vol. 38, No. 5, pp. 5090-5100, 2011. Article (CrossRef Link)

[18] A. H. Ozer and C. Ozturan, "Multi-unit Differential Auction-barter Model for Electronic Marketplaces," *Electronic Commerce Research and Applications*, Vol. 10, pp. 132-143, 2011. Article (CrossRef Link)

[19] J.S. Chang and W.H. Chang, "Analysis of Fraudulent Behavior Strategies in Online Auctions for Detecting Latent Fraudsters," *Electronic Commerce Research and Applications*, Vol.13, No. 2, pp. 79-97, 2015. Article (CrossRef Link)

[20] F.S. Hsieh and C.S. Liao, "Schemes to Reward Winners in Combinational Double Auctions based on Optimization of Surplus," *Electronic Commerce Research and Applications*, Vol.14, No. 6, pp. 405-417, 2015. Article (CrossRef Link)

[21] C. Dang, Q. Hu and J. Liu, "Bidding Strategies in Online Auctions with Different Ending Rules and Value," *Electronic Commerce Research and Applications*, Vol.14, No. 2, pp. 104-111, 2015. Article (CrossRef Link)

[22] X. Li, J. Mab, W. Wang, Y. Xiong and J. Zhang, "A Novel Smart Card and Dynamic ID based Remote User Authentication Scheme for Multi-server Environments," *Mathematical and Computer Modelling*, Vol. 58, No. 1-2, pp. 85-95, 2012. Article (CrossRef Link)

[23] D. Chaum and H. Antwerpen, "Undeniable Signatures," *Advances in Cryptology. CRYPTO'89*, Vol. 435, pp. 212-216, 1990. Article (CrossRef Link)

[24] C.P. Schnorr, "Efficient Signature Generation for Smart Cards," *Journal of Cryptology*, Vol. 4, No. 3, pp. 239-252, 1991. Article (CrossRef Link)

[25] J. Daemen and V. Rijmen, "The Design of Rijndael: AES - The Advanced Encryption Standard," *Springer*, 2002. Article (CrossRef Link)

[26] A. Menezes, P. V. Oorschot and S. Vanstone, "Handbook of Applied Cryptography," *CRC Press*, USA, pp. 321-376, 1996. Article (CrossRef Link)

[27] M. Burrows, M. Abadi and R. Needham, "Authentication: A Practical Study in Belief and Action," in *Proc. of 2nd Conference on Theoretical Aspects of Reasoning about Knowledge*, CA, USA, pp. 325-342, 1988. Article (CrossRef Link)

[28] M. Burrows, M. Abadi and R. Needham, "A Logic of Authentication," *ACM Transactions on Computer Systems*, Vol. 8, No. 1, pp. 18-36, 1990. Article (CrossRef Link)

[29] S.P. Yang and X. Li, "Defect in Protocol Analysis with BAN Logic on Man-in-the-middle Attacks," *Application Research of Computers*, Vol. 24, pp. 149-151, 2007. Article (CrossRef Link)

[30] B. Schneier, "Applied Cryptography, Protocols Algorithms, and Source Code in C," *John Wiley and Sons Inc.*, New York, U.S.A., 1994. Article (CrossRef Link)

Jung-San Lee received the BS degree in computer science and information engineering in 2002 and his Ph.D in computer science and information engineering in 2008, both from National Chung Cheng University, Chiayi, Taiwan. Since 2012, he has worked as an associate professor in the Department of Information Engineering and Computer Science at Feng Chia University, Taichung, Taiwan. His current research interests include information security, image processing, and watermarking.

Kuo-Jui Wei received the MS degree in information engineering and computer science in 2011. He is currently pursuing his Ph.D. degree in Information Engineering and Computer Science in Feng Chia University, Taichung, Taiwan. His current research interests include information security and mobile communications.

Ying-Chin Chen is currently pursuing her MS degree in information engineering and computer science in Feng Chia University, Taichung, Taiwan. Her current research interests include information security and e-commerce.

Yun-Hsiang Sun is currently pursuing his MS degree in information engineering and computer science in Feng Chia University, Taichung, Taiwan. His current research interests include information security and e-commerce.