Adding timestamps to the secure electronic auction protocol

Min-Shiang Hwang a,*, Eric Jui-Lin Lu a, Iuon-Chang Lin b

a Department of Information Management, Chaoyang University of Technology, 168, Gifeng E. Road, Wufeng, Taichung County, Taiwan, ROC, 413

b Department of Computer Science and Information Engineering, National Chung Cheng University, Chaiyi, Taiwan, ROC

Received 18 May 2001; received in revised form 21 June 2001; accepted 21 June 2001

Abstract

Recently, Subramanian proposed a secure electronic auction protocol. It was claimed that the auction protocol ensured anonymity, security, privacy, atomicity, and low overhead. However, the sensitive information of payment, such as credit card number, can be revealed by the auctioneer. The bid can be forged by a malicious auctioneer. Also, the situation of two or more bidders who offered the same price was not resolved efficiently. In this paper, we will improve the robustness of this electronic auction protocol. © 2002 Elsevier Science B.V. All rights reserved.

Keywords: Auction; Cryptography; Security

1. Introduction

Electronic auction systems can be generally classified into two categories. One is called the public bid auction, in which the bidders make public bids on the product; bids must higher than all bids in the previous round of bidding. When nobody submits a new bid in a given round, the product is sold to the highest bidder from the previous round. In this type, a bidder may submit many bids for each auction, so it is also called multi-bidding auction. The other type is the sealed bid auction. All bids are sealed. After the deadline of the bidding, the auctioneer can open the bid and decide the winning bid. Because each bidder can only submit one bid for each auction, we also call it single-bidding auction.

In the existing papers of electronic auction schemes [1,4–10], only Subramanian proposed a multi-bidding auction scheme [10]. It used the technique of public key cryptosystem to establish a secure transaction channel. This protocol involves an auctioneer, a set of third parties and a set of bidders. The auction house has a product and wants to sell for the highest price offered. The bidder makes a bid on the product before the end of a bidding period. At the end of a round, the auctioneer collects all the bids and publishes the highest bid. In the next round of bidding, bidder
must hike the bid, which must be higher than the highest bid in the previous round. If no higher
bid is offered, the auctioneer publishes the highest price and sells the product.

After the auction, the bidder who offered the highest bid is notified and the bidder has to make
a payment (e.g. using credit card or electronic cash [3]) to the auctioneer. It is important that the
auctioneer cannot duplicate the received payment information.

By using public key cryptography and formal verification, Subramanian showed that the
auction protocol ensured the following properties:
1. **Anonymity**: nobody (including the auctioneer) can know the true identity of the bidder.
2. **Security**: the integrity and confidentiality of the transaction messages must be protected.
3. **Privacy**: a third party cannot know details about the transaction such as payment amount,
   product details.
4. **Atomicity**: the transaction is either completed or aborted.
5. **Low overhead cost**: the transaction cost must be low enough to suit transactions involving a
   small of money.

In this paper, we illustrate that Subramanian’s auction protocol has three weaknesses: (1) the
bid can be forged by a malicious auctioneer, (2) sensitive information about payment may be
revealed by the auctioneer, and (3) Subramanian’s protocol does not resolve ties between bidders
efficiently. By adding timestamps [2], we will remedy these weaknesses and improve the robustness
of Subramanian’s electronic auction protocol.

2. Description of Subramanian’s electronic auction protocol

In this section, we review Subramanian’s electronic auction protocol [10]. Initially, the protocol
notations are defined as follows:
- \( U \) stands for the auction house, and the auction house’s public and private keys are \( u \) and \( 1/u \),
  respectively.
- \( B \) stands for the bidder, and \( B’ \)’s public and private keys are \( b \) and \( 1/b \), respectively.
- \( T \) stands for the third party, and the third party’s public and private keys are \( t \) and \( 1/t \), respectively.
- \( E \) and \( e \) are two encryption keys. \( 1/E \) and \( 1/e \) are two corresponding decryption keys.
- \( X \rightarrow Y : [\text{message}]^{1/x} \) means that \( X \) signs the message with \( X \)’s private key, and sends it to \( Y \).
  Therefore, \( Y \) can verify that the message is from \( X \).
- \( X \rightarrow Y : [\text{message}]^{y} \) means that \( X \) encrypts the message with \( Y \)’s public key, and sends it to \( Y \).
  Therefore, only \( Y \) can decrypt the message with his private key \( 1/y \).

Subramanian’s electronic auction protocol can be divided into three stages: (1) the advertisement
stage, (2) the bidding stage, and (3) the exchange of the product and the payment stage. The
processes of each stage are described below:

2.1. The advertisement stage

1. \( U \rightarrow \text{everybody} : [\text{product description, list of recognized third parties}]^{1/u} (M_i) \).

   For advertising, \( U \) first broadcasts the message \( M_i \) which includes product description and a list of
   third parties that \( U \) trusts. The third parties issue pseudo-identities (i.e. public keys) to each bidder.
2. \( U \rightarrow \text{everybody} : [\text{product identifier, product}]^{1/u} \).

   \( U \) also broadcasts the product identifier and the ciphertext of the product which is encrypted
   with key \( E \); the product identifier is a part of the product description, which is used to prevent
mix-ups with regard to sales of other products. Both the messages are signed with $U$’s private key $1/u$ so that the interested bidders can verify that these messages are from $U$.

2.2. The bidding stage

1. $B \rightarrow U : [M_1, b, \text{third party } T, \text{payment}^e, [\text{price}]^{1/b}]^u$.

   $B$ bids on the product by responding to $U$’s advertisement. The information of the bid includes the message $M_1$, $B$’s public key, third party $T$ who issued $B$’s public key, the encrypted payment with key $e$, and the signed price offered by $B$. $U$ can check the validity of $B$’s public key by looking up $T$’s directory. Then $U$ can use the public key $b$ to confirm that the price is offered by $B$.

2. $U \rightarrow B : [[\text{payment}^e]^{1/u}, [\text{price}]^{1/b^u}]^b$.

   After receiving the bid, $U$ acknowledges the receipt of the bid. The acknowledgement includes the encryption of the payment signed by $U$’s private key $1/u$, and the price signed by $B$’s private key $1/b$. And it is secured with $B$’s public key $b$.

3. $U \rightarrow \text{everybody} : [\text{product description, maximum price offered}]^{1/u}$.

   When a round of bidding is cut off, $U$ collects all the bids in this round and publishes the maximum bid.

   This stage is repeated until no bidder wishes to make a higher bid. The protocol assumes that $U$ can hike the maximum price expecting other bidders to offer more. Furthermore, this method may be used to resolve ties between bidders.

2.3. The exchange of the product and the payment stage

1. $U \rightarrow \text{everybody} : [\text{product description, sold for price}]^{1/u} (M_2)$.

   If no higher bid is offered, $U$ publishes the highest bid in this auction and the product is sold for the highest price quoted. The message $M_2$ is signed by $U$ so that bidders can verify the message from $U$.

2. $B \rightarrow U : [M_2, [1/e]^{1/b}]^u$.

   $B$ responds with the signature of decryption key $1/e$ for his payment. If $B$ does not respond, $U$ approaches $T$ with $B$’s bid and collects the payment.

3. $U \rightarrow B : [[1/e, 1/E]^{1/u}]^b$.

   Finally, $U$ acknowledges the payment and sends the decryption key $1/E$ of the product. The bidding for and exchange of the product and the payment processes are shown in Fig 1.

   In this auction protocol, once a bidder makes the highest bid for an auction, he cannot refuse to buy the product. Third parties act as collection agencies to ensure this feature. If the winner does not respond with the decryption key $1/e$ or delivers the wrong payment information, the third party can look up the winner’s true identity according to his pseudo-identity (i.e. public key). This conflicts with anonymity. Therefore, bidders must trust these third parties not to reveal the bidder’s identity or other information arbitrarily.

3. The weaknesses of Subramanian’s electronic auction protocol

Subramanian’s auction protocol has the following weaknesses:

1. The bid can be forged by a malicious auctioneer. On receiving a bid from bidder $B$, a malicious auctioneer can fake a bid by bidder $B$ in other auctions. In step (1) of the bidding stage, the
Bidder B 

Bank K 

Bidder B Auctioneer 

Withdrawal 


Fig. 1. The processes of bidding for and exchange of the product and the payment in Subramanian’s auction protocol.

auctioneer can obtain the information of \( b, third\ party\ T, payment^t \), and the signature of \( \text{price}^{1/b} \). Therefore, the malicious auctioneer can make a forged bid \([M_1', b, third\ party\ T, payment^t, \text{price}^{1/b}]^{uid}\) for other auctions, where \( U' \) may stand for the same or a different auctioneer. The auctioneer \( U' \) will convince that the bid is made by \( B \) by verifying the signature \( \text{price}^{1/b} \) with \( B ' \)s public key. If \( B \) is the winner of the auction, he must make the payment. Even if the malicious auctioneer does not know the decryption key \( [1/e] \) (i.e. no \( B \) responds with it), \( U' \) can collect the payment through the third party.

2. Sensitive information of payment can be revealed. Subramanian’s auction protocol assumes the existence of some acceptable electronic form of payment such as credit card number, electronic cash, or any other electronic entity that does not compromise the bidder’s privacy or anonymity. However, the information of the payment (e.g. credit card number) can be revealed once the auctioneer receives the decryption key from the winner. Furthermore, the malicious auctioneer may reveal the payment information (\( \text{payment}^t \) and \( 1/e \)) of bidder \( B_1 \) to another bidder \( B_2 \). Then, \( B_2 \) can make a bid using the revealed payment information such as \([M_1, b, third\ party\ T, \text{payment}^t, \text{price}^{1/b}]^{uid}\) in another auction. If the bid is the highest bid of the auction, \( B_2 \) submits the decryption key \( 1/e \) to the auctioneer. Hence, the bank will collect the payment from \( B_1 \), not from \( B_2 \).

3. Ties between bidders are not resolved efficiently. The auction protocol may be stuck in a situation where the highest bid for a product is offered by more than one bidder. The resolution of this conflict is using some agents to hike the maximum bid artificially. However, if no bidder makes a higher bid than the agent’s, what shall the auctioneer do?

4. The revised protocol

In this section, we modify Subramanian’s auction protocol to improve its robustness. The revised protocol is similar to Subramanian’s auction protocol, except for the following modifications.
In step (1) of the bidding stage, we add timestamp $TS$ to the electronic form of payment and price so that the step becomes

$$B \rightarrow U : [M_1, b, \text{third party } T, \text{bank } K, TS, [[TS \oplus \text{payment}^e]^k, [TS \oplus \text{price}^{1/b}]^u],$$

where $TS$ is a timestamp, $k$ is the public key of bank $K$, and $\oplus$ denotes the operation of exclusive OR. With the payment encrypted with $k$, the auctioneer will not be able to know the payment information. With the form of $[TS \oplus \text{payment}]$, the auctioneer cannot replay the payment, once $TS$ has expired. The auctioneer is still able to validate the payment by sending this message to the bank.

When $U$ receives the bid at the time $TS'$, he checks the correctness of the timestamp first. Let $\Delta T$ denote the expected legal time interval for transmission delay between the bidder’s terminal and the auctioneer’s server. If the time interval between $TS$ and $TS'$ is greater than $\Delta T$ ($TS' - TS > \Delta T$), $U$ shall reject the bid.

Then, $U$ acknowledges the receipt of the bid in step (2) of this stage and the delivered message now becomes

$$U \rightarrow B : [[[TS \oplus \text{payment}^e]^k]^{1/u}, [TS \oplus \text{price}^{1/b}]^{1/b}]^b.$$

When the time allotted for a round of bidding has expired, $U$ bulletins the maximum bid. In step (3) of the bidding stage, $TS$ has to be published along with the publication; thus, the step becomes:

$$U \rightarrow \text{everybody} : [\text{product description, TS, maximum price offered}]^{1/u}.$$

When no bidder makes a higher bid for the product, $U$ notifies each bidder about the sale price. Adding timestamp, the step becomes:

$$U \rightarrow \text{everybody} : [\text{product description, TS, sold for price}]^{1/u}.$$

When the highest bid for a product is offered by more than one bidder, the bid with lower timestamp will win the bid. Now, the auctioneer can determine who is the winner based on the timestamp. And this case resolves the conflicts that generally happen in real-world auction sites.

After $U$ publishes the price of sale, the winner responds with the deciphering key for his payment. However, it is now different from Subramanian’s auction protocol in that $U$ can no longer decrypt the message of this payment and has to forward the encrypted payment information, the deciphering key $1/e$, and the $TS$ to the bank $K$ for reimbursement. The bank can obtain the payment information by

$$[[[TS \oplus \text{payment}^e]^k]^{1/k} = [TS \oplus \text{payment}]^e, \quad (1)$$

$$[[TS \oplus \text{payment}^e]^{1/e}] = [TS \oplus \text{payment}], \quad (2)$$

$$[TS \oplus \text{payment}] \oplus TS = \text{payment.} \quad (3)$$

The processes of bidding for and exchange of the product and the payment in our proposed protocol are shown in Fig 2.
4.1. Security analysis

In our revised protocol, we resolve the three weaknesses of Subramanian’s auction protocol. Firstly, we add timestamp to the price and sign it with the bidder’s secret key. The auctioneer can compute the price using the bidder’s public key and timestamp $TS$ from the following equation:

$$\text{price} = [\left[ TS \oplus \text{price} \right]^{1/b}]^b \oplus TS. \quad (4)$$

Only the bidder knows the secret key, hence the auctioneer cannot forge the signature of $[TS \oplus \text{price}]$. If the auctioneer replays the message to another auction, the verification of the timestamp will fail. Therefore, weakness (1) is eliminated.

Secondly, we add timestamp to the payment and encrypt the message $[\left[ TS \oplus \text{payment} \right]^c]$ with the bank’s public key. Without the bank’s secret key, the auctioneer cannot decrypt the message. Therefore, sensitive information can be protected. Furthermore, the payment information is exclusive OR with $TS$, hence the scenario described in weakness (2) does not happen. The auctioneer will check the timestamp first.

Thirdly, when two or more bidders offer the same bid and the bid is the highest of this round, comparing their timestamps, the lowest timestamp will be the winner. However, if a bidder wants to choose the lower timestamp, the bid may not pass the verification $TS' - TS \leq \Delta T$.

5. Conclusions

A secure electronic auction protocol is very important in modern electronic markets. It advances the commerce of auction full of vitality. In this paper, we propose an extension of Subramanian’s auction protocol. The main advantages of our revised protocol are:

- a malicious auctioneer cannot forge a bid or replay the payment,
- it prevents the payment information from leaking out, and
- it improves the robustness of Subramanian’s protocol by increasing privacy and conflict resolution.
Acknowledgements

The authors wish to thank many anonymous referees for their suggestions to improve this paper. This research was partially supported by the National Science Council, Taiwan, ROC, under contract no.: NSC89-2213-E-324-053.

References


Min-Shiang Hwang received his B.S. in Electronic Engineering from National Taipei Institute of Technology, Taipei, Taiwan, ROC, in 1980; his M.S. in Industrial Engineering from National Tsing Hua University, Taiwan, in 1988; and his Ph.D. in Computer and Information Science from National Chiao Tung University, Taiwan, in 1995. He also studied Applied Mathematics at National Cheng Kung University, Taiwan, from 1984 to 1986. Dr. Hwang passed the National Higher Examination in field “Electronic Engineering” in 1988. He also passed the National Telecommunication Special Examination in field “Information Engineering”, qualified as advanced technician, first class, in 1990. From 1988 to 1991, he was the leader of the Computer Center at Telecommunication Laboratories (TL), Ministry of Transportation and Communications, ROC. He was also a project leader for research in computer security at TL in July 1990. He obtained the 1997, 1998, and 1999 Distinguished Research Awards of the National Science Council of the Republic of China. He is currently a professor and chairman of the Department of Information Management, Chaoyang University of Technology, Taiwan, ROC. He is a member of IEEE, ACM, and the Chinese Information Security Association. His current research interests include database and data security, cryptography, image compression, and mobile communications.

Eric Jui-Lin Lu received his B.S. degree in Transportation Engineering and Management from National Chiao Tung University, Taiwan, ROC, in 1982; M.S. degree in Computer Information Systems from San Francisco State University, CA, in 1990; and Ph.D. degree in Computer Science from University of Missouri – Rolla, MO, in 1996. He is currently an associate professor of the Department of Information Management and Director of the Computer Center, Chaoyang University of Technology, Taiwan, ROC. His current research interests include electronic commerce, distributed processing, and security.
Lin received his B.S. in Computer and Information Sciences from Tung Hai University, Taichung, Taiwan, ROC, in 1998; and M.S. in Information Management from Chaoyang University of Technology, Taiwan, in 2000. He is currently pursuing his Ph.D. degree in Computer Science and Information Engineering from National Chung Cheng University. His current research interests include electronic commerce, information security, cryptography, and mobile communications.