An Electronic Public Engineering Project Bidding Protocol via a Subliminal Channel

Chin-Ling Chen
Department of Computer Science and Information Engineering, Chaoyang University of Technology, Taichung 41349, Taiwan, (R.O.C.), e-mail: clc@mail.cyut.edu.tw
School of Information Engineering, Changchun Sci-Tech University, 1699 Donghua Street, Shuangyang District, Changchun City, Jilin Province, 130600, China, e-mail: wkhyoyo@163.com

Kun-Hao Wang
School of Information Engineering, Changchun Sci-Tech University, 1699 Donghua Street, Shuangyang District, Changchun City, Jilin Province, 130600, China, e-mail: wkhyoyo@163.com

Woei-Jiunn Tsaur
Computer Center, National Taipei University, Taiwan, R.O.C., e-mail: wjtsaur@mail.ntpu.edu.tw

Jung-Hsuan Chen
Department of Industrial Education, National Taiwan Normal University, Taipei, 10610 Taiwan, R.O.C., e-mail: jhchen@ntnu.edu.tw

Chien-Hung Chen
Department of Computer Science and Information Engineering, Chaoyang University of Technology, Taichung 41349, Taiwan, (R.O.C.), e-mail: bitz823@hotmail.com

Corresponding author: clc@mail.cyut.edu.tw
Due to the rapid development of the Internet, many Internet applications have recently become very widely used. Internet security has therefore become an important issue. This paper proposes an electronic public engineering project bidding protocol via a subliminal channel. In the proposed scheme, the subliminal channel can protect a bidder's interests, while allowing an official agent to make a fair arbitration. The proposed scheme is non-repudiable, untraceable and offers fair arbitration of public engineering projects, but is also resistant to replay, forgery and insider attacks, thus enhancing both security and fairness.

**KEYWORDS:** Subliminal channel, Fair arbitration, Security, Non-repudiation, Digital signature.

### 1. Introduction

In recent years, the Internet has developed rapidly, and many Internet transaction applications have become popular. For example, many public construction engineering projects now conform to public, fair and efficient requirements via Internet technology. For example, in e-bidding cases, some important issues should be of concern. The bidding price should be protected to defend against insider attack, and there should be a fair arbitration mechanism to arbitrate accusations.

In 1983, Simmons [15-17] first proposed the subliminal channel mechanism. Subliminal channels are employed for secret communication; they can be used to deliver subliminal messages between sender and receiver. Subliminal messages cannot be accessed except through the specific receiver. In 1997, Harn and Gong [5] proposed a digital signature using a subliminal channel; they showed how to construct a digital signature scheme with a broadband subliminal channel that does not require a subliminal receiver to share the transmitter's secret signing key. In 2010, Lin et al. [9] proposed a digital signature with multiple subliminal channels, and its applications. The proposed scheme has the advantage that the subliminal receivers cannot forge a valid signature since they do not share the signer's secret key. It can also provide more than one independent subliminal message.

In 1998, Subramanian [18] presented the design and verification of a secure electronic auction protocol. This protocol ensured anonymity, security, privacy, atomicity and low overhead cost, but also adds the properties of non-repudiation, untraceability, one-time registration and unlinkability. In 2008, Chun et al. [3] proposed a bidder-anonymous English auction scheme with privacy and public verifiability. The proposed scheme provided the following security features: anonymity, traceability, no framing, unforgeability, non-repudiation, fairness, public verifiability, unlinkability among various auction rounds, linkability within a single auction round, bidding efficiency, one-time registration, and easy revocation. In 2012, Xiong et al. [19] proposed a bidder-anonymous English auction protocol based on revocable ring signature. The proposed protocol has three appealing characteristics: first, it offers conditional privacy-preservation: while the auctioneer can verify that a bidder is an authorized participant in the system, only the collaboration of auctioneer and registration manager can reveal the true identity of a malicious bidder. Second, it is a one-time registration: the bidder can take part in plural auctions with one time registration. Third, it is spontaneous: the bidder can bid without interaction with the auctioneer and other bidders. Fan et al. [4] proposed a multi-recastable e-bidding game with dual-blindness. The proposed protocol allows all participants to take part in a sequence of different auctions for various products unlimitedly after performing a one-time registration. Where the winner does not need to re-register either. They formally proved the security of the proposed scheme and also provided comparisons to show that it was the most efficient one, compared with previous works. In order to defend against known attacks, some applications should embed authentication mechanisms [10-11] to ensure that security requirements can also be guaranteed.
This paper proposes a novel scheme for a fair bidding transaction for public construction engineering projects. The fair bidding transactions need to protect the bidder’s privacy and uphold a fair transaction process. The proposed scheme is not only able to protect the bidder’s privacy, but can also support fair arbitration via a subliminal channel.

The proposed scheme should have the following characteristics:

1. Non-repudiation [2]: Non-repudiation refers to the ability to ensure that parties cannot deny the authenticity of their signature on a message which they have sent.
2. Fair arbitration [1]: The fair arbitration mechanism can allow participating parties access to fair arbitration when they have doubts regarding aspects of the project.
3. Blind message [14]: The blind message mechanism can be provided to protect the bidder’s privacy, and other persons cannot extract the message, aside from the owner of the blind factor.
4. Unlinkability [4]: No one can trace a specific bidder from the transaction message.
5. One-time registration [4]: The protocol only requires one-time registration for each bidder.
6. Auditability [4]: Auditability is a third-party application which can assist an auctioneer to find the real identity of a bidder in certain cases, or if disputes occur.
7. Defence against replay attack [1]: A replay attack occurs when an attacker copies the message between two parties and replays it to one or more parties in order to gain access to sensitive information.
8. Defence against forgery attack [14]: In a forgery attack, an attacker masquerading as a legal party transmits the message to obtain the other party’s trust, and thereby gains access to sensitive information.
9. Defence against insider attack [14]: The attacker can access sensitive information using a legal identity to achieve an insider attack.

The remainder of this paper is arranged as follows. Section 2 introduces the framework of the proposed scheme. Section 3 offers a security analysis of the proposed scheme. Section 4 discusses the computation costs of the proposed scheme, and makes a security comparison with related works. Finally, conclusions regarding the proposed scheme are drawn in Section 5.

2. The Proposed Scheme

The overview of the proposed scheme is shown in Figure 1. There are five parties involved in the scheme:

1. Bidder (BI): A person or a company that wants to take part in a bidding case
2. Public Construction Commission (PCC): The organization which has a private enterprise or government entity hold a bidding operation
4. Proxy Server (PS): A trusted proxy server used to transfer and store important information
5. Official Agent (OA): A trusted and fair arbitrator

**Step 1**: BI → BK, PS & PCC → BK, PS: BI and PCC register with the BK and PS, respectively.

**Step 2**: BI → BK: When a bidder wants to take part in the bid, the bidder sends the bidding bond and related information to the bank.

**Step 3**: BK → BI: After receiving the message, BK stores the bidding bond, signs the response message and sends it to the BI.

**Step 4**: BI → PCC: The BI makes the bidding message, which includes the blind factor and the bidder’s identity. After that, the BI sends the bidding message to the bank.

**Step 5**: PCC → BI: After receiving the blind factor and bidding identity to the PCC.

**Step 6**: BI → PS: The BI makes the bidding message, which includes the blind factor and the bidder’s identity. After that, the PI sends the bidding message to the PS.

**Step 7**: PS → BI: The PS sends the response message to the BI.

**Step 8**: PS → PCC: The PS sends the blind factor and bidding identity to the PCC.

**Step 9**: PCC → PS: the PCC sends the response message to the PS.

**Step 10**: PCC → BI: The PCC publishes the winner.

**Step 11**: BI → OA or PCC → OA: When the BI or PCC doubts this bidding case, they can send the blind message, blind factor and subliminal message to the OA to request arbitration.
The following notations are used in the proposed protocol:

\( X_{BI}, X_{BK}, X_{PCC}, X_{PS}, X_{OA} \) : the BI, BK, PCC, PS and OA’s private keys, respectively

\( Y_{BI}, Y_{BK}, Y_{PCC}, Y_{PS}, Y_{OA} \) : the BI, BK, PCC, PS and OA’s public keys, respectively, for example: \( y_i = g^{\omega_i} \mod p \), where \( g \) is a randomly chosen generator of the multiplicative group \( \mathbb{Z}_p \)

\((d_i, s_i)\) : the \( i \)th signature pair

\( T_X - Y \) : the timestamp generated by X, and transferred to Y

\( M_x \) : the subliminal message

\( M_{x-Y} \) : the message is transferred from X to Y

\( C_i \) : the \( i \)th cipher message

\( h() \) : a one-way hash function

\( A_{2B} \) : determine if A is equal to B.

### 2.1. The Registration Phase

In the proposed scheme, the encryption and decryption mechanism is used to protect messages, based on the ElGamal scheme. The BI and PCC need to register with the bank (BK) and proxy server (PS). Then, the bank and proxy server store the BI and PCC’s identities: \( ID_{BI} \) and \( ID_{PCC} \), respectively. After this, the participating parties receive their private and public keys.
Therefore, the participating parties can select their own private keys: \( x_{\text{BP}}, x_{\text{BK}}, x_{\text{PCC}}, x_{\text{PS}} \) and \( x_{\text{OA}} \). It can then compute the responding public key.

**Step 1:** The BI, BK, PCC, PS and OA choose their private keys \( x_{\text{BP}}, x_{\text{BK}}, x_{\text{PCC}}, x_{\text{PS}} \) and \( x_{\text{OA}} \), respectively. Then these parties compute public keys \( y_{\text{BP}}, y_{\text{BK}}, y_{\text{PCC}}, y_{\text{PS}} \) and \( y_{\text{OA}} \), respectively, as follows:

\[
y_{\text{BI}} = g^{x_{\text{BI}}} \mod p \tag{1}
\]

\[
y_{\text{BK}} = g^{x_{\text{BK}}} \mod p \tag{2}
\]

\[
y_{\text{PCC}} = g^{x_{\text{PCC}}} \mod p \tag{3}
\]

\[
y_{\text{PS}} = g^{x_{\text{PS}}} \mod p \tag{4}
\]

\[
y_{\text{OA}} = g^{x_{\text{OA}}} \mod p. \tag{5}
\]

### 2.2. The Bidding Phase

The BI sends the bidding bonds to the bank (BK), and then the BK returns the response message to the BI. After this, the BI sends the blind message to the PCC to sign it, and the PCC returns the blind message. The BI then sends the bidding message to the PS. The bidding phase is shown in Figure 2.

**Step 1:** The BI computes the bidding bonds message \( M_{\text{BI-BK}} \):

\[
M_{\text{BI-BK}} = (\text{ACC}_{\text{BI}}, \text{ACC}_{\text{PCC}}, \text{ID}_{\text{pro}}, \text{AMOUNT}, \text{ID}_{\text{BI}}, T_{\text{BI-BK}}). \tag{6}
\]

The BI then chooses a random number \( r_{1} \) and computes the ciphertexts \( C_{1} \) and \( C_{2} \) as follows:

\[
C_{1} = g^{r_{1}} \mod p \tag{7}
\]

\[
C_{2} = M_{\text{BI-BK}} \times y_{\text{BK}}^{-r_{1}} \mod p. \tag{8}
\]

The BI then sends the ciphertexts \( (C_{1}, C_{2}) \) to the BK.

**Step 2:** Upon receiving the ciphertexts \( (C_{1}, C_{2}) \), the BK uses the private key \( x_{\text{BK}} \) to decrypt the message:

\[
w = (C_{1}^{-x_{\text{BK}}})^{-1} \mod p \tag{9}
\]

\[
M_{\text{BI-BK}} = C_{2} \times w \mod p \tag{10}
\]

\[
M_{\text{BI-BK}} = (\text{ACC}_{\text{BI}}, \text{ACC}_{\text{PCC}}, \text{M}_{\text{pro}}, \text{ID}_{\text{pro}}, T_{\text{BK-BI}}). \tag{11}
\]

The BK then chooses a random number \( r_{2} \) makes a response message \( M_{\text{BK-BI}} \) and computes a signature \( (d_{1}, s_{1}) \):

\[
d_{1} = g^{r_{2}} \mod p \tag{12}
\]

\[
s_{1} = r_{2}^{-1}(M_{\text{BK-BI}} - x_{\text{BK}}d_{1}) \mod p - 1. \tag{13}
\]

After this, the BK sends the signature \( (d_{1}, s_{1}) \) and message \( M_{\text{BK-BI}} \) to the BI.

**Step 3:** After receiving the signature \( (d_{1}, s_{1}) \) and message \( M_{\text{BK-BI}} \), the BI uses the public key \( y_{\text{BK}} \) to verify the signature:

\[
y_{\text{BK}}^{d_{1}} \times d_{1}^{s_{1}} \equiv g^{M_{\text{BK-BI}}} \mod p. \tag{14}
\]

Then, the BI creates a subliminal message \( M_{s} \) and message \( M_{\text{BI-PCC}} \) as follows:

\[
M_{s} = ((d_{1}, s_{1}), x_{\text{BI}}, \text{ID}_{\text{BI}}, \text{ID}_{\text{PCC}}, \text{M}_{\text{inf}}, \text{ID}_{\text{pro}}) \tag{15}
\]

\[
M_{\text{BI-PCC}} = \text{ID}_{\text{pro}}^{M_{s}} \mod p - 1. \tag{16}
\]

It sends a blind signature request message \( M_{\text{req}} \) to the PCC.

**Step 4:** The PCC chooses a random number \( k \) and computes parameter \( d_{2} \):

\[
d_{2} = g^{k} \mod p. \tag{17}
\]

The PCC sends the parameter \( d_{2} \) to the BI.

**Step 5:** The BI chooses random numbers \( (a, b, c) \) and computes:

\[
d_{2} = d_{2}^{a} y_{\text{BK}}^{b} \mod p \tag{18}
\]

\[
M_{\text{BI-PCC}} = a^{-1}(d_{2} + h(M_{\text{BI-PCC}}) - b) - \hat{d}_{2} \mod p - 1. \tag{19}
\]

\( M_{\text{BI-PCC}} \) is then sent to the PCC to sign the message.

**Step 6:** The PCC computes:

\[
\tilde{s}_{2} = (\tilde{d}_{2} + M_{\text{BI-PCC}}) x_{\text{PCC}} - k \mod p - 1 \tag{20}
\]

and then sends \( \tilde{s}_{2} \) to the BI.

**Step 7:** After receiving the blind signature \( \tilde{s}_{2} \), the BI decrypts the blind message and obtains the signature:

\[
s_{2} = a \tilde{s}_{2} - c \mod p - 1 \tag{21}
\]
Figure 2
The scenario of the bidding phase

\[
\begin{align*}
\text{BI} &\quad \text{PCC} &\quad \text{PS} \\
\text{private key } x_{BI} &\quad \text{private key } x_{PCC} &\quad \text{private key } x_{PS} \\
y_{BI} = g^{x_{BI}} \mod p &\quad y_{PCC} = g^{x_{PCC}} \mod p &\quad y_{PS} = g^{x_{PS}} \mod p \\
M_{BI-BK} = (\text{ACC}_{BI}, \text{ACC}_{PCC}, ID_{PCC}, ID_{PS}, T_{BI-BK}) &\quad M_{BK-BI} = (\text{ACC}_{BI}, \text{ACC}_{PCC}, M_{PS}, ID_{PCC}, T_{BK-BI}) &\quad M_{PS-BI} = (M_{BI-PS}, T_{BI-PS}) \\
\text{choose a random number } r_1 &\quad \text{choose a random number } r_2 &\quad \text{choose a random number } r_3 \\
C_1 = g^{r_1} \mod p &\quad d_1 = g^{r_2} \mod p &\quad d_3 = g^{r_3} \mod p \\
C_2 = M_{BI-BK} \times y_{BI}^{r_1} \mod p &\quad s_2 = r_2^{-1} (M_{BK-BI} - x_{BI} d_1) \mod p - 1 \\
(C_1, C_2) &\quad s_2 = (d_2 + M_{BI-PCC}) \times y_{PCC} - k \mod p - 1 \\
M_{BK-BI} = C_2 \times w \mod p &\quad \tilde{d}_2 = g^k \mod p \\
M_{BI-BK} = (\text{ACC}_{BI}, \text{ACC}_{PCC}, M_{PS}, ID_{PCC}, T_{BI-BK}) &\quad \text{choose a random number } k \\
M_{BI-PCC} = ID_{PCC} \mod p - 1 &\quad \text{choose random numbers } (a, b, c) \\
M_{BI-PS} = (a, c, ID_{PS}, \text{Amount}, ID_{PCC}, T_{PS}, ID_{PCC}, T_{BI-PS}) &\quad d_2 = \tilde{d}_2 - y_B g \mod p \\
\text{choose a random number } r_3 &\quad M_{BI-PS} = a^{-1} (d_2 + b (M_{BI-PCC}) - b) - \tilde{d}_2 \mod p - 1 \\
C_3 = g^{r_3} \mod p &\quad \tilde{M}_{BI-PCC} = a^{-1} (d_2 + b (M_{BI-PCC}) - b) - \tilde{d}_2 \mod p - 1 \\
C_4 = M_{BI-PS} \times y_{PS}^{r_3} \mod p &\quad s_2 = \tilde{d}_2 + \tilde{M}_{BI-PCC} \times y_{PCC} - k \mod p - 1 \\
(C_3, C_4) &\quad (M_{PS-BI}, d_3, s_3) \\
M_{PS-BI} = C_4 \times w \mod p &\quad \text{choose a random number } k_1 \\
M_{PS-PS} = (M_{PS-BI}, T_{PS-PS}) &\quad d_5 = g^{r_3} \mod p \\
\text{choose a random number } r_5 &\quad s_3 = k_1^{-1} (M_{PS-PS} - x_{PS} d_3) \mod p - 1 \\
M_{PS-PS} = (M_{PS-BI}, T_{PS-PS}) &\quad (d_3, s_3) \\
y_{PS}^{d_5} \times d_3^{r_3} = g^{M_{PS-PS}} \mod p
\end{align*}
\]
so that the BI can obtain the signature \( (M_{BI-PCC}, d_2, s_2) \). The BI makes a bidding message for the PS:

\[
M_{BI-PS} = (a, c, AMOUNT_ID, ID_{BI}, ID_{PCC}, T_{a}, ID_{PCC}, T_{BI-PS}) \tag{22}
\]

It chooses a random number \( r_3 \) and computes ciphertexts \( (C_3, C_4) \):

\[
C_3 = g^{r_3} \mod p \tag{23}
\]

\[
C_4 = M_{BI-PS} \times y_{PS}^{r_3} \mod p \tag{24}
\]

It sends the ciphertexts \( (C_3, C_4) \) to the PS.

**Step 8:** After receiving the ciphertext \( (C_3, C_4) \), the PS uses the private key \( x_{PS} \) to decrypt the messages \( C_3 \) and \( C_4 \):

\[
w_1 = (C_3^{x_{PS}})^{-1} \mod p \tag{25}
\]

\[
M_{BI-PS} = C_4 \times w_1 \mod p. \tag{26}
\]

Then, the BK chooses a random number \( k_2 \) and sends a response message \( M_{PS-BI} \) (where \( M_{PS-BI} = (M_{BI-PS}, T_{BI-PS}) \)), chooses a random number \( k_1 \) and computes a signature \( (d_3, s_3) \):

\[
d_3 = g^{k_1} \mod p \tag{27}
\]

\[
s_3 = k_1^{-1} (M_{PS-BI} - x_{PS}d_3) \mod p - 1. \tag{28}
\]

After this, the PS sends the signature \( (d_3, s_3) \) and message \( M_{PS-BI} \) to the BI.

**Step 9:** Upon receiving the signature \( (d_3, s_3) \) and message \( M_{PS-BI} \), the BI verifies the signature as follows:

\[
y_{PS} d_1 \times d_3 s_3 \times g^{M_{PS-BI}} \mod p. \tag{29}
\]

### 2.3. The Bidding Phase

After the deadline of the casting bid, PS sends the blind factor to the PCC. The PCC then uses the blind factor to obtain the bidding price. After this, the PCC publishes the winner. The scenario of the opening bid phase is shown in Figure 3.

**Step 1:** The PS chooses a random number \( k_2 \) and creates a message \( M_{PSV} \) (where \( M_{PSV} = (M_{BI-PS}, T_{PS-PCC}) \)) and a signature \( (d_4, s_4) \):

\[
d_4 = g^{k_4} \mod p \tag{30}
\]

**Step 2:** When the PS-BI receives \( M_{PSV} \) and \( M_{PS-BI} \) from the BK, it verifies the signature as follows:

\[
y_{PS} d_1 \times d_3 s_3 \times g^{M_{PS-BI}} \mod p. \tag{31}
\]

**Step 3:** The BI chooses a random number \( k_3 \) and creates a message \( M_{PCC} \) (where \( M_{PCC} = (M_{PCC-BI}, T_{PCC-BI}) \)) and a signature \( (d_5, s_5) \):

\[
d_5 = g^{k_5} \mod p \tag{32}
\]

\[
s_5 = k_5^{-1} (M_{PCC-BI} - x_{PCC}d_5) \mod p - 1. \tag{33}
\]

**Step 4:** When the PCC-BI receives \( M_{PCC} \) and \( M_{PSV} \) from the PS, it verifies the signature as follows:

\[
y_{PCC} d_4 \times d_5 s_5 \times g^{M_{PSV}} \mod p. \tag{34}
\]

**Step 5:** The PCC-BI chooses a random number \( k_4 \) and creates a message \( M_{PCC-V} \) (where \( M_{PCC-V} = (M_{PCC-BI}, T_{PCC-V}) \)) and a signature \( (d_6, s_6) \):

\[
d_6 = g^{k_6} \mod p \tag{35}
\]

\[
s_6 = k_6^{-1} (M_{PCC-V} - x_{PCC}d_6) \mod p - 1. \tag{36}
\]

**Step 6:** When the PCC-V receives \( M_{PCC-V} \) and \( M_{PCC-BI} \) from the PCC-BI, it verifies the signature as follows:

\[
y_{PCC} d_5 \times d_6 s_6 \times g^{M_{PCC-BI}} \mod p. \tag{37}
\]
\[ s_4 = k_2^{-1}(M_{PS} - x_{PC}d_4) \mod p -1. \] (31)

The PS then creates a bidding message:
\[ M_{PS-PCC} = (a, c, ID_{BI} \cdot ID_{PCC}, \text{Amount}, ID_{pro} \cdot T_{PS-PCC}, d_4, s_4, M_{PS}). \] (32)

Next it chooses a random number \( r \) and computes ciphertexts \((C_5, C_6)\):
\[ C_5 = g^{r_i} \mod p \] (33)
\[ C_6 = M_{PS-PCC} \times y_{PCC}^{r_i} \mod p. \] (34)

It then sends the ciphertexts \((C_5, C_6)\) to the PCC.

**Step 2:** After receiving the ciphertexts \((C_5, C_6)\), the PCC uses the private key \( x_{PCC} \) to decrypt the message:
\[ w_3 = (C_5)^{x_{PCC}} \mod p \] (35)
\[ M_{PS-PCC} = C_6 \times w_3 \mod p. \] (36)

The PCC verifies if the bidding message is valid or not:
\[ y_{PS} d_4 \times d_4 s_4 \equiv g^{M_{PS}} \mod p. \] (37)

Then, the PCC generates a response message \( M_{PCC-PS} \)
\((where \( M_{PCC-PS} = (M_{PCC-RES}, T_{PCC-PS}) \)).
The PCC chooses a random number \( k_3 \) and computes the signature \((d_3, s_3)\) as follows:
\[ d_3 = g^{k_3} \mod p \] (38)
\[ s_3 = k_3^{-1}(M_{PCC-PS} - x_{PCC}d_3) \mod p -1. \] (39)

Afterwards, the PCC sends the signature \((d_3, s_3)\) and \( M_{PCC-PS} \) to the PS.

**Step 3:** Upon receiving the signature \((d_3, s_3)\), the PS verifies the signature:
\[ y_{PCC} d_4 x_5 s_5 \equiv 2^{M_{PCC-PS}} \mod p. \] (40)

### 2.4. The Official Agent Arbitration Phase

In this phase, if there are concerns regarding the BI or PCC, the OA can offer fair arbitration by blind message and subliminal message. Once the accusation holds, the winner can request the bank to reveal the information of the accused party.

**Case 1:** The BI takes the subliminal message, blind message and blind factor to the OA. The OA then uses the related information to make a fair arbitration. The overview of the arbitration phase (PCC is accused) is shown in Figure 4:

**Step 1:** The BI generates an accusation message \( M_{BI-OA} \):
\[ M_{BI-OA} = (M_{S}, ID_{BI}, ID_{PCC}, ID_{pro}, M_{BI-PCC}, d_4, s_4). \] (41)

It then chooses a random number \( r \) and computes the ciphertexts \((C_7, C_8)\):
\[ C_7 = g^{r_i} \mod p \] (42)
\[ C_8 = M_{BI-OA} \times y_{OA}^{r_i} \mod p. \] (43)

It then sends the ciphertexts \((C_7, C_8)\) to the OA.

**Step 2:** Upon receiving the ciphertexts \((C_7, C_8)\), the OA uses the private key \( x_{OA} \) to decrypt the message:
\[ w_3 = (C_7)^{x_{OA}} \mod p \] (44)
\[ M_{BI-OA} = C_8 \times w_3 \mod p. \] (45)

The OA then verifies the signature \((d_3, s_3)\):
\[ y_{PCC} d_4 x_5 d_4 s_4 \equiv d_2 g^{r_i} \mod p. \] (46)

If Equation (47) holds, it then verifies the message \( M_{BI-PCC} \):
\[ M_{BI-PCC} \equiv lD_{pro}^{M_{PS}} \mod p -1. \] (47)

Using subliminal message \( M_{S} \) and accusation message \( M_{BI-OA} \), the OA is able to make a fair arbitration.

**Case 2:** The PCC takes the blind message and blind factor to the OA. Then, the OA uses the related information to make a fair arbitration. The overview of the arbitration phase (BI is accused) is shown in Figure 5:

**Step 1:** The PCC generates an accusation message \( M_{PCC-OA} \)
\((where \( M_{PCC-OA} = (ID_{BI}, ID_{PCC}, ID_{pro}, M_{PS-PCC}, D_{PS}, d_4, s_4) \)).
The PCC then chooses a random number \( r \) and computes ciphertexts \((C_9, C_{10})\):

\[ s_4 = k_2^{-1}(M_{PS} - x_{PC}d_4) \mod p -1. \] (31)

The OA then verifies the signature \((d_3, s_3)\):
\[ y_{PCC} d_4 x_5 s_5 \equiv 2^{M_{PCC-PS}} \mod p. \] (40)

Using subliminal message \( M_{S} \) and accusation message \( M_{BI-OA} \), the OA is able to make a fair arbitration.
The overview of the arbitration phase (PCC is accused)

\[
M_{BI-OA} = (M_S, ID_{BI}, ID_{PCC}, ID_{pro}, M_{BI-PCC}, M_{PSV}, d_2, s_2)
\]

choose a random number \( r_5 \)
\[
C_7 = g^{r_5} \mod p
\]
\[
C_8 = M_{BI-OA} \times y_{OA}^{r_5} \mod p
\]
\[
(C_7, C_8)
\]
\[
w_3 = (C_7^{x_{OA}})^{-1} \mod p
\]
\[
M_{BI-OA} = C_8 \times w_3 \mod p
\]
\[
y_{PCC}^{d_2 \times h(M_{BI-PCC})} \mod p
\]
\[
M_{BI-PCC} \equiv ID_{pro}^{M_{PSV}} \mod p - 1
\]

The OA then verifies the signature \((d_p, s_p)\):
\[
y_{PS}^{d_4} \times d_4^{s_p} \equiv g^{M_{PSV}} \mod p
\]
Using verified message \(M_{PSV}\) and accusation message \(M_{PCC-OA}\), the OA can make a fair arbitration. The summary flowchart of the arbitration phase is shown in Figure 6.
3. Security Analysis

3.1. The Official Agent Arbitration Phase

The proposed protocol employs a digital signature mechanism to solve the non-repudiation problem with ElGamal signature and blind signature. The verifications are shown in Table 1.

3.2. Fair Arbitration

This section will illustrate how the proposed scheme provides fair arbitration.

Case 1: The BI doubts the PCC

In the arbitration phase, after receiving the message \((C_7, C_8)\) from the BI, the OA will decrypt the ci-
phantexts and verify the BI’s blind message $M_{BI-PCC}$: $y_{BK}^{d_2 + (M_{BI-PCC}) \mod p} \equiv d_2 g^{s_2} \mod p$. If the equation holds, the OA will verify the reality of message $M_{BI-PCC}$ using $M_{BI-PCC}^2 ID_{case \ mod \ p-1}$. Then, the OA uses $M_{BI-PCC}$ and $M_5$ to arbitrate the accusation.

The derivation of the blind signature verification is shown as follows:

$$y_{PCC}^{d_3 + (M_{BI-PCC}) \mod p} = d_3 g^{s_2} \mod p$$

$$= (g^{ka} g_{X_{PCC}} b g^{c}) g^{as_2 - c}$$

$$= g^{ka} g_{X_{PCC}} b g^{as_2}$$

$$= g^{ka} g_{X_{PCC}} b g^{d_2 (d_2 + M_{BI-PCC}) x_{PCC} - k}$$

$$= g^{ka} g_{X_{PCC}} b g^{a x_{PCC} d_2} g^{a x_{PCC} M_{BI-PCC}} / g^{ka}$$

$$= g^{X_{PCC} b g^{a x_{PCC} d_2} g^{a x_{PCC} M_{BI-PCC}}}$$

$$= g^{X_{PCC} b g^{a x_{PCC} d_2} g^{a x_{PCC} (d_2 + h(M_{BI-PCC}) - b) - d_2}}$$

$$= g^{X_{PCC} b g^{a x_{PCC} d_2} g^{X_{PCC} (d_2 + h(M_{BI-PCC}) - b) - d_2}}$$

$$= g^{X_{PCC} b g^{a x_{PCC} d_2} g^{X_{PCC} (d_2 + h(M_{BI-PCC})) / b}}$$

$$= g^{X_{PCC} d_2 + h(M_{BI-PCC})}$$

$$= y_{PCC}^{d_3 + (M_{BI-PCC}) \mod p}$$

Table 1

<table>
<thead>
<tr>
<th>Non-repudiation issues</th>
<th>Proof issuer</th>
<th>Proof holder</th>
<th>Verification</th>
</tr>
</thead>
<tbody>
<tr>
<td>$(M_{BK-BI}, d_1, s_1)$</td>
<td>BK</td>
<td>BI</td>
<td>$y_{BK}^{d_1} \times d_1^{s_1} \equiv g^{M_{BK-BI}} \mod p$</td>
</tr>
<tr>
<td>$(M_{BI-PCC}, d_2, s_2)$</td>
<td>BK</td>
<td>BI</td>
<td>$y_{BK}^{d_2 + (M_{BI-PCC}) \mod p} \equiv d_2 g^{s_2} \mod p$</td>
</tr>
<tr>
<td>$(M_{PS-BI}, d_3, s_3)$</td>
<td>PS</td>
<td>BI</td>
<td>$y_{PS}^{d_3} \times d_3^{s_3} \equiv g^{M_{PS-BI}} \mod p$</td>
</tr>
<tr>
<td>$(M_{PS-PCC}, d_4, s_4)$</td>
<td>PS</td>
<td>PCC</td>
<td>$y_{PS}^{d_4} \times d_4^{s_4} \equiv g^{M_{PS-PCC}} \mod p$</td>
</tr>
<tr>
<td>$(M_{PCC-PS}, d_5, s_5)$</td>
<td>PCC</td>
<td>PS</td>
<td>$y_{PCC}^{d_5} \times d_5^{s_5} \equiv g^{M_{PCC-PS}} \mod p$</td>
</tr>
</tbody>
</table>

If the equation holds, then the OA uses $M_{PS-PCC}$ to arbitrate the accusation. This design can ensure fair arbitration.

The derivation of signature verification is shown as follows:

$$g^{M_{PS-PCC} \mod p} = y_{PS}^{d_4} \times d_4^{s_4} \mod p$$

$$= g^{X_{PS} d_4} \times g^{d_4 (M_{PS-PCC} - d_4)}$$

$$= g^{X_{PS} d_4} \times g^{M_{PS-PCC} - d_4}$$

$$= g^{M_{PS-PCC}}.$$

3.3. The Blind and Unlinkable Issue

In the bidding phase, the BI sends the blind message to the PCC to sign the message. The blind signature scheme is secure against malicious attackers who work in the PCC. Even if the PCC receives the bidding message from the BI, the PCC does not know the message in the bidding phase. In the opening bid phase, the PCC retrieves the blind message in order to use the blind factor and other information from the PS. Therefore, the malicious attacker cannot obtain the BI’s bidding message and identity in the bidding phase.

3.4. Auditability

In the official agent arbitration phase, the OA can ask the bank for the information of the BI and PCC when it receives an accusation message. Therefore, the proposed protocol provides auditability to help the OA offer fair arbitration.
3.5. One-Time Registration

The BI and PCC only need to register once with the BK and PS. This reduces computation cost and creates greater convenience for the bidder.

3.6. Defense Against Known Attacks

3.6.1. Replay Attack

The proposed scheme includes a timestamp mechanism, which varies for each transaction. If a malicious attacker attempts to replay a message, it will fail, making replay attacks impossible.

3.6.2. Forgery Attack

In the bidding phase, when the BI sends the bidding message to the PCC, it only sends the blind message; it is very difficult for an attacker to forge the message to pretend to be the bidder or other parties. The proposed scheme uses ElGamal encryption and decryption mechanism in each transaction. For example: the encryptions in the bidding phase are:

\[ C_1 = g^5 \mod p, \quad C_2 = M_{BI-BK} \times y_2^5 \mod p, \]
\[ C_3 = g^5 \mod p, \quad C_4 = M_{BI-PS} \times y_{PS}^5 \mod p \]

And the decryptions are:

\[ w = (C_1^{y_{MC}})^{-1} \mod p, \]
\[ M_{BI-BK} = C_2 \times w \mod p, \]
\[ w_i = (C_3^{y_{PS}})^{-1} \mod p \quad \text{and} \quad M_{BI-PS} = C_4 \times w_i \mod p. \]

Thus, only the real receiver has his/her own private key, and can decrypt the message. Thus attackers cannot achieve forgery attacks.

3.6.3. Insider Attack

If an attacker works in the PCC or has a relationship with PCC staff, he/she will only be able to obtain the deadline of the bid in the bidding phase from these sources, not the bidding price; the proxy server transfers the bidding message to the PCC and uses the blind signature \((d_1, s_1)\) to protect the bidding price \(M_{ul}\) in the bidding phase.

4. Discussion

This paper compares the computation cost in Table 2, and makes a security comparison in Table 3. In the

Table 2

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>The registration phase</td>
<td>(5T_{Exp})</td>
<td>(6T_{Exp})</td>
<td>(5T_{Exp} + 2T_{Mut} + 2T_H)</td>
<td>(4T_{Exp} + 1T_{Mut} + 2T_H)</td>
<td>NA</td>
</tr>
<tr>
<td>The bidding phase</td>
<td>(12T_{Exp})</td>
<td>(10T_{Exp})</td>
<td>(8T_{Exp} + 7T_{Mut} + 6T_H)</td>
<td>(2T_{Exp} + 2T_{Mut} + 2nT_{soc})</td>
<td>(2T_{Exp} + 18T_{Exp}^+ + 20T_{Mut} + 20T_{Mut}^+)</td>
</tr>
<tr>
<td>The opening bid phase</td>
<td>NA</td>
<td>NA</td>
<td>(3T_{Exp} + 1T_{Mut} + 1T_H)</td>
<td>(1T_{Exp} + 2T_{Exp} + 2nT_{soc})</td>
<td>(13T_{Exp} + 11T_{Exp}^+ + 9T_{Mut})</td>
</tr>
<tr>
<td>The product exchange and the payment phase</td>
<td>(10T_{Exp})</td>
<td>(8T_{Exp})</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
</tr>
<tr>
<td>The official agent arbitration phase</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
<td>(13T_{Exp} + 13T_{Mut} + 10T_{Mut})</td>
</tr>
<tr>
<td>Total</td>
<td>(27nT_{Exp})</td>
<td>(24nT_{Exp})</td>
<td>(11T_{Exp} + 8T_{Mut} + 7T_{Mut}^+ \times n + 6T_{Exp}^+ + 2T_{Mut}^+ + 2T_{Exp}^+)</td>
<td>(3T_{Exp} + 4T_{Exp}^+ + 4nT_{soc} + 4T_{Exp}^+ + 1T_{Mut} + 2T_H)</td>
<td>(44T_{Exp} + 39T_{Mut} + 44T_{Exp}^+ + 2T_H)</td>
</tr>
<tr>
<td>Execution time</td>
<td>(\approx 16.2 n\ ms)</td>
<td>(\approx 14.88\ n\ ms)</td>
<td>(\approx (6.84 n + 3.73)\ ms)</td>
<td>(\approx (11.78 + 0.3 n)\ ms)</td>
<td>(\approx 27.38 n\ ms)</td>
</tr>
</tbody>
</table>

Note: \(n\) is the number of bidders; \(T_{Exp}\) is the time complexity of one-way hash function; \(T_{Mut}\) is the time for executing the modular exponential operation; \(T_{soc}\) is the time complexity for executing the modular multiplication; \(T_{Exp}^+\) is the time for exclusion or operation; \(T_{Mut}^+\) is the time cost of a scalar multiplication; and \(T_{pair}\) is the time cost of a pairing operation.

\(T_{pair} = 3.10\ ms, \ T_{Exp} = 0.62\ ms\) (on 3 GHz Pentium IV [13]), \(T_{pair} = 5T_{Exp}, T_{soc} = 29T_{Mut}, T_{Exp} = 240T_{Mut} [7,12,20].\)
Table 3
The security comparison of the related works and our scheme

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Privacy</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Atomicity</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Non-repudiation</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Unforgeability</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Unlinkability</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>One-time registration</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Auditability</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Off-line TTP</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Fair arbitration protocol</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
</tbody>
</table>

proposed scheme, an exponential operation is used to achieve a secure protocol. Although the proposed scheme requires greater computation cost, it is more secure than related works.

5. Conclusions

This paper proposes an electronic public engineering project bidding protocol via a subliminal channel which is suitable for the bidding protocol for public construction projects. The proposed scheme satisfies the following properties: non-repudiation, fair arbitration, blind message, unlinkable, one-time registration and auditability. An exponential operation and ElGamal encryption are used to ensure transaction processing safety, and subliminal messages and blind signatures are used to ensure bidders’ privacy and security. Moreover, a fair arbitration protocol was designed to ensure fair transactions. Although the computation cost of the proposed scheme is higher than related works, the scheme is more secure and suitable for public engineering projects.

Acknowledgements

This research was supported by the Ministry of Science and Technology, Taiwan, R.O.C., under contract number MOST 106-2622-E-305-001-CC2, MOST 106-2221-E-324 -013 and MOST103-2632-E-324-001-MY3.

References


11. Lu, Y., Li, L., Yang, X., Yang, Y. Robust Biometrics Based Authentication and Key Agreement Scheme for Multi-Server Environments Using Smart Cards. PLoS ONE, 2015, 10(5), e0126323.https://doi.org/10.1371/journal.pone.0126323


