A Secure and Efficient Protocol for Electronic Treasury Auctions

Atilla Bektaş[1], Mehmet Sabır Kiraz[2], Osmanbey Uzunkol[2]

[1]IAM, Middle East Technical University, Ankara, Turkey

[2]MCS Labs, TÜBİTAK BİLGEM, Kocaeli, Turkey

BalkanCryptSec 2014, İTÜ, İstanbul, Turkey

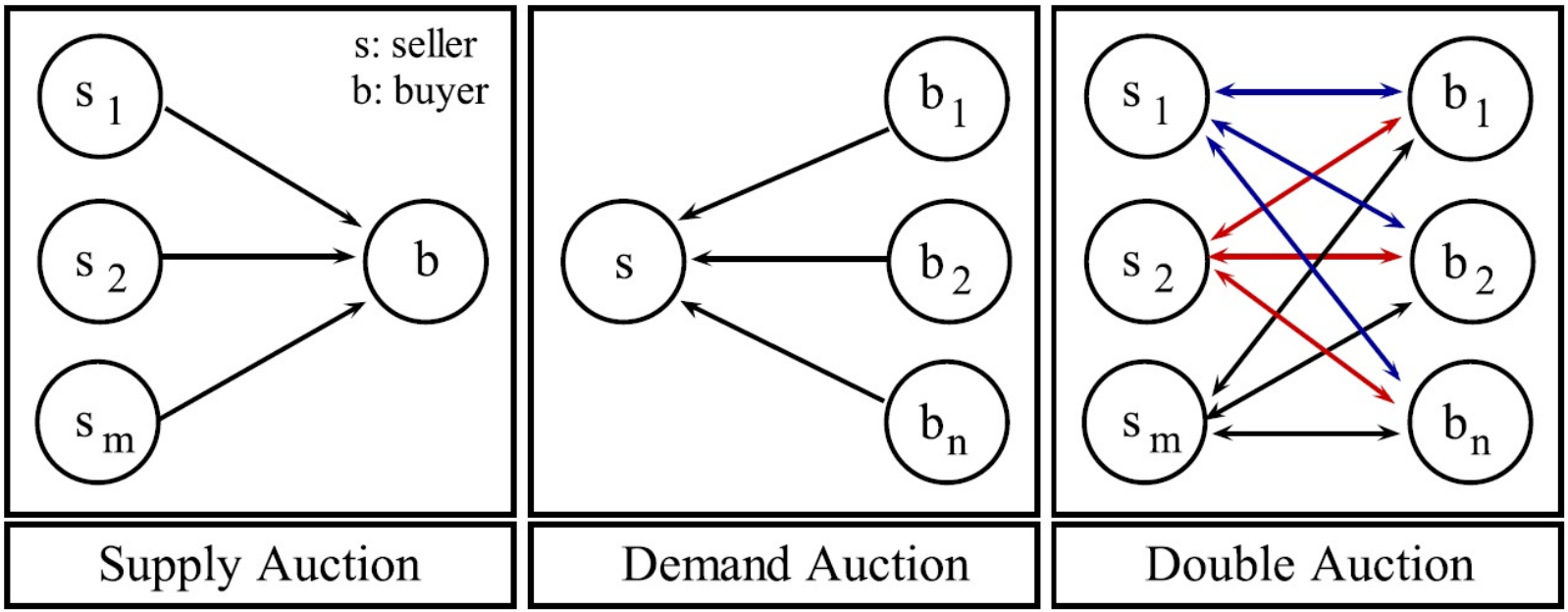
October 17, 2014

Outline

Auctions

Auction: A mechanism with predefined rules for buying and selling.

According to the number of participants



Auctions

According to items

- **Single-Unit Auction:** Only one item is available for sale.
- **Multi-Unit Auction:** More than one homogeneous/identical item is being auctioned.
- **Multi-Object Auction:** Heterogeneous/differentiated items are being auctioned.

This Work: Treasury Auctions

- Method of borrowing money from the market
- Treasury holds regular auctions for government securities (bonds and bills)
- Buyers submit bids (quantity and price)
- Bids are ranked in order
- The quantity for sale is awarded to the best bids
- USA (world's largest and most active market), UK, Germany and Turkey use similar methods

An Example of a Treasury Auction (Turkey)

- Performed only by the Treasury (Republic of Turkey Prime Ministry Undersecretariat of Treasury.)
- Central Bank = Financial agent of the auction process
- Primary Dealers = Authorized banks in Turkey
- Government Domestic Debt Securities (GDDSs)
  - Government bonds: Maturity $\geq$ 1 year (364 days)
  - Treasury bills: Maturity $<$ 1 year (364 days)

An Example of a Treasury Auction (Turkey)

Phase 0: Public Call

Treasury issues invitations for auction (is announced on the web)

Phase 1: Submission

Primary Dealers participate in the auction by submitting their unencrypted bids (offers) to the Central Bank

Phase 2: Sorting

Central Bank sorts the list of bids by unit price and sends the ordered list to the Treasury

An Example of a Treasury Auction (Turkey)

Phase 3: Cut-Off Point

Treasury determines a cut-off point manually, determines the list of accepted / rejected primary dealers and sends the list of accepted bidders to the Central Bank

Phase 4: Announcement of the Winners

Central Bank informs the bidders about the results

An Example of a Treasury Auction (Turkey)

Submitted Bids

Order	**Name of the Bank**	**Unit Price (TRY 100)** p_i	**Nominal Amount** a_i
1.	Bank 1	94.80	30,000
2.	Bank 2	94.00	50,000
3.	Bank 3	94.50	50,000
4.	Bank 2	94.80	60,000
5.	Bank 4	95.00	30,000
6.	Bank 5	94.70	60,000

An Example of a Treasury Auction (Turkey)

Sorted Bids

New Order	**Name of the Bank**	**Unit Price (TRY 100)** p_i	**Nominal Amount** a_i	**Amount** $\frac{p_i \cdot a_i}{100}$
5. $\rightarrow$ 1.	Bank 4	95.00	30,000	28,500
1. $\rightarrow$ 2.	Bank 1	94.80	30,000	28,440
4. $\rightarrow$ 3.	Bank 2	94.80	60,000	56,880
6. $\rightarrow$ 4.	Bank 5	94.70	60,000	56,820
3. $\rightarrow$ 5.	Bank 3	94.50	50,000	47,250
2. $\rightarrow$ 6.	Bank 2	94.00	50,000	47,000

$$\delta = TRY175,000 \rightarrow \sum_{i=1}^{5} \frac{p_i \cdot a_i}{100} \geq 175,000 \quad \text{and} \quad \sum_{i=1}^{4} \frac{p_i \cdot a_i}{100} < 175,000$$

An Example of a Treasury Auction (Turkey)

Cut-off Point

New Order	**Name of the Bank**	**Unit Price (TRY 100)** p_i	**Nominal Amount** a_i	**Amount** $\frac{p_i \cdot a_i}{100}$
1.	Bank 4	95.00	30,000	28,500
2.	Bank 1	94.80	30,000	28,440
3.	Bank 2	94.80	60,000	56,880
4.	Bank 5	94.70	60,000	56,820
5.	Bank 3	94.50	50,000	47,250
6.	Bank 2	94.00	50,000	47,000

$\Longrightarrow$ Cut-off point $= 4$

Current Privacy Issues

- Bids are submitted in clear text
- The names of the investors are not hidden in the list
- A malicious Treasury can change
  - the order on the lists
  - the cut-off point

Our Contribution

- Avoid manual listing and manual determination of the cut-off point
- Achieve correctness and privacy in the malicious model
- Submit bids in a secure way (i.e. confidentiality & privacy)
- Propose a model by
  - Collecting signed encrypted bids
  - Putting the list in an order and determining the cut-off point under encryption
  - Publishing only the winners
  - Ensuring losers that they indeed loose

By using SMPC, Secret Sharing and Threshold Homomorphic Cryptosystem (Paillier).

Paillier Cryptosystem

$n = pq$ where $p \neq q$ large primes
$g \in_R \mathbb{Z}^*_{n^2}$ with $n \mid \text{ord}(g)$
$\lambda := lcm(p-1, q-1)$
$\mu := (L(g^\lambda \bmod n^2))^{-1} \bmod n$ where $L(x) = \frac{x-1}{n}$

Public key (pk)	:	(n, g)
Secret key (sk)	:	(λ, μ)
Encryption	:	plaintext $m < n$ random value $r < n$ ciphertext $c = g^m.r^n \bmod n^2$
Decryption	:	ciphertext $c < n^2$ plaintext $m = L(c^\lambda \bmod n^2).\mu \bmod n$

Paillier Encryption is Additively Homomorphic

$$\begin{aligned}
\mathrm{Enc}_{pk}(m_1, r_1) \cdot \mathrm{Enc}_{pk}(m_2, r_2) &= (g^{m_1}.r_1^n \bmod n^2).(g^{m_2}.r_2^n \bmod n^2) \\
&= (g^{m_1}.r_1^n).(g^{m_2}.r_2^n) \bmod n^2 \\
&= g^{m_1+m_2}.(r_1.r_2)^n \bmod n^2 \\
&= \mathrm{Enc}_{pk}(m_1 + m_2, r_1.r_2)
\end{aligned}$$

Our Protocol: Submission and Evaluation Phase

- **Private Input**

  Primary Dealer (PD_i) : $B_i := (PD_i, p_i, a_i)$, $sk_{PD_i}^{(1)}$

  Central Bank (CB) : sk_{CB}

  Treasury (T) : δ, sk_T, $sk_{PD_i}^{(2)}$

- **Public Input**

  Primary Dealer (PD_i) : pk_{PD_i}, pk_{CB}, pk_T

  Central Bank (CB) : pk_{PD_i}, pk_{CB}, pk_T

  Treasury (T) : pk_{PD_i}, pk_{CB}, pk_T

Our Protocol: Submission and Evaluation Phase

Primary Dealer (PD_i) computes

- (1) $y_i := (p_i \cdot a_i)/100$ that is amount of payment
- (2) $S_{B_i} := \mathrm{Sign}_{PD_i}[\mathrm{Hash}(B_i)]$
- (3) $X_i := (\mathrm{Enc}_{pk_{PD_i}}(S_{B_i}), \mathrm{Enc}_{pk_T}(p_i), \mathrm{Enc}_{pk_T}(a_i), \mathrm{Enc}_{pk_T}(y_i))$

and sends X_i to the Central Bank (CB).

Our Protocol: Submission and Evaluation Phase

Treasury (T) computes

- $\mathrm{Enc}_{pk_T}(\delta)$ where δ is the amount of required debt of the Treasury

and sends $\mathrm{Sign}_T[\mathrm{Enc}_{pk_T}(\delta)]$ to the Central Bank (CB).

Our Protocol: Submission and Evaluation Phase

Central Bank (*CB*)

- (1) Verifies $\mathrm{Sign}_T[\mathrm{Enc}_{pk_T}(\delta)]$
- (2) Computes $output_1 := \prod_{i=1}^{k} \mathrm{Enc}_{pk_T}(y_i) = \mathrm{Enc}_{pk_T}(\sum_{i=1}^{k} y_i)$

$$output_2 := \prod_{i=1}^{k} \mathrm{Enc}_{pk_T}(a_i) = \mathrm{Enc}_{pk_T}(\sum_{i=1}^{k} a_i)$$

Our Protocol: Submission and Evaluation Phase

- (3) Runs subprotocols using $\mathrm{Enc}_{pk_T}(\delta)$ and X_i's
- (4) Computes $output_3 := \mathrm{Enc}_{pk_T}(p_k)$, $output_4 := \prod_{j=1}^{m} \mathrm{Enc}_{pk_T}(y_j)$

$$output_5 := \prod_{j=1}^{m} \mathrm{Enc}_{pk_T}(a_j) \text{ , } output_6 := \mathrm{Enc}_{pk_T}(p_m)$$

where k is the number of bids, m is the cut-off point and j is the position of the bid in the sorted list.

and sends $\mathrm{Sign}_{CB}[\langle \mathrm{output}_i, X_j \rangle : i = 1, \ldots, 6,\ j = 1, \ldots, m]$ to the Treasury (T).

Our Protocol: Submission and Evaluation Phase

Treasury (T)

- (1) Verifies $\text{Sign}_{CB}[\langle \text{output}_i, X_j \rangle : i = 1, \ldots, 6,\ j = 1, \ldots, m]$
- (2) Computes $\text{Dec}_{sk_T}([\langle \text{output}_i \rangle : i = 1, \ldots, 6])$
- (3) Computes $H_j := \text{Hash}(X_j)$ for $j = 1, \ldots, m$
- (4) Forms a lookup table with rows $\langle X_j, H_j \rangle$

Our Protocol: Award Phase

Primary Dealer (PD_i) computes

- Hash(X_i) and sends it with his certificate $cert_i$ to the Treasury (T).

Treasury (T)

- (1) Verifies Hash(X_i) $\stackrel{?}{\in} \{H_j : j = 1, \ldots, m\}$ and determines $res =$ "Accept/Reject"
- (2) Computes $\mathrm{Dec}_{sk_{PD_i}^{(2)}}(\mathrm{Enc}_{pk_{PD_i}}(\mathrm{Sign}_T[res]))$ and sends it to Primary Dealer (PD_i)

Our Protocol: Award Phase

Primary Dealer (PD_i)

- (1) Computes $\mathrm{Dec}_{sk_{PD_i}^{(1)}}(\mathrm{Dec}_{sk_{PD_i}^{(2)}}(\mathrm{Enc}_{pk_{PD_i}}(\mathrm{Sign}_T[res])))$ to get $\mathrm{Sign}_T[res]$
- (2) Verifies $\mathrm{Sign}_T[res]$
- (3) Computes $\mathrm{Dec}_{sk_{PD_i}^{(1)}}(\mathrm{Enc}_{pk_{PD_i}}(S_{B_i}))$ and sends it to the Treasury (T).

Our Protocol: Award Phase

Treasury (T)

- (1) Computes $\mathrm{Dec}_{sk_{PD_i}^{(2)}}(\mathrm{Dec}_{sk_{PD_i}^{(1)}}(\mathrm{Enc}_{pk_{PD_i}}(S_{B_i})))$ to get S_{B_i}
- (2) Verifies $\mathrm{Sign}_{PD_i}[\mathrm{Hash}(B_i)]$
- (3) Forms $B_j = (PD_j, p_j, a_j)$ and computes $\mathrm{Hash}(B_j)$
- (4) Verifies $\mathrm{Hash}(B_i) \stackrel{?}{=} \mathrm{Hash}(B_j)$

Security Analysis

- Our assumption is that the Treasury and the Central Bank do not collude
- Malicious parties cannot see the bids of honest primary dealers (using randomized encryption)
- No malicious party can submit a bid on behalf of an honest user (using digital signatures)
- The identity of a primary dealer is encrypted using a (2,2)-threshold homomorphic encryption scheme and the identity of the winners are only revealed during the award phase
- Accept/Reject response can only be seen by the corresponding primary dealer because the primary dealer performs the second decryption process privately (using $sk_{PD_i}^{(1)}$)

Security Analysis

- A malicious Treasury
  - cannot compute any additional information during the submission and evaluation phase (gets only the encryptions from the Central Bank)
  - cannot obtain the identity of the bidders during the submission and evaluation phase (obtains the encrypted ordered list of the accepted bidders from the Central Bank and the list is anonymised)
  - cannot learn any additional information about the rejected bids during the award phase (obtains only hashed values)
- A malicious Central Bank
  - cannot learn any useful information about bids (all the information is encrypted on this side)
  - cannot see the sum values which are total amount offered, total nominal amount offered, and after sorting process total amount accepted, total nominal amount accepted in plain form (no knowledge of the decryption key)

Questions?

THANK YOU...