Commit

minors

Browse files

main

haoqianzhang committed on Jan 13

1 parent 43ef659 commit fdd3df9

Showing **2 changed files** with **43 additions** and **33 deletions**.

Whitespace | Ignore whitespace | Split | Unified

Filter changed files

- content
  - 2-background.tex
  - 3-contract.tex

content/2-background.tex (30)

```
@@ -13,8 +13,9 @@ \subsection{Commit-and-Reveal Smart Contract for Sealed-bid Auctions}\label{sec:
 \textbf{Hiding:} No other entity except for the bidder can know the bidder's bid during the bidding period.
 % \textbf{Symbolic/cryptographic binding:} After the \bc commits a bidder's bid, it cannot be changed.
 % \textbf{Financial binding:} The bidder can always pay for the item that they bid for.
-- \textbf{Binding:} No bidder can change their bid
-- once the \bc finalizes the bidding transaction.
+ \textbf{Binding:} A bidder can not change their bid
+ once the \bc finalizes the bidding transaction
+ and can pay for what they bid.
 % \textbf{Revealing:} A bidder can choose not to reveal their bid during the revealing phase at the cost of their deposit.
 \textbf{Revealing:} All the sealed bids will be revealed during the revealing period.
 %We formally define all of these properties in \Cref{sec:analysis:scheme}.
@@ -76,7 +77,7 @@ \subsection{Commit-and-Reveal Smart Contract for Sealed-bid Auctions}\label{sec:
 }
 \;
 \Reveal{Upon receiving $i$'s bid $b_i$ and salt $r_i$ first time in revealing period }{
-    Assert(Hash$(b_i,r_i) = hash[i]$) \;
+    Assert(Hash$(b_i,r_i) = hash[i]$)\label{algline:correctness} \;
     % \If{Hash$(b_i,r_i) \neq hash[i]$}{\label{algline:correctness:start}
     %     keep deposit $d$
     % }\label{algline:correctness:end}
@@ -118,11 +119,11 @@ \subsection{Commit-and-Reveal Smart Contract for Sealed-bid Auctions}\label{sec:
 the contract determines the $winner$,
 which is the bidder who submitted the highest bid and can pay for their bid.

-\zhq{Need to update}
-Informally, the \car auction smart contract as described in~\Cref{code:traditional}, together with a cryptographic commitment scheme used to commit the bids, satisfies all the above four properties.
-The specific hiding and symbolic/cryptographic binding properties (e.g. computational, statistical etc.) follow from the underlying cryptographic commitment scheme.
-The financial binding and revealing properties follow from the fact that the deposit held by the smart contract is larger than the bids as well as the check done by the smart contract (lines \ref{algline:correctness:start}~\ref{algline:correctness:end} in~\Cref{code:traditional}) to ensure that the deposit is slashed if bidders do not reveal their bids or reveal an incorrect bid.
-Therefore, rational bidders will choose to reveal their bids during the reveal phase of~\Cref{code:traditional}.
+The \car auction smart contract
+with the cryptographic commitment scheme
+used to commit the bids
+satisfies the hiding property and
+partially assures the revealing and binding properties

+The hiding property directly
+follows from the underlying cryptographic commitment scheme.
+% The specific hiding and symbolic/cryptographic binding properties (e.g. computational, statistical etc.) follow from the underlying cryptographic commitment scheme.
+The binding and revealing properties follow from the fact that
+the deposit held by the smart contract is larger than the bids
+as well as the check done by the smart contract (\Cref{algline:correctness})
+to ensure that the smart contract will slash the deposit
+if bidders do not reveal their bids or reveal an incorrect bid.
+Therefore, rational bidders will choose to reveal their bids
+during the reveal phase.

 However, this contract has several notable drawbacks:

@@ -172,7 +183,8 @@ \subsection{Delayed Execution Abstraction}\label{sec:delayed}
 with a delay~\cite{zhang2023f3b,das2020better}.
 We require that
 \cns must execute all transactions
-with a fixed delay time, and we elaborate in more detail the reason for this requirement in~\Cref{sec:attack}.
+with a fixed delay time.
+% and we elaborate in more detail the reason for this requirement in~\Cref{sec:attack}.

 Furthermore, \cns should not observe the content of any transaction during the delayed period
 to ensure the effectiveness of the delay execution.
```

content/3-contract.tex (46)

```
@@ -68,19 +68,20 @@ \section{Auction Smart Contract with Delayed Execution}
 \subsection{Requirements}
 We require the delay time in the delayed execution
 to be the same as the confirmation time $T$
-in the underlying blockchain so that,
-when the blockchain decrypts and executes the transaction,
-the encrypted transaction has already been firmly written into the blockchain.
+in the underlying blockchain,
+Thus, when the blockchain decrypts and executes the transaction,
+the encrypted transaction has already been firmly written into the blockchain.
 We also require the bidding time in any sealed-bid auction
-to be $T$ to ensure the hiding property.
-We further demand that all transactions delay $T$ time,
+to be $T$.
+% to ensure the hiding property.
+We further demand that the \bc delay executes all transactions by $T$ time,
 including non-auction transactions,
-such as transfer transactions,
-to guarantee the binding property.
-Note that even though all transactions are delayed by $T$ time,
-they still have a similar finalization time
-as without $T$ time delay
-as illustrated in~\Cref{sec:choosingdelay}.
+such as transfer transactions.
+% to guarantee the binding property.
+% Note that even though all transactions are delayed by $T$ time,
+% they still have a similar finalization time
+% as without $T$ time delay
+% as illustrated in~\Cref{sec:choosingdelay}.

 \subsection{Pseudocode}

@@ -109,9 +110,7 @@ \subsection{Pseudocode}
 If not, the bidder,
 no matter how, loses the auction.

-
-
-\textbf{Under the \DE Environment:}
+\subsection{Under the \DE Environment}
 \contract as presented in~\Cref{code:zeroauction}
 implements an open auction as none of the bids are hidden.
 However, it becomes a \sa
@@ -242,13 +241,13 @@ \subsection{Running Examples}
 Thus, it does not affect the auction.

 \textbf{Two auctions:} A more interesting case is that
-a bidder can bid multiple auctions using the same fund.
-For example,
-auction A happens before auction B
-when there are two auctions.
-A bidder thus can bid all their fund for both auctions.
+a bidder can bid multiple auctions using the same amount of funds.
+For example, suppose there are two auctions, with
+auction A happening before auction B.
+A bidder thus can bid all their funds for both auctions.
 If the bidder wins auction A,
-they do not have enough fund to support their bid for auction B;
+they will not have enough funds to support their bid for auction B;
 thus, it becomes Scenario 2.
 If the bidder does not win auction A,
 they can still use the same funds to support their bid in auction B similar to Scenario 1.
@@ -303,7 +301,7 @@ \subsection{\contract Properties}
 \contract must have the fixed $T$ blocks
 as the bidding period.

-\zhq{
-\subsubsection{Ideal Auction}
-describe how zero auction achieves ideal and how c+r does not
-}
+% \zhq{
+% \subsubsection{Ideal Auction}
+% describe how zero auction achieves ideal and how c+r does not
+% }
```

0 comments on commit `fdd3df9`

Lock conversation

Write | Preview

Leave a comment

Markdown is supported | Paste, drop, or click to add files

Comment on this commit

Unsubscribe You're receiving notifications because you're watching this repository.

Commit

Add subsection of under the DE environment.

Browse files

main

haoqianzhang committed on Jan 13

1 parent fdd3df9 commit f1904bd

Showing **1 changed file** with **17 additions** and **0 deletions**.

Whitespace | Ignore whitespace | Split | Unified

17 ■■■■■ content/3-contract.tex

```
@@ -125,6 +125,23 @@ \subsection{Under the \DE Environment}
125 125   that contains the encrypted transactions.
126 126   The delay time parameter $T$ denotes the number of blocks of time
127 127   in which the transaction remain encrypted.
128 +
129 + As the \cns can not verify the transaction until
130 + its decryption and execution,
131 + the bidder could submit a bid more than their balance
132 + during the bidding period.
133 + If the bidder collects enough funds to support his bid
134 + before executing the transaction\footnote{
135 + Because the \bc delays all transactions by $T$ blocks,
136 + a bidder can, for example, send a transfer transaction to raise funds
137 + before the bidding transaction,
138 + even though, at the time of bidding,
139 + the \bc has not executed the transfer transaction.
140 + },
141 + the bidding transaction will be valid.
142 + In contrast, if, by the time of the execution,
143 + the bidder still does not have enough funds,
144 + the bidding transaction will fail in \Cref{algline:transferfunds}.
128 145   % the outputs of \contract will be delayed by, i.e. the transaction containing the original bid
      output will only appear on the blockchain at a block of height at least $T_0+T$.
129 146
130 147   %When a bidder wants to participate in this auction, the bidder needs to encrypt the entire
      original signed transaction to form a new encrypted transaction, which hides the bidder's bid.
```

0 comments on commit `f1904bd`

Lock conversation

Write | Preview

Leave a comment

Markdown is supported | Paste, drop, or click to add files

Comment on this commit

Unsubscribe You're receiving notifications because you're watching this repository.

Commit

minors

main

Browse files

haoqianzhang committed on Jan 13

1 parent 319614d commit bcff421

Showing **1 changed file** with **1 addition** and **2 deletions**.

Whitespace | Ignore whitespace | Split | Unified

3 content/2-background.tex

```
@@ -209,9 +209,8 @@ \subsection{Delayed Execution Abstraction}\label{sec:delayed}
209 209  The $T$ parameter represents the delay time (e.g., number of blocks) to delay the transaction
210 210  outputs or smart contract execution.
211 211  $T'$ denotes some total upper bound on the execution time of $\Pi$, i.e., the time it takes to
212 212  obtain the outputs of $\Pi'$ taking into account the delay time as well as the execution time of
213 213  the underlying protocol $\Pi'$.
214 214
215     - \subsection{Choosing Confirmation Time as Delay Time}
    215 + \textbf{Choosing Confirmation Time as Delay Time:}
216 216  \label{sec:choosingdelay}
217     -
218 217  In practice,
219 218  \bcs require recipients to wait for certain block confirmations
220 219  before accepting a transaction
```

0 comments on commit `bcff421`

Lock conversation

Write | Preview

Leave a comment

Markdown is supported | Paste, drop, or click to add files

Comment on this commit

Unsubscribe You're receiving notifications because you're watching this repository.

Commit

Fixed some errors.

Browse files

main

haoqianzhang committed on Jan 13

1 parent f1904bd commit 319614d

Showing **1 changed file** with **2 additions** and **2 deletions**.

Whitespace | Ignore whitespace | Split | Unified

4 main.tex

```
@@ -18,9 +18,9 @@
 \usepackage{epigraph}
 \usepackage[export]{adjustbox}
 \usepackage{enumitem}
-\usepackage{titlesec}
+%\usepackage{titlesec}

-\setcounter{secnumdepth}{3}
+%\setcounter{secnumdepth}{3}

 \newcommand{\eg}{{\em e.g.}\xspace}
 \newcommand{\ie}{{\em i.e.}\xspace}
```

0 comments on commit 319614d

Lock conversation

Write | Preview

Leave a comment

Markdown is supported | Paste, drop, or click to add files

Comment on this commit

Unsubscribe You're receiving notifications because you're watching this repository.

Commit

Update flow figure

Update flow figure

Browse files

main

haoqianzhang committed on Jan 13

1 parent bcff421 commit 2df32ae

Showing **2 changed files** with **0 additions** and **0 deletions**.

Whitespace | Ignore whitespace | Split | Unified

Filter changed files

- resources
  - flow.pdf
  - flow_figure.pptx

BIN -154 Bytes (99%) resources/flow.pdf

Binary file not shown.

BIN -29 Bytes (100%) resources/flow_figure.pptx

0 comments on commit `2df32ae`

Lock conversation

Write | Preview

Leave a comment

Markdown is supported | Paste, drop, or click to add files

Comment on this commit

Unsubscribe You're receiving notifications because you're watching this repository.

Commit

Update abstract and intro.

main

Browse files

haoqianzhang committed on Jan 14

1 parent 2df32ae commit 794e52f

Showing **2 changed files** with **6 additions** and **6 deletions**.

Whitespace | Ignore whitespace | Split | Unified

Filter changed files

- content
  - 1-introduction.tex
  - 2-background.tex

10 content/1-introduction.tex

```
@@ -3,7 +3,7 @@ \section{Introduction}
3   % Introduction of the auction on blockchain
4   The auction, an ancient way of negotiating
5   the exchange of goods and services,
6 - enters the world of blockchains and decentralised finance,
6 + enters the world of blockchains and decentralized finance
7   powered by general smart contracts~\cite{wood2014ethereum}.
8   While the \bc provides an ideal platform
9   for open auctions
@@ -19,7 +19,7 @@ \section{Introduction}
19  To implement a sealed-bids auction under a transparent blockchain,
20  auctioneers often rely on
21  a commit-and-reveal approach in which
22 - a bidder first sends a commit transaction
22 + a bidder first sends a commit transaction
23  which contains the hash of their bid
24  during the bidding phase,
25  and the bidder propagates the reveal transaction
@@ -36,7 +36,7 @@ \section{Introduction}
36  from bidders
37  during the bidding phase
38  and returns the deposit
39 - when a bidder finishes the auction honestly.
39 + when a bidder honestly finishes the auction.
40  \Cref{sec:background} provides an example of such a smart contract.
41
42  However, this approach still has a few drawbacks:
@@ -97,7 +97,7 @@ \section{Introduction}
97  as the required $T$ block confirmations
98  for all transactions
99  so that the blockchain executes and finalizes a transaction
100 - at the same time after a $T$ blocks delay.
100 + at the same time after a $T$ block delay.
101
102 We demonstrate \contract,
103 a \sa smart contract
@@ -123,7 +123,7 @@ \section{Introduction}
123 (a) a bidder only needs to interact with the blockchain once instead of two,
124 significantly reducing the latency overhead;
125 (b) \contract does not need to store the bids
126 - commitment from bidders;
126 + commitment from bidders in the smart contract;
127 (c)-(e) \contract eliminates the deposit requirement;
128 (f)-(g) the \bc guarantees the reveal of all bids
129 regardless of the bidders' behaviors and network environment.
```

2 content/2-background.tex

```
@@ -174,7 +174,7 @@ \subsection{Delayed Execution Abstraction}\label{sec:delayed}
174     In the blockchain with \de,
175     the \cns first wait for $T$ block delay
176     before decrypting and executing the \textcolor{mygreen}{green} transaction.
177 -   Both transactions have the same finalization time.}
177 +   Both transactions have the same finalization time.\zhq{Update}}
178     \label{Fig:flow}
179 \end{figure}
180
```

0 comments on commit `794e52f`

Lock conversation

Write | Preview

Leave a comment

Markdown is supported | Paste, drop, or click to add files

Comment on this commit

Unsubscribe You're receiving notifications because you're watching this repository.

Commit

Update Preliminaries section

main

haoqianzhang committed on Jan 14 — 1 parent 794e25f commit f1d04dc

Browse files

Showing **1 changed file** with **46 additions** and **31 deletions**.

Whitespace | Ignore whitespace | Split | Unified

77 content/2-background.tex

```diff
@@ -1,25 +1,37 @@
- \section{Background}
+ \section{Preliminaries}
  \label{sec:background}

- We present a brief background on the \car scheme with a \sa example and the \de abstraction.
+ In this section, we briefly introduce the properties of \sa,
+ the \car scheme with a \sa example and the \de abstraction.

- \subsection{Commit-and-Reveal Smart Contract for Sealed-bid Auctions}\label{sec:crdesc}
+ \subsection{Sealed-bid Auction Properties}

- % Before describing the \car smart contract for sealed-bid auctions,
+ We require a \sa smart contract to to satisfy at least the following properties (formal definition in~\Cref{sec:analysis:scheme}):
+ \begin{itemize}
+     \item \textbf{Hiding:} No bidder knows the bid of any other bidder during the bidding period.
+     \item \textbf{Binding:} A bidder can not change their bid
+ once the \bc finalizes the bidding transaction
+ and can pay for what they bid.
+     \item \textbf{Revealing:} All the sealed bids will be revealed during the revealing period.
+ \end{itemize}
+
+ To illustrate the benefit of delayed execution,
+ we did not consider the posterior privacy property,
+ which hides the losing bids from the public.
+ Additional Zero-Knowledge Proofs (ZKP)
+ or Multi-Party Computations (MPC) are needed
+ for \sas requiring the posterior privacy property
+ ~\cite{galal2018verifiable,galal2018succinctly,blass2018strain}.

- \zhq{Having a new subsection?}
- We require a \car scheme to satisfy the following properties (formal definition in~\Cref{sec:analysis:scheme})
- for a \sa smart contract:
- \textbf{Hiding:} No other entity except for the bidder can know the bidder's bid during the bidding period.
  % \textbf{Symbolic/cryptographic binding:} After the \bc commits a bidder's bid, it cannot be changed.
  % \textbf{Financial binding:} The bidder can always pay for the item that they bid for.
- \textbf{Binding:} A bidder can not change their bid
- once the \bc finalizes the bidding transaction
- and can pay for what they bid.
  % \textbf{Revealing:} A bidder can choose not to reveal their bid during the revealing phase at the cost of their deposit.
- \textbf{Revealing:} All the sealed bids will be revealed during the revealing period.
  %We formally define all of these properties in \Cref{sec:analysis:scheme}.

+ \subsection{Commit-and-Reveal Smart Contract for Sealed-bid Auction}\label{sec:crdesc}
+
+ % Before describing the \car smart contract for sealed-bid auctions,
+

  % \begin{algorithm}[t!]
  % \caption{Traditional commit-and-reveal}
@@ -119,11 +131,11 @@ \subsection{Commit-and-Reveal Smart Contract for Sealed-bid Auctions}\label{sec:
  the contract determines the $winner$,
  which is the bidder who submitted the highest bid and can pay for their bid.

- The \car auction smart contract
- with the cryptographic commitment scheme
- used to commit the bids
+ This \car auction smart contract
+ % with the cryptographic commitment scheme
+ % used to commit the bids
  satisfies the hiding property and
- partially assures the revealing and binding properties
+ partially assures the revealing and binding properties.
  The hiding property directly
  follows from the underlying cryptographic commitment scheme.
  % The specific hiding and symbolic/cryptographic binding properties (e.g. computational, statistical etc.) follow from the underlying cryptographic commitment scheme.
@@ -166,15 +178,16 @@ \subsection{Delayed Execution Abstraction}\label{sec:delayed}
      \centering
      \includegraphics[scale=0.45]{resources/flow.pdf}
      \caption{
-     In the \ub without \de,
+     In a \bc without \de,
      the \cns immediately execute the \textcolor{myblue}{blue} transaction
-     without any delay,
-     but requires to wait $T$ block confirmation
-     to firmly write it into the blockchain.
+     upon its commitment,
+     but recipients must wait for $T$ block confirmation
+     until its finalization.
      In the blockchain with \de,
-     the \cns first wait for $T$ block delay
-     before decrypting and executing the \textcolor{mygreen}{green} transaction.
-     Both transactions have the same finalization time.\zhq{Update}}
+     the \cns first wait for a $T$ block delay
+     before decrypting and executing the \textcolor{mygreen}{green} transaction
+     when the \bc finalizes the transaction.
+     Both transactions have the same finalization time.}
      \label{Fig:flow}
  \end{figure}

@@ -207,7 +220,7 @@ \subsection{Delayed Execution Abstraction}\label{sec:delayed}
  In the context of blockchains, $\Pi'$ could refer to the execution of transactions or smart contracts running on the underlying blockchain.
  For instance, for sealed-bid auctions, $\Pi'$ would be the traditional \car implementation as written in~\Cref{code:traditional}, and $T_0$ would denote the height of the block in the chain that contains the committed transactions.
  The $T$ parameter represents the delay time (e.g., number of blocks) to delay the transaction outputs or smart contract execution.
- $T'$ denotes some total upper bound on the execution time of $\Pi$, i.e., the time it takes to obtain the outputs of $\Pi'$ taking into account the delay time as well as the execution time of the underlying protocol $\Pi'$.
+ $T'$ denotes some total upper bound on the execution time of $\Pi$, i.e., the time it takes to obtain the outputs of $\Pi'$ taking into account the delay time as well as the execution time of the underlying protocol $\Pi'$\zhq{$\Pi?$, taking into account of the decryption time?}.

  \textbf{Choosing Confirmation Time as Delay Time:}
  \label{sec:choosingdelay}
@@ -223,19 +236,21 @@ \subsection{Delayed Execution Abstraction}\label{sec:delayed}
  we assume that
  our \ub requires $T$ block confirmation\footnote{
  $T$ can be 1 for the blockchains with instance finalization.}
- to write a transaction into the ledger firmly.
+ to finalize a transaction into the \bc.
  If we adopt the same $T$ for the delayed time,
  the \bc with delayed execution can finalize a transaction
- at a similar time as the \ub.
+ at the same time as the \ub.
  \Cref{Fig:flow} illustrates this process:
  In the \ub without \de,
- the \cns immediately execute the \textcolor{myblue}{blue} transaction without any delay,
- but then recipients need to wait for $T$ block confirmation
- until it is firmly written into the blockchain.
+ the \cns immediately execute
+ the \textcolor{myblue}{blue} transaction
+ upon its commitment,
+ but then recipients must wait for $T$ block confirmation
+ until its finalization.
  In contrast,
  for the blockchain with \de,
  the \cns first wait for the $T$ block delay
  before decrypting and executing the \textcolor{mygreen}{green} transaction
- when it has already been firmly written into the ledger.
- Therefore, both transactions a the similar finalization time.
+ when the \bc finalizes the transaction.
+ Therefore, both transactions have the same finalization time.
```

0 comments on commit f1d04dc

Lock conversation

Write | Preview

Leave a comment

Markdown is supported | Paste, drop, or click to add files

Comment on this commit

Unsubscribe — You're receiving notifications because you're watching this repository.

From: **WTSC'24** wtsc24@easychair.org
Subject: WTSC'24 submission 7
Date: January 13, 2024 at 17:33
To: Haoqian Zhang haoqian.zhang@epfl.ch

Dear authors,

We received your submission to WTSC'24 (8th International Workshop on Trusted Smart Contracts):

Authors : Haoqian Zhang, Michelle Yeo, Vero Estrada-Galiñanes and Bryan Ford
Title : ZeroAuction: Zero-Deposit Sealed-bid Auction via Delayed Execution
Number : 7

The submission was uploaded by Haoqian Zhang <haoqian.zhang@epfl.ch>.
You can access it via the WTSC'24 EasyChair Web page

https://easychair.org/conferences/?conf=wtsc24

Thank you for submitting to WTSC'24.

Best regards,
EasyChair for WTSC'24.

From: **Haoqian Zhang** haoqian.zhang@epfl.ch
Subject: WTSC23 enquiry
Date: January 14, 2024 at 15:42
To: abracciali@gmail.com, g.goodell@ucl.ac.uk

Dear Andrea and Geoffrey,

We want to resubmit our paper from the round 1. Should we modify the submission from the round 1 or create a new submission?

Best,
Haoqian

From: **WTSC'24** wtsc24@easychair.org
Subject: WTSC'24 notification for paper 2
Date: December 22, 2023 at 22:49
To: Haoqian Zhang haoqian.zhang@epfl.ch

Dear Haoqian Zhang, Michelle Yeo, Vero Estrada-Galiñanes, Bryan Ford,

Your paper entitled

ZeroAuction: Zero-Deposit Sealed-bid Auction via Delayed Execution

has not received a sufficient number of positive reviews and therefore cannot be accepted for WTSC'24 at this stage.

As you know, this was an early deadline.

All papers have been thoroughly reviewed and reviewers have suggested ways to improve your papers.

Authors are allowed to resubmit at the later deadline.

This is particularly recommended (and we would really like to encourage you to do so) for those papers that

- are borderline or have strong indications on what is missing for acceptance;

- declared authors in the paper, while this is a double blind review. We had some good papers that cannot be accepted for violation of the double-blind process but were interesting for WTSC. Please, consider suggestions and resubmit blind.

Below you find the (anonymous) peer reviews about your paper.

Many thanks for your interest in WTSC'24

Best regards,

PC chairs
Andrea Bracciali & Geoff Goodell

SUBMISSION: 2
TITLE: ZeroAuction: Zero-Deposit Sealed-bid Auction via Delayed Execution

---------- REVIEW 1 ----------
SUBMISSION: 2
TITLE: ZeroAuction: Zero-Deposit Sealed-bid Auction via Delayed Execution
AUTHORS: Haoqian Zhang, Michelle Yeo, Vero Estrada-Galiñanes and Bryan Ford

---------- Overall evaluation ----------
SCORE: -2 (reject)
----- TEXT:
The authors propose a new design of sealed auction called ZeroAuction that works on blockchains with delayed execution support. They compare the new design with the existing commit-and-reveal auction design. The authors claim that the new design like the previous design satisfies correctness, hiding, binding and revealing properties. Moreover the new design has several advantages: the bidders need to act only once; the smart contract needs fixed size of storage and it doesn't need to hold deposits.

The draft needs to address one point: the new design has a weaker version of binding property than the usual commit-and-reveal auction.

The binding property of the usual commit-and-reveal auction implies that once a bidder makes a bid with deposits, the bidder needs to accept the possibility that they actually need to pay for the bid. Since the draft assumes that the deposit is set higher than the bid, if the bidder becomes the winner, the bidder needs to pay for the bid in full from the deposit.

In the proposed new design, the bidder can deplete their own account during the waiting time between the transaction submission and the delayed execution. This way, without modifying the already submitted transaction, the bidder can escape from paying for the bid. The authors say that different auctions can have different delays, so at least the bidder can participate in an auction with a much shorter delay time, and exhaust their account before an auction with a longer delay time gets executed.

I think the draft needs to either acknowledge or mitigate the scenario I explained in the previous paragraph. The authors already consider something similar for the commit-and-reveal design (in footnote 1 on page 3).
---------- Best paper ----------
SELECTION: no

---------- REVIEW 2 ----------
SUBMISSION: 2
TITLE: ZeroAuction: Zero-Deposit Sealed-bid Auction via Delayed Execution
AUTHORS: Haoqian Zhang, Michelle Yeo, Vero Estrada-Galiñanes and Bryan Ford

---------- Overall evaluation ----------
SCORE: 1 (weak accept)
----- TEXT:
For the paper "ZeroAuction: Zero-Deposit Sealed-bid Auction via Delayed Execution," the following enhancements are necessary:

1. Lack of Proof-of-Concept Prototype Implementation and Evaluation:
 - The paper significantly lacks a section on the implementation and evaluation of a proof-of-concept prototype. This is a crucial element for demonstrating the practical application and effectiveness of the proposed ZeroAuction system. Including a prototype implementation would provide empirical evidence to support the theoretical claims made in the paper. An evaluation section should detail the performance, security, and scalability of the prototype, comparing it with existing systems. This would significantly strengthen

the paper's contribution to the field of blockchain technology and sealed-bid auctions. However, for a workshop submission, it is tolerable to omit a proof-of-concept prototype implementation and evaluation.

2. Improvement of Conclusion Section:
- The current conclusion of the paper does not follow the recommended three-part structure. To address this:
  - Part 1: The conclusion should begin with a summary of the paper, succinctly restating the main findings and contributions.
  - Part 2: As the paper currently lacks specific research questions in Section 1, it is challenging to address them in the conclusion. The paper should be revised to include clear research questions at the beginning, which the conclusion can then address, discussing how the findings respond to these questions.
  - Part 3: The conclusion should end with a discussion of the limitations of the study, open issues identified during the research, and suggestions for future work. This section should be informed by a thorough discussion section that compares the paper's results with related work.

3. General Recommendations for Enhancing the Paper:
- Introduction of Research Questions: As previously noted, Section 1 needs to include specific research questions based on a gap analysis in the current literature. This will provide a clear direction for the paper and a framework for the conclusion.
- Discussion Section: The paper should include a detailed discussion section that compares the research findings with existing literature. This section should highlight the paper's unique contributions and how it advances the field, leading to a more informed discussion of limitations and future research directions in the conclusion.
- Methodological Clarity: Ensure that the research methods used are explicitly stated and justified as appropriate for addressing the research questions. This clarity will enhance the paper's academic rigor and the validity of its conclusions.

Incorporating these suggestions will enhance the paper's structure, clarity, and academic contribution, making it a more valuable addition to the field of blockchain technology and cryptographic auctions.
---------- Best paper ----------
SELECTION: no

---------- REVIEW 3 ----------
SUBMISSION: 2
TITLE: ZeroAuction: Zero-Deposit Sealed-bid Auction via Delayed Execution
AUTHORS: Haoqian Zhang, Michelle Yeo, Vero Estrada-Galiñanes and Bryan Ford

---------- Overall evaluation ----------
SCORE: -1 (weak reject)
----- TEXT:
This paper presents a sealed-bid auction over a blockchain that aims to reduce blockchain latency during the bidding phase from two rounds under the commit-and-reveal paradigm to just one round. It also aims to remove deposits used to incentivize winning bidders to pay, and mitigate multiple bidding. The main building block is a blockchain with delayable execution.

May main objection, and the reason for reject, is that a CPA encryption scheme alone does not suffice to guarantee bid independence with one round of interaction in the bidding phase. It should be non-malleable (CCA2 security) to ensure that malicious bidders cannot maul already updated ciphertexts to the blockchain and bid based on previous bids (e.g. the second bidder can set his bid to bid_1+1 from the ciphertext of bid_1 without knowing what bid_1 is). Malleability attacks in encryption-based auction schemes is the main reason of having two rounds in the bidding phase, and the proposed solution does not address this issue.

The authors claim it is best to use threshold decryption but do not give details on the concrete scheme. All public-key threshold schemes that come to mind are homomorphic and hence subject to malleability attacks against bids independence in the one-round setting. It seems then that the best alternative is to encrypt under a CCA2-secure encryption scheme under the (trusted) auctioneer's public key, but then the solution is not novel, nor necessitates delayable execution (trust is posed on the auctioneer to not reveal bids to other bidders before the right time).

Other comments:

- Encrypting with users' keys requires one extra round of communication with the blockchain, which is what this paper tries to solve in the first place.

- Most of the computational overhead in state-of-the-art auctions lies in ensuring the auctioneer publishes the right winning bid without revealing the other bids (e.g. by employing ZKP machinery). On the contrary, in the proposed solution all bids are revealed on-chain, so I'm not convinced this is truly implementing a sealed-bid auction.

- The definition of cryptographic commitments in Section 5.1 is a bit odd. Also, the hiding property typically requires that the output distribution ensembles (indexed by the security parameter) of the commit function of any two messages are perfect/statistically/computationally indistinguishable.

- The literature review of cryptographic protocols for sealed-bid auctions is incomplete.
---------- Best paper ----------
SELECTION: no

---------- REVIEW 4 ----------
SUBMISSION: 2
TITLE: ZeroAuction: Zero-Deposit Sealed-bid Auction via Delayed Execution
AUTHORS: Haoqian Zhang, Michelle Yeo, Vero Estrada-Galiñanes and Bryan Ford

---------- Overall evaluation ----------
SCORE: 0 (borderline paper)
----- TEXT:
The paper proposes an auction format in which blockchain consensus delays revealing bids of players for some fixed amount of time, at least until the bidding process is over. The proposal satisfies several suitable properties: hiding of a bid by default, committment in that players can not change bids and revealing bids. As an additional property, the players do not interact with the "auction" twice, which would be a case in a classic commit and reveal scheme. In my opinion, this is the main contribution of the paper and the proposal, as other building blocks were already developed in other works. The authors also do not discuss deep what overhead it introduces on the protocol to enshrine such revealing scheme and how realistic it is to add such building block to already existing blockchains. The authors argue that their proposed auction saves a cost of keeping the array of committed bids, which I do not understand, as the chain validators still need to keep all transactions
in some mempool.
---------- Best paper ----------
SELECTION: no

SELECTION: no

From: **WTSC'24** wtsc24@easychair.org
Subject: WTSC'24 notification for paper 7
Date: February 6, 2024 at 18:33
To: Haoqian Zhang haoqian.zhang@epfl.ch

Dear Haoqian Zhang, Michelle Yeo, Vero Estrada-Galiñanes, Bryan Ford,

It is our pleasure to inform you that your paper entitled

ZeroAuction: Zero-Deposit Sealed-bid Auction via Delayed Execution

has been accepted at WTSC'24.

It will be included in the Springer-Verlag Lecture Notes in Computer Science (LNCS) series of the conference.

Below you find the (anonymous) peer reviews about your paper. Please, read them carefully and take them into account when submitting the final version in EasyChair.

Further details on the next steps will follow closer to the conference.

Best regards,

PC chairs
Andrea Bracciali & Geoff Goodell

SUBMISSION: 7
TITLE: ZeroAuction: Zero-Deposit Sealed-bid Auction via Delayed Execution

---------- REVIEW 1 ----------
SUBMISSION: 7
TITLE: ZeroAuction: Zero-Deposit Sealed-bid Auction via Delayed Execution
AUTHORS: Haoqian Zhang, Michelle Yeo, Vero Estrada-Galiñanes and Bryan Ford

---------- Overall evaluation ----------
SCORE: 0 (borderline paper)
----- TEXT:
I do still think the solution presented in the paper for sealed-bid auctions is not satisfactory for the following three reasons:

- Fairness: is dealt by imposing a strong and highly non practical requirement: delaying execution of all blockchain transactions. This means the functioning of the blockchain is disrupted for the sole reason of enabling an application running on it. (What if the bidders do not have funds in their account since the beginning of the auction? -- this is not enforced, as opposed to the auctions with deposits, where bidders must transfer funds to the smart contract to enter the auction.)

- What benefits brings delayed execution? Although now the authors address bid independence (malleability) via encrypting bid transactions with a KEM-like mechanism (encrypting the transaction with a symmetric key and the symmetric key with the public key of the PKE implementing the delayed execution scheme) it is unclear to me what's the gain of using delayed execution in the first place. Concretely, given delayed execution needs a trusted party (or a trusted consortium) why the solution is better than the trivial one of encrypting under an auctioner (or consortium) public key, who waits until all bids are received, and then reveals his decryption key and all bids.

- Lack of implementation: It seems to me the simple solution sketched above is the same as the one provided in the paper, but restated differently. The simple solution can be implemented right now, as opposed to delayable execution. Maybe this is the reason that no proof of concept is provided?

Having said the above, the authors have addressed some of the objections present in the first submission round, and shown how to do sealed-bid auctions over a blockchain (with strong assumptions on the model), which is a first step, at least.

Typos:
Section 2, pp. 3: to to satisfy
Defn 1. decryption of ciphertexts and outputs occurs _after_ T0 +T (?)
Footnote 2: Instant finalization (?)
---------- Best paper ----------
SELECTION: no

---------- REVIEW 2 ----------
SUBMISSION: 7
TITLE: ZeroAuction: Zero-Deposit Sealed-bid Auction via Delayed Execution
AUTHORS: Haoqian Zhang, Michelle Yeo, Vero Estrada-Galiñanes and Bryan Ford

---------- Overall evaluation ----------
SCORE: 2 (accept)
----- TEXT:
The paper has improved compared to the earlier version. The results are clearly present in the paper for an accept. My only critique is that the structure of the paper is a bit odd.
---------- Best paper ----------
SELECTION: no

---------- REVIEW 3 ----------
SUBMISSION: 7
TITLE: ZeroAuction: Zero-Deposit Sealed-bid Auction via Delayed Execution

AUTHORS: Haoqian Zhang, Ṁichelle Yeo, Vero Estrada-Galiñ́anes and Bryan Ford

---------- Overall evaluation ----------
SCORE: 2 (accept)
----- TEXT:
The authors propose a new design of sealed auction called ZeroAuction that works on blockchains with delayed execution support. They compare the new design with the existing commit-and-reveal auction design. The authors claim that the new design like the previous design satisfies correctness, hiding, binding and revealing properties. Moreover the new design has several advantages: the bidders need to act only once; the smart contract needs fixed size of storage and it doesn't need to hold deposits.

The idea of using delayed execution for implementing a sealed auction looks useful. I didn't find the flaws I found in the previous version. So I'm flagging for accept.
---------- Best paper ----------
SELECTION: no