CLOAK: Transitioning States on Legacy Blockchains Using Secure and Publicly Verifiable Off-Chain Multi-Party Computation

Qian Ren, Yingjun Wu, Han Liu, Anne Victor, Hong Lei, Lei Wang, Bangdao Chen
Background: Confidential smart contract

Pros:

- **Better confidentiality:** Private inputs are handled off-chain and are not public to all nodes.
- **Better scalability:** With the proof, all nodes can validate the correctness of the transaction outputs without re-executing it.
Motivating example: Blockchain + Supply chain finance

Let the company with the lowest bid win and pay the winner with its bid price.

Transferring money on-chain by multi-party bidding purchase off-chain
Problem definition: Multi-party Transaction

On-chain

\[ \sigma_{i+1} \]

\[ C_{x_1}, \ldots, C_{x_n}, \]

\[ C_{s_1}, \ldots, C_{s_n}, C_f \]

\[ C_{s'_1}, \ldots, C_{s'_n}, \text{proof} \]

\[ \sigma_{i+3} \]

Off-chain

\[ x_1, s_1 \]

\[ P_1 \]

\[ r_1, s'_1 \]

\[ x_n, s_n \]

\[ P_n \]

\[ r_n, s'_n \]

TX parameter: \( x \)

old states: \( s \)

TX return value: \( r \)

new states: \( s' \)

**Multi-party Transaction (MPT)**

- **Confidentiality**: An MPT requires secret inputs and states owned by different parties. All secrets should keep private to their owners.
- **Public Verifiability**: All nodes can verify the result and new state
Limitations of current solutions

Cryptography-based solutions: [CCS’19, SP20, Security’22]
- Cannot support MPC
- Suffer on inefficiency, less public verifiability, or generality of MPC
- Suffer on poor toolchain and error-prone implementation of MPC+ZKP
- Require $O(n)$ transactions to secure off-chain MPC
- …

TEE-based solutions [SP16, EUROSP19]
- Start with specified MPC settings, without considering the trusted negotiation needed by parties.
- Lack of security guarantees for off-chain interactions
- Require $O(n)$ transactions to secure off-chain MPC
- …

Existing solutions for confidential smart contracts can hardly fit the need of MPT
System model and goals

System model

- Parties ($P$)
- Blockchain ($BC$)
- Executor ($E$)
- TEE ($E$)

System goals

- **Confidentiality**: An MPT requires secret inputs and states owned by different parties. All secrets should keep private to their owners.
- **Public Verifiability**: All nodes could verify the result and new state.
- **Executor balance security**: The honest executor will never lose its deposit.
- **Financial Fairness**: Honest parties should never lose their deposits.
Challenges and countermeasures

**Challenges**

- **Byzantine resistance with O(1) cost**
  - Necessitate a low-cost punishment mechanism

- **Efficient nondeterministic negotiation**
  - Parties negotiate without knowing each other a priori

- **Secure off-chain interactions**
  - Identify and punish off-chain misbehaviors

- **Publicly verifiable proof**
  - Non-participants (e.g., Miners) can verify MPTs

**Countermeasures**

- **Deposit once, transact multi-times**

- **Nondeterministic negotiation subprotocol**
  - Negotiate off-chain, settle on-chain

- **Improved challenge-response mechanisms**
  - Challenge-response submission (resp. delivery)

- **TEE-based universal succinct proof**
Protocol overview

(Global) Setup phase: The executor and parties globally deposit coins to a TEE controlled account

(MPT) Negotiation phase: Parties interact with the TEE off-chain and commit the negotiation result on-chain

(MPT) Execution phase: The executor collects inputs from parties and blockchain to execute the MPT and get results

(MPT) Delivery phase: The executor delivers plaintext outputs, commit the MPT, and transition states on-chain
(Global) Setup phase: deposit once, transact multi-times

- **Cloak contract**
- **Blockchain**
- **Parties**
- **Executor**
- **TEE**

*register the PK and acc. of TEE*

*verify TEE*

*transfer coins to TEE’s account (e.g., 2ETH)*
(Global) Setup phase: deposit once, transact multi-times

- **Cloak contract**
- **Blockchain**
- **Parties**
  - register the PK and acc. of TEE
- **Executor**
- **TEE**
  - verify TEE
  - transfer coins to TEE’s account (e.g., 2ETH)
  - Global setup
(Global) Setup phase: deposit once, transact multi-times

TX1: locks MPT-specific deposits (e.g., 0.001ETH), notifies blockchain the MPT starts.

\[ s', r = f(s, x) \]
(Global) Setup phase: deposit once, transact multi-times

A party can concurrently join multiple MPTs as long as the sum of deposits required by joined MPTs does not exceed his coin balance in any time
(MPT) Negotiation phase: Nondeterministic negotiation subprotocol

TX1:
1. settles the MPT proposal (target function, input commitments, policy, etc.)
2. notifies the start of MPT to the blockchain
3. locks MPT-specific deposits for the executor and each party

A party can negotiate to join an MPT without knowing other parties a priori
(MPT) Execution phase: Solving repudiation of misbehaved subjects during off-chain interactions

Cloak contract

Blockchain

Parties

Executor

SGX

Nondeterministic negotiation

TX1

Locked by TEE

Trusted

Untrusted
(MPT) Execution phase: Solving repudiation of misbehaved subjects during off-chain interactions
(MPT) Execution phase: Solving repudiation of misbehaved subjects during off-chain interactions

Parties misbehave, e.g., crash, DoS, etc.
The executor misbehaves, e.g., crash, dropping.

Blockchain/TEE cannot distinguish the executor dropping the off-chain inputs from parties not submitting the off-chain inputs
(MPT) Execution phase: Challenge-response submission subprotocol
(MPT) Execution phase: Challenge-response submission subprotocol

Blockchain/TEE can identify misbehaved subjects during off-chain input submission without repudiation
(MPT) Delivery phase: Challenge-response delivery subprotocol
(MPT) Delivery phase: Validating state transition caused by an MPT
(MPT) Delivery phase: TEE-based universal succinct proof

The validation just relies on the integrity of TEE, rather the trustworthiness of parties or the executor
(MPT) Delivery phase: Validating state transition caused by an MPT
Cloak requires $O(1)$ (i.e., 2 TXs) for evaluating an MPT without an adversary, while $O(n)$ when an adversary presents.
Compare CLOAK with related works

Table 1: Comparison of CLOAK with related works. Here, ●, ⊙, ○, × denotes full, partial, not matched and not related, respectively. “Adversary Model” denotes how many entities’ misbehavior are considered, where an executor denotes a server hosting TEE. “min(#TX)” denotes how many transactions are required by the approach. “Public Verifiability” denotes all elements are committed on-chain and state transition can be validated, where $x$ denotes transaction parameter, $s, s'$ denotes contract old and new states respectively, $f$ denotes target function, $r$ denotes return value, and $\mathcal{P}$ denotes privacy policy that includes party-input bindings, etc. “Financial Fairness” denotes that honest parties never lose their collateral without obtaining outputs.

<table>
<thead>
<tr>
<th>Approach</th>
<th>Adversary Model</th>
<th>Chain Agnostic</th>
<th>min(#TX)</th>
<th>Confidentiality</th>
<th>Nondeterministic Negotiation</th>
<th>Public Verifiability</th>
<th>Financial Fairness</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ethereum [45]</td>
<td>$1^*$</td>
<td>⊗</td>
<td>$O(1)$</td>
<td>×</td>
<td>×</td>
<td>⊗ $O^2$</td>
<td>X</td>
</tr>
<tr>
<td>Ekiden [13]</td>
<td>$1^*$</td>
<td>⊙</td>
<td>$O(1)$</td>
<td>⊙</td>
<td>×</td>
<td>⊙ $O^2$</td>
<td>X</td>
</tr>
<tr>
<td>Confide [27]</td>
<td>$1^*$</td>
<td>[m$^*$/3]$^3$</td>
<td>$O(1)$</td>
<td>⊙</td>
<td>×</td>
<td>⊙ $O^2$</td>
<td>X</td>
</tr>
<tr>
<td>Hawk [25]</td>
<td>$n^*$</td>
<td>×</td>
<td>$O(n)$</td>
<td>⊙</td>
<td>⊙</td>
<td>⊙ $O^2$</td>
<td>X</td>
</tr>
<tr>
<td>ZEXE [7]</td>
<td>$n^*$</td>
<td>1$^*$</td>
<td>$O(1)$</td>
<td>⊙</td>
<td>⊙</td>
<td>⊙ $O^2$</td>
<td>X</td>
</tr>
<tr>
<td>Fastkitten [16]</td>
<td>(n$^* + 1^*$) - 1</td>
<td>⊙</td>
<td>$O(n)$</td>
<td>⊙</td>
<td>⊙</td>
<td>⊙ $O^2$</td>
<td>X</td>
</tr>
<tr>
<td>LucidiTEE [32]</td>
<td>m$^*$</td>
<td>1$^*$</td>
<td>$O(n)$</td>
<td>⊙</td>
<td>⊙</td>
<td>⊙ $O^2$</td>
<td>X</td>
</tr>
<tr>
<td>CLOAK</td>
<td>(n$^* + 1^*$) - 1$^6$</td>
<td>⊙</td>
<td>$O(1)$</td>
<td>⊙</td>
<td>⊙</td>
<td>⊙ $O^2$</td>
<td>X</td>
</tr>
</tbody>
</table>

- **Require at least one is honest**
- **Only 2 TX in normal cases**
- **Firstly**
- **Most general**
- **Adversary will be identified and punished**
The gas cost of Cloak reduces by 32.4% on average. As the number of parties grows, the efficiency of Cloak on gas cost stands out.
CLOAK: Transitioning States on Legacy Blockchains Using Secure and Publicly Verifiable Off-Chain Multi-Party Computation

Qian Ren, Yingjun Wu, Han Liu, Anne Victor, Hong Lei, Lei Wang, Bangdao Chen

Questions?