DeSO: Decentralized Secret Oracle for Blockchain

01

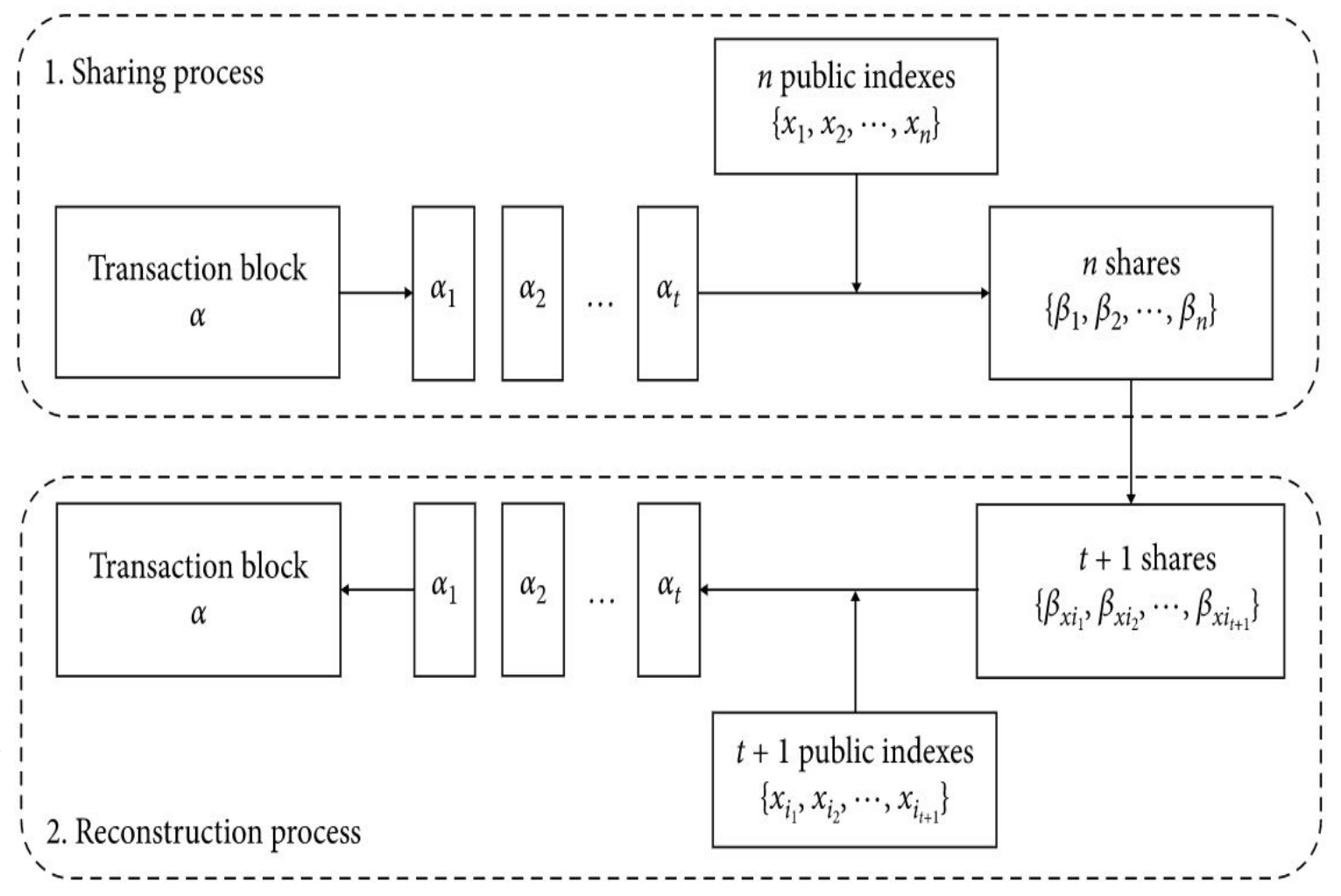
STUDENTS: Milad Fotouhi

Why DeSO ?

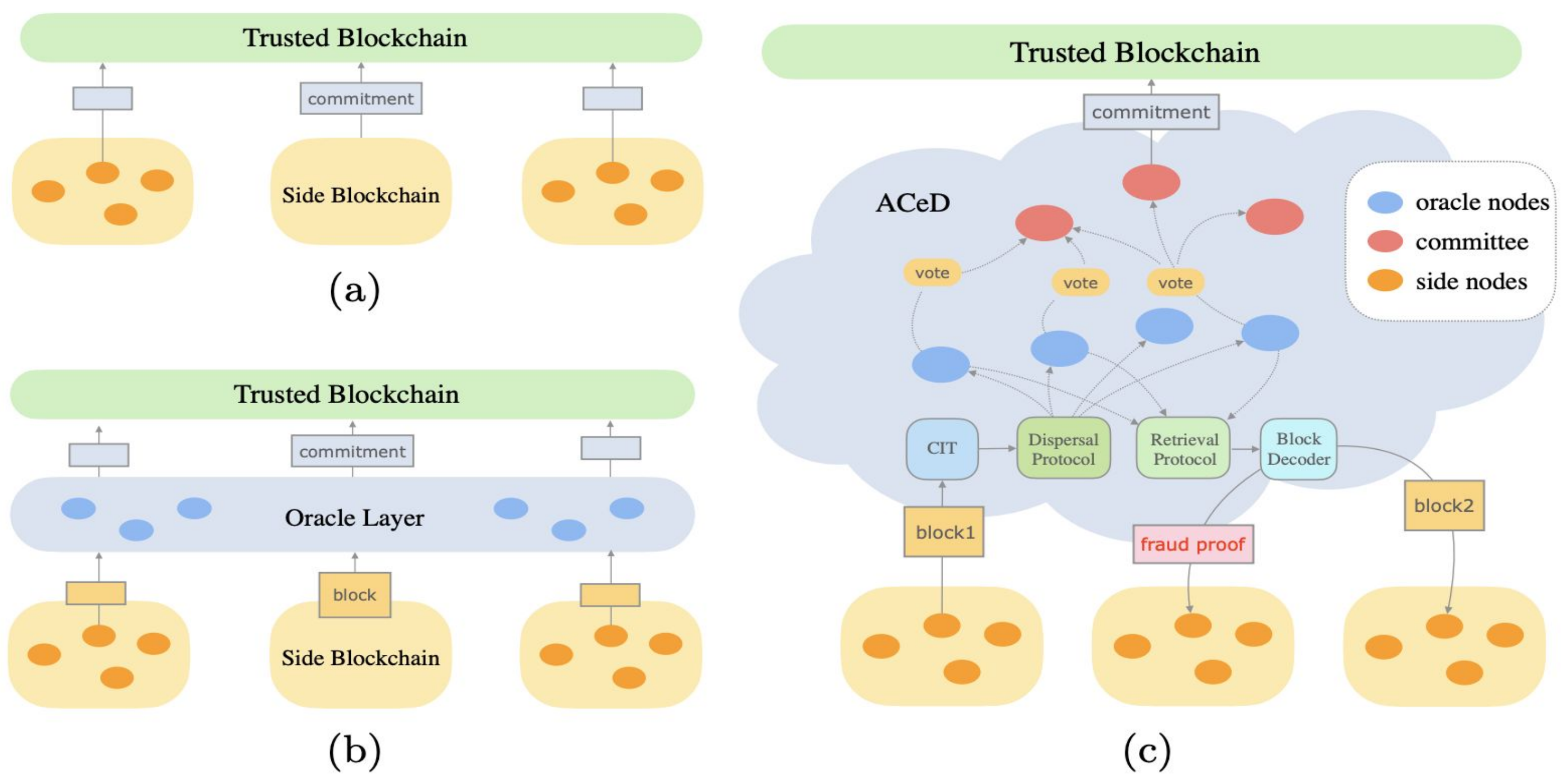
- TIn disaster management, medical records may be considered secret unless there is a disaster in which it can be released only to the appropriate authorities so that medicines can be availed in time.
- In this class of problems, decentralization plays two roles:
  - Distributed storage of the secret till the condition is met
  - And a distributed checking of whether the condition is met
- The latter is easily enabled via standard smart contracts in a blockchain, however, an appropriate solution is needed for the former problem, i.e., distributed storage of the secret.
- The main challenge is that, on the one hand, scalability requires that we use small committees to represent the entire system, but, on the other hand, a mobile adversary may be able to corrupt the entire committee if it is small. For this reason, existing proactive secret sharing solutions are either non-scalable or insecure in our setting.

Decentralized Blockchain

- Blockchain systems establish a cryptographically secure data structure for storing data in the form of a hash chain.
- Smart contracts on a blockchain permit the performance of credible transactions without the involvement or oversight of a third party. However, in a smart contract, information needs to be known and independently verified.
- Shamir's secret sharing scheme can be used to obscure a secret and reveal it only when the majority of the parties in the contract are present.
- In this project we introduce, discuss and simulate a consensus-based secret sharing protocol on smart contracts with the goal of depositing a secret with the blockchain, having the blockchain keep the secret and use it only in the specified manner (only when a specified number of members "vote" to reveal the secret).
- We implement our protocol with full functionality in 7,500 lines of Rust code, integrate the functionality as a smart contract into Ethereum via a high-performance implementation demonstrating up to 10,000 transactions per second in throughput and 6,000x reduction in gas cost on the Ethereum test-net Kovan.


1. Sharing process
n public indexes
$\{x_1, x_2, \cdots, x_n\}$
Transaction block
α
α_1
α_2
…
α_t
n shares
$\{\beta_1, \beta_2, \cdots, \beta_n\}$
Transaction block
α
α_1
α_2
…
α_t
t + 1 shares
$\{\beta_{xi_1}, \beta_{xi_2}, \cdots, \beta_{xi_{t+1}}\}$
t + 1 public indexes
$\{x_{i_1}, x_{i_2}, \cdots, x_{i_{t+1}}\}$
2. Reconstruction process

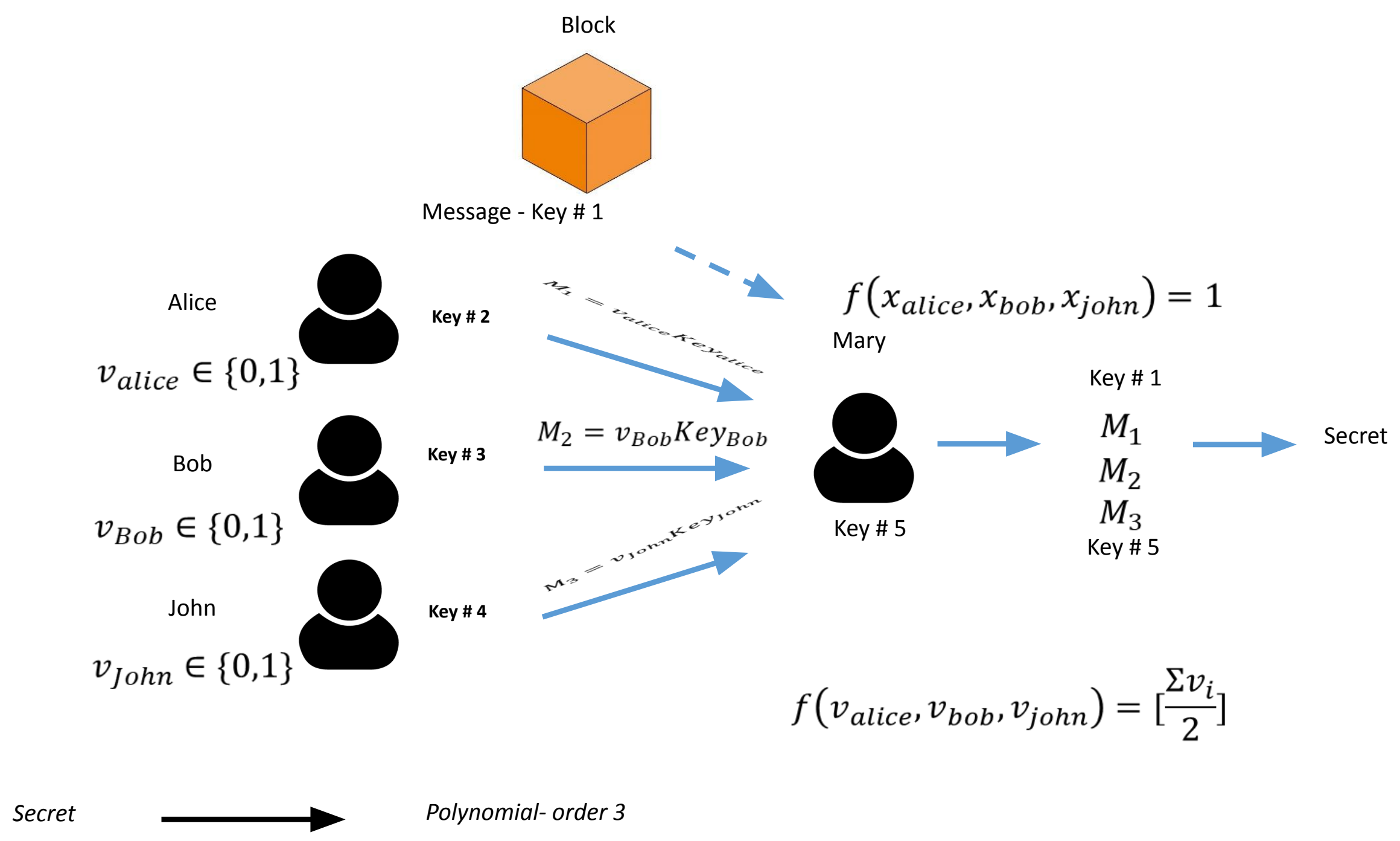
Etherium As a Secure Mechanism to Store Data



- MEthereum has demonstrated itself to be secure in practice but at the expense of poor performance.
- We have decided to implement and design a smart contract into Ethereum in order to have a high performance blockchain without sacrificing security.
- We propose an intermediate "data availability oracle" layer that interfaces between the side blockchains and the trusted blockchain. This oracle layer accepts blocks form side blockchains and pushes verifiable commitments to the trusted blockchain.

Shamir Secret Sharing

- 1- Each party or a group of them cannot be able to open the secret independently, unless the message is issued in the blockchain.
- 2- This must happen only by consensual agreement from the parties, and the secret must be only opened to a third party, when the message is issued!
- 3- Perfect secrecy must still stand!

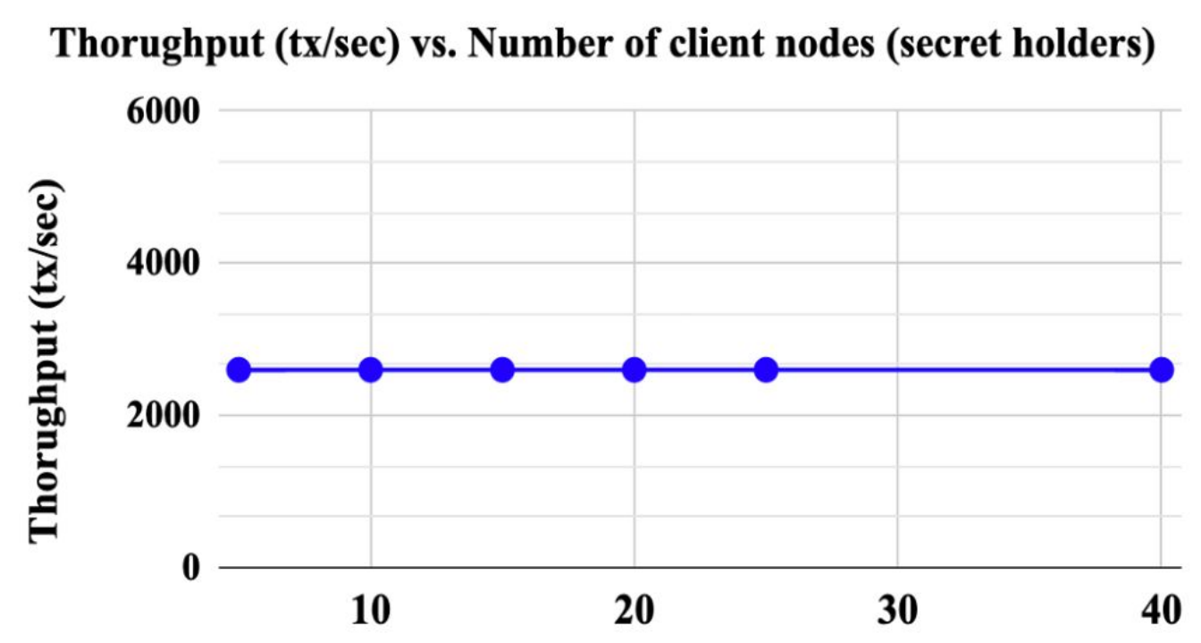


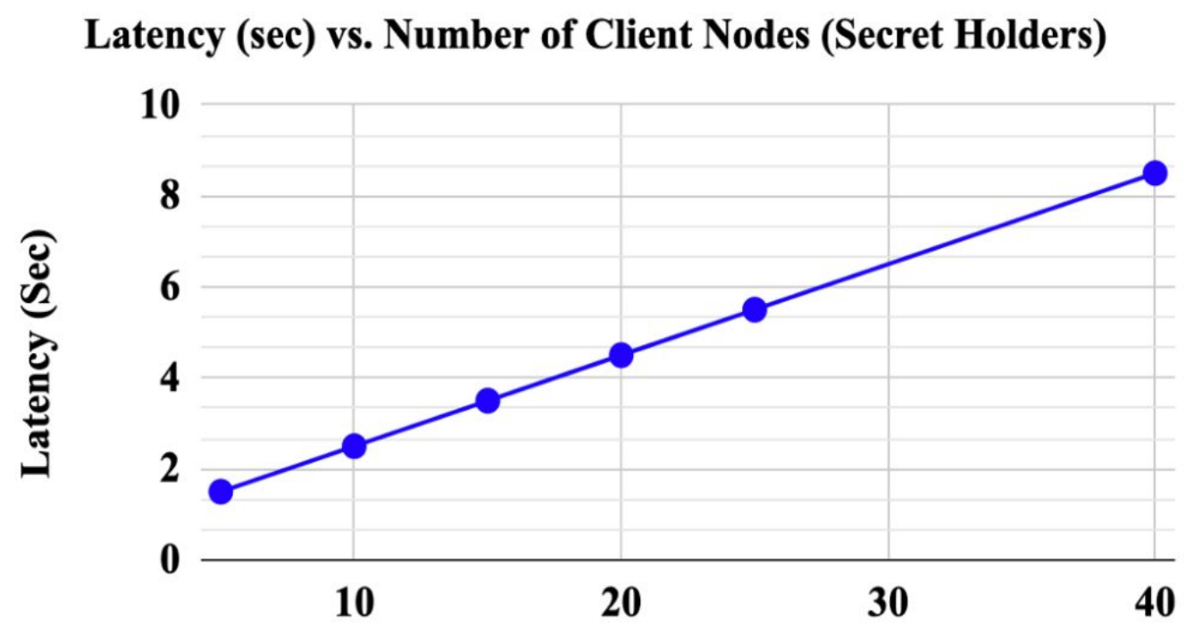
Properties of DeSO

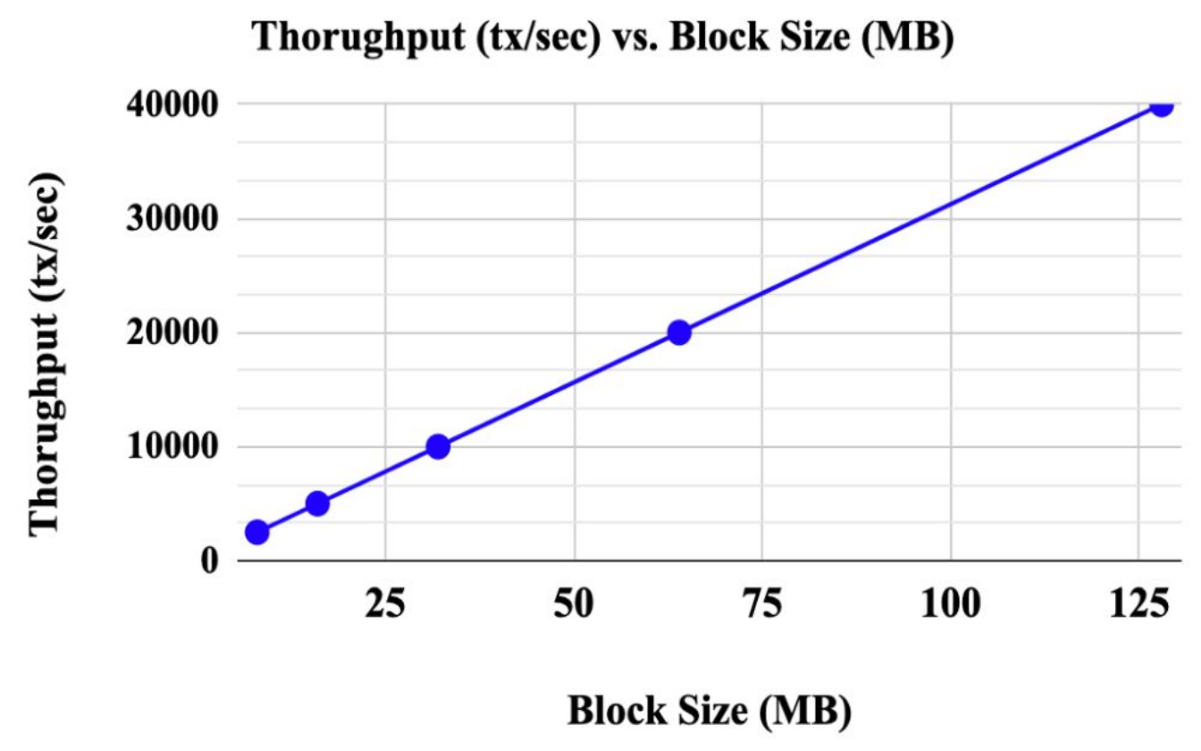
DeSO uses a data availability oracle layer that interfaces between the client nodes and the trusted blockchain. DeSO uses SSS in order to create secret shares and distribute it between client nodes. The oracle layer accepts secret share blocks from clients, pushes verifiable commitments to the trusted blockchain and ensures data availability to the client nodes.

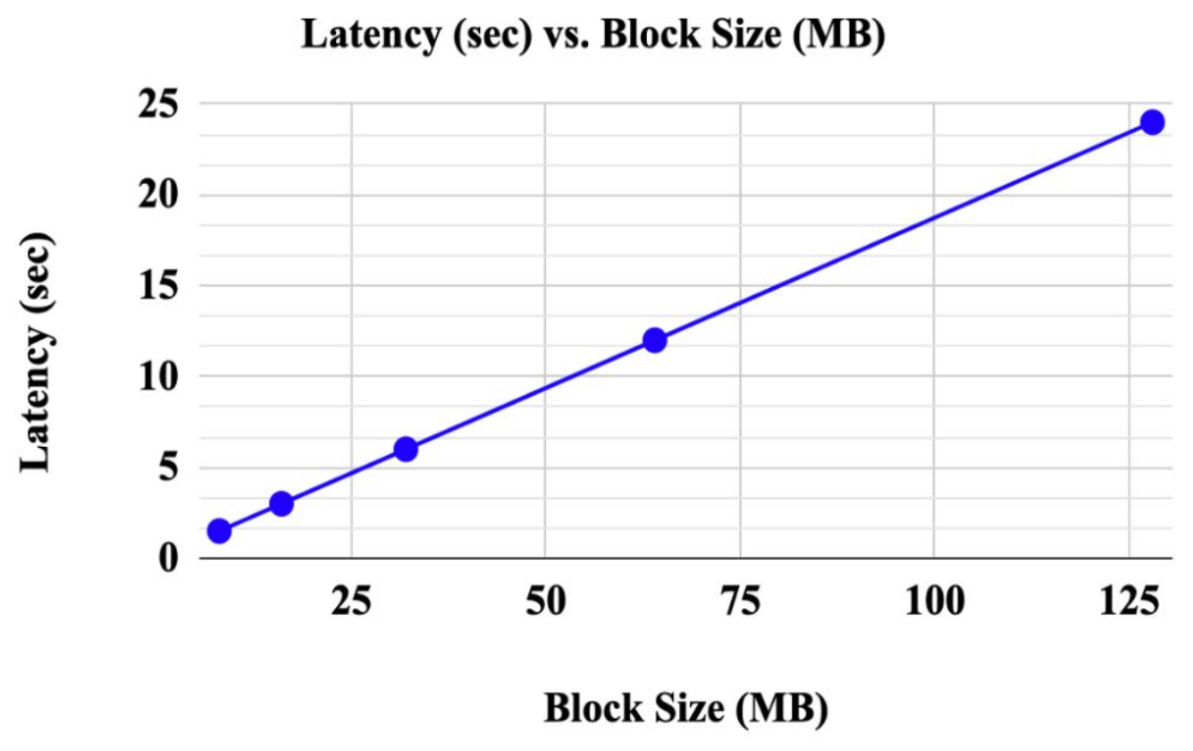
- Secure: Information-theoretic security.
- Minimal: The size of each secret share does not exceed the size of the original data.
- Extensible: When number of secret holders is kept fixed, new parties can be dynamically added or deleted without affecting the other pieces.
- Dynamic: Security can be easily enhanced without changing the secret, but by only constructing new secret shares.
- Flexible: In hierarchical organizations, we can supply each participant with different numbers of shares according to their importance. This weighting scheme allows higher ranking participants to be allocated a larger number of shares.

Future Work, References, and Acknowledgments









- We have proposed and and developed DeOS as a solution to allow a public blockchain to act as a trusted long-term repository of secrets.
- We have shown how Shamir's secret sharing scheme can be used to obscure a secret when the majority of the parties in the contract are presented.
- We have Successfully deposit a secret with the blockchain and have the blockchain keep the secret and use it only in the specified manner (only when a specified number of members "vote" to reveal the secret).
- We have shown that our service can be used in disaster management, where medical records may be considered secret and these secrets needs to be released at the disaster time only to the appropriate authorities so that medicines can be availed in time.

ELECTRICAL & COMPUTER ENGINEERING
UNIVERSITY *of* WASHINGTON

ADVISORS: Sreeram Kannan, Payman Arabshahi, Wei Cheng

SPONSORS: National Science Foundation