An Offline Delegatable Cryptocurrency System

Rujia Li[1,2] , Qin Wang[3,4] , Xinrui Zhang[5],
Qi Wang[1], David Galindo[2], Yang Xiang[3]

1 Southern University of Science and Technology, Shenzhen, China
2 University of Birmingham, Birmingham, United Kingdom
3 Swinburne University of Technology, Melbourne, Australia
4 CSIRO Data61, Sydney, Australia.
5 Nankai University, Tianjin, China

May 2021

IEEE ICBC 21

Cryptocurrency System

- Cryptocurrencies facilitate the convenience of payment.
- Online processing of transactions confronts the problems of low performance and high congestion.

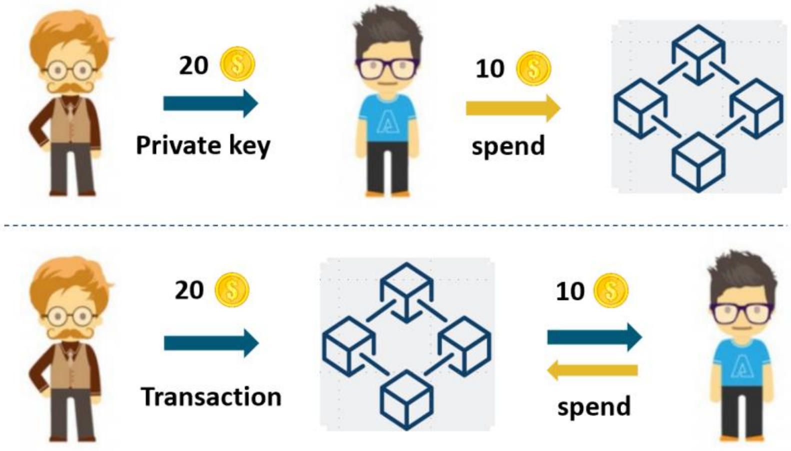
Cryptocurrency Delegation

- Delegation enables users to exchange the coin *without* having to connect to an online blockchain platform.

- Delegation confronts risks caused by unreliable participants.

- The misbehaviours may easily happen due to *the absence of effective supervision*.

Delegation Example

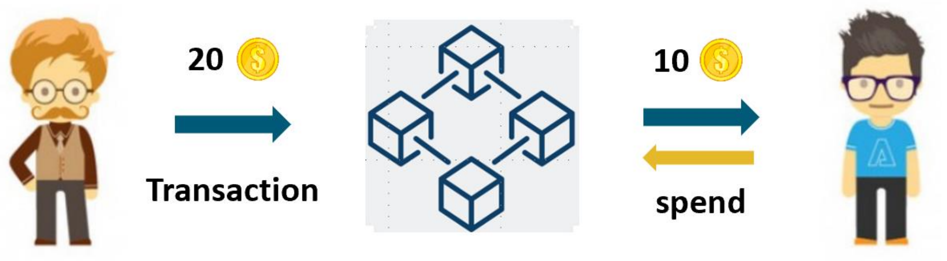


- **Coin-Transfer.** Alex asks for Bob's BTC address, and then *transfers a specific amount of coins* to Bob's address.

- **Ownership-Transfer.** Alex directly *gives his own private key to Bob*. Then, Bob can freely spend the coins using such a private key.

Delegation Drawbacks

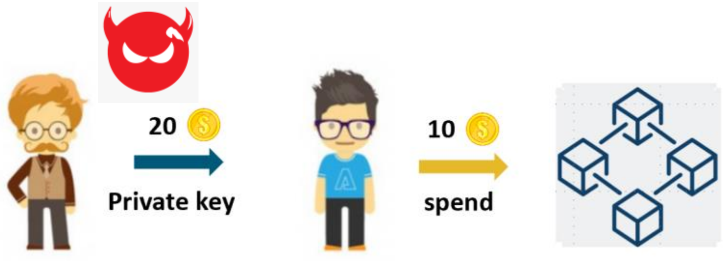
Coin-Transfer



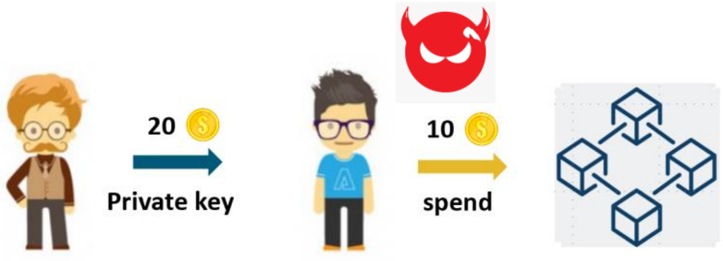
Coin-transfer requires *a strict consistency (global view)* of the blockchain, which makes it time-consuming.

Delegation Drawbacks

Ownership-Transfer.



A malicious coin owner could spend the delegated transaction before the delegate uses it.



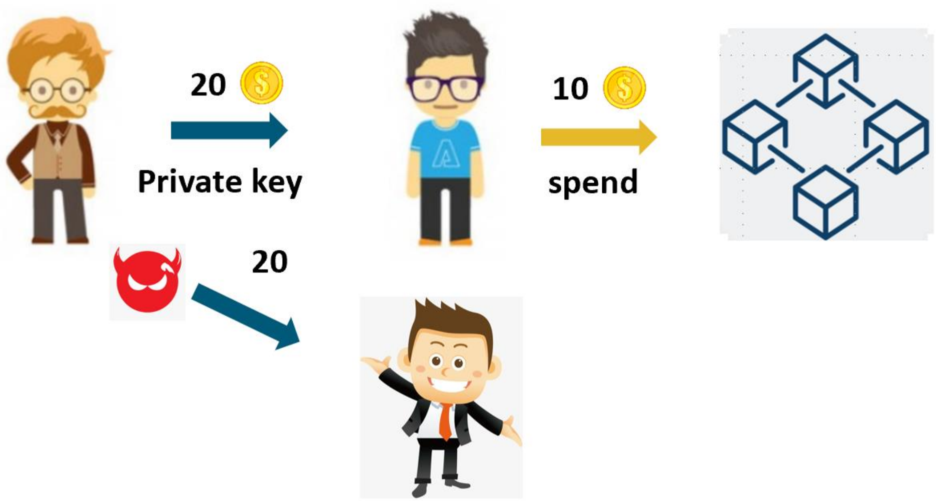
A malicious delegatee may spend all coins in the address for other purposes.

Research Problem

Is it possible to build a secure offline peer-to-peer delegatable system for decentralized cryptocurrencies?

Challenges

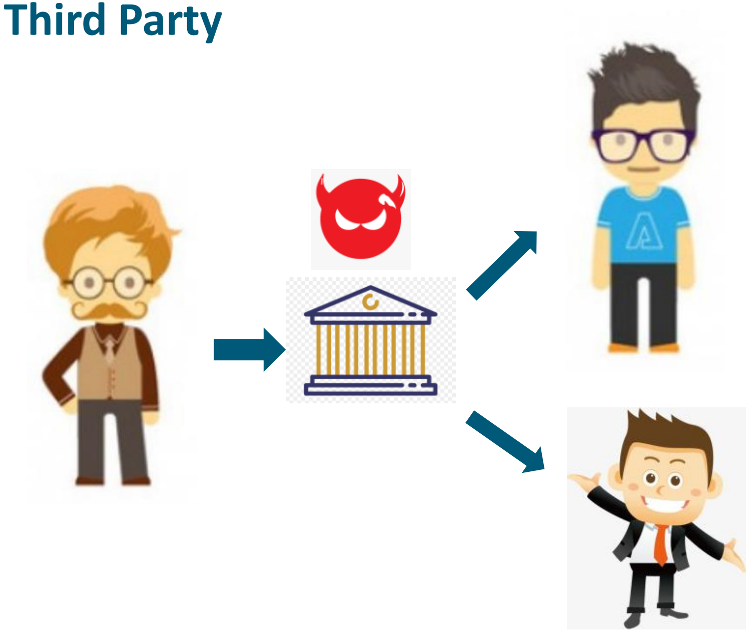
Without A Third Party



The coin might be spent twice after another successful delegation.

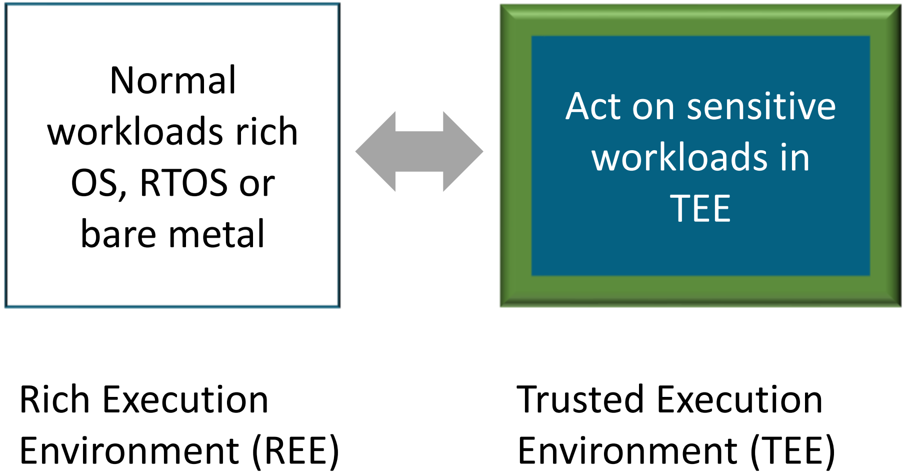
Challenges

With A Third Party



- ➢ The approach with a third party is centralized.
- ➢ The third party faces the threat of being compromised or provided with misleading assure.

TEEs Background



Normal workloads rich OS, RTOS or bare metal

Act on sensitive workloads in TEE

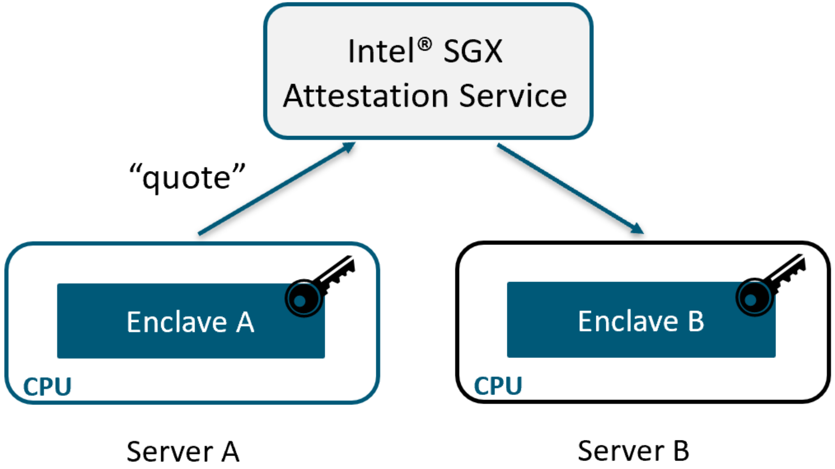
Rich Execution Environment (REE)

Trusted Execution Environment (TEE)

TEE implementation: *TrustZone®, SGX®*

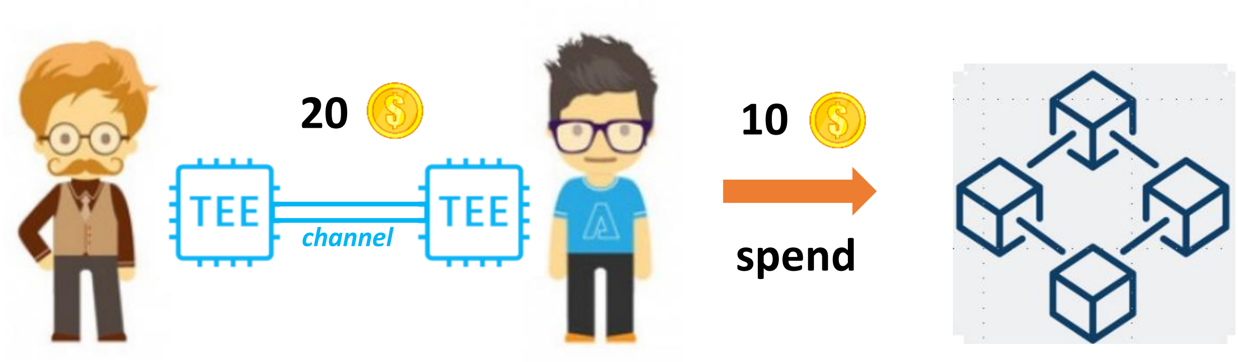
- ➢ Sealing Technology
- ➢ Local Attestation.
- ➢ Remote Attestation.

Remote Attestation



A remote party can verify whether a piece of code is running in an enclave of the Intel SGX platform.

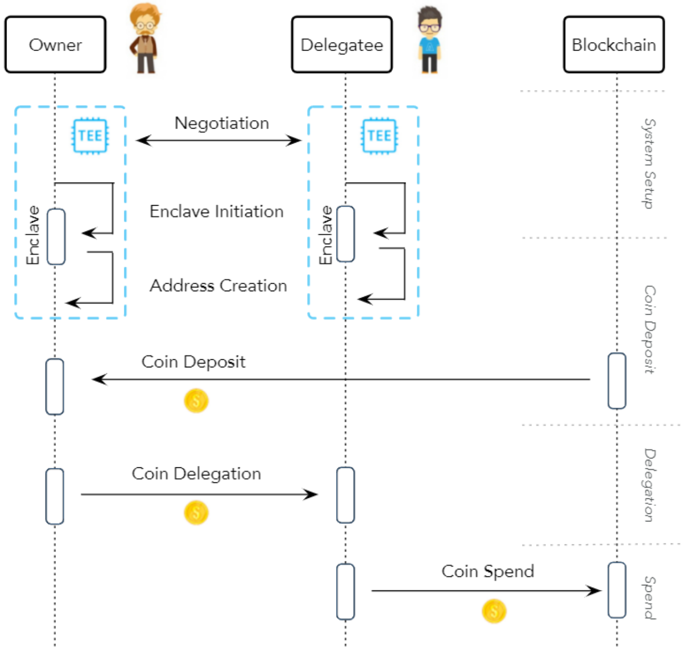
Our Solution



- ➢ The enclaves are as trusted agents between the coin owner and coin delegatee.

- ➢ Each coin owner has his own enclave. The agents are decentralized.

System Overview

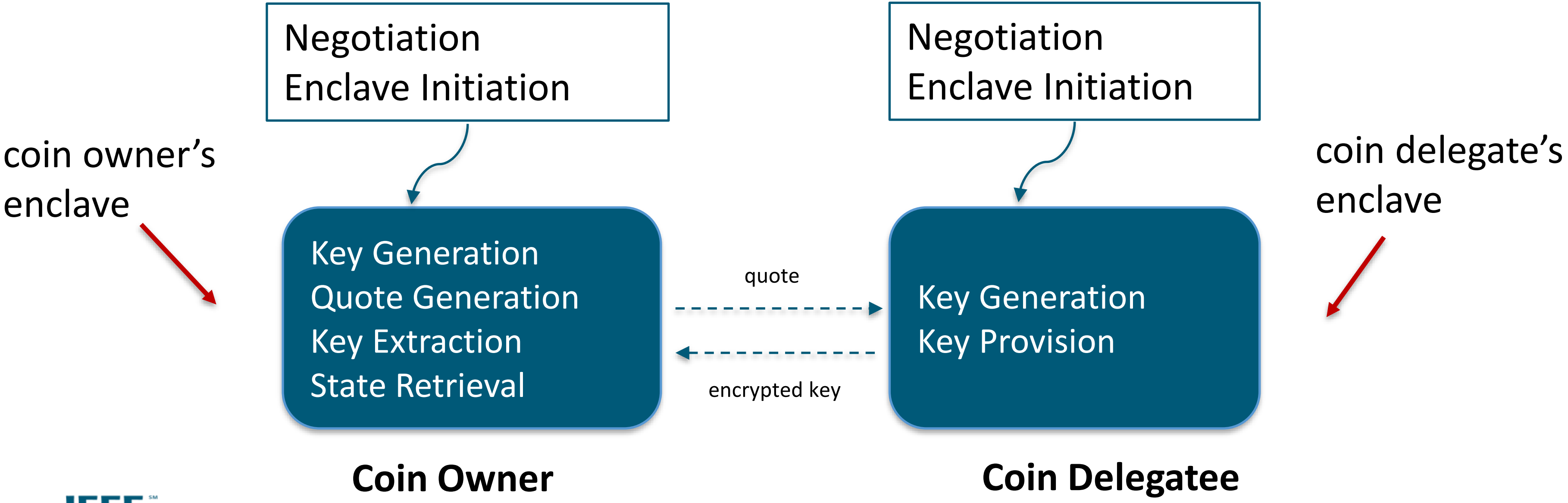


- ➢ System Setup
- ➢ Coin Deposit
- ➢ Coin Delegation
- ➢ Coin Spend

The TEEs are as decentralized trusted agents.

System Setup

In this phase, the coin owner O and the delegatee D initialize their TEEs to provide environments for the operations with respect to the further delegation.



Negotiation
Enclave Initiation

coin owner's enclave

Key Generation
Quote Generation
Key Extraction
State Retrieval

Coin Owner

quote

encrypted key

Negotiation
Enclave Initiation

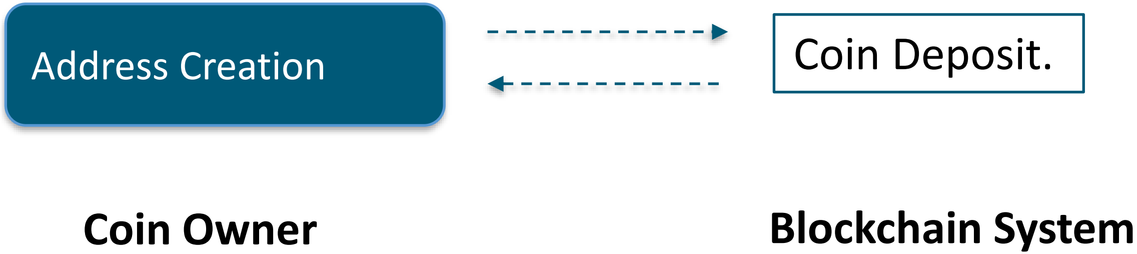
coin delegate's enclave

Key Generation
Key Provision

Coin Delegatee

Coin Deposit

The coin owner O generates an address and its corresponding private key. Afterwards, O sends coins to this address in the form of fund deposits.



Address Creation

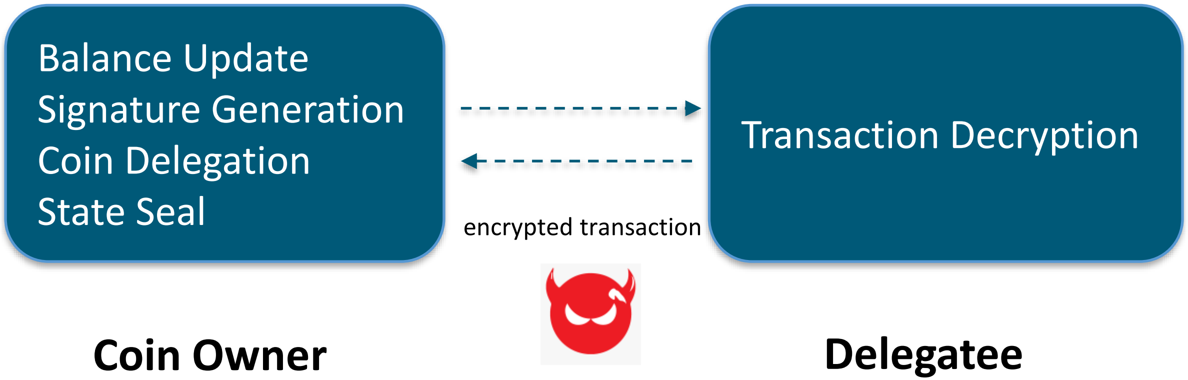
Coin Deposit.

Coin Owner

Blockchain System

Coin Delegation

In this phase, neither O nor D interacts with blockchain. O can instantly complete the coin delegation through offline transactions.



Balance Update
Signature Generation
Coin Delegation
State Seal

Transaction Decryption

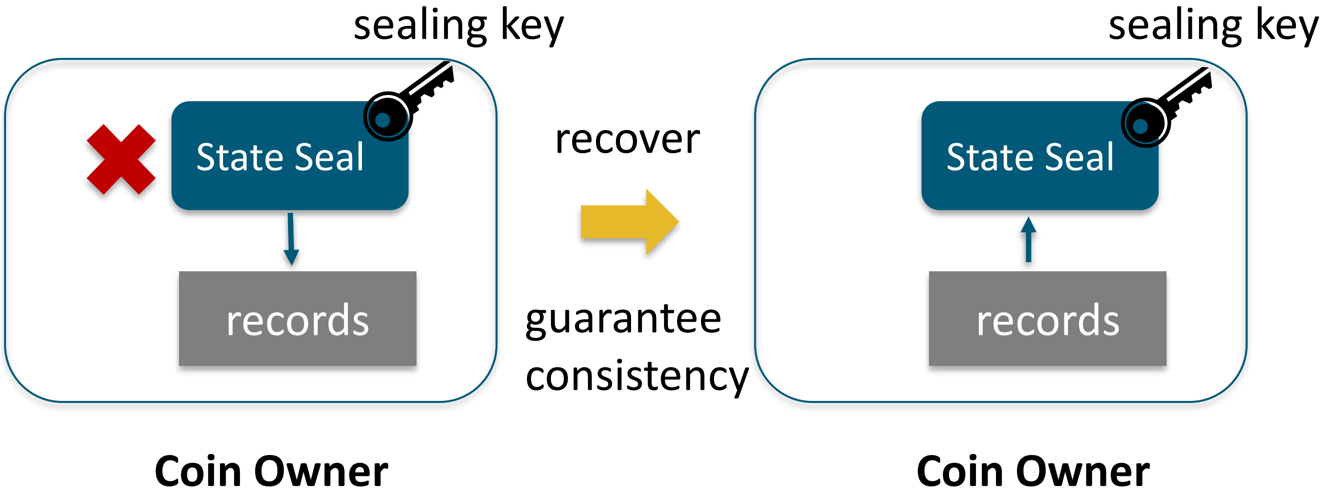
encrypted transaction

Coin Owner

Delegatee

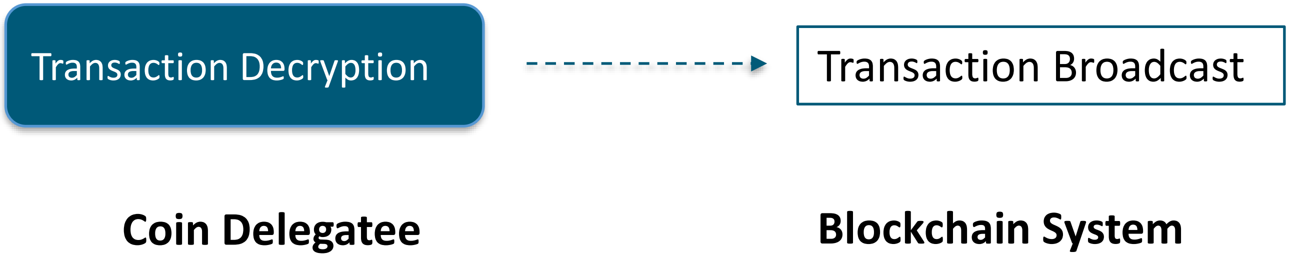
Coin Delegation

If any abort or halt happens, a re-initiated enclave starts to reload the missing information.



Coin Spend

The delegatee decrypts the encrypted transaction, and then spends coins by forwarding the transaction to the blockchain network.



Transaction Decryption

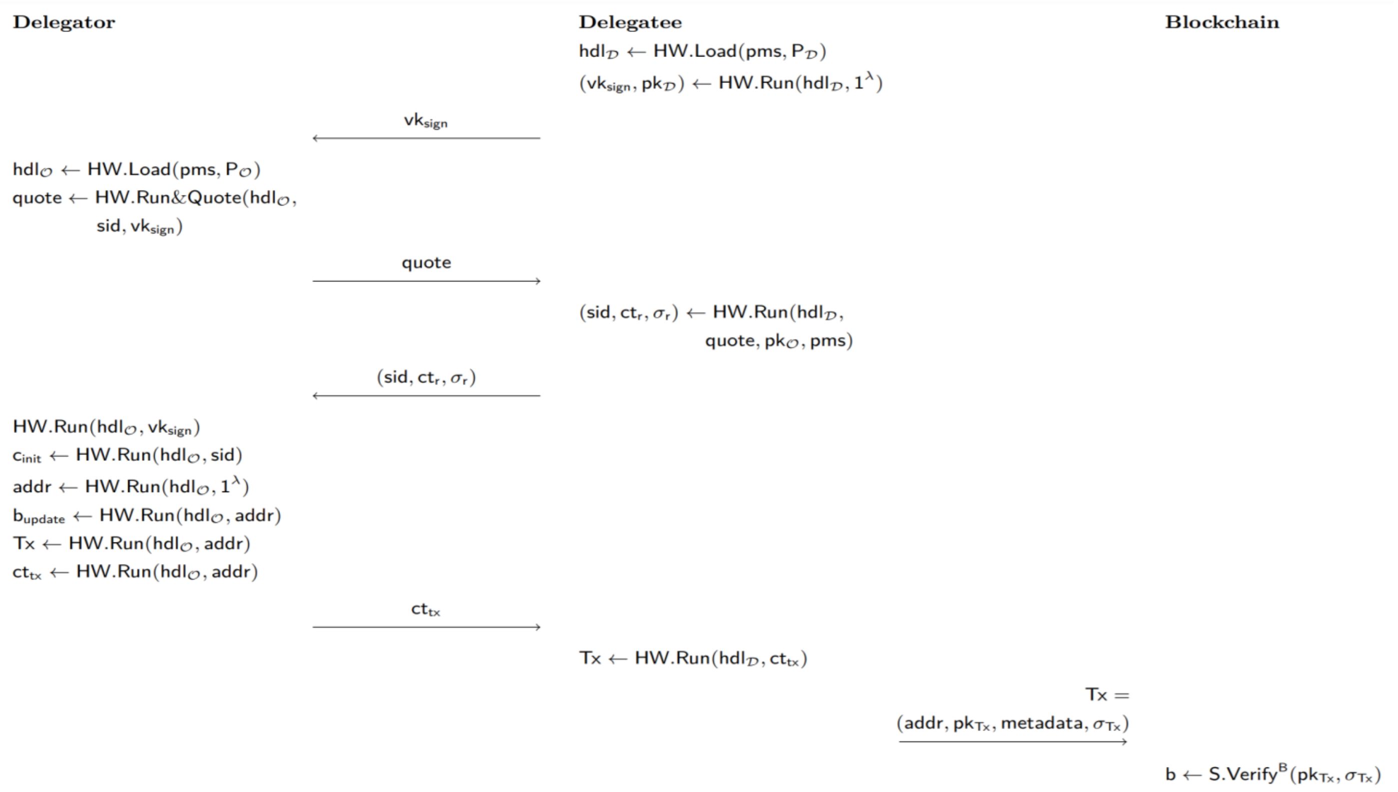
Transaction Broadcast

Coin Delegatee

Blockchain System

Formal Treatment

- TEEs are treated as black-box programs
- Simulation based approach to capture the security



Delegator **Delegatee** **Blockchain**

$\mathsf{hdl}_{\mathcal{D}} \leftarrow \mathsf{HW.Load}(\mathsf{pms}, \mathsf{P}_{\mathcal{D}})$

$(\mathsf{vk}_{\mathsf{sign}}, \mathsf{pk}_{\mathcal{D}}) \leftarrow \mathsf{HW.Run}(\mathsf{hdl}_{\mathcal{D}}, 1^{\lambda})$

$\xleftarrow{\mathsf{vk}_{\mathsf{sign}}}$

$\mathsf{hdl}_{\mathcal{O}} \leftarrow \mathsf{HW.Load}(\mathsf{pms}, \mathsf{P}_{\mathcal{O}})$

$\mathsf{quote} \leftarrow \mathsf{HW.Run\&Quote}(\mathsf{hdl}_{\mathcal{O}}, \mathsf{sid}, \mathsf{vk}_{\mathsf{sign}})$

$\xrightarrow{\mathsf{quote}}$

$(\mathsf{sid}, \mathsf{ct}_{\mathsf{r}}, \sigma_{\mathsf{r}}) \leftarrow \mathsf{HW.Run}(\mathsf{hdl}_{\mathcal{D}}, \mathsf{quote}, \mathsf{pk}_{\mathcal{O}}, \mathsf{pms})$

$\xleftarrow{(\mathsf{sid}, \mathsf{ct}_{\mathsf{r}}, \sigma_{\mathsf{r}})}$

$\mathsf{HW.Run}(\mathsf{hdl}_{\mathcal{O}}, \mathsf{vk}_{\mathsf{sign}})$

$\mathsf{c}_{\mathsf{init}} \leftarrow \mathsf{HW.Run}(\mathsf{hdl}_{\mathcal{O}}, \mathsf{sid})$

$\mathsf{addr} \leftarrow \mathsf{HW.Run}(\mathsf{hdl}_{\mathcal{O}}, 1^{\lambda})$

$\mathsf{b}_{\mathsf{update}} \leftarrow \mathsf{HW.Run}(\mathsf{hdl}_{\mathcal{O}}, \mathsf{addr})$

$\mathsf{Tx} \leftarrow \mathsf{HW.Run}(\mathsf{hdl}_{\mathcal{O}}, \mathsf{addr})$

$\mathsf{ct}_{\mathsf{tx}} \leftarrow \mathsf{HW.Run}(\mathsf{hdl}_{\mathcal{O}}, \mathsf{addr})$

$\xrightarrow{\mathsf{ct}_{\mathsf{tx}}}$

$\mathsf{Tx} \leftarrow \mathsf{HW.Run}(\mathsf{hdl}_{\mathcal{D}}, \mathsf{ct}_{\mathsf{tx}})$

$\xrightarrow{\mathsf{Tx} = (\mathsf{addr}, \mathsf{pk}_{\mathsf{Tx}}, \mathsf{metadata}, \sigma_{\mathsf{Tx}})}$

$\mathsf{b} \leftarrow \mathsf{S.Verify}^{\mathsf{B}}(\mathsf{pk}_{\mathsf{Tx}}, \sigma_{\mathsf{Tx}})$

Security Discussion

- The *private key* of a delegated transaction and the delegated transaction itself *are protected against the public.*

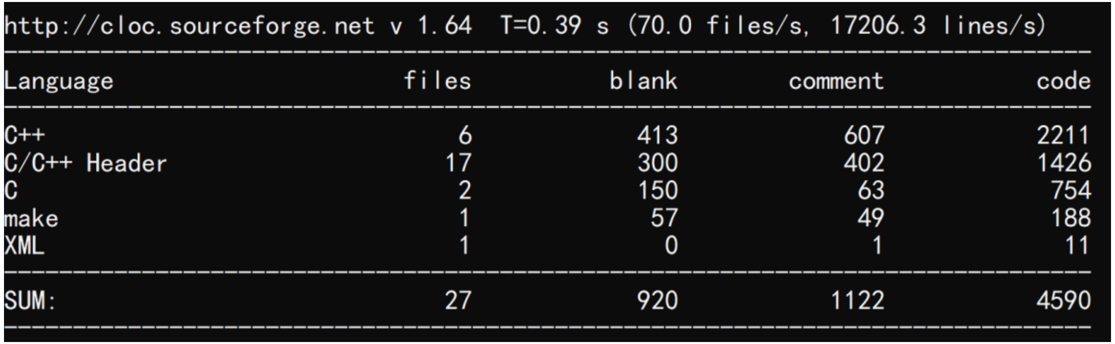
- The spendable amount of delegated coins must be *less than (or equal to)* original coins.

- The delegation *records are securely stored* to guarantee consistency considering accidental TEEs failures or malicious TEEs compromises.

Implementation

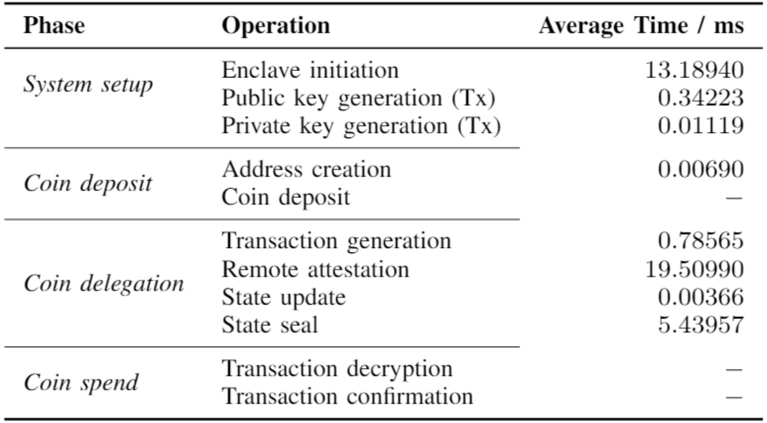
- C++
- Intel SGX SDK 1.6
- Ubuntu 20.04.1 LTS
- Bitcoin testnet
- SHA-256, ECDSA with secp256k1

Implementation codes are available at:
https://github.com/TEEs-projects/DelegaCoin

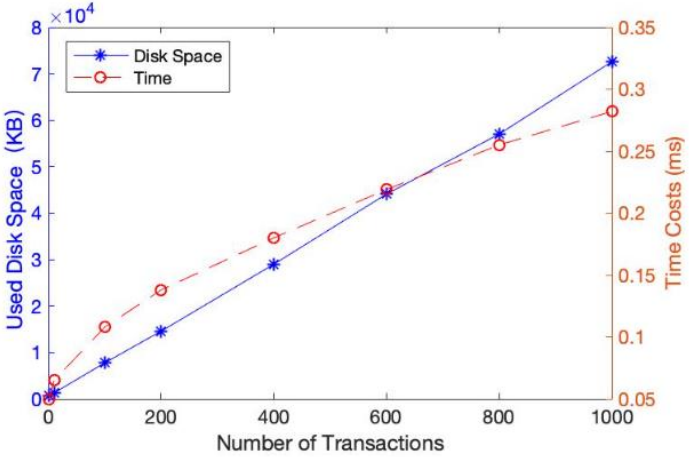


```
http://cloc.sourceforge.net v 1.64  T=0.39 s (70.0 files/s, 17206.3 lines/s)
-------------------------------------------------------------------------------
Language                     files          blank        comment           code
-------------------------------------------------------------------------------
C++                              6            413            607           2211
C/C++ Header                    17            300            402           1426
C                                2            150             63            754
make                             1             57             49            188
XML                              1              0              1             11
-------------------------------------------------------------------------------
SUM:                            27            920           1122           4590
-------------------------------------------------------------------------------
```

Evaluation



Phase	Operation	Average Time / ms
System setup	Enclave initiation	13.18940
	Public key generation (Tx)	0.34223
	Private key generation (Tx)	0.01119
Coin deposit	Address creation	0.00690
	Coin deposit	–
Coin delegation	Transaction generation	0.78565
	Remote attestation	19.50990
	State update	0.00366
	State seal	5.43957
Coin spend	Transaction decryption	–
	Transaction confirmation	–



Performance

Disk space

Summary

- Identify the challenge of current decentralized delegation
- Propose an offline delegatable payment solution
- Formally define our protocols with security analysis
- Implement the system with Intel’s SGX
- Conduct a series of experiments

References

- Gavin Wood et al. Ethereum: A secure decentralised generalised transaction ledger. In Ethereum Project Yellow Paper, 2014
- Muneeb Ali, Jude Nelson, Ryan Shea, and Michael J Freedman. Blockstack: A global naming and storage system secured by blockchains. In 2016 USENIX Annual Technical Conference (USENIX ATC), pages 181–194, 2016.
- Ben Fisch, Dhinakaran Vinayagamurthy, Dan Boneh, and Sergey Gorbunov. Iron: functional encryption using intel sgx. In Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security (CCS), pages 765–782, 2017.
- Jan-Erik Ekberg, Kari Kostiainen, and N Asokan. Trusted execution environments on mobile devices. In Proceedings of the 2013 ACM SIGSAC conference on Computer and Communications Security (CCS), pages 1497–1498, 2013.
- Dayeol Lee, David Kohlbrenner, Shweta Shinde, Krste Asanovic, and ´ Dawn Song. Keystone: An open framework for architecting trusted execution environments. In Proceedings of the Fifteenth European Conference on Computer Systems (EuroSys), pages 1–16, 2020.
- Sinisa Matetic, Moritz Schneider, Andrew Miller, Ari Juels, and Srdjan Capkun. Delegatee: Brokered delegation using trusted execution environments. In 27th fUSENIXg Security Symposium (USENIX Security), pages 1387–1403, 2018.
- Victor Costan and Srinivas Devadas. Intel sgx explained. IACR Cryptology ePrint Archive, 2016(086):1–118, 2016.
- David Bernhard, Véronique Cortier, David Galindo, Olivier Pereira, and Bogdan Warinschi. SoK: A comprehensive analysis of game-based ballot privacy definitions. In IEEE Symposium on Security and Privacy (SP), pages 499–516. IEEE, 2015.
- Kiffer, Lucianna, Rajmohan Rajaraman, and Abhi Shelat. “A better method to analyze blockchain consistency.” Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security. 2018.
- Bitcoin testnet. In https://coinfaucet.eu/en/btc-testnet/, 2020.

Thanks

An Offline Delegatable Cryptocurrency System

Rujia Li[1,2] , Qin Wang[3,4] , Xinrui Zhang[5],
Qi Wang[1], David Galindo[2], Yang Xiang[3]

1 Southern University of Science and Technology, Shenzhen, China
2 University of Birmingham, Birmingham, United Kingdom
3 Swinburne University of Technology, Melbourne, Australia
4 CSIRO Data61, Sydney, Australia.
5 Nankai University, Tianjin, China

May 2021