1

Facilitating Security and Trust among Multiple Parties through Blockchain Techniques

Yuhong Liu

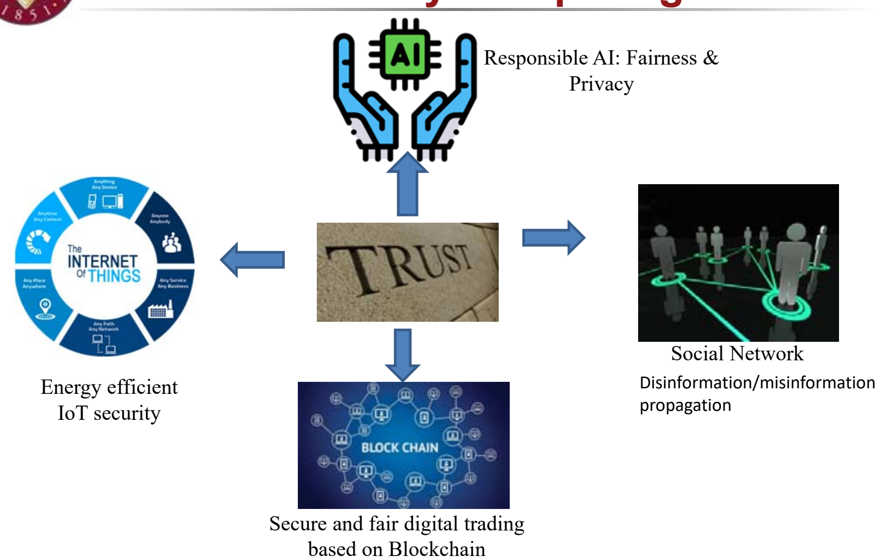
Associate Professor
Department of Computer Science and Engineering
Santa Clara University

6/23/2024

SCHOOL OF ENGINEERING

2

Research Interests – Trustworthy Computing



6/23/2024

SCHOOL OF ENGINEERING

A Smart Computing World With Emerging Challenges for Security & Trust

Enabling Technologies: Internet of Things, AI/ML, 6G, …

Image from Internet

How to facilitate trustworthy interactions among highly decentralized, heterogeneous entities?

3

SCHOOL of ENGINEERING

4

Outline

- ❑ Blockchain basis + Multi-signature
- ❑ Secure and Efficient Multi-Signature Schemes for Fabric
- ❑ Group-Oriented Multi-Signature Supporting Monotonic Endorse Policies in Hyperledger Fabric

6/23/2024 SCHOOL of ENGINEERING

5

Blockchain – Data Organization

Blockchain: A chain structure connecting blocks of transactions

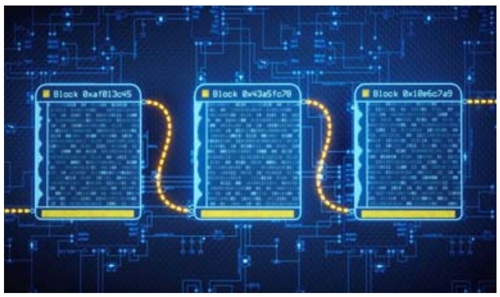
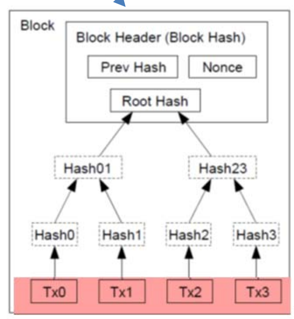


Image from https://www.hoyes.com/blog/can-blockchain-technology-save-the-credit-scoring-system/



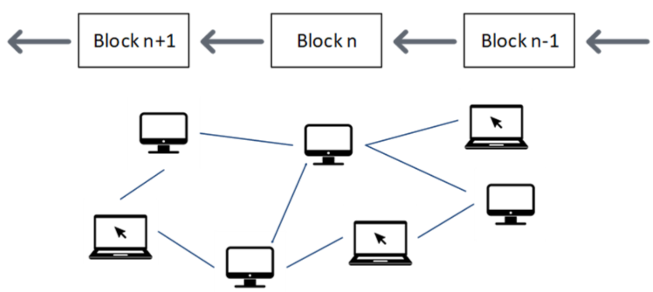
Transactions Hashed in a Merkle Tree

Image from Internet

6/23/2024 SCHOOL OF ENGINEERING

6

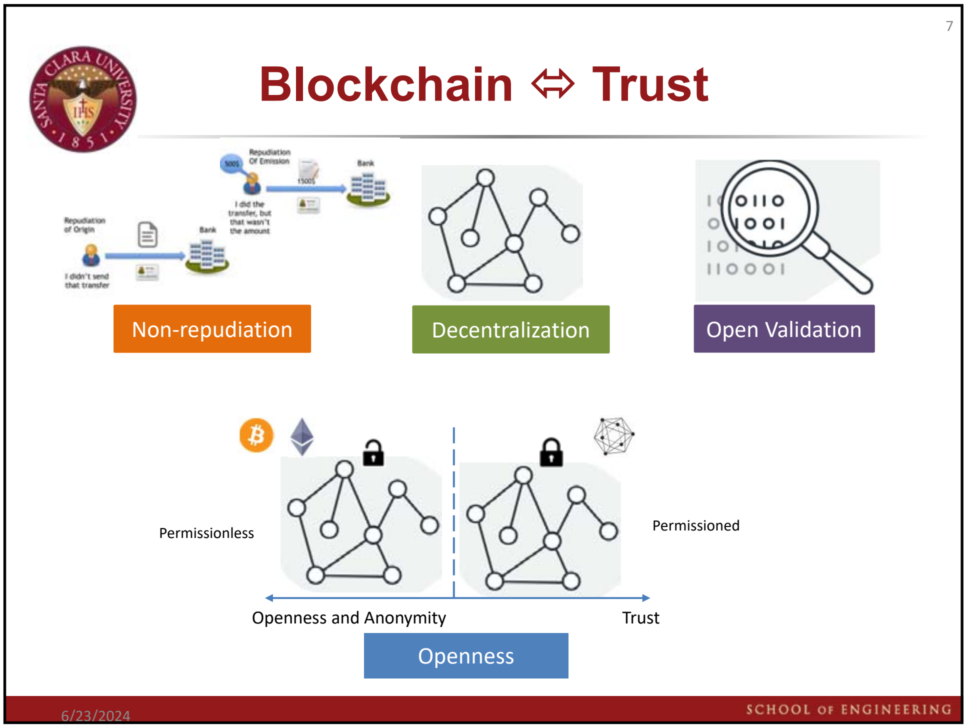
Blockchain – Hosted by a Distributed Network

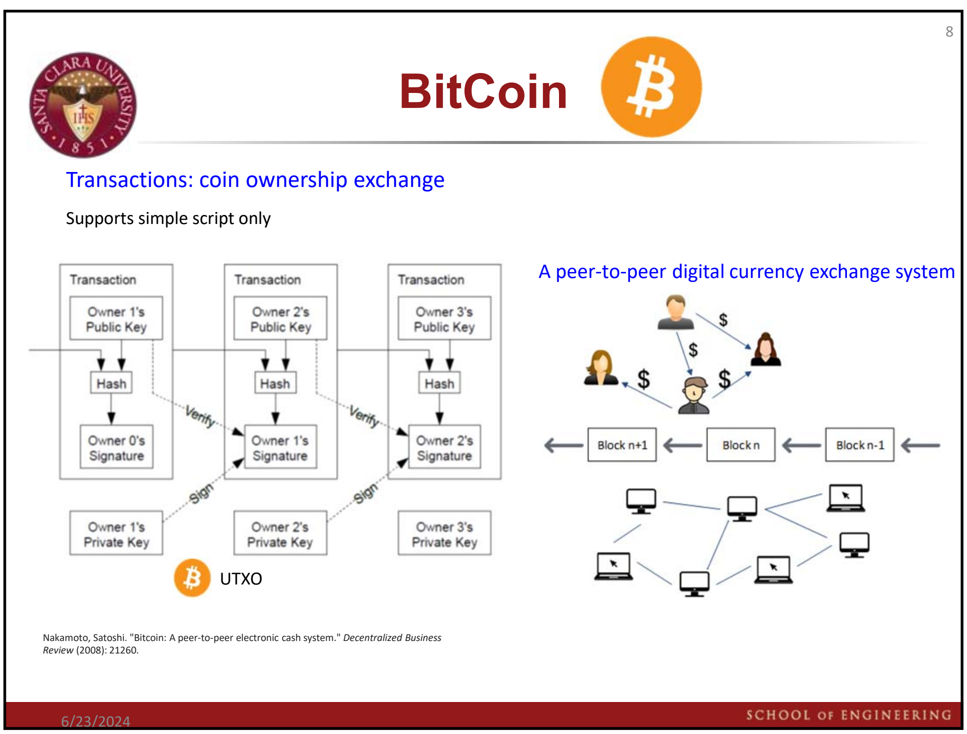


A peer-to-peer network with each node storing a copy (or a part of a copy) of the blockchain data

Data consistency: consensus algorithm

6/23/2024 SCHOOL OF ENGINEERING

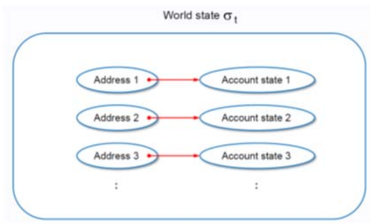

7
Blockchain ⇔ Trust
Non-repudiation
Decentralization
Open Validation
Permissionless
Permissioned
Openness and Anonymity
Trust
Openness
6/23/2024
SCHOOL of ENGINEERING


8
BitCoin
Transactions: coin ownership exchange
Supports simple script only
Transaction
Owner 1's Public Key
Hash
Owner 0's Signature
Owner 1's Private Key
Transaction
Owner 2's Public Key
Hash
Owner 1's Signature
Owner 2's Private Key
Transaction
Owner 3's Public Key
Hash
Owner 2's Signature
Owner 3's Private Key
Verify
Sign
UTXO
A peer-to-peer digital currency exchange system
Block n+1
Block n
Block n-1
Nakamoto, Satoshi. "Bitcoin: A peer-to-peer electronic cash system." Decentralized Business Review (2008): 21260.
6/23/2024
SCHOOL of ENGINEERING

9

Ethereum

blockchain system is generalized as a state machine.



The world state is a mapping between address and account state.

A transaction represents a valid arc between two states.

Smart contracts allow complex logic => Turing complete

6/23/2024

SCHOOL OF ENGINEERING

Fabric

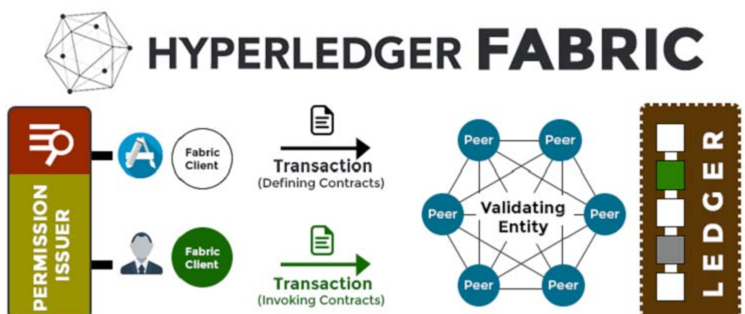


Image from Internet

- Strong identity management
- Enabled endorsement functions
- Adopt cryptographic digital signatures (ECDSA).

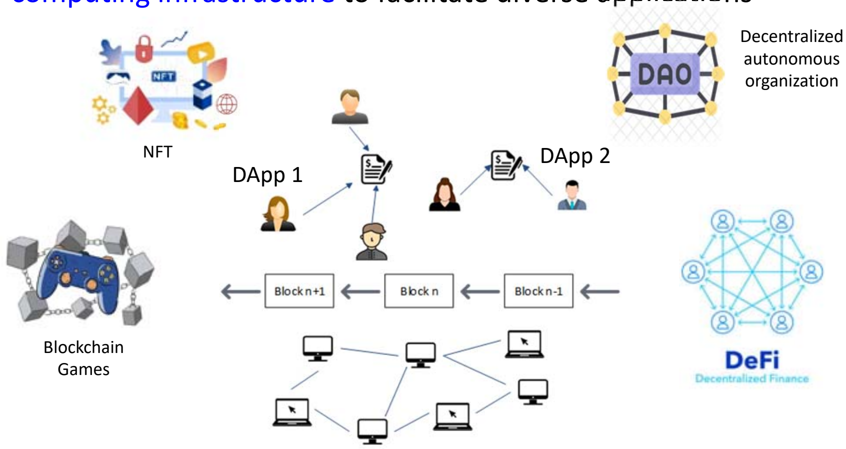
10

SCHOOL OF ENGINEERING

11

Blockchain as a Computing Infrastructure

Generalized blockchain system can potentially serve as a computing infrastructure to facilitate diverse applications



6/23/2024

SCHOOL OF ENGINEERING

Challenge: Digital Signature Efficiency

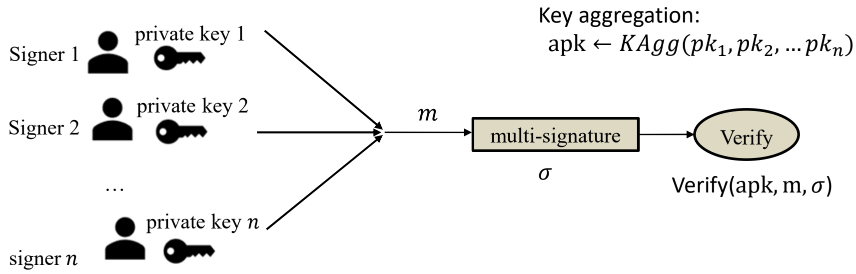
- Endorsement process based on digital signature is
  - **Inefficient:** signature collected from each endorser
  - **Resource consuming**: verification & storage of multiple signatures; significant broadcasting overhead
  - **Lack of scalability**: 100- 2000 tps.

12

SCHOOL OF ENGINEERING

A Promising Solution: Multi-Signature

Multi-signature allows a group of signers to jointly produce a single signature on the same message.



Advantage:

- One single signature saves storage space;
- One-time verification improves efficiency.

13

SCHOOL OF ENGINEERING

Existing Multi-Signature Basis

Basic Scheme	Modulus	Signature length	Computation Operations
RSA	2048 bits	2048 bits	Multiplications
Schnorr	2048 bits	448 bits	Multiplications
BLS	2048 bits	224 bits	Bilinear parings

```
Multiplication time consuming: 0.0415ms
pairing time consuming: 232.7998ms
```

- Bilinear pairing operation takes much more time than multiplication operation.
- Considering the signature length and computational cost, we study the Schnorr-based multi-signature schemes.

14

SCHOOL OF ENGINEERING

Schnorr Signatures

$pk = g^{sk}$

$r \leftarrow_R Z_q$

$t \leftarrow g^r$

$c \leftarrow H(t, m)$

$s \leftarrow r + c * sk \bmod q$

$\sigma \leftarrow (c, s)$

Verification:

$c = H(g^s * pk^{-c}, m)$

15

SCHOOL OF ENGINEERING

"Plain" Schnorr multi-signatures

Signing:

$pk_1 = g^{sk_1}$	$pk_2 = g^{sk_2}$	$pk_3 = g^{sk_3}$
$r_1 \leftarrow_R Z_q$	$r_2 \leftarrow_R Z_q$	$r_3 \leftarrow_R Z_q$
$t_1 \leftarrow g^{r_1}$	$t_2 \leftarrow g^{r_2}$	$t_3 \leftarrow g^{r_3}$
$t \leftarrow t_1 t_2 t_3$	$t \leftarrow t_1 t_2 t_3$	$t \leftarrow t_1 t_2 t_3$
$c \leftarrow H(t, m)$	$c \leftarrow H(t, m)$	$c \leftarrow H(t, m)$
$s_1 \leftarrow r_1 + c * sk_1 \bmod q$	$s_2 \leftarrow r_2 + c * sk_2 \bmod q$	$s_3 \leftarrow r_3 + c * sk_3 \bmod q$
$s \leftarrow s_1 + s_2 + s_3 \bmod q$	$s \leftarrow s_1 + s_2 + s_3 \bmod q$	$s \leftarrow s_1 + s_2 + s_3 \bmod q$
$\sigma \leftarrow (c, s)$	$\sigma \leftarrow (c, s)$	$\sigma \leftarrow (c, s)$

Verification:

$apk \leftarrow pk_1 * pk_2 * pk_3$

$c = H(g^s * apk^{-c}, m)$

16

SCHOOL OF ENGINEERING

Problem 1: Rogue-key Attacks

- *a malicious endorser arbitrarily claims his/her public key so that he/she can independently forge a joint signature*

$pk_1 = g^{sk_1}$

$pk_2 = g^{sk_2}/pk_1$

$apk = pk_1 * pk_2 = g^{sk_2}$

  - *The malicious endorser can control apk by claiming his/her public key based on the other parties public keys*
  - *Hence, he/she can compute signatures under apk by him/herself*

17

SCHOOL OF ENGINEERING

Problem 2: k-sum Attacks

- *An attack can succeed if a malicious endorser can simultaneously open k-1 signature oracle queries with honest signers on a message m as s_i, where $i \in \{1, .. k-1\}$, and get a valid signature $\sigma \leftarrow (c^*, s^*)$ on a target message $m^* \neq m$, meaning that*

$$c^* = \sum_{i=1}^{k-1} c_i \qquad s^* = \sum_{i=1}^{k-1} s_i + c^* * sk \quad \Rightarrow \quad c^* = H(g^s * apk^{-c}, m^*) = H(g^{\sum_{i=1}^{k-1} r_i}, m^*)$$

- *Since $c_i \leftarrow H(t_i, m)$, where t_i can be controlled by the attacker, an attack can succeed if the attacker is able to construct*

$$c^* = \sum_{i=1}^{k-1} c_i = \sum_{i=1}^{k-1} H(t_i, m) \quad \Rightarrow \quad \sum_{i=1}^{k-1} H(t_i, m) = H(g^{\sum_{i=1}^{k-1} r_i}, m^*)$$

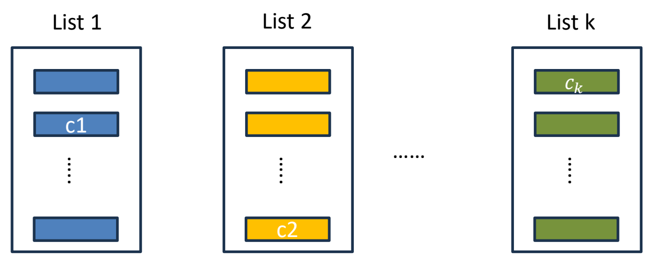
- *The last signer of the endorsement, with excessive power, can forge a joint signature on a new message.*

18

SCHOOL OF ENGINEERING

K-Sum Problem

- Wagner's generalized birthday attack (K-sum problem)
  - Given k lists of random elements in Z_q, find $(c_1, \dots c_k)$ in lists such that $c_1 + \cdots + c_k = 0 \ mod \ q$



19

SCHOOL OF ENGINEERING

Multi-Signature for Blockchain

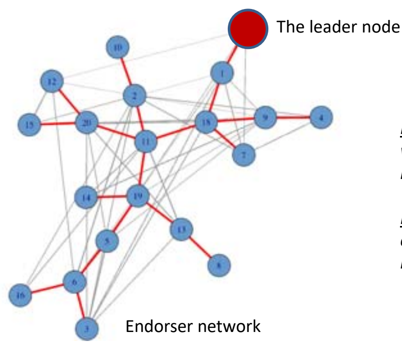
*Xiao Yue, Peng Zhang, **Yuhong Liu**, "Secure and Efficient Multi-Signature Schemes for Fabric: An Enterprise Blockchain Platform", IEEE Transactions on Information Forensics and Security, 16 (2020): 1782-1794.*

20

SCHOOL OF ENGINEERING

CoSi: Multi-signature Scheme

- A very popular Schnorr-based scheme
- High scalability due to the spanning tree structure
  - a loop-free logical topology
  - a single active path between any two network nodes.



K-sum problem attacks: the leader of the endorsement, with excessive power, can forge a joint signature on a new message.

Rogue-key attacks: a malicious endorser arbitrarily claims his/her public key so that he/she can independently forge a joint signature

21

SCHOOL OF ENGINEERING

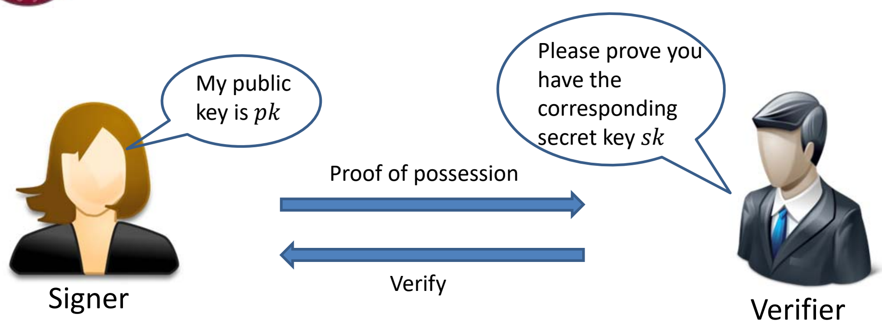
Proposed Schemes

- A Gamma Multi-Signature (GMS) – security
  - Spanning tree structure (high scalability)
  - Proof of possession against rogue-key attacks
  - Improved signing process - against k-sum attacks
- An Advanced Gamma Multi-Signature (AGMS) - efficiency
  - Improved online efficiency by reordering the signing process

22

SCHOOL OF ENGINEERING

Proof of Possession



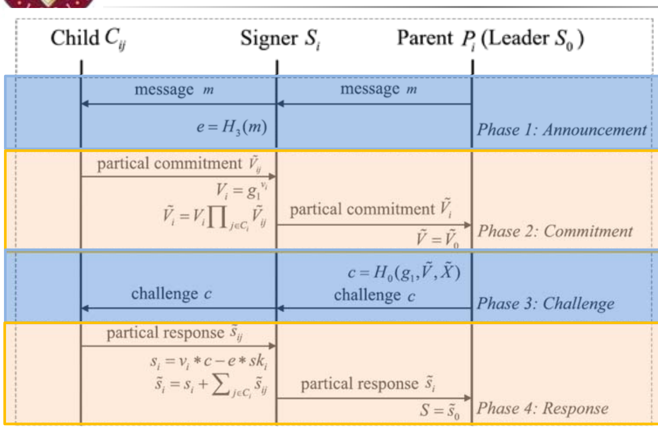
Proof of possession $\pi = (a, d) \begin{cases} a = H_1(g_1, g_1^r) \\ d = r * a - H_2(pk) * sk \bmod q \end{cases}$

verification $a = H_1(g_1, (g_1^d y^{H_2(y)})^{a^{-1}})$?

23

SCHOOL OF ENGINEERING

Proposed GMS – Signing Process



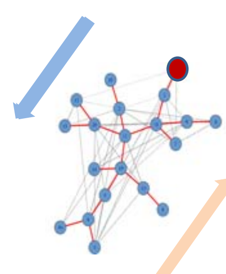


Fig. 1. The signing algorithm of our GMS scheme (We suppose that signer S_i holds the key pair (pk_i, sk_i), where $pk_i = (y_i, \pi_i)$, and parent P_i works as a leader S_0. If parent P_i is not a leader, it just works as signer S_i. Finally, the leader S_0 outputs (c, S) as the joint signature.).

24

SCHOOL OF ENGINEERING

Proposed GMS – Signing Process

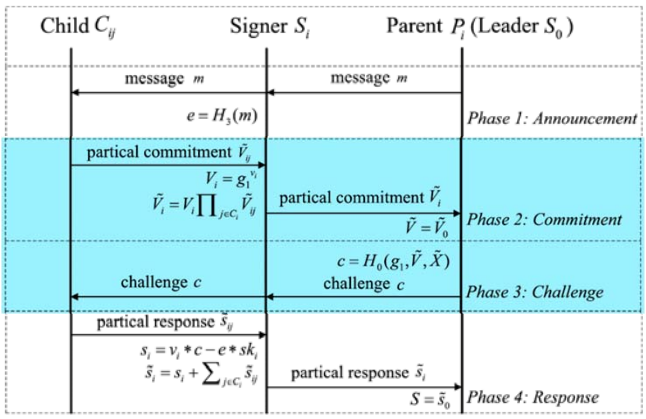
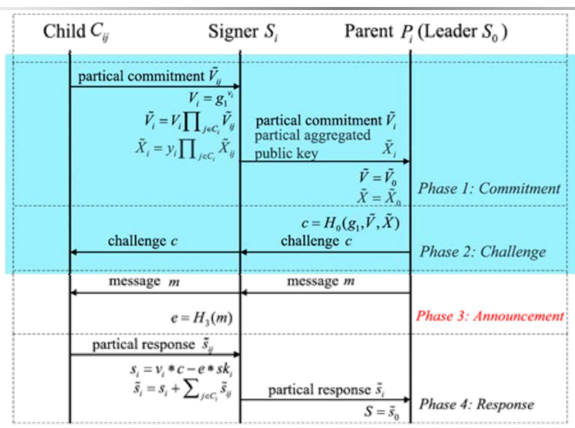


Fig. 1. The signing algorithm of our GMS scheme (We suppose that signer S_i holds the key pair (pk_i, sk_i), where $pk_i = (y_i, \pi_i)$, and parent P_i works as a leader S_0. If parent P_i is not a leader, it just works as signer S_i. Finally, the leader S_0 outputs (c, S) as the joint signature.)

25

SCHOOL OF ENGINEERING

Advanced GMS (AGMS)



Reorder phase 1, 2, 3

Fig. 2. The signing algorithm of the proposed AGMS scheme (Text in red indicates changes from Fig. 1. We suppose that signer S_i holds the key pair (pk_i, sk_i), where $pk_i = (y_i, \pi_i)$, and parent P_i works as a leader S_0. If parent P_i is not a leader, it just works as signer S_i. The key aggregation algorithm also runs together with the signing algorithm. Finally, the leader S_0 outputs (c, S) as the joint signature.)

26

SCHOOL OF ENGINEERING

Running Time on the Leader node

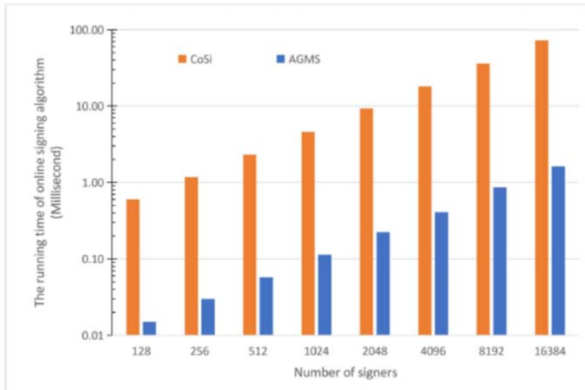


Fig. 7. The CPU running time on a leader node of CoSi and AGMS in online signing phase (y-axis has logarithmic scale.).

27

SCHOOL OF ENGINEERING

Applying on Fabric

Key revision: replace the original ECDSA by the proposed AGMS.

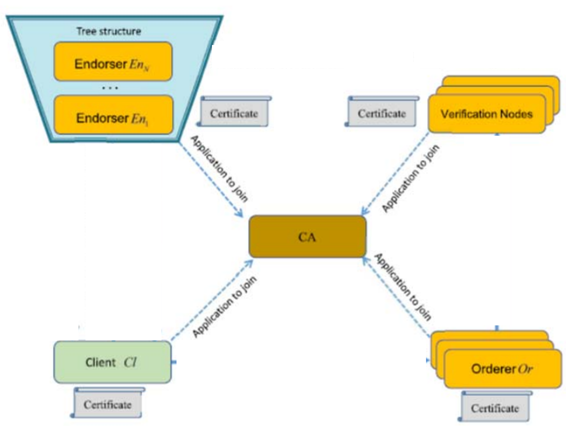


Fig. 9. The revised Fabric transaction process.

28

SCHOOL OF ENGINEERING

Performance Testing on Fabric

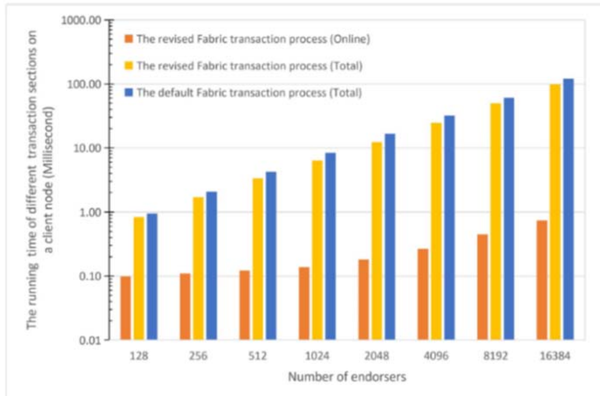


Fig. 13. The total CPU running time on a client node between the default Fabric transaction process and the revised Fabric transaction process (y-axis has logarithmic scale.).

29

SCHOOL OF ENGINEERING

Another Problem – Flexible Endorsement

- Endorsement policy in Fabric can be expressed as:

EXPR(E[, E...]),

where *EXPR* is "*AND*", "*OR*" or "*OutOf*"; and *E* is either an endorser or another nested call to *EXPR*.

- Typical endorsement policies are monotonic and group-based:

AND('Org1.member',' Org2.member', ...),

where "*OR*" expression is used to check if any member from Org1 (and Org2) has endorsed.

- Not supported by existing Multi-signature schemes: Existing multi-signature schemes mainly focus on "AND" relationship among multiple singers, they can only support individual-based endorsement policies like

"AND('member1','member2',...)".

30

SCHOOL OF ENGINEERING

*Peng Zhang, Yongwen Huang, Fa Ge, **Yuhong Liu,** "Group-Oriented Multi-Signature Supporting Monotonic Endorse Policies in Hyperledger Fabric", IEEE Blockchain 2023, Hainan, China, Dec. 17-23, 2023*

31

SCHOOL OF ENGINEERING

The Proposed Scheme

- We propose a Group-oriented Multi-Signature scheme, which supports secure and more flexible endorsement policy
- Based on the proposed scheme, the transaction protocol in Fabric is optimized, so that the block size and verification time are reduced.

32

SCHOOL OF ENGINEERING

The Proposed Scheme

Group-oriented multi-signature scheme with smart contract

- Smart contract on blockchains is distributed on peer-to-peer networks, publicly verifiable, and executed automatically.
- By introducing public and trustworthy smart contracts to be the last signer responsible for commitment operations, *k*-sum problem attacks are prevented.

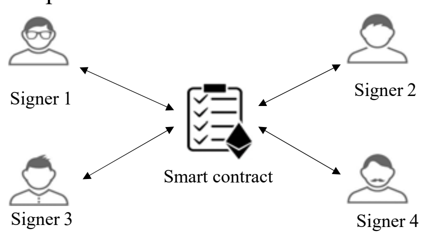


Fig. 3 The network structure of our scheme

It is responsible for:

1. Collecting commitments $R_{i,j}$ from all signers;
2. Recording the timestamp t and computing the last commitment $W = g^{H(t)}$;
3. Computing and Distributing the joint commitments.

$$R = W^{\theta} \prod_{i=1}^{\theta} R_{i,j}$$

33

SCHOOL OF ENGINEERING

The Proposed Scheme

- By introducing **Chinese Remainder Theorem** to combine all public keys of members in a group into one group public key, the public key of each member is unknown to all others except the group administrator, so that only the group public key is involved in verification.

$$\begin{cases} k_i = X_{i,1} (mod\ p_{i,1}) \\ \vdots \\ k_i = X_{i,\eta} (mod\ p_{i,\eta}) \end{cases}$$

34

SCHOOL OF ENGINEERING

Security Analysis

We conduct a security analysis on the proposed scheme. Based on the difficulty problem assumption, our scheme satisfies the unforgeability, anonymity, revocability and traceability.

- Unforgeability: The signature cannot be forged by an attacker.
- Anonymity: The identity of the signer will not be revealed.
- Revocability: Signers who exit the group cannot be regenerated into legal signatures.
- Traceability: Only the group administrator knows the identities of the members participating in the multi-signature.

35

SCHOOL OF ENGINEERING

Comparison

Table1 The comparisons of key and signature length (θ represents the number of groups)

Scheme	Public key	Private key	Signature
Musig2	$\lvert\mathbb{G}\rvert$	$\lvert\mathbb{Z}_\alpha\rvert$	$\lvert\mathbb{G}\rvert + \lvert\mathbb{Z}_\alpha\rvert$
GMS	$\lvert\mathbb{G}\rvert$	$\lvert\mathbb{Z}_\alpha\rvert$	$\lvert\mathbb{G}\rvert + (\theta + 1)\lvert\mathbb{Z}_\alpha\rvert$

Table2 The comparisons of computational cost and security assumptions
(*Exp* represents the calculation cost of an exponential operation in the group)

Scheme	Sign	Verify	Join	Revoke	Assumptions
Musig2	$7EXP$	$2EXP$	–	–	AOMDL
GMS	$2EXP$	$2EXP$	$1EXP$	$0EXP$	DL

36

SCHOOL OF ENGINEERING

Experiment Results

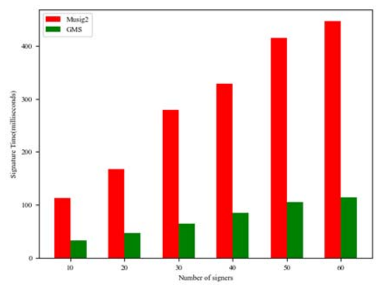


Fig. 4 Signature time comparisons of Musig2 and GMS.

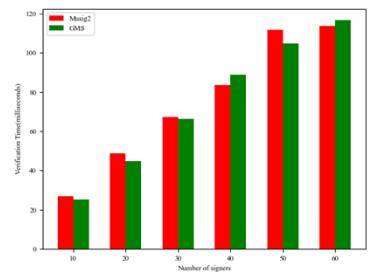


Fig. 5 Verification time comparisons of Musig2 and GMS

37

SCHOOL OF ENGINEERING

Application on Fabric

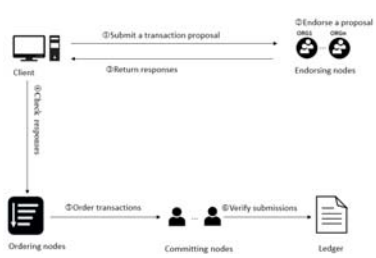


Fig. 6 Original transaction protocol in Fabric

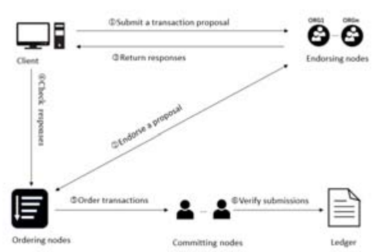


Fig. 7 Improved transaction protocol in Fabric

Since the proposed scheme introduces smart contracts, we install them on the ordering nodes, and endorsing nodes need to interact with the ordering nodes.

38

SCHOOL OF ENGINEERING

Endorsement Policy

$AND(OR(org1.member1, org1.member2), \ldots, OR(org10.member1, .., org10.member2))$

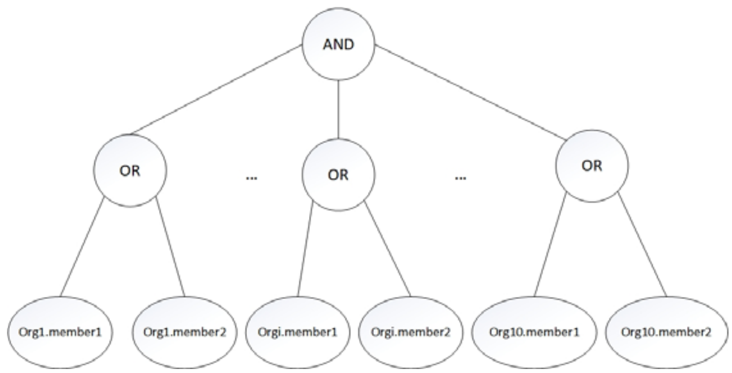


Fig. 8 The endorsement policy used in this experiment

39

SCHOOL OF ENGINEERING

Experiment Results

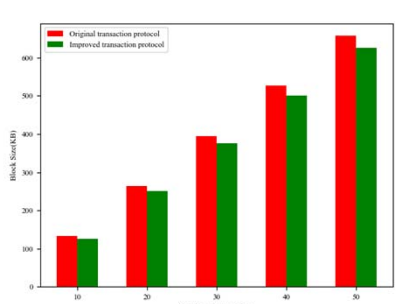


Fig. 9 The block size in original and improved transaction protocols

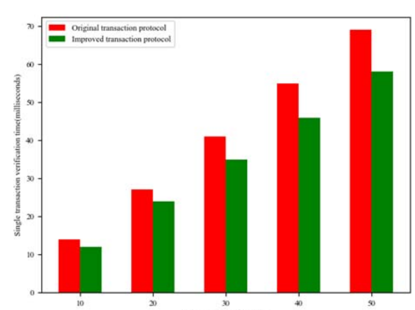


Fig. 10 The verification time for each transaction in original and improved transaction protocols

40

SCHOOL OF ENGINEERING

41

Other Relevant Studies

- Autonomous vehicle networks,
- Transactive Energy trading,
- Internet of Things,

…

More to come …

6/23/2024 SCHOOL OF ENGINEERING

42

Yuhong Liu
Associate Professor
Computer Science Engineering Department,
Santa Clara University
500 El Camino Real Santa Clara, CA, 95053
Email: yhliu@scu.edu

Thank You !

Questions ?

6/23/2024 SCHOOL OF ENGINEERING