A New Multisignature Scheme with Public Key Aggregation for Blockchain

Duc-Phong Le, Guomin Yang, Ali Ghorbani

PST'19
August 28, 2019, Fredericton

Canadian Institute for Cybersecurity (CIC)

Outline

- Introduction to Blockchain Technology
  - What is blockchain?
  - Benefits of blockchain technology
  - Digital signatures in blockchain
- DDH-based Multisignatures
  - DDH Assumptions
  - New Mulsignature Scheme
  - Security Proof
- Conclusions

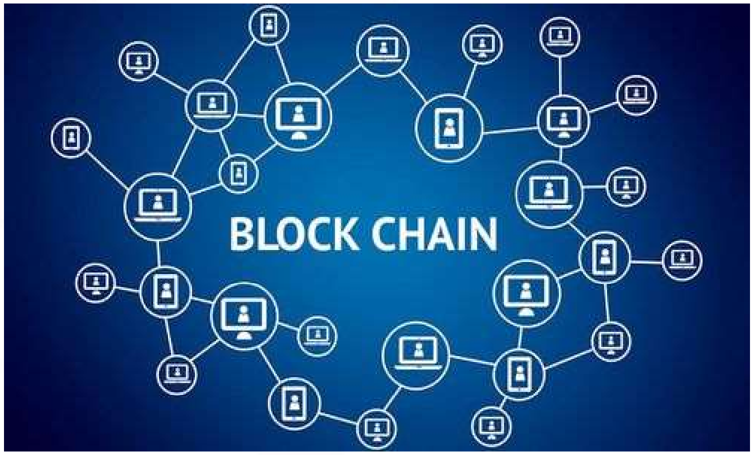
What is Blockchain?

CIC

"*A blockchain is a growing list of records, called blocks, that are linked using cryptography.*" Wikipedia.org.

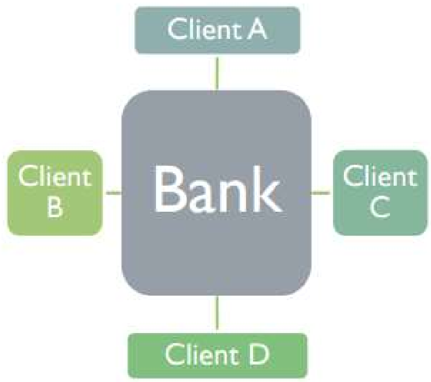
"*It is an open, distributed ledger that can record transactions between two parties efficiently and in a verifiable and permanent way.*" Wikipedia.org.

- Open
- Distributed
- Ledger
- P2P
- Permanent



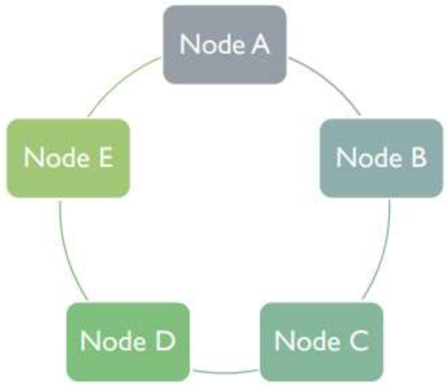
Distributed Ledger

Centralized ledger



- Each client has her own ledger, stored at Bank
- Transactions are executed through Bank

Decentralize ledger



- There is one ledger. All nodes have access to it
- Transactions are directly between two parties and approved through a consensus mechanism

How does a Blockchain work?

1. Users initiate transactions using their Digital Signatures
2. Users broadcast their transactions to Nodes
3. Nodes collaborate to validate each transaction
4. Nodes aggregate validated transactions into a new block
5. Nodes Broadcast Blocks to each other
6. Consensus protocol used
7. Block reflecting “true state” is chained to prior Block

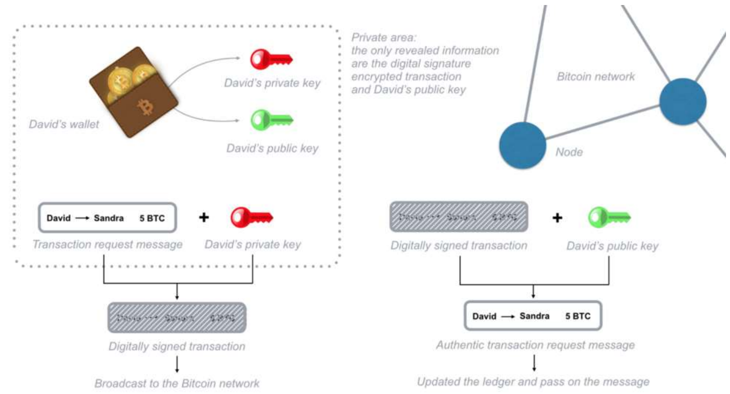
Benefits of Blockchain

Potential benefits of blockchain

- Reduce costs of overall transactions
- Reduction in systemic risks
- Irrevocable and tamper-resistant transactions
- Fraud minimisation
- Improved security and efficiency of transactions
- Enabling effective monitoring and auditing by participants, supervisors, and regulators

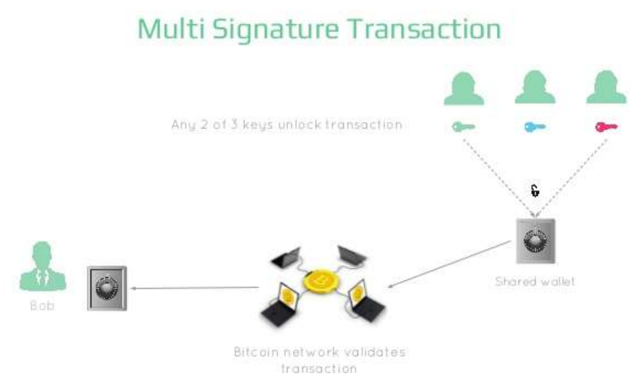
Where Blockchain Use Digital Signatures?

CIC



A digital signature is generated when a user create a new transaction

- To authorize the transaction
- To prove the ownership and non-repudiation of the transaction
- Signature and corresponding public key are stored on Blockchain for verification



Why blockchain needs mulsignatures?

- Allow co-spending, crowdfunding
- Improve the security of transactions/wallets
- Multisignature and corresponding public keys of n users are stored on Blockchain for verification

Outline

- Introduction to Blockchain Technology
  - What is blockchain?
  - Benefits of blockchain technology
  - Digital signatures in blockchain
- DDH-based Multisignatures
  - DDH Assumptions
  - New Mulsignature Scheme
  - Security Proof
- Conclusions

A New Multisignature Scheme

- **Challenges for Multisignatures on Blockchain**
  - Size of signatures and public keys grows up with the number of signers
  - Challenging in verification of public key, i.e., key pairs may not be issued by a trusted CA

- **Design Goals**
  - Multi-signature size is independent from the number of signers, i.e., equivalent to the size of a single signature
  - The size of "public key" used to verify the signature is independent from the number of signers, i.e., equivalent to the size of a single public key
    - so-called, *public key aggregation*
  - Public key could be chosen arbitrarily by users on the system

Decisional Diffie-Hellman (DDH) Assumption

CIC

Consider a cyclic group G of order q, with generator g. The DDH assumption states that, given g^a an g^b, where a, b randomly chosen from Z_q, the value g^{ab} looks like a random element in G

- Cyclic group G could be defined over integers (multiplicative group) or EC points (additive group)
- DDH is considered to be a stronger assumption than Discrete Logarithm (DL) assumption as there are groups (e.g., GDH groups) in which DDH is easy but DL is still hard
- However, for a variety of groups, DDH is equivalent to DL

DDH-based Multisignatures

CIC

Parameter generation: Given a group G of order q, a trusted center chooses a generator $\mathrm{g} \in G$, a random $\mathrm{h} \in G$, and hash functions H_{com}, H_{agg}, and H_{sig}

Key generation: Each user picks a random $x_i \in Z_q$ as private key, corresponding public keys are $pk_i = (y_i, z_i) = (g^{x_i}, h^{x_i})$

Key aggregation: Given $L = \{pk_1, \ldots, pk_n\}$, aggregated public keys are computed:

$$apk_1 \leftarrow \prod_{i=1}^{n} y_i^{H_{agg}(pk_i, L)}$$

and

$$apk_2 \leftarrow \prod_{i=1}^{n} z_i^{H_{agg}(pk_i, L)}$$

DDH-based Multisignatures

CIC

Signature generation: consists of 3 rounds

Round 1: - Each user picks a random $r_i \in Z_q$, computes $u_i = g^{r_i}$, and $v_i = h^{r_i}$, then queries $h_i = H_{com}(u_i)$ and $t_i = H_{com}(v_i)$

- Send (h_i, t_i) to every other users

Round 2: Each user receives (h_j, t_j) from user *j*, then sends back (u_i, v_i)

Round 3: - Each user receiving (u_i, v_i) will check if $h_i = H_{com}(u_i)$ and $t_i = H_{com}(v_i)$, then computes: $u = \prod u_i$ and $v = \prod v_i$

- computes $a_i = H_{agg}(pk_i, L)$, then queries $c = H_{sig}(apk_1, apk_2, u, v, L, m)$, $c_i = a_i c$, and $s_i = r_i + x_i c_i$
- send to signer *j*: s_i

Multi-Signature: (c, s), where $s = \sum_{i=1}^{n} s_i \bmod q$

Verification: Given a valid signature (c, s) and L, m, a verifier computes $u' = g^s \cdot apk_1^{-c}$, $v' = h^s \cdot apk_2^{-c}$, and check whether $c = H_{sig}(apk_1, apk_2, u', u', L, m)$

Security

Attack Scenario

- Attacker in collaboration with a honest user *P* to generate multi-signatures on some *adaptively chosen* messages *m*
- Then, attacker produces a new multi-signature on a new message *m'* without a help from the honest user *P*
- She has to convince any verifier that the honest signer participated in signing the message *m'*

Attacker Model

- Attacker is able to choose his public key arbitrarily, e.g., she can include the targeted user's public key in her public key for the purpose of compromising a multisignature on a targeted message
- Attacker is able to choose any messages in a adaptive way to request valid multi-signatures signed with the honest user

Security

Theorem 3. The proposed multisignature scheme is (t, q_H, q_S, ϵ)-unforgeable in the random oracle model if $q > 8q_H/\epsilon$ and if the DDH problem is (t', ϵ')-unforgeable in $\mathbb{G}$, where

$$\epsilon' \geq \epsilon/(8q_H)$$

and

$$t' \leq t + (q_H + q_S + 1)t_{exp}.$$

If the DDH assumption holds, the proposed multisignature scheme is secure against adaptively chosen messages attacks in plain public key model

- Under the decisional Diffie-Hellman (DDH) assumption, the public key $\mathrm{pk_i} = (\mathrm{y_i}, \mathrm{z_i}) = (\mathrm{g^{x_i}}, \mathrm{h^{x_i}})$ looks *random* to an eavesdropper

Conclusion

- The first multisignature scheme, that
  - is proven secure under the ***DDH assumption***
  - is secure in the ***plain public key model***, i.e., the attacker is able to choose an arbitrary public key, including a function of the honest user's public key
  - supports ***public key aggregation***, i.e., for smaller blockchains as all public key should stored on the blockchains

CIC

Thank you !