Blockchain Interoperability

Goals of blockchain

- Borderless
- Censorship resistant (neutrality)
- Open

Applications

- Healthcare [1]
- Finance
- Supply-chain [2]

[1] Holbl, Marko, Kompara, Marko & Nemec Zlatolas, Lili (2018) 'A systematic review of the use of blockchain in healthcare', *Symmetry* 10, 470

[2] Notheisen, Benedikt & Shanmugam, Arun Prasad (2017) 'Trading real-world assets on blockchain', *Business & Information Systems Engineering*

Prevalent current situation

- Every blockchain can't communicate with other blockchains
- No extension

Consequences:

- Vendor lock-in and centralization
- Complex blockchain design and less security
- Coin Maximalism
- No blockchain adoption

Interoperability advantages

- Help solving scalability vs decentralization trade-off
- Add blockchain features
- Different assets besides currencies

Three Layers Structure

The three layers of abstraction:

- Layer 0: message exchange
- Layer 1: blockchains structure
- Layer 2: side-channels

Layer 0 - Basic Idea

- Clients of a blockchain can modify only their blockchain
- Clients of a blockchain can view changes of other blockchains

Layer 0 - Methods

- Proof of Lock [3]
- Notary schemes [4]
- Atomic Swaps [5]

[3] Back, Adam, Corallo, Matt, Dashjr, Luke, Friedenbach, Mark, Maxwell, Gregory, Miller, Andrew, Poelstra, Andrew & Wuille, Pieter (2014) 'Enabling Blockchain Innovations with Pegged Sidechains'

[4] Dilley, Johnny, Poelstra, Andrew, Wilkins, Jonathan, Piekarska, Marta & Friedenbach, Mark (2016) 'Strong federations: An interoperable blockchain solution to centralized third-party risks', *arXiv preprint arXiv:1612.05491*

[5] Miraz, Mahdi & Donald, David C (2019) 'Atomic Cross-chain Swaps: Development, Trajectory and Potential of Non-monetary Digital Token Swap Facilities', *Annals of Emerging Technologies in Computing (AETiC) Vol* 3

Proof of Lock

1. Client A locks funds on blockchain B_1
2. Client A has/produces proof of lock and presents it to clients on blockchain B_2
3. A client B on blockchain B_2 verifies the proof and releases/creates equivalent funds on blockchain $B2$

Proof of Lock - Comments

Con:

- Implements light-clients
- Risk of blocking transaction
- Slow
- Need to peg blockchain

Pro:

- Easy to implement
- Decentralized

Notary schemes

- Third party does the exchange
- Examples
  - Online exchanges
  - Smart contract
  - Federation with majority

Notary schemes - Comments

Con:

- Centralized
- Single point of failure

Pro:

- Easy to implement
- Fast

Atomic Swaps

- Either successfully concluded or revert
- Currently only with hashtimelock commitment schemes
  - Startup phase
  - Commitment phase
  - Claim phase

Hashtimelock - Startup phase/1

1. Alice creates a number x , hash it $H(x)$ and gives the hash to Bob)
2. Alice creates a transaction T_1 which would transfer on B_1 n_1 tokens from Alice's address to Bob's one
   - Can be spent only providing x via a message signed by Bob or with a general transaction signed by both Alice and Bob

Hashtimelock - Startup phase/2

3. Alice creates a transaction T_2 which would transfer, after t hours, on B_1 the tokens from T_1 to her address; she sends this transaction to Bob
4. Bob signs transaction T_2 and returns it to Alice
   - This is Alice's warranty that even if Bob turns out to be malicious, she will have her tokens back

End of the startup phase

Hashtimelock - Commitment phase/1

1. Alice publishes T_1 on B_1
2. Bob creates a transaction T_3 which would transfer on B_2 n_2 tokens from Bob's address to Alice's one
   - Can be spent only providing x via a message signed by Alice or with a general transaction signed by both Alice and Bob

Hashtimelock - Commitment phase/2

3. Bob creates a transaction T_4 which would transfer, after $\frac{t}{2}$ hours, on B_2 the tokens from T_3 to his address; he sends this transaction to Alice
4. Alice signs transaction T_4 and returns it to Bob
   - This is Bob's warranty that even if Alice turns out to be malicious, he will have his tokens back
5. Bob publishes T_3 on B_2

End of commitment phase

Hashtimelock - Claim phase

1. Alice spends T_3 before $\frac{t}{2}$ hours and gives x
2. Bob spends T_1 before t hours using x

Hashtimelock - Comments

Con:

- Difficult to implement
- Slow
- Prone to DDoS
- Weak in case of reorg
- Same hash function on both blockchain

Pro:

- Censorship resistant
- Highest decentralization

Layer 1 - Configurations

- Tree
- Star
- Parallel chains [6]

[6] Lerner, Sergio Demian (2016) *Drivechains, Sidechains and Hybrid 2-way Peg Designs_R9*

Tree

- Root chain
- Every blockchain reports its state to the parent chain for inter-chain transaction
- Examples:
  - Treechain (deprecated)
  - Plasma [7]

[7] Poon, Joseph & Buterin, Vitalik () 'Plasma: Scalable Autonomous Smart Contracts'

Star

- One central blockchain
- All the blockchain communication must pass by the central hub
- Example:
  - Cosmos
  - Polkadot [8]

[8] Wood, Dr Gavin 'Polkadot: Vision for a Heterogeneous Multi-chain framework'

Parallel chains

- Separated blockchains capable of exchanging messages
- Examples:
  - RSK
  - Blockstack

Layer 2 - Side-channels

- Not all the transaction must be recorded (e.g. smaller recurrent payments)
- Active development because of scalability improvements
- Examples:
  - Lightning Network (Bitcoin)
  - Raiden Network (Ethereum)

Contacts

- Guido Boella: guido.boella@unito.it
- Fadi Barbara: fadi@di.unito.it