Raiden Network

Haoran Qi
11/02/2018

Summary

- An off-chain scaling solution for performing ERC20-compliant token transfers on the Ethereum blockchain
- Allows secure transfers of tokens between participants without the need for global consensus
- Lightning Network on Ethereum

The Netting Channel Smart Contract

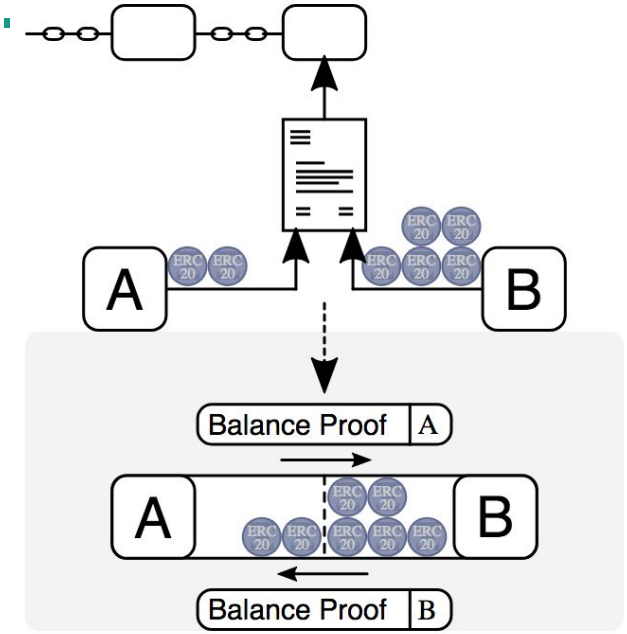
- = bidirectional payment channel
- Withdraw token:
  - One participant can withdraw token with signatures from both parties in the channel.

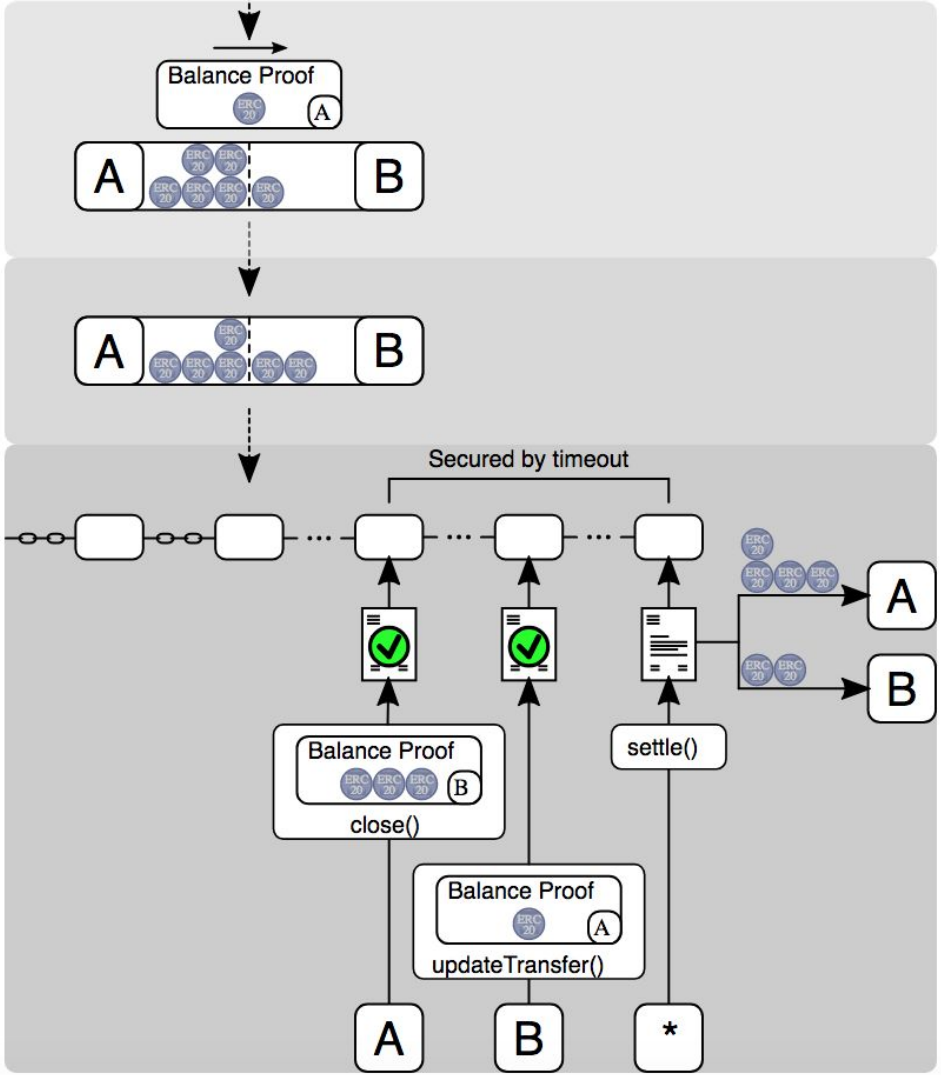
A channel's life cycle

- Deployment
- Funding / Usage
- Close
- Settle

Balance Proofs

- Contains the final sum of all Raiden transfers sent to a participant up to a certain point, digitally signed by the sender
- Properties
  - A nonce
  - The transferred amount
  - The root node of the pending locks merkle tree
  - A signature containing all the above


A
B
Balance Proof
A
A
B
Balance Proof
B


Balance Proof
A
A
B
A
B
Secured by timeout
A
B
Balance Proof
B
close()
settle()
Balance Proof
A
updateTransfer()
A
B
*

Raiden Transfers

- Direct Transfers
- Mediated Transfers
- Refund Transfers

Direct Transfers

- Not rely on locks
- Automatically completed once the network packet is sent off

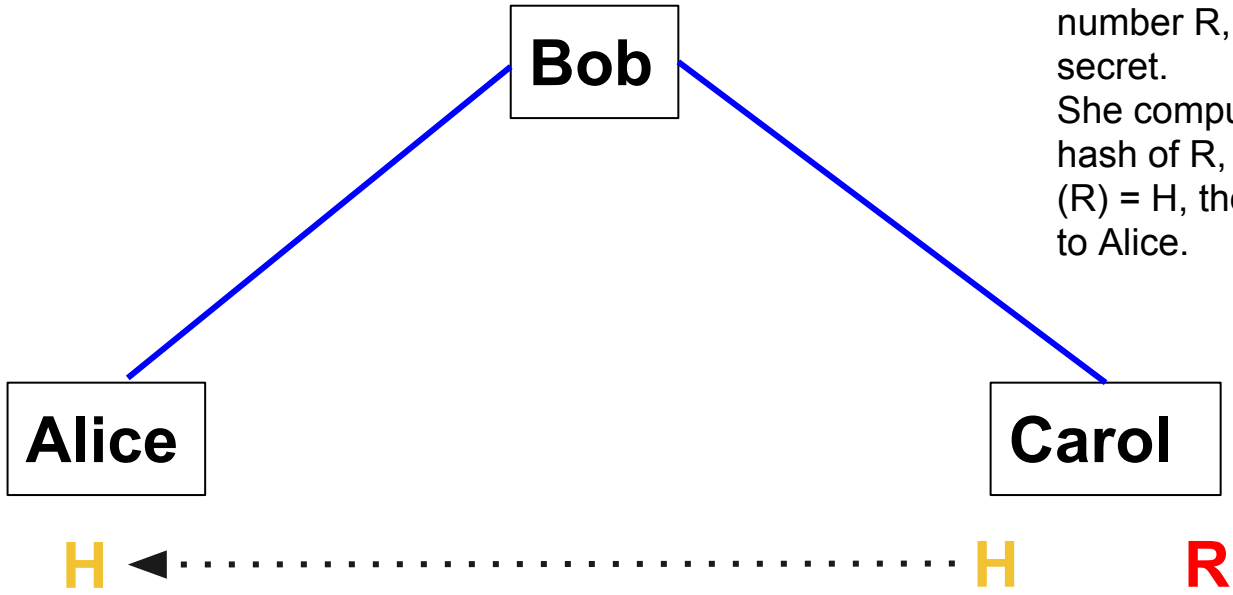
Direct Transfers

- Alice wants to transfer n tokens to Bob.
- Alice creates a new transfer with.
  - transferred_amount = current_value + n
  - locksroot = current_locksroot_value
  - nonce = current_value + 1
- Alice signs the transfer and sends it to Bob and at this point should consider the transfer complete.

Mediated Transfers

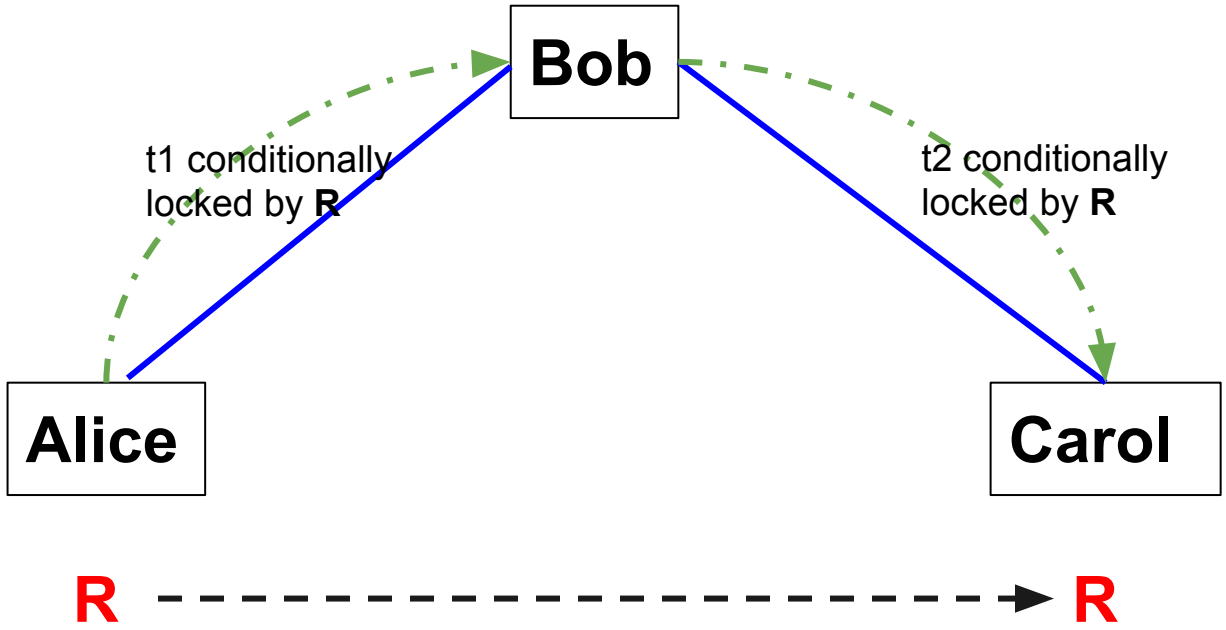
- Hashlocked transfer
- Currently raiden supports only one type of lock
- Lock includes a secret an an expiration

Hash-Locked Contracts



Carol makes a random number R, and keeps it secret.
She computes the hash of R, hash (R) = H, then sends it to Alice.

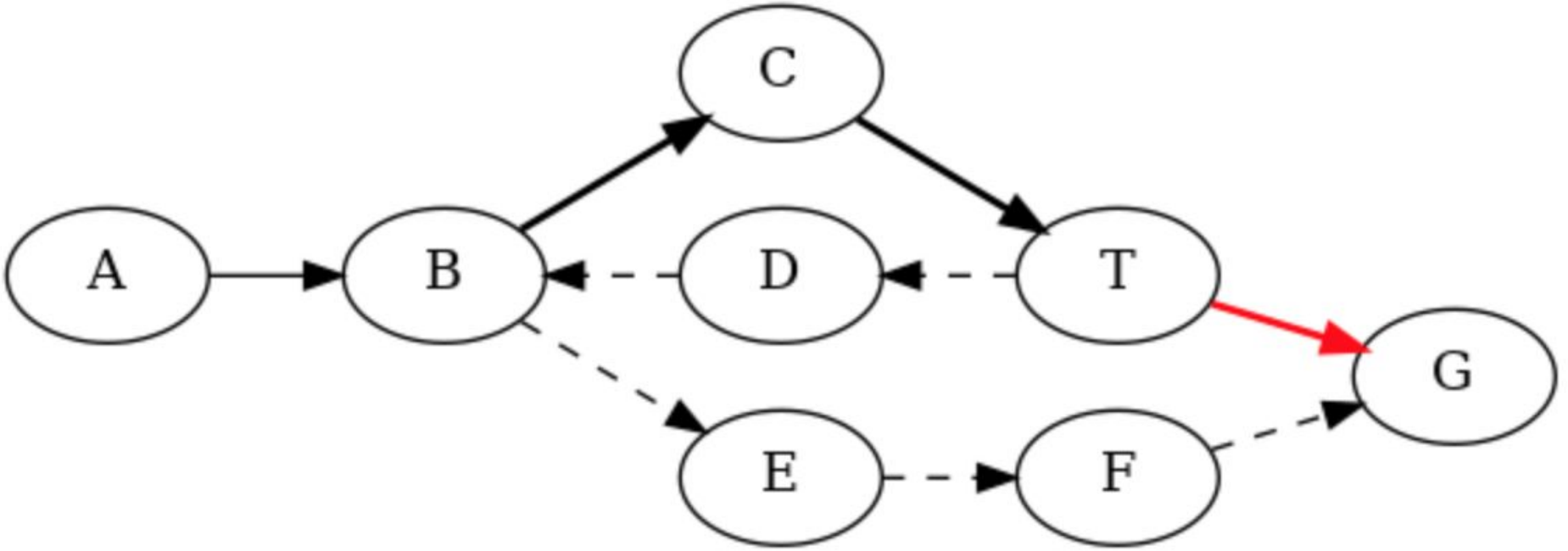
Mediated Transfers



Transfer Routing

- Scalable Routing is indeed one of the biggest issues of payment channel networks
- A trade-off between centralization, privacy, and efficiency

Transfer Routing(Naive Solution)



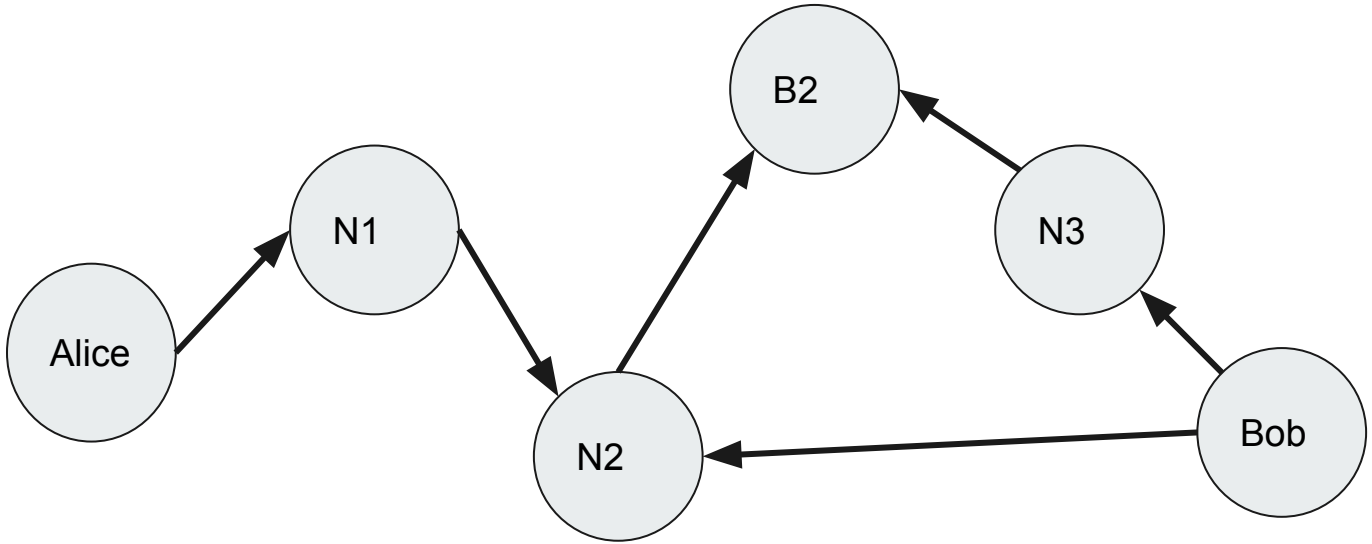
Transfer Routing(Efficient Solution)

- Centralized service
- Used in Raiden Network per their latest specification released in Sep 2018

More on Transfer Routing(Lightning Network)

- At intervals, the following process is initiated:
  - A beacon node B is selected via a pseudo-random process.
  - Neighbors of B broadcast their shortest route to B to their neighbors.
  - The neighbors of neighbors of B now become aware of a route to B and in turn broadcast their shortest route to B.
  - This cascades through the network until every reachable node has broadcasted their shortest route to the beacon node B.
  - Whenever a node becomes aware of a new shorter route, it broadcasts this updated shortest route as well.
  - After a short wait, start from top with a new beacon node B1.

More on Transfer Routing(Lightning Network)



References

- https://raiden.network
- https://raiden-network.readthedocs.io/en/stable/spec.html
- https://media.readthedocs.org/pdf/raiden-network-specification/latest/raiden-network-specification.pdf
- https://bitcoin.stackexchange.com/questions/43687/how-are-paths-found-in-lightning-network

Thank you!

Haoran Qi
11/02/2018