Bitcoin based messaging protocol

Akshay Kolli

University of Massachusetts, Lowell
Akshay_kolli@student.uml.edu

***Abstract*—Bitcoin technologies provide an excellent way to handle decentralized, anonymous transactions. The decentralized systems are robust, provide great security are transparent and allow for complete privacy. We focus our work on applying this technology to theorize a messaging service using crypto technologies. The sender of the message can send a message to the receiver via any server that is willing to transmit the message without having to rely on a commercial entity. We propose a mechanism using a 'smart contract' built on the block chain. The smart contract ensure that it is in the best interest of all parties involved to ensure smooth delivery of the messages and delivery of services. When receiving a message the server enters a smart contract with the sender, which the sender is able to receive upon delivery of the message to Bob.**

I. Introduction

End-to-End messaging services are considered the safest, most private and reliable way for an average member of the public to send messages. This method relies on a third party to host a server to facilitate the messaging service. We believe that this dependency on a third party such as Whats app is a security issues and forces the users to become dependent on its services. We propose a protocol, the implementation of which will allow for people to anonymously send messages to one another using the servers which incentives to anonymously provide this service, with greater privacy and reliability.

Third party services bring forward privacy issues, and there are many

II. Background

The IEEEtran class file is used to format your paper and style the text. All margins, column widths, line spaces, and text fonts are prescribed; please do not alter them. You may note peculiarities. For example, the head margin measures proportionately more than is customary. This measurement and others are deliberate, using specifications that anticipate your paper as one part of the entire proceedings, and not as an independent document. Please do not revise any of the current designations.

III. Working

Let us consider three parties, The sender (Alice), the server (Bob) and the receiver (Carol). When Alice firsts decides to communicate with Carol, Alice exchanges a series of keys with Carol. Alice will encrypt messages with the keys from Carol, so that an intermediary may not read them (End to End encryption).

When Alice is ready to send a message to Carol, Alice contacts any available server in her network, and we shall refer to the server that responds to Alice's request as Bob. Alice gives the encrypted message along with its intended recipient and "delivery fee" to Bob. "delivery fee" refers to the fee that Bob will collect upon delivery of Alice's message to Carol. Here, the fee is paid in Bitcoin, similar to its current implementation of transaction fee. Alice sends out a request to all nodes with the size of her message, recipient and the size of fees Alice is willing to pay, and any server that deems that the fee is acceptable for the size of message she wants to send can accept her request to transmit the message. Bob, if he thinks the fee is reasonable, will accept the message and generate a one time use secret key and use that key to encrypt the message and then store it.

When Carol is ready to receive the message, Carol contacts the servers in her network if there are any messages for her. This request for messages to her will propagate in the network as each node that is asked for a message check will ask other nodes. If Bob is in her network, Bob will respond to Carol's request.

Carol can verify that the message was indeed from Alice by verifying her signature. Carol cannot read the message as it has been encrypted by Bob. To decrypt the message Carol requires the secret key which Bob used to encrypt the message. Bob releases this key upon receiving the payment from Carol.

That is the underlying methodology.

Our current implementation includes a 'server', which accepts new blocks from a 'client' and adds them to the block chain. A 'client' which receives transactions from programs names 'Alice', 'Bob' and 'Carol' and mines blocks with the transactions inside them. 'Alice' and 'Carol' send and receive messages, while Bob is the server that acts as a node to facilitate the communication.

IV. Future implementations and Goals

Future implementations can use Multi signature accounts or similar techniques to achieve a more reliable method of payment, allowing payment from the sender, or allow the sender and receiver to open an account with the server to receive and make continued payments, similar to the implementation of a lightning network.

Conclusion

The preferred spelling of the word "acknowledgment" in America is without an "e" after the "g". Avoid the stilted expression "one of us (R. B. G.) thanks ...". Instead, try "R. B. G. thanks...". Put sponsor acknowledgments in the unnumbered footnote on the first page.

REFERENCES

[1] Zibin Zheng, Shaoan Xie, Hong-Ning Dai, Weili Chen, Xiangping Chen, Jian Weng, Muhammad Imran, An overview on smart contracts: Challenges, advances and platforms, Future Generation Computer Systems,

[2] Tharaka Hewa, Mika Ylianttila, Madhusanka Liyanage, Survey on blockchain based smart contracts: Applications, opportunities and challenges, Journal of Network and Computer Applications,

[3] He, Zhiguo,Blockchain Disruption and Smart Contracts, 2019 JF The Review of Financial Studies