Abstract.

Current light clients like metamask and mobile apps can be a little unsecured since they are just requesting eth_getBalance from default RPC url's. The idea is to have RPC-based wallet completely trustless by first syncing to the latest header of the beacon chain and then use the eth_getProof endpoint to get the balance plus a proof that it is actually part of the root hash that we obtained. Using Merkle Inclusion proofs to the latest block header allows us to verify that the data is correct. Another approach is we can search the header for blooms that tells us that a particular block may contain a transaction that are logged that we are interested in. We can actually request the transaction receipt block, and search in that block for bloom filters.

Research goals

Outline of research & impact

Currently the only way to verify transactions and data is through using Metamask and/or a combination of it plus Etherscan. What this means is that we are just relying on a single light protocol server to help tell us if a transaction is bad or not. This might present some issues for us down the line. A good solution for this would be to have trustless light clients that can sync to the latest header of the beacon chain and verify using Merkle Inclusion proofs to the latest block header. Making this system super lightweight and executing it to mobile phones and IOT devices would be ideal as well as applying it to all Layer 2 ecosystem would benefit a lot of people starting from developers as they would be able to utilize these layer 2 light clients to build better and more secured dapps. Users to would be able to verify transactions more easily and securely without having to utilize alot of resources for a full node or even to rely on "sort of central" light protocol servers. Also the ethereum ecosystem would be able to benefit on light clients not just verifying the transactions only for themeselves but maybe able to pass on those log messages to other users and contribute to a more decentralized system.

Output desired

At the end of the research, we hope to be able to create a prototype for a Light client for both optimistic and zk rollup Layer 2 implementation on laptops and IOS/Android devices. Then hopefully as we conduct conduct usability tests while we gather feedbacks we can improve the product, design new features and improve functionality that delivers better value to users. After that we hope to continue making it interoperable to all Layer 2 chains and bridges later on.

Another success factor is if we can allow the light client to contribute to the gossip network and therefore able to help persistently validate the other requests in the network.