Zero-Knowledge Proof for Attack Prevention in The Ethereum Blockchain

Anders Malta Jakobsen*, Oliver Holmgaard†

Abstract—This is a placeholder abstract test. The whole template is used in semester projects at Aalborg University (AAU).

1 Introduction

TODO

2 Background

In this section, we will go through some of the concepts that will be used in the rest of the paper as well as some surrounding context like attacks performed.

2.1 Ethereum and Proof of Stake

Ethereum is a blockchain platform that allows developers to create decentralized applications using smart contracts. Previously operating with a Proof of Work (PoW) consensus algorithm, Ethereum transitioned to a Proof of Stake (POS) consensus algorithm in 2022. This transition was done to reduce the energy consumption of the network and to increase the scalability of the network. The transition was done in a series of upgrades called the Ethereum 2.0 upgrade. POS works by selecting validators to create new blocks based on the amount of cryptocurrency they have staked.

2.2 Zero-Knowledge Proofs

A Zero-Knowledge Proof (ZKP) is a cryptographic method that allows one party to prove to another party that something is true without revealing any information.

Two of the subcategories of ZKPs are Zero-Knowledge Succinct Non-Interactive Argument of Knowledge (ZK-SNARK) and Zero-Knowledge Scalable Transparent Argument of Knowledge (ZK-STARK). ZK-SNARK is the more common of the two. It uses elliptic curve cryptography to create proofs based on the assumption that it is hard to find the discrete logarithm form the publicly known base point. ZK-SNARK has a start-up ritual which requires a trusted setup by all parties involved that the original proving key is destroyed as to not be able to create fake proofs. A criticism of ZK-SNARK is that it is not quantum resistant because of the reliance on elliptic curve cryptography.

ZK-STARK on the other hand, is a newer and more complex ZKP. Despite not having non-interactive in its name, ZK-STARK is also non-interactive. Different form ZK-SNARK, ZK-STARK uses hashing functions to create proofs instead of

- *All authors are affiliated with the Dept. of Computer Science, Aalborg University, Aalborg, Denmark*
- *E-mails: *amja23, †oholmg20 @student.aau.dk*

elliptic curves. This method is post-quantum secure and does not require a trusted setup. It does, however, come with a higher computational cost and is not as widely used and documented as ZK-SNARK.

3 Related Work

The usage of ZKPs in Ethereum is not a new concept. In fact, it currently uses them both on- and off-chain. The following provides a short overview of some of the already existing solutions as well as one still being in development.

3.1 MACI

In defense of a potential bribery attack, see section 2, the Ethereum blockchain implements a private voting system called Minimum Anti-Collusion Infrastructure (MACI) [1, 2].

What MACI does is essentially hiding what each person has voted for. It does so by demanding the voters to send their votes encrypted to a central coordinator. This coordinator constructs ZK-SNARK proofs, which verifies that all messages were processed correctly, and that the final result corresponds to the sum of all valid votes.

As votes are now hidden, the adversary is not able, by oneself, to prove that the bribee voted in way of said bribery. Though the bribee could decrypt their own message and show the vote to the adversary.

MACI has fixed this problem by implementing public key switching. This means that a voter can request a new public key. In addition to this, a vote is only valid if it uses the most recent public key of the voter. Therefore, a bribee can show its first vote obeying the adversary, generate a new public key, and send a new, now honest, vote. The old vote will then become invalid as it uses a deprecated public key.

3.2 Off-chain roll-ups

As a result of Ethereum's popularity, the network could easily get congested if developers were not actively trying to distribute computations [3].

One of the first ideas was to introduce an on-chain technique called sharding. It is a technique where the database would be split into different parts between subsets of validators. Sharding was never deployed on the blockchain though, and instead Ethereum uses off-chain solutions to off-load computations. The idea of off-chain solutions, called L2-roll-ups, is that users commit their work to off-chain nodes. These perform the computations, thereafter they submit the work to the Ethereum Mainnet chain.

One of these solutions is called a Zero-Knowledge Rollup (ZK-rollup).

Write about scalability, zk-evm, and Linea

3.3 Whisk

4 Analysis

This is a potential analysis section.

5 Discussion

This is a potential discussion section.

6 Conclusion

This is a potential conclusion section.

References

[1] ethereum.org, "What are zero-knowledge proofs?," 2024, Accessed: 15-10-2024.

[2] K. Charbonnet, "A technical introduction to maci 1.0," 2022, Accessed: 15-10-2024.

[3] ethereum.org, "Scaling," 2024, Accessed: 15-10-2024.

Appendix A

Compiling in draft

You can also compile the document in draft mode. This shows todos, and increases the space between lines to make space for your supervisors feedback.

Appendix B

Attacks on Ethereum

B.1 Reorg

B.2 DoS

B.3 Balancing Attack

B.4 Finality Attack (Bouncing Attack)

B.5 Avalanche Attack

B.6 Bribery

B.7 Staircase Attack