|  |  |  |
| --- | --- | --- |
|  |  | Hash-Based Multi-Signatures for Post-Quantum Ethereum  *By QuantumShield* |
| Problem Statement: With the development of quantum computing, current cryptographic systems, such as the BLS signatures used in Ethereum’s proof-of-stake, will no longer be secure.  As BLS relies on the discrete logarithm problem in elliptic curve groups, while being infeasible to solve with classical computers, quantum computers can solve DLPs using Shor’s algorithm, allowing us to compute BLS private keys from public keys in effective time.  The paper “[Hash-Based Multi-Signatures for Post-Quantum Ethereum](https://eprint.iacr.org/2025/055)” proposes an alternative based on quantum resistant cryptographic hash functions that remain secure as they do not rely on elliptical curves for security.  Although researchers have created a Rust based proof of concept, it is heavily plagued with performance and compatibility issues. This project aims to implement the scheme in C to maximise on chain performance and compatibility, serving as a successor for Ethereum’s current BLS security. Our Team:  |  |  | | --- | --- | | Achintha Namaratne | z5413821 | | Sam Marinovich | z5480700 | | Yifei Jia | z5665143 | | Zihan Xu | z5489858 | | Jinye Hu | z5513840 | |  | Tentative timelines  |  |  | | --- | --- | | Week | Item | | Week 5 | Analysing the initial paper and further research on the topic. | | Week 6 | Designing and coding our C based implementation using pre-existing hashing libraries with proven security such as, Libkeccak. | | Week 7 | | Week 8 | Benchmarking using Supercop and final design verification using Valgrind. | | Week 9 | Complete and finalise our findings in the report | | Week 10 | Buffer to account for any unexpected issues |  Responsibilities  |  |  | | --- | --- | | Team member | Role | | Achintha Namaratne | Base implementation | | Sam Marinovich | Algorithm Optimizations | | Jinye Hu | Performance Optimizations | | Yifei Jia | Testing and Performance Evaluation | | Zihan Xu | Reporting | |