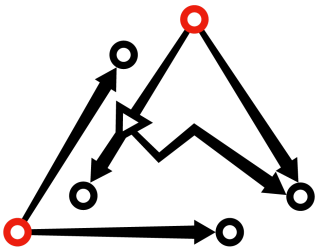
ProgPow Algorithm
Security Audit Proposal
Ethereum Cat Herders



Least Authority
PRIVACY MATTERS

Scope

Ethereum Cat Herders have requested Least Authority perform a security audit of ProgPow, a Proof-of-Work algorithm for Ethash, in order to verify the security of the algorithm and provide clear metrics about its performance.

The following code repositories are considered in-scope for the review:

- ProgPow: https://github.com/ifdefelse/ProgPOW

However, third party vendor code is considered out of scope.

Areas of Concern

The following are areas of concern that will be investigated during the audit, along with any similar potential issues:

- (1) The expected effects of ProgPoW on the security of Ethereum vis-a-vis:
  - Security of the algorithm
  - Attack surface
  - Cost of 51% attack
  - Other security risks that may result from a change to ProgPoW
- (2) ProgPoW meeting the claimed goal of ASIC resistance
- (3) Known methods to speed up the calculation of the hash function
- (4) Length of time it would it take to create a ProgPoW ASIC (if R&D begins immediately) and the expected efficiency gains from the first generation of said ASICs
- (9) Expected changes to hash power and miner balance
- (10) Identify any potential advantages or disadvantages that ProgPoW would present in comparison to Ethash in terms of “fair mining” and evaluate any potential uneven distribution of advantage between manufacturers or third-parties
- (11) Other effects impacting the ecosystem at large (distribution, economies of scale, cost, etc.) and other externalities of such a change
- (13) ProgPoW’s ability to provide better decentralization
- Anything else as identified during the initial analysis phase

Schedule

The following schedule is planned:

- **March 25 - April 26**: Code review completed
- **May 1**: Delivery of Initial Audit Report
- **May 27 - 30:** Verification completed

- **May 31:** Delivery of Final Audit Report

If there is any delay in approving or starting of the project or if both teams agree certain areas of concern should receive additional attention, the project schedule may be modified or extended. Also, depending on the resolutions and mitigations proposed, consulting beyond the end of the project may be helpful.

Project Phases

Least Authority will investigate the areas of concern and assist with resolving any issues discovered during the audit with the following phases.

Review Phase

- **Project Discovery and Planning:** We would like to start this project with a kick-off meeting to get an overview of the service from the perspective of the company and the technical leads. We will also discuss what resources need to be shared, revisit the project schedule and agree upon necessary communication channels.
- **Exploration and Implementation Investigation:** In this phase we read design documentation, review other audit results or resources, and generally prepare for where vulnerabilities may be present. We will log our efforts to find vulnerabilities and related potential issues. Throughout the project we will hold meetings and share notes as is appropriate, depending on the process and focus of the investigations. We will also capture notes on how any issues that we do find could be mitigated or resolved.
- **Initial Audit Report Delivery:** At this point in our schedule, we wrap up our investigative work, document any unresolved issues or open questions and suggested resolutions in the report. This report is intended for internal use, only. A meeting to discuss the results of the report is recommended.

Resolution and Verification Phase

- **Remediation:** After delivering the report, we will be available for consultation, as needed. The mitigation and remediation recommendations should be scrutinized by the developers and deployment engineers, and successful mitigation and remediation is an ongoing collaborative process after we deliver our report, and before the details are made public.
- **Verification:** When the development team believes that all issues with sufficient impact have been addressed, we will review the system to confirm that these changes are made. It may be the case that some resolutions need to be discussed further because of the nature of design, code, operational deployment, and other engineering changes, as well as mistakes or misunderstandings.
- **Updated Audit Report Delivery:** We will update our report to reflect the resolved or mitigated status of any issues in the initial report. We will collaborate with the developers to make sure we all can agree that the report is appropriate to share publicly, but still transparent about and true to our original findings. Only after we agree with the development team that all vulnerabilities with sufficient impact have been appropriately mitigated do we publish our results.

Deliverables

An Initial Audit Report will be provided after the initial review and a Final Audit Report will be provided at the conclusion of verification phase of the project. Throughout the project we will hold meetings and share notes as is appropriate, depending on the process and focus of the investigations. These deliverables could also include code suggestions, instructions, or other forms of supporting material.

Publication & Responsible Disclosure

After verification is completed and the Final Audit Report is delivered, the team will publish an audit overview report that will be posted on the Least Authority website blog, and can optionally be posted on the Company website (along with the Final Audit Report, if Company chooses). Both teams will have an opportunity to review the summary post before it is published.

Least Authority is committed to its clients and will work to ensure companies have the appropriate information and time needed to address threats and vulnerabilities identified during our initial audit. In an effort to stay true to our mission of promoting ethical practices as it relates to security and privacy, Least Authority also has a responsibility to the community utilizing the tools and technologies which we audit. For this reason, Least Authority tries to adhere to a principle of responsible disclosure, followed by full disclosure as the last resort. In the event that Company does not respond to requests for verification of the issues investigated and reported in the Initial Audit Report, Least Authority will conduct an automatic verification **90 days** following delivery of the Initial Audit Report prior to publication of our findings in a Final Audit Report, as is outlined above.

Project Team

Jean-Paul Calderone
Jean-Paul has been programming for over fifteen years and programming in Python for over ten. He was Divmod, Inc.'s second employee and worked there for years building context-sensitive, highly integrated communications servers. He also did brief stints at NIH and Deutsche Bank. He is best known as a core developer on the Twisted project.

Gordon Hall
Gordon is a free software activist and engineer. They currently maintains the ORC project, an onion-routed distributed document storage system. Gordon is a founding director of Counterpoint, a non-profit hackerspace outside of Atlanta that builds and teaches tools for security, decentralization, and effective action.

Meejah
Author of txtorcon, core developer of Crossbar.io/Autobahn, and contributor to other open source projects such as Twisted and Tor. Mike has 20 years of development experience in many fields.

Ramakrishnan Muthukrishnan
Ramakrishnan (Ramki) is a Debian developer and lives in Bangalore, India. He has contributed to a bunch of Free software projects like GNU Emacs, Linux kernel and the GNU Radio. He likes to tinker with low-level system software and also enjoys learning and playing with Functional Programming.

James Prestwich
James believes that better technical systems create better social systems. He focuses on identifying and addressing flaws in incentivized systems. James is currently researching technical and regulatory frameworks for atomic cross-chain financial contracts with the aim of creating trustless derivatives markets.

Dominic Tarr
Dominic has been using node.js since version 0.2.1 and has published hundreds of open source javascript modules, known for his work in streams, databases, and database replication this lead to an interest in cryptography and security and developing the secure-scuttlebutt protocol.

Jan Winkelmann

Jan is a software engineer primarily working in Go. His interests lie in the intersection of systems design, security and cryptography. He is currently working on IPFS and secure scuttlebutt to advance the state of secure decentralized tools for collaboration.

Hind Abu-Amr

Hind happened upon managing software design project and program management after studying politics and economics. She eventually decided to pursue a career in human rights, humanitarian, and development work which would bring her back to Europe, the Middle East, and Africa - where she grew up. Her passion for human rights (including privacy, security, and freedom of speech) and her experience in management in the "tech" world lead Hind to pursuing opportunities in the data privacy and security world.

Liz Steininger

Liz is a supporter of open source software that encourages transparency and access to information, along with software that enables individuals to freely express themselves and retain the ability to control their own information. She has over 15 years of experience as a Program and Project Manager, Strategist and Analyst working towards these goals.

Least Authority Audit Team: Additional Reviewing & Support

The Least Authority team has skills for reviewing code in C, C++, Python, Haskell, Rust, Node.js, Solidity and JavaScript for common security vulnerabilities and specific attack vectors. The team has reviewed implementations of cryptographic protocols and distributed system architecture, including in cryptocurrency, blockchains, payments, and smart contracts. Additionally, the team can utilize various tools to scan code and networks and build custom tools as necessary.