ANASTASIA LABS

PROJECT CLOSE-OUT REPORT

Project Number	1200175
Project manager	Philip Disarro
Date Started	21-Oct-2024
Date Completed	05-May-2025

ANASTASIA LABS

Contents

List of KPIs **1**
Challenge KPIs 1
Project KPIs 2
Key achievements **3**
Key learnings **4**
Next steps **5**
Final thoughts **5**
Resources **5**
Project 5
Close-out Video 5

Project Name: Opshin Audit
URL: https://projectcatalyst.io/funds/12/cardano-open-developers/opshin-audit

List of KPIs

Challenge KPIs

- **Improve Smart Contract Security:** By conducting an in-depth audit of the OpShin language, the project directly addresses security risks associated with smart contract development. The expected outcome is a measurable reduction in critical and high-severity vulnerabilities, making DApps built with OpShin safer for users.
- **Enhance Developer Knowledge and Best Practices:** The project provides clear, actionable best practices and development guidelines for using OpShin effectively. This helps elevate the quality of code across the ecosystem and supports developer onboarding.
- **Drive Ecosystem Trust and Maturity:** The public audit and transparent communication of findings will reinforce OpShin's position as a secure and trustworthy platform for smart contract development. By identifying edge cases and potential problems early in OpShin, the audit enables developers to avoid time-consuming debugging and security rework.

Project KPIs

- **Reduction in Critical Vulnerabilities:** As part of our comprehensive audit of the OpShin language, we identified 9 critical and 16 major vulnerabilities within the codebase. These findings highlight previously unaddressed security risks that could impact smart contract reliability and user safety. By surfacing these issues early and providing detailed recommendations, the audit strengthens the foundation for secure smart contract development using OpShin. This outcome not only enhances developer confidence in using OpShin but also helps reduce the risk of exploit-prone contracts being deployed on mainnet.
- **Audit Delivered with Categorized Findings:** The audit report provides a detailed breakdown of each identified issue, categorized by Security, Performance, Maintainability, and Usability, and classified by severity as Critical, High, Medium, or Informational. Each finding includes an assessment of its potential risk or exploitability, along with a clear explanation of its impact on smart contract behavior and user assets. Wherever applicable, the report outlines steps to reproduce the issue, along with practical recommendations and mitigation strategies for each finding.
- **Number of Publicly Released Outputs:** As part of OpShin's commitment to transparency and community impact, several key deliverables have been made publicly accessible. These include a comprehensive audit report, detailed edge case documentation, and the public dissemination of the finalized report across multiple platforms. All outputs are clearly written and freely available to the community. These contributions not only demonstrate the value delivered through the audit but also support ecosystem-wide learning, adoption, and long-term growth.

Key achievements

- **Comprehensive Audit Report Completed:** Delivered a full audit report of the OpShin codebase, including a quantitative breakdown of vulnerabilities across critical, high, medium, and low severities.
- **Edge Cases Identified and Documented:** Successfully uncovered and described multiple unique edge cases relevant to smart contract development, enhancing code resilience and safety.
- **High Code Coverage Achieved:** Measured and reported the percentage of OpShin code covered by unit tests, strengthening test reliability and identifying gaps.
- **Detailed and Actionable Recommendations:** Provided clear, prioritized recommendations for improvements, addressing both immediate security concerns and long-term maintainability.
- **Collaboration and Validation by the OpShin Team:** Collected structured feedback from the core OpShin team and confirmed that all critical and high-severity issues were addressed via commits, pull requests, or updated documentation.

Key learnings

- **Value of Early Edge Case Identification:** Many vulnerabilities stemmed from uncommon or edge-case logic paths, reinforcing the importance of proactive edge case analysis in language-level audits.
- **Manual Review Remains Crucial:** While automated tools and unit tests offered initial coverage, several high-impact issues were only discovered through deep manual inspection, especially in performance and maintainability areas.
- **Collaboration Drives Quality:** Direct engagement with the OpShin team significantly improved issue resolution and knowledge transfer. Iterative feedback ensured alignment on severity, prioritization, and practical fixes.
- **Documentation is a Force Multiplier:** The creation of a public-facing audit report, updated OpShin book, and edge case documentation not only delivered immediate value but also contributed to ecosystem trust and onboarding for future developers.
- **Security and Developer Experience Go Hand-in-Hand:** Improving language reliability also improves developer confidence and productivity. A secure, well-documented language reduces debugging and accelerates safe contract development.

Next steps

- To be filled by OpShin team

Final thoughts

This audit has been a valuable step toward strengthening the security, reliability, and developer experience of the OpShin language. By uncovering vulnerabilities, documenting critical edge cases, and providing actionable recommendations, we've laid a strong foundation for safer smart contract development within the Cardano ecosystem. The collaborative engagement with the OpShin team throughout the process ensured that findings were not only identified but also meaningfully addressed. As the language continues to evolve, we hope this work serves as both a reference and a catalyst for continued improvements, open collaboration, and greater adoption across the community.

Resources

Project

Main Github Repo
Catalyst Proposal

Close-out Video

Link Placeholder