Security Audit Report

HOOT Token

4/2/2023

Revision: 4/2/2023

PREPARED FOR:
A Birds Nest, Hoot.Finance

ARCADIA CONTACT INFO
Email: audits@arcadiamgroup.com

Telegram: https://t.me/thearcadiagroup

Table of Contents

Executive Summary

1. Introduction and Audit Scope

A Representative Party of Hoot Finance ("**CLIENT**") engaged The Arcadia Group ("Arcadia"), a software development, research, and security company, to conduct a review of the following HOOT smart contracts on **Ethereum** at contract address **0x12A7530D6f9e1a9B0351D78aB711f7C2c033873A**

2. Audit Summary

a. Audit Methodology

Arcadia completed this security review using various methods primarily consisting of dynamic and static analysis. This process included a line-by-line analysis of the in-scope contracts, optimization analysis, analysis of key functionalities and limiters, and reference against intended functionality.

The followings are the steps we perform during smart contract engagements:

- Investigating the project and its technical architecture overview through its documentation
- Understanding the overview of the smart contracts, the functions of the contracts, the inheritance, and how the contracts interface with each others thanks to the graph created by Solidity Visual Developer
- Manual smart contract audit:
    - Review the code to find any issue that could be exploited by known attacks listed by Consensys
    - Identifying which existing projects the smart contracts are built upon and what are the known vulnerabilities and remediations to the existing projects
    - Line-by-line manual review of the code to find any algorithmic and arithmetic related vulnerabilities compared to what should be done based on the project's documentation
    - Find any potential code that could be refactored to save gas
    - Run through the unit-tests and test-coverage if exists
- Automated smart contract audit:
    - Scanning for vulnerabilities in the smart contracts using Static Code Analysis Software
    - Making a static analysis of the smart contracts using Slither
- Additional review: a follow-up review is done when the smart contracts have any new update. The follow-up is done by reviewing all changes compared to the audited commit revision and its impact to the existing source code and found issues.
- In this engagement, the primary contract in question utilizes a templated standard ERC20 Implementation ("StandardERC20"), as such, review for modification from the standard was the primary focus. Review for functionality such as the ability to mint, burn or upgrade the contracts occurred.

b. Summary

Severity Rating	Number of Original Occurrences	Number of Remaining Occurrences
CRITICAL	0	0
HIGH	0	0
MEDIUM	0	0
LOW	0	0
INFORMATIONAL	0	0

Conclusion

The contract is a StandardERC20 implementation without any unexpected mint, burn or upgrade functionality.

Disclaimer