Smart Contract Security Audit

RSE (RSE)

Contract Address: 0xe86d12aaA412F9a28F2208293314084a3339915B

www.solidityshark.com

Disclaimer

This is a limited report on our findings based on our analysis, in accordance with good industry practice as at the date of this report, in relation to cybersecurity vulnerabilities and issues in the framework and algorithms based on smart contracts, the details of which are set out in this report. In order to get a full view of our analysis, it is crucial for you to read the full report. While we have done our best in conducting our analysis and producing this report, it is important to note that you should not rely on this report and cannot claim against us on the basis of what it says or doesn't say, or how we produced it, and it is important for you to conduct your own independent investigations before making any decisions. We go into more detail on this in the below disclaimer below – please make sure to read it in full.

DISCLAIMER: By reading this report or any part of it, you agree to the terms of this disclaimer. If you do not agree to the terms, then please immediately cease reading this report, and delete and destroy any and all copies of this report downloaded and/or printed by you. This report is provided for information purposes only and on a non-reliance basis and does not constitute investment advice. No one shall have any right to rely on the report or its contents, and Solidity Shark and its affiliates (including holding companies, shareholders, subsidiaries, employees, directors, officers and other representatives) (Solidity Shark) owe no duty of care towards you or any other person, nor does Solidity Shark make any warranty or representation to any person on the accuracy or completeness of the report. The report is provided "as is", without any conditions, warranties or other terms of any kind except as set out in this disclaimer, and Solidity Shark hereby excludes all representations, warranties, conditions and other terms (including, without limitation, the warranties implied by law of satisfactory quality, fitness for purpose and the use of reasonable care and skill) which, but for this clause, might have effect in relation to the report. Except and only to the extent that it is prohibited by law, Solidity Shark hereby excludes all liability and responsibility, and neither you nor any other person shall have any claim against Solidity Shark, for any amount or kind of loss or damage that may result to you or any other person (including without limitation, any direct, indirect, special, punitive, consequential or pure economic loss or damages, or any loss of income, profits, goodwill, data, contracts, use of money, or business interruption, and whether in delict, tort (including without limitation negligence), contract, breach of statutory duty, misrepresentation (whether innocent or negligent) or otherwise under any claim of any nature whatsoever in any jurisdiction) in any way arising from or connected with this report and the use, inability to use or the results of use of this report, and any reliance on this report.
The analysis of the security is purely based on the smart contracts alone. No applications or operations were reviewed for security. No product code has been reviewed.

Audit Overview

Audited Project: RSE
Contract Address: 0xe86d12aaA412F9a28F2208293314084a3339915B
Languages: Solidity (Smart contract)
Platforms and Tools: Remix IDE, Truffle, Truffle Team, Ganache, Solhint, VScode, Mythril, Contract Library

Total supply: 10,000,000
Token ticker: RSE
Decimals: 9
Compiler Version: 0.8.4+commit.c7e474f2
Contract Deployer Address:
0x4bf93a2eb509d6f8008e8aa13dc794ed1d8c24d1
Optimization Enabled: Yes with 200 runs
Blockchain: Binance Smart Chain Project

The audit items and results:
Other unknown security vulnerabilities are not included in the audit responsibility scope

Audit Result: Passed
Audit Date: May 26, 2022
Audit Team: Solidity Shark
https://www.solidityshark.com

Introduction

This Audit Report mainly focuses on the overall security of the project's Smart Contract. With this report, we have tried to ensure the reliability and correctness of their smart contract by complete and rigorous assessment of their system's architecture and the smart contract codebase.

Auditing Approach and Methodologies applied by the Solidity Shark team has performed a rigorous review of the project starting with analyzing the code design patterns in which we reviewed the smart contract architecture to ensure it is structured and safe use of third-party smart contracts and libraries. Our team then performed a formal line by line inspection of the Smart Contract to find any potential issue like race conditions, transaction-ordering dependence, timestamp dependence, and denial of service attacks.

In Automated Testing, we tested the Smart Contract with our in-house developed tools to identify vulnerabilities and security flaws. The code was reviewed in collaboration of our multiple team members and this included -

- Testing the Smart Contract to determine proper logic has been followed throughout the whole process.
- Analyzing the complexity of the code in depth and detailed, manual review of the code, line by-line.
- Deploying the code on testnet using multiple clients to run live tests.
- Analyzing failure preparations to check how the Smart Contract performs in case of any bugs and vulnerabilities.
- Checking whether all the libraries used in the code are on the latest version.
- Analyzing the security of the on-chain data.

Audit Goals

The focus of the audit was to verify that the Smart Contract System is secure, resilient and working according to the specifications. The audit activities can be grouped in the following three categories: Security Identifying security related issues within each contract and the system of contract. Sound Architecture Evaluation of the architecture of this system through the lens of established smart contract best practices and general software best practices. Code Correctness and Quality A full review of the contract source code.

The primary areas of focus include:

- Accuracy
- Readability
- Sections of code with high complexity
- Quantity and quality of test coverage

Issue Categories

Every issue in this report was assigned a severity level from the following

Issues Checking Status

High Severity	Medium Severity	Low Severity
0	0	0
High Severity Issues found within the smart contract code	Medium Severity Issues found within the smart contract code	Low Severity Issues found within the smart contract code

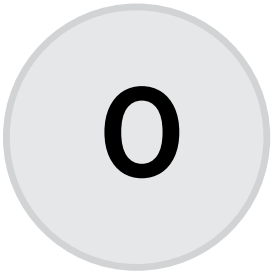
Issue Description	Status
Compiler errors.	Passed
Race conditions and Reentrancy. Cross-function race conditions.	Passed
Possible delays in data delivery.	Passed
Oracle calls.	Passed
Front running.	Passed
Timestamp dependence	Passed
Integer Overflow and Underflow.	Passed
DoS with Revert.	Passed
DoS with block gas limit.	Passed
Methods execution permissions.	Passed
Economy model of the contract.	Passed
Private user data leaks	Passed
Malicious Event log.	Passed
Scoping and Declarations.	Passed
Uninitialized storage pointers.	Passed
Arithmetic accuracy.	Passed
Design Logic.	Passed
Cross-function race conditions.	Passed
Safe Open Zeppelin contracts implementation and usage.	Passed
Fallback function security.	Passed

Owner Functions

The owner can call the **renounceOwnership** function.
The owner can call the **transferOwnership** function.
The owner can call the **excludeFromReward** function.
The owner can call the **includeInReward** function.
The owner can call the **excludeFromFee** function.
The owner can call the **includeInFee** function.
The owner can call the **setTaxFeePercent** function.
The owner can call the **setSwapAndLiquifyEnabled** function.

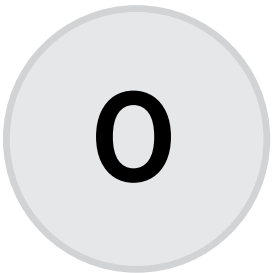
Manual Audit

In the Manual Audit of the project's smart contract the code was reviewed & read line-by-line by our developers. We used Remix IDE's, JavaScript VM and Kovan networks as tools to complete this section of the audit.



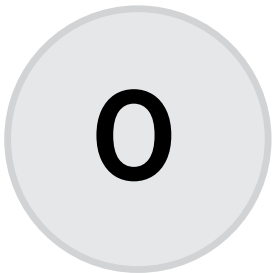
Critical Severity Issues

No Critical severity issues found.



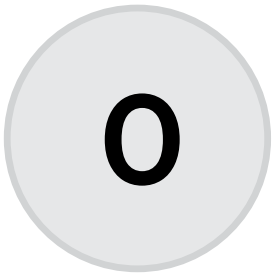
High Severity Issues

No High severity issues found.



Medium Severity Issues

No Medium severity issues found.



Low Severity Issues

No Low severity issues found.

Remix Complier Warnings

We check for Remix Compiler warnings within this section . If there are any errors the contract cannot be compiled and deployed. No issues were found at this time.

The Stuff You Care About

Ownership

The Smart Contract is Owned

Mint Function

No Mint Function Found

Ability to Set Fees?

Owner Cannot Set Fees Above 25%

Max Transaction Amount

Owner Cannot Set Transaction Amount

Blacklist

Owner Cannot Block Wallet Addresses

Pause

Owner Cannot Pause Trading

The Smart Contract Does Not
Contain Any High Severity Issues.

Verify this report on telegram: @SolidityShark

This report has been prepared by:

www.SolidityShark.com

t.me/SolidityShark