

Entry Token (ENTRY)

2 MAY 2018 / TABLE OF CONTENTS

INTRODUCTION	2
AUDIT METHODOLOGY	3
Design Patterns	3
Static Analysis	3
Manual Analysis	3
Network Behavior	3
Contracts Reviewed	4
Remediation Audit	4
AUDIT SUMMARY	5
Analysis Results	5
Test Results	5
Token Allocation Results	5
Explicit Vulnerability Check Results	5
ISSUES DISCOVERED	6
Severity Levels	6
Issues	6
CONCLUSION	6

INTRODUCTION

CoinMercenary provides comprehensive, independent smart contract auditing.

We help stakeholders confirm the quality and security of their smart contracts using our comprehensive and standardized audit process. Each audit is unbiased and verified by multiple reputable auditors.

The scope of this audit was to analyze and document the Entry Token (ENTRY) token generation event contract.

This audit provides practical assurance of the logic and implementation of the contract.

AUDIT METHODOLOGY

CoinMercenary audits consist of four categories of analysis.

Design Patterns

We first inspect the overall structure of the smart contract, including both manual and automated analysis.

The design pattern analysis checks appropriate test coverage, utilizes a linter to ensure consistent style and composition, and code comments are reviewed. Overall architecture and safe usage of third party smart contracts are checked to ensure the contract is structured in a way that will not result in future issues.

Static Analysis

The static analysis portion of our audit is performed using a series of automated tools, purposefully designed to test the security of the contract. These tools include:

- **Manticore** - Dynamic binary analysis tool with EVM support.
- **Mythril** - Reversing and bug hunting framework for the Ethereum blockchain.
- **Oyente** - Analyzes Solidity code to find common vulnerabilities.
- **Solgraph** - DOT graph creation for visualizing function control flow of a Solidity contract to highlight potential security vulnerabilities.

Data flow and control flow are also analyzed to identify vulnerabilities.

Manual Analysis

Performing a hands on review of the smart contract to identify common vulnerabilities is the most intensive portion of our audit. Checks for race conditions, transaction-ordering dependence, timestamp dependence, and denial of service attacks are part of our standardized process.

Network Behavior

In addition to our design pattern check, we also specifically look at network behavior. We model how the smart contract will operate once in production,

then determine the answers to questions such as: how much gas will be used, are there any optimizations, how will the contract interact?

Contracts Reviewed

On May 2, 2018, the following contract files and their respective SHA256 fingerprints were reviewed:

Filename	SHA256 Fingerprint
EntryToken.sol	cbf889c2e1c883691b3014777c6e1fa2657bacd11eb07cdf1520d712ef10e686

Remediation Audit

Not required.

Filename	SHA256 Fingerprint

AUDIT SUMMARY

The contracts have been found to be free of security issues.

Analysis Results

	Initial Audit	Remediation Audit
Design Patterns	Passed	Not required
Static Analysis	Passed	Not required
Manual Analysis	Passed	Not required
Token Allocation	Passed	Not required
Network Behavior	Passed	Not required

Test Results

No test coverage available.

Token Allocation Results

- Hard cap defined in contract
- Stage caps match whitepaper
- Locking contract (TokenTimelock) available
- Contract presale and ICO dates match whitepaper
- Allocations pre-defined in contract

Explicit Vulnerability Check Results

Known Vulnerability	Results
Parity Multisig Bug 2	Not vulnerable
Callstack Depth Attack	Not vulnerable
Transaction-Ordering Dependence	Not vulnerable
Timestamp Dependency	Not vulnerable
Re-Entrancy Vulnerability	Not vulnerable

ISSUES DISCOVERED

Issues below are listed from most critical to least critical. Severity is determined by an assessment of the risk of exploitation or otherwise unsafe behavior.

Severity Levels

- **Informational** - No impact on the contract.
- **Low** - Minimal impact on operational ability.
- **Medium** - Affects the ability of the contract to operate.
- **High** - Affects the ability of the contract to work as designed in a significant way.
- **Critical** - Funds may be allocated incorrectly, lost or otherwise result in a significant loss.

Issues

No issues discovered.

CONCLUSION

The reviewed smart contracts are well crafted and follow common security practices. No critical problems have been found.

The care and attention to detail by the Entry Token team for their token generation event shows their commitment to security.

We're proud to have Entry Group as a customer, and look forward to seeing their upcoming success.