HOSHO

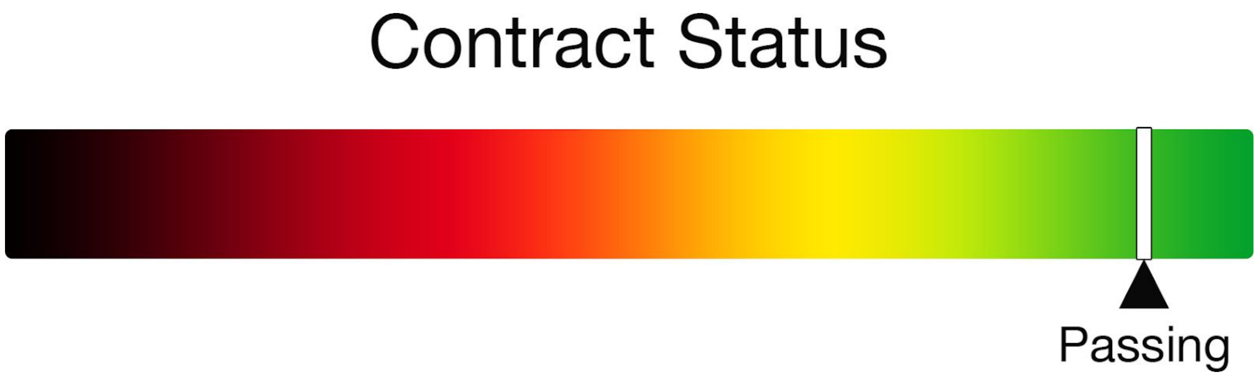
LibraCredit Airdrop Contract Audit

by Hosho

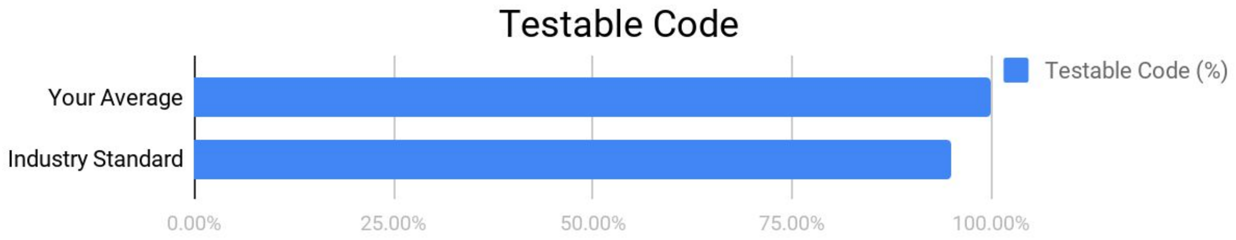
May 28th, 2018

Executive Summary

This document outlines the overall security of LibraCredit's smart contract as evaluated by Hosho's Smart Contract auditing team. The scope of this audit was to analyze and document LibraCredit's contract codebase for quality, security, and correctness.



No issues found in the AirdropLibraToken contract and supporting contracts. (See Complete Analysis)



Testable code coverage is 100.00% which is higher than industry average. (See Coverage Report)

It should be noted that this audit is not an endorsement of the reliability or effectiveness of the contract, rather limited to an assessment of the logic and implementation. In order to ensure a secure contract that's able to withstand the Ethereum network's fast-paced and rapidly changing environment, we at Hosho recommend that the LibraCredit Team put in place a bug bounty program to encourage further and active analysis of the smart contract.

Table Of Contents

1. Auditing Strategy and Techniques Applied

The Hosho Team has performed a thorough review of the smart contract code, the latest version as written and updated on May 13, 2018. All main contract files were reviewed using the following tools and processes. (See All Files Covered)

Throughout the review process, care was taken to ensure that the token contract:

- Documentation and code comments match logic and behavior;
- Follows best practices in efficient use of gas, without unnecessary waste;
- Uses methods safe from reentrance attacks; and
- Is not affected by the latest vulnerabilities

The Hosho Team has followed best practices and industry-standard techniques to verify the implementation of LibraCredit's contract. To do so, reviewed line-by-line by our team of expert pentesters and smart contract developers, documenting any issues as they were discovered. Part of this work included writing a unit test suite using the Truffle testing framework. In summary, our strategies consist largely of manual collaboration between multiple team members at each stage of the review:

1. Due diligence in assessing the overall code quality of the codebase.
2. Cross-comparison with other, similar smart contracts by industry leaders.
3. Testing contract logic against common and uncommon attack vectors.
4. Thorough, manual review of the codebase, line-by-line.
5. Deploying the smart contract to testnet and production networks using multiple client implementations to run live tests.

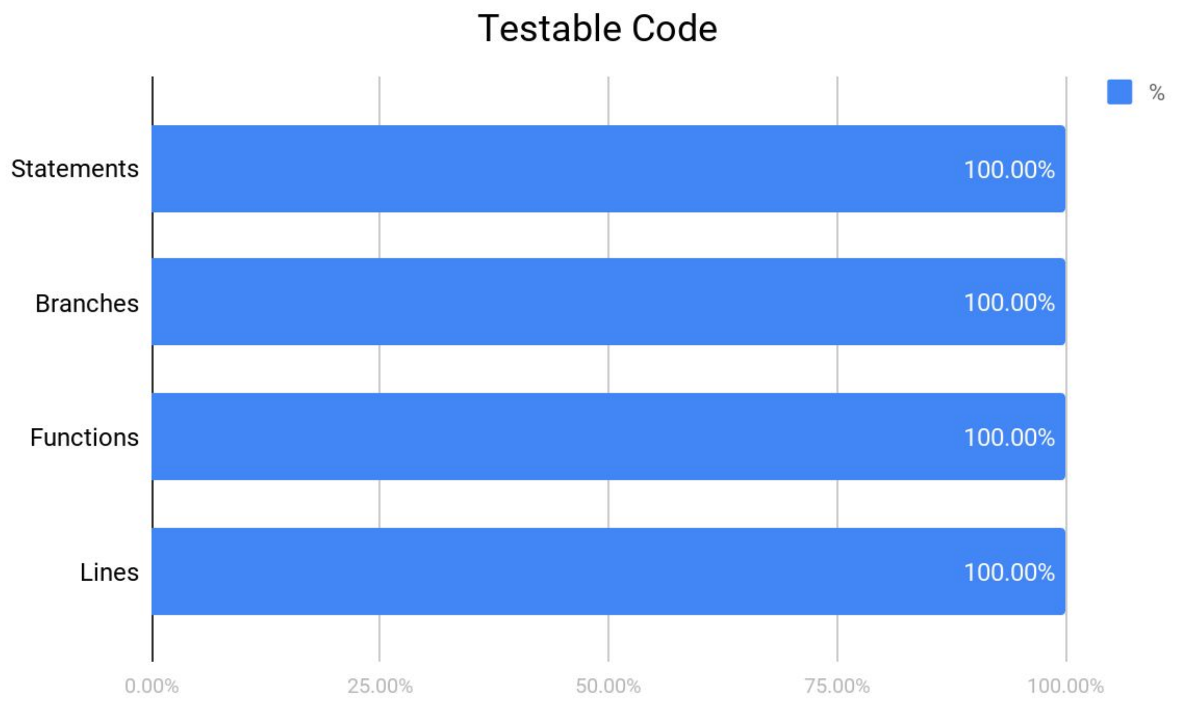
2. Structure Analysis and Test Results

2.1. Summary

This report covers the LibraCredit Airdrop functionality as well as the other necessary contracts in the codebase needed to test the airdrop, including the LibraToken. The airdrop is a time based drop with an adjustable end time. Proper protections have been put in place to ensure that only administrators can take actions such as drop the tokens or update the end time of the drop.

2.2 Coverage Report

As part of our work assisting LibraCredit in verifying the correctness of their contract code, our team was responsible for writing a unit test suite using the Truffle testing framework.



For individual files see Additional Coverage Report

2.3 Failing Tests

No failing tests.

See Test Suite Results for all tests.

3. Complete Analysis

For ease of navigation, sections are arranged from most critical to least critical. Issues are tagged “Resolved” or “Unresolved” depending on whether they have been fixed or addressed. Furthermore, the severity of each issue is written as assessed by the risk of exploitation or other unexpected or otherwise unsafe behavior:

- **Informational** - The issue has no impact on the contract’s ability to operate.
- **Low** - The issue has minimal impact on the contract’s ability to operate.
- **Medium** - The issue affects the ability of the contract to operate in a way that doesn’t significantly hinder its behavior.
- **High** - The issue affects the ability of the contract to compile or operate in a significant way.
- **Critical** - The issue affects the contract in such a way that funds may be lost, allocated incorrectly, or otherwise result in a significant loss.

No issues were discovered within the airdrop contract and any of the supporting contracts.

4. Closing Statement

We are grateful to have been given the opportunity to work with the LibraCredit Team.

The team of experts at Hosho, having backgrounds in all aspects of blockchain, cryptography, and cybersecurity, can say with confidence that the LibraCredit contract is free of any critical issues.

The statements made in this document should not be interpreted as investment or legal advice, nor should its authors be held accountable for decisions made based on them.

We at Hosho recommend that the LibraCredit Team put in place a bug bounty program to encourage further analysis of the smart contract by other third parties.

5. Test Suite Results

Coverage Report:

```
 Ensure 'LibraToken' defines the ERC20 Token Standard Interface
   √ Should have the correct 'name' definition
   √ Should have the correct 'approve' definition
   √ Should have the correct 'totalSupply' definition
   √ Should have the correct 'transferFrom' definition
   √ Should have the correct 'decimals' definition
   √ Should have the correct 'balanceOf' definition
   √ Should have the correct 'symbol' definition
   √ Should have the correct 'transfer' definition
   √ Should have the correct 'allowance' definition
   √ Should have the correct 'Transfer' definition
   √ Should have the correct 'Approval' definition

 Contract: Specific Tests for LibraToken
   √ Should require a transfer
   √ Should airdrop tokens (797ms)
   √ Should airdrop tokens to multiple addresses (1085ms)
   √ Should required number of addresses and amount of transfers to be equal (105ms)
   √ Should not drop 0 tokens (103ms)
   √ Should require onlyOwnerOrAdmin to drop tokens (75ms)
   √ Should add an account as admin (108ms)
   √ Should remove an account as admin (194ms)
   √ Should only remove admins (93ms)
   √ Should check remaining balance of supply (712ms)
   √ Should check total distributed
```

```
    √ Should check all addresses that were distributed to
    √ Should check how much an address was distributed
    √ Should change airdrop end time (107ms)
    √ Should fail if outside airdrop event time (105ms)


  Contract: Ownership Tests for AirdropLibraToken
    Deployment
      √ Should deploy with the owner being the deployer of the contract
    Transfer
      √ Should not allow a non-owner to transfer ownership (65ms)
      √ Should not allow the owner to transfer to 0x0 (81ms)
      √ Should renounce ownership (91ms)
      √ Should allow the owner to transfer ownership (132ms)


  Contract: ERC-20 Tests for LibraToken
    √ Should deploy a token with the proper configuration (64ms)
    √ Should allocate tokens per the minting function, and validate balances (531ms)
    √ Should transfer tokens from 0xd86543882b609b1791d39e77f0efc748dfff7dff to 0x42adbad92ed3e86db13e4f6380223f36df9980ef (164ms)
    √ Should not transfer negative token amounts (73ms)
    √ Should not transfer more tokens than you have (78ms)
    √ Should allow 0xa3883a50d7d537cec8f9bad8e8404aa8ff3078f3 to authorize 0x341106cb00828c87cd3ac0de55eda7255e04933f to transfer 1000 tokens (96ms)
    √ Should allow 0xa3883a50d7d537cec8f9bad8e8404aa8ff3078f3 to zero out the 0x341106cb00828c87cd3ac0de55eda7255e04933f authorization (96ms)
    √ Should allow 0x667632a620d245b062c0c83c9749c9bfadf84e3b to authorize 0x53353ef6da4bbb18d242b53a17f7a976265878d5 for 1000 token spend, and 0x53353ef6da4bbb18d242b53a17f7a976265878d5 should be able to send these tokens to 0x341106cb00828c87cd3ac0de55eda7255e04933f (368ms)
    √ Should not allow 0x53353ef6da4bbb18d242b53a17f7a976265878d5 to transfer negative tokens from 0x667632a620d245b062c0c83c9749c9bfadf84e3b (58ms)
```

√ Should not allow 0x53353ef6da4bbb18d242b53a17f7a976265878d5 to transfer tokens from 0x667632a620d245b062c0c83c9749c9bfadf84e3b to 0x0 (52ms)

√ Should not transfer tokens to 0x0 (63ms)

√ Should not allow 0x53353ef6da4bbb18d242b53a17f7a976265878d5 to transfer more tokens than authorized from 0x667632a620d245b062c0c83c9749c9bfadf84e3b (70ms)

√ Should allow an approval to be set, then increased, and decreased (514ms)

6. All Contract Files Tested

Commit Hash: 6a7194e879681a4d7e552dcee8e3e857626d8d59

File	Fingerprint (SHA256)
contracts/AirdropLibraToken.sol	`ed65b60b8ac35490fe96ff03db30278185440f4a5f3261667e244f07a79ebe95co`
contracts/LibraToken.sol	`5c8bf26b371e30c832021afe30d17b892d4bfa3350aef5fde2258f10cd3a1124`
contracts/zeppelin-solidty/contracts/math/SafeMath.sol	`030e12e10469d1f04c41b25f0b912efa2d036f317c83fe573c531306070cf223`
contracts/zeppelin-solidty/contracts/ownership.Ownable.sol	`ce2206349552db0cb886beb8f2cb9279bc74851a596aed4ce114e138e3be19fc`
contracts/zeppelin-solidty/contacts/token/ERC20/BasicToken.sol	`2ce519dcadb6455185e1478ddeb47520a7a00f6f311f69969f6ac00315f66206`
contracts/zeppelin-solidty/contacts/token/ERC20/ERC20.sol	`ed00fe45c0deef6a6f741c1dc57c4b5d4754404e02a3ea36da2fad8157cf4e1f`
contracts/contracts/zeppelin-solidty/contacts/token/ERC20/ERC20Basic.sol	`7d99160795719766f3dc95d53b24c87ac6a0235429992ad63eccd48be34241cb`
contracts/zeppelin-solidty/contacts/token/ERC20/StandardToken.sol	`3bd6286097107bf9c99f32eaf2e35f96fa49ad808c43da725b04caf89655d686`

7. Individual File Coverage Report

File	**% Statements**	**% Branches**	**% Functions**	**% Lines**
contracts/AirdropLibraToken.sol	100.00%	100.00%	100.00%	100.00%
contracts/LibraToken.sol	100.00%	100.00%	100.00%	100.00%
contracts/zeppelin-solidty/contracts/math/SafeMath.sol	100.00%	100.00%	100.00%	100.00%
contracts/zeppelin-solidty/contracts/ownership.Ownable.sol	100.00%	100.00%	100.00%	100.00%
contracts/zeppelin-solidty/contacts/token/ERC20/BasicToken.sol	100.00%	100.00%	100.00%	100.00%
contracts/zeppelin-solidty/contacts/token/ERC20/ERC20.sol	100.00%	100.00%	100.00%	100.00%
contracts/contracts/zeppelin-solidty/contacts/token/ERC20/ERC20Basic.sol	100.00%	100.00%	100.00%	100.00%
contracts/zeppelin-solidty/contacts/token/ERC20/StandardToken.sol	100.00%	100.00%	100.00%	100.00%
All files	**100.00%**	**100.00%**	**100.00%**	**100.00%**