CERTIK

Security Assessment

CheersLand

Oct 11th, 2021

Table of Contents

Summary

This report has been prepared for CheersLand to discover issues and vulnerabilities in the source code of the CheersLand project as well as any contract dependencies that were not part of an officially recognized library. A comprehensive examination has been performed, utilizing Static Analysis and Manual Review techniques.

The auditing process pays special attention to the following considerations:

- Testing the smart contracts against both common and uncommon attack vectors.
- Assessing the codebase to ensure compliance with current best practices and industry standards.
- Ensuring contract logic meets the specifications and intentions of the client.
- Cross referencing contract structure and implementation against similar smart contracts produced by industry leaders.
- Thorough line-by-line manual review of the entire codebase by industry experts.

Additionally, this audit is based on a premise that all external smart contracts are implemented safely. And the following `sol` file is not within the scope of the audit:

- `./../owner/Auth.sol`

We suggest recommendations that could better serve the project from the security perspective:

- Enhance general coding practices for better structures of source codes;
- Add enough unit tests to cover the possible use cases;
- Provide more comments per each function for readability, especially contracts that are verified in public;
- Provide more transparency on privileged activities once the protocol is live.

Overview

Project Summary

Project Name	CheersLand
Platform	Ethereum
Language	Solidity
Codebase	https://github.com/CheersLand/cheersland-igo-smart-contract
Commit	ec525d786518905c37bcf3142901a50d5344abc4

Audit Summary

Delivery Date	Oct 11, 2021
Audit Methodology	Static Analysis, Manual Review
Key Components	

Vulnerability Summary

Vulnerability Level	Total	Pending	Declined	Acknowledged	Partially Resolved	Resolved
Critical	0	0	0	0	0	0
Major	2	0	0	1	0	1
Medium	1	0	0	0	0	1
Minor	2	0	0	0	0	2
Informational	1	0	0	0	0	1
Discussion	0	0	0	0	0	0

Audit Scope

ID	File	SHA256 Checksum
FIP	FundraisingIdoPool.sol	8cf9607fa80dfafc8de2d2ae2b33220b91c75bec244b13b2ebf0b2e9a4cff4cb

Understandings

Overview

`CheersLand` is a project for users to purchase `lpAddress` token.

Users use `fundRaisingAddress` token to buy the `lpAddress`.

There is a fundraising goal that is set by the team. When the fundraising goal is reached, the team will only receive the designated amount mandated for the fundraising goal. The remaining tokens will be returned to users.

There is a `startTime` and `endTime` which represents the start time and end time of the purchase activity. There is a `claimTime`. After claim time, users can get `lpAddress` they purchased, and reclaim the excess proceeds from the remaining `fundRaisingAddress`. The team can only claim the designated amount described per the `fundRaisingAddress` not greater than the fundraising goal and extract the surplus `lpAddress`.

There is a white list strategy. Variable `rand` is used to mark users at a whitelist level. `rank` has four values: 0, 1, 2, 3, respectively means: 0: never be a whitelist, 1: normal whitelist, 2: super whitelist, 3: ever a whitelist, and later become not a whitelist. Only users in the whitelist can purchase `lpAddress`. Users in the normal whitelist have a default purchase limit `whiteListQuota`. Users in the super whitelist can have a purchase limit that is set by `operator`.

If the user's mortgage num is greater than the `threshold` in a third pool `poolAddress`, he will be set as a normal white list when he purchases `lpAddress`.

Additionally, the contract `FundraisingIdoPool` can work correctly only when there is enough `lpAddress` in it. According to the codebase, the amount of `lpAddress` should be `lpQuantitySold` when the activity starts.

Privileged Functions

The project contains the following privileged functions. They are used to modify the contract configurations and address attributes. We grouped these functions below:

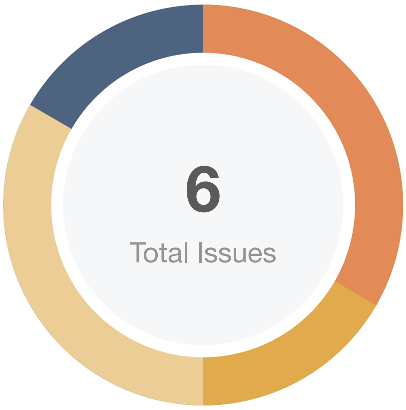
The `onlyOperator` modifier:

contract `FundraisingIdoPool.sol`:

- function setPoolAddress()

- function setAdminAddress()
- function setWhiteListQuota()
- function setExchangeRate()
- function setUpperLimit()
- function setEndTime()
- function setClaimTime()
- function addSuperWhiteList()
- function setWhiteList()
- function ownerClaim()
- function extractSurplusLp()

Findings



Critical	**0** (0.00%)
Major	**2** (33.33%)
Medium	**1** (16.67%)
Minor	**2** (33.33%)
Informational	**1** (16.67%)
Discussion	**0** (0.00%)

ID	Title	Category	Severity	Status
CheersLand-01	Centralization Risk	**Centralization / Privilege**	**Major**	Acknowledged
FIP-01	Missing Input Validation	Volatile Code	Minor	Resolved
FIP-02	Function Visibility Optimization	Gas Optimization	Informational	Resolved
FIP-03	Unchecked Value of ERC-20 `transfer()`/`transferFrom()` Call	Volatile Code	Minor	Resolved
FIP-04	Potential Reentrancy Attack	Logical Issue	Medium	Resolved
FIP-05	Users May Be Unable To Claim The Tokens They Purchased	Logical Issue	Major	Resolved

CheersLand-01 | Centralization Risk

Category	Severity	Location	Status
Centralization / Privilege	Major	Global	Acknowledged

Description

In the contract `FundraisingIdoPool`, the role `operator` has the authority over the following function:

1. Update `poolAddress` through `setPoolAddress`.
2. Update `_adminAddress` through `setAdminAddress` function.
3. Update `whiteListQuota` through `setWhiteListQuota` function.
4. Update `exchangeRate` through `setExchangeRate` function.
5. Update `upperLimit[_account]` through `setUpperLimit` function.
6. Update `endTime` through `setEndTime` function.
7. Update `claimTime` through `setClaimTime` function.
8. Set super white list through `addSuperWhiteList` function.
9. Set white list through `setWhiteList` function.
10. Claim `fundRaisingAddress` through `ownerClaim` function.
11. Extract surplus `lpAddress` through `extractSurplusLp` function.

without obtaining the consensus of the community.

Recommendation

We advise the client to carefully manage the `owner`, `injector`, `operator` account's private key to avoid any potential risks of being hacked. In general, we strongly recommend centralized privileges or roles in the protocol to be improved via a decentralized mechanism or smart-contract-based accounts with enhanced security practices, e.g., Multisignature wallets.

Indicatively, here are some feasible suggestions that would also mitigate the potential risk at the different levels in terms of short-term and long-term:

- Time-lock with reasonable latency, e.g., 48 hours, for awareness on privileged operations;
- Assignment of privileged roles to multi-signature wallets to prevent a single point of failure due to the private key;
- Introduction of a DAO/governance/voting module to increase transparency and user involvement.

Alleviation

No Alleviation.

FIP-01 | Missing Input Validation

Category	Severity	Location	Status
Volatile Code	Minor	FundraisingIdoPool.sol: 92, 93, 99, 108, 112	Resolved

Description

The given input is missing the check for the non-zero address.

Recommendation

We advise adding the check for the passed-in values to prevent unexpected error as below:

Alleviation

The team heeded our advice and added a zero check. Code change was applied in commit : d333c208cc2f1c9f52244f66773eda6f9fd7d3ad.

FIP-02 | Function Visibility Optimization

Category	Severity	Location	Status
Gas Optimization	Informational	FundraisingIdoPool.sol: 136	Resolved

Description

The following functions are declared as `public`, contain array function arguments, and are not invoked in any of the contracts contained within the project's scope. The functions that are never called internally within the contract should have external visibility.

Recommendation

We advise that the functions' visibility specifiers are set to `external` and the array-based arguments change their data location from `memory` to `calldata`, optimizing the gas cost of the function.

Alleviation

The team heeded our advice and change `public` to `external`. Code change was applied in commit : d333c208cc2f1c9f52244f66773eda6f9fd7d3ad.

FIP-03 | Unchecked Value of ERC-20 `transfer()`/`transferFrom()` Call

Category	Severity	Location	Status
Volatile Code	Minor	FundraisingIdoPool.sol: 213, 240, 243, 263, 276	Resolved

Description

The linked `transfer()`/`transferFrom()` invocations do not check the return value of the function call which should yield a `true` result in case of proper ERC-20 implementation.

Recommendation

As many tokens do not follow the ERC-20 standard faithfully, they may not return a `bool` variable in this function's execution meaning that simply expecting it can cause incompatibility with these types of tokens. Instead, we advise that OpenZeppelin's `SafeERC20.sol` implementation is utilized for interacting with the `transfer()` and `transferFrom()` functions of ERC-20 tokens. The OZ implementation optionally checks for a return value rendering compatible with all ERC-20 token implementations.

Alleviation

The team heeded our advice and change to `safeTransfer` and `safeTransferFrom`. Code change was applied in commit : d333c208cc2f1c9f52244f66773eda6f9fd7d3ad.

FIP-04 | Potential Reentrancy Attack

Category	Severity	Location	Status
Logical Issue	Medium	FundraisingIdoPool.sol: 239~246, 262~266	Resolved

Description

A reentrancy attack can occur when the contract creates a function that makes an external call to another untrusted contract before resolving any effects. If the attacker can control the untrusted contract, they can make a recursive call back to the original function, repeating interactions that would have otherwise not run after the external call resolved the effects.

Recommendation

We recommend using the Checks-Effects-Interactions Pattern to avoid the risk of calling unknown contracts or applying OpenZeppelin ReentrancyGuard library - `nonReentrant` modifier for the aforementioned functions to prevent reentrancy attack.

Alleviation

The team heeded our advice and added an inspector lock. Code change was applied in commit : d333c208cc2f1c9f52244f66773eda6f9fd7d3ad.

FIP-05 | Users May Be Unable To Claim The Tokens They Purchased

Category	Severity	Location	Status
Logical Issue	Major	FundraisingIdoPool.sol: 271~279	Resolved

Description

`operator` can transfer all the `lpAddress` in contract `FundraisingIdoPool` to his own account after `claimTime` through function `extractSurplusLp()`. This operation may be earlier than the users to call `calim()`. This will cause the users unable to get the `lpAddress` they purchased.

Recommendation

The team should ensure users can get all the `lpAddress` they purchased.

Alleviation

The team heeded our advice and removed the `extractSurplusLp` function. Added a logic in function `ownerClaim` to ensure there will be enough `lpAddress` in the contract for users to claim. Code change was applied in commit : d333c208cc2f1c9f52244f66773eda6f9fd7d3ad.

Appendix

Finding Categories

Centralization / Privilege

Centralization / Privilege findings refer to either feature logic or implementation of components that act against the nature of decentralization, such as explicit ownership or specialized access roles in combination with a mechanism to relocate funds.

Gas Optimization

Gas Optimization findings do not affect the functionality of the code but generate different, more optimal EVM opcodes resulting in a reduction on the total gas cost of a transaction.

Logical Issue

Logical Issue findings detail a fault in the logic of the linked code, such as an incorrect notion on how block.timestamp works.

Volatile Code

Volatile Code findings refer to segments of code that behave unexpectedly on certain edge cases that may result in a vulnerability.

Checksum Calculation Method

The "Checksum" field in the "Audit Scope" section is calculated as the SHA-256 (Secure Hash Algorithm 2 with digest size of 256 bits) digest of the content of each file hosted in the listed source repository under the specified commit.

The result is hexadecimal encoded and is the same as the output of the Linux "sha256sum" command against the target file.

Disclaimer

This report is subject to the terms and conditions (including without limitation, description of services, confidentiality, disclaimer and limitation of liability) set forth in the Services Agreement, or the scope of services, and terms and conditions provided to you ("Customer" or the "Company") in connection with the Agreement. This report provided in connection with the Services set forth in the Agreement shall be used by the Company only to the extent permitted under the terms and conditions set forth in the Agreement. This report may not be transmitted, disclosed, referred to or relied upon by any person for any purposes, nor may copies be delivered to any other person other than the Company, without CertiK's prior written consent in each instance.

This report is not, nor should be considered, an "endorsement" or "disapproval" of any particular project or team. This report is not, nor should be considered, an indication of the economics or value of any "product" or "asset" created by any team or project that contracts CertiK to perform a security assessment. This report does not provide any warranty or guarantee regarding the absolute bug-free nature of the technology analyzed, nor do they provide any indication of the technologies proprietors, business, business model or legal compliance.

This report should not be used in any way to make decisions around investment or involvement with any particular project. This report in no way provides investment advice, nor should be leveraged as investment advice of any sort. This report represents an extensive assessing process intending to help our customers increase the quality of their code while reducing the high level of risk presented by cryptographic tokens and blockchain technology.

Blockchain technology and cryptographic assets present a high level of ongoing risk. CertiK's position is that each company and individual are responsible for their own due diligence and continuous security. CertiK's goal is to help reduce the attack vectors and the high level of variance associated with utilizing new and consistently changing technologies, and in no way claims any guarantee of security or functionality of the technology we agree to analyze.

The assessment services provided by CertiK is subject to dependencies and under continuing development. You agree that your access and/or use, including but not limited to any services, reports, and materials, will be at your sole risk on an as-is, where-is, and as-available basis. Cryptographic tokens are emergent technologies and carry with them high levels of technical risk and uncertainty. The assessment reports could include false positives, false negatives, and other unpredictable results. The services may access, and depend upon, multiple layers of third-parties.

ALL SERVICES, THE LABELS, THE ASSESSMENT REPORT, WORK PRODUCT, OR OTHER MATERIALS, OR ANY PRODUCTS OR RESULTS OF THE USE THEREOF ARE PROVIDED "AS IS" AND "AS

AVAILABLE" AND WITH ALL FAULTS AND DEFECTS WITHOUT WARRANTY OF ANY KIND. TO THE MAXIMUM EXTENT PERMITTED UNDER APPLICABLE LAW, CERTIK HEREBY DISCLAIMS ALL WARRANTIES, WHETHER EXPRESS, IMPLIED, STATUTORY, OR OTHERWISE WITH RESPECT TO THE SERVICES, ASSESSMENT REPORT, OR OTHER MATERIALS. WITHOUT LIMITING THE FOREGOING, CERTIK SPECIFICALLY DISCLAIMS ALL IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT, AND ALL WARRANTIES ARISING FROM COURSE OF DEALING, USAGE, OR TRADE PRACTICE. WITHOUT LIMITING THE FOREGOING, CERTIK MAKES NO WARRANTY OF ANY KIND THAT THE SERVICES, THE LABELS, THE ASSESSMENT REPORT, WORK PRODUCT, OR OTHER MATERIALS, OR ANY PRODUCTS OR RESULTS OF THE USE THEREOF, WILL MEET CUSTOMER'S OR ANY OTHER PERSON'S REQUIREMENTS, ACHIEVE ANY INTENDED RESULT, BE COMPATIBLE OR WORK WITH ANY SOFTWARE, SYSTEM, OR OTHER SERVICES, OR BE SECURE, ACCURATE, COMPLETE, FREE OF HARMFUL CODE, OR ERROR-FREE. WITHOUT LIMITATION TO THE FOREGOING, CERTIK PROVIDES NO WARRANTY OR UNDERTAKING, AND MAKES NO REPRESENTATION OF ANY KIND THAT THE SERVICE WILL MEET CUSTOMER'S REQUIREMENTS, ACHIEVE ANY INTENDED RESULTS, BE COMPATIBLE OR WORK WITH ANY OTHER SOFTWARE, APPLICATIONS, SYSTEMS OR SERVICES, OPERATE WITHOUT INTERRUPTION, MEET ANY PERFORMANCE OR RELIABILITY STANDARDS OR BE ERROR FREE OR THAT ANY ERRORS OR DEFECTS CAN OR WILL BE CORRECTED.

WITHOUT LIMITING THE FOREGOING, NEITHER CERTIK NOR ANY OF CERTIK'S AGENTS MAKES ANY REPRESENTATION OR WARRANTY OF ANY KIND, EXPRESS OR IMPLIED AS TO THE ACCURACY, RELIABILITY, OR CURRENCY OF ANY INFORMATION OR CONTENT PROVIDED THROUGH THE SERVICE. CERTIK WILL ASSUME NO LIABILITY OR RESPONSIBILITY FOR (I) ANY ERRORS, MISTAKES, OR INACCURACIES OF CONTENT AND MATERIALS OR FOR ANY LOSS OR DAMAGE OF ANY KIND INCURRED AS A RESULT OF THE USE OF ANY CONTENT, OR (II) ANY PERSONAL INJURY OR PROPERTY DAMAGE, OF ANY NATURE WHATSOEVER, RESULTING FROM CUSTOMER'S ACCESS TO OR USE OF THE SERVICES, ASSESSMENT REPORT, OR OTHER MATERIALS.

ALL THIRD-PARTY MATERIALS ARE PROVIDED "AS IS" AND ANY REPRESENTATION OR WARRANTY OF OR CONCERNING ANY THIRD-PARTY MATERIALS IS STRICTLY BETWEEN CUSTOMER AND THE THIRD-PARTY OWNER OR DISTRIBUTOR OF THE THIRD-PARTY MATERIALS.

THE SERVICES, ASSESSMENT REPORT, AND ANY OTHER MATERIALS HEREUNDER ARE SOLELY PROVIDED TO CUSTOMER AND MAY NOT BE RELIED ON BY ANY OTHER PERSON OR FOR ANY PURPOSE NOT SPECIFICALLY IDENTIFIED IN THIS AGREEMENT, NOR MAY COPIES BE DELIVERED TO, ANY OTHER PERSON WITHOUT CERTIK'S PRIOR WRITTEN CONSENT IN EACH INSTANCE.

NO THIRD PARTY OR ANYONE ACTING ON BEHALF OF ANY THEREOF, SHALL BE A THIRD PARTY OR OTHER BENEFICIARY OF SUCH SERVICES, ASSESSMENT REPORT, AND ANY ACCOMPANYING

MATERIALS AND NO SUCH THIRD PARTY SHALL HAVE ANY RIGHTS OF CONTRIBUTION AGAINST CERTIK WITH RESPECT TO SUCH SERVICES, ASSESSMENT REPORT, AND ANY ACCOMPANYING MATERIALS.

THE REPRESENTATIONS AND WARRANTIES OF CERTIK CONTAINED IN THIS AGREEMENT ARE SOLELY FOR THE BENEFIT OF CUSTOMER. ACCORDINGLY, NO THIRD PARTY OR ANYONE ACTING ON BEHALF OF ANY THEREOF, SHALL BE A THIRD PARTY OR OTHER BENEFICIARY OF SUCH REPRESENTATIONS AND WARRANTIES AND NO SUCH THIRD PARTY SHALL HAVE ANY RIGHTS OF CONTRIBUTION AGAINST CERTIK WITH RESPECT TO SUCH REPRESENTATIONS OR WARRANTIES OR ANY MATTER SUBJECT TO OR RESULTING IN INDEMNIFICATION UNDER THIS AGREEMENT OR OTHERWISE.

FOR AVOIDANCE OF DOUBT, THE SERVICES, INCLUDING ANY ASSOCIATED ASSESSMENT REPORTS OR MATERIALS, SHALL NOT BE CONSIDERED OR RELIED UPON AS ANY FORM OF FINANCIAL, TAX, LEGAL, REGULATORY, OR OTHER ADVICE.

About

Founded in 2017 by leading academics in the field of Computer Science from both Yale and Columbia University, CertiK is a leading blockchain security company that serves to verify the security and correctness of smart contracts and blockchain-based protocols. Through the utilization of our world-class technical expertise, alongside our proprietary, innovative tech, we're able to support the success of our clients with best-in-class security, all whilst realizing our overarching vision; provable trust for all throughout all facets of blockchain.