LendHub Audit Report

Version 1.0.0

Presented by Fairyproof

Feburary 4, 2021

01. Introduction

This document includes the results of the audit performed by the Fairyproof team on the lendhub project, at the request of the lendhub team.

The audited code can be found in the public lendhub Github repository, and the version used for this report is commit

5af1362d241f9ac5205d13e0676ffb8ebc79220d

The goal of this audit is to review lendhub's solidity implementation for its decentralized lending application, study potential security vulnerabilities, its general design and architecture, and uncover bugs that could compromise the software in production.

We make observations on specific areas of the code that present concrete problems, as well as general observations that traverse the entire codebase horizontally, which could improve its quality as a whole.

— Disclaimer

Note that as of the date of publishing, the contents of this document reflect the current understanding of known security patterns and state of the art regarding smart contract security.

And the solidity implementation was audited based on Ethereum's running environment. Whether or not this implementation can run on other blockchains or would encounter issues running on other blockchains is not covered by this audit.

Risks or issues introduced by this implementation interacting with contracts from other projects are not covered by this audit.

Contracts such as `mdexPair.sol` and `ImdexPair.sol` which are imported from third party projects are not audited. And any issues or risks introduced by these contracts are not covered by this audit either.

Given the size of the project, the findings detailed here are not to be considered exhaustive, and further testing and audit is recommended after the issues covered are fixed.

— Methodology

Lendhub's codebase was studied in detail in order to acquire a clear impression of how the its specifications were implemented. The codebase was then subject to deep analysis and scrutiny, resulting in a series of observations. The problems and their potential solutions are discussed in this document and, whenever possible, we identify common sources for such problems and comment on them as well.

— Structure of the document

This report contains a list of issues and comments on all the contracts under the directory https://github.com/lendhub/lendhub/tree/master/contracts and its sub-directories. Each issue is assigned a severity level based on the potential impact of the issue and recommendations to fix it, if applicable. For ease of navigation, an index by topic and another by severity are both provided at the beginning of the report.

— Documentation

For this audit, we used the following sources of truth about how the lendhub system should work:

https://www.lendhub.org/

https://www.yuque.com/lendhub/kb/gzvbof

https://www.yuque.com/lendhub/kb/ufuguf

These were considered the specification, and when discrepancies arose with the actual code behavior, we consulted with the lendhub team or reported an issue.

— Comments from Auditee

No vulnerabilities with critical or high severities were found in the lendhub's codebase. All the vulnerabilities with medium and low severities were acknowledged by the team, and the team intends to try every means to avoid triggering these vulnerabilities and will fix them in future upgrades.

The lendhub's codebase passed the audit performed by the Fairyproof team.

02. About Fairyproof

Fairyproof is a leading technology firm in the blockchain industry, providing consulting and security audits for organizations. Fairyproof has developed industry security standards for designing and deploying smart contract systems.

03. Severity level reference

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

Critical severity issues need to be fixed as soon as possible.

High severity issues will probably bring problems and should be fixed.

Medium severity issues could potentially bring problems and should eventually be fixed.

Low severity issues are minor details and warnings that can remain unfixed but would be better fixed at some point in the future.

04. List of issues by severity

A. Critical

- N/A

B. High

- N/A

C. Medium

- Chef.sol

Missing Address Check

D. Low

- Comptroller.sol

State Variable Not Reset

- Chef.sol

Incorrect Algorithm

05. List of issues by contract file

- Comptroller.sol

State Variable Not Reset: Low

- Chef.sol

Missing Address Check: Medium

Incorrect Algorithm: Low

06. Issue descriptions and recommendations by contract file

- Comptroller.sol

State Variable Not Reset: Low

Source:

Line 1342：in the function `_dropCompMarket` after a market is dropped its state is not reset. In case this market is re-added afterwards it will take its obsolete state. And this may cause confustion and contaminate data.

Recommendation:

Consider resetting a market's state variable after the market is dropped or reinitializing a market's state variable when the previously dropped market is re-added.

Update: Acknowledged by the lendhub team. The team thinks this will not cause data contamination or security issues, and prefers to keep it for now and may make a change later.

- Chef.sol

Missing Address Check: Medium

Source:

Line 85：in the function `add` if `_lpToken` is set to "0" it will cause the subsequent call of the function `massUpdatePools` to fail.

Recommendation:

Consider adding the following statement before the conditional check `if (_withUpdate) {`:

`require(address(_lpToken) != address(0), "zero address");`。

Update: Acknowledged by the lendhub team. However since the contract has been deployed it is infeasible to change the code, the team will keep this in mind and make sure not to pass a "0" address to `_lpToken`.

Incorrect Algorithm: Low

Source:

Line 114: in the function `pendingLHB` when the remaining reward of a pool is close to 0, the reward which is calculated by the function's algorithm and allocated to a user doesn't match the actual reward a user should get.

Recommendation:

Consider making a public announcement notifying existing and potential users of this issue.

Update: Acknowledged by the lendhub team.