#HashEx
BLOCKCHAIN SECURITY

Teddy Cash

smart contracts final audit report

October 2021

hashex.org

contact@hashex.org

Contents

1. Disclaimer 3

2. Overview 4

3. Found issues 6

4. Contracts 7

5. Conclusion 8

Appendix A. Issues' severity classification 9

Appendix B 10

8. References 11

1. Disclaimer

This is a limited report on our findings based on our analysis, in accordance with good industry practice 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 disclaimer below – please make sure to read it in full.

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 HashEx and its affiliates (including holding companies, shareholders, subsidiaries, employees, directors, officers, and other representatives) (HashEx) owe no duty of care towards you or any other person, nor does HashEx 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 HashEx 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, HashEx hereby excludes all liability and responsibility, and neither you nor any other person shall have any claim against HashEx, 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 the 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. HashEx owns all copyright rights to the text, images, photographs, and other content provided in the following document. When using or sharing partly or in full, third parties must provide a direct link to the original document mentioning the author (hashex.org).

2. Overview

HashEx was commissioned by the Teddy Cash team to verify the identity of the forked Liquity code. Verification was conducted between September 29 and October 03, 2021.

The reviewed code is deployed to Avalanche C-Chain network: list of contract addresses.

- priceFeed: 0x1465528eA599Cc9FB5268CFa94C3FaB21029a8DE
- sortedTroves: 0x5272DfB4851723328dA7730BE944502E5C965f40
- troveManager: 0xd22b04395705144Fd12AfFD854248427A2776194
- activePool: 0x23AD4Cb653813c319b18a63300e54DF4a6fc9a86
- stabilityPool: 0x7AEd63385C03Dc8ed2133F705bbB63E8EA607522
- gasPool: 0x0D8D21B2da99320aF54b72Ac0d3FCe90921Cc665
- defaultPool: 0x43F9CD378DaEA426844B13E553EB529551576C7a
- collSurplusPool: 0xBC6C16283c1260CE5CF72C951b4D399E81FBcA36
- borrowerOperations: 0xF582CAE047853cbe7F0Bc8f8321bEF4a1eBE0307
- hintHelpers: 0xE90A069D197d7aF40bD9Aef20c907F2E4dD7d4Fc
- TSD Token: 0x4fbf0429599460D327BD5F55625E30E4fC066095
- uniToken: 0x67E395B6ACd948931eeE8F52C7c1Fe537E7f1a7a
- unipool: 0x9717Ff7406Be065EA177bA9ab1bE704060Af8370
- TEDDY Staking: 0xb4387D93B5A9392f64963cd44389e7D9D2E1053c
- lockupContractFactory: 0xdD6e1b601A2D86264d62D6ca85Ce2bC944981106

- communityIssuance: 0xb4Fbc7839ce88029c8c1c6274660118e27B6f982
- TEDDY Token: 0x094bd7B2D99711A1486FB94d4395801C6d0fdDcC
- multiTroveGetter: 0x6cf187ADa698F3bD01A4931D6ce4cD053Fce294e

The purpose of this audit was to achieve the following:

- ensure the identity of Teddy Cash contracts and Liquity.org as a source of the fork.

Information in this report should be used to understand the risk exposure of smart contracts, and as a guide to improving the security posture of smart contracts by remediating the issues that were identified.

2.1 Summary

Project name	Teddy Cash
URL	https://teddy.cash
Platform	Avalanche Network
Language	Solidity

2.2 Contracts

Name	Address
TroveManager	0xd22b04395705144Fd12AfFD854248427A2776194

3. Found issues

Medium	1 (100%)

C46. TroveManager

ID	Severity	Title	Status
C46Ia7	Medium	Differences from Liquity	Acknowledged

4. Contracts

C46. TroveManager

Overview

Troves controller contract.

Issues

C46Ia7 Differences from Liquity

Medium Acknowledged

The `collToOffset` variable and all the logic behind it (L3323, 3589, 3657, 3756, 3895) is absent in the deployed TroveManager from the Liquity project [1]. The same code was in the Liquity GitHub codebase from May'21 to Sep'21. The feature was introduced in b9dcdbd and removed in the 47fb4d5 commit. See Liquity's team's explanation for reverting the changes here. Estimation of an impact of the issue was out of the scope of the current audit.

Team response

We have analysed the live deployment and come to the conclusion that it is a non-issue. The net effect of the change is that the last (partial) liquidation during recovery mode will cause the next recovery mode to be triggered fractions of a percent earlier (150.x% rather than 150%).

5. Conclusion

The audited project is a fork of Liquity.org contracts with minor changes. Besides the changes in token contracts, i.e. name, symbol, and tokenomics parameter changes, we have found differences in the TroveManager contract. The TroveManager contract matches the Liquity's Github repo at commit b9dcdbd, but differs from the Liquity's deployed version (0xA39739EF8b0231DbFA0DcdA07d7e29faAbCf4bb2).

Teddy Cash team has responded to this audit and concluded that these changes only affect a very rare edge case, and if that edge case happens it can make the recovery mode slightly more conservative.

Appendix A. Issues' severity classification

- **Critical.** Issues that may cause an unlimited loss of funds or entirely break the contract workflow. Malicious code (including malicious modification of libraries) is also treated as a critical severity issue. These issues must be fixed before deployments or fixed in already running projects as soon as possible.
- **High.** Issues that may lead to a limited loss of funds, break interaction with users, or other contracts under specific conditions. Also, issues in a smart contract, that allow a privileged account the ability to steal or block other users' funds.
- **Medium.** Issues that do not lead to a loss of funds directly, but break the contract logic. May lead to failures in contracts operation.
- **Low.** Issues that are of a non-optimal code character, for instance, gas optimization tips, unused variables, errors in messages.
- **Informational.** Issues that do not impact the contract operation. Usually, informational severity issues are related to code best practices, e.g. style guide.

Appendix B

- Business logic overview
- Functionality checks
- Following best practices
- Access control and authorization
- Reentrancy attacks
- Front-run attacks
- DoS with (unexpected) revert
- DoS with block gas limit
- Transaction-ordering dependence
- ERC/BEP and other standards violation
- Unchecked math
- Implicit visibility levels
- Excessive gas usage
- Timestamp dependence
- Forcibly sending ether to a contract
- Weak sources of randomness
- Shadowing state variables
- Usage of deprecated code

8. References

1. Liquity's TroveManager contract on Ethereum

 contact@hashex.org

 @hashex_manager

 blog.hashex.org

 linkedin

 github

 twitter

#HashEx
BLOCKCHAIN SECURITY