Technology of Nepal

Dexit.Finance

Security Audit Report

Disclaimer

This is a limited report on our findings based on our analysis, in accordance with good industry practice as at the date of this report, in relation to cybersecurity vulnerabilities and issues in the smart contract source code analysed, the details of which are set out in this report, (Source Code); and the Source Code compiling, deploying and performing the intended functions. In order to get a full view of our findings and the scope 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.

DISCLAIMER: 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 Technology of Nepal and its affiliates (including holding companies, shareholders, subsidiaries, employees, directors, officers and other representatives) owe no duty of care towards you or any other person, nor does Technology of Nepal 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 Technology of Nepal 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, Technology of Nepal hereby excludes all liability and responsibility, and neither you nor any other person shall have any claim against Technology of Nepal, 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 use of this report, and any reliance on this report.

Introduction

Technology of Nepal was commissioned by Dexit Finance to conduct a smart contract audit. The audit was performed between 30 July 2021 and 02 Aug 2021, consuming a total of 4 resource days.

Summary

This report has been prepared for Dexit Finance smart contracts, to discover issues and vulnerabilities in the source code of their Smart Contract as well as any contract dependencies that were not part of an officially recognized library. A comprehensive examination has been performed, utilizing DynamicAnalysis, 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.

The security assessment resulted in findings that ranged from critical to informational. We recommend addressing these findings to ensure a high level of security standards and industry practices. We suggest recommendations that could better serve the project from the security perspective:

Overview

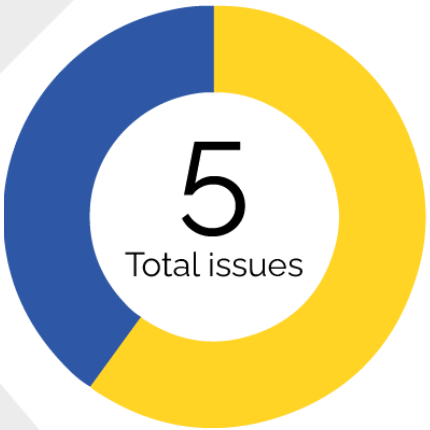
Project Summary

Project Name	Dexit Finance
Website	https://dexit.finance/
Description	Dexit is static Reflection Auto-Generating Liquidity Protocol
Contract Address	0x2b2ff80c489dad868318a19fd6f258889a026da5
Platform	Binance Smart Chain
Language	Solidity
Codebase	https://github.com/Dexit-Finance/DexitFinance/blob/main/DexitFinance.sol
Commits	9b7a8642966b2f8c562a0f49583da164cd4ba7a5

Audit Summary

Delivery Date	Date 2. August 2021
Audit Methodology	Static Analysis, Manual review
Key Component	DexitFinance.sol

Vulnerability Summary



Critical	0	(0.00%)
Major	0	(0.00%)
Medium	0	(0.00%)
Minor	3	(60.00%)
Informal	2	(40.00%)
Discussion	0	(0.00%)

This report is organized into the following sections.

1. Executive Summary
2. Audit Details
3. Contract Functions
4. Details Finding

The information in this report should be used to better understand the risk exposure of the smart contracts, and as a guide to improving the security posture of the smart contracts by remediating issues identified. The results of this audit are only a reflection of the source code reviewed at the time of the audit and of the source code that was determined to be in-scope.

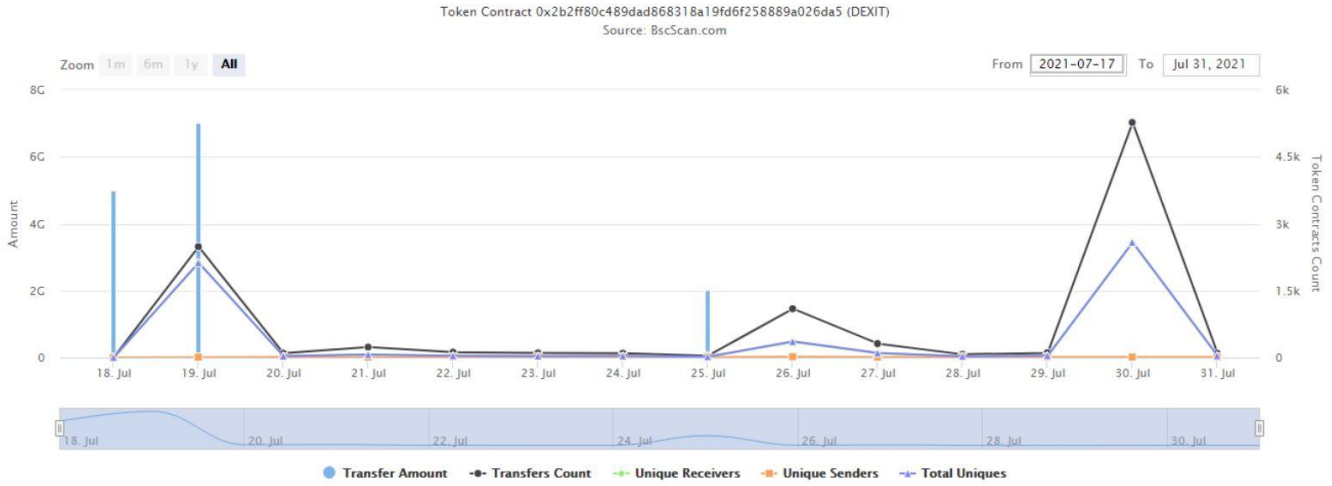
The purpose of this audit was to achieve the following:

- Identify potential security flaws
- Ensure that the smart contracts functioned according to the documentation provided

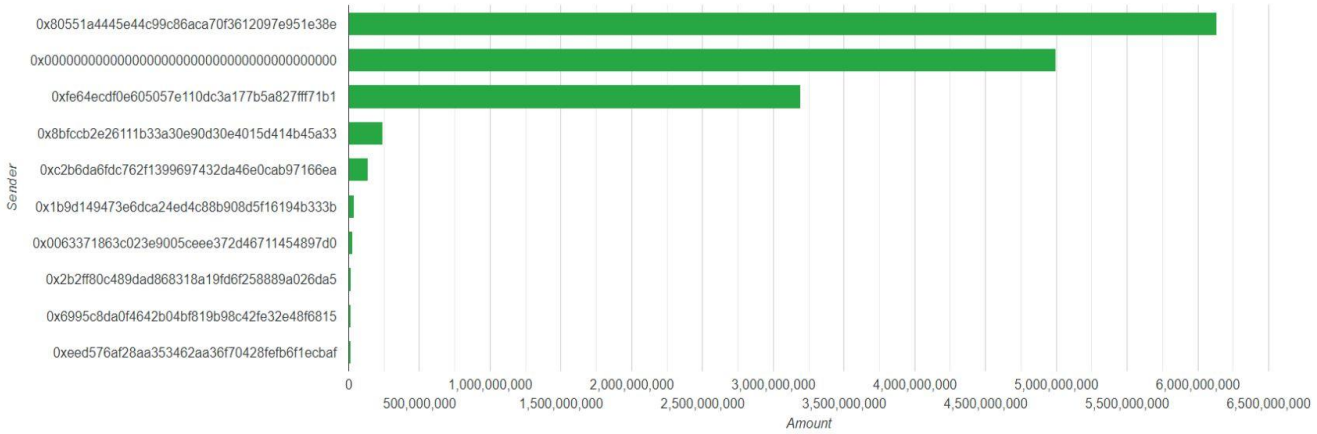
Assessing the off-chain functionality associated with the contracts, for example, backend web application code, was out of scope of this audit. Due to the unregulated nature and ease of transfer of cryptocurrencies, operations that store or interact with these assets are considered very high risk with regards to cyber-attacks. As such, the highest level of security should be observed when interacting with these assets. This requires a forward-thinking approach, which takes into account the new and experimental nature of blockchain technologies. Strategies that should be used to encourage secure code development include:

- Security should be integrated into the development lifecycle and the level of perceived security should not be limited to a single code audit.
- Defensive programming should be employed to account for unforeseen circumstances.
- Current best practices should be followed where possible

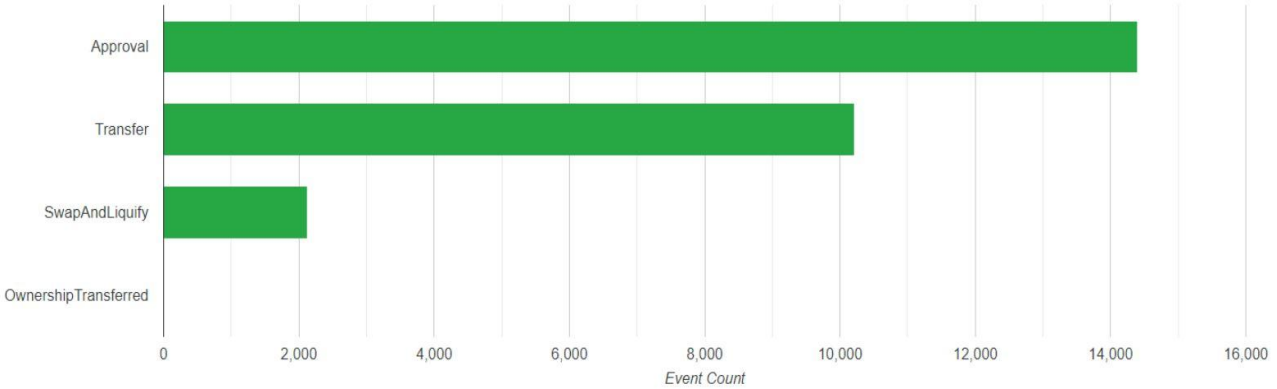
Dexit Finance Contract Interaction Details



Dexit Finance Distribution Count by Event



Dexit Finance Distribution Count By Volume



S.N	Issue description	Status
1	Compiler warnings.	✔
2	Race conditions and Reentrancy. Cross-function race conditions.	✔
3	Possible delays in data delivery.	✔
4	Oracle calls.	✔
5	Front running.	✔
6	Timestamp dependence.	✔
7	Integer Overflow and Underflow.	✔
8	DoS with Revert	✔
9	DoS with block gas limit.	✔
10	Methods execution permissions.	✔
11	Economy model. If application logic is based on an incorrect economic model, the application would not function correctly and participants would incur financial losses. This type of issue is most often found in bonus rewards systems, Staking and Farming contracts, Vault and Vesting contracts, etc.	✔
12	The impact of the exchange rate on the logic.	✔
13	Private user data leaks.	✔
14	Malicious Event log.	✔
15	Scoping and Declarations	✔
16	Uninitialized storage pointers.	✔
17	Arithmetic accuracy.	✔
18	Design Logic.	✔
19	Cross-function race conditions	✔
20	Safe Zeppelin module.	✔
21	Fallback function security.	✔

Executive Summary

This report presents the findings of an audit performed by Technology of Nepal on Dexit Finance.

Security issues

High Severity issue

- No high severity issues found

Medium Severity issues

- No medium severity issues found

Low Severity Issues

- Out of gas Issue:

  The function includeAccount() uses the loop to find and remove addresses from the _excluded list. Function will be aborted with OUT_OF_GAS exception if there will be a long excluded addresses list.

  The function _getCurrentSupply also uses the loop for evaluating total supply. It also could be aborted with OUT_OF_GAS exception if there will be a long-excluded addresses list.

Recommendations

Use EnumerableSet instead of array or do not use long arrays

Privileged Functions

The contract contains the following privileged functions that are restricted by the onlyOwner modifier. They are used to modify the contract configurations and address attributes. We grouped these functions below:

Account management functions for inclusion and exclusion in the fee and reward system:

excludeFromReward(address account)

includeInReward(address account)

includeInFee(address account)

excludeFromFee(address account)

Modification of liquidation, tax and max transaction percent of the system

function setTaxFeePercent(uint taxFee)

function setLiquidityFeePercent(uint liquidityFee)

function setMaxTxPercent(uint maxTxPercent)

function setSwapAndLiquifyEnabled(bool _enabled)

Here is a list of the transactions associated with the locked LPs:

https://dxsale.app/app/v2_9/dxlockview?id=1&add=0x80551A4445E44c99c86AcA70F3612097E 951e38E&type=lplock&chain=BSC

https://dxsale.app/app/v2_9/dxlockview?id=0&add=0x80551A4445E44c99c86AcA70F3612097E 951e38E&type=lplock&chain=BSC

Conclusion

Smart contracts do not contain any high severity issues!

Technology of Nepal note:
Please check the disclaimer above and note, the audit makes no statements or warranties on business model, investment attractiveness or code sustainability. The report is provided for the only contract mentioned in the report and does not include any other potential contracts deployed by Owner.