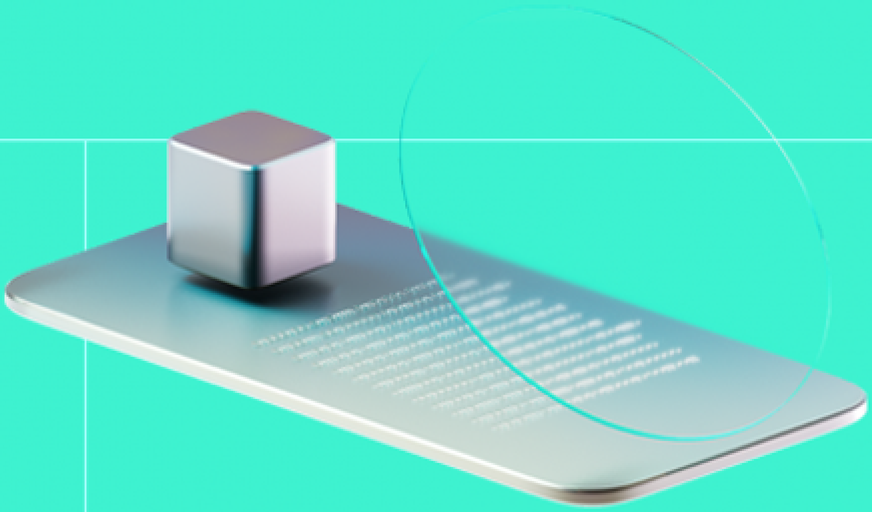
HACKEN

Smart Contract Code Review And Security Analysis Report



Customer: DeXe Network

Date: 24/09/2020

We express our gratitude to the DeXe Network team for the collaborative engagement that enabled the execution of this Smart Contract Security Assessment.

DeXe Network is a platform for Defi asset management.

Document

Name	Smart Contract Code Review and Security Analysis Report for DeXe Network
Audited By	Hacken
Changelog	20/09/2020 - Initial Audit
	24/09/2020 - Second Review
Platform	EVM
Language	Solidity
Tags	Token, Token-sales, Staking
Methodology	https://hackenio.cc/sc_methodology

Review

Scope

Audit Archive File	dexe-8ca55a54680edb118108384318f1867caf65565b.zip
SHA256 Checksum	c3b6df51f4b88bc7f518c425786a518548d89052153a65310058f7f74057ff18

Audit Summary

The system users should acknowledge all the risks summed up in the risks section of the report

2	2	0	0
Total Findings	Resolved	Accepted	Mitigated

Findings by Severity

Severity	Count
Critical	0
High	0
Medium	1
Low	1

Vulnerability	Status
F-2020-1920 - Improper Limit	Fixed
F-2020-1921 - Unused Field	Fixed

Table of Contents

Conclusion

Smart contracts within the scope was manually reviewed and analyzed with static analysis tools. For the contract high level description of functionality was presented in As-is overview section of the report.

Audit report contains all found security vulnerabilities and other issues in the reviewed code.

Security engineers found 1 medium and 1 low severity issue during the initial audit. **All the issues have been fixed before the secondary audit**.

The code is well-tested and works as described in the whitepaper.

Findings

Vulnerability Details

F-2020-1920 - Improper Limit - Medium

Description: DISTRIBUTOR_LIMIT field is set to 10 billion tokens. That is more than a total supply.

Status: Fixed

Classification

Severity: Medium

Recommendations

F-2020-1921 - Unused Field - Low

Description: `priceFeed` field of the Dexe contract is never used. It's recommended to remove unused fields and variables.

Status: Fixed

Classification

Severity: Low

Recommendations

Disclaimers

Hacken Disclaimer

The smart contracts given for audit have been analyzed based on best industry practices at the time of the writing of this report, with cybersecurity vulnerabilities and issues in smart contract source code, the details of which are disclosed in this report (Source Code); the Source Code compilation, deployment, and functionality (performing the intended functions).

The report contains no statements or warranties on the identification of all vulnerabilities and security of the code. The report covers the code submitted and reviewed, so it may not be relevant after any modifications. Do not consider this report as a final and sufficient assessment regarding the utility and safety of the code, bug-free status, or any other contract statements.

While we have done our best in conducting the analysis and producing this report, it is important to note that you should not rely on this report only — we recommend proceeding with several independent audits and a public bug bounty program to ensure the security of smart contracts.

English is the original language of the report. The Consultant is not responsible for the correctness of the translated versions.

Technical Disclaimer

Smart contracts are deployed and executed on a blockchain platform. The platform, its programming language, and other software related to the smart contract can have vulnerabilities that can lead to hacks. Thus, the Consultant cannot guarantee the explicit security of the audited smart contracts.

Appendix 1. Severity Definitions

When auditing smart contracts, Hacken is using a risk-based approach that considers **Likelihood**, **Impact**, **Exploitability** and **Complexity** metrics to evaluate findings and score severities.

Reference on how risk scoring is done is available through the repository in our Github organization:

hknio/severity-formula

Severity	Description
Critical	Critical vulnerabilities are usually straightforward to exploit and can lead to the loss of user funds or contract state manipulation.
High	High vulnerabilities are usually harder to exploit, requiring specific conditions, or have a more limited scope, but can still lead to the loss of user funds or contract state manipulation.
Medium	Medium vulnerabilities are usually limited to state manipulations and, in most cases, cannot lead to asset loss. Contradictions and requirements violations. Major deviations from best practices are also in this category.
Low	Major deviations from best practices or major Gas inefficiency. These issues will not have a significant impact on code execution, do not affect security score but can affect code quality score.

Appendix 2. Scope

The scope of the project includes the following smart contracts from the provided repository:

Scope Details	
Archive Name	dexe-8ca55a54680edb118108384318f1867caf65565b.zip
SHA256 Checksum	c3b6df51f4b88bc7f518c425786a518548d89052153a65310058f7f74057ff18
Whitepaper	Not provided
Requirements	Not provided
Technical Requirements	Not provided