Smart Contract Code Review And Security Analysis Report

Customer: Decubate

Date: 03/04/2024
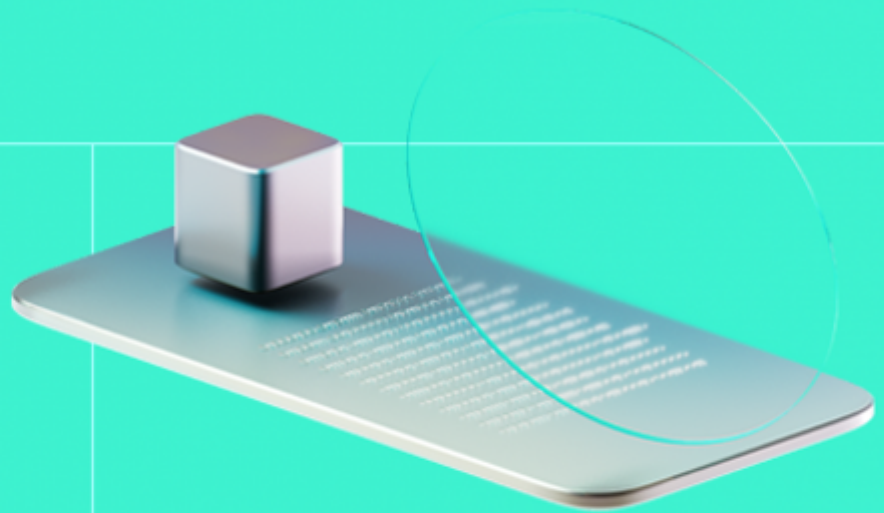
Playbux is a metaverse platform that allows users to represent crypto assets and customize characters. PBUX token serves as the main currency and method of exchange for both governance of the platform as well as acquisition of all tradable virtual goods, assets and services within the Playbux metaverse.

Platform: EVM

Language: Solidity

Tags: ERC-20

Timeline: 02/04/2024 - 03/04/2024

Methodology: https://hackenio.cc/sc_methodology

Review Scope

Repository	https://github.com/Decubate-com/smart-contracts/
Commit	45427c9851f7db1f1f773a50cdfe44a457adf874

Audit Summary

10/10
Security Score

10/10
Code quality score

0%
Test coverage

10/10
Documentation quality score

Total 10/10

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

0
Total Findings

0
Resolved

0
Accepted

0
Mitigated

Findings by severity

Critical	0
High	0
Medium	0
Low	0

Document

Name	Smart Contract Code Review and Security Analysis Report for Playbux
Audited By	Alkautsar Fatahillah
Approved By	Yves Toiser
Website	http://playbux.co/
Changelog	02/04/2024 - Preliminary Report

Table of Contents

System Overview

Playbux is an ERC-20 with the following contracts:

PBUX Token — simple ERC-20 token that mints all initial supply to a deployer. Additional minting is not allowed. A burnable feature is implemented.

It has the following attributes:

- Name: Playbux
- Symbol: PBUX
- Decimals: 18
- Total supply: 400m tokens.

Executive Summary

This report presents an in-depth analysis and scoring of the customer's smart contract project. Detailed scoring criteria can be referenced in the scoring methodology.

Documentation quality

The total Documentation Quality score is **10** out of **10**.

- Functional requirements are provided.
- Technical description is provided.
- NatSpecs are provided.

Code quality

The total Code Quality score is **10** out of **10**.

- The development environment is configured.
- There are no code quality issues.

Test coverage

Code coverage of the project is **0%** (branch coverage).

- Tests are not provided (according to our methodology, tests are not mandatory for projects with less than 250 lines of codes).

Security score

Upon auditing, the code was found to contain **0** critical, **0** high, **0** medium, and **0** low severity issues, leading to a security score of **10** out of **10**.

All identified issues are detailed in the "Findings" section of this report.

Summary

The comprehensive audit of the customer's smart contract yields an overall score of **10**. This score reflects the combined evaluation of documentation, code quality, test coverage, and security aspects of the project.

Risks

- All the tokens are minted to a single address. The secureness of the supply depends on the secureness of key storage.

Findings

Vulnerability Details

Observation Details

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

Repository	https://github.com/Decubate-com/smart-contracts/
Commit	45427c9851f7db1f1f773a50cdfe44a457adf874
Whitepaper	https://www.playbux.co/playbux-litepaper-announcement
Requirements	
Technical Requirements	

Contracts in Scope

contracts/PBUXToken.sol