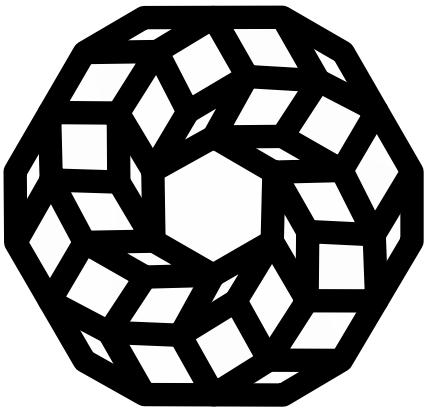

PowerIndex Pool Security Analysis

by Pessimistic

This report is private.

Published: December 14, 2020

Abstract

In this report, we consider the security of PowerIndexPool smart contracts of PowerPool project. Our task is to find and describe security issues in smart contracts of the platform.

Disclaimer

The audit does not give any warranties on the security of the code. One audit cannot be considered enough. We always recommend proceeding with several independent audits and a public bug bounty program to ensure the security of smart contracts. Besides, security audit is not an investment advice.

Summary

In this report, we considered the security of PowerIndexPool smart contracts of PowerPool project. We performed our audit according to the procedure described below.

The analysis showed two issues of low severity. We also noted that admin can change fees at any moment. However, the fees range is hardcoded.

The overall code quality is good.

The project has no documentation.

General recommendations

We recommend adding documentation to the project and fixing the issues.

Procedure

In our audit, we consider the following crucial features of the code:

1. Whether code logic corresponds to the specification.
2. Whether the code is secure.
3. Whether the code meets best practices.

We perform our audit according to the following procedure:

- Automated analysis
  - We scan project's code base with automated tools: MythX and SmartCheck.
  - We manually verify (reject or confirm) all the issues found by tools.
- Manual audit
  - We inspect the specification and check whether the logic of smart contracts is consistent with it.
  - We manually analyze code base for security vulnerabilities.
  - We assess overall project structure and quality.
- Report
  - We reflect all the gathered information in the report.

Project overview

Project description

In our analysis, we consider PowerIndexPool contracts of PowerPool project on GitHub repository, commit 9661812578c730ef5b85792002c205e595f57f07.

The scope of the audit included the analysis of following contracts:

- **contracts/balancer-core/**
- **contracts/powerIndexPool.sol**

The project has no documentation.

The total LOC of audited sources is 1404.

Manual analysis

The contracts were completely manually analyzed, their logic was checked. Besides, the results of the automated analysis were manually verified. All the confirmed issues are described below.

Critical issues

Critical issues seriously endanger smart contracts security. We highly recommend fixing them.

The audit showed no critical issues.

Medium severity issues

Medium issues can influence project operation in current implementation. We highly recommend addressing them.

The audit showed no issues of medium severity.

Low severity issues

Low severity issues can influence project operation in future versions of code. We recommend taking them into account.

Gas consumption

In **BPool** contract, `.length` property is used inside `for` loops at lines 483 and 530.

Consider saving it to a local variable to optimize gas consumption.

Code quality

In **BPool** contract, `callVoting()` function is payable for no reason.

Notes

Owner powers

The owner of **BPool** contract can change transaction fees at any moment with `setSwapFee()` and `setCommunityFeeAndReceiver()` functions. However, the fees limit is hardcoded and can only be changed within `0.000001–0.1` range.

This analysis was performed by Pessimistic:

Evgeny Marchenko, Senior Security Engineer
Boris Nikashin, Analyst
Alexander Seleznev, Founder

December 14, 2020