SmartDec

PumaPay Security Recheck

This report is public.

Published: July 1, 2020

Abstract

In this report, we consider the security of the PumaPay project. Our task is to find and describe security issues in the 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 PumaPay smart contracts. We performed our audit according to the procedure described below.

The initial audit showed a discrepancy with the documentation (medium-severity issue) and two issues of low severity.

In the latest version, the discrepancy was fixed as well as one of low-severity issues.

General recommendations

We recommend fixing the issue with tests.

Procedure

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

1. Whether the code is secure.
2. Whether the code corresponds to the documentation (including whitepaper).
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 SmartDec Scanner
  - we manually verify (reject or confirm) all the issues found by tools
- manual audit
  - 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 smart contracts of PumaPay project on Git repository, commit edd92c0f5367a5e1859975b88855293a9e98b8e3.

Latest version of the code

After the initial audit, developers updated the code to the latest version (Git repository, commit 342b28c26012e6ed689b3af20bced1077151e3b2).

Project architecture

For the audit, we were provided with a git repository. The project has tests and documentation.

The scope of the audit included only MultiVesting.sol file.

The total LOC of audited sources is 106.

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.

Discrepancy with the documentation

`addVesting()` function at line 56 has no check if `_startedAt` moment is in the future or now. In the documentation, it is declared:

```
It's also possible to start vesting from some date in the future by calling
address user, uint256 amount, uint256 startedAt
```

We recommend adding a proper check to `addVesting()` function.

The issue has been fixed and is not present in the latest version of the code.

Low severity issues

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

Code style

- At line 21, time interval is given in seconds: `SECONDS_IN_30_DAYS = 2592000`.
  We recommend using `30 days` instead to improve code readability.
- At lines 95 and 143, consider using `getAvailableAmount()`.
- Function `getAvailableAmountAtTimestamp()` at line 160 is often used with a particular `Vesting` entry.
  Consider using local variable to improve readability and optimize gas consumption:
  `Vesting memory vesting = vestingMap[_beneficiary][_vestingId];`

The issues have been fixed and are not present in the latest version of the code.

Tests

- scripts for running tests/coverage depend on globally installed packages.
- `npm i -g solcjs` fails (see documentation)

This analysis was performed by SmartDec.

Evgeny Marchenko, Lead Developer
Boris Nikashin, Analyst
Alexander Seleznev, Chief Business Development Officer

July 1, 2020