Synchrocoin
Smart Contract: Final Review

Mikhail Vladimirov and Dmitry Khovratovich

11th July 2018

This document describes the audit process of the Synchrocoin smart contract performed by ABDK Consulting.

1. Introduction

We've been asked to review the Synchrocoin smart contract given as a single file. The code was partly based on the OpenZeppelin codebase. Whereas the latter is relatively safe, we have identified a number of issues that make the Synchrocoin contract insecure. Most of them were related to the migration process.

The Synchrocoin team has fixed all the major and moderate issues that were found. We identified no serious problems in the new version.

2. Remaining Issues

2.1 EIP-20 Compliance Issues

All EIP-20 requirements concerns have been fixed.

2.2 Major Flaws

All major flaws which were found in the smart contract have been fixed.

2.3 Moderate Flaws

All moderate flaws found in the token smart contract have been fixed.

2.4 Documentation Issues

Line 329, 419: the mapping has two keys. Their meaning should be described in documentation comment.

2.5 Unclear Behavior

All issues of the smart contract, where the contract behavior is unclear, have been fixed.

2.6 Suboptimal Code

This section lists suboptimal code patterns, which were found in the token smart contract.

1. Line 453: instead of `address` there probably should be `ERC223Receiver`.
2. Line 573: there should be `ERC20`, rather than `address`.

2.7 Arithmetic Overflow Issues

This section lists issues of token smart contract related to the arithmetic overflows.

Line 194: the check `assert(c / a == b)` relies on undocumented overflow behavior of Solidity. It would be better to prevent overflow rather then to detect it afterwards.