SmartDec

COTI Dime Token Security Analysis

This report is public.

Published: August 9, 2018

Abstract

In this report, we consider the security of the COTI Dime Token. Our task is to find and describe security issues in the smart contracts of the token.

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 have considered the security of COTI Dime Token smart contracts. We performed our audit according to the procedure described below.

The audit has shown neither critical nor medium severity or low severity issues.

General recommendations

The contracts code is of exceptional code quality. The contracts code does not contain issues that endanger project security. Thus, we do not have any additional recommendations.

Checklist

Security

The audit has shown no vulnerabilities. Here by vulnerabilities we mean security issues that can be exploited by an external attacker. This does not include low severity issues, documentation mismatches, overpowered contract owner, and some other kinds of bugs.

Safe arithmetics

Token smart contract is secure from arithmetics issues.

ERC20 compliance

We have checked ERC20 compliance during the audit. The audit has shown that **CotiDime.sol** contract is fully ERC20 compliant.

ERC20 MUST

The audit has shown no ERC20 “MUST” requirements violations.

ERC20 SHOULD

The audit has shown no ERC20 “SHOULD” requirements violations.

The text below is for technical use; it details the statements made in Summary, General recommendations and Checklist.

Procedure

We perform our audit according to the following procedure:

- we manually analyze smart contracts for security vulnerabilities
- we scan code with several publicly available automated Solidity analysis tools (SmartCheck, Remix) during manual analysis if it is necessary
- we check ERC20 compliance
- we reflect all the gathered information in the report

Checked vulnerabilities

We have scanned COTI Dime smart contract for commonly known and more specific vulnerabilities. Here are some of the commonly known vulnerabilities that we considered (the full list includes them but is not limited to them):

- Reentrancy
- Timestamp Dependence
- Gas Limit and Loops
- DoS with (Unexpected) Throw
- DoS with (Unexpected) revert
- DoS with Block Gas Limit
- Transaction-Ordering Dependence
- Use of tx.origin
- Exception disorder
- Gasless send
- Balance equality
- Byte array
- Transfer forwards all gas
- ERC20 API violation
- Malicious libraries
- Compiler version not fixed
- Redundant fallback function
- Send instead of transfer
- Style guide violation
- Unchecked external call
- Unchecked math
- Unsafe type inference
- Implicit visibility level
- Address hardcoded
- Using delete for arrays
- Integer overflow/underflow
- Locked money
- Private modifier
- Revert/require functions
- Using var
- Visibility
- Using blockhash
- Using SHA3
- Using suicide
- Using throw
- Using inline assembly

Project overview

In our analysis we consider COTI Dime token's code ("coti-dime-token-master.zip", sha1sum: 98d85132510827a9863feccc3ddb25f5c37abab9).

The scope of audit includes **CotiDime.sol** file.

Audit results

The token's code was completely manually analyzed and checked for ERC20 standard compliance.
All found issues (bugs, vulnerabilities and standard violations) are described below.

Critical issues

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

The audit has shown no critical issues.

Medium severity issues

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

The audit has shown no medium severity issues.

Low severity issues

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

The audit has shown no low severity issues.

This analysis was performed by SmartDec.

Evgeniy Marchenko, Lead developer
Alexander Seleznev, Chief Business Development Officer
Alexander Drygin, Analyst
Pavel Kondratenkov, Analyst

Sergey Pavlin, Chief Operating Officer 

August 9, 2018