Snowden Coin

Smart Contract Audit Report

By SigloTech S.L., October 29, 2018

www.siglotech.com

DISCLAIMER

The scope of this smart contract audit only covers smart contract code explicitly specified in this report. It does not represent a security analysis of any other contracts, employed software, or any other operational security of the company presenting the smart contract.

The audit makes no warranties or statements about the utility of the code, safety of the code, its commercial applications, viability of the business model, and legal compliance of the code in any jurisdiction. THIS DOCUMENT IS NOT AN INVESTMENT ADVICE.

No statements or claims are being made about the fitness of the smart contracts for any purpose, or their bug free status.

Introduction

Telecontrol Unterhaltungselektronik AG engaged SigloTech S.L. to perform smart contract audit for the Snowden Coin project. The objective of the audit was to evaluate quality and security of the Snowden Coin smart contracts. Audit took place between October 26, 2018 and September October 29, 2018 in Málaga, Spain and was performed by the SigloTech engineering team.

Audit Methodology

What did we audit?

SigloTech has performed an initial review and thorough review of the smart contract code deployed to Rinkeby testnet at the address https://rinkeby.etherscan.io/address/0xe46a95c450aa638de67fe517b59674d919df2b90#code.

We audited the following smart contracts files containing smart contracts:

- Owned
- SnowdenToken

File	SHA1 hash
SnowdenToken.sol	7e63eeec9f14566d2c9d80700dbd2179d52e3b10
Owned.sol	d6a22eb3716a0573977e56467bbf9e244520c0a6

How did we perform the audit?

SigloTech has followed best practices and industry-standard techniques to verify the Snowden Coin smart contracts, namely:

- We performed manual review of the code-base line by line.
- We performed automated tests.
- Two separate engineers performed the audit and we crosschecked the results.
- We tested the underlying smart contracts against common attack vectors.
- We deployed the smart contracts on the Ethereum testnet and performed live tests.

Smart Contract Structure

Owned

Owned smart contract provides basic functions for checking and transferring of ownership of the smart contract.

SnowdenToken

This smart contract is an ERC20 compliant token. The token is free from reentrancy bugs.

Coverage Report

The following table shows coverage statistics for each smart contract audited.

Smart Contract	Statements	Branches	Functions	Lines
SnowdenToken	100%	100%	100%	100%
Ownable	100%	100%	100%	100%

Issues and Vulnerabilities

Audit did not reveal any severe vulnerabilities or security issues with none of the audited smart contracts.

The SnowdenToken smart contract depends on block.timestamp and while it has minor chance of being gamed by miners, it is still a better solution then using block.number

There is a minor issue with comments of the requireTrade function. While the comments state that the function must return a Boolean value, the actual function does not return a value.

```
/**
     * @notice Ensure that account is allowed to trade
     * @param from Address of the account to send from
     * @return True if this trade is allowed
     */
    function requireTrade(address from) public view {
        require(!readOnly, "Read only mode engaged");

        uint256 i = 0;
        address current = addressLinkedList[0];
        while (current != 0) {
            if(current == from) {
                uint256 timestamp = freezeUntil[current];
                require(timestamp < block.timestamp, "Trades from your
account are temporarily not possible. This is due to ICO rules.");

                break;
            }

            current = addressLinkedList[current];
            i++;
        }
    }
```

From our perspective, there are efficiency issues associated with how the addresses are stored in the **addressLinkedList** storage variable. A more preferable way could have been using a mapping coupled with an array or a more traditional linked list solution. However, there is no problem with the logic of the current implementation.

Snowden Coin smart contract does not allow burning tokens.

Conclusion

Snowden Coin smart contracts are well written and the analysis shows that they do not contain critical vulnerabilities.