Jorge Rodriguez

CYBERSEC CONTRACT AUDIT REPORT

SOLOMON - CTDSEC.com

Introduction

During January of 2021, Solomon engaged CTDSec to audit smart contracts that they created. The engagement was technical in nature and focused on identifying security flaws in the design and implementation of the contracts. Solomon provided CTDSec with access to their code repository and whitepaper.

Disclaimer

The audit makes no statements or warranties about utility of the code, safety of the code, suitability of the business model, regulatory regime for the business model, or any other statements about fitness of the contracts to purpose, or their bugfree status. The audit documentation is for discussion purposes only.

I always recommend having a bug bounty program opened to detect future bugs.

Coverage

Target Code and Revision

For this audit, we performed research, investigation, and review of the Solomon contract followed by issue reporting, along with mitigation and remediation instructions outlined in this report. The following code files are considered in-scope for the review:

Etherscan:
https://etherscan.io/address/0x07a0ad7a9dfc3854466f8f29a173bf04bba5686e

- SlmToken.sol
- LockableToken.sol

Attacks made to the contract

In order to check for the security of the contract, we tested several attacks in order to make sure that the contract is secure and follows best practices.

Correctness of the protocol implementation [Result KO - Solved by dev team]

User funds are secure on the blockchain and cannot be transferred without user permission [Result KO - Solved by dev team]

Vulnerabilities within each component as well as secure interaction between the network components [Result OK]

Correctly passing requests to the network core [Result OK]

Data privacy, data leaking, and information integrity [Result OK]

Susceptible to reentrancy attack [Result OK]

Key management implementation: secure private key storage and proper management of encryption and signing keys [Result OK]

Handling large volumes of network traffic [Result OK]

Resistance to DDoS and similar attacks [Result OK]

Aligning incentives with the rest of the network [Result OK]

Any attack that impacts funds, such as draining or manipulating of funds [Result KO - Solved by dev team]

Mismanagement of funds via transactions [Result OK]

Inappropriate permissions and excess authority [Result KO - Solved by dev team]

Special token issuance model [Result OK]

Vulnerabilities

High risk [2 vulnerabilities]

During the cybersecurity audit, the CTDSEC team has been able to find functions that endanger the economy of the token and investor funds.

1 - Locking:

```
function lock() external onlyOwner {
        locked = true;
    }
```

Owner can make locking available, this will cause that transfers will be disabled but OWNER can make exceptions:

```
function setTradeException(address sender, bool tradeAllowed) external onlyOwner {
        require(sender != address(0), "LockableToken: Invalid address");
        lockExceptions[sender] = tradeAllowed;
    }
```

2 - Mint:

```
function mint(address to, uint256 amount) public onlyOwner {
        _mint(to, amount);
    }
```

The contract enables the owner to mine tokens at any time.

Solution:

Since the current functions of the contract can cause serious problems in the token economy, our solution is for the team to renounce ownership of the contract so that the current functions are no longer enabled and it becomes a 100% decentralized contract.

Summary of the Audit

It is important that previously publication of any contract a security audit team reviews the contract to avoid security errors and problems that may affect the investor.

We always recommend conducting an audit prior to any contract deployment.

The proposed remedial measure must be applicable to the safety of investors.

UPDATE:

After working together with the solomon team on the different proposals to solve the problem, the team has agreed to implement the resignation of ownership (We attach confirmation):

https://etherscan.io/tx/0xf128418f5f66fff99605373b6be8ed3d1e12a63653dc96caa3d6b99ce7e289b8

Having renounced ownership all problems are solved and the contract is SAFE.