Myfi smart contract

Security Audit Report

Audit Team： EXTROPY security

teamAuditdate： April15, 2021

Heco Smart Contract Security Audit Report

1. Overview

On April 15, 2021, the security team of EXTROPY received the security audit request of the **Myfi project**. The team will conduct a report on the **Myfi smart contract** from April 13, 2021 to April 15, 2021. During the audit process, the security audit experts of EXTPROPY security team communicate with the relevant interface people of the **Myfi project**, maintain information symmetry, conduct security audits under controllable operational risks, and try to avoid project generation and operation during the test process. Cause risks. Through communicate and feedback with **Myfi project** party, it is confirmed that the loopholes and risks found in the audit process have been repaired or within the acceptable range.

The result of this Myfi smart contract security audit: passed.

Audit Report MD5：949DD06B2CCB2A8D2922C5647

2. Background

2.1 Project Description

- Project name: Myfi
- official website: https://myfi.vip
- Contract type: Myfi contract
- Code language: Solidity

2.2 Audit Range

Myfi official chain token contract :

https://hecoinfo.com/address/0xC256Ff57f07faB28227CED2b7c32ba7C9689ae0b#code
https://hecoinfo.com/address/0xC129a2B60358332f0E3c936aC84a57F20EF93815#code

2.3 Security Audit List

The security experts of EXTROPY security team conduct security audits on the security audit list within the agreement, The scope of this smart contract security audit does not include new attack methods that may appear in the future, does not include the code after contract upgrades or tampering, and is not included in the subsequent cross-country, does not include cross-chain deployment, does not include project front-end code security and project platform server security.

This smart contract security audit list includes the following:

- Integer overflow
- Reentry attack
- Floating point numbers and numerical precision
- Default visibility
- Tx.origin authentication
- Wrong constructor
- Return value not verified
- Insecure random numbers
- Timestamp dependency
- Transaction order is dependent
- Delegatecall
- Call
- Denial of service
- Logic design flaws
- Fake recharge vulnerability
- Short address attack
- Uninitialized storage pointer
- Additional token issuance
- Frozen account bypass
- Access control
- Gas usage

3. Contract Structure Analysis

3.1 Directory Structure

```
└— myfi
    BasicContract.sol
    IMdex.sol
    MyfiToken.sol
    MYMasterChef.sol
```

3.2 BasicContract contract

Contract

token

ER20Burnable

- burn(uint256 amount)
- burnFrom(address account, uint256 amount)

ERC20

- name()
- symbol()
- decimals()
- totalSupply()
- balanceOf(address account)
- transfer(address recipient, uint256 amount)
- allowance(address owner, address spender)
- approve(address spender, uint256 amount)
- transferFrom(address sender, address recipient, uint256 amount)
- increaseAllowance(address spender, uint256 addedValue)
- decreaseAllowance(address spender, uint256 subtractedValue)
- _transfer(address sender, address recipient, uint256 amount)
- _mint(address account, uint256 amount)
- _burn(address account, uint256 amount)
- _approve(address owner, address spender, uint256 amount)

- setupDecimals(uint8decimals)
- _beforeTokenTransfer(address from, address to, uint256 amount)

Context

- _msgSender()
- _msgData()

Ownable

- owner()
- renounceOwnership()
- transferOwnership(address newOwner)

Operator

- operator()
- isOperator()
- transferOperator(address newOperator_)
- transferOperator(addressnewOperator)

Interface

IERC20

- totalSupply()
- balanceOf(address account)
- transfer(address recipient, uint256 amount)
- allowance(address owner, address spender)
- approve(address spender, uint256 amount)
- transferFrom(address sender, address recipient, uint256 amount)

Library

Address

- isContract(address account)
- sendValue(address payable recipient, uint256 amount)functionCall(address target, bytes memory data)
- functionCall(address target, bytes memory data, string memory errorMessage)
- functionCallWithValue(address target, bytes memory data, uint256 value)
- functionCallWithValue(address target, bytes memory data, uint256 value, string memory errorMessage)
- _functionCallWithValue(address target, bytes memory data, uint256 weiValue, string memory errorMessage)

SafeMath

3.3 IMdex contract

Contract

IMdexFactory

- createPair(address tokenA, address tokenB)()

IMdexRouter

- addLiquidity(address tokenA,address tokenB,uint amountADesired,uint amountBDesired,uint amountAMin,uint amountBMin,address to,uint deadline)
- addLiquidityETH(address token,uint amountTokenDesired,uint amountTokenMin,uint amountETHMin,address to,uint deadline)
- swapExactTokensForTokensSupportingFeeOnTransferTokens(uint amountIn,uint amountOutMin,address[] calldata path,address to,uint deadline)
- swapExactETHForTokensSupportingFeeOnTransferTokens(uint amountOutMin,address[] calldata path,address to,uint deadline)
- swapExactTokensForETHSupportingFeeOnTransferTokens(uint amountIn,uint amountOutMin,address[] calldata path,address to,uint deadline)

3.4 MyfiToken contract

Contract

Context

- _msgSender()
- _msgData()

Ownable

- owner()
- renounceOwnership()
- transferOwnership(address newOwner)

MyfiToken

- name()
- symbol()
- decimals()
- totalSupply()
- balanceOf(address account)

- transfer(address recipient, uint256 amount)
- allowance(address owner, address spender)
- approve(address spender, uint256 amount)
- transferFrom(address sender, address recipient, uint256 amount)
- increaseAllowance(address spender, uint256 addedValue)
- decreaseAllowance(address spender, uint256 subtractedValue)
- _transfer(address sender, address recipient, uint256 amount)
- _mint(address account, uint256 amount)
- _burn(address account, uint256 amount)
- _approve(address owner, address spender, uint256 amount)
- mint(address _to, uint256 _amount)
- addMinter(address _addMinter)
- delMinter(address _delMinter)
- getMinterLength()
- isMinter(address account)
- getMinter(uint256 _index)
- setFees(uint256 taxFee_,uint256 liquidityFee_,uint256 destroyFee_)
- excludeFromFee(address account,bool excluded)
- setSwapAndLiquifyEnabled(bool _enabled)
- setSellToLiquidityDif(uint256 _difBlock)
- setTaxCollector(address _taxCollector)
- _getValues(uint256 amount)
- swapAndLiquify(uint256 thisTokenBalance)
- swapTokensForEth(uint256 tokenAmount)
- addLiquidity(uint256 tokenAmount, uint256 ethAmount)

Interface

IERC20

- totalSupply()
- balanceOf(address account)
- transfer(address recipient, uint256 amount)
- allowance(address owner, address spender)
- approve(address spender, uint256 amount)
- transferFrom(address sender, address recipient, uint256 amount)

Library

Address

- isContract(address account)
- sendValue(address payable recipient, uint256 amount)

- functionCall(address target, bytes memory data)
- functionCall(address target, bytes memory data, string memory errorMessage)
- functionCallWithValue(address target, bytes memory data, uint256 value)
- functionCallWithValue(address target, bytes memory data, uint256 value, string memory errorMessage)
- _functionCallWithValue(address target, bytes memory data, uint256 weiValue, string memory errorMessage)

EnumerableSet

- _add(Set storage set, bytes32 value)
- _remove(Set storage set, bytes32 value)
- _contains(Set storage set, bytes32 value)
- _length(Set storage set)
- _at(Set storage set, uint256 index)
- add(Bytes32Set storage set, bytes32 value)
- remove(Bytes32Set storage set, bytes32 value)
- contains(Bytes32Set storage set, bytes32 value)
- length(Bytes32Set storage set)
- at(Bytes32Set storage set, uint256 index)
- add(AddressSet storage set, address value)
- remove(AddressSet storage set, address value)
- contains(AddressSet storage set, address value)
- length(AddressSet storage set)
- at(AddressSet storage set, uint256 index)
- add(UintSet storage set, uint256 value)
- remove(UintSet storage set, uint256 value)
- contains(UintSet storage set, uint256 value)
- length(UintSet storage set)
- at(UintSet storage set, uint256 index)

Math

SafeMath

SafeERC20

3.5 MYMasterChef.sol contract

Contract

Context

- _msgSender()

- _msgData()

IToken

- mint(address account, uint256 amount)

Ownable

- owner()
- renounceOwnership()
- transferOwnership(address newOwner)

ReentrancyGuard

- modifier nonReentrant()

MYMasterChef

- getUserRefInfo(address _user)
- poolLength()
- setFundAddr(address payable _fundAddr)
- setBasisSharePerBlock(uint256 _basisSharePerBlock,uint256 _halvedBlock)
- setRefBonuses(uint8[] memory _refBonuses)
- add(uint256 _allocPoint,IERC20 _want,bool _withUpdate,address _strat)
- setPoolPoint(uint256 _pid, uint256 _allocPoint, bool _withUpdate)
- setRefFeeRate(uint256 _refFeeRate)
- _repeatCheck(address _lpToken)
- isStartMining()
- getMultiplier(uint256 _from, uint256 _to)
- rewardPerBlock()
- getPoolReward(uint256 _from, uint256 _to, uint256 _allocPoint)
- pendingBasisShare(uint256 _pid, address _user)
- massUpdatePools()
- updatePool(uint256 _pid)
- deposit(uint256 _pid, uint256 _amount)
- emergencyWithdraw(uint256 _pid)
- withdraw(uint256 _pid, uint256 _lpAmount)
- withdrawBonus()
- safeBasisShareTransfer(address _to, uint256 _amount)

Interface

Library

Address

- isContract(address account)
- sendValue(address payable recipient, uint256 amount)
- functionCall(address target, bytes memory data)
- functionCall(address target, bytes memory data, string memory errorMessage)
- functionCallWithValue(address target, bytes memory data, uint256 value)
- functionCallWithValue(address target, bytes memory data, uint256 value, string memory
- errorMessage)
- _functionCallWithValue(address target, bytes memory data, uint256 weiValue, string
- memory errorMessage)

EnumerableSet

- _add(Set storage set, bytes32 value)
- _remove(Set storage set, bytes32 value)
- _contains(Set storage set, bytes32 value)
- _length(Set storage set)
- _at(Set storage set, uint256 index)
- add(Bytes32Set storage set, bytes32 value)
- remove(Bytes32Set storage set, bytes32 value)
- contains(Bytes32Set storage set, bytes32 value)
- length(Bytes32Set storage set)
- at(Bytes32Set storage set, uint256 index)
- add(AddressSet storage set, address value)
- remove(AddressSet storage set, address value)
- contains(AddressSet storage set, address value)
- length(AddressSet storage set)
- at(AddressSet storage set, uint256 index)
- add(UintSet storage set, uint256 value)
- remove(UintSet storage set, uint256 value)
- contains(UintSet storage set, uint256 value)
- length(UintSet storage set)
- at(UintSet storage set, uint256 index)

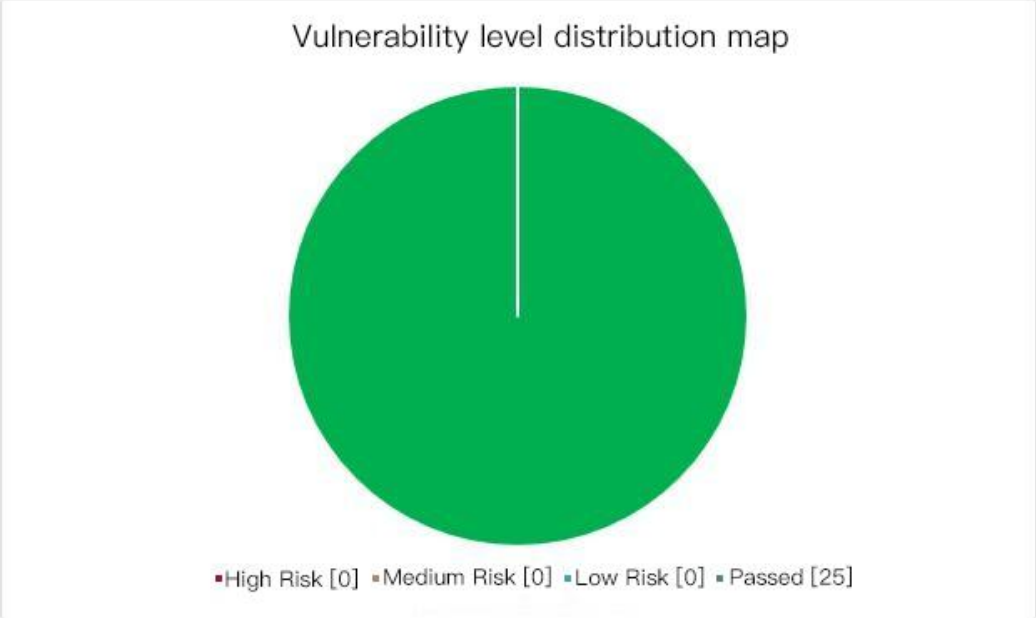
SafeMath

SafeERC20

4. Audit Details

4.1 Vulnerabilities Distribution

Vulnerabilities in this security audit are distributed by risk level, as



follows:

This smart contract security audit has 0 high-risk vulnerabilities, 0 medium-risk vulnerabilities, 0 low-risk vulnerabilities, and 25 passed, with a high security level.

4.2 Vulnerabilities Details

A security audit was conducted on the smart contract within the agreement, and no security vulnerabilities that could be directly exploited and generated security problems were found, and the security audit was passed.

4.3 Other Risks

Related issues and risks have been communicated with the official confirmation, the risks have been repaired or within the tolerable range, there are no serious risk issues.

5. Vulnerability assessment criteria

Disclaimer:

EXTPROPY security team only issues a report and assumes corresponding responsibilities for the facts that occurred or existed before the issuance of this report, Since the facts that occurred after the issuance of the report cannot determine the security status of the smart contract, it is not responsible for this.

EXTPROPY security team conducts security audits on the security audit items in the project agreement, and is not responsible for the project background and other circumstances, The subsequent on-chain deployment and operation methods of the project party are beyond the scope of this audit.

This report only conducts a security audit based on the information provided by the information provider to EXTPROPY at the time the report is issued, If the information of this project is concealed or the situation reflected is inconsistent with the actual situation, EXTPROPY security team shall not be liable for any losses and adverse effects caused thereby.

EXTROPY.IO

Security Audit Report

Audit Team： EXTROPY security

team Official Web：

http://extropy.io Phone: +44

(0)1865 261 424

Email: info@extropy.io