freshcoins

SMART CONTRACT CODE REVIEW AND SECURITY ANALYSIS REPORT

Moon Community Token

$MCT

09/11/2022

TOKEN OVERVIEW

Fees

• Buy fees: 9%

• Sell fees: 9%

Fees privileges

• Can set fees up to 100%

Ownership

• Owned

Minting

• No mint function

Max Tx Amount / Max Wallet Amount

• Can't set max tx amount and / or wallet limitations

Blacklist

• No blacklist function

Other privileges

• Can exclude from fees

TABLE OF CONTENTS

DISCLAIMER

The information provided on this analysis document is only
for general information and should not be used as a reason to invest.

FreshCoins Team will take no payment for manipulating the results
of this audit.

The score and the result will stay on this project page information
on our website https://freshcoins.io
FreshCoins Team does not guarantees that a project will not sell off
team supply, or any other scam strategy (RUG or Honeypot etc)

INTRODUCTION

FreshCoins (Consultant) was contracted by
Moon Community Token (Customer) to conduct a Smart Contract Code
Review and Security Analysis.

0x9C0387C1A50B5D48E6d234413Da62B19Bd22e7d6

Network: Binance Smart Chain (BSC)

This report presents the findings of the security assessment of
Customer's smart contract and its code review conducted on **09/11/2022**

AUDIT OVERVIEW

76

Security Score

80

Static Scan
Automatic scanning for common vulnerabilities

72

ERC Scan
Automatic checks for ERC's conformance

1 High

1 Medium

1 Low

0 Optimizations

0 Informational

No.	Issue description	Checking Status
1	Compiler Errors / Warnings	Passed
2	Reentrancy and Cross-function	Passed
3	Front running	Passed
4	Timestamp dependence	Passed
5	Integer Overflow and Underflow	Passed
6	Reverted DoS	Passed
7	DoS with block gas limit	Passed
8	Methods execution permissions	Passed
9	Exchange rate impact	Passed
10	Malicious Event	Passed
11	Scoping and Declarations	Passed
12	Uninitialized storage pointers	Passed
13	Design Logic	Passed
14	Safe Zeppelin module	Passed

OWNER PRIVILEGES

● **Contract owner can't mint tokens after initial contract deploy**

● **Contract owner can't exclude an address from transactions**

● **Contract owner can exclude/include wallet from fees**

```solidity
function excludeFromFee(address account) public onlyOwner {
    _isExcludedFromFee[account] = true;
}

function includeInFee(address account) public onlyOwner {
    _isExcludedFromFee[account] = false;
}
```

● **Contract owner can exclude/include wallet from rewards**

```solidity
function excludeFromReward(address account) public onlyOwner {
    // require(account != 0x7a250d5630B4cF539739dF2C5dAcb4c659F2488D, 'We can not exclude Uniswap router.');
    require(!_isExcluded[account], "Account is already excluded");
    if (_rOwned[account] > 0) {
        _tOwned[account] = tokenFromReflection(_rOwned[account]);
    }
    _isExcluded[account] = true;
    _excluded.push(account);
}

function includeInReward(address account) external onlyOwner {
    require(_isExcluded[account], "Account is already excluded");
    for (uint256 i = 0; i < _excluded.length; i++) {
        if (_excluded[i] == account) {
            _excluded[i] = _excluded[_excluded.length - 1];
            _tOwned[account] = 0;
            _isExcluded[account] = false;
            _excluded.pop();
            break;
        }
    }
}
```

● **Contract owner can change swap settings**

```solidity
function setSwapAndLiquifyEnabled(bool _enabled) public onlyOwner {
    swapAndLiquifyEnabled = _enabled;
    emit SwapAndLiquifyEnabledUpdated(_enabled);
}
```

● Contract owner can set fees up to 100%

```solidity
function setTaxFeePercent(uint256 taxFeeBps) external onlyOwner {
    require(taxFeeBps >= 0 && taxFeeBps <= 10**4, "Invalid bps");
    _taxFee = taxFeeBps;
}

function setLiquidityFeePercent(uint256 liquidityFeeBps)
    external
    onlyOwner
{
    require(
        liquidityFeeBps >= 0 && liquidityFeeBps <= 10**4,
        "Invalid bps"
    );
    _liquidityFee = liquidityFeeBps;
}
```

● Contract owner can renounce ownership

```solidity
function renounceOwnership() public virtual onlyOwner {
    _transferOwnership(address(0));
}
```

● Contract owner can transfer ownership

```solidity
function transferOwnership(address newOwner) public virtual onlyOwner {
    require(newOwner != address(0), "Ownable: new owner is the zero address");
    _transferOwnership(newOwner);
}
```

Recommendation:

The team should carefully manage the private keys of the owner's account. We strongly recommend a powerful security mechanism that will prevent a single user from accessing the contract admin functions. The risk can be prevented by temporarily locking the contract or renouncing ownership.

CONCLUSION AND ANALYSIS

Smart Contracts within the scope were manually reviewed and analyzed with static tools.

Audit report overview contains all found security vulnerabilities and other issues in the reviewed code.

Found no HIGH issues during the first review.

TOKEN DETAILS

Details

Buy fees:	9%
Sell fees:	9%
Max TX:	N/A
Max Sell:	N/A

Honeypot Risk

Ownership:	Owned
Blacklist:	Not detected
Modify Max TX:	Not detected
Modify Max Sell:	Not detected
Disable Trading:	Not detected

Others

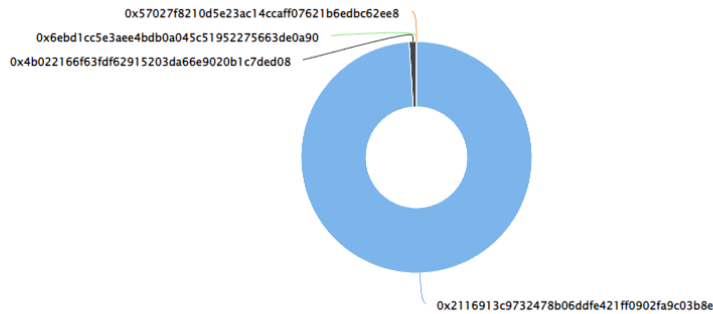|	
Liquidity:	N/A
Holders:	Clean

MTC TOKEN ANALYTICS
& TOP 10 TOKEN HOLDERS

♀ The top 10 holders collectively own 100.00%
(99,999,999,999,942.30 Tokens) of Moon Community Token 🌙

♀ Token Total Supply: 100,000,000,000,000.00 Token | Total Token Holders: 10

Moon Community Token 🌙 Top 10 Token Holders
Source: BscScan.com

0x57027f8210d5e23ac14ccaff07621b6edbc62ee8
0x6ebd1cc5e3aee4bdb0a045c51952275663de0a90
0x4b022166f63fdf62915203da66e9020b1c7ded08

0x2116913c9732478b06ddfe421ff0902fa9c03b8e

(A total of 99,999,999,999,942.30 tokens held by the top 10 accounts from the total supply of 100,000,000,000,000.00 token)

Rank	Address	Quantity (Token)	Percentage
1	0x2116913c9732478b06ddfe421ff0902fa9c03b8e	98,992,088,506,612.149451536	98.9921%
2	0x4b022166f63fdf62915203da66e9020b1c7ded08	1,000,000,001,000	1.0000%
3	0x6ebd1cc5e3aee4bdb0a045c51952275663de0a90	7,910,285,710.2	0.0079%
4	0x57027f8210d5e23ac14ccaff07621b6edbc62ee8	1,100,000	0.0000%
5	0x0db67456eeeef6df2aba2af6313b989e8df6ab77	100,000	0.0000%
6	PancakeSwap V2: MCT🌙	5,911.007902919	0.0000%
7	0xedeeff704a0c0ae871e615421dd86c769181bb33	673.187666799	0.0000%
8	0x40ac3afe94bd2be380978fab3a4692f2afe39eaa	21.827199939	0.0000%
9	0x6f219329f149b26dd197b146148885fb372a6281	11.428840561	0.0000%
10	0xf809a4156b5abd7b74a94d8a99c93ccbdadb7d07	2.47877307	0.0000%

TECHNICAL DISCLAIMER

Smart contracts are deployed and executed on the blockchain platform.
The platform, its programming language, and other software related
to the smart contract can have its vulnerabilities that can lead to hacks.
The audit can't guarantee the explicit security of the
audited project / smart contract.