freshcoins

SMART CONTRACT CODE REVIEW AND SECURITY ANALYSIS REPORT

MASTERM

$MXMX

11/08/2022

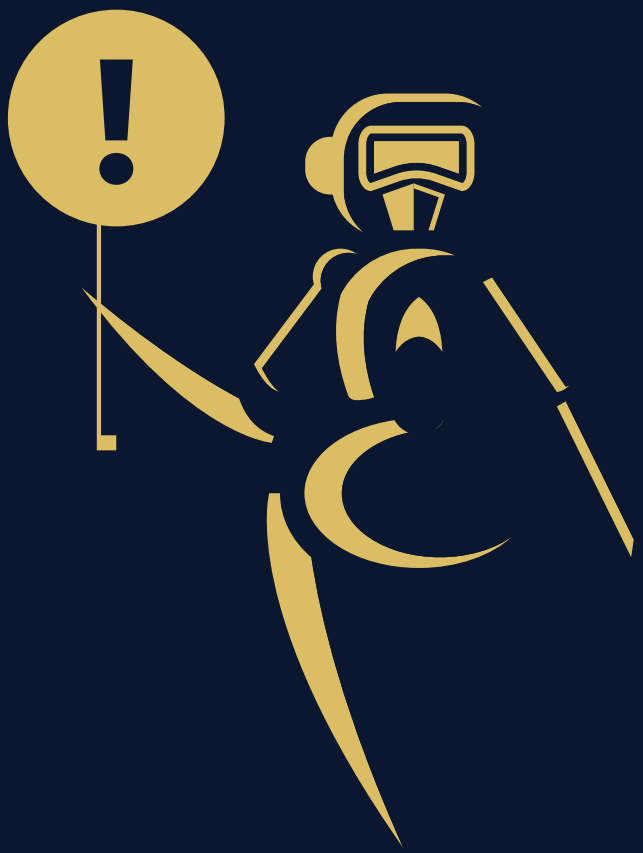
TABLE OF CONTENTS

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 MASTERM (Customer) to conduct a Smart Contract Code Review and Security Analysis.

0xc93B7e6d6445f8e7de92abDDbFBC8057CdCaA1a6

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 11/08/2022

AUDIT OVERVIEW

94

Security Score

90

Static Scan
Automatic scanning for common vulnerabilities

98

ERC Scan
Automatic checks for ERC's conformance

0 High

1 Medium

2 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(s) from tax

```
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

```
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

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

Contract owner can change fees up to 25%

```
function setTaxFeePercent(uint256 taxFeeBps) external onlyOwner {
        _taxFee = taxFeeBps;
        require(
        _taxFee + _liquidityFee + _marketingFee <= 10**4 / 4,
        'Total fee is over 25%'
        );
}

function setLiquidityFeePercent(uint256 liquidityFeeBps) external onlyOwner {
        _liquidityFee = liquidityFeeBps;
        require(
        _taxFee + _liquidityFee + _marketingFee <= 10**4 / 4,
        'Total fee is over 25%'
        );
}
```

Contract owner can renounce ownership

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

Contract owner can transfer ownership

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

function _transferOwnership(address newOwner) internal virtual {
      address oldOwner = _owner;
      _owner = newOwner;
      emit OwnershipTransferred(oldOwner, newOwner);
}
```

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

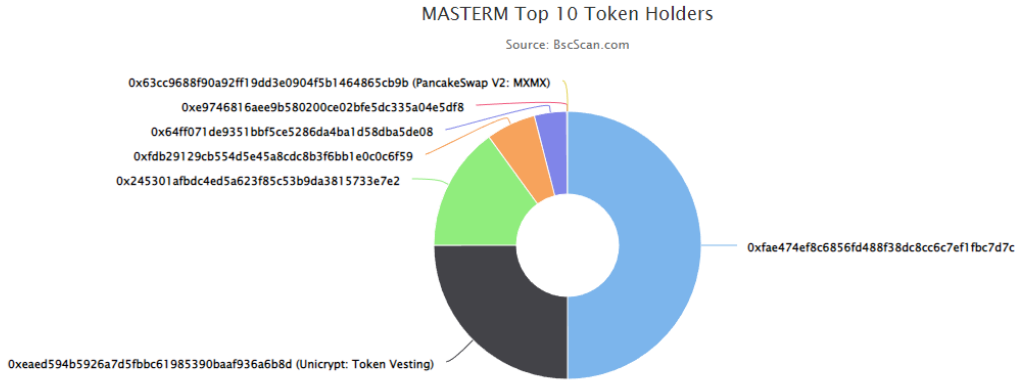
Rug Pull Risk

Liquidity:	N/A
Holders:	Clean

MASTERM TOKEN ANALYTICS & TOP 10 TOKEN HOLDERS

The top 10 holders collectively own 100.00% (99,999,991.22 Tokens) of MASTERM

Token Total Supply: 100,000,000.00 Token | Total Token Holders: 11



(A total of 99,999,991.22 tokens held by the top 10 accounts from the total supply of 100,000,000.00 token)

Rank	Address	Quantity (Token)	Percentage
1	0xfae474ef8c6856fd488f38dc8cc6c7ef1fbc7d7c	50,000,000	50.0000%
2	Unicrypt: Token Vesting	25,000,006.585551157759336589	25.0000%
3	0x245301afbdc4ed5a623f85c53b9da3815733e7e2	15,000,012.013758095762732162	15.0000%
4	0xfdb29129cb554d5e45a8cdc8b3f6bb1e0c0c6f59	6,000,000.868505808173933638	6.0000%
5	0x64ff071de9351bbf5ce5286da4ba1d58dba5de08	3,911,847.637803112036510093	3.9118%
6	0xe9746816aee9b580200ce02bfe5dc335a04e5df8	87,728.322156231375826128	0.0877%
7	PancakeSwap V2: MXMX	357.409766983284193371	0.0004%
8	0xc93b7e6d6445f8e7de92abddbfbc8057cdcaa1a6	35.122939508049795145	0.0000%
9	0x27e12f6998f60b8b82492c7d33fbc282cf4e78b8	2.374694853567091636	0.0000%
10	0xcfccee1052d26a4742ac571d3f55ba239271d71f	0.884089140089562045	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.