freshcoins

SMART CONTRACT CODE REVIEW AND SECURITY ANALYSIS REPORT

Happy 4th of July

$4THJULY

06/05/2022

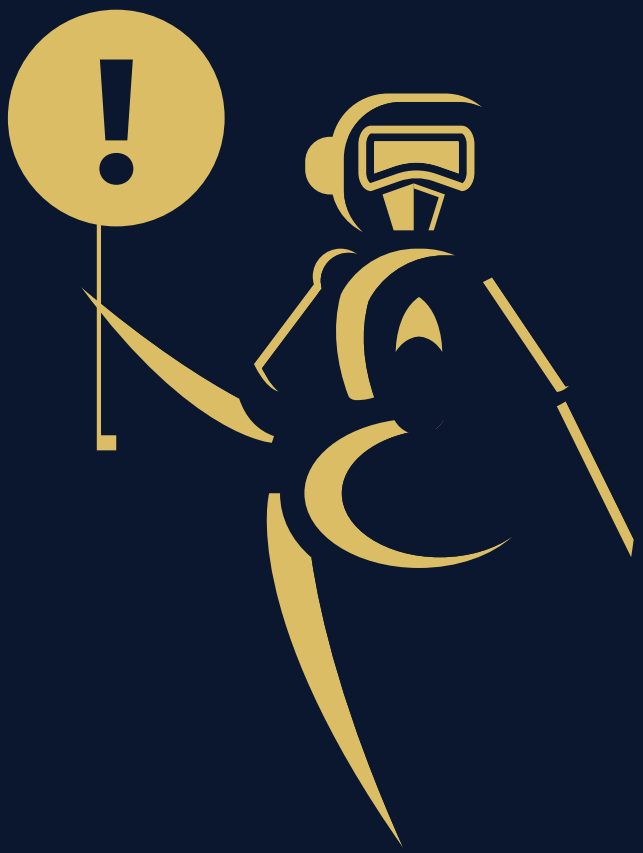
TABLE OF CONTENTS

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 Happy 4th of July (Customer) to conduct a Smart Contract Code Review and Security Analysis.

0xD3E371dB6977eEeAa338bA86c90Df2Fb3b5993d6

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 06/05/2022

AUDIT OVERVIEW

97

Security Score

97

Static Scan
Automatic scanning for common vulnerabilities

98

ERC Scan
Automatic checks for ERC's conformance

0 High

0 Medium

0 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 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(!_isExcluded[account], "Account is already excluded from reward");
    if(_rOwned[account] > 0) {
        _tOwned[account] = tokenFromReflection(_rOwned[account]);
    }
    _isExcluded[account] = true;
    _excluded.push(account);
}

function includeInReward(address account) external onlyOwner() {
    require(_isExcluded[account], "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 withdraw tokens from smart contract excluding 4THJULY tokens

```
function recoverBEP20(address tokenAddress, uint256 tokenAmount) public onlyOwner {
    // do not allow recovering self token
    require(tokenAddress != address(this), "Self withdraw");
    IERC20(tokenAddress).transfer(owner(), tokenAmount);
}
```

Contract owner can change fees up to 50%

```
uint8 public maxLiqFee = 10;
uint8 public maxTaxFee = 10;
uint8 public maxBurnFee = 10;
uint8 public maxWalletFee = 10;
uint8 public maxBuybackFee = 10;

function setAllFeePercent(uint8 taxFee, uint8 liquidityFee, uint8 burnFee, uint8 walletFee, uint8 buybackFee)
external onlyOwner() {
    require(taxFee >= 0 && taxFee <=maxTaxFee,"TF err");
    require(liquidityFee >= 0 && liquidityFee <=maxLiqFee,"LF err");
    require(burnFee >= 0 && burnFee <=maxBurnFee,"BF err");
    require(walletFee >= 0 && walletFee <=maxWalletFee,"WF err");
    require(buybackFee >= 0 && buybackFee <=maxBuybackFee,"BBF err");
    _taxFee = taxFee;
    _liquidityFee = liquidityFee;
    _burnFee = burnFee;
    _buybackFee = buybackFee;
    _walletFee = walletFee;
}
```

Contract owner can change feeWallet address

feeWallet: 0xfc4b42c273830ce2ce6d94d0139929a7f74ae405

```
function setFeeWallet(address payable newFeeWallet) external onlyOwner {
    require(newFeeWallet != address(0), "ZERO ADDRESS");
    feeWallet = newFeeWallet;
}
```

Contract owner can change max tx amount (with threshold)

min value: 1%

max value: 100%

```
uint8 public minMxTxPercentage = 1;

function setMaxTxPercent(uint256 maxTxPercent) external onlyOwner() {
    require(maxTxPercent >= minMxTxPercentage && maxTxPercent <=100,"err");
    _maxTxAmount = _tTotal.mul(maxTxPercent).div(
      10**2
    );
}
```

Contract owner can change max wallet amount (with threshold)

min value: 1%

max value: 100%

```
uint8 public minMxWalletPercentage = 1;

function setMaxWalletPercent(uint256 maxWalletPercent) external onlyOwner() {
    require(maxWalletPercent >= minMxWalletPercentage && maxWalletPercent <=100,"err");
    _maxWalletAmount = _tTotal.mul(maxWalletPercent).div(
      10**2
    );
}
```

Contract owner can change swap settings

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

Contract owner can renounce ownership

```
function renounceOwnership() public virtual onlyOwner {
        emit OwnershipTransferred(_owner, address(0));
        _owner = 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");
        emit OwnershipTransferred(_owner, newOwner);
        _owner = 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 issue during the first review.

TOKEN DETAILS

Details

Buy fees:	5%
Sell fees:	5%
Max TX:	1,000,000,000
Max Sell:	N/A

Honeypot Risk

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

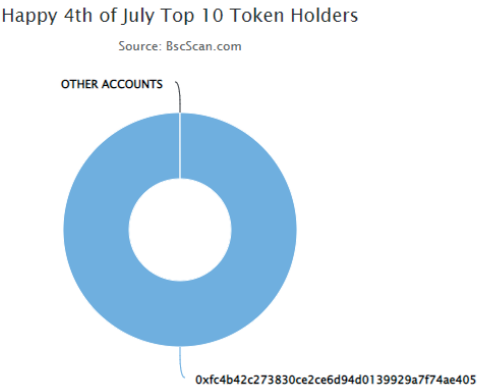
Rug Pull Risk

Liquidity:	N/A
Holders:	Clean

4THJULY TOKEN ANALYTICS & TOP 10 TOKEN HOLDERS

The top 10 holders collectively own 100.00% (1,000,000,000.00 Tokens) of Happy 4th of July

Token Total Supply: 1,000,000,000.00 Token | Total Token Holders: 1



(A total of 1,000,000,000.00 tokens held by the top 10 accounts from the total supply of 1,000,000,000.00 token)

Rank	Address	Quantity (Token)	Percentage
1	0xfc4b42c273830ce2ce6d94d0139929a7f74ae405	1,000,000,000	100.0000%

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.