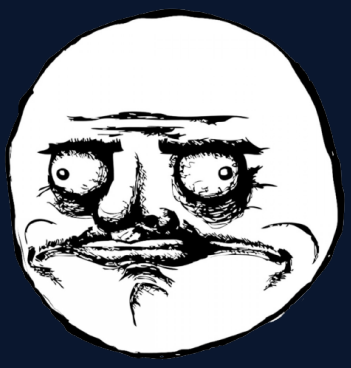
freshcoins

SMART CONTRACT CODE REVIEW AND SECURITY ANALYSIS REPORT



Me Gusta

$MEGUSTA

30/05/2023

TOKEN OVERVIEW

Fees

- Buy fees: 1%
- Sell fees: 1%

Fees privileges

- Can’t change / set fees

Ownership

- Owned

Minting

- No mint function

Max Tx Amount / Max Wallet Amount

- Can’t change / set max tx amount or max wallet amount

Blacklist

- Blacklist function not detected

Other privileges

- Can exclude / include from fees

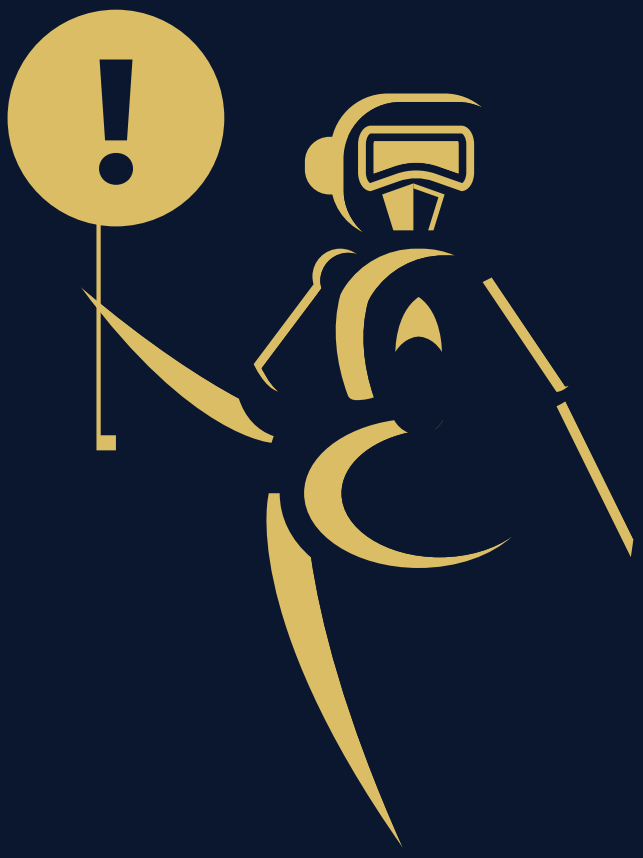
TABLE OF CONTENTS

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

0xC39996C87D2F8431DaA23BdF567A8f1445111254a

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 30/05/2023

AUDIT OVERVIEW

94

Security Score

98

Static Scan
Automatic scanning for common vulnerabilities

92

ERC Scan
Automatic checks for ERC's conformance

0 High

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

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

- Contract owner can change swap settings

```
function setSwapEnabled(bool _enabled) public onlyOwner {
    swapEnabled = _enabled;
    emit SwapEnabledUpdated(_enabled);
}
```

- Contract owner can change _marketingAddress and _charityAddress addresses

Current values:

_marketingAddress : N/A

_charityAddress : N/A

```
function setMarketingAddress(address payable newMarketingAddress) external onlyOwner {
    if (newMarketingAddress == address(0)) {
        require(currentMarketingFee == 0, "Marketing fee must be zero when set as the zero address");
    }

    _marketingAddress = newMarketingAddress;
}

function setCharityAddress(address payable newCharityAddress) external onlyOwner {
    if (newCharityAddress == address(0)) {
        require(currentCharityFee == 0, "Charity fee must be zero when set as the zero address");
    }

    _charityAddress = newCharityAddress;
}
```

- **Contract owner has the ability to set the tax fees to 0 and revert them back to their previous values.**

```
function restoreCharityFee() external onlyOwner {
    currentCharityFee = _charityFee;
}

function restoreMarketingFee() external onlyOwner {
    currentMarketingFee = _marketingFee;
}

function restoreAllFees() external onlyOwner {
    currentCharityFee = _charityFee;
    currentMarketingFee = _marketingFee;
}

function removeCharityFee() external onlyOwner {
    currentCharityFee = 0;
}

function removeMarketingFee() external onlyOwner {
    currentMarketingFee = 0;
}

function removeAllFees() external onlyOwner {
    if (currentCharityFee == 0 && currentMarketingFee == 0) return;

    currentCharityFee = 0;
    currentMarketingFee = 0;
}
```

- **Contract owner can withdraw tokens from smart contract**

```
function withdrawExcessETH(address payable ethReceiver, uint256 ethToWithdraw) external nonReentrant
onlyOwner {
    require(ethToWithdraw < address(this).balance, "Not enough ETH stored on the contract");

    (bool success, ) = ethReceiver.call{value: ethToWithdraw}("");
    require(success, "Unable to send to given address");
}
```

- **Contract owner can renounce 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);
}
```

Contract owner can transfer ownership

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

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:	1%
Sell fees:	1%
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:	100% unlocked tokens

MEGUSTA TOKEN ANALYTICS & TOP 10 TOKEN HOLDERS

The top 10 holders collectively own 100.00% (100,000,000,000,000.00 Tokens) of Me Gusta

Token Total Supply: 100,000,000,000,000.00 Token | Total Token Holders: 1

(A total of 100,000,000,000,000.00 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	0x598da1eeb3b66b38ee94e6575903054aac3a3d85	100,000,000,000,000	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.