



SMART CONTRACT CODE REVIEW AND SECURITY ANALYSIS REPORT



Meta Simulator Coin
\$MSC



22/01/2022



TABLE OF CONTENTS

- 1 DISCLAIMER
- 2 INTRODUCTION
- 3-4 WEBSITE DIAGNOSTIC
- 5-6 AUDIT OVERVIEW
- 7-9 OWNER PRIVILEGES
- 10 CONCLUSION AND ANALYSIS
- 11 TOKEN DETAILS
- 12 META SIMULATOR COIN TOKEN DISTRIBUTION & TOP 10 TOKEN HOLDERS
- 13 TECHNICAL DISCLAIMER



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 Meta Simulator Coin (Customer) to conduct a Smart Contract Code Review and Security Analysis.

0x260e9Ee08b9dcC108BfaB26665CF24990d59997e

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 22/01/2022



WEBSITE DIAGNOSTIC

<https://metasimulatorcoin.com/>



0-49



50-89



90-100



Performance



Accessibility



Best Practices



SEO



Progressive
Web App

Metrics



First Contentful Paint

3.4 s



Time to interactive

8.2 s



Speed Index

6.1 s



Total Blocking Time

1,570 ms



Large Contentful Paint

11.9 s



Cumulative Layout Shift

0.033

WEBSITE IMPROVEMENTS

Eliminate render-blocking resources

Reduce unused CSS

Reduce unused JavaScript

Ensure text remains visible during webfont load

Image elements do not have explicit `width` and `height`

Avoid enormous network payloads Total size was 2,947 KiB

Buttons do not have an accessible name

Background and foreground colors do not have a sufficient contrast ratio

AUDIT OVERVIEW



Security Score



Static Scan
Automatic scanning for common vulnerabilities



ERC Scan
Automatic checks for ERC's conformance



High



Medium



Low



Optimizations



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.

Owner must be a contract with transparent rules for using `setSwapEnabled` and `setIsBanned` functions

Contract owner can exclude/include wallet from fees

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

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

Contract owner can exclude/include wallet from reward

```
function excludeFromReward(address account) public onlyOwner() {
    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 included");
    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 max tx amount

```
function setMaxTxPercent(uint256 maxTxPercent) external onlyOwner() {
    _maxTxAmount = _tTotal.mul(maxTxPercent).div(
        10**2
    );
}
```

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);
}
```

Contract owner can change buy fees

```
function setBuyTaxFeePercent(uint256 taxFee) external onlyOwner {
    _buyTaxFee = taxFee;
    require(_buyTaxFee + _buyMarketingFee + _buyLiquidityFee <= 20, "Total fees exceed 20%");
}

function setBuyMarketingFeePercent(uint256 marketingFee) external onlyOwner() {
    _buyMarketingFee = marketingFee;
    require(_buyTaxFee + _buyMarketingFee + _buyLiquidityFee <= 20, "Total fees exceed 20%");
}

function setBuyLiquidityFeePercent(uint256 liquidityFee) external onlyOwner() {
    _buyLiquidityFee = liquidityFee;
    require(_buyTaxFee + _buyMarketingFee + _buyLiquidityFee <= 20, "Total fees exceed 20%");
}
```

Contract owner can change sell fees

```
function setSellTaxFeePercent(uint256 taxFee) external onlyOwner() {
    _sellTaxFee = taxFee;
    require(_sellTaxFee + _sellMarketingFee + _sellLiquidityFee <= 25, "Total fees exceed 25%");
}

function setSellMarketingFeePercent(uint256 marketingFee) external onlyOwner() {
    _sellMarketingFee = marketingFee;
    require(_sellTaxFee + _sellMarketingFee + _sellLiquidityFee <= 25, "Total fees exceed 25%");
}

function setSellLiquidityFeePercent(uint256 liquidityFee) external onlyOwner() {
    _sellLiquidityFee = liquidityFee;
    require(_sellTaxFee + _sellMarketingFee + _sellLiquidityFee <= 25, "Total fees exceed 25%");
}
```

Contract owner can change swap settings

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

Contract owner can enable/disable trading

```
function setSwapEnabled(bool _enabled) external onlyOwner() {
    swapEnabled = _enabled;
    emit SetSwapEnabled(_enabled);
}
```

Contract owner can change sell tx amount

```
function setNumTokensSellToAddToLiquidity(uint256 _amount) external onlyOwner() {
    require(_amount > 0);
    numTokensSellToAddToLiquidity = _amount;
    emit NumTokensSellToAddToLiquidityUpdated(_amount);
}
```

Contract owner can enable/disable contract fees

```
function setContractFeesEnabled(bool _bool) external onlyOwner() {
    _contractFeesEnabled = _bool;
    emit SetContractFeesEnabled(_bool);
}
```

Contract owner can exclude an address from transactions

Note: `setIsBanned` function can only ban or unban contract addresses

```
function setIsBanned(address _address, bool value) external onlyOwner() {
    require(_address.isContract(), "Error: Can only ban or unban contract addresses");
    _isBanned[_address] = value;
    emit SetIsBanned(_address, value);
}
```

Recomandation:

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 issue during the first review.

TOKEN DETAILS

Details

Buy fees:	13%
Sell fees:	16%
Max TX:	10000000000000000000000000000000
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



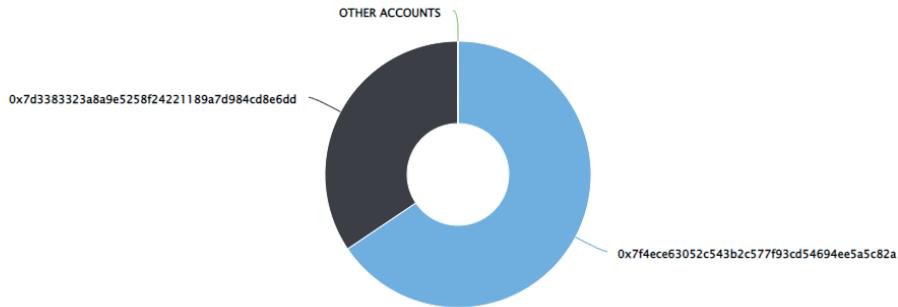
META SIMULATOR COIN TOKEN DISTRIBUTION & TOP 10 TOKEN HOLDERS

💡 The top 100 holders collectively own 100.00% (10,000,000,000.00 Tokens) of Meta Simulator Coin

📍 Token Total Supply: 10,000,000,000.00 Token | Total Token Holders: 2

Meta Simulator Coin Top 100 Token Holders

Source: BscScan.com



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

Rank	Address	Quantity (Token)	Percentage
1	0x7f4ece63052c543b2c577f93cd54694ee5a5c82a	6,556,640,000	65.5664%
2	0xd3383323a8a9e5258f24221189a7d984cd8e6dd	3,443,360,000	34.4336%

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.

