Cyberscope

# Audit Report

# MetaCubez

August 2022

| | |
|---|---|
| Type | BEP20 |
| Network | BSC |
| Address | 0x228c6616db646ba6aa7eeef4ede34adb4770d5bf |
| Audited by | © cyberscope |

# Table of Contents

# Contract Review

| | |
|---|---|
| **Contract Name** | MetaCubez |
| **Compiler Version** | v0.6.12+commit.27d51765 |
| **Optimization** | 200 runs |
| **Licence** | Unlicense |
| **Explorer** | https://bscscan.com/token/0x228c6616db646ba6aa7eeef4ede34adb4770d5bf |
| **Symbol** | MCubez |
| **Decimals** | 4 |
| **Total Supply** | 200,000,000 |
| **Domain** | https://metacubez.io |

# Source Files

| **Filename** | **SHA256** |
|---|---|
| **contract.sol** | c902c96af01e66353cdf0baab5d3b7daa71b04aa899249806db9851fefee983f |

# Audit Updates

| | |
|---|---|
| **Initial Audit** | 16th August 2022 |
| **Corrected** | |

# Contract Analysis

Critical Medium Minor Pass

| Severity | Code | Description |
|---|---|---|
| | ST | Contract Owner is not able to stop or pause transactions |
| | OCTD | Contract Owner is not able to transfer tokens from specific address |
| | OTUT | Owner Transfer User's Tokens |
| | ELFM | Contract Owner is not able to increase fees more than a reasonable percent (25%) |
| | ULTW | Contract Owner is not able to increase the amount of liquidity taken by dev wallet more than a reasonable percent |
| | MT | Contract Owner is not able to mint new tokens |
| | BT | Contract Owner is not able to burn tokens from specific wallet |
| | BC | Contract Owner is not able to blacklist wallets from selling |

# ST - Stop Transactions

| | |
|---|---|
| **Criticality** | medium |
| **Location** | contract.sol#L992 |

## Description

The contract owner has the authority to stop transactions for all users excluding the owner. The owner may take advantage of it by setting the _maxTxAmount to zero.

```
function _transfer( address from, address to, uint256 amount ) private {
        require(from != address(0), "ERC20: transfer from the zero address");
        require(to != address(0), "ERC20: transfer to the zero address");
        require(amount > 0, "Transfer amount must be greater than zero");
        if(from != owner() && to != owner())
            require(amount <= _maxTxAmount, "Transfer amount exceeds the maxTxAmount.");
```

The contract owner has the authority to stop transactions for all users excluding the owner. The owner may take advantage of it by setting the taxFee and the liquidityFee to the maximum value.

```
_tokenTransfer(from,to,amount,takeFee);
```

## Recommendation

The contract could embody a check for not allowing setting the _maxTxAmount and transaction fees less than a reasonable amount.

A suggested implementation could check that the maximum _maxTxAmount should be more than a fixed percentage of the total supply.

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. That risk can be prevented by temporarily locking the contract or renouncing ownership.

# ELFM - Exceed Limit Fees Manipulation

| | |
|---|---|
| **Criticality** | critical |
| **Location** | contract.sol#L880,884 |

## Description

The contract owner has the authority to increase over the allowed limit of 25%. The owner may take advantage of it by calling the `setTaxFeePercent` function with a high percentage value.

```
function setTaxFeePercent(uint256 taxFee) external onlyOwner() {
        _taxFee = taxFee;
}

function setLiquidityFeePercent(uint256 liquidityFee) external onlyOwner() {
        _liquidityFee = liquidityFee;
}
```

## Recommendation

The contract could embody a check for the maximum acceptable value.

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. That risk can be prevented by temporarily locking the contract or renouncing ownership.

# Contract Diagnostics

Critical Medium Minor

| Severity | Code | Description |
|---|---|---|
| | FSA | Fixed Swap Address |
| | L01 | Public Function could be Declared External |
| | L02 | State Variables could be Declared Constant |
| | L04 | Conformance to Solidity Naming Conventions |
| | L07 | Missing Events Arithmetic |
| | L09 | Dead Code Elimination |

# FSA - Fixed Swap Address

| | |
|---|---|
| **Criticality** | minor |
| **Location** | contract.sol#L741 |

## Description

The swap address is assigned once in the constructor and it can not be changed. The decentralized swaps sometimes create a new swap version or abandon the current. A contract that cannot change the swap address may not be able to catch-up the upgrade.

```
constructor () public {
        _rOwned[_msgSender()] = _rTotal;

        IUniswapV2Router02 _uniswapV2Router =
IUniswapV2Router02(0x05fF2B0DB69458A0750badebc4f9e13aDd608C7F);
         // Create a uniswap pair for this new token
        uniswapV2Pair = IUniswapV2Factory(_uniswapV2Router.factory())
            .createPair(address(this), _uniswapV2Router.WETH());

        // set the rest of the contract variables
        uniswapV2Router = _uniswapV2Router;
```

## Recommendation

It could be better to allow the swap address mutation in case of future swap updates.

# L01 - Public Function could be Declared External

| | |
|---|---|
| **Criticality** | minor |
| **Location** | contract.sol#L771,431,809,762,750,776,890,440,872,446,758,801,459,835,805,818,754,785,780,976,451,791,796,868 |

## Description

Public functions that are never called by the contract should be declared external to save gas.

```
excludeFromFee
decreaseAllowance
increaseAllowance
lock
isExcludedFromFee
approve
transferFrom
symbol
reflectionFromToken
...
```

## Recommendation

Use the external attribute for functions never called from the contract.

# L02 - State Variables could be Declared Constant

| | |
|---|---|
| **Criticality** | minor |
| **Location** | contract.sol#L700,699,695,716,701 |

## Description

Constant state variables should be declared constant to save gas.

```
_decimals
numTokensSellToAddToLiquidity
_tTotal
_name
_symbol
```

## Recommendation

Add the constant attribute to state variables that never change.

# L04 - Conformance to Solidity Naming Conventions

| | |
|---|---|
| **Criticality** | minor |
| **Location** | contract.sol#L715,504,521,955,503,706,949,543,703,890 |

## Description

Solidity defines a naming convention that should be followed. Rule exceptions:

- Allow constant variable name/symbol/decimals to be lowercase.
- Allow _ at the beginning of the mixed_case match for private variables and unused parameters.

```
_enabled
_taxFee
WETH
_amount
_liquidityFee
DOMAIN_SEPARATOR
MINIMUM_LIQUIDITY
PERMIT_TYPEHASH
_maxTxAmount
...
```

## Recommendation

Follow the Solidity naming convention.
https://docs.soliditylang.org/en/v0.4.25/style-guide.html#naming-conventions.

# L07 - Missing Events Arithmetic

| | |
|---|---|
| **Criticality** | minor |
| **Location** | contract.sol#L884,876,880 |

## Description

Detected missing events for critical arithmetic parameters. There are functions that have no event emitted, so it is difficult to track off-chain changes.

```
_liquidityFee = liquidityFee
_taxFee = taxFee
_maxTxAmount = _tTotal.mul(maxTxPercent).div(10 ** 2)
```

## Recommendation

Emit an event for critical parameter changes.

# L09 - Dead Code Elimination

| | |
|---|---|
| **Criticality** | minor |
| **Location** | contract.sol#L291,342,352,357,264,327,317 |

## Description

Functions that are not used in the contract, and make the code's size bigger.

```
functionCall
isContract
_functionCallWithValue
functionCallWithValue
sendValue
```

## Recommendation

Remove unused functions.

# Contract Functions

| Contract | Type | Bases | | |
|---|---|---|---|---|
| | Function Name | Visibility | Mutability | Modifiers |
| | | | | |
| **IERC20** | Interface | | | |
| | totalSupply | External | | - |
| | balanceOf | External | | - |
| | transfer | External | ✓ | - |
| | allowance | External | | - |
| | approve | External | ✓ | - |
| | transferFrom | External | ✓ | - |
| | | | | |
| **SafeMath** | Library | | | |
| | add | Internal | | |
| | sub | Internal | | |
| | sub | Internal | | |
| | mul | Internal | | |
| | div | Internal | | |
| | div | Internal | | |
| | mod | Internal | | |
| | mod | Internal | | |
| | | | | |
| **Context** | Implementation | | | |
| | _msgSender | Internal | | |
| | _msgData | Internal | | |
| | | | | |
| **Address** | Library | | | |
| | isContract | Internal | | |
| | sendValue | Internal | ✓ | |
| | functionCall | Internal | ✓ | |
| | functionCall | Internal | ✓ | |
| | functionCallWithValue | Internal | ✓ | |

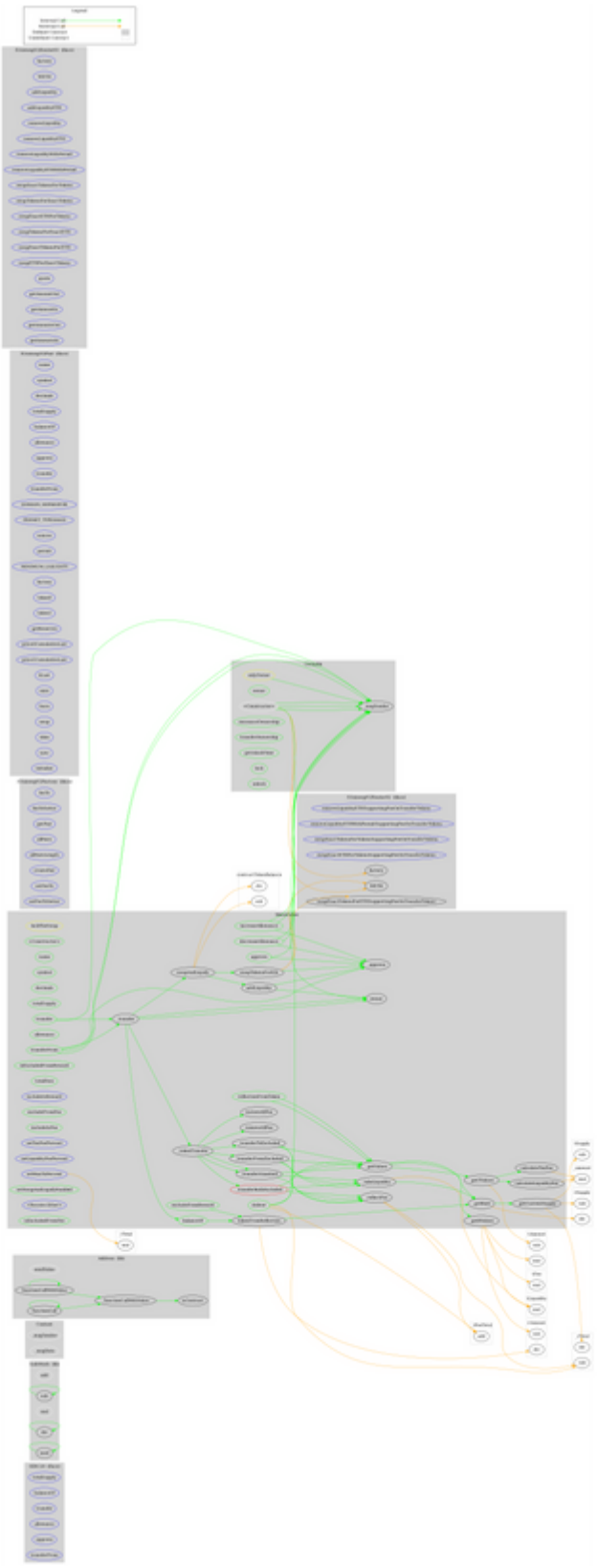
| | | | | |
|---|---|---|---|---|
| | functionCallWithValue | Internal | ✓ | |
| | _functionCallWithValue | Private | ✓ | |
| | | | | |
| **Ownable** | Implementation | Context | | |
| | <Constructor> | Internal | ✓ | |
| | owner | Public | | - |
| | renounceOwnership | Public | ✓ | onlyOwner |
| | transferOwnership | Public | ✓ | onlyOwner |
| | geUnlockTime | Public | | - |
| | lock | Public | ✓ | onlyOwner |
| | unlock | Public | ✓ | - |
| | | | | |
| **IUniswapV2Factory** | Interface | | | |
| | feeTo | External | | - |
| | feeToSetter | External | | - |
| | getPair | External | | - |
| | allPairs | External | | - |
| | allPairsLength | External | | - |
| | createPair | External | ✓ | - |
| | setFeeTo | External | ✓ | - |
| | setFeeToSetter | External | ✓ | - |
| | | | | |
| **IUniswapV2Pair** | Interface | | | |
| | name | External | | - |
| | symbol | External | | - |
| | decimals | External | | - |
| | totalSupply | External | | - |
| | balanceOf | External | | - |
| | allowance | External | | - |
| | approve | External | ✓ | - |
| | transfer | External | ✓ | - |
| | transferFrom | External | ✓ | - |
| | DOMAIN_SEPARATOR | External | | - |

| | | | | |
|---|---|---|---|---|
| | PERMIT_TYPEHASH | External | | - |
| | nonces | External | | - |
| | permit | External | ✓ | - |
| | MINIMUM_LIQUIDITY | External | | - |
| | factory | External | | - |
| | token0 | External | | - |
| | token1 | External | | - |
| | getReserves | External | | - |
| | price0CumulativeLast | External | | - |
| | price1CumulativeLast | External | | - |
| | kLast | External | | - |
| | mint | External | ✓ | - |
| | burn | External | ✓ | - |
| | swap | External | ✓ | - |
| | skim | External | ✓ | - |
| | sync | External | ✓ | - |
| | initialize | External | ✓ | - |
| | | | | |
| **IUniswapV2Router01** | Interface | | | |
| | factory | External | | - |
| | WETH | External | | - |
| | addLiquidity | External | ✓ | - |
| | addLiquidityETH | External | Payable | - |
| | removeLiquidity | External | ✓ | - |
| | removeLiquidityETH | External | ✓ | - |
| | removeLiquidityWithPermit | External | ✓ | - |
| | removeLiquidityETHWithPermit | External | ✓ | - |
| | swapExactTokensForTokens | External | ✓ | - |
| | swapTokensForExactTokens | External | ✓ | - |
| | swapExactETHForTokens | External | Payable | - |
| | swapTokensForExactETH | External | ✓ | - |
| | swapExactTokensForETH | External | ✓ | - |
| | swapETHForExactTokens | External | Payable | - |
| | quote | External | | - |

| | | | | |
|---|---|---|---|---|
| | getAmountOut | External | | - |
| | getAmountIn | External | | - |
| | getAmountsOut | External | | - |
| | getAmountsIn | External | | - |
| | | | | |
| **IUniswapV2Router02** | Interface | IUniswapV2Router01 | | |
| | removeLiquidityETHSupportingFeeOnTransferTokens | External | ✓ | - |
| | removeLiquidityETHWithPermitSupportingFeeOnTransferTokens | External | ✓ | - |
| | swapExactTokensForTokensSupportingFeeOnTransferTokens | External | ✓ | - |
| | swapExactETHForTokensSupportingFeeOnTransferTokens | External | Payable | - |
| | swapExactTokensForETHSupportingFeeOnTransferTokens | External | ✓ | - |
| | | | | |
| **MetaCubez** | Implementation | Context, IERC20, Ownable | | |
| | <Constructor> | Public | ✓ | - |
| | name | Public | | - |
| | symbol | Public | | - |
| | decimals | Public | | - |
| | totalSupply | Public | | - |
| | balanceOf | Public | | - |
| | transfer | Public | ✓ | - |
| | allowance | Public | | - |
| | approve | Public | ✓ | - |
| | transferFrom | Public | ✓ | - |
| | increaseAllowance | Public | ✓ | - |
| | decreaseAllowance | Public | ✓ | - |
| | isExcludedFromReward | Public | | - |
| | totalFees | Public | | - |
| | deliver | Public | ✓ | - |
| | reflectionFromToken | Public | | - |
| | tokenFromReflection | Public | | - |

| | | | | |
|---|---|---|---|---|
| | excludeFromReward | Public | ✓ | onlyOwner |
| | includeInReward | External | ✓ | onlyOwner |
| | _transferBothExcluded | Private | ✓ | |
| | excludeFromFee | Public | ✓ | onlyOwner |
| | includeInFee | Public | ✓ | onlyOwner |
| | setTaxFeePercent | External | ✓ | onlyOwner |
| | setLiquidityFeePercent | External | ✓ | onlyOwner |
| | setMaxTxPercent | External | ✓ | onlyOwner |
| | setSwapAndLiquifyEnabled | Public | ✓ | onlyOwner |
| | <Receive Ether> | External | Payable | - |
| | _reflectFee | Private | ✓ | |
| | _getValues | Private | | |
| | _getTValues | Private | | |
| | _getRValues | Private | | |
| | _getRate | Private | | |
| | _getCurrentSupply | Private | | |
| | _takeLiquidity | Private | ✓ | |
| | calculateTaxFee | Private | | |
| | calculateLiquidityFee | Private | | |
| | removeAllFee | Private | ✓ | |
| | restoreAllFee | Private | ✓ | |
| | isExcludedFromFee | Public | | - |
| | _approve | Private | ✓ | |
| | _transfer | Private | ✓ | |
| | swapAndLiquify | Private | ✓ | lockTheSwap |
| | swapTokensForEth | Private | ✓ | |
| | addLiquidity | Private | ✓ | |
| | _tokenTransfer | Private | ✓ | |
| | _transferStandard | Private | ✓ | |
| | _transferToExcluded | Private | ✓ | |
| | _transferFromExcluded | Private | ✓ | |

# Contract Flow



# Domain Info

| | |
|---|---|
| **Domain Name** | metacubez.io |
| **Registry Domain ID** | 2cc0117a5fa0419fa6d4ad26d8082ef1-DONUTS |
| **Creation Date** | 2022-04-12T21:12:51Z |
| **Updated Date** | 2022-05-20T12:35:30Z |
| **Registry Expiry Date** | 2024-04-12T21:12:51Z |
| **Registrar WHOIS Server** | http://www.hostinger.com |
| **Registrar URL** | http://www.hostinger.com |
| **Registrar** | Hostinger, UAB |
| **Registrar IANA ID** | 1636 |

The domain has been created in over 1 year before the creation of the audit.

There is no public billing information, the creator is protected by the privacy settings.

# Summary

There are some functions that can be abused by the owner like stopping transactions and manipulating fees. A multi-wallet signing pattern will provide security against potential hacks. Temporarily locking the contract or renouncing ownership will eliminate all the contract threats.

# Disclaimer

All the content provided in this document is for general information only and should not be used as financial advice or a reason to buy any investment.

Cyberscope team provides no guarantees against the sale of team tokens or the removal of liquidity by the project audited in this document. Always Do your own research and protect yourselves from being scammed.

The Cyberscope team has audited this project for general information and only expresses their opinion based on similar projects and checks from popular diagnostic tools. Under no circumstances did Cyberscope receive a payment to manipulate those results or change the awarding badge that we will be adding in our website.

Always Do your own research and protect yourselves from scams. This document should not be presented as a reason to buy or not buy any particular token.

The Cyberscope team disclaims any liability for the resulting losses.

# About Cyberscope

Coinscope audit and K.Y.C. service has been rebranded to Cyberscope.

Coinscope is the leading early coin listing, voting and auditing authority firm. The audit process is analyzing and monitoring many aspects of the project. That way, it gives the community a good sense of security using an informative report and a generic score.

Cyberscope and Coinscope are aiming to make crypto discoverable and efficient globally. They provide all the essential tools to assist users draw their own conclusions.

The Cyberscope team

https://www.cyberscope.io