Cyberscope

Audit Report

Magic manor

July 2022

Type	BEP20
Network	BSC
Address	0x7773feaf976599a9d6a3a7b5dc43d02ac166f255
Audited by	© cyberscope

Table of Contents

Table of Contents 1

Contract Review 3

Source Files 3

Audit Updates 3

Contract Analysis 4

ST - Stop Transactions 5

Description 5

Recommendation 5

OCTD - Owner Contract Tokens Drain 6

Description 6

Recommendation 6

Contract Diagnostics 7

STC - Succeeded Transfer Check 8

Description 8

Recommendation 8

L01 - Public Function could be Declared External 9

Description 9

Recommendation 9

L02 - State Variables could be Declared Constant 10

Description 10

Recommendation 10

L04 - Conformance to Solidity Naming Conventions 11

Description 11

Recommendation 11

L05 - Unused State Variable 12

Description 12

Recommendation 12
L09 - Dead Code Elimination 13
Description 13
Recommendation 13
L11 - Unnecessary Boolean equality 14
Description 14
Recommendation 14
Contract Functions 15
Contract Flow 20
Domain Info 21
Summary 22
Disclaimer 23
About Cyberscope 24

Contract Review

Contract Name	MGC
Compiler Version	v0.8.13+commit.abaa5c0e
Optimization	200 runs
Licence	None
Explorer	https://bscscan.com/token/0x7773feaf976599a9d6a3a7b5dc43d02ac166f255
Symbol	MGC
Decimals	18
Total Supply	210,000,000
Domain	https://metaswap.cx/

Source Files

Filename	**SHA256**
contract.sol	8461228e011baab54f6a7ccb8753b46e3a049445f2d110a60c37e98680916e1e

Audit Updates

Initial Audit	12th July 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#L499

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 tradingOpen to false.

```
require(tradingOpen, "Trade not open.");
```

Recommendation

The contract could embody a check for not allowing setting the tradingOpen to false.

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.

OCTD - Owner Contract Tokens Drain

Criticality	minor
Location	contract.sol#L568,L572

Description

The contract owner has the authority to claim all the balance of the contract. The owner may take advantage of it by calling the `emergencyWithdraw` and `emergencyWithdrawToken` functions.

```
function emergencyWithdraw() external onlyOwner {
        payable(owner()).transfer(address(this).balance);
}

function emergencyWithdrawToken(address token, uint256 amount) external onlyOwner {
    IERC20(token).transfer(msg.sender, amount);
}
```

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

Contract Diagnostics

● Critical ● Medium ● Minor

Severity	Code	Description
●	STC	Succeeded Transfer Check
●	L01	Public Function could be Declared External
●	L02	State Variables could be Declared Constant
●	L04	Conformance to Solidity Naming Conventions
●	L05	Unused State Variable
●	L09	Dead Code Elimination
●	L11	Unnecessary Boolean equality

STC - Succeeded Transfer Check

Criticality	minor
Location	contract.sol#L568,L572

Description

According to the ERC20 specification, the transfer methods should be checked if the result is successful. Otherwise, the contract may wrongly assume that the transfer has been established.

```
function emergencyWithdraw() external onlyOwner {
        payable(owner()).transfer(address(this).balance);
}

function emergencyWithdrawToken(address token, uint256 amount) external onlyOwner {
        IERC20(token).transfer(msg.sender, amount);
}
```

Recommendation

The contract should check if the result of the transfer methods is successful.

L01 - Public Function could be Declared External

Criticality	minor
Location	contract.sol#L464,110,558,459,422,426,444,118,439,410,550,418,541,430,435,414

Description

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

```
symbol
allowance
transfer
deliver
decimals
setExcludeds
name
approve
transferOwnership
...
```

Recommendation

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

L02 - State Variables could be Declared Constant

Criticality	minor
Location	contract.sol#L366,367

Description

Constant state variables should be declared constant to save gas.

```
_symbol
_name
```

Recommendation

Add the constant attribute to state variables that never change.

L04 - Conformance to Solidity Naming Conventions

Criticality	minor
Location	contract.sol#L402,198,386,199,541,376,213,369,230

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.

```
WETH
DEXs
MINIMUM_LIQUIDITY
_swapV2Router
_lp
PERMIT_TYPEHASH
_route
DOMAIN_SEPARATOR
_pair
```

Recommendation

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

L05 - Unused State Variable

Criticality	minor
Location	contract.sol#L367,366

Description

There are segments that contain unused state variables.

```
_name
_symbol
```

Recommendation

Remove unused state variables.

L09 - Dead Code Elimination

Criticality	minor
Location	contract.sol#L141,151,154,158,135,148,145

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.

L11 - Unnecessary Boolean equality

Criticality	minor
Location	contract.sol#L541,550

Description

The comparison to boolean constants is redundant. Boolean constants can be used directly and do not need to be compared to true or false.

```
isExcludeds[addr] == true
isExcludeds[lp] == true
```

Recommendation

Remove the equality to the boolean constant.

Contract Functions

Contract	Type	Bases		
	Function Name	**Visibility**	**Mutability**	**Modifiers**
IERC20	Interface			
	totalSupply	External		-
	balanceOf	External		-
	transfer	External	✓	-
	allowance	External		-
	approve	External	✓	-
	transferFrom	External	✓	-
IERC20Metadata	Interface	IERC20		
	name	External		-
	symbol	External		-
	decimals	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		
Ownable	Implementation	Context		

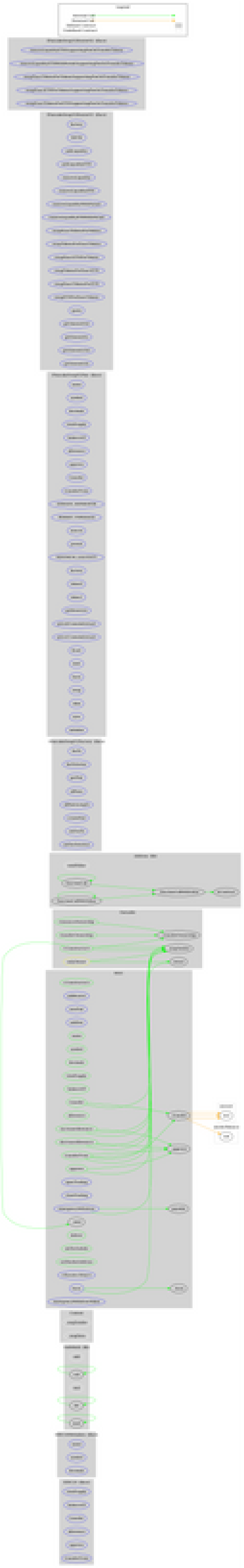
	<Constructor>	Public	✓	-
	owner	Public		-
	renounceOwnership	Public	✓	onlyOwner
	transferOwnership	Public	✓	onlyOwner
	_transferOwnership	Internal	✓	
Address	Library			
	isContract	Internal		
	sendValue	Internal	✓	
	functionCall	Internal	✓	
	functionCall	Internal	✓	
	functionCallWithValue	Internal	✓	
	functionCallWithValue	Internal	✓	
	_functionCallWithValue	Private	✓	
IPancakeSwapV2Factory	Interface			
	feeTo	External		-
	feeToSetter	External		-
	getPair	External		-
	allPairs	External		-
	allPairsLength	External		-
	createPair	External	✓	-
	setFeeTo	External	✓	-
	setFeeToSetter	External	✓	-
IPancakeSwapV2Pair	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	✓	-
IPancakeSwapV2Router01	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		-
IPancakeSwapV2Router02	Interface	IPancakeSwapV2Router01		
	removeLiquidityETHSupportingFeeOnTransferTokens	External	✓	-
	removeLiquidityETHWithPermitSupportingFeeOnTransferTokens	External	✓	-
	swapExactTokensForTokensSupportingFeeOnTransferTokens	External	✓	-
	swapExactETHForTokensSupportingFeeOnTransferTokens	External	Payable	-
	swapExactTokensForETHSupportingFeeOnTransferTokens	External	✓	-
MGC	Implementation	Context, IERC20, IERC20Metadata, Ownable		
	<Constructor>	Public	✓	-
	addRouter	External	✓	onlyOwner
	newPair	External	✓	onlyOwner
	addDex	External	✓	onlyOwner
	name	Public		-
	symbol	Public		-
	decimals	Public		-
	totalSupply	Public		-
	balanceOf	Public		-
	transfer	Public	✓	-
	allowance	Public		-
	approve	Public	✓	-

	transferFrom	Public	✓	-
	increaseAllowance	Public	✓	-
	decreaseAllowance	Public	✓	-
	openTrading	External	✓	onlyOwner
	closeTrading	External	✓	onlyOwner
	_transfer	Internal	✓	
	_mint	Internal	✓	
	_burn	Internal	✓	
	burn	External	✓	-
	_approve	Internal	✓	
	deliver	Public	✓	onlyOwner
	setExcludeds	Public	✓	onlyOwner
	setMarketAddress	Public	✓	onlyOwner
	<Receive Ether>	External	Payable	-
	emergencyWithdraw	External	✓	onlyOwner
	emergencyWithdrawToken	External	✓	onlyOwner

Contract Flow



Domain Info

Domain Name	metaswap.cx
Registry Domain ID	1903597-CoCCA
Creation Date	2022-01-07T09:48:10+00:00
Updated Date	2022-01-12T12:02:44+00:00
Registry Expiry Date	2023-01-07T09:48:10+00:00
Registrar WHOIS Server	
Registrar URL	
Registrar	CentralNic Ltd
Registrar IANA ID	

The domain has been created in 6 months 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 transferring tokens to the team's wallet. A multi-wallet signing pattern will provide security against potential hacks. Temporarily locking the contract or renouncing ownership will eliminate all the contract threats. There is also a limit of 1,3% on all transactions.

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