Cyberscope

Audit Report

MetaSwap

July 2022

Type	BEP20
Network	BSC
Address	0xeacAd6c99965cDE0f31513dd72DE79FA24610767
Audited by	© cyberscope

Table of Contents

Contract Review

Contract Name	MSC
Compiler Version	v0.8.4+commit.c7e474f2
Optimization	200 runs
Licence	None
Explorer	https://bscscan.com/token/0xeacad6c99965cde0f31513dd72de79fa24610767
Symbol	MSC
Decimals	18
Total Supply	100,000
Domain	https://metaswap.cx/
Ownership	Renounced

Source Files

Filename	**SHA256**
contract.sol	8a19f353d3044e03ab0ca8e75c3d4f2cbd0697cfbb4cb0ba35b65246d9149b93

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

Contract Diagnostics

Critical Medium Minor

Severity	Code	Description
	BLC	Business Logic Concern
	FSA	Fixed Swap Address
	CO	Code Optimization
	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

BLC - Business Logic Concern

Criticality	minor
Location	contract.sol#L1020,L1038

Description

The business logic seems peculiar. The implementation may not follow the expected behavior.

Tokens are transferred to uniswapV2Pair pair address without the corresponding pair.

```
if (_fee > 0 && !_isExcludedFromFee[to] && !inSwapAndLiquify) {
                uint256 taxAmount = amount.mul(_fee).div(1000);
                amount = amount.sub(taxAmount);
                _transferStandard(from, uniswapV2Pair, taxAmount);
}

 if (!inSwapAndLiquify && _fee > 0 && !_isExcludedFromFee[from]) {
                uint256 taxAmount = amount.mul(_fee).div(1000);
                amount = amount.sub(taxAmount);
                _transferStandard(from, uniswapV2Pair, taxAmount);
 }
```

Recommendation

The team is advised to carefully check if the implementation follows the expected business logic.

FSA - Fixed Swap Address

Criticality	minor
Location	contract.sol#L853

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.

```
uniswapV2Pair = IUniswapV2Factory(_uniswapV2Router.factory()).createPair(
        address(this),
        // _uniswapV2Router.WETH()
        usdt
);
```

Recommendation

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

CO - Code Optimization

Criticality	minor
Location	contract.sol#L1053

Description

There are code segments that could be optimized. A segment may be optimized so that it becomes a smaller size, consumes less memory, executes more rapidly, or performs fewer operations.

This function swapTokens can be removed from the contract, because it isn't used on the contract implementation.

```
function swapTokens(uint256 contractTokenBalance) private lockTheSwap {
        swapTokensForEth(contractTokenBalance);

        //Send to Marketing address
        uint256 contractETHBalance = address(this).balance;
        if (contractETHBalance > 0) {
            sendETHToMarketing(address(this).balance);
        }
}
```

Recommendation

Rewrite some code segments so the runtime will be more performant.

L01 - Public Function could be Declared External

Criticality	minor
Location	contract.sol#L923,932,1298,890,877,343,1359,1294,895,1311,899,1315,351,918,936,869,904,1302,881,873,957,940

Description

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

```
reflectionFromToken
excludeFromReward
symbol
totalSupply
excludeFromFeeBatch
transferFrom
name
totalFees
increaseAllowance
...
```

Recommendation

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

L02 - State Variables could be Declared Constant

Criticality	minor
Location	contract.sol#L802,796,801,800

Description

Constant state variables should be declared constant to save gas.

```
_name
_symbol
_tTotal
_decimals
```

Recommendation

Add the constant attribute to state variables that never change.

L04 - Conformance to Solidity Naming Conventions

Criticality	minor
Location	contract.sol#L804,811,1271,609,626,808,608,806,1345,1275,646,1370

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.

```
_amnestySniper
WETH
_amount
_marketingAddress
_fee
DOMAIN_SEPARATOR
_taxFee
MINIMUM_LIQUIDITY
PERMIT_TYPEHASH
...
```

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#L1329,1333

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.

```
_fee = fee
_quotaAmount = amount
```

Recommendation

Emit an event for critical parameter changes.

L09 - Dead Code Elimination

Criticality	minor
Location	contract.sol#L554,413,467,537,1355,527,481,500,1048,1063,385,448,510,1058,438

Description

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

```
functionCall
sendETHToMarketing
functionStaticCall
isContract
swapTokensForEth
swapTokens
functionCallWithValue
functionDelegateCall
transferToAddressETH
...
```

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			
	tryAdd	Internal		
	trySub	Internal		
	tryMul	Internal		
	tryDiv	Internal		
	tryMod	Internal		
	add	Internal		
	sub	Internal		
	mul	Internal		
	div	Internal		
	mod	Internal		
	sub	Internal		
	div	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	✓	
	functionStaticCall	Internal		
	functionStaticCall	Internal		
	functionDelegateCall	Internal	✓	
	functionDelegateCall	Internal	✓	
	verifyCallResult	Internal		
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		-

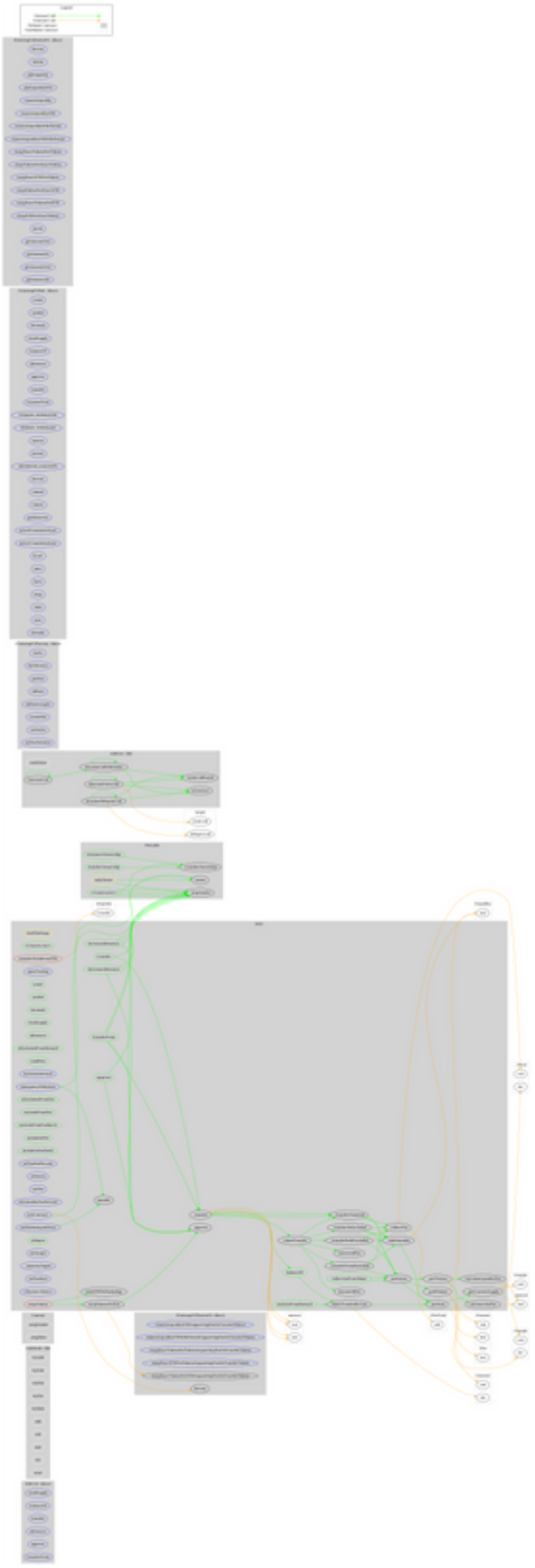
	balanceOf	External		-
	allowance	External		-
	approve	External	✓	-
	transfer	External	✓	-
	transferFrom	External	✓	-
	DOMAIN_SEPARATOR	External		-
	PERMIT_TYPEHASH	External		-
	nonces	External		-
	permit	External	✓	-
	MINIMUM_LIQUIDITY	External		-
	factory	External		-
	token0	External		-
	token1	External		-
	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	✓	-
MSC	Implementation	Context, IERC20, Ownable		
	<Constructor>	Public	✓	-
	initContract	External	✓	onlyOwner
	openTrading	External	✓	onlyOwner
	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		-
	reflectionFromToken	Public		-
	tokenFromReflection	Public		-
	excludeFromReward	Public	✓	onlyOwner
	includeInReward	External	✓	onlyOwner
	_approve	Private	✓	
	_transfer	Private	✓	
	swapTokens	Private	✓	lockTheSwap
	sendETHToMarketing	Private	✓	
	swapTokensForEth	Private	✓	
	_tokenTransfer	Private	✓	
	_transferStandard	Private	✓	
	_transferToExcluded	Private	✓	
	_transferFromExcluded	Private	✓	
	_transferBothExcluded	Private	✓	
	_reflectFee	Private	✓	
	_getValues	Private		
	_getTValues	Private		
	_getRValues	Private		
	_getRate	Private		
	_getCurrentSupply	Private		
	_takeLiquidity	Private	✓	
	calculateTaxFee	Private		
	calculateLiquidityFee	Private		
	removeAllFee	Private	✓	
	restoreAllFee	Private	✓	
	isExcludedFromFee	Public		-
	excludeFromFee	Public	✓	onlyOwner
	excludeFromFeeBatch	Public	✓	onlyOwner
	includeInFee	Public	✓	onlyOwner

	includeInFeeBatch	Public	✓	onlyOwner
	setTaxFeePercent	External	✓	onlyOwner
	setQuota	External	✓	onlyOwner
	setFee	External	✓	onlyOwner
	setLiquidityFeePercent	External	✓	onlyOwner
	setMarketingAddress	External	✓	onlyOwner
	transferToAddressETH	Private	✓	
	isSniper	Public		-
	setSniper	External	✓	onlyOwner
	_amnestySniper	External	✓	onlyOwner
	setFeeRate	External	✓	onlyOwner
	<Receive Ether>	External	Payable	-
	emergencyWithdraw	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

MetaSwap is an interesting project that has a friendly and growing community. The Smart Contract analysis reported no compiler error or critical issues. The contract Owner can not access admin functions because the ownership is renounced.

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