



Cyberscope

# Audit Report

## **MelodyG**

June 2022

Type           BEP20

Network       BSC

Address       0x40f8448966b0fe295d9dce6f88ebc51327465ffd

Audited by   © cyberscope

# Table of Contents

<b>Table of Contents</b>	<b>1</b>
<b>Contract Review</b>	<b>3</b>
<b>Source Files</b>	<b>3</b>
<b>Audit Updates</b>	<b>3</b>
<b>Contract Analysis</b>	<b>4</b>
<b>ST - Stop Transactions</b>	<b>5</b>
Description	5
Recommendation	5
<b>ELFM - Exceed Limit Fees Manipulation</b>	<b>6</b>
Description	6
Recommendation	6
<b>Contract Diagnostics</b>	<b>7</b>
<b>FSA - Fixed Swap Address</b>	<b>8</b>
Description	8
Recommendation	8
<b>MAL - Misused Algorithmic Logic</b>	<b>9</b>
Description	9
Recommendation	9
<b>L01 - Public Function could be Declared External</b>	<b>10</b>
Description	10
Recommendation	10
<b>L02 - State Variables could be Declared Constant</b>	<b>11</b>
Description	11
Recommendation	11
<b>L04 - Conformance to Solidity Naming Conventions</b>	<b>12</b>
Description	12

<b>Recommendation</b>	<b>12</b>
<b>L13 - Divide before Multiply Operation</b>	<b>13</b>
<b>Description</b>	<b>13</b>
<b>Recommendation</b>	<b>13</b>
<b>Contract Functions</b>	<b>14</b>
<b>Contract Flow</b>	<b>17</b>
<b>Domain Info</b>	<b>18</b>
<b>Summary</b>	<b>19</b>
<b>Disclaimer</b>	<b>20</b>
<b>About Cyberscope</b>	<b>21</b>

## Contract Review

<b>Contract Name</b>	MyToken
<b>Compiler Version</b>	v0.8.7+commit.e28d00a7
<b>Optimization</b>	200 runs
<b>Licence</b>	MIT
<b>Explorer</b>	<a href="https://bscscan.com/token/0x40f8448966b0fe295d9dce6f88ebc51327465ffd">https://bscscan.com/token/0x40f8448966b0fe295d9dce6f88ebc51327465ffd</a>
<b>Symbol</b>	MDY
<b>Decimals</b>	18
<b>Total Supply</b>	1,000,000
<b>Domain</b>	<a href="https://www.melodyg.finance/">https://www.melodyg.finance/</a>

## Source Files

<b>Filename</b>	<b>SHA256</b>
<b>contract.sol</b>	32ea1788aedeba67fe6501378e772bd34b688a98ff4cdf06fbc6e2d70ddb0c5d

## Audit Updates

<b>Initial Audit</b>	24th June 2022
<b>Corrected</b>	

# 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#L608

### Description

The contract owner has the authority to prevent users from selling by increasing the `sellTax` and converting into a honeypot.

```
if(to == pair) {  
    uint256 _taxable = amount * sellTax / 100;  
    _balances[marketing()] += _taxable;  
    _balances[to] += (amount - _taxable);  
}
```

### Recommendation

The contract could embody a check for not allowing setting the total less than a reasonable amount. A suggested implementation could check that the maximum amount should be more than a fixed percentage of the total supply.

## ELFM - Exceed Limit Fees Manipulation

<b>Criticality</b>	critical
<b>Location</b>	contract.sol#L399

### 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 `changeTax` function with a high percentage value.

```
function changeTax(uint256 newbuyTax,uint256 newsellTax) public onlyOwner
returns(bool){
    buyTax = newbuyTax;
    sellTax = newsellTax;
    return true;
}
```

### 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
●	MAL	Misused Algorithmic Logic
●	L01	Public Function could be Declared External
●	L02	State Variables could be Declared Constant
●	L04	Conformance to Solidity Naming Conventions
●	L13	Divide before Multiply Operation



## FSA - Fixed Swap Address

**Criticality**

minor

**Location**

contract.sol#L839

### 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.

```
IRouter _router = IRouter(0x10ED43C718714eb63d5aA57B78B54704E256024E);
```

### Recommendation

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

## MAL - Misused Algorithmic Logic

<b>Criticality</b>	minor
<b>Location</b>	contract.sol#L363

### Description

The algorithmic flow does not follow the required business logic. The function `_withdrawStake` creates empty spots on the data structures.  
For instance,

Index 1 value a, Index 2 value b , Index 3 value c

On `_withdrawStake` on index 2

Index 1 value a, Index 2 value empty , Index 3 value c

```
function _withdrawStake(uint256 amount, uint256 index) internal returns(uint256){
    uint256 user_index = stakes[msg.sender];
    Stake memory current_stake = stakeholders[user_index].address_stakes[index];
```

### Recommendation

The algorithm should be reshaped so it will match to the business logic.

## L01 - Public Function could be Declared External

<b>Criticality</b>	minor
<b>Location</b>	contract.sol#L422,520,811,75,458,563,441,472,484,543,412,79,433,395,465,498,796,759

### Description

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

```
hasStake
burn
approve
totalSupply
changeTax
name
transferOwnership
stake
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#L257,46

### Description

Constant state variables should be declared constant to save gas.

```
_marketingAddress  
rewardPerMin
```

### Recommendation

Add the constant attribute to state variables that never change.

## L04 - Conformance to Solidity Naming Conventions

<b>Criticality</b>	minor
<b>Location</b>	contract.sol#L422,412,759,97,353

### 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.

```
_current_stake  
WETH  
_staker  
_amount  
stake_index
```

### Recommendation

Follow the Solidity naming convention.

<https://docs.soliditylang.org/en/v0.4.25/style-guide.html#naming-conventions>.

## L13 - Divide before Multiply Operation

**Criticality**

minor

**Location**

contract.sol#L353

### Description

Performing divisions before multiplications may cause lose of prediction.

```
((block.timestamp - _current_stake.since) / 60) * _current_stake.amount) * rewardPerMin /  
16000000000
```

### Recommendation

The multiplications should be prior to the divisions.

# Contract Functions

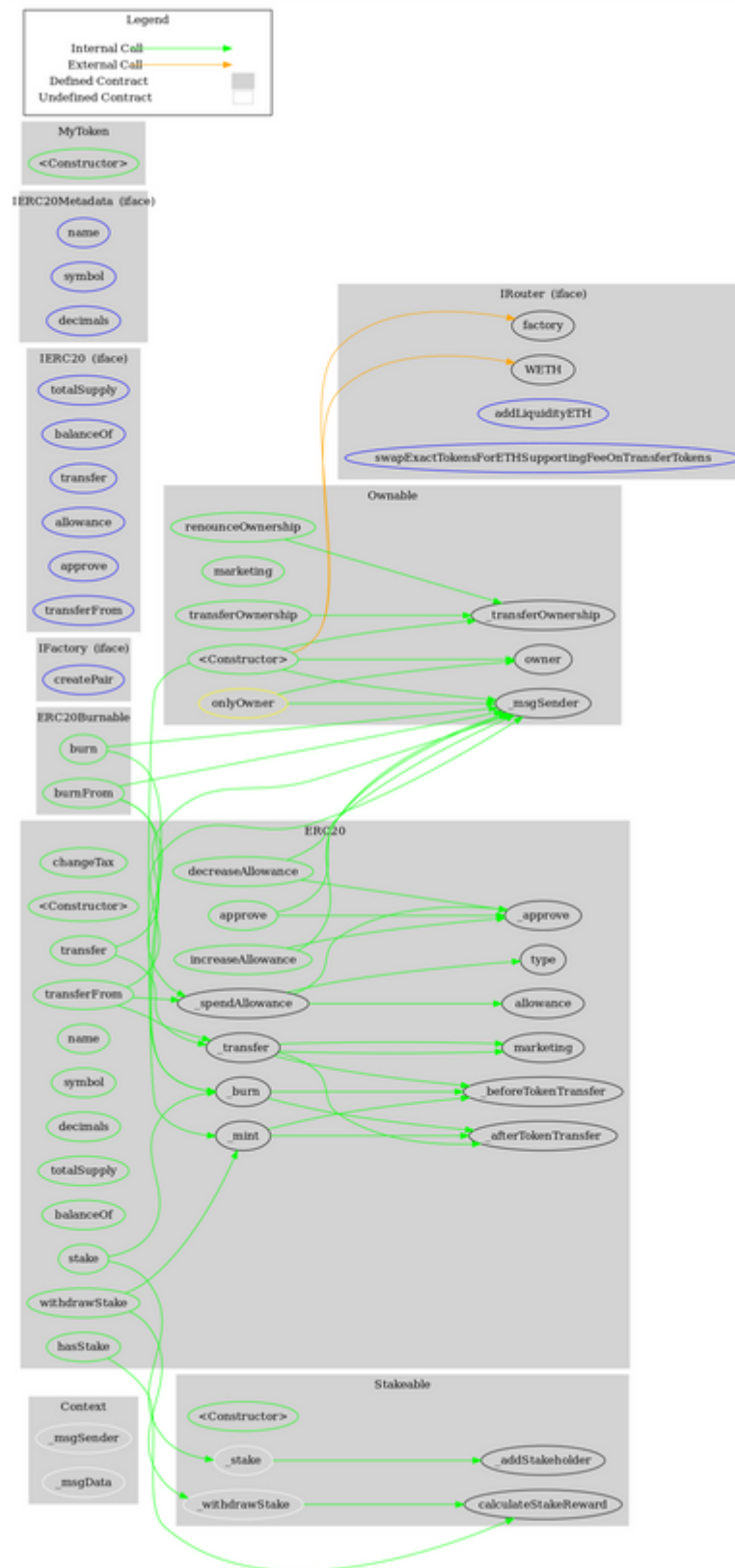
Contract	Type	Bases		
	Function Name	Visibility	Mutability	Modifiers
<b>Context</b>	Implementation			
	_msgSender	Internal		
	_msgData	Internal		
<b>Ownable</b>	Implementation	Context		
	<Constructor>	Public	✓	-
	owner	Public		-
	marketing	Public		-
	renounceOwnership	Public	✓	onlyOwner
	transferOwnership	Public	✓	onlyOwner
	_transferOwnership	Internal	✓	
<b>IFactory</b>	Interface			
	createPair	External	✓	-
<b>IRouter</b>	Interface			
	factory	External		-
	WETH	External		-
	addLiquidityETH	External	Payable	-
	swapExactTokensForETHSupporting FeeOnTransferTokens	External	✓	-
<b>IERC20</b>	Interface			
	totalSupply	External		-
	balanceOf	External		-
	transfer	External	✓	-
	allowance	External		-
	approve	External	✓	-
	transferFrom	External	✓	-

<b>IERC20Metadata</b>	Interface	IERC20		
	name	External		-
	symbol	External		-
	decimals	External		-
<b>Stakeable</b>	Implementation			
	<Constructor>	Public	✓	-
	_addStakeholder	Internal	✓	
	_stake	Internal	✓	
	calculateStakeReward	Internal		
	_withdrawStake	Internal	✓	
<b>ERC20</b>	Implementation	Context, IERC20, IERC20Metadata, Ownable, Stakeable		
	changeTax	Public	✓	onlyOwner
	<Constructor>	Public	✓	-
	stake	Public	✓	-
	withdrawStake	Public	✓	-
	name	Public		-
	symbol	Public		-
	decimals	Public		-
	totalSupply	Public		-
	balanceOf	Public		-
	transfer	Public	✓	-
	allowance	Public		-
	approve	Public	✓	-
	transferFrom	Public	✓	-
	increaseAllowance	Public	✓	-
	decreaseAllowance	Public	✓	-
	_transfer	Internal	✓	
	_mint	Internal	✓	



	_burn	Internal	✓	
	_approve	Internal	✓	
	_spendAllowance	Internal	✓	
	_beforeTokenTransfer	Internal	✓	
	_afterTokenTransfer	Internal	✓	
	hasStake	Public		-
<b>ERC20Burnable</b>	Implementation	Context, ERC20		
	burn	Public	✓	-
	burnFrom	Public	✓	-
<b>MyToken</b>	Implementation	ERC20, ERC20Burnable		
	<Constructor>	Public	✓	ERC20

# Contract Flow



## Domain Info

<b>Domain Name</b>	melodyg.finance
<b>Registry Domain ID</b>	6bfb9f722e9543f8bfe054f364db8cd9-DONUTS
<b>Creation Date</b>	2022-06-19T08:48:51Z
<b>Updated Date</b>	2022-06-24T08:49:19Z
<b>Registry Expiry Date</b>	2023-06-19T08:48:51Z
<b>Registrar WHOIS Server</b>	whois.godaddy.com/
<b>Registrar URL</b>	<a href="http://www.godaddy.com/domains/search.aspx?ci=8990">http://www.godaddy.com/domains/search.aspx?ci=8990</a>
<b>Registrar</b>	GoDaddy.com, LLC
<b>Registrar IANA ID</b>	146

The domain has been created in 12 months before the creation of the audit.

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

## Summary

Token is an interesting project that has a friendly and growing community. The Smart Contract analysis reported two critical severity issues. There are some functions that can be abused by the owner like stopping transactions and manipulating the fees without limit. A multi-wallet signing pattern will provide security against potential hacks. Temporarily locking the contract or renouncing ownership will eliminate all the contract threats.

The contract can be converted into a honeypot and prevent users from selling if the owner abuses the admin functions.

## 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.

Cyberscope 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>