



RUGFREECOINS

MUZI<OIN

# MuziKoin Token

RugfreeCoins Verified on November 27th, 2023

# Overview

- ✓ No mint function found, the owner cannot mint tokens after initial deployment.
- ✓ The owner can't set a max transaction limit
- ✓ The owner can't pause trading once it's enabled
- ✓ The owner can't change fees over 10%.
- ✓ The owner can't blacklist wallets.
- ✓ The owner can't set a max wallet limit
- ✓ The owner can't claim the contract's balance of its own token.

## ! HIGH SEVERITY ISSUES

**In the swap, the marketing wallet receives BNB. If the owner changes the marketing wallet and it is unable to receive BNB, the swap will fail, and sales will also be halted.**

# Contents

Overview.....	2
Contents.....	3
Audit details.....	4
Disclaimer.....	5
Background.....	6
Tokenomics.....	7
Target market and the concept.....	8
Potential to grow with score points.....	9
Contract details.....	10
Contract code function details.....	11
Contract description table.....	12
Inheritance Hierarchy.....	17
Security issue checking status.....	18
Owner privileges.....	19
Audit conclusion.....	22

## Audit details



**Audited project**  
MuziKoin Token



**Contract Address**  
0xCe379bAD99f287E1eacB1bF80F3670011D2B2981



**Client contact**  
MuziKoin Token Team



**Blockchain**  
Binance Smart chain



**Project website**  
[www.muzikoin.com](http://www.muzikoin.com)

# Disclaimer

This is a limited report on our findings based on our analysis, in accordance with good industry practice as at the date of this report, in relation to cybersecurity vulnerabilities and issues in the framework and algorithms based on smart contracts, the details of which are set out in this report. In order to get a full view of our analysis, it is crucial for you to read the full report. While we have done our best in conducting our analysis and producing this report, it is important to note that you should not rely on this report and cannot claim against us on the basis of what it says or doesn't say, or how we produced it, and it is important for you to conduct your own independent investigations before making any decisions. We go into more detail on this in the disclaimer below – **please make sure to read it in full.**

## ❗ DISCLAIMER

By reading this report or any part of it, you agree to the terms of this disclaimer. If you do not agree to the terms, then please immediately cease reading this report, and delete and destroy any and all copies of this report downloaded and/or printed by you. **This report is provided for information purposes only and on a non-reliance basis, and does not constitute investment advice.** No one shall have any right to rely on the report or its contents, and **RugfreeCoins and its affiliates (including holding companies, shareholders, subsidiaries, employees, directors, officers and other representatives) (RugfreeCoins) owe no duty of care towards you or any other person**, nor does RugfreeCoins make any warranty or representation to any person on the accuracy or completeness of the report.

The report is provided "as is", without any conditions, warranties or other terms of any kind except as set out in this disclaimer, and RugfreeCoins hereby excludes all representations, warranties, conditions and other terms (including, without limitation, the warranties implied by law of satisfactory quality, fitness for purpose and the use of reasonable care and skill) which, but for this clause, might have effect in relation to the report. Except and only to the extent that it is prohibited by law, RugfreeCoins hereby excludes all liability and responsibility, and neither you nor any other person shall have any claim against RugfreeCoins, for any amount or kind of loss or damage that may result to you or any other person (including without limitation, any direct, indirect, special, punitive, consequential or pure economic loss or damages, or any loss of income, profits, goodwill, data, contracts, use of money, or business interruption, and whether in delict, tort (including without limitation negligence), contract, breach of statutory duty, misrepresentation (whether innocent or negligent) or otherwise under any claim of any nature whatsoever in any jurisdiction) in any way arising from or connected with this report and the use, inability to use or the results of use of this report, and any reliance on this report.

The analysis of the security is purely based on the smart contracts alone. No applications or operations were reviewed for security. No product code has been reviewed.

# Background

**RugfreeCoins** was commissioned by the **MuziKoin Token Team** to perform an audit of the smart contract.

<https://bscscan.com/token/0xce379bad99f287e1eacb1bf80f3670011d2b2981>

This audit focuses on verifying that the smart contract is secure, resilient, and working according to the specifications.

The information in this report should be used to understand the risk exposure of the smart contract, project feasibility, and long-term sustainability, and as a guide to improving the smart contract's security posture by remediating the identified issues.

# Tokenomics

## ▲ 1% tax when buying

1% of trade goes to the marketing wallet in BNB

## ▲ 2% tax when selling

2% of trade goes to the marketing wallet in BNB












---

## Target market and the concept

- ▶ Anyone who's interested in the Crypto space with long-term investment plans.
- ▶ Anyone who's ready to earn a passive income by holding tokens.
- ▶ Anyone who's interested in trading tokens.
- ▶ Anyone who's interested in taking part in the MuziKoin token ecosystem.
- ▶ Anyone who's interested in taking part in the future plans of MuziKoin Token.
- ▶ Anyone who's interested in making financial transactions with any other party using MuziKoin Token as the currency.



## Potential to grow with score points

 Project efficiency	8 / 10
 Project uniqueness	7 / 10
 Information quality	8 / 10
 Service quality	8 / 10
 System quality	8 / 10
 Impact on the community	8 / 10
 Impact on the business	9 / 10
 Preparing for the future	8 / 10
 Smart contract security	9 / 10
 Smart contract functionality assessment	9 / 10
 <b>Total Score</b>	<b>8.2/ 10</b>

# Contract details

Token contract details for 27th of November 2023































Contract name	MuziKoin
Contract address	0xCe379bAD99f287E1eacB1bF80F3670011D2B2981
Token supply	900,000,000
Token ticker	MZK
Decimals	18
Token holders	1
Transaction count	1
Contract deployer address	0xf5732b73Ee5A73B08AD6a4F5bA7E15D800dC6afe
Contract's current owner address	0x8781d5534F657e8b2f5Ab858fFFC749ef2c15471
Marketing wallet	0x7FDE78dA656E68B6fBB3593B0683fa71a43D0854
CX wallet	0x8781d5534F657e8b2f5Ab858fFFC749ef2c15471

# Contract code function details

Nº	Category	Item	Result
1	Coding conventions	ERC20 Token standards	PASS ▾
		Compile errors	PASS ▾
		Compiler version security	PASS ▾
		Visibility specifiers	PASS ▾
		Gas consumption	PASS ▾
		SafeMath features	PASS ▾
		Fallback usage	PASS ▾
		tx.origin usage	PASS ▾
		Deprecated items	PASS ▾
		Redundant code	PASS ▾
2	Function call audit	Overriding variables	PASS ▾
		Authorization of function call	PASS ▾
		Low level function (call/delegate call) security	PASS ▾
		Returned value security	PASS ▾
3	Business security & centralisation	Self destruct function security	PASS ▾
		Access control of owners	HIGH ▾
		Business logics	PASS ▾
4	Integer overflow/underflow	Business implementation	PASS ▾
5	Reentrancy		PASS ▾
6	Exceptional reachable state		PASS ▾
7	Transaction ordering dependence		PASS ▾
8	Block properties dependence		PASS ▾
9	Pseudo random number generator (PRNG)		PASS ▾
10	DoS (Denial of Service)		PASS ▾
11	Token vesting implementation		PASS ▾
12	Fake deposit		PASS ▾
13	Event security		PASS ▾













# Contract description table


The below table represents the summary of the contracts and methods in the token contract. We scanned the whole contract and listed down all the Interfaces, functions, and implementations with their visibility and mutability.

Contract	Type	Bases		
L	Function Name	Visibility	Mutability	Modifiers
Context	Implementation			
L	_msgSender	Internal 		
L	_msgData	Internal 		
IERC20	Interface			
L	totalSupply	External 		NO 
L	balanceOf	External 		NO 
L	transfer	External 		NO 
L	allowance	External 		NO 
L	approve	External 		NO 
L	transferFrom	External 		NO 
Address	Library			
L	isContract	Internal 		
L	sendValue	Internal 		
L	functionCall	Internal 		
L	functionCall	Internal 		
L	functionCallWithValue	Internal 		
L	functionCallWithValue	Internal 		
L	functionStaticCall	Internal 		
L	functionStaticCall	Internal 		



L	functionDelegateCall	Internal 🔒	🛑	
L	functionDelegateCall	Internal 🔒	🛑	
L	verifyCallResultFromTarget	Internal 🔒		
L	verifyCallResult	Internal 🔒		
L	_revert	Private 🔒🔑		
<b>Ownable</b>	<b>Implementation</b>	<b>Context</b>		
L		Public !	🛑	NO !
L	owner	Public !		NO !
L	_checkOwner	Internal 🔒		
L	renounceOwnership	Public !	🛑	onlyOwner
L	transferOwnership	Public !	🛑	onlyOwner
L	_transferOwnership	Internal 🔒	🛑	
<b>IUniswapV2 Factory</b>	<b>Interface</b>			
L	feeTo	External !		NO !
L	feeToSetter	External !		NO !
L	getPair	External !		NO !
L	allPairs	External !		NO !
L	allPairsLength	External !		NO !
L	createPair	External !	🛑	NO !
L	setFeeTo	External !	🛑	NO !
L	setFeeToSetter	External !	🛑	NO !
<b>IUniswapV2Pair</b>	<b>Interface</b>			
L	name	External !		NO !
L	symbol	External !		NO !
L	decimals	External !		NO !
L	totalSupply	External !		NO !

L	balanceOf	External !		NO !
L	allowance	External !		NO !
L	approve	External !		NO !
L	transfer	External !		NO !
L	transferFrom	External !		NO !
L	DOMAIN_SEPARATOR	External !		NO !
L	PERMIT_TYPEHASH	External !		NO !
L	nonces	External !		NO !
L	permit	External !		NO !
L	MINIMUM_LIQUIDITY	External !		NO !
L	factory	External !		NO !
L	token0	External !		NO !
L	token1	External !		NO !
L	getReserves	External !		NO !
L	price0CumulativeLast	External !		NO !
L	price1CumulativeLast	External !		NO !
L	kLast	External !		NO !
L	burn	External !		NO !
L	swap	External !		NO !
L	skim	External !		NO !
L	sync	External !		NO !
L	initialize	External !		NO !
<b>IUniswapV2 Router01</b>	<b>Interface</b>			
L	factory	External !		NO !
L	WETH	External !		NO !
L	addLiquidity	External !		NO !
L	addLiquidityETH	External !		NO !
L	removeLiquidity	External !		NO !

L	removeLiquidityETH	External !		NO !
L	removeLiquidityWithPermit	External !		NO !
L	removeLiquidityETHWithPermit	External !		NO !
L	swapExactTokensForTokens	External !		NO !
L	swapTokensForExactTokens	External !		NO !
L	swapExactETHForTokens	External !		NO !
L	swapTokensForExactETH	External !		NO !
L	swapExactTokensForETH	External !		NO !
L	swapETHForExactTokens	External !		NO !
L	quote	External !		NO !
L	getAmountOut	External !		NO !
L	getAmountIn	External !		NO !
L	getAmountsOut	External !		NO !
L	getAmountsIn	External !		NO !
<b>IUniswapV2 Router02</b>	<b>Interface</b>	<b>IUniswapV2 Router01</b>		
L	removeLiquidityETHSupportingFeeOnTransferTokens	External !		NO !
L	removeLiquidityETHWithPermitSupportingFeeOnTransferTokens	External !		NO !
L	swapExactTokensForTokensSupportingFeeOnTransferTokens	External !		NO !
L	swapExactETHForTokensSupportingFeeOnTransferTokens	External !		NO !
L	swapExactTokensForETHSupportingFeeOnTransferTokens	External !		NO !
<b>MuziKoin</b>	<b>Implementation</b>	<b>Context, IERC20, Ownable</b>		
L		Public !		NO !
L	totalSupply	Public !		NO !
L	balanceOf	Public !		NO !

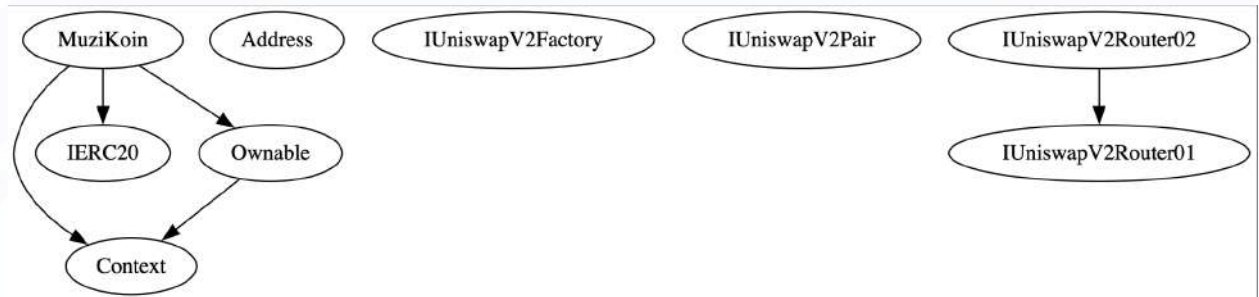
L	transfer	Public !		NO !
L	allowance	Public !		NO !
L	approve	Public !		NO !
L	transferFrom	Public !		NO !
L	increaseAllowance	Public !		NO !
L	decreaseAllowance	Public !		NO !
L	_approve	Private 		
L	_transfer	Private 		
L	swapAndLiquify	Public !		lockTheSwap
L	swapTokensForEth	Private 		
L	_tokenTransfer	Private 		
L	isExcludedFromFees	External !		NO !
L	setExcludeFromFees	External !		onlyOwner
L	setTokensToSwap	External !		onlyOwner
L	setSwapAndLiquifyEnabled	External !		onlyOwner
L	setMarketingWallet	External !		onlyOwner
L	setBuyFee	External !		onlyOwner
L	setSellFee	External !		onlyOwner
L	transferToAddressETH	Private 		
L		External !		NO !
L	swapETHForTokens	Private 		
L	recoverETHfromContract	External !		onlyOwner
L	recoverTokensFromContract	External !		onlyOwner

### Legend

Symbol	Meaning
	Function can modify state
	Function is payable



# Inheritance Hierarchy



# Security issue checking status

## ❖ High severity issues

**In the swap, the marketing wallet receives BNB. If the owner changes the marketing wallet and it is unable to receive BNB, the swap will fail, and sales will also be halted.**

## ❖ Medium severity issues

No medium severity issues found

## ❖ Low severity issues

No low-severity issues found

# Owner privileges

- ❖ The owner can include/exclude wallets from fees

```
function setExcludeFromFees(address account, bool status) external onlyOwner {
    require(!_isExcludedFromFees[account] != status, "The wallet is already have that status.");
    _isExcludedFromFees[account] = status;
    emit ExcludeStatus(account, status);
}
```

- ❖ The owner can change the swap point

```
function setTokensToSwap(
    uint256 _minimumTokensBeforeSwap
) external onlyOwner {
    require(
        _minimumTokensBeforeSwap >= 100 ether,
        "You need to enter more than 100 tokens."
    );
    minimumTokensBeforeSwap = _minimumTokensBeforeSwap;
    emit UpdateTokensToSwap(minimumTokensBeforeSwap);
}
```

- ❖ The owner can enable/disable swapping

```
function setSwapAndLiquifyEnabled(bool _enabled) external onlyOwner {
    require(swapAndLiquifyEnabled != _enabled, "Value already set");
    swapAndLiquifyEnabled = _enabled;
    emit SwapAndLiquifyEnabledUpdated(_enabled);
}
```

- ❖ The owner can change the marketing wallet address

```
function setMarketingWallet(address _marketingWallet) external onlyOwner {
    require(_marketingWallet != address(0), "setMarketingWallet: ZERO");
    marketingWallet = payable(_marketingWallet);
    emit UpdateMarketingWallet(marketingWallet, true);
}
```

- ❖ Owner can change buy and sell fees, each fee maximum of up to 5%

```
//set a new Buy Fee %.
function setBuyFee(uint256 _buyFee) external onlyOwner {
    require(_buyFee <= 5, "Buy Fee cannot be more than 5%");
    buyFee = _buyFee;
    emit UpdateBuyFee(buyFee);
}
//set a new Sell Fee %
function setSellFee(uint256 _sellFee) external onlyOwner {
    require(_sellFee <= 5, "Sell Fee cannot be more than 5%");
    sellFee = _sellFee;
    emit UpdateSellFee(sellFee);
}
```

- ❖ The owner can get BNB from the contract

```
function recoverETHfromContract() external onlyOwner {  
    uint ethBalance = address(this).balance;  
    (bool succ, ) = payable(marketingWallet).call{value: ethBalance}("");  
    require(succ, "Transfer failed");  
    emit recoveredETH(ethBalance);  
}
```

- ❖ The owner can get any BEP20 tokens from the contract (can not get native tokens)

```
function recoverTokensFromContract(  
    address _tokenAddress,  
    uint256 _amount  
) external onlyOwner {  
    require(  
        _tokenAddress != address(this),  
        "Owner can't claim contract's balance of its own tokens"  
    );  
    bool succ = IERC20(_tokenAddress).transfer(marketingWallet, _amount);  
    require(succ, "Transfer failed");  
    emit recoveredTokens(_amount);  
}
```

# Audit conclusion

RugFreeCoins team has performed in-depth testing, line-by-line manual code review, and automated audit of the smart contract. The smart contract was analyzed mainly for common smart contract vulnerabilities, exploits, manipulations, and hacks. According to the smart contract audit.

Smart contract functional Status:	PASS ▾
Smart contract security Status:	HIGH ISSUES ▾
Number of risk issues:	0
Solidity code functional issue level:	PASS ▾
Number of owner privileges:	7
Centralization risk correlated to the active owner:	HIGH ▾
Smart contract active ownership:	ACTIVE ▾