



RUGFREECOINS



ONE Token

RugfreeCoins Verified on October 30th, 2023

Overview

- ✗ Mint function found, the owner can mint tokens after initial deployment.
- ✗ The owner can set a max transaction limit
- ✓ The owner can't pause trading once it's enabled
- ✓ The owner can't change fees over 20%.
- ✗ The owner can 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

The owner can set the max sell transaction amount to 0, if set to 0, no one will be able to sell tokens

```
function updateMaxSellTransactionAmount(  
    uint256 newAmount  
) external onlyOwner {  
    emit MaxSellTransactionAmountUpdated(  
        maxSellTransactionAmount,  
        newAmount  
    );  
    maxSellTransactionAmount = newAmount * 10 ** 18;  
}
```

Owner can mint any amount of tokens after the launch

```
function mint(address _to, uint256 _tokenAmount) public onlyOwner {  
    _mint(_to, _tokenAmount * (10 ** 18));  
}
```

Owner can blacklist any wallets from the contract if the owner blacklists pair address no one will be able to buy and sell

```
function setBlacklistAddress(  
    address account,  
    bool value  
) external onlyOwner {  
    _isBlacklisted[account] = value;  
    emit BlacklistAddressUpdated(account, value);  
}
```

Contents

Overview.....	2
Contents.....	4
Audit details.....	5
Disclaimer.....	6
Background.....	7
Tokenomics.....	8
Target market and the concept.....	9
Potential to grow with score points.....	10
Contract details.....	11
Contract code function details.....	12
Contract description table.....	13
Inheritance Hierarchy.....	17
Security issue checking status.....	18
Owner privileges.....	20
Audit conclusion.....	24

Audit details



Audited project

ONE Token



Contract Address

0xA75fC235FB39E8b5862Af643B8F30fEAAA0557C8



Client contact

ONE Token Team



Blockchain

Binance Smart chain



Project website

<https://oneoportunity.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 **ONE Token Team** to perform an audit of the smart contract.

<https://bscscan.com/address/0xA75fC235FB39E8b5862Af643B8F30fEAAA0557C8>

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

▲ 8% tax when buying & selling

8% of trade goes to the marketing wallet in USDT












▲ 8% tax when buying & selling

2% of trade goes to a wallet in tokens

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 ONE token ecosystem.
- ▶ Anyone who's interested in taking part in the future plans of ONE Token.
- ▶ Anyone who's interested in making financial transactions with any other party using ONE Token as the currency.

Potential to grow with score points

 Project efficiency	8 / 10
 Project uniqueness	9 / 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	7 / 10
 Smart contract functionality assessment	8 / 10
 Total Score	8.1 / 10

Contract details

Token contract details for 30th of October 2023

Contract name	ONE
Contract address	0xA75fC235FB39E8b5862Af643B8F30fEAAA0557C8
Token supply	8,000,000,000
Token ticker	ONE
Decimals	18
Token holders	9
Transaction count	25
Contract deployer address	0x0353b3Aeb5628A9C1AD973982CC3A5882598a5C5
Contract's current owner address	0xB7b52Eb105475615dd790b09AA60846caF72144e
Marketing wallet	0x977fa7611AAC600f3dAB784e2F07c909730c6E70
Transfer fee wallet	0xb90282a32B7196Ec83323304dfcfE93B9d1Ec2Ad








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	LOW ▾
		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
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	sendValue	Internal 🔒	●	
Context	Implementation			
L	_msgSender	Internal 🔒		
L	_msgData	Internal 🔒		
Ownable	Implementation	Context		
L		Public !	●	NO !
L	owner	Public !		NO !
L	renounceOwnership	Public !	●	onlyOwner
L	transferOwnership	Public !	●	onlyOwner
L	_setOwner	Private 🔒🔑	●	

IFactory	Interface			
L	createPair	External !		NO !
IPair	Interface			
L	getReserves	External !		NO !
L	token0	External !		NO !
IRouter	Interface			
L	factory	External !		NO !
L	WETH	External !		NO !
L	addLiquidityETH	External !		NO !
L	swapExactTokensForTokensSupportingFeeOnTransferTokens	External !		NO !
L	swapExactETHForTokensSupportingFeeOnTransferTokens	External !		NO !
L	swapExactTokensForETHSupportingFeeOnTransferTokens	External !		NO !
IERC20 Metadata	Interface	IERC20		
L	name	External !		NO !
L	symbol	External !		NO !
L	decimals	External !		NO !
Safe MathInt	Library			
L	mul	Internal 		
L	div	Internal 		
L	sub	Internal 		
L	add	Internal 		
L	abs	Internal 		
L	toUint256Safe	Internal 		

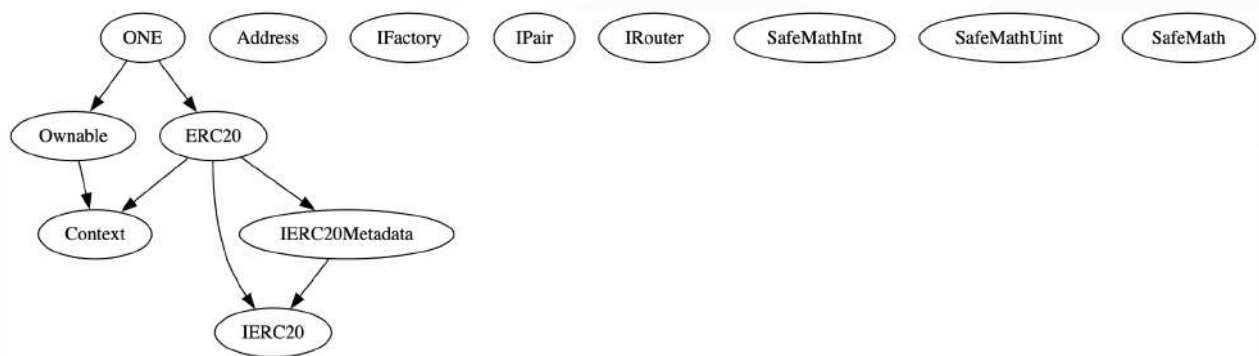
SafeMath Uint	Library			
L	toInt256Safe	Internal 		
SafeMath	Library			
L	add	Internal 		
L	sub	Internal 		
L	sub	Internal 		
L	mul	Internal 		
L	div	Internal 		
L	div	Internal 		
L	mod	Internal 		
L	mod	Internal 		
ERC20	Implementation	Context, IERC20, IERC20 Metadata		
L		Public 		NO 
L	name	Public 		NO 
L	symbol	Public 		NO 
L	decimals	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	_transfer	Internal 🔒	🔴	
L	_mint	Internal 🔒	🔴	
L	_burn	Internal 🔒	🔴	
L	_approve	Internal 🔒	🔴	
L	_beforeTokenTransfer	Internal 🔒	🔴	
ONE	Implementation	ERC20, Ownable		
L		Public !	🔴	ERC20
L		External !	🟢	NO !
L	updateSwapTokensAtAmount	External !	🔴	onlyOwner
L	updateMaxBuyTransactionAmount	External !	🔴	onlyOwner
L	updateMaxSellTransactionAmount	External !	🔴	onlyOwner
L	mint	Public !	🔴	onlyOwner
L	setBlacklistAddress	External !	🔴	onlyOwner
L	setMarketingWallet	External !	🔴	onlyOwner
L	setTransferFeeWallet	External !	🔴	onlyOwner
L	setBuyFee	External !	🔴	onlyOwner
L	setSellFee	External !	🔴	onlyOwner
L	setTransferFee	External !	🔴	onlyOwner
L	excludeFromFees	Public !	🔴	onlyOwner
L	excludeMultipleAccountsFromFees	External !	🔴	onlyOwner
L	setAutomatedMarketMakerPair	External !	🔴	onlyOwner
L	isExcludedFromFees	Public !		NO !
L	_setAutomatedMarketMakerPair	Private 🔒	🔴	
L	_transfer	Internal 🔒	🔴	
L	swapTokensForMarketingUSDT	Private 🔒	🔴	

Legend

Symbol	Meaning
	Function can modify state
	Function is payable

Inheritance Hierarchy



Security issue checking status

❖ High severity issues

Owner can set the max sell transaction amount to 0, if set to 0, no one will be able to sell tokens

```
function updateMaxSellTransactionAmount(
    uint256 newAmount
) external onlyOwner {
    emit MaxSellTransactionAmountUpdated(
        maxSellTransactionAmount,
        newAmount
    );
    maxSellTransactionAmount = newAmount * 10 ** 18;
}
```

Owner can mint any amount of tokens after the launch

```
function mint(address _to, uint256 _tokenAmount) public onlyOwner {
    _mint(_to, _tokenAmount * (10 ** 18));
}
```

Owner can blacklist any wallets from the contract if the owner blacklists pair address no one will able to buy and sell

```
function setBlacklistAddress(  
    address account,  
    bool value  
) external onlyOwner {  
    _isBlacklisted[account] = value;  
    emit BlacklistAddressUpdated(account, value);  
}
```

❖ Medium severity issues

No medium severity issues found

❖ Low severity issues

When excluding multiple wallets from fees, the owner can input any number of wallets without limit. If the owner enters a large number of wallets array this function may fail due to out of gas limit.

```
function excludeMultipleAccountsFromFees(  
    address[] calldata accounts,  
    bool excluded  
) external onlyOwner {  
    for (uint256 i = 0; i < accounts.length; i++) {  
        _isExcludedFromFees[accounts[i]] = excluded;  
    }  
  
    emit ExcludeMultipleAccountsFromFees(accounts, excluded);  
}
```

Owner privileges

- ❖ Owner can change swap limit

```
function updateSwapTokensAtAmount(uint256 newAmount) external onlyOwner {  
    emit SwapTokensAtAmountUpdated(swapTokensAtAmount, newAmount);  
    swapTokensAtAmount = newAmount * 10 ** 18;  
}
```

- ❖ Owner can set max buy amount without any limit

```
function updateMaxBuyTransactionAmount(  
    uint256 newAmount  
) external onlyOwner {  
    emit MaxBuyTransactionAmountUpdated(maxBuyTransactionAmount, newAmount);  
    maxBuyTransactionAmount = newAmount * 10 ** 18;  
}
```

- ❖ Owner can change max sell tx amount without any limit

```
function updateMaxSellTransactionAmount(  
    uint256 newAmount  
) external onlyOwner {  
    emit MaxSellTransactionAmountUpdated(  
        maxSellTransactionAmount,  
        newAmount  
    );  
    maxSellTransactionAmount = newAmount * 10 ** 18;  
}
```

- ❖ Owner can mint number of token after deployment

```
function mint(address _to, uint256 _tokenAmount) public onlyOwner {  
    _mint(_to, _tokenAmount * (10 ** 18));  
}
```

- ❖ Owner can blacklist any wallet/contract from the token

```
function setBlacklistAddress(  
    address account,  
    bool value  
) external onlyOwner {  
    _isBlacklisted[account] = value;  
    emit BlacklistAddressUpdated(account, value);  
}
```

- ❖ Owner can change marketing wallet address

```
function setMarketingWallet(address payable _newWallet) external onlyOwner {  
    emit MarketingWalletUpdated(MarketingWallet, _newWallet);  
    MarketingWallet = _newWallet;  
  
    _isExcludedFromFees[_newWallet] = true;  
    emit ExcludeFromFees(_newWallet, true);  
}
```

- ❖ Owner can change transfer fee receiver wallet

```
function setTransferFeeWallet(
    address payable _newWallet
) external onlyOwner {
    emit TransferFeeWalletUpdated(TransferFeeWallet, _newWallet);
    TransferFeeWallet = _newWallet;

    _isExcludedFromFees[_newWallet] = true;
    emit ExcludeFromFees(_newWallet, true);
}
```

- ❖ Owner can change all fees each fees maximum up-to 10%

```
function setBuyFee(uint256 _BuyFee) external onlyOwner {
    require(_BuyFee <= MAX_FEE, "Max Buy fee was 10");

    BuyFee = _BuyFee;

    totalBuyFees = BuyFee;
    emit UpdateBuyFees(_BuyFee);
}

function setSellFee(uint256 _SellFee) external onlyOwner {
    require(_SellFee <= MAX_FEE, "Max Sell fee was 10");

    SellFee = _SellFee;

    totalSellFees = SellFee;
    emit UpdateSellFees(_SellFee);
}

function setTransferFee(uint256 _TransferFee) external onlyOwner {
    require(_TransferFee <= MAX_FEE, "Max Transfer fee was 10");

    TransferFee = _TransferFee;

    totalTransferFees = TransferFee;
    emit UpdateTransferFees(_TransferFee);
}
```

- ❖ Owner can include/exclude wallets from fees

```
function excludeFromFees(address account, bool excluded) public onlyOwner {
    _isExcludedFromFees[account] = excluded;
    emit ExcludeFromFees(account, excluded);
}
```

- ❖ Owner can include/exclude multiple wallets from fees

```
function excludeMultipleAccountsFromFees(
    address[] calldata accounts,
    bool excluded
) external onlyOwner {
    for (uint256 i = 0; i < accounts.length; i++) {
        _isExcludedFromFees[accounts[i]] = excluded;
    }

    emit ExcludeMultipleAccountsFromFees(accounts, excluded);
}
```

- ❖ Owner can add/remove new pair address

```
function setAutomatedMarketMakerPair(
    address newpair,
    bool value
) external onlyOwner {
    _setAutomatedMarketMakerPair(newpair, value);
}
```


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:	4
Solidity code functional issue level:	PASS ▾
Number of owner privileges:	11
Centralization risk correlated to the active owner:	HIGH ▾
Smart contract active ownership:	ACTIVE ▾