RugFreeCoins Audit

Sisters Token

Smart Contract Security Audit

June 11, 2022

Contents

Audit details

Audited project
Sisters Token

Contract Address

0xFAb9b22CE76aFDD3F526AbF4CB035aE160Bdcdf1

Client contact

Sisters Team

Blockchain

Binance smart chain

Project website

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 Sisters Team to perform an audit of the smart contract.

https://bscscan.com/token/0xFAb9b22CE76aFDD3F526AbF4CB035aE160Bdcdf1

The focus of this audit is to verify 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, long-term sustainability, and as a guide to improving the security posture of the smart contract by remediating the issues that were identified.

.

About the project

Sisters is a token built on the Binance Smart Chain that is with an innovative investment use case the main purpose of which is to seek out constant revenue sources, which in turn, powers reward combined with the most interesting games and applications. Each transaction, purchase, and sale incur a 5% fee.

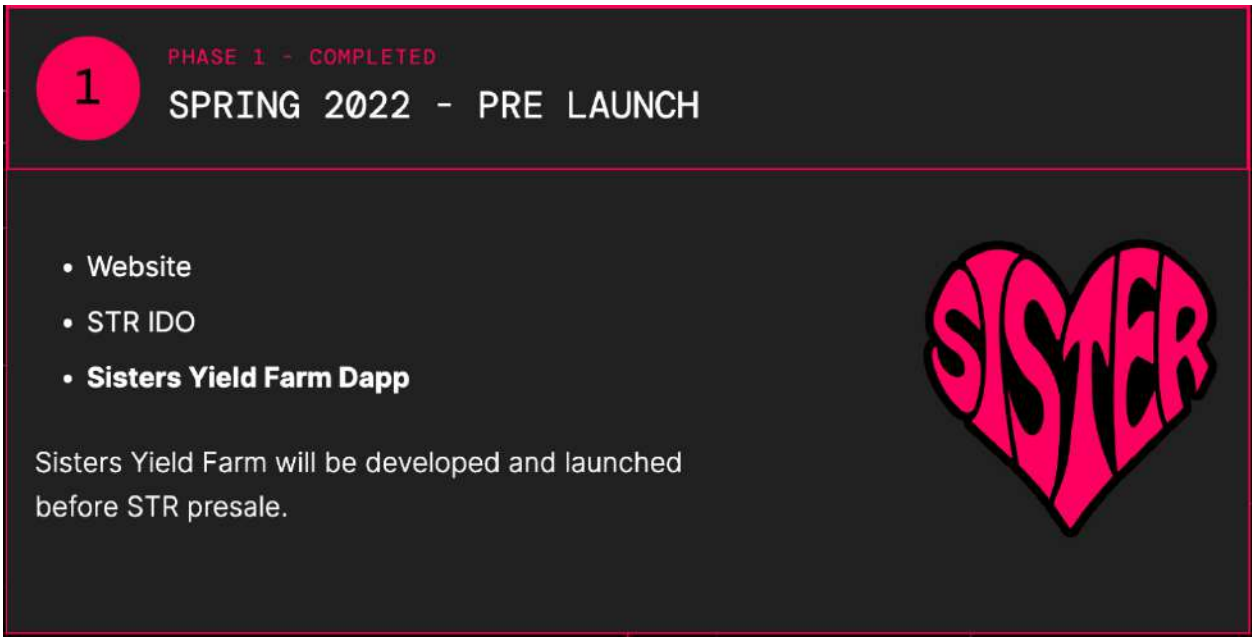
Features

- The **Sisters token** will be distributed in tokens among every holder proportional to how many tokens each individual holds in values of **1% when buying and selling.**

- The **sustainability fee of 2% when buying and selling for marketing** is what allows Sisters token to hold the aforementioned promise. Tokens will be swapped into BNB and will be sent to a marketing wallet. This way, Sisters Token will have enough funds to promote the coin and spend for future development and marketing without selling tokens as the traditional way.
-

  The additional component included under the sustainability section is a **liquidity fee of 2% when buying and selling**, which is a redistribution mechanism that ensures the trading pool always has sufficient liquidity.

ROADMAP



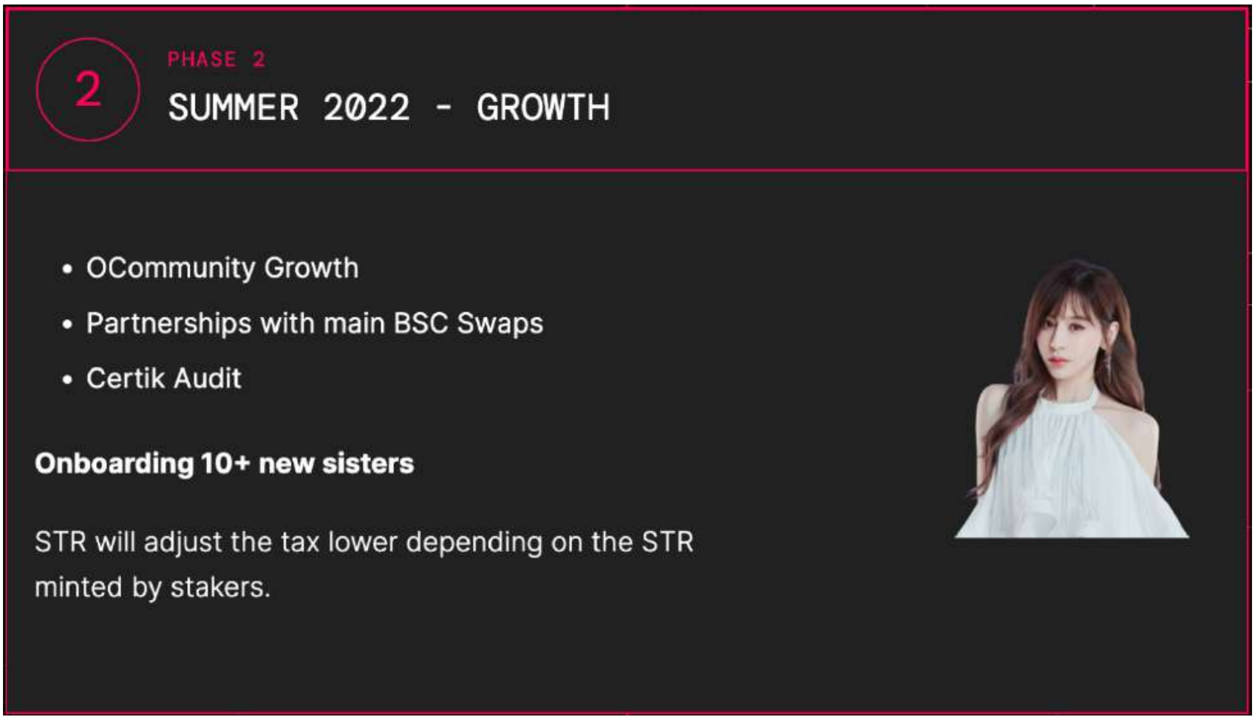
1

PHASE 1 - COMPLETED

SPRING 2022 - PRE LAUNCH

- Website
- STR IDO
- **Sisters Yield Farm Dapp**

Sisters Yield Farm will be developed and launched before STR presale.



2

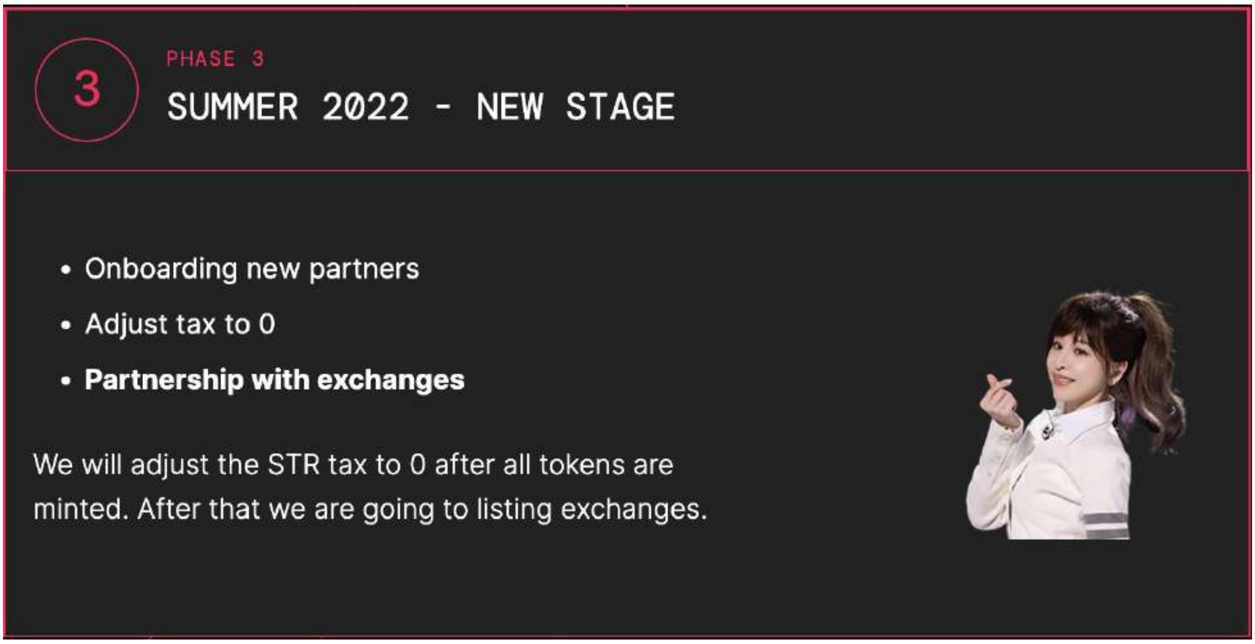
PHASE 2

SUMMER 2022 - GROWTH

- OCommunity Growth
- Partnerships with main BSC Swaps
- Certik Audit

Onboarding 10+ new sisters

STR will adjust the tax lower depending on the STR minted by stakers.



3

PHASE 3

SUMMER 2022 - NEW STAGE

- Onboarding new partners
- Adjust tax to 0
- **Partnership with exchanges**

We will adjust the STR tax to 0 after all tokens are minted. After that we are going to listing exchanges.

Tokenomics

5% fee when buying and selling

- 1% of trade goes to holders pockets in token rewards
- 2% of trade goes to the marketing wallet in BNB
- 2% of trade goes to the liquidity pool

Target market and the concept

Target market

- 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 trading or holding NFTs.
- Anyone who’s interested in Stake STR and LP tokens in Sisters Farm to earn high APR profits.
- Anyone who’s interested in taking part in the future plans of the Sisters token.
- Anyone who’s interested in making financial transactions with any other party using Sisters token as the currency.

Core concept

The Sisters reward system

1% of each transaction when buying and selling is split amongst all holders in native tokens. Holders will be eligible to receive tokens in each transaction and rewards are proportional to how many tokens each individual holds.

Sustainable mechanism

The **sustainability fee of 2% when buying and selling for marketing** is what allows Sisters Token to promote the token and use funds to further the development of the platform. Tokens will be swapped into BNB and will be sent to a marketing wallet. This way, Sisters Token will have access to the funds without selling tokens as the traditional way, which will enable them to consume funds without hurting the project.

The liquidity fee of 2% when buying and selling, is a redistribution mechanism that ensures the trading pool always has sufficient liquidity.

Potential to grow with score points

1.	Project efficiency	8/10
2.	Project uniqueness	8/10
3	Information quality	8/10
4	Service quality	8/10
5	System quality	8/10
6	Impact on the community	8/10
7	Impact on the business	8/10
8	Preparing for the future	7/10
9	Smart contract security	9/10
10	Smart contract functionality assessment	10/10
Total Points		**8.2/10**

Contract details

Token contract details for 11th June 2022

Contract name	Sisters
Contract address	0xFAb9b22CE76aFDD3F526AbF4CB035aE160Bdcdf1
Token supply	4,500,000,000
Token ticker	STR
Decimals	9
Token holders	3
Transaction count	7
Marketing wallet	0x203540a51da4a260004c0976920c139eb8018b7a
Contract deployer address	0x3Cb320ba158efbCB0a9a06D36A7780A87171F99B
Contract’s current owner address	0x3cb320ba158efbcb0a9a06d36a7780a87171f99b

Contract code function details

No	Category	Item	Result
1	Coding conventions	BRC20 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
		Overriding variables	pass
2	Function call audit	Authorization of function call	pass
		Low level function (call/delegate call) security	pass
		Returned value security	pass
		Selfdestruct function security	pass
3	Business security	Access control of owners	Centralization issues
		Business logics	pass
		Business implementations	pass
4	Integer overflow/underflow		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❗
Context	**Implementation**			
L	_msgSender	Internal 🔒		
L	_msgData	Internal 🔒		

Ownable	**Implementation**	**Context**		
└		Public ❗️	🛑	NO❗️
└	owner	Public ❗️		NO❗️
└	renounceOwnership	Public ❗️	🛑	onlyOwner
└	transferOwnership	Public ❗️	🛑	onlyOwner
└	_setOwner	Private 🔐	🛑	
IFactory	**Interface**			
└	createPair	External ❗️	🛑	NO❗️
IRouter	**Interface**			
└	factory	External ❗️		NO❗️
└	WETH	External ❗️		NO❗️
└	addLiquidityETH	External ❗️	💵	NO❗️
└	swapExactTokensForETHSupportingFeeOnTransferTokens	External ❗️	🛑	NO❗️
STR	**Implementation**	**Context, IERC20, Ownable**		
└		Public ❗️	🛑	NO❗️
└	name	Public ❗️		NO❗️

└	symbol	Public ❗		NO❗
└	decimals	Public ❗		NO❗
└	totalSupply	Public ❗		NO❗
└	balanceOf	Public ❗		NO❗
└	transfer	Public ❗	🛑	NO❗
└	allowance	Public ❗		NO❗
└	approve	Public ❗	🛑	NO❗
└	transferFrom	Public ❗	🛑	NO❗
└	increaseAllowance	Public ❗	🛑	NO❗
└	decreaseAllowance	Public ❗	🛑	NO❗
└	isExcludedFromReward	Public ❗		NO❗
└	reflectionFromToken	Public ❗		NO❗
└	tokenFromReflection	Public ❗		NO❗
└	excludeFromReward	Public ❗	🛑	onlyOwner
└	includeInReward	External ❗	🛑	onlyOwner
└	excludeFromFee	Public ❗	🛑	onlyOwner
└	includeInFee	Public ❗	🛑	onlyOwner
└	isExcludedFromFee	Public ❗		NO❗
└	setTaxes	Public ❗	🛑	onlyOwner
└	_reflectRfi	Private 🔐	🛑	

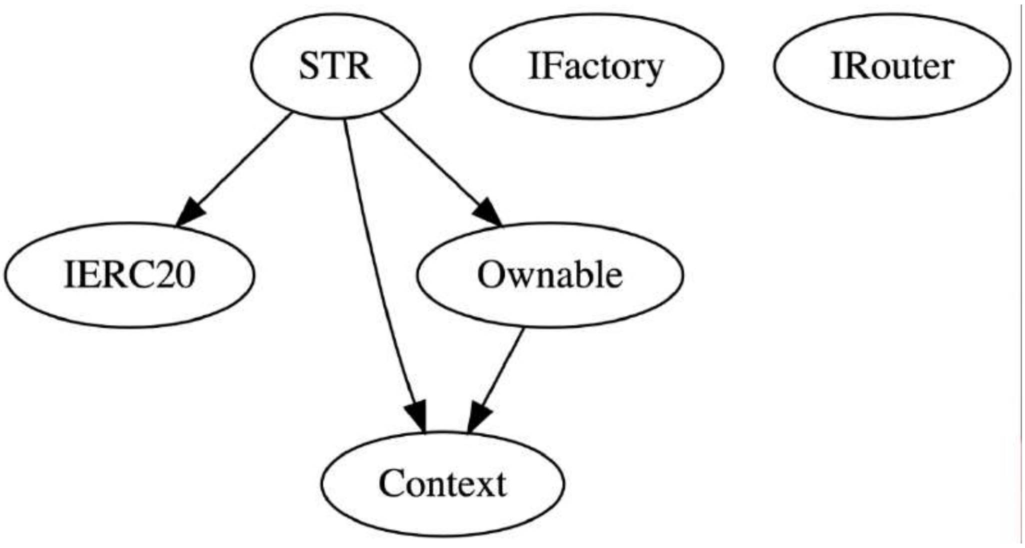
└	_takeLiquidity	Private 🔐	🛑	
└	_takeMarketing	Private 🔐	🛑	
└	_getValues	Private 🔐		
└	_getTValues	Private 🔐		
└	_getRValues	Private 🔐		
└	_getRate	Private 🔐		
└	_getCurrentSupply	Private 🔐		
└	_approve	Private 🔐	🛑	
└	_transfer	Private 🔐	🛑	
└	_tokenTransfer	Private 🔐	🛑	
└	swapAndLiquify	Private 🔐	🛑	lockTheSwap
└	addLiquidity	Private 🔐	🛑	
└	swapTokensForBNB	Private 🔐	🛑	
└	updateMarketingWallet	External ❗	🛑	onlyOwner
└	updatMaxBuyAmt	External ❗	🛑	onlyOwner

L	updateSwapTokensAtAmount	External ❗	🛑	onlyOwner
L	updateSwapEnabled	External ❗	🛑	onlyOwner
L	updateRouterAndPair	External ❗	🛑	onlyOwner
L	rescueBNB	External ❗	🛑	onlyOwner
L	rescueAnyBEP20Tokens	Public ❗	🛑	onlyOwner
L		External ❗	💵	NO❗

Legend

Symbol	Meaning
🛑	Function can modify state
💵	Function is payable

Inheritance Hierarchy



Security issue checking status

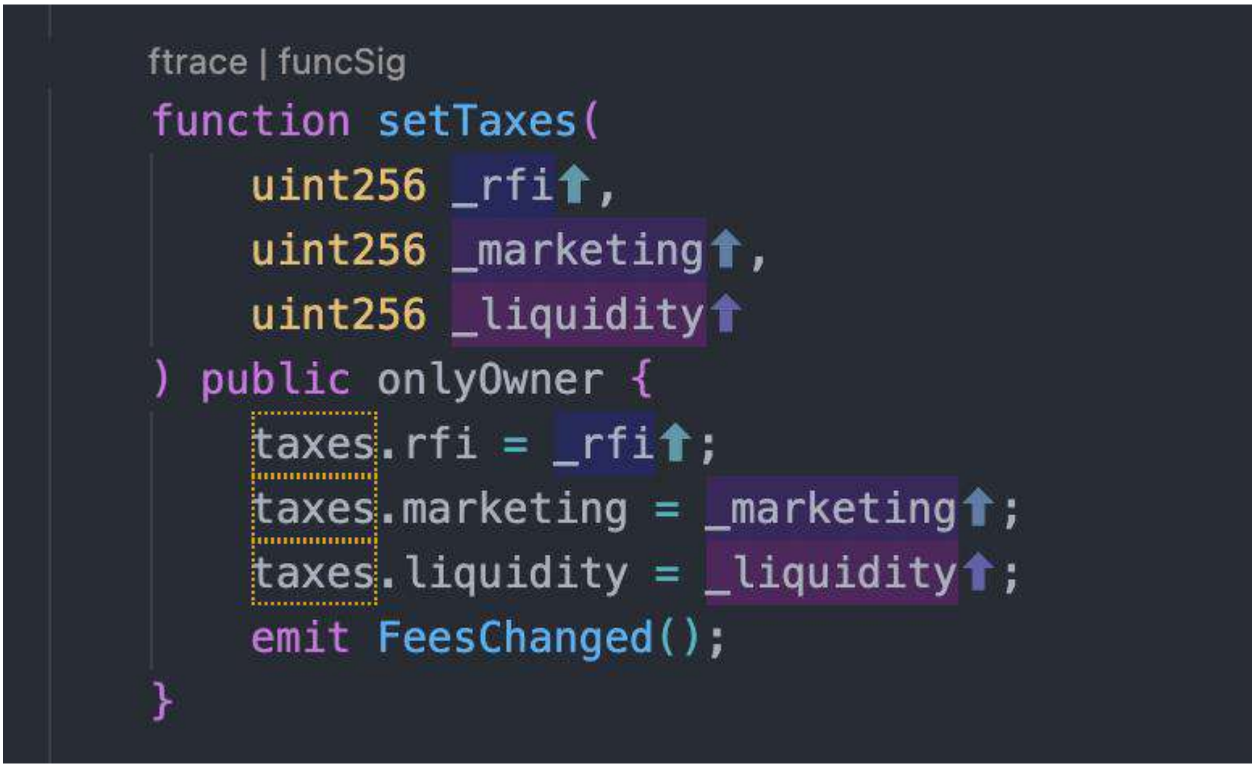
- **High severity issues**
  No High severity issues found

- **Medium severity issues**
  No medium severity issues found

- **Low severity issues**
  No low severity issues found

- **Centralization Risk**

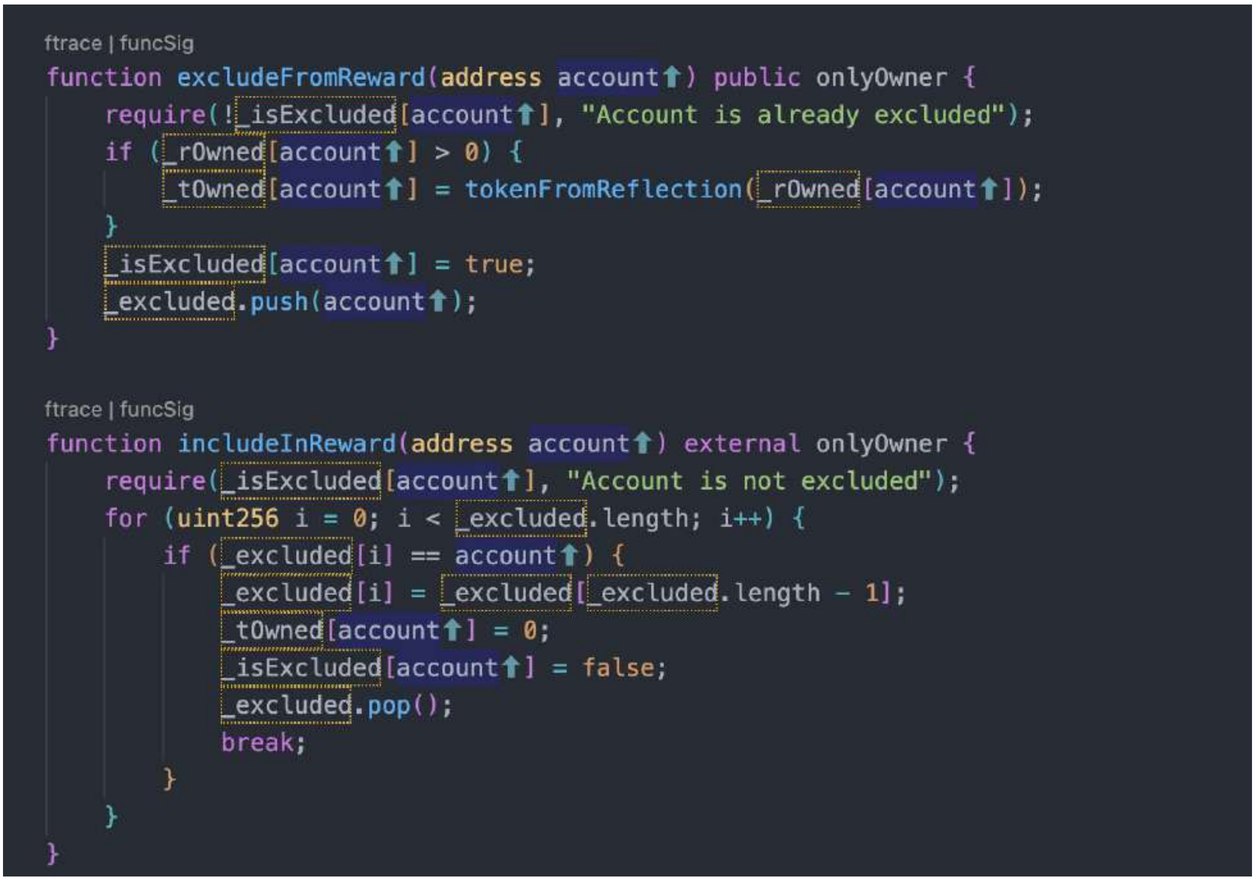
- The owner can change all fees without any maximum limit



```
ftrace | funcSig
function setTaxes(
    uint256 _rfi,
    uint256 _marketing,
    uint256 _liquidity
) public onlyOwner {
    taxes.rfi = _rfi;
    taxes.marketing = _marketing;
    taxes.liquidity = _liquidity;
    emit FeesChanged();
}
```

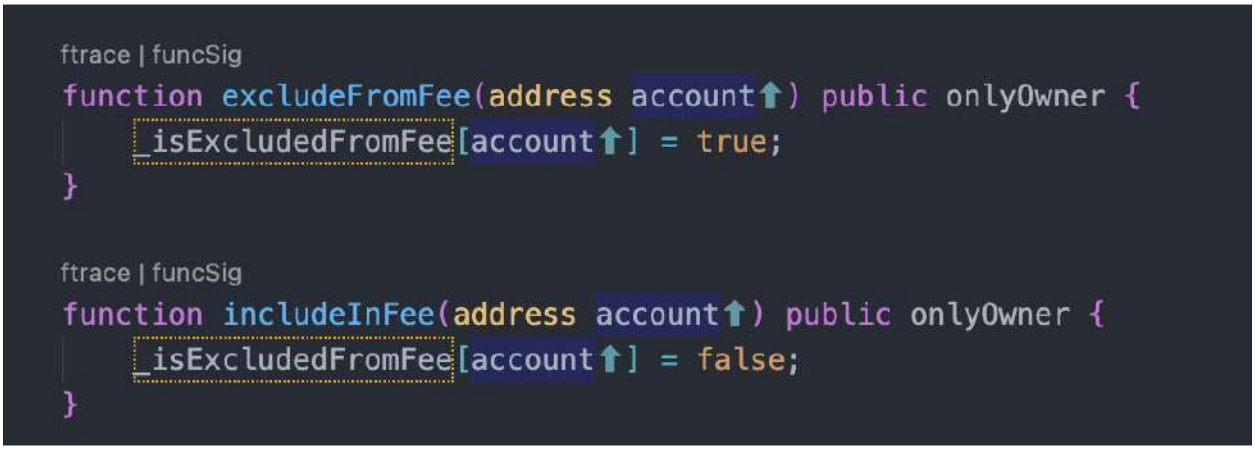
Owner privileges

❖ The owner can include and exclude wallets from rewards



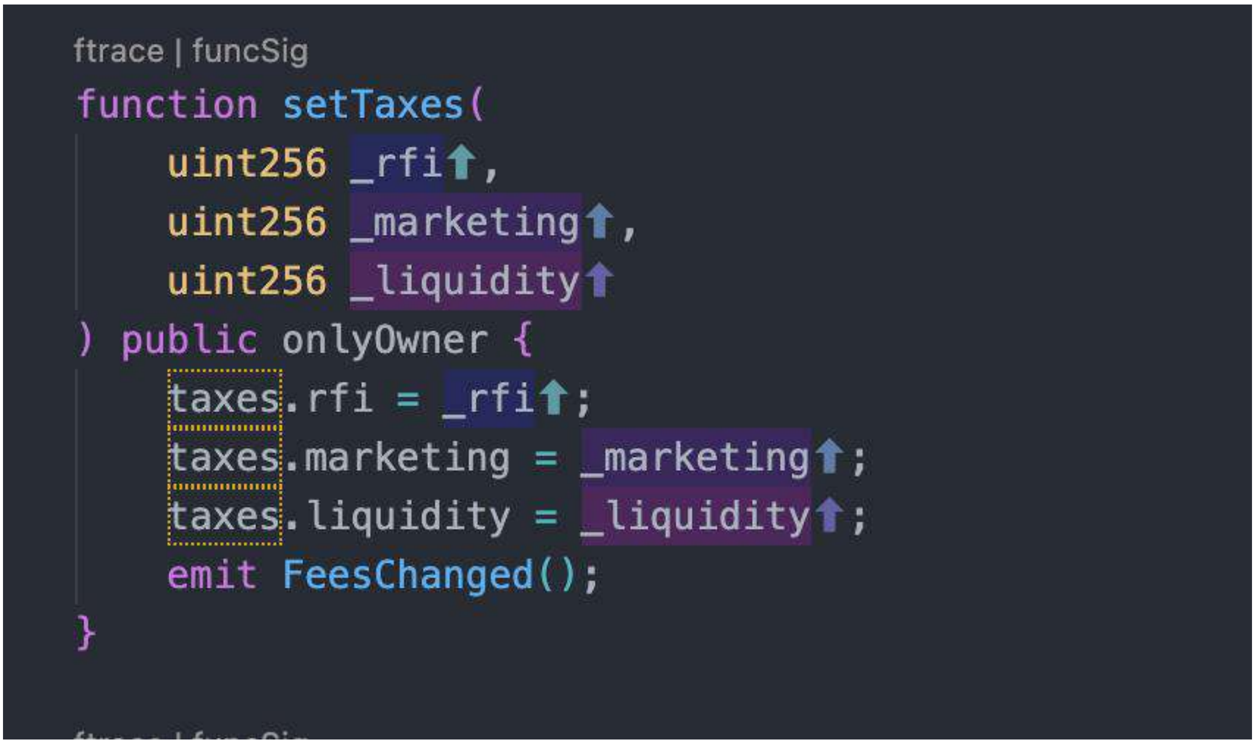
```
ftrace | funcSig
function excludeFromReward(address account) public onlyOwner {
    require(!_isExcluded[account], "Account is already excluded");
    if (_rOwned[account] > 0) {
        _tOwned[account] = tokenFromReflection(_rOwned[account]);
    }
    _isExcluded[account] = true;
    _excluded.push(account);
}

ftrace | funcSig
function includeInReward(address account) external onlyOwner {
    require(_isExcluded[account], "Account is not excluded");
    for (uint256 i = 0; i < _excluded.length; i++) {
        if (_excluded[i] == account) {
            _excluded[i] = _excluded[_excluded.length - 1];
            _tOwned[account] = 0;
            _isExcluded[account] = false;
            _excluded.pop();
            break;
        }
    }
}
```

❖ The owner can include/exclude wallets from fees



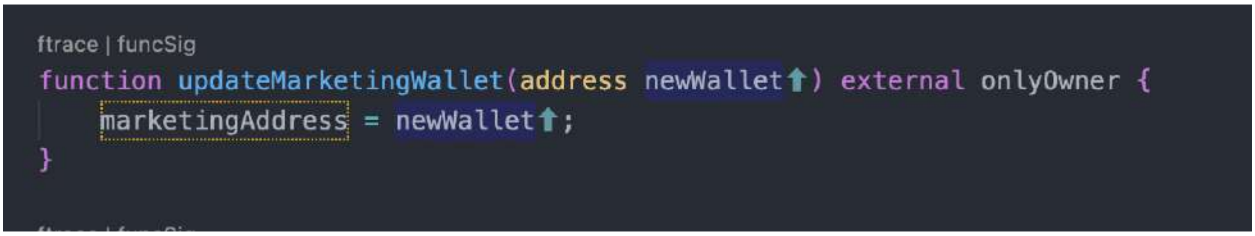
```
ftrace | funcSig
function excludeFromFee(address account) public onlyOwner {
    _isExcludedFromFee[account] = true;
}

ftrace | funcSig
function includeInFee(address account) public onlyOwner {
    _isExcludedFromFee[account] = false;
}
```

- The owner can change all fees



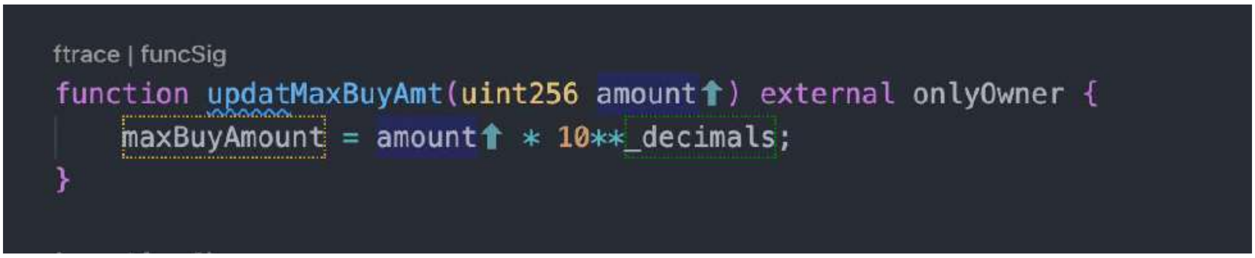
```
ftrace | funcSig
function setTaxes(
        uint256 _rfi,
        uint256 _marketing,
        uint256 _liquidity
) public onlyOwner {
        taxes.rfi = _rfi;
        taxes.marketing = _marketing;
        taxes.liquidity = _liquidity;
        emit FeesChanged();
}
```

- The owner can change marketing wallet address



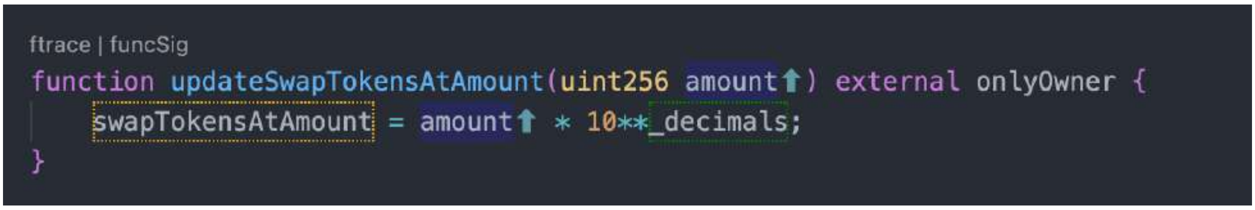
```
ftrace | funcSig
function updateMarketingWallet(address newWallet) external onlyOwner {
    marketingAddress = newWallet;
}
```

- The owner can change max buy amount



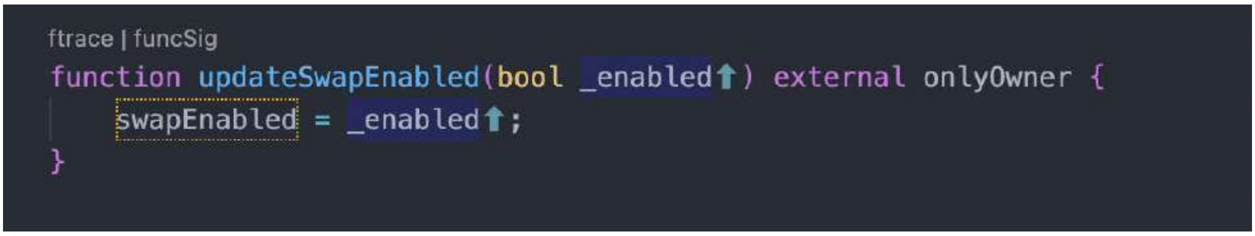
```
ftrace | funcSig
function updatMaxBuyAmt(uint256 amount) external onlyOwner {
    maxBuyAmount = amount * 10**_decimals;
}
```

- ❖ The owner can change swap point



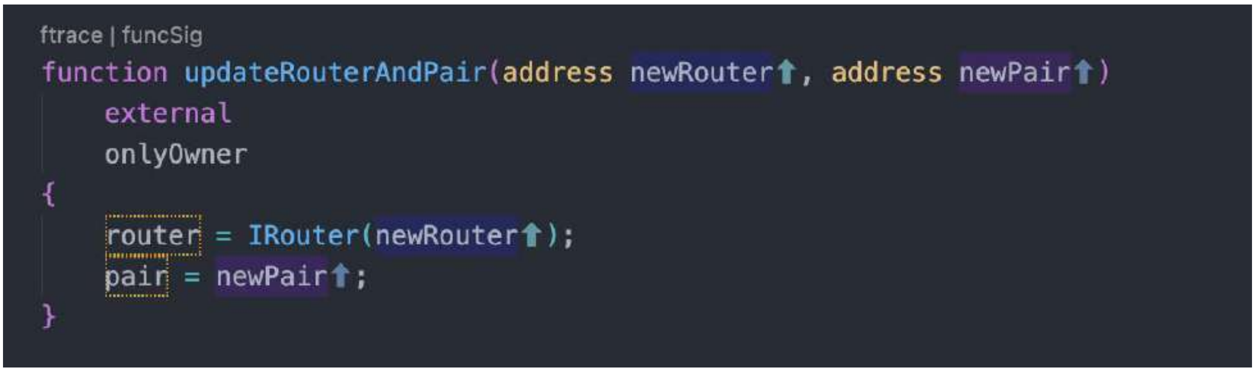
```
ftrace | funcSig
function updateSwapTokensAtAmount(uint256 amount) external onlyOwner {
    swapTokensAtAmount = amount * 10**_decimals;
}
```

- ❖ The owner can enable/disable swapping



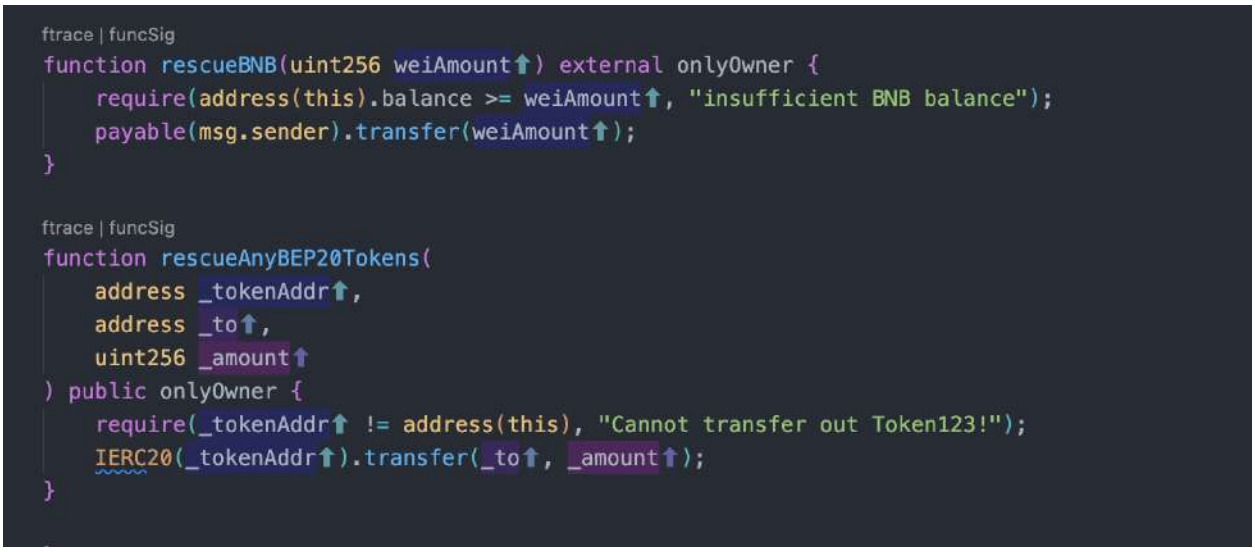
```
ftrace | funcSig
function updateSwapEnabled(bool _enabled) external onlyOwner {
    swapEnabled = _enabled;
}
```

- ❖ The owner can change router and pair address



```
ftrace | funcSig
function updateRouterAndPair(address newRouter, address newPair)
    external
    onlyOwner
{
    router = IRouter(newRouter);
    pair = newPair;
}
```

- ❖ The owner can get bnb and any BEP20 tokens in contract to the owner wallet



```
ftrace | funcSig
function rescueBNB(uint256 weiAmount) external onlyOwner {
    require(address(this).balance >= weiAmount, "insufficient BNB balance");
    payable(msg.sender).transfer(weiAmount);
}

ftrace | funcSig
function rescueAnyBEP20Tokens(
    address _tokenAddr,
    address _to,
    uint256 _amount
) public onlyOwner {
    require(_tokenAddr != address(this), "Cannot transfer out Token123!");
    IERC20(_tokenAddr).transfer(_to, _amount);
}
```

Audit conclusion

RugFreeCoins team has performed in-depth testings, 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: **PASSED**

Number of risk issues: **1**

Solidity code functional issue level: **PASSED**

Number of owner privileges: **9**

Centralization risk correlated to the active owner: **HIGH**

Smart contract active ownership: **YES**