



Cyberscope

Audit Report

Robot Dog

April 2022

Type BEP20

Network BSC

Address 0xfCAE297514B07c043D0bC0b5C372E6E7e1e094e2

Audited by © cyberscope

Table of Contents

Table of Contents	1
Contract Review	3
Source Files	3
Audit Updates	3
Contract Analysis	4
ELFM - Exceed Limit Fees Manipulation	5
Description	5
Recommendation	5
Contract Diagnostics	7
L01 - Public Function could be Declared External	8
Description	8
Recommendation	8
L02 - State Variables could be Declared Constant	9
Description	9
Recommendation	9
L04 - Conformance to Solidity Naming Conventions	10
Description	10
Recommendation	10
Contract Functions	11
Contract Flow	15
Domain Info	16
Summary	17
Disclaimer	18
About Cyberscope	19

Contract Review

Contract Name	RDOG
Compiler Version	v0.6.12+commit.27d51765
Optimization	200 runs
Licence	MIT
Explorer	https://bscscan.com/token/0xfCAE297514B07c043D0bC0b5C372E6E7e1e094e2
Symbol	RDOG
Decimals	18
Total Supply	10,000,000,000
Domain	robotdogdao.com

Source Files

Filename	SHA256
contract.sol	92d431d0d6b103d6291d95589b6e7e8a8179f19083365438da892068dfff8180

Audit Updates

Initial Audit	17th April 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

ELFM - Exceed Limit Fees Manipulation

Criticality	critical
Location	contract.sol#L573,770

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 `setRates` function with high percentage values.

```
function setRates(
    uint256 taxFee,
    uint256 liquidityFee,
    uint256 lpHolderFee,
    uint256 lpPowerFee,
    uint256 fundFee,
    uint256 communityFee,
    uint256 burnFee
) external onlyOwner {
    require(
        taxFee ==
            (liquidityFee +
             lpHolderFee +
             lpPowerFee +
             fundFee +
             communityFee +
             burnFee),
        "params error"
    );
    _taxFee = taxFee;
    _liquidityFee = liquidityFee;
    _lpHolderFee = lpHolderFee;
    _lpPowerFee = lpPowerFee;
    _fundFee = fundFee;
    _communityFee = communityFee;
    _burnFee = burnFee;
}
```

The contract owner can also convert the contract into a **HONEYPOT** by disabling the buyfees, and increasing the sell fee to very high value.

```
    if (
      (_buyFeeEnable && isMarketPair[sender]) ||
      (_sellFeeEnable && isMarketPair[recipient])
    ) {
      tax = amount.mul(_taxFee).div(100);
    }
```

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
●	L01	Public Function could be Declared External
●	L02	State Variables could be Declared Constant
●	L04	Conformance to Solidity Naming Conventions

L01 - Public Function could be Declared External

Criticality

minor

Location

contract.sol#L125,134,442,450,459,468,477,494,506,814,818,822,836

Description

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

```
takeOutTokenInCase
decimals
symbol
name
decreaseAllowance
increaseAllowance
transferFrom
approve
allowance
...
```

Recommendation

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

L02 - State Variables could be Declared Constant

Criticality

minor

Location

contract.sol#L409

Description

Constant state variables should be declared constant to save gas.

`DeadAddress`

Recommendation

Add the constant attribute to state variables that never change.

L04 - Conformance to Solidity Naming Conventions

Criticality	minor
Location	contract.sol#L147,716,721,726,731,736,741,746,383,384,390,392,393,394,395,396,397,398,399,400,404,405,406,407,408,409,837,838,839

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.

```
_to  
_amount  
_token  
DeadAddress  
Community  
Fund  
LPPower  
LPHolder  
Liquidity  
...
```

Recommendation

Follow the Solidity naming convention.

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

Contract Functions

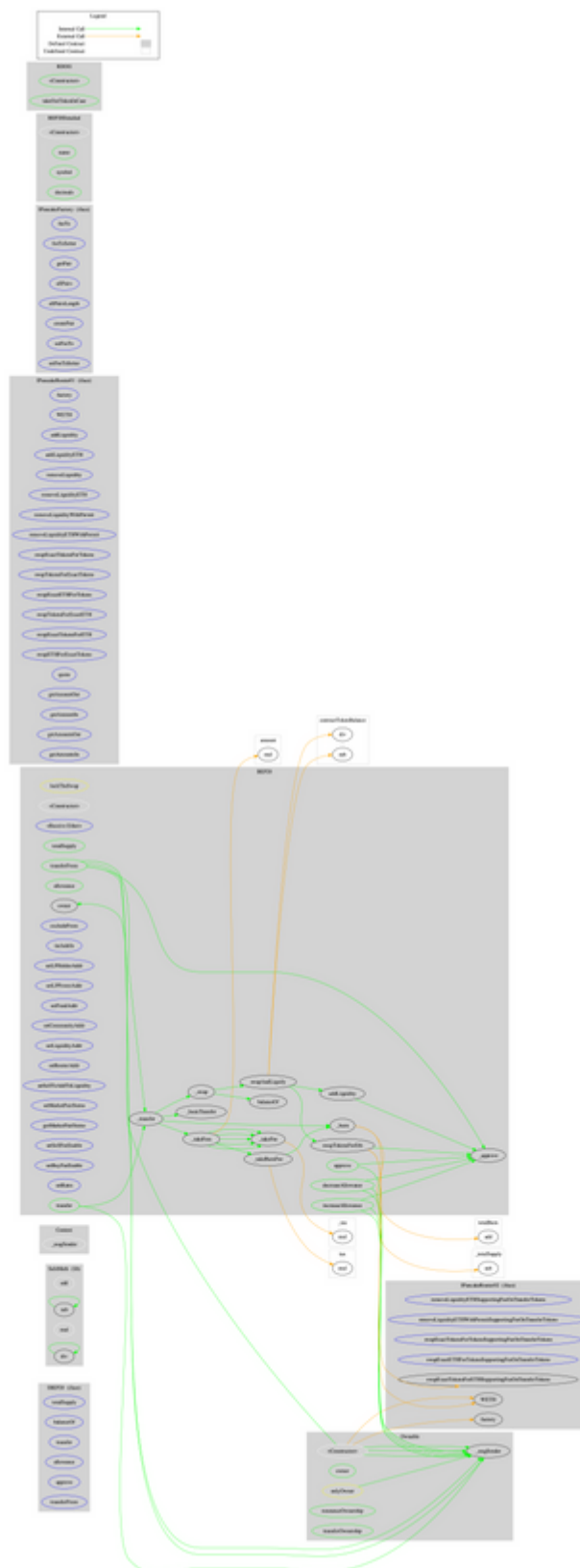
Contract	Type	Bases		
	Function Name	Visibility	Mutability	Modifiers
IBEP20	Interface			
	totalSupply	External		-
	balanceOf	External		-
	transfer	External	✓	-
	allowance	External		-
	approve	External	✓	-
	transferFrom	External	✓	-
SafeMath	Library			
	add	Internal		
	sub	Internal		
	sub	Internal		
	mul	Internal		
	div	Internal		
	div	Internal		
Context	Implementation			
	_msgSender	Internal		
Ownable	Implementation	Context		
	<Constructor>	Internal	✓	
	owner	Public		-
	renounceOwnership	Public	✓	onlyOwner
	transferOwnership	Public	✓	onlyOwner
IPancakeRouter01	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		-
IPancakeRouter02	Interface	IPancakeRouter01		
	removeLiquidityETHSupportingFeeOnTransferTokens	External	✓	-
	removeLiquidityETHWithPermitSupportingFeeOnTransferTokens	External	✓	-
	swapExactTokensForTokensSupportingFeeOnTransferTokens	External	✓	-
	swapExactETHForTokensSupportingFeeOnTransferTokens	External	Payable	-
	swapExactTokensForETHSupportingFeeOnTransferTokens	External	✓	-
IPancakeFactory	Interface			
	feeTo	External		-
	feeToSetter	External		-
	getPair	External		-
	allPairs	External		-
	allPairsLength	External		-

	createPair	External	✓	-
	setFeeTo	External	✓	-
	setFeeToSetter	External	✓	-
BEP20	Implementation	Context, Ownable, IBEP20		
	<Constructor>	Internal	✓	
	<Receive Ether>	External	Payable	-
	totalSupply	Public		-
	balanceOf	Public		-
	transfer	Public	✓	-
	allowance	Public		-
	approve	Public	✓	-
	transferFrom	Public	✓	-
	increaseAllowance	Public	✓	-
	decreaseAllowance	Public	✓	-
	_transfer	Internal	✓	
	_basicTransfer	Internal	✓	
	_takeFees	Internal	✓	
	_takeFee	Private	✓	
	_takeBurnFee	Private	✓	
	_burn	Private	✓	
	_approve	Internal	✓	
	_swap	Internal	✓	
	swapAndLiquify	Private	✓	lockTheSwap
	swapTokensForEth	Private	✓	
	addLiquidity	Private	✓	
	excludeFrom	External	✓	onlyOwner
	includeIn	External	✓	onlyOwner
	setLPHolderAddr	External	✓	onlyOwner
	setLPPowerAddr	External	✓	onlyOwner
	setFundAddr	External	✓	onlyOwner
	setCommunityAddr	External	✓	onlyOwner
	setLiquidityAddr	External	✓	onlyOwner
	setRouterAddr	External	✓	onlyOwner

	setSellToAddToLiquidity	External	✓	onlyOwner
	setMarketPairStatus	External	✓	onlyOwner
	getMarketPairStatus	External		-
	setSellFeeEnable	External	✓	onlyOwner
	setBuyFeeEnable	External	✓	onlyOwner
	setRates	External	✓	onlyOwner
BEP20Detailed	Implementation	BEP20		
	<Constructor>	Internal	✓	BEP20
	name	Public		-
	symbol	Public		-
	decimals	Public		-
RDOG	Implementation	BEP20Detailed		
	<Constructor>	Public	✓	BEP20Detailed
	takeOutTokenInCase	Public	✓	onlyOwner

Contract Flow



Domain Info

Domain Name	robotdogdao.com
Registry Domain ID	2687747777_DOMAIN_COM-VRSN
Creation Date	2022-04-08T15:22:33.00Z
Updated Date	0001-01-01T00:00:00.00Z
Registry Expiry Date	2027-04-08T15:22:33.00Z
Registrar WHOIS Server	whois.namecheap.com
Registrar URL	http://www.namecheap.com
Registrar	NAMECHEAP INC
Registrar IANA ID	1068

The domain has been created 9 days before the creation of the audit. It will expire in almost 5 years.

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

Summary

Robot Dog is an interesting project that has a friendly and growing community. There are some functions that can be abused by the owner, like manipulating fees up to 100%. The contract can also be converted into a **HONEYPOT** and prevent users from selling if the contract configuration is abused by the owner. A multi-wallet signing pattern will provide security against potential hacks. Temporarily locking the contract or renouncing ownership will eliminate all the contract threats.

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 provides all the essential tools to assist users draw their own conclusions.



The Cyberscope team

<https://www.cyberscope.io>