



Cyberscope

# Audit Report

## **LFGdoge**

March 2022

Type       BEP20

Network     BSC

Address     0x0711b6F1D4205AAf7b02a6De014394646b1D841b

Audited by  © cyberscope

# Table of Contents

<b>Table of Contents</b>	<b>1</b>
<b>Contract Review</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>ULTW - Unlimited Liquidity to Team Wallet</b>	<b>7</b>
Description	7
Recommendation	7
<b>BC - Blacklisted Contracts</b>	<b>8</b>
Description	8
Recommendation	8
<b>Contract Diagnostics</b>	<b>9</b>
<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>L05 - Unused State Variable</b>	<b>12</b>
Description	12
Recommendation	12

<b>L04 - Conformance to Solidity Naming Conventions</b>	<b>13</b>
Description	13
Recommendation	13
<b>L09 - Dead Code Elimination</b>	<b>14</b>
Description	14
Recommendation	14
<b>L07 - Missing Events Arithmetic</b>	<b>15</b>
Description	15
Recommendation	15
<b>Contract Functions</b>	<b>16</b>
<b>Contract Flow</b>	<b>21</b>
<b>Domain Info</b>	<b>22</b>
<b>Summary</b>	<b>23</b>
<b>Disclaimer</b>	<b>24</b>
<b>About Cyberscope</b>	<b>25</b>

## Contract Review

<b>Contract Name</b>	LFGdoge
<b>Compiler Version</b>	v0.8.4+commit.c7e474f2
<b>Optimization</b>	200 runs
<b>Licence</b>	None
<b>Explorer</b>	<a href="https://bscscan.com/token/0x0711b6F1D4205AAf7b02a6De014394646b1D841b">https://bscscan.com/token/0x0711b6F1D4205AAf7b02a6De014394646b1D841b</a>
<b>Symbol</b>	LFGdoge
<b>Decimals</b>	9
<b>Total Supply</b>	1,000,000,000,000,000
<b>Source</b>	contract.sol
<b>Domain</b>	lfgdoge.com

## Audit Updates

<b>Initial Audit</b>	10th March 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#L635

### Description

The contract owner has the authority to stop transactions for all users excluding the owner. The owner may take advantage of it by setting the `_maxTxAmount` or `_walletMax` to zero.

```
if(!isTxLimitExempt[sender] && !isTxLimitExempt[recipient]) {  
    require(amount <= _maxTxAmount, "Transfer amount exceeds the maxTxAmount.");  
}
```

### Recommendation

The contract could embody a check for not allowing setting the `_maxTxAmount` 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.

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.

## ELFM - Exceed Limit Fees Manipulation

<b>Criticality</b>	critical
<b>Location</b>	contract.sol#L534

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

```
function setTaxes(uint256 newMarketingFee) external onlyOwner() {  
    _marketingFee = newMarketingFee;  
    _totalTax = _marketingFee;  
}
```

### 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.

## ULTW - Unlimited Liquidity to Team Wallet

Criticality	minor
Location	contract.sol#L671

### Description

The contract owner has the authority to transfer funds to the team wallet. These funds have been accumulated from fees collected from the contract and sold for bnb.

```
function swapAndLiquify(uint256 tAmount) private lockTheSwap {
    swapTokensForEth(tAmount);
    uint256 amountReceived = address(this).balance;

    if(amountReceived > 0)
        transferToAddressETH(marketingWalletAddress, amountReceived);
}
```

### Recommendation

The contract could embody a check for the maximum amount of tokens that can be sold in one transaction.

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.



## BC - Blacklisted Contracts

Criticality	medium
Location	contract.sol#L1

### Description

The contract owner has the authority to stop contracts from transactions. The owner may take advantage of it by calling the `isbotBlackList` function.

```
require(!isbotBlackList[sender], "account is bot");
```

### Recommendation

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
●	L05	Unused State Variable
●	L04	Conformance to Solidity Naming Conventions
●	L09	Dead Code Elimination
●	L07	Missing Events Arithmetic

## L01 - Public Function could be Declared External

<b>Criticality</b>	minor
<b>Location</b>	contract.sol#L165,170,176,471,475,479,483,491,495,500 and 11 more

### Description

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

```
transferFrom
transfer
changeRouterVersion
rescueToken
getCirculatingSupply
setSwapAndLiquifyByLimitOnly
setSwapAndLiquifyEnabled
setIsExcludedFromFee
setMarketPairStatus
...
```

### Recommendation

Use the external attribute for functions never called from the contract

## L02 - State Variables could be Declared Constant

**Criticality**

minor

**Location**

contract.sol#L389,387,388,146,145

### Description

Constant state variables should be declared constant to save gas.

```
asdasd
_lockTime
_symbol
_name
_decimals
```

### Recommendation

Add the constant attribute to state variables that never change.

## L05 - Unused State Variable

**Criticality**

minor

**Location**

contract.sol#L145,146

### Description

There are segments that contain unused state variables.

```
_lockTime  
asdasd
```

### Recommendation

Remove unused state variables.

## L04 - Conformance to Solidity Naming Conventions

**Criticality**

minor

**Location**

contract.sol#L213,214,230,249,563,699,394,403,405,408 and 1 more

### 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.

```
_walletMax  
_maxTxAmount  
_totalTax  
_marketingFee  
_balances  
_account  
_enabled  
WETH  
MINIMUM_LIQUIDITY  
...
```

### Recommendation

Follow the Solidity naming convention.

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

## L09 - Dead Code Elimination

<b>Criticality</b>	minor
<b>Location</b>	contract.sol#L123,106,110,114,118,87,98

### Description

Functions that are not used in the contract, and make the code's size bigger.

```
sendValue  
isContract  
functionCallWithValue  
functionCall  
_functionCallWithValue
```

### Recommendation

Remove unused functions.

## L07 - Missing Events Arithmetic

<b>Criticality</b>	minor
<b>Location</b>	contract.sol#L534,539,551,555

### Description

Detected missing events for critical arithmetic parameters. There are functions that have no event emitted, so it is difficult to track off-chain changes.

```
minimumTokensBeforeSwap = newLimit
_walletMax = newLimit
_maxTxAmount = maxTxAmount
_totalTax = _marketingFee
```

### Recommendation

Emit an event for critical parameter changes.



# Contract Functions

Contract	Type	Bases		
	Function Name	Visibility	Mutability	Modifiers
<b>Context</b>	Implementation			
	_msgSender	Internal		
	_msgData	Internal		
<b>IERC20</b>	Interface			
	totalSupply	External		-
	balanceOf	External		-
	transfer	External	✓	-
	allowance	External		-
	approve	External	✓	-
	transferFrom	External	✓	-
<b>SafeMath</b>	Library			
	add	Internal		
	sub	Internal		
	sub	Internal		
	mul	Internal		
	div	Internal		
	div	Internal		
	mod	Internal		
	mod	Internal		
<b>Address</b>	Library			
	isContract	Internal		
	sendValue	Internal	✓	
	functionCall	Internal	✓	
	functionCall	Internal	✓	
	functionCallWithValue	Internal	✓	
	functionCallWithValue	Internal	✓	

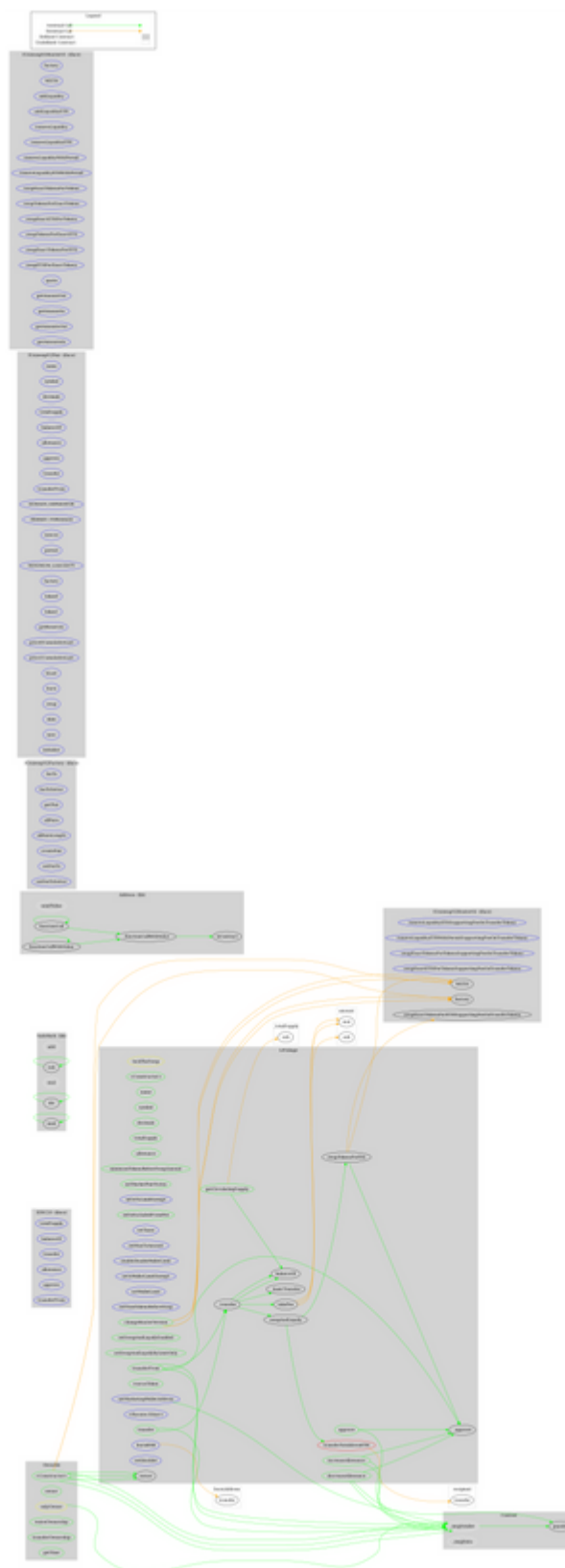
	_functionCallWithValue	Private	✓	
<b>Ownable</b>	Implementation	Context		
	<Constructor>	Public	✓	-
	owner	Public		-
	waiveOwnership	Public	✓	onlyOwner
	transferOwnership	Public	✓	onlyOwner
	getTime	Public		-
<b>IUniswapV2Factory</b>	Interface			
	feeTo	External		-
	feeToSetter	External		-
	getPair	External		-
	allPairs	External		-
	allPairsLength	External		-
	createPair	External	✓	-
	setFeeTo	External	✓	-
	setFeeToSetter	External	✓	-
<b>IUniswapV2Pair</b>	Interface			
	name	External		-
	symbol	External		-
	decimals	External		-
	totalSupply	External		-
	balanceOf	External		-
	allowance	External		-
	approve	External	✓	-
	transfer	External	✓	-
	transferFrom	External	✓	-
	DOMAIN_SEPARATOR	External		-
	PERMIT_TYPEHASH	External		-
	nonces	External		-
	permit	External	✓	-
	MINIMUM_LIQUIDITY	External		-

	factory	External		-
	token0	External		-
	token1	External		-
	getReserves	External		-
	price0CumulativeLast	External		-
	price1CumulativeLast	External		-
	kLast	External		-
	burn	External	✓	-
	swap	External	✓	-
	skim	External	✓	-
	sync	External	✓	-
	initialize	External	✓	-
<b>IUniswapV2Router01</b>	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		-

IUniswapV2Router02	Interface	IUniswapV2Router01		
	removeLiquidityETHSupportingFeeOnTransferTokens	External	✓	-
	removeLiquidityETHWithPermitSupportingFeeOnTransferTokens	External	✓	-
	swapExactTokensForTokensSupportingFeeOnTransferTokens	External	✓	-
	swapExactETHForTokensSupportingFeeOnTransferTokens	External	Payable	-
	swapExactTokensForETHSupportingFeeOnTransferTokens	External	✓	-
LFGdoge	Implementation	Context, IERC20, Ownable		
	<Constructor>	Public	✓	-
	name	Public		-
	symbol	Public		-
	decimals	Public		-
	totalSupply	Public		-
	balanceOf	Public		-
	allowance	Public		-
	increaseAllowance	Public	✓	-
	decreaseAllowance	Public	✓	-
	minimumTokensBeforeSwapAmount	Public		-
	approve	Public	✓	-
	_approve	Private	✓	
	setMarketPairStatus	Public	✓	onlyOwner
	setIsTxLimitExempt	External	✓	onlyOwner
	setIsExcludedFromFee	Public	✓	onlyOwner
	setTaxes	External	✓	onlyOwner
	setMaxTxAmount	External	✓	onlyOwner
	enableDisableWalletLimit	External	✓	onlyOwner
	setIsWalletLimitExempt	External	✓	onlyOwner
	setWalletLimit	External	✓	onlyOwner
	setNumTokensBeforeSwap	External	✓	onlyOwner
	setMarketingWalletAddress	External	✓	onlyOwner

	setSwapAndLiquifyEnabled	Public	✓	onlyOwner
	setSwapAndLiquifyByLimitOnly	Public	✓	onlyOwner
	getCirculatingSupply	Public		-
	burnBNB	External	✓	onlyOwner
	rescueToken	Public	✓	onlyOwner
	transferToAddressETH	Private	✓	
	changeRouterVersion	Public	✓	onlyOwner
	<Receive Ether>	External	Payable	-
	transfer	Public	✓	-
	transferFrom	Public	✓	-
	_transfer	Private	✓	
	_basicTransfer	Internal	✓	
	swapAndLiquify	Private	✓	lockTheSwap
	swapTokensForEth	Private	✓	
	setblocklist	External	✓	onlyOwner
	takeFee	Internal	✓	

# Contract Flow



## Domain Info

<b>Domain Name</b>	lfgdoge.com
<b>Registry Domain ID</b>	2680473873_DOMAIN_COM-VRSN
<b>Creation Date</b>	2022-03-09T19:24:04Z
<b>Updated Date</b>	2022-03-09T19:24:04Z
<b>Registry Expiry Date</b>	2023-03-09T19:24:04Z
<b>Registrar WHOIS Server</b>	whois.godaddy.com
<b>Registrar URL</b>	<a href="http://www.godaddy.com">http://www.godaddy.com</a>
<b>Registrar</b>	GoDaddy.com, LLC
<b>Registrar IANA ID</b>	146

The domain has been created about 19 hours before the creation of the audit. It will expire in 12 months.

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

## Summary

There are some functions that can be abused by the owner, like manipulating fees, transferring funds to the team's wallet, blacklisting contracts and stopping transactions. 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.

Coinscope 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>