Cyberscope

Audit Report

Doge Superman

March 2022

Type	BEP20
Network	BSC
Address	0xe636192996aF2F728af308ddEc2380e15d238bdE
Audited by	© cyberscope

Table of Contents

Contract Review

Contract Name	DogeSuperman
Compiler Version	v0.8.7+commit.e28d00a7
Optimization	200 runs
Licence	None
Explorer	https://bscscan.com/token/0xe636192996aF2F728af308ddEc2380e15d238bdE
Symbol	DogeS
Decimals	18
Total Supply	1,000,000,000,000
Source	DogeSuperman.sol
Domain	doges.fun

Audit Updates

Initial Audit	6th March 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

ST - Stop Transactions

Criticality	medium
Location	contract.sol#L784,789

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 `_maxWalletToken` to zero.

```
require(amount <= _maxTxAmount, "You are trying to buy more than the max
transaction limit.");
```

```
require((heldTokens + amount) <= _maxWalletToken,"You are trying to buy too many
tokens. You have reached the limit for one wallet.");}
```

Recommendation

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

ULTW - Unlimited Liquidity to Team Wallet

Criticality	minor
Location	contract.sol#L874

Description

The contract owner receives all the funds from either the swap and liquify feature or by manually calling the `process_Tokens_Now`. These funds have been accumulated from fees collected from the contract. If there are a lot of accumulated fees this will significantly decrease the token's price.

```
swapTokensForBNB(contractTokenBalance);
uint256 contractBNB = address(this).balance;
sendToWallet(Wallet_Dev,contractBNB);
```

Recommendation

The contract could embody a check for not liquefying and sending to the contract owners more than a reasonable amount. The contract could implement a feature that works as a balancer, like auto generated liquidity or buyback.

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	critical
Location	contract.sol#L634

Description

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

```
function blacklist_Add_Wallets(address[] calldata addresses) external onlyOwner
{

    uint256 startGas;
    uint256 gasUsed;

for (uint256 i; i < addresses.length; ++i) {
    if(gasUsed < gasleft()) {
    startGas = gasleft();
    if(!_isBlacklisted[addresses[i]]){
    _isBlacklisted[addresses[i]] = true;}
    gasUsed = startGas - gasleft();
}
}
}
```

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
●	L14	Uninitialized Variables in Local Scope

L01 - Public Function could be Declared External

Criticality	minor
Location	contract.sol#L159,165,495,499,503,507,515,520,524,529 and 12 more

Description

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

```
set_New_Pair_Address
set_New_Router_Address
set_New_Router_and_Make_Pair
process_Tokens_Now
blacklist_Switch
set_Number_Of_Transactions_Before_Liquify_Trigger
set_Swap_And_Liquify_Enabled
Wallet_Update_Dev
includeInFee
...
```

Recommendation

Use the external attribute for functions never called from the contract

L02 - State Variables could be Declared Constant

Criticality	minor
Location	contract.sol#L384,385,397,395,396,399,398,407

Description

Constant state variables should be declared constant to save gas.

```
maxPossibleFee
_tTotal
_tFeeTotal
_symbol
_name
_decimals
Wallet_zero
Wallet_Burn
```

Recommendation

Add the constant attribute to state variables that never change.

L05 - Unused State Variable

Criticality	minor
Location	contract.sol#L385,399,429,434

Description

There are segments that contain unused state variables.

```
_previousMaxTxAmount
_previousMaxWalletToken
_tFeeTotal
Wallet_zero
```

Recommendation

Remove unused state variables.

L04 - Conformance to Solidity Naming Conventions

Criticality	minor
Location	contract.sol#L196,197,210,227,582,593,606,612,634,652 and 31 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.

```
_maxTxAmount
_maxWalletToken
_sellFee
_buyFee
_TotalFee
Wallet_zero
Wallet_Burn
Wallet_Dev
_isBlacklisted
...
```

Recommendation

Follow the Solidity naming convention.
https://docs.soliditylang.org/en/v0.4.25/style-guide.html#naming-conventions

L09 - Dead Code Elimination

Criticality	minor
Location	contract.sol#L120,80,84,88,92,110,114,99,103,68 and 5 more

Description

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

```
mul
div
_msgData
sendValue
isContract
functionStaticCall
functionDelegateCall
functionCallWithValue
functionCall
...
```

Recommendation

Remove unused functions.

L07 - Missing Events Arithmetic

Criticality	minor
Location	contract.sol#L582,612,718,723

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.

```
_maxWalletToken = _tTotal * maxWallPercent_x100 / 10000
_maxTxAmount = _tTotal * maxTxPercent_x100 / 10000
swapTrigger = number_of_transactions
_sellFee = Sell_Fee
```

Recommendation

Emit an event for critical parameter changes.

L14 - Uninitialized Variables in Local Scope

Criticality	minor
Location	contract.sol#L639,637,657,655

Description

The are variables that are defined in the local scope and are not initialized.

```
gasUsed
i
```

Recommendation

All the local scoped variables should be initialized.

Contract Functions

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

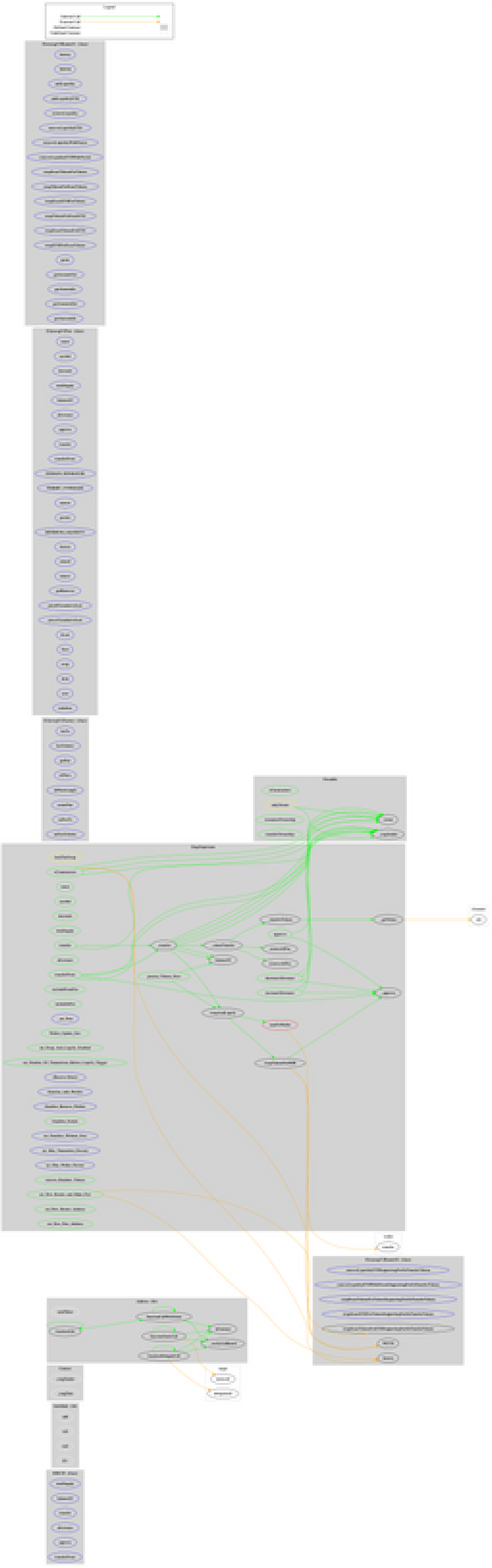
	functionDelegateCall	Internal	✓	
	functionDelegateCall	Internal	✓	
	_verifyCallResult	Private		
Ownable	Implementation	Context		
	<Constructor>	Public	✓	-
	owner	Public		-
	renounceOwnership	Public	✓	onlyOwner
	transferOwnership	Public	✓	onlyOwner
IUniswapV2Factory	Interface			
	feeTo	External		-
	feeToSetter	External		-
	getPair	External		-
	allPairs	External		-
	allPairsLength	External		-
	createPair	External	✓	-
	setFeeTo	External	✓	-
	setFeeToSetter	External	✓	-
IUniswapV2Pair	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	✓	-
IUniswapV2Router01	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	✓	-
DogeSuperman	Implementation	Context, IERC20, Ownable		
	<Constructor>	Public	✓	-
	name	Public		-
	symbol	Public		-
	decimals	Public		-
	totalSupply	Public		-
	balanceOf	Public		-
	transfer	Public	✓	-
	allowance	Public		-
	approve	Public	✓	-
	transferFrom	Public	✓	-
	increaseAllowance	Public	✓	-
	decreaseAllowance	Public	✓	-
	excludeFromFee	Public	✓	onlyOwner
	includeInFee	Public	✓	onlyOwner
	_set_Fees	External	✓	onlyOwner
	Wallet_Update_Dev	Public	✓	onlyOwner
	set_Swap_And_Liquify_Enabled	Public	✓	onlyOwner
	set_Number_Of_Transactions_Before_Liquify_Trigger	Public	✓	onlyOwner
	<Receive Ether>	External	Payable	-
	blacklist_Add_Wallets	External	✓	onlyOwner
	blacklist_Remove_Wallets	External	✓	onlyOwner

	blacklist_Switch	Public	✓	onlyOwner
	set_Transfers_Without_Fees	External	✓	onlyOwner
	set_Max_Transaction_Percent	External	✓	onlyOwner
	set_Max_Wallet_Percent	External	✓	onlyOwner
	removeAllFee	Private	✓	
	restoreAllFee	Private	✓	
	_approve	Private	✓	
	_transfer	Private	✓	
	sendToWallet	Private	✓	
	swapAndLiquify	Private	✓	lockTheSwap
	process_Tokens_Now	Public	✓	onlyOwner
	swapTokensForBNB	Private	✓	
	remove_Random_Tokens	Public	✓	onlyOwner
	set_New_Router_and_Make_Pair	Public	✓	onlyOwner
	set_New_Router_Address	Public	✓	onlyOwner
	set_New_Pair_Address	Public	✓	onlyOwner
	_tokenTransfer	Private	✓	
	_transferTokens	Private	✓	
	_getValues	Private		

Contract Flow



Domain Info

Domain Name	doges.fun
Registry Domain ID	D279581717-CNIC
Creation Date	2022-03-04T06:10:29+00:00
Updated Date	2022-03-04T06:10:30+00:00
Registry Expiry Date	2023-03-04T23:59:59+00:00
Registrar WHOIS Server	whois.godaddy.com
Registrar URL	https://www.godaddy.com/
Registrar	Go Daddy, LLC
Registrar IANA ID	146

The domain has been created 2 days 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 blacklisting contracts, stopping transactions and transferring funds to the team’s wallet. The maximum fee percentage that can be set is 24%. 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