



Smart Contract Audit

FOR
DOGEGPT
DATED : 12 May 23'



AUDIT SUMMARY

Project name - DOGEGPT

Date: 12 May, 2023

Scope of Audit- Audit Ace was consulted to conduct the smart contract audit of the solidity source codes.

Audit Status: Failed

Issues Found

Status	Critical	High	Medium	Low	Suggestion
Open	2	0	1	0	13
Acknowledged	0	0	0	0	0
Resolved	0	0	0	0	0



USED TOOLS

Tools:

1- Manual Review:

A line by line code review has been performed by audit ace team.

2- BSC Test Network: All tests were conducted on the BSC Test network, and each test has a corresponding transaction attached to it. These tests can be found in the "Functional Tests" section of the report.

3- Slither :

The code has undergone static analysis using Slither.

Testnet version:

The tests were performed using the contract deployed on the BSC Testnet, which can be found at the following address:

<https://testnet.bscscan.com/address/0xD93b59384d312c32097e1C2CbC4339c304C8A01C#code>



Token Information

Token Name : DOGEGPT

Token Symbol: DGPT

Decimals: 18

Token Supply: 280,000,000

Token Address:

0x25F17570111cdD390AFF5170ae6d622f35DcDBc1

Checksum:

7aa64444cf24e56bb0b112abdef9115c4fe415fd

Owner:

0xF8C97c6123cf976051A60DcC88ED250f10DBbC04

(at time of writing the audit)

Deployer:

0xF8C97c6123cf976051A60DcC88ED250f10DBbC04



TOKEN OVERVIEW

Fees:

Buy Fees: up to 100%

Sell Fees: up to 100%

Transfer Fees: up to 100%

Fees Privilege: Owner

Ownership: Owned

Minting: No mint function

Max Tx Amount/ Max Wallet Amount: No

Blacklist: No

Other Privileges: changing fee - including and
excluding form fee - changing max tx



AUDIT METHODOLOGY

The auditing process will follow a routine as special considerations by Auditace:

- Review of the specifications, sources, and instructions provided to Auditace to make sure the contract logic meets the intentions of the client without exposing the user's funds to risk.
- Manual review of the entire codebase by our experts, which is the process of reading source code line-by-line in an attempt to identify potential vulnerabilities.
- Specification comparison is the process of checking whether the code does what the specifications, sources, and instructions provided to Auditace describe.
- Test coverage analysis determines whether the test cases are covering the code and how much code is exercised when we run the test cases.
- Symbolic execution is analysing a program to determine what inputs cause each part of a program to execute.
- Reviewing the codebase to improve maintainability, security, and control based on the established industry and academic practices.

VULNERABILITY CHECKLIST



Return values of low-level calls



Gasless Send



Private modifier



Using block.timestamp



Multiple Sends



Re-entrancy



Using Suicide



Tautology or contradiction



Gas Limit and Loops



Timestamp Dependence



Address hardcoded



Revert/require functions



Exception Disorder



Use of tx.origin



Using inline assembly



Integer overflow/underflow



Divide before multiply



Dangerous strict equalities



Missing Zero Address Validation



Using SHA3



Compiler version not fixed



Using throw

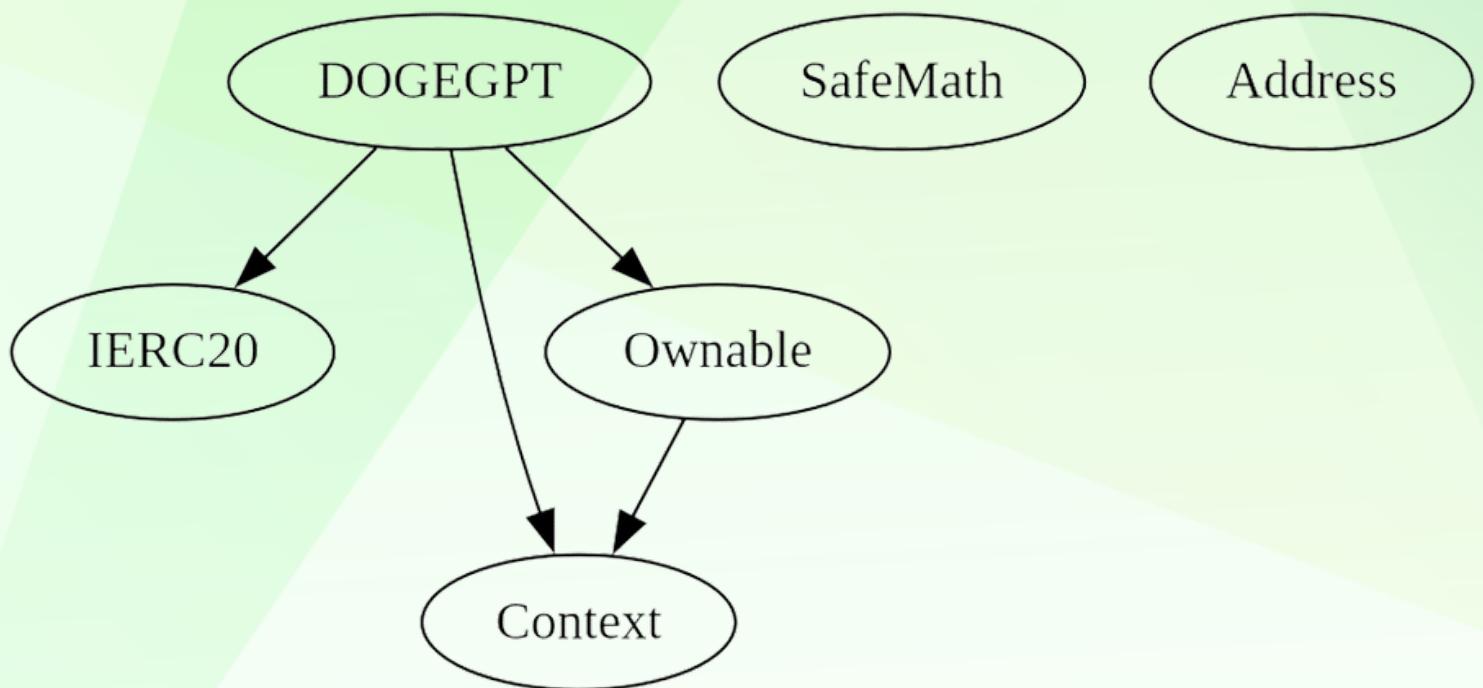
CLASSIFICATION OF RISK

Severity	Description
◆ Critical	These vulnerabilities could be exploited easily and can lead to asset loss, data loss, asset, or data manipulation. They should be fixed right away.
◆ High-Risk	A vulnerability that affects the desired outcome when using a contract, or provides the opportunity to use a contract in an unintended way.
◆ Medium-Risk	A vulnerability that could affect the desired outcome of executing the contract in a specific scenario.
◆ Low-Risk	A vulnerability that does not have a significant impact on possible scenarios for the use of the contract and is probably subjective.
◆ Gas Optimization / Suggestion	A vulnerability that has an informational character but is not affecting any of the code.

Findings

Severity	Found
◆ Critical	2
◆ High-Risk	0
◆ Medium-Risk	1
◆ Low-Risk	0
◆ Gas Optimization / Suggestions	0

INHERITANCE TREE





POINTS TO NOTE

- **Owner is able to set 100% fee on buy/sell/transfers**
- Owner is not able to blacklist an arbitrary address.
- **Owner is able to disable trades**
- Owner is able to set max buy/sell/transfer/hold amount to 0
- Owner is not able to mint new tokens

CONTRACT ASSESSMENT

Contract	Type	Bases				
			Function Name	**Visibility**	**Mutability**	**Modifiers**
	L					
	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 !
	SafeMath	Library				
	L	tryAdd	Internal			
	L	trySub	Internal			
	L	tryMul	Internal			
	L	tryDiv	Internal			
	L	tryMod	Internal			
	L	add	Internal			
	L	sub	Internal			
	L	mul	Internal			
	L	div	Internal			
	L	mod	Internal			
	L	sub	Internal			
	L	div	Internal			
	L	mod	Internal			
	Context	Implementation				
	L	_msgSender	Internal			
	L	_msgData	Internal			
	Address	Library				
	L	isContract	Internal			
	L	sendValue	Internal			
	L	functionCall	Internal			
	L	functionCall	Internal			
	L	functionCallWithValue	Internal			
	L	functionCallWithValue	Internal			

CONTRACT ASSESSMENT

```
| L | functionStaticCall | Internal | 🔒 | ||| |
| L | functionStaticCall | Internal | 🔒 | |||  
| L | functionDelegateCall | Internal | 🔒 | ⚡ | |||  
| L | functionDelegateCall | Internal | 🔒 | ⚡ | |||  
| L | _verifyCallResult | Private | 🔒 | |||  
|||||||  
| **Ownable** | Implementation | Context | |||  
| L | <Constructor> | Public ! | ⚡ | NO ! |  
| L | owner | Public ! | | NO ! |  
| L | renounceOwnership | Public ! | ⚡ | onlyOwner |  
| L | transferOwnership | Public ! | ⚡ | onlyOwner |  
|||||||  
| **IUniswapV2Factory** | Interface | |||  
| L | feeTo | External ! | | NO ! |  
| L | feeToSetter | External ! | | NO ! |  
| L | getPair | External ! | | NO ! |  
| L | allPairs | External ! | | NO ! |  
| L | allPairsLength | External ! | | NO ! |  
| L | createPair | External ! | ⚡ | NO ! |  
| L | setFeeTo | External ! | ⚡ | NO ! |  
| L | setFeeToSetter | External ! | ⚡ | NO ! |  
|||||||  
| **IUniswapV2Pair** | Interface | |||  
| L | name | External ! | | NO ! |  
| L | symbol | External ! | | NO ! |  
| L | decimals | External ! | | NO ! |  
| L | totalSupply | External ! | | NO ! |  
| L | balanceOf | External ! | | NO ! |  
| L | allowance | External ! | | NO ! |  
| L | approve | External ! | ⚡ | NO ! |  
| L | transfer | External ! | ⚡ | NO ! |  
| L | transferFrom | External ! | ⚡ | NO ! |  
| L | DOMAIN_SEPARATOR | External ! | | NO ! |  
| L | PERMIT_TYPEHASH | External ! | | NO ! |  
| L | nonces | External ! | | NO ! |
```



CONTRACT ASSESSMENT

```
| L | permit | External ! |  | NO ! | |
| L | MINIMUM_LIQUIDITY | External ! |  | NO ! |
| L | factory | External ! |  | NO ! |
| L | token0 | External ! |  | NO ! |
| L | token1 | External ! |  | NO ! |
| L | getReserves | External ! |  | NO ! |
| L | price0CumulativeLast | External ! |  | NO ! |
| L | price1CumulativeLast | External ! |  | NO ! |
| L | kLast | External ! |  | NO ! |
| L | mint | External ! |  | NO ! |
| L | burn | External ! |  | NO ! |
| L | swap | External ! |  | NO ! |
| L | skim | External ! |  | NO ! |
| L | sync | External ! |  | NO ! |
| L | initialize | External ! |  | NO ! |
|||||||
| **IUniswapV2Router01** | Interface | ||
| L | factory | External ! |  | NO ! |
| L | WETH | External ! |  | NO ! |
| L | addLiquidity | External ! |  | NO ! |
| L | addLiquidityETH | External ! |  | NO ! |
| L | removeLiquidity | External ! |  | NO ! |
| L | removeLiquidityETH | External ! |  | NO ! |
| L | removeLiquidityWithPermit | External ! |  | NO ! |
| L | removeLiquidityETHWithPermit | External ! |  | NO ! |
| L | swapExactTokensForTokens | External ! |  | NO ! |
| L | swapTokensForExactTokens | External ! |  | NO ! |
| L | swapExactETHForTokens | External ! |  | NO ! |
| L | swapTokensForExactETH | External ! |  | NO ! |
| L | swapExactTokensForETH | External ! |  | NO ! |
| L | swapETHForExactTokens | External ! |  | NO ! |
| L | quote | External ! |  | NO ! |
| L | getAmountOut | External ! |  | NO ! |
| L | getAmountIn | External ! |  | NO ! |
| L | getAmountsOut | External ! |  | NO ! |
| L | getAmountsIn | External ! |  | NO ! |
```

CONTRACT ASSESSMENT

|||||

```
|| **IUniswapV2Router02** | Interface | IUniswapV2Router01 |||
| L | removeLiquidityETHSupportingFeeOnTransferTokens | External ! | ○ | NO ! |
| L | removeLiquidityETHWithPermitSupportingFeeOnTransferTokens | External ! | ○
| NO ! |
| L | swapExactTokensForTokensSupportingFeeOnTransferTokens | External ! | ○
| NO ! |
| L | swapExactETHForTokensSupportingFeeOnTransferTokens | External ! | $ | NO !
|
| L | swapExactTokensForETHSupportingFeeOnTransferTokens | External ! | ○ | NO !
|
```

|||||

```
|| **DOGEGPT** | Implementation | Context, IERC20, Ownable |||
| L | <Constructor> | 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 | isExcludedFromReward | Public ! | | NO ! |
| L | totalFees | Public ! | | NO ! |
| L | deliver | Public ! | ○ | NO ! |
| L | reflectionFromToken | Public ! | | NO ! |
| L | tokenFromReflection | Public ! | | NO ! |
| L | excludeFromReward | Public ! | ○ | onlyOwner |
| L | includeInReward | External ! | ○ | onlyOwner |
| L | _transferBothExcluded | Private 🔒 | ○ | |
| L | excludeFromFee | Public ! | ○ | onlyOwner |
| L | includeInFee | Public ! | ○ | onlyOwner |
```

CONTRACT ASSESSMENT

		L setTaxFeePercent External	!		∅		onlyOwner	
		L setDevelopmentFeePercent External	!		∅		onlyOwner	
		L setLiquidityFeePercent External	!		∅		onlyOwner	
		L setMaxTxPercent External	!		∅		onlyOwner	
		L setSwapAndLiquifyEnabled Public	!		∅		onlyOwner	
		L <Receive Ether> External	!		Ξ		NO !	
		L _reflectFee Private	🔐		∅			
		L _getValues Private	🔐					
		L _getTValues Private	🔐					
		L _getRValues Private	🔐					
		L _getRate Private	🔐					
		L _getCurrentSupply Private	🔐					
		L _takeLiquidity Private	🔐		∅			
		L _takeDevelopment Private	🔐		∅			
		L calculateTaxFee Private	🔐					
		L calculateDevelopmentFee Private	🔐					
		L calculateLiquidityFee Private	🔐					
		L removeAllFee Private	🔐		∅			
		L restoreAllFee Private	🔐		∅			
		L isExcludedFromFee Public	!			NO !		
		L _approve Private	🔐		∅			
		L _transfer Private	🔐		∅			
		L swapAndLiquify Private	🔐		∅		lockTheSwap	
		L swapTokensForEth Private	🔐		∅			
		L addLiquidity Private	🔐		∅			
		L _tokenTransfer Private	🔐		∅			
		L _transferStandard Private	🔐		∅			
		L _transferToExcluded Private	🔐		∅			
		L _transferFromExcluded Private	🔐		∅			

Legend

	Symbol		Meaning	
-----	-----			
	∅		Function can modify state	
	Ξ		Function is payable	

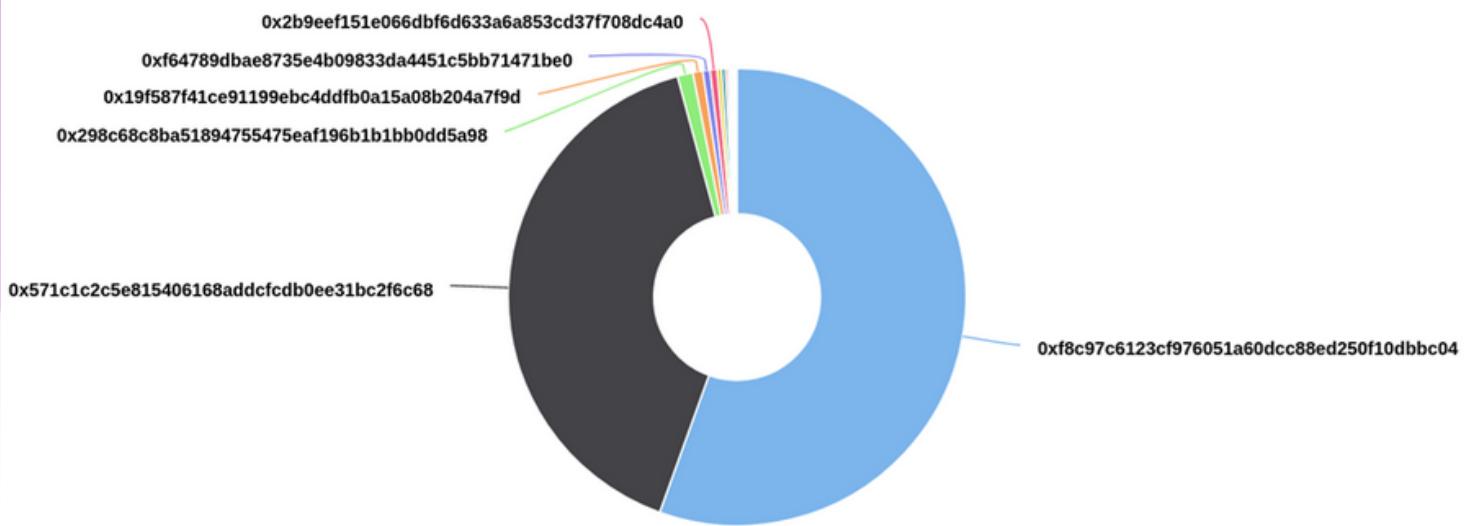
TOKENOMICS AT TIME OF AUDIT

 The top 100 holders collectively own 99.95% (99,950,147,750.23 Tokens) of DOGEGPT

 Token Total Supply: 100,000,000,000.00 Token | Total Token Holders: 17

DOGEGPT Top 100 Token Holders

Source: BscScan.com





STATIC ANALYSIS

```
Variable DOGEOPT._transferFromExcluded(address,address,uint256).rTransferAmount (contracts/Token.sol#1208) is too similar to DOGEOPT._getTValues(uint256).tTransferAmount (contracts/Token.sol#963-965)
Variable DOGEOPT._getRValues(uint256,uint256,uint256,uint256).rTransferAmount (contracts/Token.sol#980-982) is too similar to DOGEOPT._transferBothExcluded(address,address,uint256).tTransferAmount (contracts/Token.sol#873)
Variable DOGEOPT._transferToExcluded(address,address,uint256).rTransferAmount (contracts/Token.sol#1185) is too similar to DOGEOPT._transferToExcluded(address,address,uint256).tTransferAmount (contracts/Token.sol#1187)
Variable DOGEOPT._getRValues(uint256,uint256,uint256,uint256).rDevelopment (contracts/Token.sol#979) is too similar to DOGEOPT._transferStandard(address,address,uint256).tDevelopment (contracts/Token.sol#1168)
Variable DOGEOPT._takeDevelopment(uint256).rDevelopment (contracts/Token.sol#1016) is too similar to DOGEOPT._getTValues(uint256).tDevelopment (contracts/Token.sol#962)
Variable DOGEOPT._takeDevelopment(uint256).rDevelopment (contracts/Token.sol#1016) is too similar to DOGEOPT._transferToExcluded(address,address,uint256).tDevelopment (contracts/Token.sol#1199)
Variable DOGEOPT._takeDevelopment(uint256).rDevelopment (contracts/Token.sol#1016) is too similar to DOGEOPT._getRValues(uint256,uint256,uint256,uint256).tDevelopment (contracts/Token.sol#973)
Variable DOGEOPT._takeDevelopment(uint256).rDevelopment (contracts/Token.sol#1016) is too similar to DOGEOPT._transferStandard(address,address,uint256).tDevelopment (contracts/Token.sol#1168)
Variable DOGEOPT._getValues(uint256).rTransferAmount (contracts/Token.sol#939) is too similar to DOGEOPT._transferToExcluded(address,address,uint256).tTransferAmount (contracts/Token.sol#1187)
Variable DOGEOPT._transferBothExcluded(address,address,uint256).rTransferAmount (contracts/Token.sol#871) is too similar to DOGEOPT._transferStandard(address,address,uint256).tTransferAmount (contracts/Token.sol#1165)
Variable DOGEOPT._getRValues(uint256,uint256,uint256,uint256).rTransferAmount (contracts/Token.sol#980-982) is too similar to DOGEOPT._transferStandard(address,address,uint256).tTransferAmount (contracts/Token.sol#1165)
Variable DOGEOPT._getValues(uint256).rTransferAmount (contracts/Token.sol#939) is too similar to DOGEOPT._transferStandard(address,address,uint256).tTransferAmount (contracts/Token.sol#1165)
Variable DOGEOPT._transferFromExcluded(address,address,uint256).rTransferAmount (contracts/Token.sol#1208) is too similar to DOGEOPT._transferToExcluded(address,address,uint256).tTransferAmount (contracts/Token.sol#1187)
Variable DOGEOPT._getValues(uint256).rTransferAmount (contracts/Token.sol#939) is too similar to DOGEOPT._transferBothExcluded(address,address,uint256).tTransferAmount (contracts/Token.sol#873)
Variable DOGEOPT._transferBothExcluded(address,address,uint256).rTransferAmount (contracts/Token.sol#871) is too similar to DOGEOPT._transferFromExcluded(address,address,uint256).tTransferAmount (contracts/Token.sol#1210)
Variable DOGEOPT._getValues(uint256).rTransferAmount (contracts/Token.sol#939) is too similar to DOGEOPT._getValues(uint256).tTransferAmount (contracts/Token.sol#934)
Variable DOGEOPT._transferFromExcluded(address,address,uint256).rTransferAmount (contracts/Token.sol#1208) is too similar to DOGEOPT._transferStandard(address,address,uint256).tTransferAmount (contracts/Token.sol#1165)
Variable DOGEOPT._getValues(uint256).rTransferAmount (contracts/Token.sol#939) is too similar to DOGEOPT._transferFromExcluded(address,address,uint256).tTransferAmount (contracts/Token.sol#1210)
Variable DOGEOPT._transferBothExcluded(address,address,uint256).rTransferAmount (contracts/Token.sol#871) is too similar to DOGEOPT._transferToExcluded(address,address,uint256).tTransferAmount (contracts/Token.sol#1187)
Variable DOGEOPT._getRValues(uint256,uint256,uint256,uint256).rTransferAmount (contracts/Token.sol#980-982) is too similar to DOGEOPT._transferToExcluded(address,address,uint256).tTransferAmount (contracts/Token.sol#1187)
Variable DOGEOPT._getValues(uint256).rTransferAmount (contracts/Token.sol#939) is too similar to DOGEOPT._getTValues(uint256).tTransferAmount (contracts/Token.sol#963-965)
Variable DOGEOPT._transferToExcluded(address,address,uint256).rTransferAmount (contracts/Token.sol#1185) is too similar to DOGEOPT._transferStandard(address,address,uint256).tTransferAmount (contracts/Token.sol#1165)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#variable-names-too-similar

DOGEOPT.slitherConstructorVariables() (contracts/Token.sol#652-1223) uses literals with too many digits:
- _Total = 100000000000 * 10 ** 18 (contracts/Token.sol#664)
DOGEOPT.slitherConstructorVariables() (contracts/Token.sol#652-1223) uses literals with too many digits:
- maxTxAmount = 10000000000 * 10 ** 18 (contracts/Token.sol#681)
DOGEOPT.slitherConstructorVariables() (contracts/Token.sol#652-1223) uses literals with too many digits:
- numTokensSellToAddToLiquidity = 1000000000 * 10 ** 18 (contracts/Token.sol#682)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#too-many-digits

DOGEOPT.decimals (contracts/Token.sol#669) should be constant
DOGEOPT.developmentWalletAddress (contracts/Token.sol#661-662) should be constant
DOGEOPT.name (contracts/Token.sol#667) should be constant
DOGEOPT.symbol (contracts/Token.sol#668) should be constant
DOGEOPT._Total (contracts/Token.sol#664) should be constant
DOGEOPT.numTokensSellToAddToLiquidity (contracts/Token.sol#682) should be constant
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#state-variables-that-could-be-declared-constant
/contracts/Token.sol analyzed (10 contracts, 94 detectors, 129 results)
```

**Result => A static analysis of contract's source code has been performed using slither,
No major issues were found in the output**



FUNCTIONAL TESTING

Router (PCS V2):

0xD99D1c33F9fC3444f8101754aBC46c52416550D1

All the functionalities have been tested, no issues were found

1- Adding liquidity (**passed**):

<https://testnet.bscscan.com/tx/0xbc021debbca450ad80b6ed870bb58edac9dfc3bc388782c2ad9d6bb0a8685fa5>

2- Buying when excluded (0% tax) (**passed**):

<https://testnet.bscscan.com/tx/0xa10ec991426d48b635047ab0b69e938655dfc03577462d18604c9be5210b460a>

3- Selling when excluded (0% tax) (**passed**):

<https://testnet.bscscan.com/tx/0x5fafcd5c326c399db46895736f263999c8e1093e702878d499b4fb42e6df77b8>

4- Transferring when excluded from fees (0% tax) (**passed**):

<https://testnet.bscscan.com/tx/0xc0831bd756f0127b4b255ec5dc3c6876c7bd1a6979a820a582f05366ff94ae24>

5- Buying when not excluded from fees (up to 100% tax) (**passed**):

<https://testnet.bscscan.com/tx/0x870285e6cc5b0be1a83a14db08d595dcf3b671c6c4064f0e85ba2ed8cbbfae1a>

6- Selling when not excluded from fees (up to 100% tax) (**passed**):

<https://testnet.bscscan.com/tx/0xda7823face7f143cd7e8786b0079773eef83d56c7f3bb1c9eb13705deef17015>



FUNCTIONAL TESTING

7- Transferring when not excluded from fees (up to 100% tax)(passed):

<https://testnet.bscscan.com/tx/0x0ba4c9dd61fd93075179a8dc0e5ffff55cd4150e04d8089456defd318f8c6fff>

8- Auto-liquidity(passed):

<https://testnet.bscscan.com/tx/0x1b49cc908ad5757c217c8ea33eb45bb63dda3d5b64bfdae85e4f2d06c4aa71ff>



MANUAL TESTING

Centralization – Excessive fees

Status: Not Resolved

Severity: Critical

Overview:

Owner of the contract is able to set fees up to 100% for buy/sell/transfer tax.

```
function setTaxFeePercent(uint256 taxFee) external onlyOwner {  
    _taxFee = taxFee;  
}
```

```
function setDevelopmentFeePercent(  
    uint256 developmentFee  
) external onlyOwner {  
    _developmentFee = developmentFee;  
}
```

```
function setLiquidityFeePercent(uint256 liquidityFee) external onlyOwner {  
    _liquidityFee = liquidityFee;  
}
```

Suggestion:

Ensure that sum of fees is always less than 12.5% (25% total tax on buy+sell)

```
function setTaxFeePercent(uint256 taxFee) external onlyOwner {  
    _taxFee = taxFee;  
    require(_taxFee + _developmentFee + _liquidityFee < 125, "can not set fees higher than  
12.5%")  
}
```

```
function setDevelopmentFeePercent(  
    uint256 developmentFee  
) external onlyOwner {  
    _developmentFee = developmentFee;  
    require(_taxFee + _developmentFee + _liquidityFee < 125, "can not set fees higher than  
12.5%")  
}
```

```
function setLiquidityFeePercent(uint256 liquidityFee) external onlyOwner {  
    _liquidityFee = liquidityFee;  
    require(_taxFee + _developmentFee + _liquidityFee < 125, "can not set fees higher than  
12.5%")  
}
```



MANUAL TESTING

Centralization – Max buy/sell/transfer

Status: Not Resolved

Severity: Critical

Overview:

Owner of the contract is able to set max buy/sell/transfer amount to 0, disabling trades for all holders except owner.

```
function setMaxTxPercent(uint256 maxTxPercent) external onlyOwner {  
    _maxTxAmount = _tTotal.mul(maxTxPercent).div(10 ** 3);  
}
```

Suggestion:

Ensure that maxTxAmount is always higher than 0.1% of total supply

```
function setMaxTxPercent(uint256 maxTxPercent) external onlyOwner {  
    _maxTxAmount = _tTotal.mul(maxTxPercent).div(10 ** 3);  
    require(_maxTxAmount >= _tTotal / 1000, "can not set max tx less than 0.1% of supply")  
}
```



MANUAL TESTING

Centralization – LP tokens going to an EOA

Severity: **Medium**

function: addLiquidity

Status: Not Resolved

Overview:

LP tokens generated from auto-liquidity are sent to an EOA account, this accumulated tokens may be used by a malicious owner to remove a portion of tokens from the pool

```
function addLiquidity(uint256 tokenAmount, uint256 ethAmount) private {
    _approve(address(this), address(uniswapV2Router), tokenAmount);
    uniswapV2Router.addLiquidityETH{value: ethAmount}(
        address(this),
        tokenAmount,
        0,
        0,
        owner(),
        block.timestamp
    );
}
```

Suggestion

To mitigate this Centralization issue:

- Burn the new LP tokens

or

- Lock the new LP tokens



MANUAL TESTING

Informational – Stuck tokens and BNB

Status: Not Resolved

Overview:

Currently there is no way for owner to withdraw stuck ERC20 tokens or BNBs inside the contract.

Suggestion:

Create a function to be able to withdraw stuck ERC20 tokens as well as BNB from the contract



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. 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 Auditace 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 Auditace 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 Auditace team disclaims any liability for the resulting losses.



ABOUT AUDITACE

We specialize in providing thorough and reliable audits for Web3 projects. With a team of experienced professionals, we use cutting-edge technology and rigorous methodologies to evaluate the security and integrity of blockchain systems. We are committed to helping our clients ensure the safety and transparency of their digital assets and transactions.



<https://auditace.tech/>



https://t.me/Audit_Ace



https://twitter.com/auditace_



<https://github.com/Audit-Ace>
