



Smart Contract Audit

FOR
Bull Doge

DATED : 12 Apr 23'



AUDIT SUMMARY

Project name - Bull Doge

Date: 12 April, 2023

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

Audit Status: Passed

Issues Found

Status	Critical	High	Medium	Low	Suggestion
Open	0	0	0	0	0
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 done on BSC Test network, each test has its transaction has attached to it.

3- Slither : Static Analysis

Testnet Link: all tests were done using this contract, tests are done on BSC Testnet

<https://testnet.bscscan.com/token/0xF986c93d8cB1136F95d86c741f984C346fCcA5a>



Token Information

Token Name : Bull Doge

Token Symbol: bDoge

Decimals: 18

Token Supply: 1,000,000,000,000,000

Token Address:

0xc32410f3fE5fF77c2880a4F31eAA677C25e8a87c

Checksum:

d8f25894d1b2a6d13080fc5bc09d9672a22495e8

Owner:

0xC2697b1c67C421D9fEDB742dd1F7Ec9f14E370F3



TOKEN OVERVIEW

Fees:

Buy Fees: upto 25%

Sell Fees: upto 25%

Transfer Fees: upto 25%

Fees Privilege: Owner

Ownership : Owned

Minting: No mint function

Max Tx Amount/ Max Wallet Amount: No

Blacklist: No

Other Privileges: including and excluding form fee - changing swap threshold - modifying fees



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



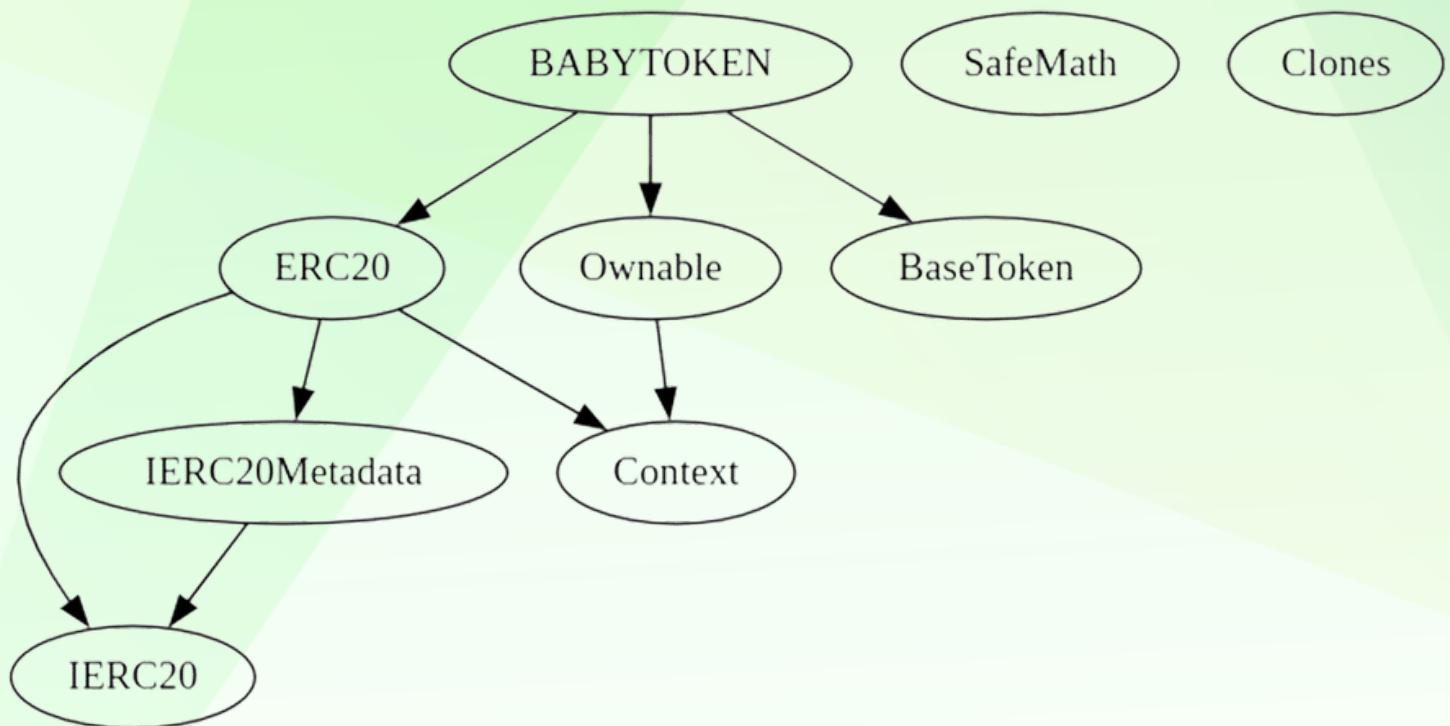
CLASSIFICATION OF RISK

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	0
◆ High-Risk	0
◆ Medium-Risk	0
◆ Low-Risk	0
◆ Gas Optimization / Suggestions	0

INHERITANCE TREE





POINTS TO NOTE

- Owner is able to change buy/sell/transfer fees but sum of fees can not exceed 25%
- Contract is recognized as proxy in bscscan, however this is because dividend distributor is minimal proxy contract which is not upgradeable currently
- Owner is not able to set max buy/sell/transfer/hold amount
- Owner is not able to blacklist an arbitrary wallet
- Owner is not able to disable trades
- Owner is not able to mint new tokens
- Token has pinksale anti-bot



OVERVIEW

This token has three different types of fees, which are used for marketing, maintaining liquidity, and providing rewards to holders. The fees collected are converted to a stablecoin called BUSD and sent to a marketing wallet. Another part of the fee is used to maintain liquidity and is sent to a dividend tracker to provide BUSD rewards to holders. Rewards are distributed to holders every time a buy, sell or transfer occurs, and can also be manually claimed from the BscScan website.



TOKEN DISTRIBUTION

it should be noted that the owner currently holds 100% of the total supply. However, information about the distribution of these tokens is not available, and it is recommended that investors exercise caution when considering this aspect.

CONTRACT ASSESSMENT

Contract	Type	Bases			
	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 !	
	IERC20Metadata	Interface	IERC20		
L	name	External !	NO !		
L	symbol	External !	NO !		
L	decimals	External !	NO !		
	Context	Implementation			
L	_msgSender	Internal 			
L	_msgData	Internal 			
	ERC20	Implementation	Context, IERC20, IERC20Metadata		
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	_transfer	Internal 			
L	_mint	Internal 			
L	_burn	Internal 			
L	_approve	Internal 			
L	_beforeTokenTransfer	Internal 			
L	_afterTokenTransfer	Internal 			
	Ownable	Implementation	Context		

CONTRACT ASSESSMENT

```
| L | <Constructor> | Public ! | ○ | NO! | |
| L | owner | Public ! | | NO! |
| L | renounceOwnership | Public ! | ○ | onlyOwner |
| L | transferOwnership | Public ! | ○ | onlyOwner |
| L | _setOwner | Private 🔒 | ○ | |
|||||||
| **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 🔒 | |||
|||||||
| **Clones** | Library | ||
| L | clone | Internal 🔒 | ○ | ||
| L | cloneDeterministic | Internal 🔒 | ○ | ||
| L | predictDeterministicAddress | Internal 🔒 | |||
| L | predictDeterministicAddress | Internal 🔒 | |||
|||||||
| **Address** | Library | ||
| L | isContract | Internal 🔒 | |||
| L | sendValue | Internal 🔒 | ○ | ||
| L | functionCall | Internal 🔒 | ○ | ||
| L | functionCall | Internal 🔒 | ○ | ||
| L | functionCallWithValue | Internal 🔒 | ○ | ||
| L | functionCallWithValue | Internal 🔒 | ○ | ||
| L | functionStaticCall | Internal 🔒 | |||
| L | functionStaticCall | Internal 🔒 | |||
| L | functionDelegateCall | Internal 🔒 | ○ | ||
| L | functionDelegateCall | Internal 🔒 | ○ | ||
| L | verifyCallResult | Internal 🔒 | |||
|||||||
| **IUniswapV2Factory** | Interface | ||
| L | feeTo | External ! | | NO! |
```

CONTRACT ASSESSMENT

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!
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!
IUniswapV2Router02 Interface IUniswapV2Router01
L removeLiquidityETHSupportingFeeOnTransferTokens External ! NO!
L removeLiquidityETHWithPermitSupportingFeeOnTransferTokens External ! NO!
L swapExactTokensForTokensSupportingFeeOnTransferTokens External ! NO!
L swapExactETHForTokensSupportingFeeOnTransferTokens External ! NO!
L swapExactTokensForETHSupportingFeeOnTransferTokens External ! NO!
IPinkAntiBot Interface
L setTokenOwner External ! NO!
L onPreTransferCheck External ! NO!
IERC20Upgradeable Interface
L totalSupply External ! NO!

CONTRACT ASSESSMENT

```
| L | balanceOf | External ! | | NO! | | |
| L | transfer | External ! | |  | | NO! |
| L | allowance | External ! | | NO! |
| L | approve | External ! | |  | | NO! |
| L | transferFrom | External ! | |  | | NO! |
|||||||
| **IERC20MetadataUpgradeable** | Interface | IERC20Upgradeable |||
| L | name | External ! | | NO! |
| L | symbol | External ! | | NO! |
| L | decimals | External ! | | NO! |
|||||||
| **Initializable** | Implementation | ||
|||||||
| **ContextUpgradeable** | Implementation | Initializable |||
| L | __Context_init | Internal  | |  | | initializer |
| L | __Context_init_unchained | Internal  | |  | | initializer |
| L | _msgSender | Internal  | | |
| L | _msgData | Internal  | | |
|||||||
| **ERC20Upgradeable** | Implementation | Initializable, ContextUpgradeable, IERC20Upgradeable, IERC20MetadataUpgradeable |||
| L | __ERC20_init | Internal  | |  | | initializer |
| L | __ERC20_init_unchained | Internal  | |  | | initializer |
| 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 | _transfer | Internal  | |  | | |
| L | _mint | Internal  | |  | | |
| L | _burn | Internal  | |  | | |
| L | _approve | Internal  | |  | | |
| L | _beforeTokenTransfer | Internal  | |  | | |
| L | _afterTokenTransfer | Internal  | |  | | |
|||||||
| **OwnableUpgradeable** | Implementation | Initializable, ContextUpgradeable |||
```



CONTRACT ASSESSMENT

```
| L | __Ownable_init | Internal 🔒 | ○ | initializer | |
| L | __Ownable_init_unchained | Internal 🔒 | ○ | initializer |
| L | owner | Public ! | | NO! |
| L | renounceOwnership | Public ! | ○ | onlyOwner |
| L | transferOwnership | Public ! | ○ | onlyOwner |
| L | _setOwner | Private 🔒 | ○ | |
|||||||
| **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! |
| 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! |
|||||||
| **SafeMathInt** | Library | ||
| L | mul | Internal 🔒 | | |
| L | div | Internal 🔒 | | |
| L | sub | Internal 🔒 | | |
| L | add | Internal 🔒 | | |
```



CONTRACT ASSESSMENT

```

| L | abs | Internal 🔒 | ||| |
| L | toUInt256Safe | Internal 🔒 | |||
||||||
| **SafeMathUint** | Library | |||
| L | toInt256Safe | Internal 🔒 | |||
||||||
| **IterableMapping** | Library | |||
| L | get | Internal 🔒 | |||
| L | getIndexOfKey | Internal 🔒 | |||
| L | getKeyAtIndex | Internal 🔒 | |||
| L | size | Internal 🔒 | |||
| L | set | Internal 🔒 | ⚡️ | |||
| L | remove | Internal 🔒 | ⚡️ | |||
||||||
| **DividendPayingTokenInterface** | Interface | |||
| L | dividendOf | External ! | | NO! |
| L | withdrawDividend | External ! | ⚡️ | NO! |
||||||
| **DividendPayingTokenOptionalInterface** | Interface | |||
| L | withdrawableDividendOf | External ! | | NO! |
| L | withdrawnDividendOf | External ! | | NO! |
| L | accumulativeDividendOf | External ! | | NO! |
||||||
| **DividendPayingToken** | Implementation | ERC20Upgradeable, OwnableUpgradeable,
DividendPayingTokenInterface, DividendPayingTokenOptionalInterface |||
| L | __DividendPayingToken_init | Internal 🔒 | ⚡️ | initializer | |
| L | distributeCAKEDividends | Public ! | ⚡️ | onlyOwner |
| L | withdrawDividend | Public ! | ⚡️ | NO! |
| L | _withdrawDividendOfUser | Internal 🔒 | ⚡️ | |
| L | dividendOf | Public ! | | NO! |
| L | withdrawableDividendOf | Public ! | | NO! |
| L | withdrawnDividendOf | Public ! | | NO! |
| L | accumulativeDividendOf | Public ! | | NO! |
| L | _transfer | Internal 🔒 | ⚡️ | |
| L | _mint | Internal 🔒 | ⚡️ | |
| L | _burn | Internal 🔒 | ⚡️ | |
| L | _setBalance | Internal 🔒 | ⚡️ | |
||||||
| **BABYTOKENDividendTracker** | Implementation | OwnableUpgradeable, DividendPayingToken |||
| L | initialize | External ! | ⚡️ | initializer |
| L | _transfer | Internal 🔒 | |||
| L | withdrawDividend | Public ! | | NO! |

```



CONTRACT ASSESSMENT

```
| L | excludeFromDividends | External ! | ○ | onlyOwner | |
| L | isExcludedFromDividends | Public ! | | NO! |
| L | updateClaimWait | External ! | ○ | onlyOwner |
| L | updateMinimumTokenBalanceForDividends | External ! | ○ | onlyOwner |
| L | getLastProcessedIndex | External ! | | NO! |
| L | getNumberOfTokenHolders | External ! | | NO! |
| L | getAccount | Public ! | | NO! |
| L | getAccountAtIndex | Public ! | | NO! |
| L | canAutoClaim | Private 🔒 | | |
| L | setBalance | External ! | ○ | onlyOwner |
| L | process | Public ! | ○ | NO! |
| L | processAccount | Public ! | ○ | onlyOwner |
|||||||
**BaseToken** | Implementation | ||
|||||||
**AntiBotBABYTOKEN** | Implementation | ERC20, Ownable, BaseToken | ||
| L | <Constructor> | Public ! | 📄 | ERC20 |
| L | <Receive Ether> | External ! | 📄 | NO! |
| L | setSwapTokensAtAmount | External ! | ○ | onlyOwner |
| L | excludeFromFees | External ! | ○ | onlyOwner |
| L | excludeMultipleAccountsFromFees | External ! | ○ | onlyOwner |
| L | setMarketingWallet | External ! | ○ | onlyOwner |
| L | setTokenRewardsFee | External ! | ○ | onlyOwner |
| L | setLiquiditFee | External ! | ○ | onlyOwner |
| L | setMarketingFee | External ! | ○ | onlyOwner |
| L | _setAutomatedMarketMakerPair | Private 🔒 | ○ | |
| L | updateGasForProcessing | Public ! | ○ | onlyOwner |
| L | updateClaimWait | External ! | ○ | onlyOwner |
| L | getClaimWait | External ! | | NO! |
| L | updateMinimumTokenBalanceForDividends | External ! | ○ | onlyOwner |
| L | getMinimumTokenBalanceForDividends | External ! | | NO! |
| L | getTotalDividendsDistributed | External ! | | NO! |
| L | isExcludedFromFees | Public ! | | NO! |
| L | withdrawableDividendOf | Public ! | | NO! |
| L | dividendTokenBalanceOf | Public ! | | NO! |
| L | excludeFromDividends | External ! | ○ | onlyOwner |
| L | isExcludedFromDividends | Public ! | | NO! |
| L | getAccountDividendsInfo | External ! | | NO! |
| L | getAccountDividendsInfoAtIndex | External ! | | NO! |
| L | processDividendTracker | External ! | ○ | NO! |
| L | claim | External ! | ○ | NO! |
```

CONTRACT ASSESSMENT

L getLastProcessedIndex External ! NO!
L getNumberOfDividendTokenHolders External ! NO!
L _transfer Internal 🔒 🔞
L swapAndSendToFee Private 🔒 🔞
L swapAndLiquify Private 🔒 🔞
L swapTokensForEth Private 🔒 🔞
L swapTokensForCake Private 🔒 🔞
L addLiquidity Private 🔒 🔞
L swapAndSendDividends Private 🔒 🔞
DividendPayingToken Implementation ERC20, DividendPayingTokenInterface, Ownable
L <Constructor> Public ! 🔞 ERC20
L <Receive Ether> External ! 💸 NO!
L distributeDividends Public ! 💸 NO!
L _withdrawDividendOfUser Internal 🔒 🔞
L setRewardToken External ! 🔞 onlyOwner
L swapBnbForCustomToken Internal 🔒 🔞
L dividendOf Public ! NO!
L withdrawableDividendOf Public ! NO!
L withdrawnDividendOf Public ! NO!
L accumulativeDividendOf Public ! NO!
L _transfer Internal 🔒 🔞
L _tokengeneration Internal 🔒 🔞
L _burn Internal 🔒 🔞
L _setBalance Internal 🔒 🔞
ERC20 Implementation Context, IERC20, IERC20Metadata
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 _transfer Internal 🔒 🔞
L _tokengeneration Internal 🔒 🔞



CONTRACT ASSESSMENT

```
| L | _burn | Internal 🔒 | 🔴 | | |
| L | _approve | Internal 🔒 | 🔴 | | |
| L | _beforeTokenTransfer | Internal 🔒 | 🔴 | | |
|||||||
| **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! | |
|||||||
| **IERC20Metadata** | Interface | IERC20 |||
| L | name | External ! | | NO! | |
| L | symbol | External ! | | NO! | |
| L | decimals | External ! | | NO! | |
|||||||
| **Context** | Implementation | |||
| L | _msgSender | Internal 🔒 | | |
| L | _msgData | Internal 🔒 | | |
|||||||
| **SafeMath** | Library | |||
| L | add | Internal 🔒 | | |
| L | sub | Internal 🔒 | | |
| L | sub | Internal 🔒 | | |
| L | mul | Internal 🔒 | | |
| L | div | Internal 🔒 | | |
| L | div | Internal 🔒 | | |
| L | mod | Internal 🔒 | | |
| L | mod | Internal 🔒 | | |
|||||||
| **SafeMathInt** | Library | |||
| L | mul | Internal 🔒 | | |
| L | div | Internal 🔒 | | |
| L | sub | Internal 🔒 | | |
| L | add | Internal 🔒 | | |
| L | abs | Internal 🔒 | | |
| L | toUint256Safe | Internal 🔒 | | |
|||||||
| **SafeMathUint** | Library | |||
| L | toInt256Safe | Internal 🔒 | | |
|||||||
```

CONTRACT ASSESSMENT

```

| **DividendPayingTokenInterface** | Interface | ||| |
| L | dividendOf | External ! | | NO! |
| L | distributeDividends | External ! | |  | NO! |
| L | withdrawableDividendOf | External ! | | NO! |
| L | withdrawnDividendOf | External ! | | NO! |
| L | accumulativeDividendOf | External ! | | NO! |
|||||||
| **Ownable** | Implementation | Context | |||
| L | <Constructor> | Public ! | |  | NO! |
| L | owner | Public ! | | NO! |
| L | renounceOwnership | Public ! | |  | onlyOwner |
| L | transferOwnership | Public ! | |  | onlyOwner |
|||||||
| **IPair** | Interface | |||
| L | sync | External ! | |  | NO! |
|||||||
| **IFactory** | Interface | |||
| L | createPair | External ! | |  | NO! |
| L | getPair | External ! | | NO! |
|||||||
| **IRouter** | Interface | |||
| L | factory | External ! | | NO! |
| L | WETH | External ! | | NO! |
| L | addLiquidityETH | External ! | |  | NO! |
| L | swapExactTokensForTokensSupportingFeeOnTransferTokens | External ! | |  | NO! |
| L | swapExactETHForTokens | External ! | |  | NO! |
| L | swapExactTokensForETHSupportingFeeOnTransferTokens | External ! | |  | NO! |
|||||||
| **IterableMapping** | Library | |||
| L | get | Internal  | | |
| L | getIndexOfKey | Internal  | | |
| L | getKeyAtIndex | Internal  | | |
| L | size | Internal  | | |
| L | set | Internal  | | |
| L | remove | Internal  | | |

```

CONTRACT ASSESSMENT

Legend

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



STATIC ANALYSIS

**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/0xcb21fb25a5fd96662222a652e1bf7e167f7771aa49ea52fcaeaf575c44b82147>

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

<https://testnet.bscscan.com/tx/0x35b0edfaeacb28c3d84043917e0af35e56020dfd4924c253912f48f60566c54f>

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

<https://testnet.bscscan.com/tx/0xbfd55f86ae8bf06c319fcf4b4467d93789f6c351d16b14c2f1eb18c4adfa2226>

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

<https://testnet.bscscan.com/tx/0x3a791c714b245e6caf42b503b23e0a71bd3e3ba7dc616cf07a8d4e5e62d62fd>

5- Buying when not excluded (upto 25% tax) (passed):

<https://testnet.bscscan.com/tx/0x2ce71e2ea7cae376a966e0cc1ab685cd790e14c05c1d48f52af62e3a24079fc1>

6- Selling when not excluded (upto 25% tax) (passed):

<https://testnet.bscscan.com/tx/0xd5413e12b0fc45ac79a855f2f5ca3cde6c2ade92b86d3fcfd58e474857a89c8b>



FUNCTIONAL TESTING

7- Transferring when not excluded (upto 25% tax) (passed):

<https://testnet.bscscan.com/tx/0xb1b01a547bea2c2292339bd5fd942a83c402c82e9689a9c9f4fa3d1b40f87030>

8- Internal swap (passed):

As can seen in this transaction, marketing wallet received BUSD

<https://testnet.bscscan.com/address/0xa2da001d772453f7a1d520148663462ebcbd79b4#tokentxns>

9- Auto Liquidity (passed):

<https://testnet.bscscan.com/tx/0xd5413e12b0fc45ac79a855f2f5ca3cde6c2ade92b86d3fcfd58e474857a89c8b>

distributor is sending BUSD to holders (auto distribution)

10- Distribution of rewards (passed):

<https://testnet.bscscan.com/token/0x2a07846012568d2f400b05f>



MANUAL TESTING

Contract Generated by Pinksale

No Issues Found

Social Media Overview

Here are the Social Media Accounts of
Bull Doge



<https://medium.com/@bulldogeproject>



<https://twitter.com/BullDOGE459549>



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>
