auditace.tech

AuditAce

FROM INCEPTION TO SUCCESS

Smart Contract Audit

FOR

Bull Doge

DATED : 05 Apr 23'

AUDIT SUMMARY

Project name – Bull Doge

Date: 05 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	2	1	0	0
Acknowledged	0	0	0	0	0
Resolved	0	1	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/0x9d04441d16135ef3ff9e93c847dc52bcbeaaeab4#balances

Token Information

Token Name : Bull Doge

Token Symbol: BDoge

Decimals: 9

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

Token Address:
0xFa6F45d567b0641ED967922E025A9765d3443d11

Checksum:
ed6d88fd3c00c14d53967e23ca7a6ff30db504e0

Owner:
0x2cF7adEe50F96f9078c042A435a0bA00e1Dd4151

TOKEN OVERVIEW

Fees:

Buy Fees: upto 12%

Sell Fees: upto 12%

Transfer Fees: 0%

Fees Privilige: Owner

Ownership : Owned

Minting: No mint function

Max Tx Amount/ Max Wallet Amount: No

Blacklist: No

Other Priviliges: including in fees, exluding from fees, modifying buy/sell 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 isexercised 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
- ✓ Private modifier
- ✓ Multiple Sends
- ✓ Using Suicide
- ✓ Gas Limitand Loops
- ✓ Address hardcoded
- ✓ Exception Disorder
- ✓ Using inline assembly
- ✓ Divide before multiply
- ✓ Missing Zero Address Validation
- ✓ Compiler version not fixed
- ✓ **Gasless Send**
- ✓ Using block.timestamp
- ✓ Re-entrancy
- ✓ Tautology or contradiction
- ✓ Timestamp Dependence
- ✓ Revert/require functions
- ✓ Use of tx.origin
- ✓ Integer overflow/underflow
- ✓ Dangerous strict equalities
- ✓ Using SHA3
- ✓ Using throw

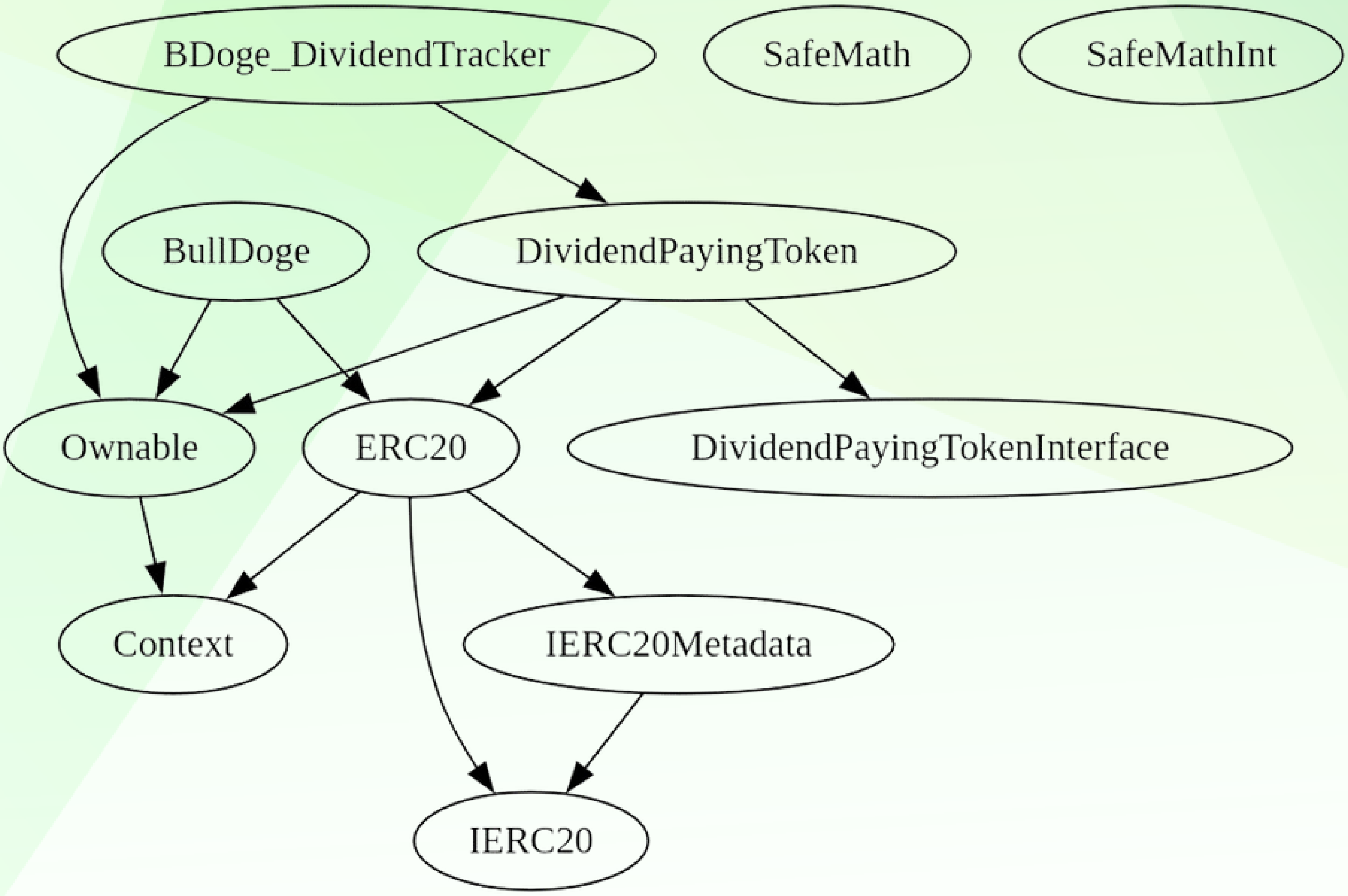
CLASSIFICATION OF RISK

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	2 (1 Resolved)
Medium-Risk	1
Low-Risk	0
Gas Optimization / Suggestions	0

INHERITANCE TREE



POINTS TO NOTE

- Owner is able to set buy/sell fees up to 12% each
- Owner is not able to set transfer fees (0% always)
- 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

CONTRACT ASSESMENT

Contract	Type	Bases		
└	**Function Name**	**Visibility**	**Mutability**	**Modifiers**
Context	Implementation			
└	_msgSender	Internal 🔒		
└	_msgData	Internal 🔒		
IERC20	Interface			
└	totalSupply	External ❗		NO❗
└	balanceOf	External ❗		NO❗
└	transfer	External ❗	🛑	NO❗
└	allowance	External ❗		NO❗
└	approve	External ❗	🛑	NO❗
└	transferFrom	External ❗	🛑	NO❗
IERC20Metadata	Interface	IERC20		
└	name	External ❗		NO❗
└	symbol	External ❗		NO❗
└	decimals	External ❗		NO❗
ERC20	Implementation	Context, IERC20, IERC20Metadata		
└	<Constructor>	Public ❗	🛑	NO❗
└	name	Public ❗		NO❗
└	symbol	Public ❗		NO❗
└	decimals	Public ❗		NO❗
└	totalSupply	Public ❗		NO❗
└	balanceOf	Public ❗		NO❗
└	transfer	Public ❗	🛑	NO❗
└	allowance	Public ❗		NO❗
└	approve	Public ❗	🛑	NO❗
└	transferFrom	Public ❗	🛑	NO❗
└	increaseAllowance	Public ❗	🛑	NO❗
└	decreaseAllowance	Public ❗	🛑	NO❗
└	_transfer	Internal 🔒	🛑	
└	_tokengeneration	Internal 🔒	🛑	
└	_burn	Internal 🔒	🛑	
└	_approve	Internal 🔒	🛑	
└	_beforeTokenTransfer	Internal 🔒	🛑	
SafeMath	Library			
└	add	Internal 🔒		

CONTRACT ASSESMENT

```
| └ | sub | Internal 🔒 |   | |
| └ | sub | Internal 🔒 |   | |
| └ | mul | Internal 🔒 |   | |
| └ | div | Internal 🔒 |   | |
| └ | div | Internal 🔒 |   | |
| └ | mod | Internal 🔒 |   | |
| └ | mod | Internal 🔒 |   | |
||||||
| **SafeMathInt** | Library |  |||
| └ | mul | Internal 🔒 |   | |
| └ | div | Internal 🔒 |   | |
| └ | sub | Internal 🔒 |   | |
| └ | add | Internal 🔒 |   | |
| └ | abs | Internal 🔒 |   | |
| └ | toUint256Safe | Internal 🔒 |   | |
||||||
| **SafeMathUint** | Library |  |||
| └ | toInt256Safe | Internal 🔒 |   | |
||||||
| **Ownable** | Implementation | Context |||
| └ | <Constructor> | Public ❗️ | 🛑  |NO❗️ |
| └ | owner | Public ❗️ |   |NO❗️ |
| └ | renounceOwnership | Public ❗️ | 🛑  | onlyOwner |
| └ | transferOwnership | Public ❗️ | 🛑  | onlyOwner |
||||||
| **IPair** | Interface |  |||
| └ | sync | External ❗️ | 🛑  |NO❗️ |
||||||
| **IFactory** | Interface |  |||
| └ | createPair | External ❗️ | 🛑  |NO❗️ |
| └ | getPair | External ❗️ |   |NO❗️ |
||||||
| **IRouter** | Interface |  |||
| └ | factory | External ❗️ |   |NO❗️ |
| └ | WETH | External ❗️ |   |NO❗️ |
| └ | addLiquidityETH | External ❗️ |  💵 |NO❗️ |
| └ | swapExactTokensForTokensSupportingFeeOnTransferTokens | External ❗️ | 🛑  |NO❗️ |
| └ | swapExactETHForTokens | External ❗️ |  💵 |NO❗️ |
| └ | swapExactTokensForETHSupportingFeeOnTransferTokens | External ❗️ | 🛑  |NO❗️ |
||||||
| **DividendPayingTokenInterface** | Interface |  |||
| └ | dividendOf | External ❗️ |   |NO❗️ |
```

CONTRACT ASSESMENT

```
| └ | distributeDividends | External ❗️ | 💵 |NO❗️ |
| └ | withdrawableDividendOf | External ❗️ |   |NO❗️ |
| └ | withdrawnDividendOf | External ❗️ |   |NO❗️ |
| └ | accumulativeDividendOf | External ❗️ |   |NO❗️ |
||||||
| **DividendPayingToken** | Implementation | ERC20, DividendPayingTokenInterface, Ownable |||
| └ | <Constructor> | Public ❗️ | 🛑  | ERC20 |
| └ | <Receive Ether> | External ❗️ | 💵 |NO❗️ |
| └ | distributeDividends | Public ❗️ | 💵 |NO❗️ |
| └ | _withdrawDividendOfUser | Internal 🔒 | 🛑  | |
| └ | setRewardToken | External ❗️ | 🛑  | onlyOwner |
| └ | swapBnbForCustomToken | Internal 🔒 | 🛑  | |
| └ | dividendOf | Public ❗️ |   |NO❗️ |
| └ | withdrawableDividendOf | Public ❗️ |   |NO❗️ |
| └ | withdrawnDividendOf | Public ❗️ |   |NO❗️ |
| └ | accumulativeDividendOf | Public ❗️ |   |NO❗️ |
| └ | _transfer | Internal 🔒 | 🛑  | |
| └ | _tokengeneration | Internal 🔒 | 🛑  | |
| └ | _burn | Internal 🔒 | 🛑  | |
| └ | _setBalance | Internal 🔒 | 🛑  | |
||||||
| **IterableMapping** | Library |  |||
| └ | get | Internal 🔒 |   | |
| └ | getIndexOfKey | Internal 🔒 |   | |
| └ | getKeyAtIndex | Internal 🔒 |   | |
| └ | size | Internal 🔒 |   | |
| └ | set | Internal 🔒 | 🛑  | |
| └ | remove | Internal 🔒 | 🛑  | |
||||||
| **Address** | Library |  |||
| └ | sendValue | Internal 🔒 | 🛑  | |
||||||
| **BullDoge** | Implementation | ERC20, Ownable |||
| └ | <Constructor> | Public ❗️ | 🛑  | ERC20 |
| └ | <Receive Ether> | External ❗️ | 💵 |NO❗️ |
| └ | updateDividendTracker | Public ❗️ | 🛑  | onlyOwner |
| └ | processDividendTracker | External ❗️ | 🛑  |NO❗️ |
| └ | claim | External ❗️ | 🛑  |NO❗️ |
| └ | rescueBEP20Tokens | External ❗️ | 🛑  | onlyOwner |
| └ | forceSend | External ❗️ | 🛑  |NO❗️ |
| └ | excludeFromFees | Public ❗️ | 🛑  | onlyOwner |
| └ | excludeMultipleAccountsFromFees | Public ❗️ | 🛑  | onlyOwner |
```

CONTRACT ASSESMENT

```
| └ | excludeFromDividends | External ❗️ | 🛑  | onlyOwner |
| └ | setMarketingWallet | External ❗️ | 🛑  | onlyOwner |
| └ | setSwapTokensAtAmount | External ❗️ | 🛑  | onlyOwner |
| └ | setBuyTaxes | External ❗️ | 🛑  | onlyOwner |
| └ | setSellTaxes | External ❗️ | 🛑  | onlyOwner |
| └ | setSwapEnabled | External ❗️ | 🛑  | onlyOwner |
| └ | enableTradingEnabled | External ❗️ | 🛑  | onlyOwner |
| └ | setAntiBotBlocks | External ❗️ | 🛑  | onlyOwner |
| └ | setMinBalanceForDividends | External ❗️ | 🛑  | onlyOwner |
| └ | _setAutomatedMarketMakerPair | Private 🔐 | 🛑  | |
| └ | setGasForProcessing | External ❗️ | 🛑  | onlyOwner |
| └ | setClaimWait | External ❗️ | 🛑  | onlyOwner |
| └ | getClaimWait | External ❗️ |   |NO❗️ |
| └ | getTotalDividendsDistributed | External ❗️ |   |NO❗️ |
| └ | isExcludedFromFees | Public ❗️ |   |NO❗️ |
| └ | withdrawableDividendOf | Public ❗️ |   |NO❗️ |
| └ | getCurrentRewardToken | External ❗️ |   |NO❗️ |
| └ | dividendTokenBalanceOf | Public ❗️ |   |NO❗️ |
| └ | getAccountDividendsInfo | External ❗️ |   |NO❗️ |
| └ | getAccountDividendsInfoAtIndex | External ❗️ |   |NO❗️ |
| └ | getLastProcessedIndex | External ❗️ |   |NO❗️ |
| └ | getNumberOfDividendTokenHolders | External ❗️ |   |NO❗️ |
| └ | _transfer | Internal 🔒 | 🛑  | |
| └ | swapAndLiquify | Private 🔐 | 🛑  | |
| └ | swapTokensForBNB | Private 🔐 | 🛑  | |
| └ | addLiquidity | Private 🔐 | 🛑  | |
||||||
| **BDoge_DividendTracker** | Implementation | Ownable, DividendPayingToken |||
| └ | <Constructor> | Public ❗️ | 🛑  | DividendPayingToken |
| └ | _transfer | Internal 🔒 |   | |
| └ | setMinBalanceForDividends | External ❗️ | 🛑  | onlyOwner |
| └ | excludeFromDividends | External ❗️ | 🛑  | onlyOwner |
| └ | updateClaimWait | External ❗️ | 🛑  | onlyOwner |
| └ | getLastProcessedIndex | External ❗️ |   |NO❗️ |
| └ | getNumberOfTokenHolders | External ❗️ |   |NO❗️ |
| └ | getCurrentRewardToken | External ❗️ |   |NO❗️ |
| └ | getAccount | Public ❗️ |   |NO❗️ |
| └ | getAccountAtIndex | Public ❗️ |   |NO❗️ |
| └ | canAutoClaim | Private 🔐 |   | |
| └ | setBalance | Public ❗️ | 🛑  | onlyOwner |
| └ | process | Public ❗️ | 🛑  |NO❗️ |
```

CONTRACT ASSESMENT

```
| └ | processAccount | Public ❗ | 🛑  | onlyOwner |
||||||
| **ERC20** | Implementation | Context, IERC20, IERC20Metadata |||
| └ | <Constructor> | Public ❗ | 🛑  |NO❗ |
| └ | name | Public ❗ |   |NO❗ |
| └ | symbol | Public ❗ |   |NO❗ |
| └ | decimals | Public ❗ |   |NO❗ |
| └ | totalSupply | Public ❗ |   |NO❗ |
| └ | balanceOf | Public ❗ |   |NO❗ |
| └ | transfer | Public ❗ | 🛑  |NO❗ |
| └ | allowance | Public ❗ |   |NO❗ |
| └ | approve | Public ❗ | 🛑  |NO❗ |
| └ | transferFrom | Public ❗ | 🛑  |NO❗ |
| └ | increaseAllowance | Public ❗ | 🛑  |NO❗ |
| └ | decreaseAllowance | Public ❗ | 🛑  |NO❗ |
| └ | _transfer | Internal 🔒 | 🛑  | |
| └ | _mint | Internal 🔒 | 🛑  | |
| └ | _burn | Internal 🔒 | 🛑  | |
| └ | _approve | Internal 🔒 | 🛑  | |
| └ | _spendAllowance | Internal 🔒 | 🛑  | |
| └ | _beforeTokenTransfer | Internal 🔒 | 🛑  | |
| └ | _afterTokenTransfer | Internal 🔒 | 🛑  | |
||||||
| **IERC20** | Interface |  |||
| └ | totalSupply | External ❗ |   |NO❗ |
| └ | balanceOf | External ❗ |   |NO❗ |
| └ | transfer | External ❗ | 🛑  |NO❗ |
| └ | allowance | External ❗ |   |NO❗ |
| └ | approve | External ❗ | 🛑  |NO❗ |
| └ | transferFrom | External ❗ | 🛑  |NO❗ |
||||||
| **IERC20Metadata** | Interface | IERC20 |||
| └ | name | External ❗ |   |NO❗ |
| └ | symbol | External ❗ |   |NO❗ |
| └ | decimals | External ❗ |   |NO❗ |
||||||
| **Context** | Implementation |  |||
| └ | _msgSender | Internal 🔒 |   | |
| └ | _msgData | Internal 🔒 |   | |
||||||
| **Ownable** | Implementation | Context |||
| └ | <Constructor> | Public ❗ | 🛑  |NO❗ |
```

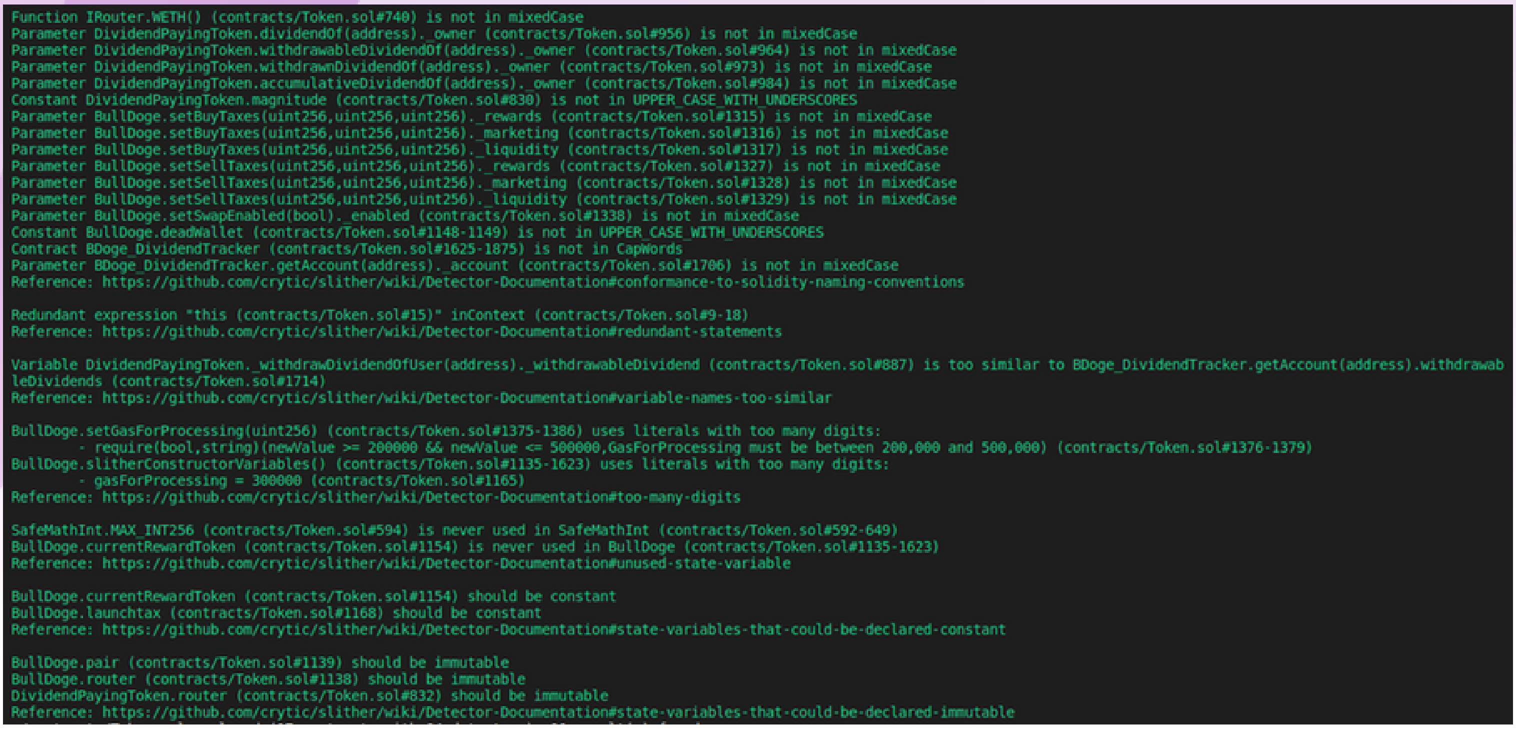
CONTRACT ASSESMENT

└	owner	Public ❗		NO❗
└	_checkOwner	Internal 🔒		
└	renounceOwnership	Public ❗	🛑	onlyOwner
└	transferOwnership	Public ❗	🛑	onlyOwner
└	_transferOwnership	Internal 🔒	🛑	

Legend

Symbol	Meaning
🛑	Function can modify state
💵	Function is payable

STATIC ANALYSIS



```
Function IRouter.WETH() (contracts/Token.sol#740) is not in mixedCase
Parameter DividendPayingToken.dividendOf(address)._owner (contracts/Token.sol#956) is not in mixedCase
Parameter DividendPayingToken.withdrawableDividendOf(address)._owner (contracts/Token.sol#964) is not in mixedCase
Parameter DividendPayingToken.withdrawnDividendOf(address)._owner (contracts/Token.sol#973) is not in mixedCase
Parameter DividendPayingToken.accumulativeDividendOf(address)._owner (contracts/Token.sol#984) is not in mixedCase
Constant DividendPayingToken.magnitude (contracts/Token.sol#830) is not in UPPER_CASE_WITH_UNDERSCORES
Parameter BullDoge.setBuyTaxes(uint256,uint256,uint256)._rewards (contracts/Token.sol#1315) is not in mixedCase
Parameter BullDoge.setBuyTaxes(uint256,uint256,uint256)._marketing (contracts/Token.sol#1316) is not in mixedCase
Parameter BullDoge.setBuyTaxes(uint256,uint256,uint256)._liquidity (contracts/Token.sol#1317) is not in mixedCase
Parameter BullDoge.setSellTaxes(uint256,uint256,uint256)._rewards (contracts/Token.sol#1327) is not in mixedCase
Parameter BullDoge.setSellTaxes(uint256,uint256,uint256)._marketing (contracts/Token.sol#1328) is not in mixedCase
Parameter BullDoge.setSellTaxes(uint256,uint256,uint256)._liquidity (contracts/Token.sol#1329) is not in mixedCase
Parameter BullDoge.setSwapEnabled(bool)._enabled (contracts/Token.sol#1338) is not in mixedCase
Constant BullDoge.deadWallet (contracts/Token.sol#1148-1149) is not in UPPER_CASE_WITH_UNDERSCORES
Contract BDoge_DividendTracker (contracts/Token.sol#1625-1875) is not in CapWords
Parameter BDoge_DividendTracker.getAccount(address)._account (contracts/Token.sol#1706) is not in mixedCase
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#conformance-to-solidity-naming-conventions

Redundant expression "this (contracts/Token.sol#15)" inContext (contracts/Token.sol#9-18)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#redundant-statements

Variable DividendPayingToken._withdrawDividendOfUser(address)._withdrawableDividend (contracts/Token.sol#887) is too similar to BDoge_DividendTracker.getAccount(address).withdrawab
leDividends (contracts/Token.sol#1714)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#variable-names-too-similar

BullDoge.setGasForProcessing(uint256) (contracts/Token.sol#1375-1386) uses literals with too many digits:
        - require(bool,string)(newValue >= 200000 && newValue <= 500000,GasForProcessing must be between 200,000 and 500,000) (contracts/Token.sol#1376-1379)
BullDoge.slitherConstructorVariables() (contracts/Token.sol#1135-1623) uses literals with too many digits:
        - gasForProcessing = 300000 (contracts/Token.sol#1165)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#too-many-digits

SafeMathInt.MAX_INT256 (contracts/Token.sol#594) is never used in SafeMathInt (contracts/Token.sol#592-649)
BullDoge.currentRewardToken (contracts/Token.sol#1154) is never used in BullDoge (contracts/Token.sol#1135-1623)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#unused-state-variable

BullDoge.currentRewardToken (contracts/Token.sol#1154) should be constant
BullDoge.launchtax (contracts/Token.sol#1168) should be constant
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#state-variables-that-could-be-declared-constant

BullDoge.pair (contracts/Token.sol#1139) should be immutable
BullDoge.router (contracts/Token.sol#1138) should be immutable
DividendPayingToken.router (contracts/Token.sol#832) should be immutable
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#state-variables-that-could-be-declared-immutable
```

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/0xa188571a3d37e884c372a17bf3f310ca313ee6382bf0a84b53c5c2b0213a7961

2- Buying when excluded (0% tax) (passed):
https://testnet.bscscan.com/tx/0x4a0015402cc5a8460bb85b6fffb9caa6edff314c06db8d6e801a973112e3b773

3- Selling when excluded (0% tax) (passed):
https://testnet.bscscan.com/tx/0xfcfa601d0dba830deeb7d7072729a2728e78c14a73b6129201667a808817ef4f

4- Transferring when excluded (0% tax) (passed):
https://testnet.bscscan.com/tx/0x81bb326c6cc2c207b1264bb2554909faa420befeffaba3d9eff0255dd5d25505

5- Buying when not excluded (upto 12% tax) (passed):
https://testnet.bscscan.com/tx/0x7f2553727c2dc607b398667c7ce59819154aff7a82fe6e590aa907a388e5d152

6- Selling when not excluded (upto 12% tax) (passed):
https://testnet.bscscan.com/tx/0x8ee579ac57f8b5420dd472a0ea6fa65306d8ae6f06cad4501071a5e66af9e2d4

FUNCTIONAL TESTING

7- Transferring when not excluded (0% tax) (passed):
https://testnet.bscscan.com/tx/0x42281a7b288d4df14bcd4adc589eabb52f947fce9c2a89a3a525f91dc4ab9103

8- Internal swap (passed):
As can seen in this transaction, marketing wallet received BNB
https://testnet.bscscan.com/address/0xc8c6688dfcb2260191e016f05b4db736e5674756#internaltx

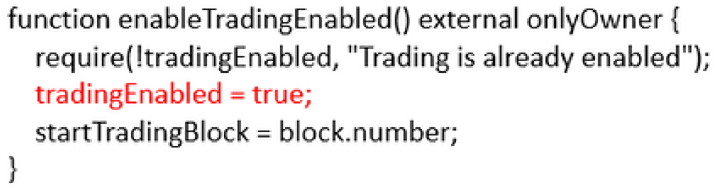
9- Auto Liquidity (passed):
Auto liquidity generated tokens are sent to owner’s wallet
https://testnet.bscscan.com/token/0x1a1c0fb651cf0fbb39e050876ca20ce532e1ebaf?a=0x322dab6325de6f5bc2ba8efecc2bcbecac4f89f3

10- Distribution of rewards (passed):
BUSD tokens are distributed between holders, this can be seen in this transaction
https://testnet.bscscan.com/tx/0x57728bd8646597847ead32c80e8370dc8f60a025e049efd2483e7856467d69bc

MANUAL TESTING

Centralization - Owner must enable trading

Severity: High
Function: openTrade
Lines: 1342
Status: Resolved
Overview:
The owner must activate trading for investors to buy, sell, or transfer tokens. If trading remains disabled, token holders will be unable to trade their tokens.



```
function enableTradingEnabled() external onlyOwner {
    require(!tradingEnabled, "Trading is already enabled");
    tradingEnabled = true;
    startTradingBlock = block.number;
}
```

Recommendation:
Incorporate a safety mechanism that allows investors to activate trading if a specified duration has elapsed since the conclusion of the presale or consider alternative ways such as allowing trades ater investors claimed their presale tokens.
Allevation:
Contract is owned by Safu Dev, hence enabling trade is guaranteed

MANUAL TESTING

Logical - Setting internal swap threshold to 0 can disable sells

Severity: High
Function: setSwapThreshold
Lines: 1310
Status: No Resolved

If the **swaptokensAtAmount** is set to 0, sell transactions will fail at the _transfer function. This occurs because the checks for performing a swapAndLiquify will still pass even if the swapThreshold is set to 0 and the contract has 0 tokens. Consequently, the transaction will fail while attempting to swap 0 tokens (i.e., **swaptokensAtAmount**) to BNB. Additionally, setting the swapThreshold to an excessively large number leads to a high slippage percentage during sell transactions.

```
function setSwapTokensAtAmount(uint256 amount) external onlyOwner {
  swapTokensAtAmount = amount * 10 ** 9;
}

if (
  canSwap &&
  !swapping &&
  swapEnabled &&
  !automatedMarketMakerPairs[from] &&
  !_isExcludedFromFees[from] &&
  !_isExcludedFromFees[to]
) {
  swapping = true;
  if (swapTax > 0) {
    swapAndLiquify(swapTokensAtAmount, swapTax);
  }
  swapping = false;
}
```

Recommendation:

Ensure that the swapThreshold is set to a value greater than a reasonable minimum and less than a reasonable maximum. This will help prevent issues related to disabled sell transactions or high slippage percentages during trades.

MANUAL TESTING

Centralization – Owner receives LP tokens

Severity: Medium
Function: addLiquidity
Lines: 1615
Status: No Resolved
Auto liquidity generated pool tokens are sent to ower's wallet. Overtime this accumulated LP tokens will contains a larger share of liquidity and if used in a malicious way, may impact token price in a negative way.

```
  function addLiquidity(uint256 tokenAmount, uint256 ethAmount) private {
    // approve token transfer to cover all possible scenarios
    _approve(address(this), address(router), tokenAmount);

    // add the liquidity
    router.addLiquidityETH{value: ethAmount}(
      address(this),
      tokenAmount,
      0, // slippage is unavoidable
      0, // slippage is unavoidable
      owner(),
      block.timestamp
    );
  }
```

Recommendation:
consider burning auto-liquidity generated pool tokens.

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 specializes 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