



Smart Contract Audit

FOR
Original Pepe

DATED : 6 May 23'



AUDIT SUMMARY

Project name – Original Pepe

Date: 6 May, 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 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/token/0xfFb6982C92ddBaD3c5ba3248D4bf4aA9b0B02Cf7>



Token Information

Token Name : Original Pepe

Token Symbol: OGPEPE

Decimals: 18

Token Supply: 100,000,000,000

Token Address:

0x2a868Dc1E17A6Be7D2355dAe9b499A0fab223D24

Checksum:

df96bb414bb532fa83ebf92d041cccf9ed2827d8

Owner:

0xCc2309E3DF77E2b59d2a2B4871cF0a206eC14de0
(at time of audit)

Deployer:

0x7EA547A2B0FC764da7CfEcCD26927A43a7A6Bb8
B



TOKEN OVERVIEW

Fees:

Buy Fees: 0%

Sell Fees: 10%

Transfer Fees: 0%

Fees Privilege: None

Ownership: owned

Minting: No mint function

Max Tx Amount/ Max Wallet Amount: Yes

Blacklist: No

Other Privileges: including or excluding from fees -
changing swap threshold



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 Limitand 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

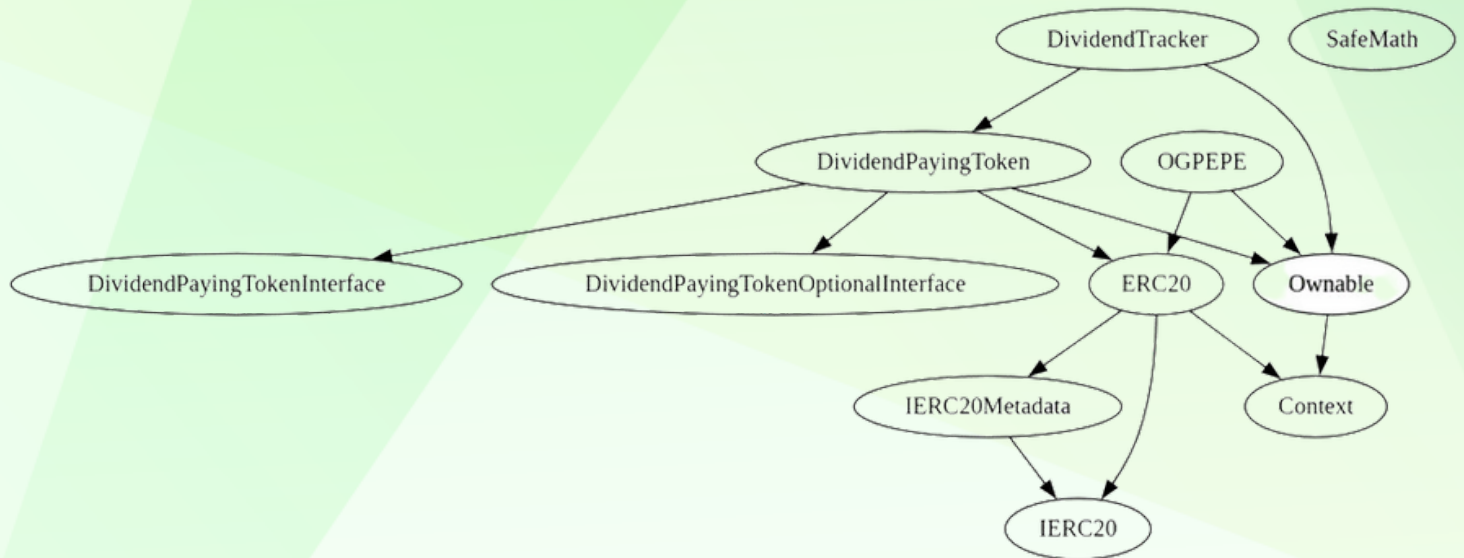
◆ Low-Risk

0

◆ Gas Optimization / Suggestions

0

INHERITANCE TREE



POINTS TO NOTE

- Owner is not able to change current tax (10% for sells)
 - Owner is not able to set transfer or buy tax (0% forever)
 - 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 mint new tokens
 - Owner is not able to disable trades
 - Trades are already enabled for public
 - Token has BNB rewards with auto-distribution
 - Fees are used for rewards (BNB) and marketing
-

CONTRACT ASSESMENT

Contract	Type	Bases			
:-----: :-----: :-----: :-----: :-----:					
└─	**Function Name**	**Visibility**	**Mutability**	**Modifiers**	
Context Implementation					
└─	_msgSender	Internal	🔒		
└─	_msgData	Internal	🔒		
Ownable Implementation Context					
└─	<Constructor>	Public	!		● NO !
└─	owner	Public	!		NO !
└─	renounceOwnership	Public	!		● onlyOwner
└─	transferOwnership	Public	!		● onlyOwner
└─	_transferOwnership	Internal	🔒		●
SafeMath Library					
└─	add	Internal	🔒		
└─	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	🔒		
IterableMapping Library					
└─	get	Internal	🔒		
└─	getIndexOfKey	Internal	🔒		
└─	getKeyAtIndex	Internal	🔒		
└─	size	Internal	🔒		
└─	set	Internal	🔒		●
└─	remove	Internal	🔒		●



CONTRACT ASSESMENT

```
||||| |
| **IUniswapV2Factory** | Interface | |||
|  | feeTo | External ! | | NO ! |
|  | feeToSetter | External ! | | NO ! |
|  | getPair | External ! | | NO ! |
|  | allPairs | External ! | | NO ! |
|  | allPairsLength | External ! | | NO ! |
|  | createPair | External ! | ● | NO ! |
|  | setFeeTo | External ! | ● | NO ! |
|  | setFeeToSetter | External ! | ● | NO ! |
|||||
| **IUniswapV2Pair** | Interface | |||
|  | name | External ! | | NO ! |
|  | symbol | External ! | | NO ! |
|  | decimals | External ! | | NO ! |
|  | totalSupply | External ! | | NO ! |
|  | balanceOf | External ! | | NO ! |
|  | allowance | External ! | | NO ! |
|  | approve | External ! | ● | NO ! |
|  | transfer | External ! | ● | NO ! |
|  | transferFrom | External ! | ● | NO ! |
|  | DOMAIN_SEPARATOR | External ! | | NO ! |
|  | PERMIT_TYPEHASH | External ! | | NO ! |
|  | nonces | External ! | | NO ! |
|  | permit | External ! | ● | NO ! |
|  | MINIMUM_LIQUIDITY | External ! | | NO ! |
|  | factory | External ! | | NO ! |
|  | token0 | External ! | | NO ! |
|  | token1 | External ! | | NO ! |
|  | getReserves | External ! | | NO ! |
|  | price0CumulativeLast | External ! | | NO ! |
|  | price1CumulativeLast | External ! | | NO ! |
|  | kLast | External ! | | NO ! |
|  | mint | External ! | ● | NO ! |
|  | burn | External ! | ● | NO ! |
|  | swap | External ! | ● | NO ! |
|  | skim | External ! | ● | NO ! |
|  | sync | External ! | ● | NO ! |
|  | initialize | External ! | ● | NO ! |
|||||
| **IUniswapV2Router01** | Interface | |||
```

CONTRACT ASSESMENT

```

|  | factory | External ! | | NO ! |
|  | WETH | External ! | | NO ! |
|  | addLiquidity | External ! | ● | NO ! |
|  | addLiquidityETH | External ! | 💰 | NO ! |
|  | removeLiquidity | External ! | ● | NO ! |
|  | removeLiquidityETH | External ! | ● | NO ! |
|  | removeLiquidityWithPermit | External ! | ● | NO ! |
|  | removeLiquidityETHWithPermit | External ! | ● | NO ! |
|  | swapExactTokensForTokens | External ! | ● | NO ! |
|  | swapTokensForExactTokens | External ! | ● | NO ! |
|  | swapExactETHForTokens | External ! | 💰 | NO ! |
|  | swapTokensForExactETH | External ! | ● | NO ! |
|  | swapExactTokensForETH | External ! | ● | NO ! |
|  | swapETHForExactTokens | External ! | 💰 | NO ! |
|  | quote | External ! | | NO ! |
|  | getAmountOut | External ! | | NO ! |
|  | getAmountIn | External ! | | NO ! |
|  | getAmountsOut | External ! | | NO ! |
|  | getAmountsIn | External ! | | NO ! |
|  |  |
|  | **IUniswapV2Router02** | Interface | IUniswapV2Router01 |  |
|  | removeLiquidityETHSupportingFeeOnTransferTokens | External ! | ● | NO ! |
|  | removeLiquidityETHWithPermitSupportingFeeOnTransferTokens | External ! | ● | NO ! |
|  | swapExactTokensForTokensSupportingFeeOnTransferTokens | External ! | ● | NO ! |
|  | swapExactETHForTokensSupportingFeeOnTransferTokens | External ! | 💰 | NO ! |
|  | swapExactTokensForETHSupportingFeeOnTransferTokens | External ! | ● | NO ! |
|  |  |
|  | **IERC20** | Interface |  |
|  | totalSupply | External ! | | NO ! |
|  | balanceOf | External ! | | NO ! |
|  | allowance | External ! | | NO ! |
|  | transfer | External ! | ● | NO ! |
|  | approve | External ! | ● | NO ! |
|  | transferFrom | External ! | ● | NO ! |
|  |  |
|  | **Address** | Library |  |
|  | isContract | Internal 🔒 | |  |
|  | sendValue | Internal 🔒 | ● |  |
|  | functionCall | Internal 🔒 | ● |  |
|  | functionCall | Internal 🔒 | ● |  |
|  | functionCallWithValue | Internal 🔒 | ● |  |

```



CONTRACT ASSESMENT

```
|  | functionCallWithValue | Internal | | |
|  | _functionCallWithValue | Private | | |
| | | |
| **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 | | |
|  | _mint | Internal | | |
|  | _burn | Internal | | |
|  | _approve | Internal | | |
|  | _beforeTokenTransfer | Internal | | |
| | | |
| **DividendPayingTokenInterface** | Interface | | |
|  | dividendOf | External | | NO |
|  | withdrawDividend | External | | NO |
| | | |
| **DividendPayingTokenOptionalInterface** | Interface | | |
|  | withdrawableDividendOf | External | | NO |
|  | withdrawnDividendOf | External | | NO |
|  | accumulativeDividendOf | External | | NO |
| | | |
| **DividendPayingToken** | Implementation | ERC20, Ownable, DividendPayingTokenInterface,
DividendPayingTokenOptionalInterface | | |
|  | <Constructor> | Public | | ERC20 |
|  | distributeDividends | Public | | onlyOwner |
|  | withdrawDividend | Public | | NO |
|  | _withdrawDividendOfUser | Internal | | |
```

CONTRACT ASSESMENT

```

|  | dividendOf | Public ! | | NO ! |
|  | withdrawableDividendOf | Public ! | | NO ! |
|  | withdrawnDividendOf | Public ! | | NO ! |
|  | accumulativeDividendOf | Public ! | | NO ! |
|  | _transfer | Internal 🔒 | ● | |
|  | _mint | Internal 🔒 | ● | |
|  | _burn | Internal 🔒 | ● | |
|  | _setBalance | Internal 🔒 | ● | |
|||||
| **DividendTracker** | Implementation | Ownable, DividendPayingToken |||
|  | <Constructor> | Public ! | ● | DividendPayingToken |
|  | <Receive Ether> | External ! | 💰 | NO ! |
|  | _transfer | Internal 🔒 | | |
|  | withdrawDividend | Public ! | | NO ! |
|  | updateMinimumTokenBalanceForDividends | External ! | ● | onlyOwner |
|  | excludeFromDividends | External ! | ● | onlyOwner |
|  | updateClaimWait | External ! | ● | onlyOwner |
|  | setLastProcessedIndex | External ! | ● | onlyOwner |
|  | getLastProcessedIndex | External ! | | NO ! |
|  | getNumberOfTokenHolders | External ! | | NO ! |
|  | getAccount | Public ! | | NO ! |
|  | getAccountAtIndex | Public ! | | NO ! |
|  | canAutoClaim | Private 🔒 | | |
|  | setBalance | External ! | ● | onlyOwner |
|  | process | Public ! | ● | NO ! |
|  | processAccount | Public ! | ● | onlyOwner |
|||||
| **OGPEPE** | Implementation | ERC20, Ownable |||
|  | <Constructor> | Public ! | ● | ERC20 |
|  | <Receive Ether> | External ! | 💰 | NO ! |
|  | claimStuckTokens | External ! | ● | onlyOwner |
|  | excludeFromFees | External ! | ● | onlyOwner |
|  | isExcludedFromFees | Public ! | | NO ! |
|  | changeMarketingWallet | External ! | ● | onlyOwner |
|  | _transfer | Internal 🔒 | ● | |
|  | swapAndSendDividends | Private 🔒 | ● | |
|  | setSwapEnabled | External ! | ● | onlyOwner |
|  | setSwapTokensAtAmount | External ! | ● | onlyOwner |
|  | updateGasForProcessing | Public ! | ● | onlyOwner |
|  | updateMinimumBalanceForDividends | External ! | ● | onlyOwner |
|  | updateClaimWait | External ! | ● | onlyOwner |

```



CONTRACT ASSESMENT

	↳		getClaimWait		External	!			NO	!		
	↳		getTotalDividendsDistributed		External	!			NO	!		
	↳		withdrawableDividendOf		Public	!			NO	!		
	↳		dividendTokenBalanceOf		Public	!			NO	!		
	↳		totalRewardsEarned		Public	!			NO	!		
	↳		excludeFromDividends		External	!		●		onlyOwner		
	↳		isExcludedFromDividends		Public	!			NO	!		
	↳		getAccountDividendsInfo		External	!			NO	!		
	↳		getAccountDividendsInfoAtIndex		External	!			NO	!		
	↳		processDividendTracker		External	!		●		NO	!	
	↳		claim		External	!		●		NO	!	
	↳		claimAddress		External	!		●		onlyOwner		
	↳		getLastProcessedIndex		External	!			NO	!		
	↳		setLastProcessedIndex		External	!		●		onlyOwner		
	↳		getNumberOfDividendTokenHolders		External	!			NO	!		

Legend

	Symbol		Meaning	
	:-----:		-----	
	●		Function can modify state	
	💰		Function is payable	



STATIC ANALYSIS

```
Function IUniswapV2Pair.DOMAIN_SEPARATOR() (contracts/Token.sol#303) is not in mixedCase
Function IUniswapV2Pair.PERMIT_TYPEHASH() (contracts/Token.sol#305) is not in mixedCase
Function IUniswapV2Pair.MINIMUM_LIQUIDITY() (contracts/Token.sol#336) is not in mixedCase
Function IUniswapV2Router01.WETH() (contracts/Token.sol#378) is not in mixedCase
Parameter DividendPayingToken.dividendOf(address). owner (contracts/Token.sol#958) is not in mixedCase
Parameter DividendPayingToken.withdrawableDividendOf(address). owner (contracts/Token.sol#963) is not in mixedCase
Parameter DividendPayingToken.withdrawnDividendOf(address). owner (contracts/Token.sol#969) is not in mixedCase
Parameter DividendPayingToken.accumulativeDividendOf(address). owner (contracts/Token.sol#975) is not in mixedCase
Constant DividendPayingToken.magnitude (contracts/Token.sol#900) is not in UPPER_CASE_WITH_UNDERSCORES
Parameter DividendTracker.updateMinimumTokenBalanceForDividends(uint256). newMinimumBalance (contracts/Token.sol#1076) is not in mixedCase
Parameter DividendTracker.getAccount(address). account (contracts/Token.sol#1121) is not in mixedCase
Parameter OGPEPE.changeMarketingWallet(address). marketingWallet (contracts/Token.sol#1440) is not in mixedCase
Parameter OGPEPE.setSwapEnabled(bool). enabled (contracts/Token.sol#1574) is not in mixedCase
Variable OGPEPE.DEAD (contracts/Token.sol#1313) is not in mixedCase
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#conformance-to-solidity-naming-conventions

Variable IUniswapV2Router01.addLiquidity(address,address,uint256,uint256,uint256,uint256,address,uint256).amountADesired (contracts/Token.sol#383) is too similar to IUni
swapV2Router01.addLiquidity(address,address,uint256,uint256,uint256,uint256,address,uint256).amountBDesired (contracts/Token.sol#384)
Variable DividendPayingToken.withdrawDividendOfUser(address). withdrawableDividend (contracts/Token.sol#937) is too similar to DividendTracker.getAccount(address).withd
rawableDividends (contracts/Token.sol#1129)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#variable-names-too-similar

OGPEPE.setSwapTokensAtAmount(uint256) (contracts/Token.sol#1579-1586) uses literals with too many digits:
- require(bool,string)(newAmount > totalSupply() / 100000,SwapTokensAtAmount must be greater than 0.001% of total supply) (contracts/Token.sol#1580-1583)
OGPEPE.updateGasForProcessing(uint256) (contracts/Token.sol#1588-1599) 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#1589-1592)
OGPEPE.slitherConstructorVariables() (contracts/Token.sol#1295-1720) uses literals with too many digits:
- gasForProcessing = 300000 (contracts/Token.sol#1326)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#too-many-digits

SafeMathInt.MAX_INT256 (contracts/Token.sol#125) is never used in SafeMathInt (contracts/Token.sol#123-165)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#unused-state-variable

ERC20.antiBotEnabled (contracts/Token.sol#711) should be constant
OGPEPE.DEAD (contracts/Token.sol#1313) should be constant
OGPEPE._totalFeesOnBuy (contracts/Token.sol#1305) should be constant
OGPEPE._totalFeesOnSell (contracts/Token.sol#1306) should be constant
OGPEPE.marketingFeeOnBuy (contracts/Token.sol#1302) should be constant
OGPEPE.marketingFeeOnSell (contracts/Token.sol#1303) should be constant
OGPEPE.rewardFeeOnBuy (contracts/Token.sol#1299) should be constant
OGPEPE.rewardFeeOnSell (contracts/Token.sol#1300) should be constant
OGPEPE.rewardToken (contracts/Token.sol#1325) should be constant
OGPEPE.swapWithLimit (contracts/Token.sol#1318) should be constant
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#state-variables-that-could-be-declared-constant

OGPEPE.dividendTracker (contracts/Token.sol#1324) should be immutable
OGPEPE.uniswapV2Pair (contracts/Token.sol#1311) should be immutable
OGPEPE.uniswapV2Router (contracts/Token.sol#1310) 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/0x07679cff0330c4715f24285142023b85419cd78c52885b232811102c4dbf7b7d>

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

<https://testnet.bscscan.com/tx/0xdf3168a234093bc30dc88707176facea06174bad a134da72a199079b4de57075>

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

<https://testnet.bscscan.com/tx/0x399f038a5137c1b0f50110f5703a33c520981f776 c9c5075032e53de8b9383be>

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

<https://testnet.bscscan.com/tx/0x22cf0e6ceb44e575a130fd9e8d014694b115764a4 3bdbbe54f5c57d0e86b09d1>

5- Buying when not excluded from fees (0% tax) (passed):

<https://testnet.bscscan.com/tx/0xa6250fe6c270e012b51e1050d841ae8c8db761ba5 859cea9bd0f5e7657831db5>

6- Selling when not excluded from fees (10% tax) (passed):

<https://testnet.bscscan.com/tx/0x24c519d1eae5eb9feed317e0f83d4f348958537b 9aac5626af1aa90a7282bbef>



FUNCTIONAL TESTING

7- Transferring when not excluded from fees (0% tax) (passed):

<https://testnet.bscscan.com/tx/0x15ec6b8584fedc7d1a68e7af384448436ff6394cc60e572d2f0fb4003fcaa0eb>

8- Internal swap (passed):

Marketing wallet received BNB

<https://testnet.bscscan.com/address/0x59320e4fc5507deb206b1af9a6d439b1c0dc0d60#internaltx>



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.



MANUAL TESTING

NO ISSUES FOUND



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>
