



Smart Contract Audit

FOR

Elantra official coin - \$EOC

DATED : 19 august 23'



MANUAL TESTING

Centralization – Maximum Buy/Sell/Transfer

Severity: High

function: setMaxTxAmount

Status: Open

Overview:

Owner is able to set `_maxTxAmount` to 0 which means no one would be able to buy/sell/transfer tokens (except owner)

```
function setMaxTxAmount(uint256 maxTxAmount) external onlyOwner {  
    _maxTxAmount = maxTxAmount;  
}
```

Suggestion

as a general suggestion, ensure that `_maxTxAmount` is greater than 0.1% of total supply:

```
function setMaxTxAmount(uint256 maxTxAmount) external onlyOwner {  
    require(_maxTxAmount >= totalSupply() / 1000);  
    _maxTxAmount = maxTxAmount;  
}
```



MANUAL TESTING

Centralization – Excessive Fees

Severity: High

function: setTaxFee – setBuybackFee - setMarketingFee

Status: Open

Overview:

Owner is able to set up to 100% tax on buy/sell/transfers

```
function setTaxFee(uint256 taxFee) external onlyOwner {  
    _taxFee = taxFee;  
}  
  
function setBuybackFee(uint256 _buybackFee) external onlyOwner {  
    buybackFee = _buybackFee;  
    _liquidityFee = buybackFee.add(marketingFee);  
}  
  
function setMarketingFee(uint256 _marketingFee) external onlyOwner {  
    require(  
        _marketingFee >= dappbuilderFee,  
        "Marketing fee can not be less than 5 percent"  
    );  
    marketingFee = _marketingFee;  
    _liquidityFee = buybackFee.add(marketingFee);  
}
```

Suggestion

Set an upper bound for maximum amount of buy/sell/transfer fees.

```
function setTaxFee(uint256 taxFee) external onlyOwner {  
    _taxFee = taxFee;  
    require(_taxFee + _liquidityFee <= 100);  
}  
  
function setBuybackFee(uint256 _buybackFee) external onlyOwner {  
    buybackFee = _buybackFee;  
    _liquidityFee = buybackFee.add(marketingFee);  
    require(_taxFee + _liquidityFee <= 100);  
}  
  
function setMarketingFee(uint256 _marketingFee) external onlyOwner {  
    require(  
        _marketingFee >= dappbuilderFee,  
        "Marketing fee can not be less than 5 percent"  
    );  
    marketingFee = _marketingFee;  
    _liquidityFee = buybackFee.add(marketingFee);  
    require(_taxFee + _liquidityFee <= 100);  
}
```



MANUAL TESTING

Logical – Setting swap threshold to zero

Severity: High

function: setNumTokensSellToAddToLiquidity

Status: Open

Overview:

Setting minimumTokensBeforeSwap to zero can disable internal swaps since contract will try to swap zero tokens (minimumTokensBeforeSwap) into BNB which will revert due to pancakeswap contract logic

```
function setNumTokensSellToAddToLiquidity(
    uint256 _minimumTokensBeforeSwap
) external onlyOwner {
    minimumTokensBeforeSwap = _minimumTokensBeforeSwap;
}
```

Suggestion

Ensure that minimumTokensBeforeSwap is greater than 0

```
function setNumTokensSellToAddToLiquidity(
    uint256 _minimumTokensBeforeSwap
) external onlyOwner {
    require( minimumTokensBeforeSwap > 0);
    minimumTokensBeforeSwap = _minimumTokensBeforeSwap;
}
```



AUDIT SUMMARY

Project name - Elantra official coin - \$EOC

Date: 19 Aug, 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	0	3	1	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/0x5815c0d948154AA2399952d1dB523121E8D005FE>



Token Information

Token Name : Elantra official coin - \$EOC

Token Symbol: \$EOC

Decimals: 9

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

Token Address:

0xd08F831c108E4AC49B427BB53124BDcC26A38972

Checksum:

a068195fcae36cd9091ccf8a3bc311a3e75cac76

Owner:

0x00

(at time of writing the audit)

Deployer:

0x0456E649bD83e4786988716aD5D688ebF0EAebfE



TOKEN OVERVIEW

Fees:

Buy Fees: 15%

Sell Fees: 15%

Transfer Fees: 15%

Fees Privilege: static (for 1 year)

Ownership: renounced (for 1 year)

Minting: No mint function

Max Tx Amount/ Max Wallet Amount: No

Blacklist: No

Other Privileges: Initial distribution of the tokens
modifying fees
adjusting maximum buy/sell/transfers



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



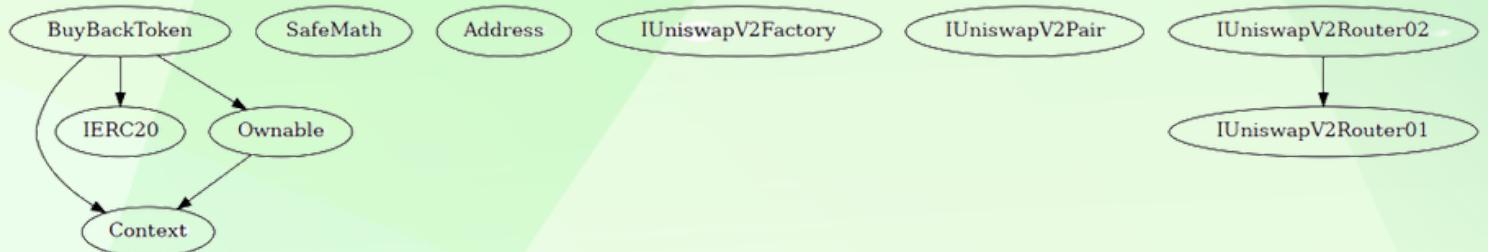
CLASSIFICATION OF RISK

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	3
◆ Medium-Risk	1
◆ Low-Risk	0
◆ Gas Optimization / Suggestions	0

INHERITANCE TREE





POINTS TO NOTE

- **Ownership of contract is renounced for one year, below features will be available to owner after 1 year**
- **Maximum buy/sell/transfer amount is set to 0.003% of total supply for 1 year**
- Owner is able to change buy/sell/transfer fees within 0-100%
- Owner is not able to blacklist an arbitrary wallet
- Owner is able to disable trades
- Owner is not able to mint new tokens
- Owner is able to set maximum buy/sell/transfer limit

CONTRACT ASSESSMENT

Contract	Type	Bases		
L **Function Name**	**Visibility**	**Mutability**		
Modifiers				
Context Implementation				
L _msgSender Internal				
L _msgData 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	
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				
Address Library				
L isContract Internal				
L sendValue Internal				
L functionCall Internal				
L functionCall Internal				

CONTRACT ASSESSMENT

```

| L | functionCallWithValue | Internal 🔒 | 🔴 | |
| L | functionCallWithValue | Internal 🔒 | 🔴 | |
| L | _functionCallWithValue | Private 🔒 | 🔴 | |
|||||
| **Ownable** | Implementation | Context |||
| L | <Constructor> | Public ! | 🔴 | NO ! |
| L | owner | Public ! | | NO ! |
| L | renounceOwnership | Public ! | 🔴 | onlyOwner |
| L | transferOwnership | Public ! | 🔴 | onlyOwner |
| L | getUnlockTime | Public ! | | NO ! |
| L | getTime | Public ! | | NO ! |
| L | lock | Public ! | 🔴 | onlyOwner |
| L | unlock | Public ! | 🔴 | NO ! |
|||||
| **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 ! |

```

CONTRACT ASSESSMENT

```

| 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 | 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 !

```

CONTRACT ASSESSMENT

```

| 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 ! | | | |
|||||
| **BuyBackToken** | 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 ! |

```

CONTRACT ASSESSMENT

```

| L | transferFrom | Public ! | 🔴 | NO ! |
| L | increaseAllowance | Public ! | 🔴 | NO ! |
| L | decreaseAllowance | Public ! | 🔴 | NO ! |
| L | isExcludedFromReward | Public ! | | NO ! |
| L | totalFees | Public ! | | NO ! |
| L | minimumTokensBeforeSwapAmount | Public ! | | NO ! |
| L | buyBackUpperLimitAmount | Public ! | | NO ! |
| L | deliver | Public ! | 🔴 | NO ! |
| L | reflectionFromToken | Public ! | | NO ! |
| L | tokenFromReflection | Public ! | | NO ! |
| L | excludeFromReward | Public ! | 🔴 | onlyOwner |
| L | includeInReward | External ! | 🔴 | onlyOwner |
| L | _approve | Private 🔒 | 🔴 | |
| L | _transfer | Private 🔒 | 🔴 | |
| L | swapTokens | Private 🔒 | 🔴 | lockTheSwap |
| L | buyBackTokens | Private 🔒 | 🔴 | lockTheSwap |
| L | swapTokensForEth | Private 🔒 | 🔴 | |
| L | swapETHForTokens | Private 🔒 | 🔴 | |
| L | addLiquidity | Private 🔒 | 🔴 | |
| L | _tokenTransfer | Private 🔒 | 🔴 | |
| L | _transferStandard | Private 🔒 | 🔴 | |
| L | _transferToExcluded | Private 🔒 | 🔴 | |
| L | _transferFromExcluded | Private 🔒 | 🔴 | |
| L | _transferBothExcluded | Private 🔒 | 🔴 | |
| L | _reflectFee | Private 🔒 | 🔴 | |
| L | _getValues | Private 🔒 | | |
| L | _getTValues | Private 🔒 | | |
| L | _getRValues | Private 🔒 | | |
| L | _getRate | Private 🔒 | | |
| L | _getCurrentSupply | Private 🔒 | | |

```

CONTRACT ASSESSMENT

```

| L | _takeLiquidity | Private 🔒 | ⚡ | |
| L | calculateTaxFee | Private 🔒 | |
| L | calculateLiquidityFee | Private 🔒 | |
| L | removeAllFee | Private 🔒 | ⚡ | |
| L | restoreAllFee | Private 🔒 | ⚡ | |
| L | isExcludedFromFee | Public ! | NO ! |
| L | excludeFromFee | Public ! | ⚡ | onlyOwner |
| L | includeInFee | Public ! | ⚡ | onlyOwner |
| L | setTaxFee | External ! | ⚡ | onlyOwner |
| L | setBuybackFee | External ! | ⚡ | onlyOwner |
| L | setMaxTxAmount | External ! | ⚡ | onlyOwner |
| L | setMarketingFee | External ! | ⚡ | onlyOwner |
| L | setNumTokensSellToAddToLiquidity | External ! | ⚡ | onlyOwner |
| L | setBuybackUpperLimit | External ! | ⚡ | onlyOwner |
| L | setMarketingAddress | External ! | ⚡ | onlyOwner |
| L | setSwapAndLiquifyEnabled | Public ! | ⚡ | onlyOwner |
| L | setBuyBackEnabled | Public ! | ⚡ | onlyOwner |
| L | presale | External ! | ⚡ | onlyOwner |
| L | transferToAddressETH | Private 🔒 | ⚡ | |
| L | <Receive Ether> | External ! | 💸 | NO ! |

```

Legend

Symbol	Meaning
:-----:	-----
⚡	Function can modify state
💸	Function is payable



STATIC ANALYSIS

```
Variable BuyBackToken._transferFromExcluded(address,address,uint256).xTransferAmount (contracts/Token.sol#1096) is too similar to BuyBackToken._transferToExcluded(address,address,uint256).tTransferAmount (contracts/Token.sol#1077)
Variable BuyBackToken._getValues(uint256).xTransferAmount (contracts/Token.sol#1149) is too similar to BuyBackToken._getTValues(uint256).tTransferAmount (contracts/Token.sol#1178)
Variable BuyBackToken._transferFromExcluded(address,address,uint256).xTransferAmount (contracts/Token.sol#1096) is too similar to BuyBackToken._getValues(uint256).tTransferAmount (contracts/Token.sol#1145)
Variable BuyBackToken._transferBothExcluded(address,address,uint256).xTransferAmount (contracts/Token.sol#1117) is too similar to BuyBackToken._getTValues(uint256).tTransferAmount (contracts/Token.sol#1170)
Variable BuyBackToken.reflectionFromToken(uint256,bool).xTransferAmount (contracts/Token.sol#865) is too similar to BuyBackToken._transferFromExcluded(address,address,uint256).tTransferAmount (contracts/Token.sol#1098)
Variable BuyBackToken._transferFromExcluded(address,address,uint256).xTransferAmount (contracts/Token.sol#1096) is too similar to BuyBackToken._transferStandard(address,address,uint256).tTransferAmount (contracts/Token.sol#1087)
Variable BuyBackToken._transferStandard(address,address,uint256).xTransferAmount (contracts/Token.sol#1055) is too similar to BuyBackToken._getTValues(uint256).tTransferAmount (contracts/Token.sol#1170)
Variable BuyBackToken._transferBothExcluded(address,address,uint256).xTransferAmount (contracts/Token.sol#1117) is too similar to BuyBackToken._transferToExcluded(address,address,uint256).tTransferAmount (contracts/Token.sol#1077)
Variable BuyBackToken._getValues(uint256).xTransferAmount (contracts/Token.sol#1149) is too similar to BuyBackToken._transferFromExcluded(address,address,uint256).tTransferAmount (contracts/Token.sol#1178)
Variable BuyBackToken._transferFromExcluded(address,address,uint256).xTransferAmount (contracts/Token.sol#1117) is too similar to BuyBackToken._transferToExcluded(address,address,uint256).tTransferAmount (contracts/Token.sol#1098)
Variable BuyBackToken._transferBothExcluded(address,address,uint256).xTransferAmount (contracts/Token.sol#1117) is too similar to BuyBackToken._getValues(uint256).tTransferAmount (contracts/Token.sol#1145)
Variable BuyBackToken.reflectionFromToken(uint256,bool).xTransferAmount (contracts/Token.sol#865) is too similar to BuyBackToken._transferBothExcluded(address,address,uint256).tTransferAmount (contracts/Token.sol#1119)
Variable BuyBackToken._transferStandard(address,address,uint256).xTransferAmount (contracts/Token.sol#1055) is too similar to BuyBackToken._transferToExcluded(address,address,uint256).tTransferAmount (contracts/Token.sol#1077)
Variable BuyBackToken._getValues(uint256).xTransferAmount (contracts/Token.sol#1149) is too similar to BuyBackToken._transferStandard(address,address,uint256).tTransferAmount (contracts/Token.sol#1087)
Variable BuyBackToken._transferBothExcluded(address,address,uint256).xTransferAmount (contracts/Token.sol#1117) is too similar to BuyBackToken._transferStandard(address,address,uint256).tTransferAmount (contracts/Token.sol#1087)
Variable BuyBackToken._transferBothExcluded(address,address,uint256).xTransferAmount (contracts/Token.sol#1117) is too similar to BuyBackToken._transferStandard(address,address,uint256).tTransferAmount (contracts/Token.sol#1087)
Variable BuyBackToken._transferStandard(address,address,uint256).xTransferAmount (contracts/Token.sol#1055) is too similar to BuyBackToken._getValues(uint256).tTransferAmount (contracts/Token.sol#1145)
Variable BuyBackToken._getValues(uint256).xTransferAmount (contracts/Token.sol#1149) is too similar to BuyBackToken._transferFromExcluded(address,address,uint256).tTransferAmount (contracts/Token.sol#1098)
Variable BuyBackToken._transferFromExcluded(address,address,uint256).xTransferAmount (contracts/Token.sol#1096) is too similar to BuyBackToken._transferBothExcluded(address,address,uint256).tTransferAmount (contracts/Token.sol#1119)
Variable BuyBackToken._transferStandard(address,address,uint256).xTransferAmount (contracts/Token.sol#1055) is too similar to BuyBackToken._transferStandard(address,address,uint256).tTransferAmount (contracts/Token.sol#1170)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#variable-names-too-similar
INFO:Detectors:
Loop condition 'i < _excluded.length' (contracts/Token.sol#1195) should use cached array length instead of referencing 'length' member of the storage array.
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#cache-array-length
INFO:Detectors:
BuyBackToken._decimals (contracts/Token.sol#640) should be immutable
BuyBackToken._tTotal (contracts/Token.sol#634) should be immutable
BuyBackToken.dappbuilderAddress (contracts/Token.sol#620) should be immutable
BuyBackToken.dappbuilderFee (contracts/Token.sol#654) 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

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

<https://testnet.bscscan.com/tx/0xbe12a1dfd1ada8bbf83632cde7f938ed92eab14b810d4a3f4a42e3bbe4abfeae>

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

<https://testnet.bscscan.com/tx/0x118b24bf1835dd14c1d4d425b4e657502b4b76f9073aa1dd3ace36f647763ca0>

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

<https://testnet.bscscan.com/tx/0x615018dfa3ffd4f57780a733f66dff95ac498c4b5bf6039c1a5366cb9456e44>

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

<https://testnet.bscscan.com/tx/0x7244136070c2fe1e15bca370757b9236cbe55f237d53e13eb705df119d95ccbd>

5- Buying when not excluded from fees (15% tax) (**passed**):

<https://testnet.bscscan.com/tx/0x63d6628e6957fb2a2f9409e247c85fee2d2e85a3a1930d50fb68a8c48c87805d>

6- Selling when not excluded from fees (15% tax) (**passed**):

<https://testnet.bscscan.com/tx/0x305f795ff5a38367afac11dcd49977df3206d6a8840cb2b818ccaa4bd26c01f4>



FUNCTIONAL TESTING

7- Transferring (15% tax) (passed):

<https://testnet.bscscan.com/tx/0x2176574ae2860dcd7d002e546cdab9568d0afd0a93f3da993ee0a4483d1b0187>

8- Internal swap(ETH sent to marketing wallet) (passed):

<https://testnet.bscscan.com/tx/0x305f795ff5a38367afac11dcd49977df3206d6a8840cb2b818ccaa4bd26c01f4>



MANUAL TESTING

Centralization – Maximum Buy/Sell/Transfer

Severity: High

function: setMaxTxAmount

Status: Open

Overview:

Owner is able to set `_maxTxAmount` to 0 which means no one would be able to buy/sell/transfer tokens (except owner)

```
function setMaxTxAmount(uint256 maxTxAmount) external onlyOwner {  
    _maxTxAmount = maxTxAmount;  
}
```

Suggestion

as a general suggestion, ensure that `_maxTxAmount` is greater than 0.1% of total supply:

```
function setMaxTxAmount(uint256 maxTxAmount) external onlyOwner {  
    require(_maxTxAmount >= totalSupply() / 1000);  
    _maxTxAmount = maxTxAmount;  
}
```



MANUAL TESTING

Centralization – Excessive Fees

Severity: High

function: setTaxFee – setBuybackFee - setMarketingFee

Status: Open

Overview:

Owner is able to set up to 100% tax on buy/sell/transfers

```
function setTaxFee(uint256 taxFee) external onlyOwner {  
    _taxFee = taxFee;  
}  
  
function setBuybackFee(uint256 _buybackFee) external onlyOwner {  
    buybackFee = _buybackFee;  
    _liquidityFee = buybackFee.add(marketingFee);  
}  
  
function setMarketingFee(uint256 _marketingFee) external onlyOwner {  
    require(  
        _marketingFee >= dappbuilderFee,  
        "Marketing fee can not be less than 5 percent"  
    );  
    marketingFee = _marketingFee;  
    _liquidityFee = buybackFee.add(marketingFee);  
}
```

Suggestion

Set an upper bound for maximum amount of buy/sell/transfer fees.

```
function setTaxFee(uint256 taxFee) external onlyOwner {  
    _taxFee = taxFee;  
    require(_taxFee + _liquidityFee <= 100);  
}  
  
function setBuybackFee(uint256 _buybackFee) external onlyOwner {  
    buybackFee = _buybackFee;  
    _liquidityFee = buybackFee.add(marketingFee);  
    require(_taxFee + _liquidityFee <= 100);  
}  
  
function setMarketingFee(uint256 _marketingFee) external onlyOwner {  
    require(  
        _marketingFee >= dappbuilderFee,  
        "Marketing fee can not be less than 5 percent"  
    );  
    marketingFee = _marketingFee;  
    _liquidityFee = buybackFee.add(marketingFee);  
    require(_taxFee + _liquidityFee <= 100);  
}
```



MANUAL TESTING

Logical – Setting swap threshold to zero

Severity: High

function: setNumTokensSellToAddToLiquidity

Status: Open

Overview:

Setting minimumTokensBeforeSwap to zero can disable internal swaps since contract will try to swap zero tokens (minimumTokensBeforeSwap) into BNB which will revert due to pancakeswap contract logic

```
function setNumTokensSellToAddToLiquidity(
    uint256 _minimumTokensBeforeSwap
) external onlyOwner {
    minimumTokensBeforeSwap = _minimumTokensBeforeSwap;
}
```

Suggestion

Ensure that minimumTokensBeforeSwap is greater than 0

```
function setNumTokensSellToAddToLiquidity(
    uint256 _minimumTokensBeforeSwap
) external onlyOwner {
    require( minimumTokensBeforeSwap > 0);
    minimumTokensBeforeSwap = _minimumTokensBeforeSwap;
}
```



MANUAL TESTING

Logical – Stuck ETH and tokens

Severity: Medium

function: ----

Status: Open

Overview:

There are no functions to withdraw stuck ETH or ERC20 tokens from the contract

Suggestion

Implement functions for withdrawing stuck ETH and tokens 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>
