Techrate1 Techrate

TechRate

AUDIT COMPANY

Smart Contract Security Audit

TechRate

August, 2021

Audit Details

Audited project

RealTrumpTokenV2

Deployer address

0x689f9fe7e376e5ad840254c40044c74c062ba14c

Client contacts:

RealTrumpTokenV2 team

Blockchain

Binance Smart Chain

Project website:

realtrumptoken.com

Disclaimer

This is a limited report on our findings based on our analysis, in accordance with good industry practice as at the date of this report, in relation to cybersecurity vulnerabilities and issues in the framework and algorithms based on smart contracts, the details of which are set out in this report. In order to get a full view of our analysis, it is crucial for you to read the full report. While we have done our best in conducting our analysis and producing this report, it is important to note that you should not rely on this report and cannot claim against us on the basis of what it says or doesn't say, or how we produced it, and it is important for you to conduct your own independent investigations before making any decisions. We go into more detail on this in the below disclaimer below – please make sure to read it in full.

DISCLAIMER: By reading this report or any part of it, you agree to the terms of this disclaimer. If you do not agree to the terms, then please immediately cease reading this report, and delete and destroy any and all copies of this report downloaded and/or printed by you. This report is provided for information purposes only and on a non-reliance basis, and does not constitute investment advice. No one shall have any right to rely on the report or its contents, and TechRate and its affiliates (including holding companies, shareholders, subsidiaries, employees, directors, officers and other representatives) (TechRate) owe no duty of care towards you or any other person, nor does TechRate make any warranty or representation to any person on the accuracy or completeness of the report. The report is provided "as is", without any conditions, warranties or other terms of any kind except as set out in this disclaimer, and TechRate hereby excludes all representations, warranties, conditions and other terms (including, without limitation, the warranties implied by law of satisfactory quality, fitness for purpose and the use of reasonable care and skill) which, but for this clause, might have effect in relation to the report. Except and only to the extent that it is prohibited by law, TechRate hereby excludes all liability and responsibility, and neither you nor any other person shall have any claim against TechRate, for any amount or kind of loss or damage that may result to you or any other person (including without limitation, any direct, indirect, special, punitive, consequential or pure economic loss or damages, or any loss of income, profits, goodwill, data, contracts, use of money, or business interruption, and whether in delict, tort (including without limitation negligence), contract, breach of statutory duty, misrepresentation (whether innocent or negligent) or otherwise under any claim of any nature whatsoever in any jurisdiction) in any way arising from or connected with this report and the use, inability to use or the results of use of this report, and any reliance on this report.

The analysis of the security is purely based on the smart contracts alone. No applications or operations were reviewed for security. No product code has been reviewed.

Background

TechRate was commissioned by RealTrumpTokenV2 to perform an audit of smart contracts:

https://bscscan.com/address/0xf41082c4cb71fb4628a9b17214b2624e0e2048a9#code

The purpose of the audit was to achieve the following:

- Ensure that the smart contract functions as intended.
- Identify potential security issues with the smart contract.

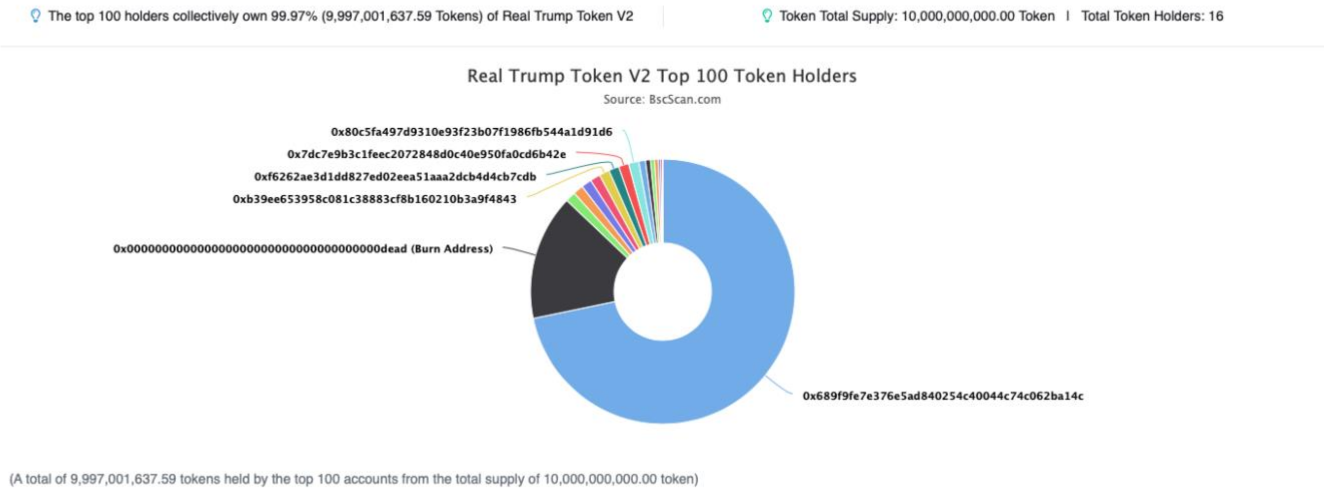
The information in this report should be used to understand the risk exposure of the smart contract, and as a guide to improve the security posture of the smart contract by remediating the issues that were identified.

Contracts Details

Token contract details for 31.08.2021

Contract name	RealTrumpTokenV2
Contract address	0xF41082C4CB71FB4628a9b17214B2624e0e2048a9
Total supply	10,000,000,000
Token ticker	RTTv2
Decimals	9
Token holders	16
Transactions count	17
Top 100 holders dominance	99.97%
Liquidity fee	0
Tax fee	0
Total fees	600000000000000
Uniswap V2 pair	0x4f3ea40fe0663e8d8bd2affd2f794b348dcb2191
Contract deployer address	0x689f9fe7e376e5ad840254c40044c74c062ba14c
Contract's current owner address	0x689f9fe7e376e5ad840254c40044c74c062ba14c

RealTrumpTokenV2 Token Distribution



The top 100 holders collectively own 99.97% (9,997,001,637.59 Tokens) of Real Trump Token V2

Token Total Supply: 10,000,000,000.00 Token | Total Token Holders: 16

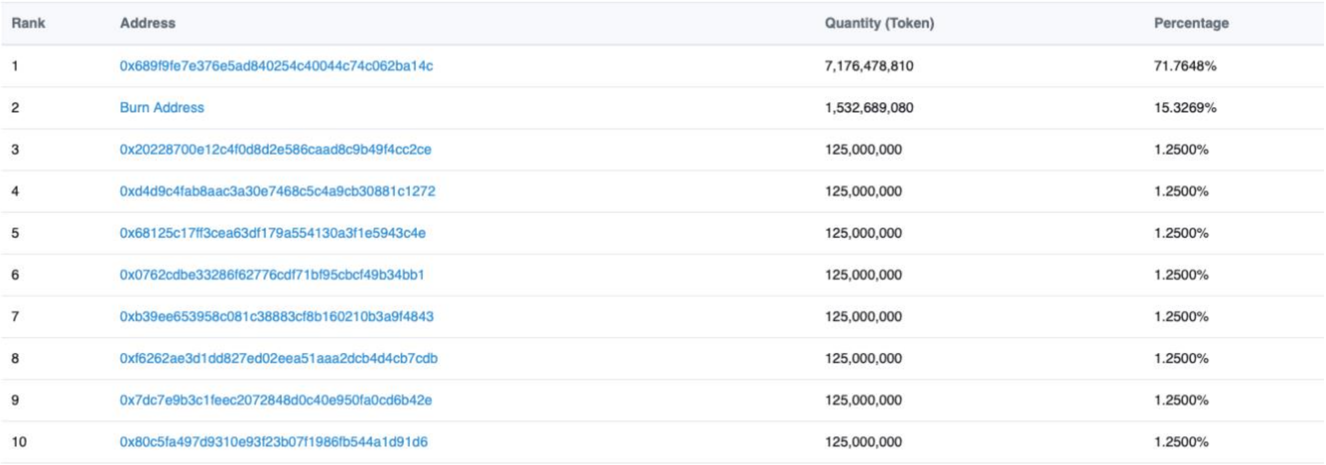
Real Trump Token V2 Top 100 Token Holders

Source: BscScan.com

0x80c5fa497d9310e93f23b07f1986fb544a1d91d6
0x7dc7e9b3c1feec2072848d0c40e950fa0cd6b42e
0xf6262ae3d1dd827ed02eea51aaa2dcb4d4cb7cdb
0xb39ee653958c081c38883cf8b160210b3a9f4843
0x000000000000000000000000000000000000dead (Burn Address)
0x689f9fe7e376e5ad840254c40044c74c062ba14c

(A total of 9,997,001,637.59 tokens held by the top 100 accounts from the total supply of 10,000,000,000.00 token)

RealTrumpTokenV2 Top 10 Token Holders



Rank	Address	Quantity (Token)	Percentage
1	0x689f9fe7e376e5ad840254c40044c74c062ba14c	7,176,478,810	71.7648%
2	Burn Address	1,532,689,080	15.3269%
3	0x20228700e12c4f0d8d2e586caad8c9b49f4cc2ce	125,000,000	1.2500%
4	0xd4d9c4fab8aac3a30e7468c5c4a9cb30881c1272	125,000,000	1.2500%
5	0x68125c17ff3cea63df179a554130a3f1e5943c4e	125,000,000	1.2500%
6	0x0762cdbe33286f62776cdf71bf95cbcf49b34bb1	125,000,000	1.2500%
7	0xb39ee653958c081c38883cf8b160210b3a9f4843	125,000,000	1.2500%
8	0xf6262ae3d1dd827ed02eea51aaa2dcb4d4cb7cdb	125,000,000	1.2500%
9	0x7dc7e9b3c1feec2072848d0c40e950fa0cd6b42e	125,000,000	1.2500%
10	0x80c5fa497d9310e93f23b07f1986fb544a1d91d6	125,000,000	1.2500%

Contract functions details

+ Context
  - [Int] _msgSender
  - [Int] _msgData

+ [Int] IERC20
  - [Ext] totalSupply
  - [Ext] balanceOf
  - [Ext] transfer #
  - [Ext] allowance
  - [Ext] approve #
  - [Ext] transferFrom #

+ [Lib] SafeMath
  - [Int] add
  - [Int] sub
  - [Int] sub
  - [Int] mul
  - [Int] div
  - [Int] div
  - [Int] mod
  - [Int] mod

+ [Lib] Address
  - [Int] isContract
  - [Int] sendValue #
  - [Int] functionCall #
  - [Int] functionCall #
  - [Int] functionCallWithValue #
  - [Int] functionCallWithValue #
  - [Prv] _functionCallWithValue #

+ Ownable (Context)
  - [Int] <Constructor> #
  - [Pub] owner
  - [Pub] transferOwnership #
    - modifiers: onlyOwner
  - [Pub] getUnlockTime
  - [Pub] getTime
  - [Pub] lock #
    - modifiers: onlyOwner
  - [Pub] unlock #

+ [Int] IUniswapV2Factory
  - [Ext] feeTo
  - [Ext] feeToSetter
  - [Ext] getPair
  - [Ext] allPairs
  - [Ext] allPairsLength
  - [Ext] createPair #
  - [Ext] setFeeTo #
  - [Ext] setFeeToSetter #

```
+ [Int] IUniswapV2Pair
  - [Ext] name
  - [Ext] symbol
  - [Ext] decimals
  - [Ext] totalSupply
  - [Ext] balanceOf
  - [Ext] allowance
  - [Ext] approve #
  - [Ext] transfer #
  - [Ext] transferFrom #
  - [Ext] DOMAIN_SEPARATOR
  - [Ext] PERMIT_TYPEHASH
  - [Ext] nonces
  - [Ext] permit #
  - [Ext] MINIMUM_LIQUIDITY
  - [Ext] factory
  - [Ext] token0
  - [Ext] token1
  - [Ext] getReserves
  - [Ext] price0CumulativeLast
  - [Ext] price1CumulativeLast
  - [Ext] kLast
  - [Ext] burn #
  - [Ext] swap #
  - [Ext] skim #
  - [Ext] sync #
  - [Ext] initialize #

+ [Int] IUniswapV2Router01
  - [Ext] factory
  - [Ext] WETH
  - [Ext] addLiquidity #
  - [Ext] addLiquidityETH ($)
  - [Ext] removeLiquidity #
  - [Ext] removeLiquidityETH #
  - [Ext] removeLiquidityWithPermit #
  - [Ext] removeLiquidityETHWithPermit #
  - [Ext] swapExactTokensForTokens #
  - [Ext] swapTokensForExactTokens #
  - [Ext] swapExactETHForTokens ($)
  - [Ext] swapTokensForExactETH #
  - [Ext] swapExactTokensForETH #
  - [Ext] swapETHForExactTokens ($)
  - [Ext] quote
  - [Ext] getAmountOut
  - [Ext] getAmountIn
  - [Ext] getAmountsOut
  - [Ext] getAmountsIn

+ [Int] IUniswapV2Router02 (IUniswapV2Router01)
  - [Ext] removeLiquidityETHSupportingFeeOnTransferTokens #
  - [Ext] removeLiquidityETHWithPermitSupportingFeeOnTransferTokens #
  - [Ext] swapExactTokensForTokensSupportingFeeOnTransferTokens #
  - [Ext] swapExactETHForTokensSupportingFeeOnTransferTokens ($)
```

  - [Ext] swapExactTokensForETHSupportingFeeOnTransferTokens #

+ RealTrumpTokenV2 (Context, IERC20, Ownable)
  - [Pub] <Constructor> #
  - [Pub] name
  - [Pub] symbol
  - [Pub] decimals
  - [Pub] totalSupply
  - [Pub] balanceOf
  - [Pub] transfer #
  - [Pub] allowance
  - [Pub] approve #
  - [Pub] transferFrom #
  - [Pub] increaseAllowance #
  - [Pub] decreaseAllowance #
  - [Pub] isExcludedFromReward
  - [Pub] totalFees
  - [Pub] minimumTokensBeforeSwapAmount
  - [Pub] deliver #
  - [Pub] reflectionFromToken
  - [Pub] tokenFromReflection
  - [Pub] excludeFromReward #
    - modifiers: onlyOwner
  - [Ext] includeInReward #
    - modifiers: onlyOwner
  - [Prv] _approve #
  - [Prv] _transfer #
  - [Prv] swapAndLiquify #
    - modifiers: lockTheSwap
  - [Prv] swapTokensForEth #
  - [Prv] addLiquidity #
  - [Prv] _tokenTransfer #
  - [Prv] _transferStandard #
  - [Prv] _transferToExcluded #
  - [Prv] _transferFromExcluded #
  - [Prv] _transferBothExcluded #
  - [Prv] _reflectFee #
  - [Prv] _getValues
  - [Prv] _getTValues
  - [Prv] _getRValues
  - [Prv] _getRate
  - [Prv] _getCurrentSupply
  - [Prv] _takeLiquidity #
  - [Prv] calculateTaxFee
  - [Prv] calculateLiquidityFee
  - [Prv] removeAllFee #
  - [Prv] restoreAllFee #
  - [Pub] isExcludedFromFee
  - [Pub] excludeFromFee #
    - modifiers: onlyOwner
  - [Pub] isExcludedFromMaxTxAmount
  - [Pub] excludeFromMaxTxAmount #
    - modifiers: onlyOwner
  - [Pub] includeToMaxTxAmount #
    - modifiers: onlyOwner

  - [Pub] includeInFee #
    - modifiers: onlyOwner
  - [Ext] setTaxFeePercent #
    - modifiers: onlyOwner
  - [Ext] setLiquidityFeePercent #
    - modifiers: onlyOwner
  - [Ext] setMaxTxAmount #
    - modifiers: onlyOwner
  - [Ext] setNumTokensSellToAddToLiquidity #
    - modifiers: onlyOwner
  - [Pub] setSwapAndLiquifyEnabled #
    - modifiers: onlyOwner
  - [Ext] setCharityAddress #
    - modifiers: onlyOwner
  - [Ext] setUtilityAddress #
    - modifiers: onlyOwner
  - [Ext] setDevelopmentAddress #
    - modifiers: onlyOwner
  - [Ext] setDonationAddress #
    - modifiers: onlyOwner
  - [Pub] transferContractBalance #
    - modifiers: onlyOwner
  - [Prv] transferOutETH #
  - [Ext] <Fallback> ($)

($) = payable function
# = non-constant function

Issues Checking Status

	Issue description	Checking status
1.	Compiler errors.	Passed
2.	Race conditions and Reentrancy. Cross-function race conditions.	Passed
3.	Possible delays in data delivery.	Passed
4.	Oracle calls.	Passed
5.	Front running.	Passed
6.	Timestamp dependence.	Passed
7.	Integer Overflow and Underflow.	Passed
8.	DoS with Revert.	Passed
9.	DoS with block gas limit.	Low issues
10.	Methods execution permissions.	Passed
11.	Economy model of the contract.	Passed
12.	The impact of the exchange rate on the logic.	Passed
13.	Private user data leaks.	Passed
14.	Malicious Event log.	Passed
15.	Scoping and Declarations.	Passed
16.	Uninitialized storage pointers.	Passed
17.	Arithmetic accuracy.	Passed
18.	Design Logic.	Passed
19.	Cross-function race conditions.	Passed
20.	Safe Open Zeppelin contracts implementation and usage.	Passed
21.	Fallback function security.	Passed

Security Issues

High Severity Issues

No high severity issues found.

Medium Severity Issues

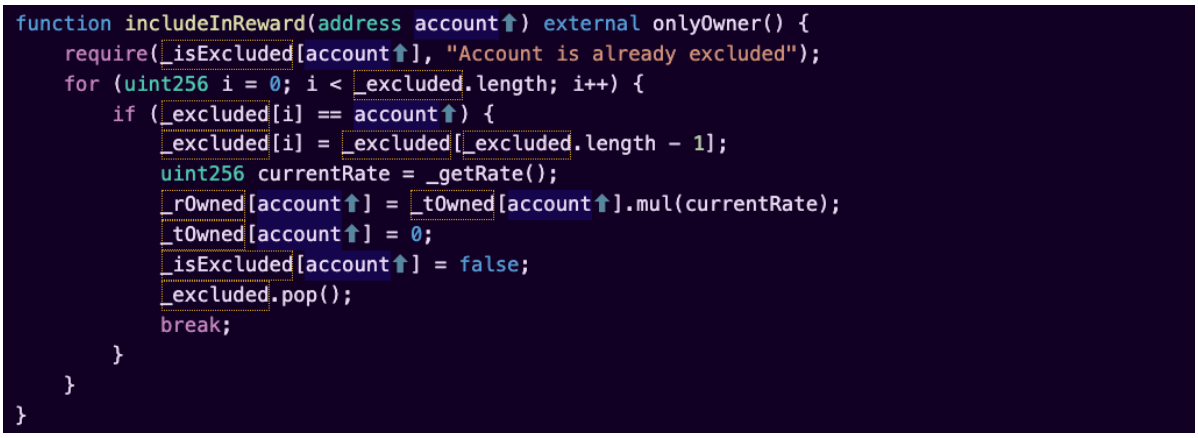
No medium severity issues found.

Low Severity Issues

1. Out of gas

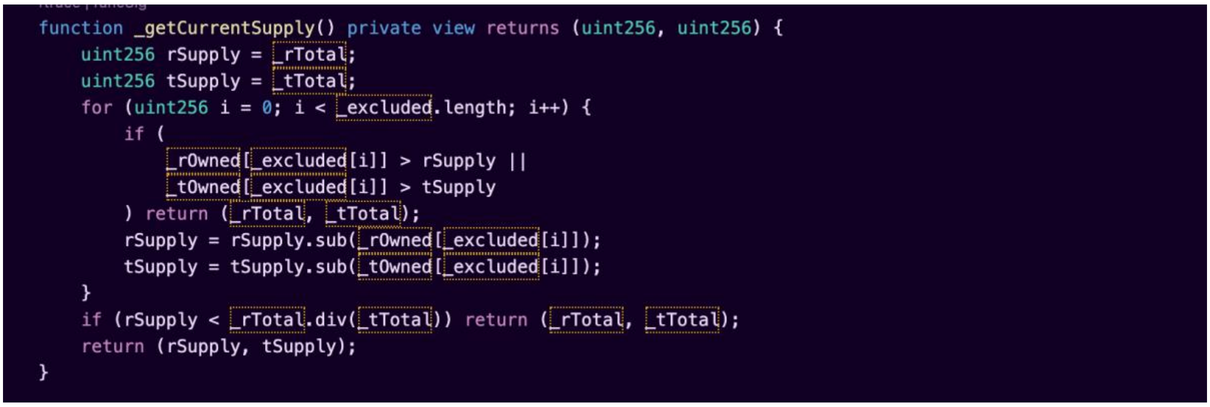
Issue:

- The function includeInReward() uses the loop to find and remove addresses from the _excluded list. Function will be aborted with OUT_OF_GAS exception if there will be a long excluded addresses list.



```
function includeInReward(address account) external onlyOwner() {
    require(_isExcluded[account], "Account is already excluded");
    for (uint256 i = 0; i < _excluded.length; i++) {
        if (_excluded[i] == account) {
            _excluded[i] = _excluded[_excluded.length - 1];
            uint256 currentRate = _getRate();
            _rOwned[account] = _tOwned[account].mul(currentRate);
            _tOwned[account] = 0;
            _isExcluded[account] = false;
            _excluded.pop();
            break;
        }
    }
}
```

- The function _getCurrentSupply also uses the loop for evaluating total supply. It also could be aborted with OUT_OF_GAS exception if there will be a long excluded addresses list.



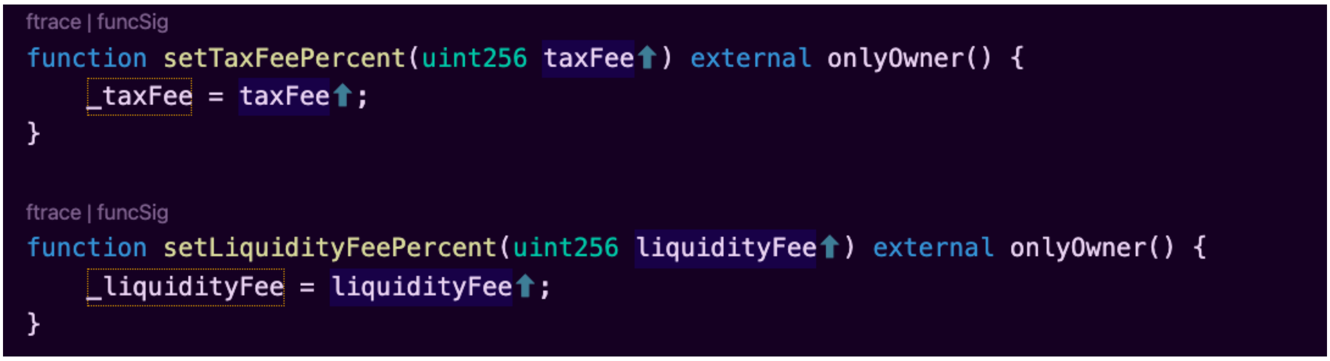
```
function _getCurrentSupply() private view returns (uint256, uint256) {
    uint256 rSupply = _rTotal;
    uint256 tSupply = _tTotal;
    for (uint256 i = 0; i < _excluded.length; i++) {
        if (
            _rOwned[_excluded[i]] > rSupply ||
            _tOwned[_excluded[i]] > tSupply
        ) return (_rTotal, _tTotal);
        rSupply = rSupply.sub(_rOwned[_excluded[i]]);
        tSupply = tSupply.sub(_tOwned[_excluded[i]]);
    }
    if (rSupply < _rTotal.div(_tTotal)) return (_rTotal, _tTotal);
    return (rSupply, tSupply);
}
```

Recommendation:

Check that the excluded array length is not too big.

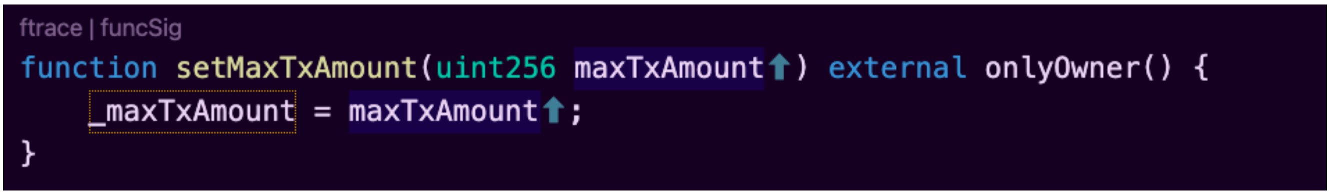
Owner privileges (In the period when the owner is not renounced)

- Owner can change tax and liquidity fees.



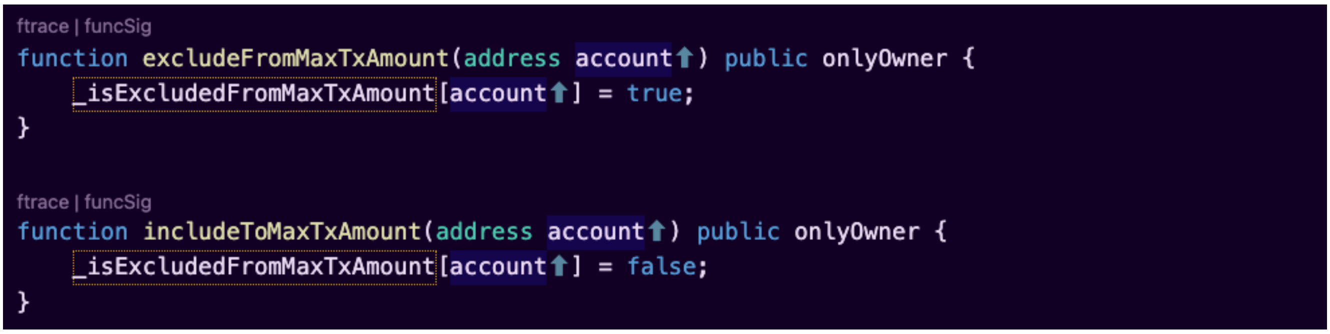
```
function setTaxFeePercent(uint256 taxFee) external onlyOwner() {
    _taxFee = taxFee;
}

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

- Owner can change maximum transaction amount.



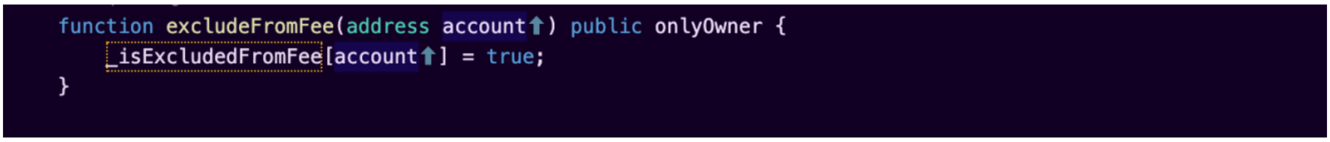
```
function setMaxTxAmount(uint256 maxTxAmount) external onlyOwner() {
    _maxTxAmount = maxTxAmount;
}
```

- Owner can include/exclude from maximum transaction amount.



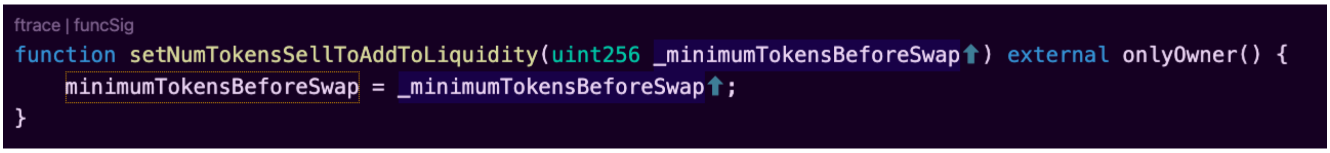
```
function excludeFromMaxTxAmount(address account) public onlyOwner {
    _isExcludedFromMaxTxAmount[account] = true;
}

function includeToMaxTxAmount(address account) public onlyOwner {
    _isExcludedFromMaxTxAmount[account] = false;
}
```

- Owner can exclude from the fee.



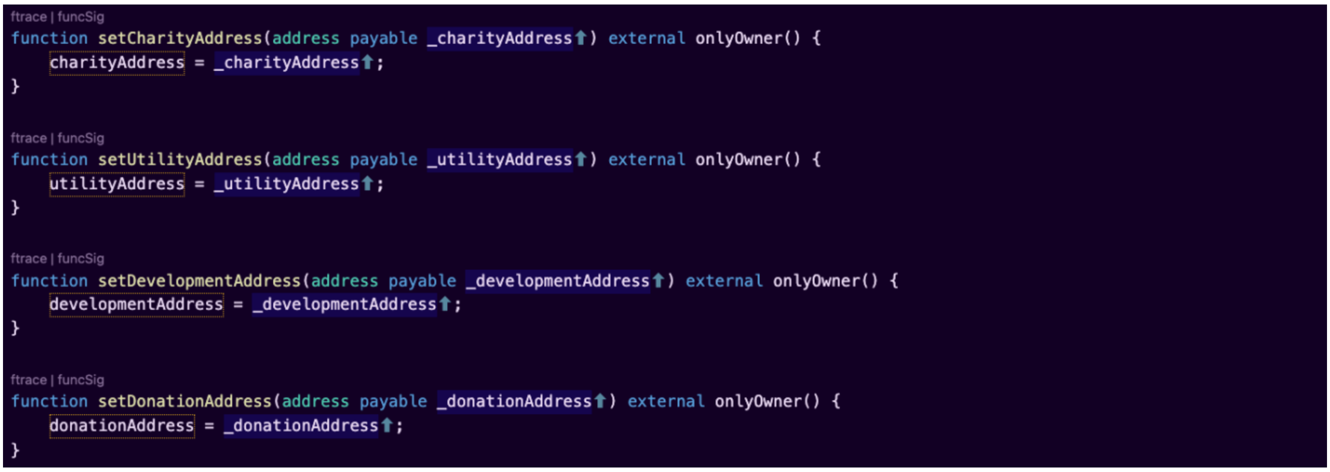
```
function excludeFromFee(address account) public onlyOwner {
    _isExcludedFromFee[account] = true;
}
```

- Owner can change minimum number of tokens to add to liquidity.



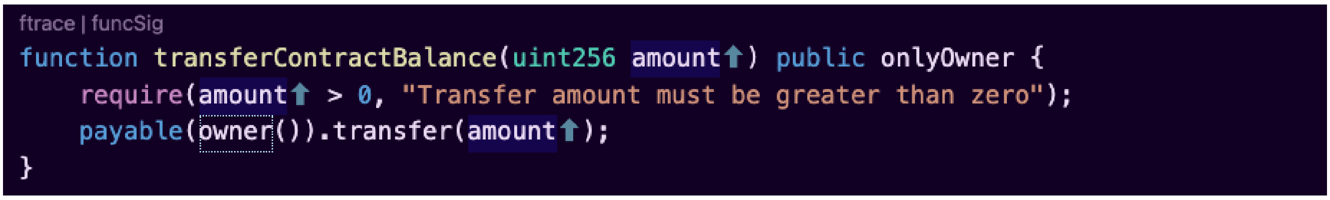
```
function setNumTokensSellToAddToLiquidity(uint256 _minimumTokensBeforeSwap) external onlyOwner() {
    minimumTokensBeforeSwap = _minimumTokensBeforeSwap;
}
```

- Owner can change charity, utility, development and donation address.



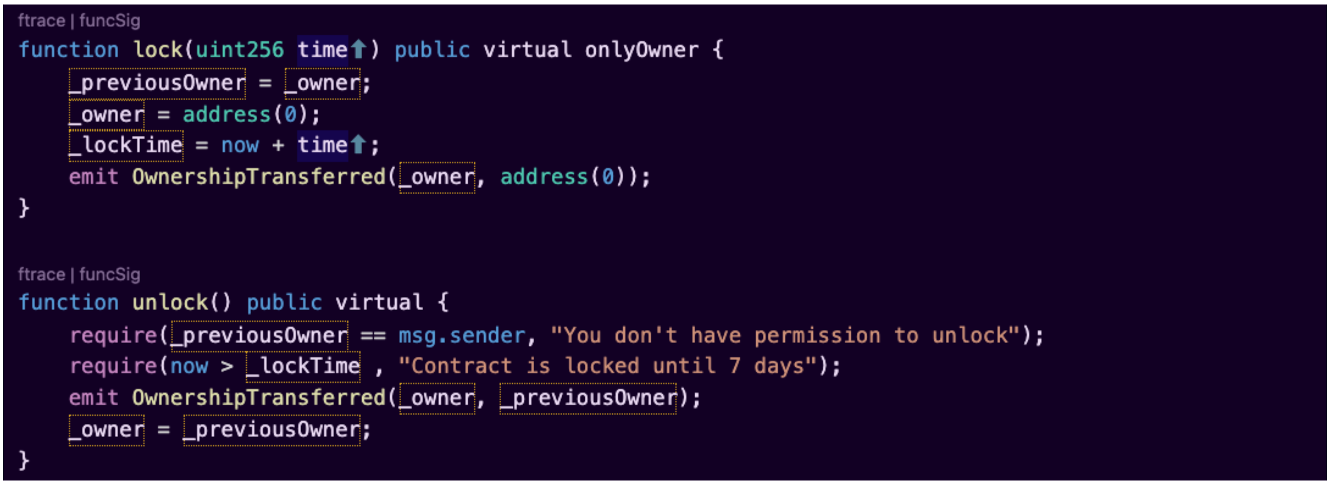
```
ftrace | funcSig
function setCharityAddress(address payable _charityAddress) external onlyOwner() {
    charityAddress = _charityAddress;
}

ftrace | funcSig
function setUtilityAddress(address payable _utilityAddress) external onlyOwner() {
    utilityAddress = _utilityAddress;
}

ftrace | funcSig
function setDevelopmentAddress(address payable _developmentAddress) external onlyOwner() {
    developmentAddress = _developmentAddress;
}

ftrace | funcSig
function setDonationAddress(address payable _donationAddress) external onlyOwner() {
    donationAddress = _donationAddress;
}
```

- Owner can withdraw contract BNBs.



```
ftrace | funcSig
function transferContractBalance(uint256 amount) public onlyOwner {
    require(amount > 0, "Transfer amount must be greater than zero");
    payable(owner()).transfer(amount);
}
```

- Owner can lock and unlock. By the way, using these functions the owner could retake privileges even after the ownership was renounced.



```
ftrace | funcSig
function lock(uint256 time) public virtual onlyOwner {
    _previousOwner = _owner;
    _owner = address(0);
    _lockTime = now + time;
    emit OwnershipTransferred(_owner, address(0));
}

ftrace | funcSig
function unlock() public virtual {
    require(_previousOwner == msg.sender, "You don't have permission to unlock");
    require(now > _lockTime , "Contract is locked until 7 days");
    emit OwnershipTransferred(_owner, _previousOwner);
    _owner = _previousOwner;
}
```

Conclusion

Smart contracts contain low severity issues! Liquidity pair contract's security is not checked due to out of scope. The further transfers and operations with the funds raise are not related to this particular contract.

Liquidity locking details NOT provided by the team.

TechRate note:

Please check the disclaimer above and note, the audit makes no statements or warranties on business model, investment attractiveness or code sustainability. The report is provided for the only contract mentioned in the report and does not include any other potential contracts deployed by Owner.