Techrate1 Techrate

TechRate

AUDIT COMPANY

Smart Contract Security Audit

TechRate

November, 2021

Audit Details

Audited project

ShinChan Token

Deployer address

0xfa2e8c05d198412156c0c5dcb022129612cba1fb

Client contacts:

ShinChan Token team

Blockchain

Ethereum

Project website:

https://shinchantoken.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 ShinChan Token to perform an audit of smart contracts:

https://etherscan.io/address/0xbaa9af8a83500ac4137c555b9e58ccb3e1f2269d#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.

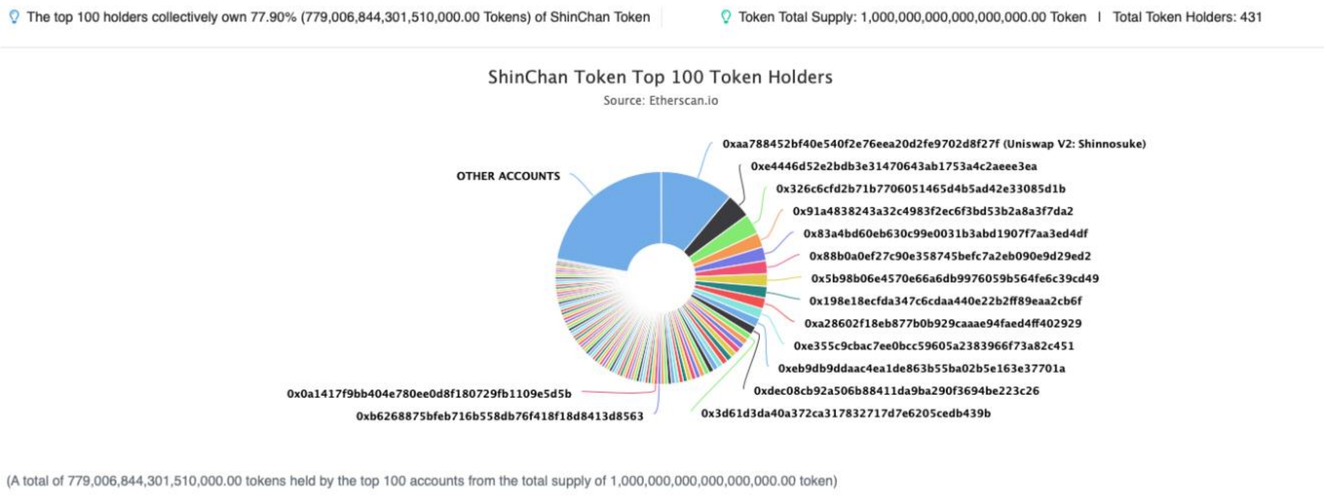
The information in this report should be used to understand the risk exposure of the smart contract, and as a guide to improve the security posture of the smart contract by remediating the issues that were identified.

Contracts Details

Token contract details for 10.11.2021

Contract name	ShinChan Token
Contract address	0xBaa9AF8a83500Ac4137C555b9E58CCB3e1f2269D
Total supply	1,000,000,000,000,000,000
Token ticker	Shinnosuke
Decimals	9
Token holders	431
Transactions count	924
Top 100 holders dominance	77.90%
Marketing fee	11
Tax fee	1
Total fees	23140055870309735118421461
Uniswap V2 pair	0xaa788452bf40e540f2e76eea20d2fe9702d8f27f
Contract deployer address	0xfa2e8c05d198412156c0c5dcb022129612cba1fb
Contract's current owner address	0x0000000000000000000000000000000000000000

ShinChan Token Token Distribution



The top 100 holders collectively own 77.90% (779,006,844,301,510,000.00 Tokens) of ShinChan Token | Token Total Supply: 1,000,000,000,000,000,000.00 Token | Total Token Holders: 431

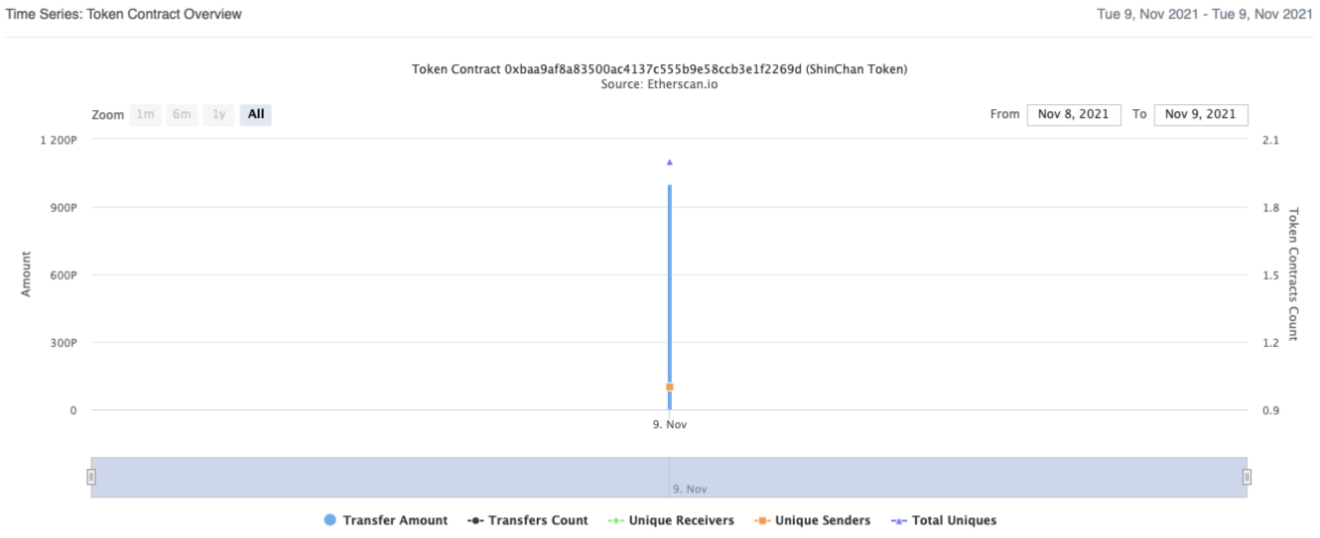
ShinChan Token Top 100 Token Holders

Source: Etherscan.io

OTHER ACCOUNTS
0xaa788452bf40e540f2e76eea20d2fe9702d8f27f (Uniswap V2: Shinnosuke)
0xe4446d52e2bdb3e31470643ab1753a4c2aeee3ea
0x326c6cfd2b71b7706051465d4b5ad42e33085d1b
0x91a4838243a32c4983f2ec6f3bd53b2a8a3f7da2
0x83a4bd60eb630c99e0031b3abd1907f7aa3ed4df
0x88b0a0ef27c90e358745befc7a2eb090e9d29ed2
0x5b98b06e4570e66a6db9976059b564fe6c39cd49
0x198e18ecfda347c6cdaa440e22b2ff89eaa2cb6f
0xa28602f18eb877b0b929caaae94faed4ff402929
0xe355c9cbac7ee0bcc59605a2383966f73a82c451
0xeb9db9ddaac4ea1de863b55ba02b5e163e37701a
0xdec08cb92a506b88411da9ba290f3694be223c26
0x3d61d3da40a372ca317832717d7e6205cedb439b
0x0a1417f9bb404e780ee0d8f180729fb1109e5d5b
0xb6268875bfeb716b558db76f418f18d8413d8563

(A total of 779,006,844,301,510,000.00 tokens held by the top 100 accounts from the total supply of 1,000,000,000,000,000,000.00 token)

ShinChan Token Contract Interaction Details



Time Series: Token Contract Overview

Tue 9, Nov 2021 - Tue 9, Nov 2021

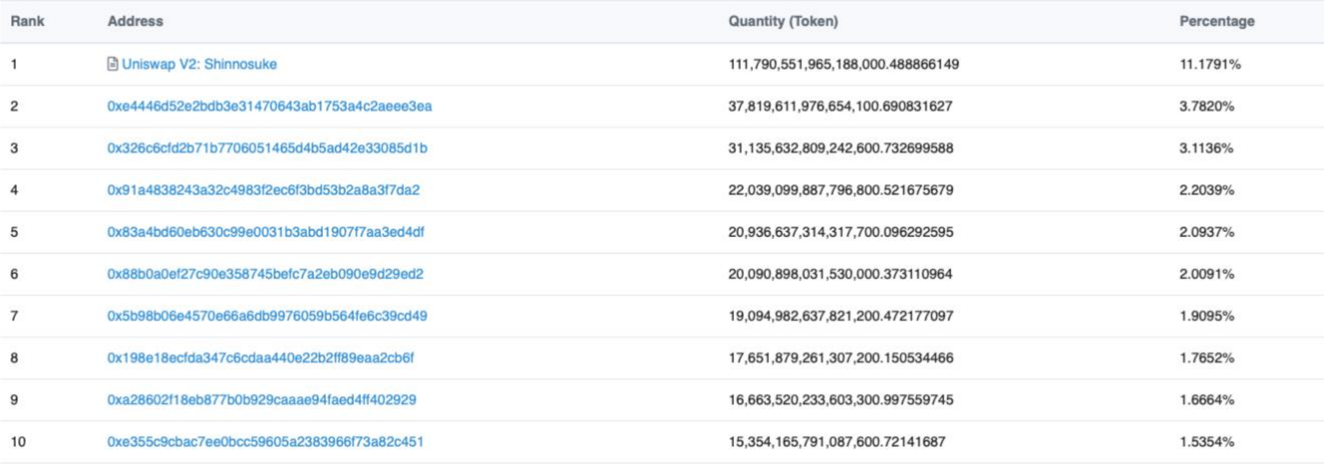
Token Contract 0xbaa9af8a83500ac4137c555b9e58ccb3e1f2269d (ShinChan Token)

Source: Etherscan.io

Zoom 1m 6m 1y **All**

From Nov 8, 2021 To Nov 9, 2021

ShinChan Token Top 10 Token Holders



Rank	Address	Quantity (Token)	Percentage
1	Uniswap V2: Shinnosuke	111,790,551,965,188,000.488866149	11.1791%
2	0xe4446d52e2bdb3e31470643ab1753a4c2aeee3ea	37,819,611,976,654,100.690831627	3.7820%
3	0x326c6cfd2b71b7706051465d4b5ad42e33085d1b	31,135,632,809,242,600.732699588	3.1136%
4	0x91a4838243a32c4983f2ec6f3bd53b2a8a3f7da2	22,039,099,887,796,800.521675679	2.2039%
5	0x83a4bd60eb630c99e0031b3abd1907f7aa3ed4df	20,936,637,314,317,700.096292595	2.0937%
6	0x88b0a0ef27c90e358745befc7a2eb090e9d29ed2	20,090,898,031,530,000.373110964	2.0091%
7	0x5b98b06e4570e66a6db9976059b564fe6c39cd49	19,094,982,637,821,200.472177097	1.9095%
8	0x198e18ecfda347c6cdaa440e22b2ff89eaa2cb6f	17,651,879,261,307,200.150534466	1.7652%
9	0xa28602f18eb877b0b929caaae94faed4ff402929	16,663,520,233,603,300.997559745	1.6664%
10	0xe355c9cbac7ee0bcc59605a2383966f73a82c451	15,354,165,791,087,600.72141687	1.5354%

Contract functions details

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

+ [Lib] SafeMath
  - [Int] tryAdd
  - [Int] trySub
  - [Int] tryMul
  - [Int] tryDiv
  - [Int] tryMod
  - [Int] add
  - [Int] sub
  - [Int] mul
  - [Int] div
  - [Int] mod
  - [Int] sub
  - [Int] div
  - [Int] mod

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

+ [Lib] Address
  - [Int] isContract
  - [Int] sendValue #
  - [Int] functionCall #
  - [Int] functionCall #
  - [Int] functionCallWithValue #
  - [Int] functionCallWithValue #
  - [Int] functionStaticCall
  - [Int] functionStaticCall
  - [Int] functionDelegateCall #
  - [Int] functionDelegateCall #
  - [Prv] _verifyCallResult

+ Ownable (Context)
  - [Pub] <Constructor> #
  - [Pub] owner
  - [Pub] renounceOwnership #
    - modifiers: onlyOwner
  - [Pub] transferOwnership #
    - modifiers: onlyOwner
  - [Pub] isAdmin
  - [Pub] isOwner
  - [Pub] isAdminOrOwner
  - [Pub] setAdmin #

    - modifiers: onlyOwner

+ [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] mint #
  - [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 #

+ ShinChanToken (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] reflectionFromToken
  - [Pub] tokenFromReflection
  - [Pub] excludeFromReward #
    - modifiers: onlyOwner
  - [Ext] includeInReward #
    - modifiers: onlyOwner
  - [Pub] excludeFromFee #
    - modifiers: onlyAdmins
  - [Pub] includeInFee #
    - modifiers: onlyAdmins
  - [Prv] removeAllFee #
  - [Prv] restoreAllFee #
  - [Ext] <Fallback> ($)
  - [Prv] _reflectFee #
  - [Ext] addToBlackList #
    - modifiers: onlyOwner
  - [Ext] removeFromBlackList #
    - modifiers: onlyOwner
  - [Prv] _getValues
  - [Prv] _getTValues
  - [Prv] _getRValues
  - [Prv] _getRate
  - [Prv] _getCurrentSupply
  - [Prv] _takeMarketing #
  - [Prv] calculateTaxFee
  - [Prv] calculateMarketingFee

- **[Pub]** isExcludedFromFee
- **[Prv]** _approve **#**
- **[Prv]** _transfer **#**
- **[Prv]** setFees **#**
- **[Prv]** SwapAndSend **#**
  - modifiers: lockTheSwap
- **[Prv]** _tokenTransfer **#**
- **[Prv]** _transferStandard **#**
- **[Prv]** _transferToExcluded **#**
- **[Prv]** _transferFromExcluded **#**
- **[Prv]** _transferBothExcluded **#**
- **[Ext]** setDefaultMarketingFee **#**
  - modifiers: onlyAdmins
- **[Ext]** setReflectionFee **#**
  - modifiers: onlyAdmins
- **[Ext]** setMarketingFee4Sellers **#**
  - modifiers: onlyAdmins
- **[Pub]** setFeesOnSellersAndBuyers **#**
  - modifiers: onlyAdmins
- **[Ext]** setLaunchingTime **#**
  - modifiers: onlyOwner
- **[Pub]** setSwapAndSendEnabled **#**
  - modifiers: onlyOwner
- **[Pub]** setnumTokensToExchangeForMarketing **#**
  - modifiers: onlyOwner
- **[Ext]** _setMarketingWallet **#**
  - modifiers: onlyAdmins
- **[Ext]** _setMaxTxAmount **#**
  - modifiers: onlyOwner
- **[Ext]** setMaxTxPercent **#**
  - modifiers: onlyOwner

($) = 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

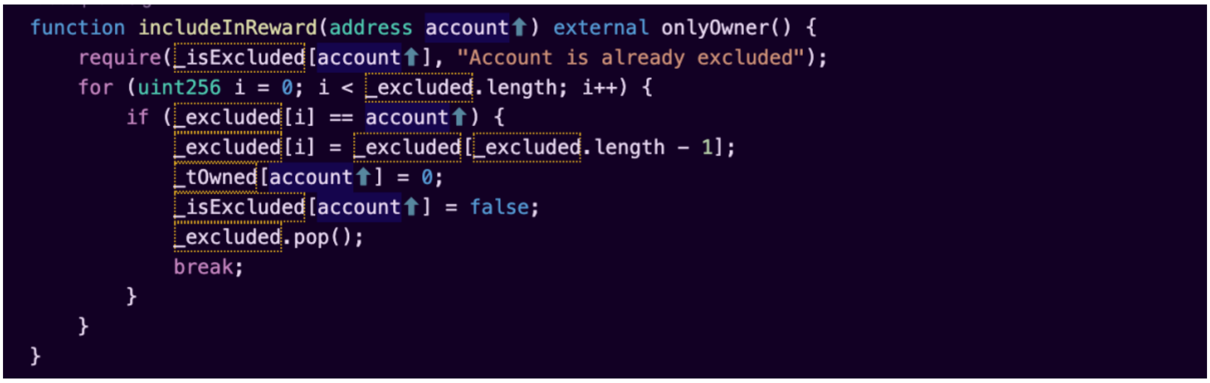
No medium severity issues found.

Low Severity Issues

1. Out of gas

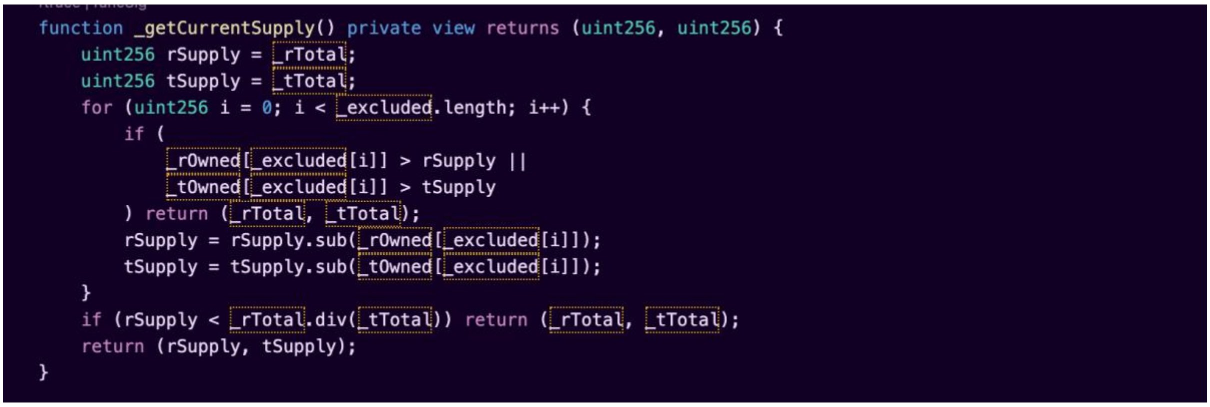
Issue:

- 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];
            _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.

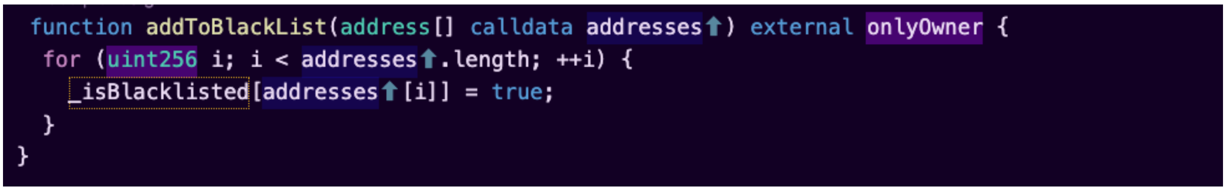


```
function _getCurrentSupply() private view returns (uint256, uint256) {
    uint256 rSupply = _rTotal;
    uint256 tSupply = _tTotal;
    for (uint256 i = 0; i < _excluded.length; i++) {
        if (
            _rOwned[_excluded[i]] > rSupply ||
            _tOwned[_excluded[i]] > tSupply
      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

- The function addToBlackList() uses the loop to add addresses from function argument to blacklist. It also could be aborted with OUT_OF_GAS exception if there will be a long addresses list.



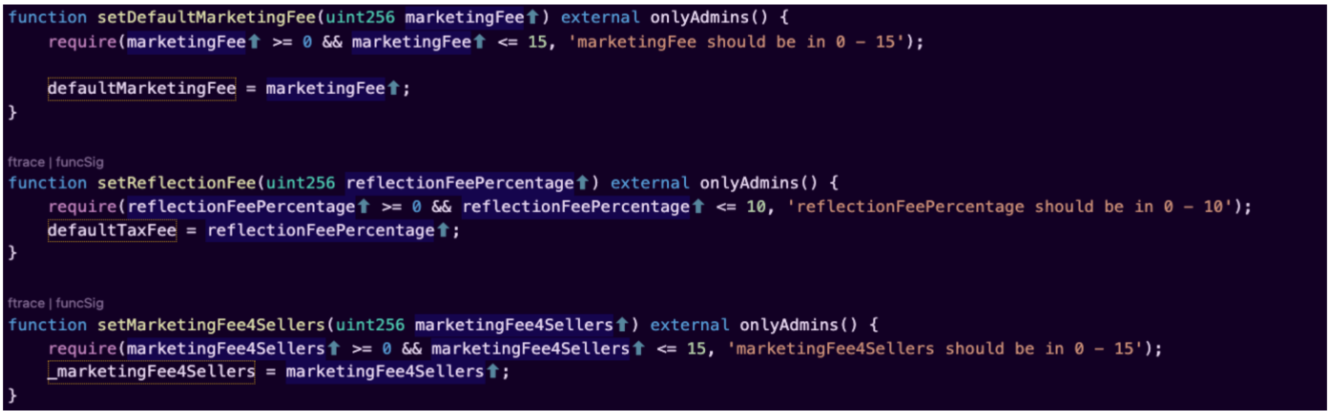
```
 function addToBlackList(address[] calldata addresses) external onlyOwner {
  for (uint256 i; i < addresses.length; ++i) {
    _isBlacklisted[addresses[i]] = true;
  }
}
```

Recommendation:

Check that the array length is not too big.

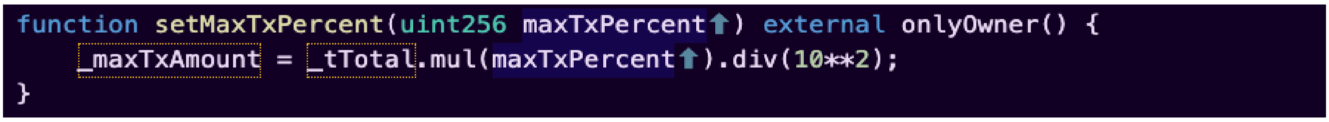
Owner privileges (In the period when the owner is not renounced)

- Admins can change fees.



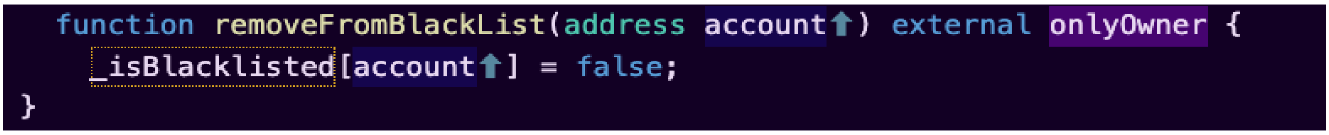
```
function setDefaultMarketingFee(uint256 marketingFee) external onlyAdmins() {
    require(marketingFee >= 0 && marketingFee <= 15, 'marketingFee should be in 0 - 15');

    defaultMarketingFee = marketingFee;
}

ftrace | funcSig
function setReflectionFee(uint256 reflectionFeePercentage) external onlyAdmins() {
    require(reflectionFeePercentage >= 0 && reflectionFeePercentage <= 10, 'reflectionFeePercentage should be in 0 - 10');
    defaultTaxFee = reflectionFeePercentage;
}

ftrace | funcSig
function setMarketingFee4Sellers(uint256 marketingFee4Sellers) external onlyAdmins() {
    require(marketingFee4Sellers >= 0 && marketingFee4Sellers <= 15, 'marketingFee4Sellers should be in 0 - 15');
    _marketingFee4Sellers = marketingFee4Sellers;
}
```

- Owner can change the maximum transaction amount.



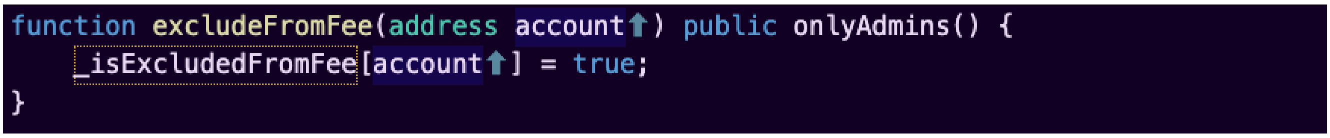
```
function setMaxTxPercent(uint256 maxTxPercent) external onlyOwner() {
    _maxTxAmount = _tTotal.mul(maxTxPercent).div(10**2);
}
```

- Owner can remove addresses from blacklist.



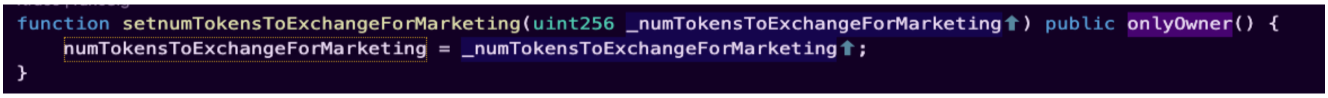
```
  function removeFromBlackList(address account) external onlyOwner {
    _isBlacklisted[account] = false;
}
```

- Admins can exclude from the fee.



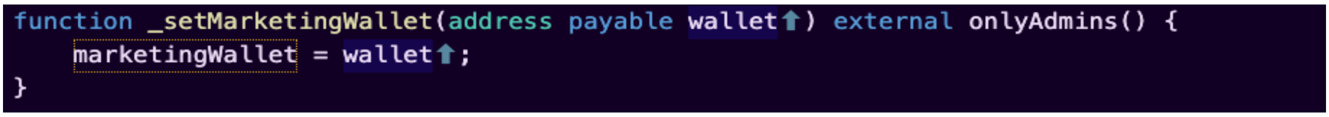
```
function excludeFromFee(address account) public onlyAdmins() {
    _isExcludedFromFee[account] = true;
}
```

- Owner can change number of tokens to exchange for marketing.



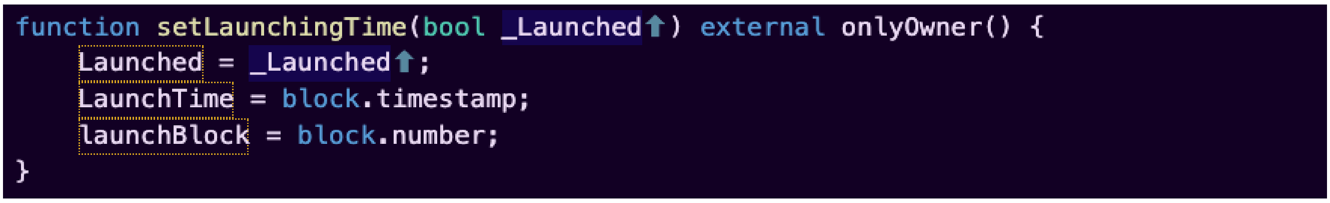
```
function setnumTokensToExchangeForMarketing(uint256 _numTokensToExchangeForMarketing) public onlyOwner() {
    numTokensToExchangeForMarketing = _numTokensToExchangeForMarketing;
}
```

- Owner can change marketing wallet.



```
function _setMarketingWallet(address payable wallet) external onlyAdmins() {
    marketingWallet = wallet;
}
```

- Owner can change launching time.



```
function setLaunchingTime(bool _Launched) external onlyOwner() {
    Launched = _Launched;
    LaunchTime = block.timestamp;
    launchBlock = block.number;
}
```

Conclusion

Smart contracts contain low severity issues! Liquidity pair contract's security is not checked due to out of scope. Fee increases on sell.

Liquidity locking details provided by the team:
https://app.unicrypt.network/amm/uni-v2/pair/0xaa788452bf40e540f2e76eea20d2fe9702d8f27f

Ownership renounced:
https://etherscan.io/tx/0x5a7b9b7feb03bb1800ed85f7d2c768b51fa7471ca7aabbd2533c1b3493b60da6

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.