TechRate

Blockchain solutions and consulting

Smart Contract Security Audit

Audit details:

Audited project:	Shilling Token
Deployer address:	0x45670e7a43df1380322acf1f6db857a0052e196b
Client contacts:	Shilling Token team
Blockchain:	Binance Smart Chain
Project website:	Not provided

May, 2021

TechRate

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 Shilling Token to perform an audit of smart contracts:

- *https://bscscan.com/address/0x643b6ef6306417a0b3fa2813eb5baf30f5dd8736#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.

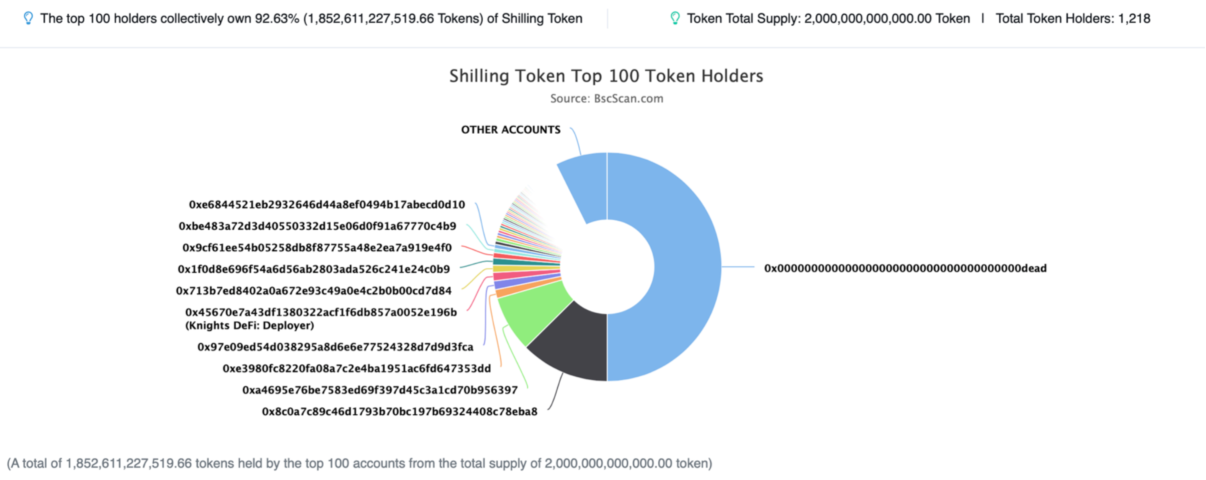
The information in this report should be used to understand the risk exposure of the smart contract, and as a guide to improve the security posture of the smart contract by remediating the issues that were identified.

Contracts details

Token contract details for 09.05.2021.

Contract name:	Shilling Token
Contract address:	0x643b6ef6306417a0b3fa2813eb5baf30f5dd8736
Total supply:	2000000000000000000000000000000
Token ticker:	SHILLING
Decimals:	18
Token holders:	1218
Transactions count:	8241
Top 100 holders dominance:	92.63 %
Liquidity fee:	8
Tax fee:	2
Total fees:	41797528535860328759802709076
Pancake pair:	0xa4695e76be7583ed69f397d45c3a1cd70b956397
Contract deployer address:	0x45670e7a43df1380322acf1f6db857a0052e196b
Contract's current owner address:	0x0000000000000000000000000000000000000000

Shilling Token token distribution



The top 100 holders collectively own 92.63% (1,852,611,227,519.66 Tokens) of Shilling Token

Token Total Supply: 2,000,000,000,000.00 Token | Total Token Holders: 1,218

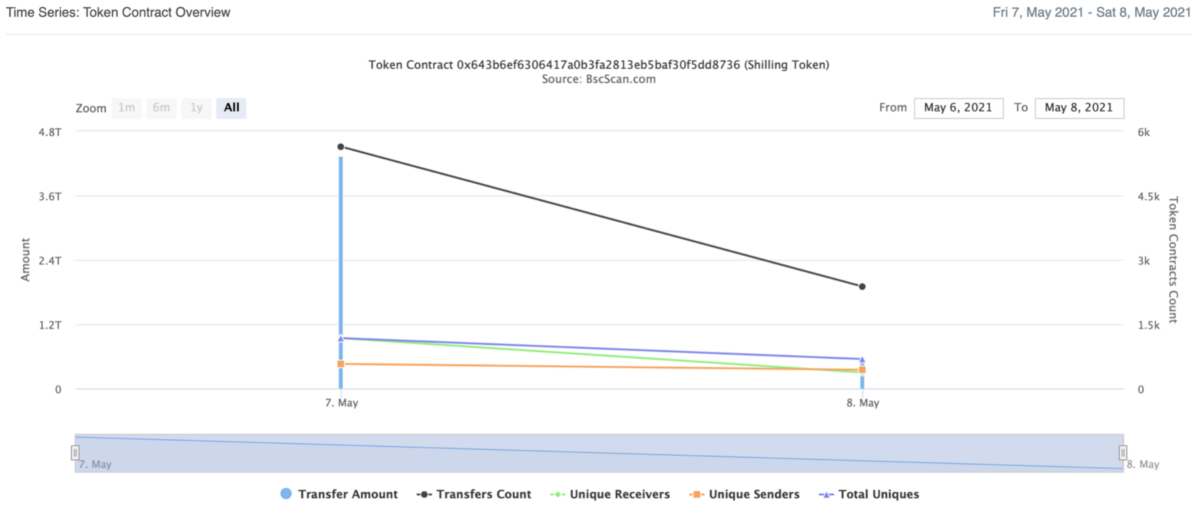
Shilling Token Top 100 Token Holders

Source: BscScan.com

OTHER ACCOUNTS
0xe6844521eb2932646d44a8ef0494b17abecd0d10
0xbe483a72d3d40550332d15e06d0f91a67770c4b9
0x9cf61ee54b05258db8f87755a48e2ea7a919e4f0
0x1f0d8e696f54a6d56ab2803ada526c241e24c0b9
0x713b7ed8402a0a672e93c49a0e4c2b0b00cd7d84
0x45670e7a43df1380322acf1f6db857a0052e196b (Knights DeFi: Deployer)
0x97e09ed54d038295a8d6e6e77524328d7d9d3fca
0xe3980fc8220fa08a7c2e4ba1951ac6fd647353dd
0xa4695e76be7583ed69f397d45c3a1cd70b956397
0x8c0a7c89c46d1793b70bc197b69324408c78eba8
0x000000000000000000000000000000000000dead

(A total of 1,852,611,227,519.66 tokens held by the top 100 accounts from the total supply of 2,000,000,000,000.00 token)

Shilling Token contract interaction details



Time Series: Token Contract Overview

Fri 7, May 2021 - Sat 8, May 2021

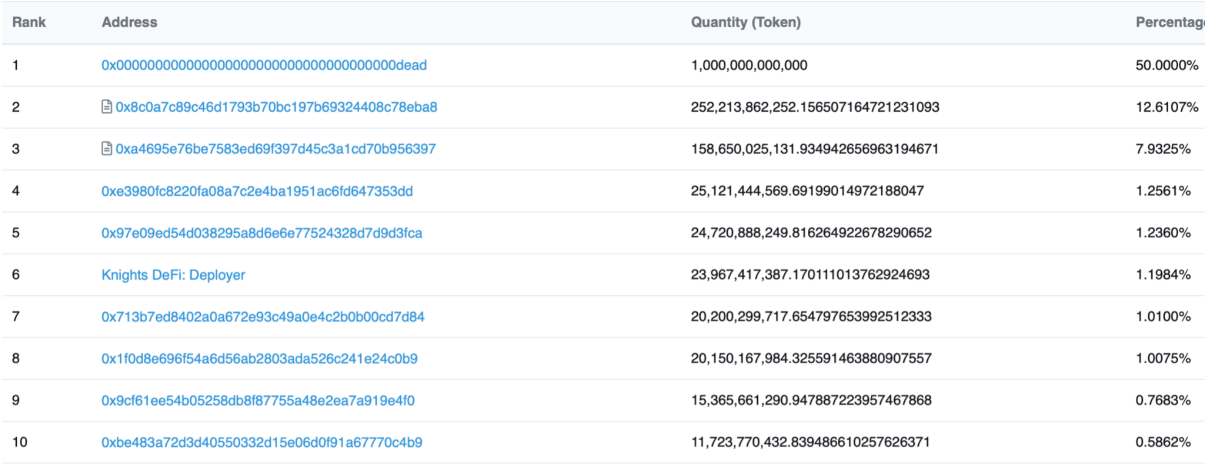
Token Contract 0x643b6ef6306417a0b3fa2813eb5baf30f5dd8736 (Shilling Token)

Source: BscScan.com

Zoom 1m 6m 1y All

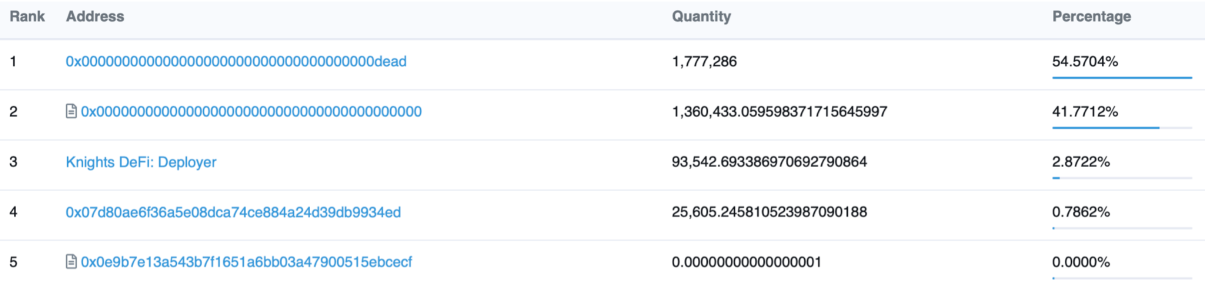
From May 6, 2021 To May 8, 2021

Shilling Token top 10 token holders



Rank	Address	Quantity (Token)	Percentag
1	0x000000000000000000000000000000000000dead	1,000,000,000,000	50.0000%
2	🗎 0x8c0a7c89c46d1793b70bc197b69324408c78eba8	252,213,862,252.156507164721231093	12.6107%
3	🗎 0xa4695e76be7583ed69f397d45c3a1cd70b956397	158,650,025,131.934942656963194671	7.9325%
4	0xe3980fc8220fa08a7c2e4ba1951ac6fd647353dd	25,121,444,569.69199014972188047	1.2561%
5	0x97e09ed54d038295a8d6e6e77524328d7d9d3fca	24,720,888,249.816264922678290652	1.2360%
6	Knights DeFi: Deployer	23,967,417,387.170111013762924693	1.1984%
7	0x713b7ed8402a0a672e93c49a0e4c2b0b00cd7d84	20,200,299,717.654797653992512333	1.0100%
8	0x1f0d8e696f54a6d56ab2803ada526c241e24c0b9	20,150,167,984.325591463880907557	1.0075%
9	0x9cf61ee54b05258db8f87755a48e2ea7a919e4f0	15,365,661,290.947887223957467868	0.7683%
10	0xbe483a72d3d40550332d15e06d0f91a67770c4b9	11,723,770,432.839486610257626371	0.5862%

Shilling Token LP top 10 token holders



Rank	Address	Quantity	Percentage
1	0x000000000000000000000000000000000000dead	1,777,286	54.5704%
2	🗎 0x0000000000000000000000000000000000000000	1,360,433.059598371715645997	41.7712%
3	Knights DeFi: Deployer	93,542.693386970692790864	2.8722%
4	0x07d80ae6f36a5e08dca74ce884a24d39db9934ed	25,605.245810523987090188	0.7862%
5	🗎 0x0e9b7e13a543b7f1651a6bb03a47900515ebcecf	0.00000000000000001	0.0000%

Contract functions details

```
+ [Int] IBEP20
  - [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

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

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

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

+ [Int] IPancakeFactory
```

```
  - [Ext] feeTo
  - [Ext] feeToSetter
  - [Ext] getPair
  - [Ext] allPairs
  - [Ext] allPairsLength
  - [Ext] createPair #
  - [Ext] setFeeTo #
  - [Ext] setFeeToSetter #

+ [Int] IPancakePair
  - [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] IPancakeRouter01
  - [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] IPancakeRouter02 (IPancakeRouter01)
  - [Ext] removeLiquidityETHSupportingFeeOnTransferTokens #
  - [Ext] removeLiquidityETHWithPermitSupportingFeeOnTransferTokens #
  - [Ext] swapExactTokensForTokensSupportingFeeOnTransferTokens #
  - [Ext] swapExactETHForTokensSupportingFeeOnTransferTokens ($)
  - [Ext] swapExactTokensForETHSupportingFeeOnTransferTokens #

+ ReentrancyGuard
  - [Pub] <Constructor> #

+ Shilling (Context, IBEP20, Ownable, ReentrancyGuard)
  - [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] setTransferDelayTime #
    - modifiers: onlyOwner
  - [Pub] deliver #
  - [Pub] reflectionFromToken
  - [Pub] tokenFromReflection
  - [Pub] excludeFromReward #
    - modifiers: onlyOwner
  - [Ext] includeInReward #
    - modifiers: onlyOwner
  - [Prv] _transferBothExcluded #

- [Pub] excludeFromFee #
  - modifiers: onlyOwner
- [Pub] excludeFromClaimTopUp #
  - modifiers: onlyOwner
- [Pub] includeInClaimTopUp #
  - modifiers: onlyOwner
- [Pub] includeInFee #
  - modifiers: onlyOwner
- [Ext] setTaxFeePercent #
  - modifiers: onlyOwner
- [Ext] setLiquidityFeePercent #
  - modifiers: onlyOwner
- [Pub] setSwapAndLiquifyEnabled #
  - modifiers: onlyOwner
- [Ext] <Fallback> ($)
- [Prv] _reflectFee #
- [Prv] _getValues
- [Prv] _getTValues
- [Prv] _getRValues
- [Prv] _getRate
- [Prv] _getCurrentSupply
- [Prv] _takeLiquidity #
- [Prv] calculateTaxFee
- [Prv] calculateLiquidityFee
- [Prv] removeAllFee #
- [Prv] restoreAllFee #
- [Pub] isExcludedFromFee
- [Prv] _approve #
- [Prv] _transfer #
- [Prv] _tokenTransfer #
- [Prv] _transferStandard #
- [Prv] _transferToExcluded #
- [Prv] _transferFromExcluded #
- [Pub] setMaxTxPercent #
  - modifiers: onlyOwner
- [Pub] setExcludeFromMaxTx #
  - modifiers: onlyOwner
- [Pub] calculateBNBReward
- [Pub] getRewardCycleBlock
- [Pub] claimBNBReward #
  - modifiers: isHuman,nonReentrant
- [Prv] topUpClaimCycleAfterTransfer #
- [Prv] ensureMaxTxAmount
- [Pub] disruptiveTransfer ($)
- [Prv] swapAndLiquify #
- [Ext] activateContract #
  - modifiers: onlyOwner

- [Ext] removeLaunchLimits #
- [Pub] calculateTopUpClaim
- [Prv] swapTokensForEth #
- [Prv] swapETHForTokens #
- [Prv] addLiquidity #

($) = 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

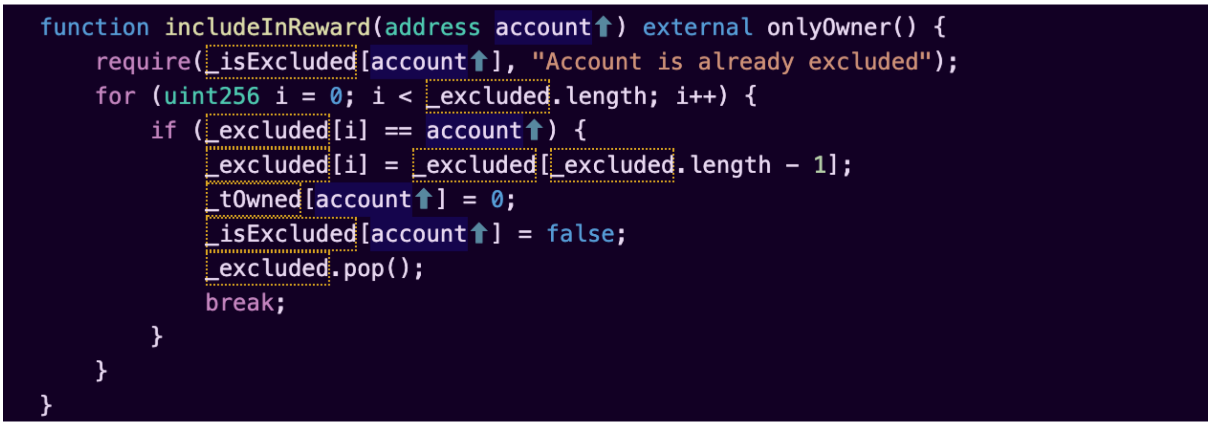
No medium severity issues found.

Low Severity Issues

1. Out of gas

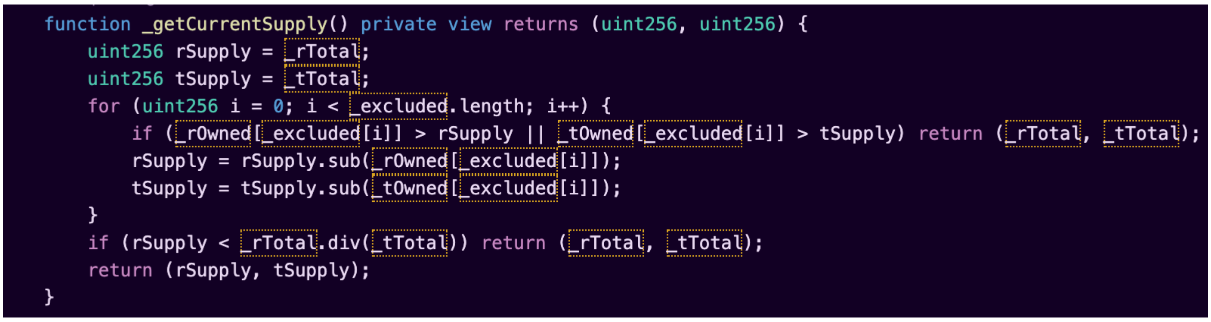
Issue:

- 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);
}
```

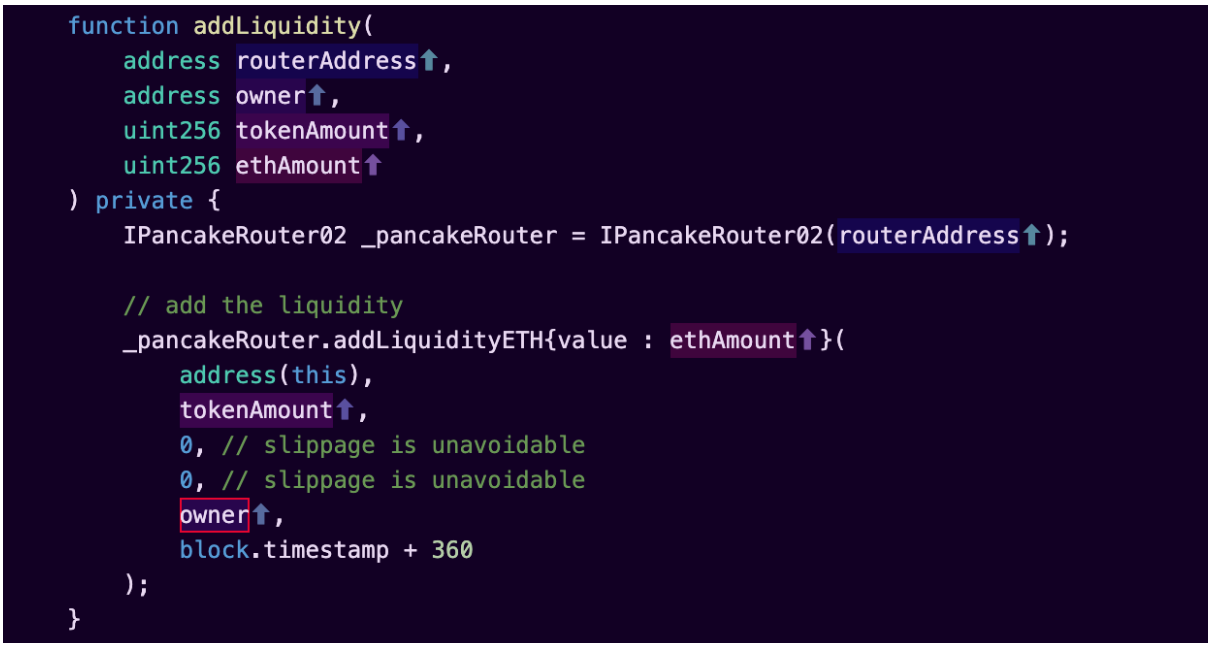
Recommendation:

Use EnumerableSet instead of array or do not use long arrays.

Conclusion

Smart contracts contain low severity issues! Liquidity pair contract is not checked due to out of scope.

Liquidity is added to the owner address, so owner should lock the liquidity manually.



```
    function addLiquidity(
        address routerAddress,
        address owner,
        uint256 tokenAmount,
        uint256 ethAmount
    ) private {
        IPancakeRouter02 _pancakeRouter = IPancakeRouter02(routerAddress);

        // add the liquidity
        _pancakeRouter.addLiquidityETH{value : ethAmount}(
            address(this),
            tokenAmount,
            0, // slippage is unavoidable
            0, // slippage is unavoidable
            owner,
            block.timestamp + 360
        );
    }
```

As you can see in the LP pair contract holders chart, about 96 percent of the liquidity is on the dead and zero addresses.

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.