Techrate1 Techrate

TechRate

AUDIT COMPANY

Smart Contract Security Audit

TechRate

August, 2021

Audit Details

Audited project

MicroSHIBA

Deployer address

0x3d09950F46A355B0b8E2D7372f6251C61ed7B9Fb

Client contacts:

MicroSHIBA team

Blockchain

Binance Smart Chain

Project website:

https://www.microshiba.net

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 MicroSHIBA to perform an audit of smart contracts:

https://bscscan.com/address/0xb78e41ae7fc8121978168557bb096e1a858be2ea#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

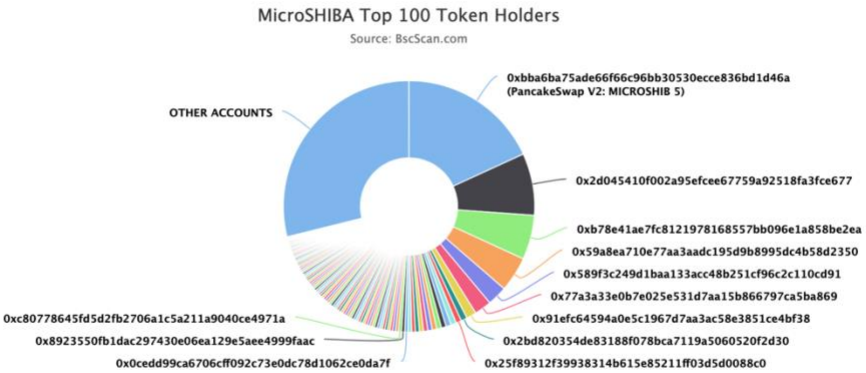
Token contract details for 07.08.2021

Contract name	MicroSHIBA
Contract address	0xb78e41AE7fc8121978168557bB096e1A858be2Ea
Total supply	1,000,000,000,000,000
Token ticker	MICROSHIB
Decimals	9
Token holders	7,953
Transactions count	20,422
Top 100 holders dominance	71.13%
Liquidity fee	14
Tax fee	1
Total fees	11677064354558461557254
Uniswap V2 pair	0xbba6ba75ade66f66c96bb30530ecce836bd1d46a
Contract deployer address	0x3d09950F46A355B0b8E2D7372f6251C61ed7B9Fb
Contract's current owner address	0x1c3509508dedfcc9d7895115ae97f96c97954690

MicroSHIBA Token Distribution

The top 100 holders collectively own 71.13% (711,304,214,629,969.00 Tokens) of MicroSHIBA

Token Total Supply: 1,000,000,000,000,000.00 Token | Total Token Holders: 7,953

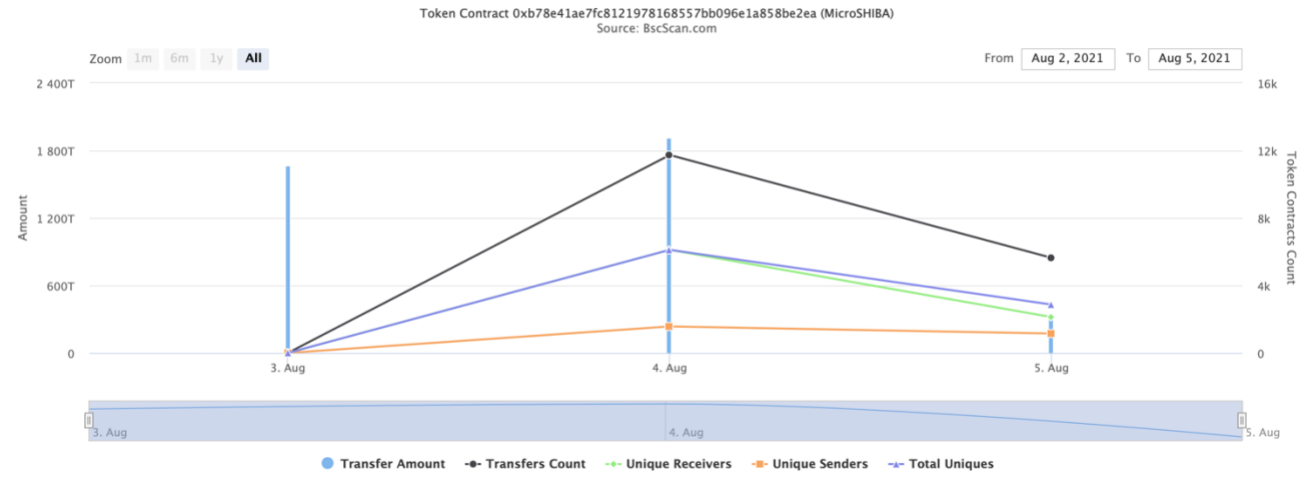


(A total of 711,304,214,629,969.00 tokens held by the top 100 accounts from the total supply of 1,000,000,000,000,000.00 token)

MicroSHIBA Contract Interaction Details

Time Series: Token Contract Overview

Tue 3, Aug 2021 - Thu 5, Aug 2021



MicroSHIBA Top 10 Token Holders

Rank	Address	Quantity (Token)	Percentage
1	PancakeSwap V2: MICROSHIB 5	181,769,515,605,132.008642187	18.1770%
2	0x2d045410f002a95efcee67759a92518fa3fce677	79,603,200,000,000	7.9603%
3	0xb78e41ae7fc8121978168557bb096e1a858be2ea	57,081,839,568,711.217968443	5.7082%
4	0x59a8ea710e77aa3aadc195d9b8995dc4b58d2350	44,970,121,059,125.638832586	4.4970%
5	0x589f3c249d1baa133acc48b251cf96c2c110cd91	25,503,291,041,562.152308849	2.5503%
6	0x77a3a33e0b7e025e531d7aa15b866797ca5ba869	21,905,879,098,750.68325212	2.1906%
7	0x91efc64594a0e5c1967d7aa3ac58e3851ce4bf38	12,888,911,081,581.434220406	1.2889%
8	0x2bd820354de83188f078bca7119a5060520f2d30	7,994,100,467,238.579467647	0.7994%
9	0x25f89312f39938314b615e85211ff03d5d0088c0	7,026,357,286,152.590628154	0.7026%
10	0x03288648e603bd6e350da1d3bd5532ac0a46cb9c	7,009,430,491,315.160725683	0.7009%

Contract functions details

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

+ [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

+ Ownable (Context)
  - [Pub] <Constructor> #
  - [Pub] owner
  - [Pub] renounceOwnership #
    - modifiers: onlyOwner
  - [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] IUniswapV2Router
  - [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 ($)
  - [Ext] quote
  - [Ext] getAmountOut
  - [Ext] getAmountIn
  - [Ext] getAmountsOut
  - [Ext] getAmountsIn
  - [Ext] removeLiquidityETHSupportingFeeOnTransferTokens #
  - [Ext] removeLiquidityETHWithPermitSupportingFeeOnTransferTokens #
  - [Ext] swapExactTokensForTokensSupportingFeeOnTransferTokens #
  - [Ext] swapExactETHForTokensSupportingFeeOnTransferTokens ($)
  - [Ext] swapExactTokensForETHSupportingFeeOnTransferTokens #

+ [Int] IDividendDistributor
  - [Ext] setDistributionCriteria #
  - [Ext] setShare #
  - [Ext] deposit ($)
  - [Ext] process #

+ DividendDistributor (IDividendDistributor)

- [Pub] <Constructor> #
- [Ext] setDistributionCriteria #
  - modifiers: onlyToken
- [Ext] setShare #
  - modifiers: onlyToken
- [Ext] deposit ($)
  - modifiers: onlyToken
- [Ext] process #
  - modifiers: onlyToken
- [Int] shouldDistribute
- [Int] distributeDividend #
- [Ext] claimDividend #
- [Pub] getUnpaidEarnings
- [Int] getCumulativeDividends
- [Int] addShareholder #
- [Int] removeShareholder #
- [Pub] readShares
- [Prv] transferToAddressETH #

+ Microshiba (Context, IBEP20, 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 #
  - [Int] 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
- [Ext] manualSwapandLiquify #
  - modifiers: onlyOwner
- [Ext] manualDistributorProcess #
  - modifiers: onlyOwner
- [Ext] manualSetShare #
  - modifiers: onlyOwner
- [Pub] checkUptrend
- [Pub] setLastBlockNumber #
  - modifiers: onlyOwner
- [Pub] setLastPairBalance #
  - modifiers: onlyOwner
- [Pub] setLaunchLiqPair #
  - modifiers: onlyOwner
- [Pub] isSniper
- [Prv] removeAllFee #
- [Prv] restoreAllFee #
- [Pub] isExcludedFromFee
- [Pub] excludeFromFee #
  - modifiers: onlyOwner
- [Pub] includeInFee #
  - modifiers: onlyOwner
- [Pub] GetSwapMinutes
- [Ext] SetSwapMinutes #
  - modifiers: onlyOwner
- [Ext] setTaxFeePercent #
  - modifiers: onlyOwner
- [Ext] setLiquidityFeePercent #
  - modifiers: onlyOwner
- [Prv] _checkLiquidityAdd #
- [Ext] removeSniper #
  - modifiers: onlyOwner
- [Ext] setMaxTxAmount #
  - modifiers: onlyOwner
- [Ext] setMinimumTokensBeforeSwap #
  - modifiers: onlyOwner
- [Ext] setBuybackAddress #
  - modifiers: onlyOwner
- [Ext] setBuybackDivisor #
  - modifiers: onlyOwner
- [Ext] setBlockChunk #
  - modifiers: onlyOwner
- [Pub] setcheckUptrendActive #
  - modifiers: onlyOwner
- [Pub] setMultiBuybackWallet #
  - modifiers: onlyOwner
- [Pub] setSwapAndLiquifyEnabled #
  - modifiers: onlyOwner
- [Pub] setDividendContractActive #

  - modifiers: onlyOwner
- [Prv] transferToAddressETH #
- [Int] launched
- [Pub] setCurrentLiqPair #
  - modifiers: onlyOwner
- [Pub] launch #
  - modifiers: onlyOwner
- [Ext] afterPreSale #
  - modifiers: onlyOwner
- [Ext] setDistributionCriteria #
  - modifiers: onlyOwner
- [Ext] setDistributorSettings #
  - modifiers: onlyOwner
- [Ext] manualDistributorDeposit ($)
  - modifiers: onlyOwner
- [Pub] manualGetUnpaidShares
- [Ext] setIsDividendExempt #
  - modifiers: onlyOwner
- [Ext] setDxSaleAddress #
  - modifiers: onlyOwner
- [Ext] readShares
- [Pub] changeRouterVersion #
  - modifiers: onlyOwner
- [Ext] <Fallback> ($)
- [Pub] transferForeignToken #
  - modifiers: onlyOwner
- [Ext] Sweep #
  - 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 not 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.

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

- Owner can manually call swap and liquify.
- Owner can manually process distribution.
- Owner can manually change share for shareholder.
- Owner can change last block number.
- Owner can update last pair balance.
- Owner can change pair address.
- Owner can exclude from the fee.
- Owner can change _intervalSecondsForSwap value.
- Owner can change tax and liquidity fees.
- Owner can remove sniper.
- Owner can change maximum transaction amount.
- Owner can change minimum tokens before swap.
- Owner can change buyback addresses.
- Owner can change blockchunk.
- Owner can disable/enable checkUptrendActive.
- Owner can disable/enable multiBuybackWallet.
- Owner can disable/enable swapAndLiquify.
- Owner can disable/enable dividendContract.
- Owner can set currentLiqPair.
- Owner can enable before and after presale modes.
- Owner can change distribution criteria.
- Owner can change distribution GAS.
- Owner can manually deposit distribution.
- Owner can include in and exclude from dividends.
- Owner can add addresses in multiple exclusions.
- Owner can change router address.
- Owner can withdraw any BEP20 tokens and BNBs from the contract.
- Owner can lock and unlock. By the way, using these functions the owner could retake privileges even after the ownership was renounced.

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 provided by the team:
https://dxsale.app/app/v2_9/dxlockview?id=2136&add=0&type=lpdefi&chain=BSC

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.