TechRate

AUDIT COMPANY

Smart Contract Security Audit

Audit Details

Audited project

miniShibaToken

Deployer address

0x07b7768F6863943a16372876A8000232b6391AdC

Client contacts:

miniShibaToken team

Blockchain

Binance Smart Chain

Project website:

https://minishib.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 miniShibaToken to perform an audit of smart contracts:
https://bscscan.com/address/0x35bd36597ff784fcc30d08ebe4bd917374c40d4b#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 18.07.2021

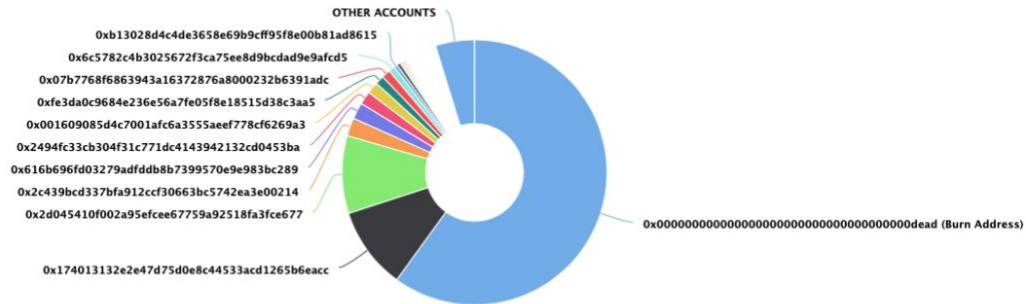|	
Contract name	miniShibaToken
Contract address	0x35Bd36597fF784FCc30D08EBE4Bd917374C40d4B
Total supply	100,000,000,000,000,000
Token ticker	miniSHIB
Decimals	9
Token holders	7,949
Transactions count	14,018
Top 100 holders dominance	95.24%
Liquidity fee	5
Tax fee	5
Total fees	30806962482666630299357155
Uniswap V2 pair	0x174013132e2e47d75d0e8c44533acd1265b6eacc
Contract deployer address	0x07b7768F6863943a16372876A8000232b6391AdC
Contract's current owner address	0x07b7768f6863943a16372876a8000232b6391adc

miniShibaToken Token Distribution

miniShibaToken Top 100 Token Holders
Source: BscScan.com

OTHER ACCOUNTS

0xb13028d4c4de3658e69b9cff95f8e00b81ad8615
0x6c5782c4b3025672f3ca75ee8d9bcdad9e9afcd5
0x07b7768f6863943a16372876a8000232b6391adc
0xfe3da0c9684e236e56a7fe05f8e18515d38c3aa5
0x001609085d4c7001afc6a3555aeef778cf6269a3
0x2494fc33cb304f31c771dc4143942132cd0453ba
0x616b696fd03279adfddb8b7399570e9e983bc289
0x2c439bcd337bfa912ccf30663bc5742ea3e00214
0x2d045410f002a95efcee67759a92518fa3fce677
0x174013132e2e47d75d0e8c44533acd1265b6eacc

0x0000000000000000000000000000000000000dead (Burn Address)

(A total of 95,244,848,090,800,200.00 tokens held by the top 100 accounts from the total supply of 100,000,000,000,000,000.00 token)
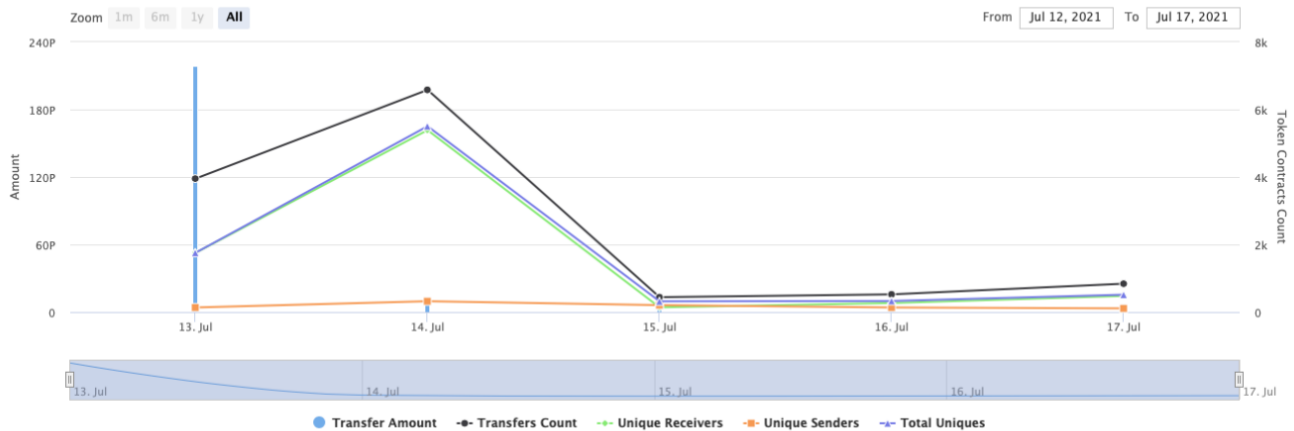
miniShibaToken Contract Interaction Details

Time Series: Token Contract Overview   Tue 13, Jul 2021 - Sat 17, Jul 2021

Token Contract 0x35bd36597ff784fcc30d08ebe4bd917374c40d4b (miniShibaToken)
Source: BscScan.com


Zoom 1m 6m 1y All   From Jul 12, 2021 To Jul 17, 2021

● Transfer Amount   -●- Transfers Count   -●- Unique Receivers   -■- Unique Senders   -▲- Total Uniques

miniShibaToken Top 10 Token Holders

Rank	Address	Quantity (Token)	Percentage
1	Burn Address	60,003,164,535,668,900.902076948	60.0032%
2	0x174013132e2e47d75d0e8c44533acd1265b6eacc	9,997,908,583,669,440.614035402	9.9979%
3	0x2d045410f002a95efcee67759a92518fa3fce677	9,474,624,459,363,670.319273023	9.4746%
4	0x2c439bcd337bfa912ccf30663bc5742ea3e00214	2,221,518,811,301,820.297087571	2.2215%
5	0x616b696fd03279adfddb8b7399570e9e983bc289	1,997,923,191,928,120.762942736	1.9979%
6	0x2494fc33cb304f31c771dc4143942132cd0453ba	1,611,950,880,942,470.363824738	1.6120%
7	0x001609085d4c7001afc6a3555aeef778cf6269a3	1,426,430,538,872,480.521365482	1.4264%
8	0xfe3da0c9684e236e56a7fe05f8e18515d38c3aa5	1,091,102,081,360,590.018976063	1.0911%
9	0x07b7768f6863943a16372876a8000232b6391adc	1,061,160,096,796,940.07983557	1.0612%
10	0x6c5782c4b3025672f3ca75ee8d9bcdad9e9afcd5	749,007,091,175,630.763849969	0.7490%

miniShibaToken LP Token Holders

Rank	Address	Quantity	Percentage
1	0x8655e5c4d701186d16765d1cdcef6d5287e4679a	94,904.361815708804352667	89.7861%
2	0x07b7768f6863943a16372876a8000232b6391adc	10,295.778556741692158589	9.7405%
3	0x07d80ae6f36a5e08dca74ce884a24d39db9934ed	482.742441434726015176	0.4567%
4	0x53818ff47a813bdd95de8852ab2e667c4ace3d8d	12.775921439667466917	0.0121%
5	0xcb8da56c486f40166b609b7c2363cf82ddb106e1	4.436457189223126456	0.0042%
6	0x132017d29a742f6acb5ffe4ce7f73fe3c216120a	0.383315997505695926	0.0004%
7	0xf06bc89e8456ad4172dc299b4bbce24ba47a76eb	0.012937446296095496	0.0000%
8	0x0000000000000000000000000000000000000000	0.000000000000001	0.0000%
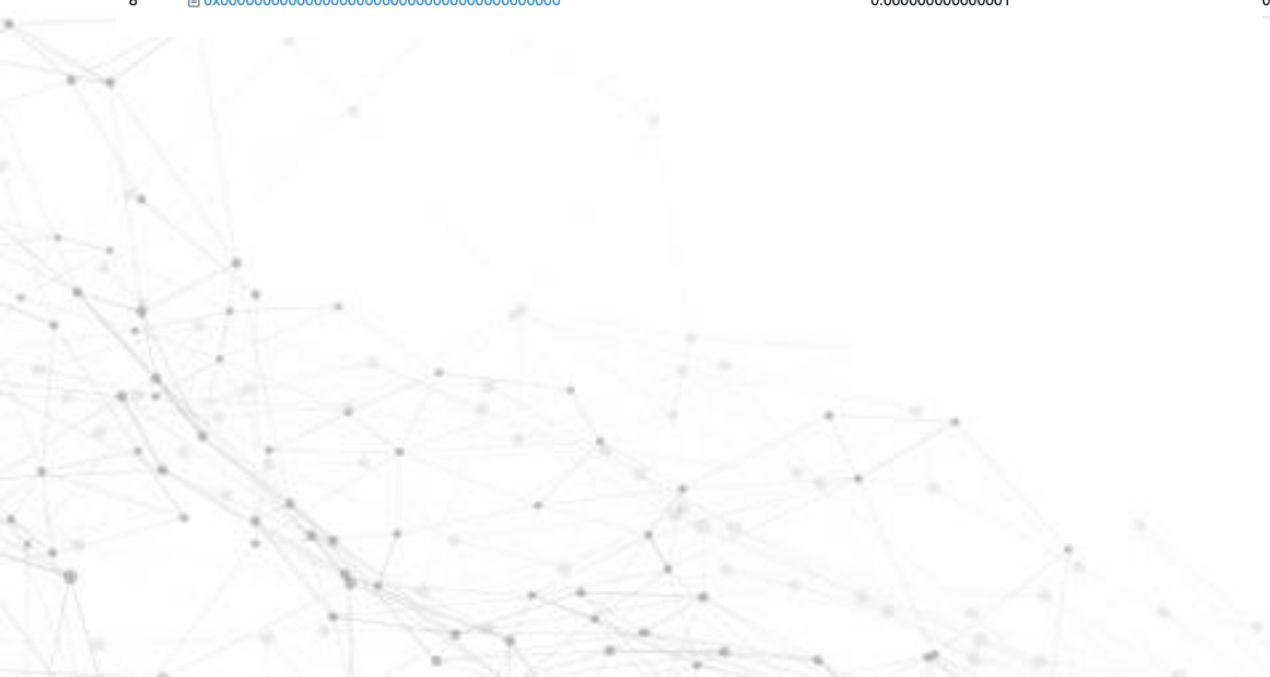
Contract functions details

+ [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

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

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

+ Ownable **(Context)**
 - **[Pub]** owner
 - **[Pub]** renounceOwnership **#**
   - modifiers: onlyOwner
 - **[Pub]** transferOwnership **#**
   - modifiers: onlyOwner
 - **[Pub]** geUnlockTime
 - **[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]** 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 **#**

+ miniShibaToken **(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]** deliver **#**
- **[Pub]** reflectionFromToken
- **[Pub]** tokenFromReflection
- **[Pub]** excludeFromReward **#**
  - modifiers: onlyOwner
- **[Ext]** includeInReward **#**
  - modifiers: onlyOwner
- **[Prv]** _transferBothExcluded **#**
- **[Pub]** excludeFromFee **#**
  - modifiers: onlyOwner
- **[Pub]** includeInFee **#**
  - modifiers: onlyOwner
- **[Ext]** setTaxFeePercent **#**
  - modifiers: onlyOwner
- **[Ext]** setLiquidityFeePercent **#**
  - modifiers: onlyOwner
- **[Pub]** setNumTokensSellToAddToLiquidity **#**
  - modifiers: onlyOwner
- **[Pub]** setMaxTxPercent **#**
  - modifiers: onlyOwner
- **[Pub]** setSwapAndLiquifyEnabled **#**
  - modifiers: onlyOwner
- **[Ext]** <Fallback> **($)**
- **[Prv]** _reflectFee **#**
- **[Prv]** _getValues
- **[Prv]** _getTValues
- **[Prv]** _getRValues
- **[Prv]** _getRate
- **[Prv]** _getCurrentSupply
- **[Prv]** _takeLiquidity **#**
- **[Pub]** claimTokens **#**
  - modifiers: onlyOwner
- **[Prv]** calculateTaxFee
- **[Prv]** calculateLiquidityFee
- **[Prv]** removeAllFee **#**
- **[Prv]** restoreAllFee **#**
- **[Pub]** isExcludedFromFee

- **[Prv]** _approve **#**
- **[Prv]** _transfer **#**
- **[Prv]** swapAndLiquify **#**
  - modifiers: lockTheSwap
- **[Prv]** swapTokensForEth **#**
- **[Prv]** addLiquidity **#**
- **[Prv]** _tokenTransfer **#**
- **[Prv]** _transferStandard **#**
- **[Prv]** _transferToExcluded **#**
- **[Prv]** _transferFromExcluded **#**

($) = 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.

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

- **Owner can change the tax and liquidity fee.**

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

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

- **Owner can change the maximum transaction amount.**

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

- **Owner can exclude from the fee.**

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

- **Owner can claim all tokens from contract balance.**

```
ftrace | funcSig
function claimTokens() public onlyOwner {
        payable(_owner).transfer(address(this).balance);
}
```

- **Owner can change numTokensSellToAddToLiquidity.**

```
ftrace | funcSig
function setNumTokensSellToAddToLiquidity(uint256 swapNumber↑) public onlyOwner {
    numTokensSellToAddToLiquidity = swapNumber↑ * 10 ** _decimals;
}
```

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

```solidity
//Locks the contract for owner for the amount of time provided
function lock(uint256 time) public virtual onlyOwner {
    _previousOwner = _owner;
    _owner = address(0);
    _lockTime = now + time;
    emit OwnershipTransferred(_owner, address(0));
}

//Unlocks the contract for owner when _lockTime is exceeds
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.

Liquidity locking details provided by the team:
https://dxsale.app/app/v2_9/dxlockview?id=0&add=0x07b7768F686 3943a16372876A8000232b6391AdC&type=lplock&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.