HackSafe

Smart Contract Security Audit Report

METAZOON TOKEN

December 2022

Security Status

★★★★☆

www.hacksafe.io

Audit Details

Audited project
METAZOON TOKEN

Deployer address
0xb1a93b51a3a333912fd8e40c10d44b1c5dd08945

Client contacts
METAZOON TOKEN Team

Blockchain
Binance smart chain

Website
Not provided

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) (HackSafe) owe no duty of care towards you or any other person, nor does HackSafe 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 HackSafe 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, HackSafe hereby excludes all liability and responsibility, and neither you nor any other person shall have any claim against HackSafe, 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.

Procedure

Step 1 - In-Depth Manual Review

Manual line-by-line code reviews to ensure the logic behind each function is sound and safe from various attack vectors. This is the most important and lengthy portion of the audit process (as automated tools often cannot find the nuances that lead to exploits such as flash loan attacks).

Step 2 - Automated Testing

Simulation of a variety of interactions with your Smart Contract on a test blockchain leveraging a combination of automated test tools and manual testing to determine if any security vulnerabilities exist.

Step 3 – Leadership Review

The engineers assigned to the audit will schedule meetings with our leadership team to review the contracts, any comments or findings, and ask questions to further apply adversarial thinking to discuss less common attack vectors.

Step 4 - Resolution of Issues

Consulting with the team to provide our recommendations to ensure the code's security and optimize its gas efficiency, if possible. We assist project team's in resolving any outstanding issues or implementing our recommendations.

Step 5 - Published Audit Report

Boiling down results and findings into an easy-to-read report tailored to the project. Our audit reports highlight resolved issues and any risks that exist to the project or its users, along with any remaining suggested remediation measures. Diagrams are included at the end of each report to help users understand the interactions which occur within the project.

Background

HackSafe was commissioned by METAZOON TOKEN to perform an audit of smart contracts:

- https://bscscan.com/token/0x60e9129dCF1D2e235eA0b572fdDc2bfbBbE9c0bc#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 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.

Contract Details

Token contract details for 30.12.2022

Token Type	: meme
Contract name	: MetaZoon
Contract address	: 0x60e9129dCF1D2e235eA0b572fdDc2bfbBbE9c0bc
Total supply	: 100,000,000,000,000,000
Token ticker	: MEZ
Decimals	: 9
Token Holders	: 10,710
Transactions count	: 27,219
Liquidity fee	: 5
Tax fee	: 5
Compiler version	: v0.6.12+commit.27d51765
Contract deployer address	: 0xb1a93b51a3a333912fd8e40c10d44b1c5dd08945
Owner address	: 0xb1a93b51a3a333912fd8e40c10d44b1c5dd08945

Audit Summary

According to the standard audit assessment, Customer`s solidity smart contracts are **“Secure”.** This token contract does contain owner control, which do not make it fully decentralized.

Insecure	Poor secured	Secure	Well-secured

You are here 

We used various tools like Slither, Mythril and Remix IDE. At the same time this finding is based on critical analysis of the manual audit. All issues found during automated analysis were manually reviewed and applicable vulnerabilities are presented in the issues checking status.

We found 0 critical, 0 high, 1 medium and 1 low.

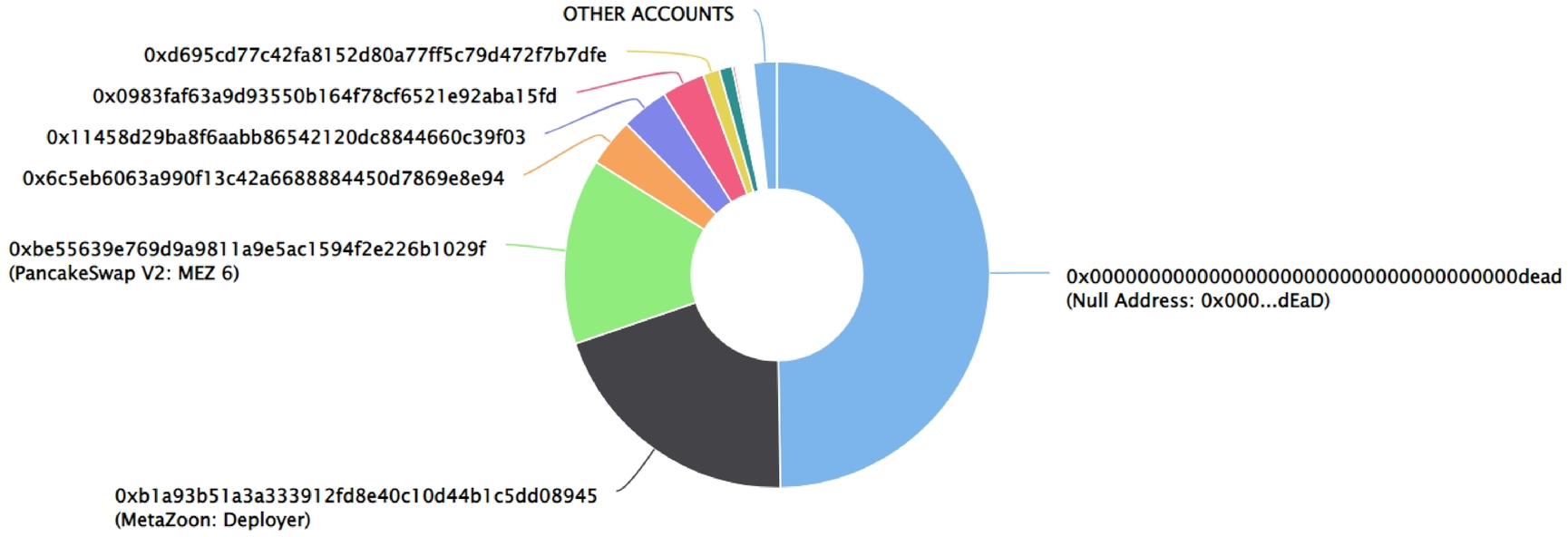
METAZOON TOKEN Distribution

The top 100 holders collectively own 98.23%
(98,227,102,017,187,400.00 Tokens) of METAZOON TOKEN

Token Total Supply: 100,000,000,000,000,000.00 Token | Total Token Holders: 10,710

METAZOON TOKEN Top 100 Token Holders

Source: BscScan.com



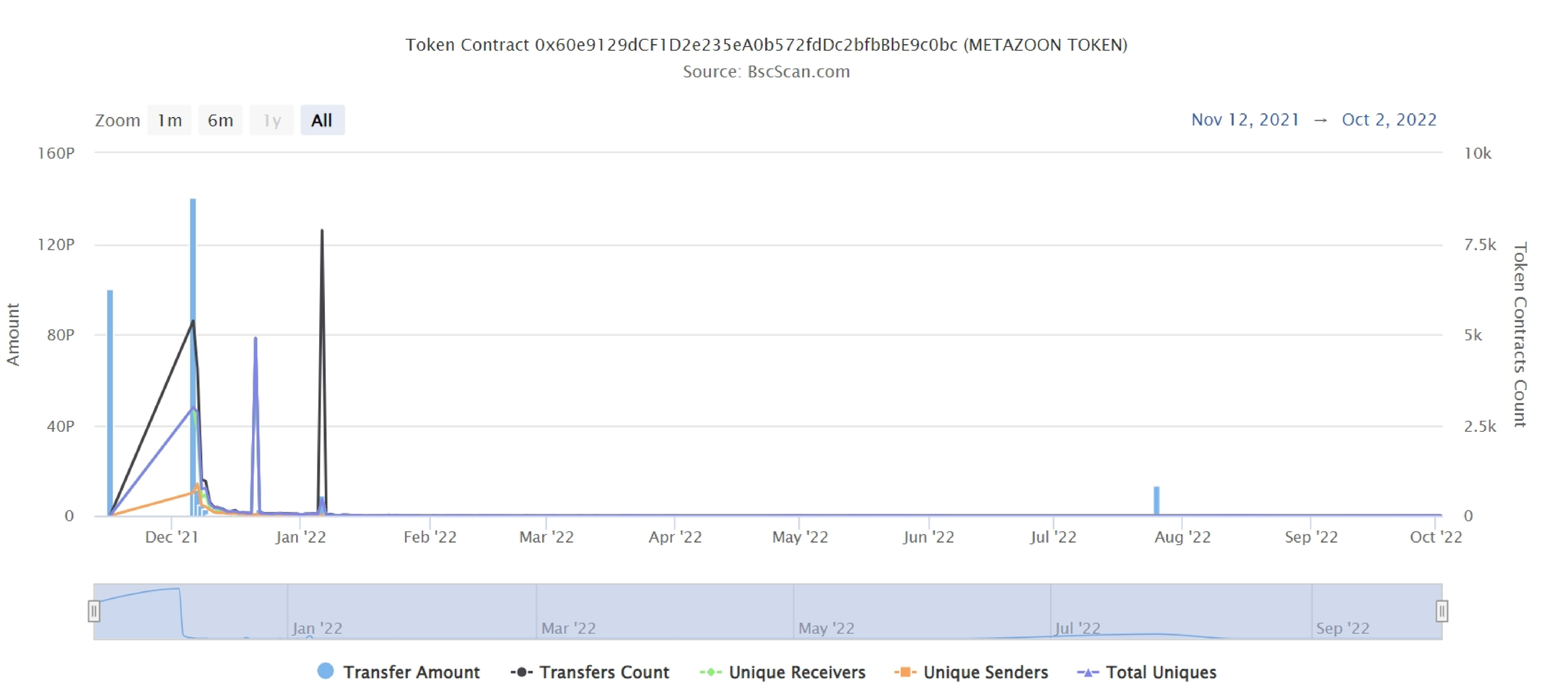
METAZOON TOKEN Top 20 Token Holders

(A total of 98,227,102,017,187,400.00 tokens held by the top 100 accounts from the total supply of 100,000,000,000,000,000.00 token)

Rank	Address	Quantity (Token)	Percentage
1	Null Address: 0x000...dEaD	49,748,476,802,620,100.272464867	49.7485%
2	MetaZoon: Deployer	20,016,288,117,383,400.920554164	20.0163%
3	PancakeSwap V2: MEZ 6	14,112,457,945,922,500.457088037	14.1125%
4	0x6c5eb6063a990f13c42a6688884450d7869e8e94	3,635,887,565,254,830.276894694	3.6359%
5	0x11458d29ba8f6aabb86542120dc8844660c39f03	3,592,970,091,271,940.308823304	3.5930%
6	0x0983faf63a9d93550b164f78cf6521e92aba15fd	3,267,552,527,996,610.158298978	3.2676%
7	0xd695cd77c42fa8152d80a77ff5c79d472f7b7dfe	1,266,196,255,074,550.280769765	1.2662%
8	Radio Caca: RACA Token	1,000,159,167,103,000.849631734	1.0002%
9	0x4c3a49be6bca3a36848772e2ae375ce99d8ffbe6	202,605,516,815,509.379757607	0.2026%
10	0x0e840e1ab25dc07a4c63ae864e51d61e8e43645e	91,253,125,738,243.198505954	0.0913%
11	0x2db20712a84a50c77e065cfd50c67419a6ecb970	56,295,971,181,159.676598309	0.0563%
12	0x398659ac553c112d5126097e83f77dd052b8caad	52,397,186,031,064.124359647	0.0524%
13	0x7566cf57cd23e672b2d468120de8cd3c14922735	51,498,167,846,573.394140094	0.0515%
14	0xae901c32275c1817f3589da0d75265d6c9abe7d9	43,229,794,201,237.964882846	0.0432%
15	0x48e157bd67f1ee215a70c076380fb6f13cc59158	33,750,313,594,785.923428335	0.0338%
16	0x6ea74d43f4219705294d2d4e94872efc1c46349b	33,351,140,133,014.824738756	0.0334%
17	0x0f456cd7cdc806496972ea89db250b76b98171c7	30,091,420,930,138.686895826	0.0301%
18	0x0a3f059f212dfa35c26d51baeeff564cf103d99c	27,946,362,170,015.960091257	0.0279%
19	0x28705c26e20856f4cb90caa5702ae6b796a3522d	26,974,396,340,330.680734956	0.0270%
20	0xed09e9ccd61983385128142afa7a9443df03dff0	26,877,821,336,940.421060673	0.0269%

METAZOON TOKEN Distribution

METAZOON TOKEN Contract Overview



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 #
    -[Pvt] _functionCallWithValue #

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

Contract functions details

+[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 #

Contract functions details

+[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 #

+MetaZoon (Context, IERC20, Ownable)
    -[Pub] <Constructor> #
    -[Pub] name
    -[Pub] symbol
    -[Pub] decimals
    -[Pub] totalSupply
    -[Pub] balanceOf
    -[Pub] transfer #
    -[Pub] allowance
    -[Pub] approve #
    -[Pub] transferFrom #

Contract functions details

-[Pub] increaseAllowance #
-[Pub] decreaseAllowance #
-[Pub] isExcludedFromReward
-[Pub] totalFees
-[Pub] deliver #
-[Pub] reflectionFromToken
-[Pub] tokenFromReflection
-[Pub] excludeFromReward #
  - modifiers: onlyOwner
-[Ext] includeInReward #
  - modifiers: onlyOwner
-[Pvt] _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] <receive> ($)
-[Pvt] _reflectFee #
-[Pvt] _getValues
-[Pvt] _getTValues
-[Pvt] _getRValues
-[Pvt] _getRate
-[Pvt] _getCurrentSupply
-[Pvt] _takeLiquidity #
-[Pub] claimTokens #
  - modifiers: onlyOwner
-[Pvt] calculateTaxFee
-[Pvt] calculateLiquidityFee

Contract functions details

-[Pvt] removeAllFee #
-[Pvt] restoreAllFee #
-[Pub] isExcludedFromFee
-[Pvt] _approve #
-[Pvt] _transfer #
-[Pvt] swapAndLiquify #
- modifiers: lockTheSwap
-[Pvt] swapTokensForEth #
-[Pvt] addLiquidity #
-[Pvt] _tokenTransfer #
-[Pvt] _transferStandard #
-[Pvt] _transferToExcluded #
-[Pvt] _transferFromExcluded #

($) = payable function
= non-constant function

Issues Checking Status

No.	Title	Status
1.	Compiler error	Passed
2.	Missing Input Validation	Passed
3.	Race conditions and Reentrancy. Cross-function race conditions.	Passed
4.	Possible delays in data delivery	Passed
5.	Oracle calls.	Passed
6.	Timestamp dependence.	Passed
7.	Integer Overflow and Underflow	Passed
8.	DoS with Revert.	Passed
9.	DoS with block gas limit.	Medium Issue
10.	Methods execution permissions.	Passed
11.	Economy model of the contract.	Passed
12.	Private use data leaks.	Passed
13.	Malicious Event log.	Passed
14.	Scoping and Declarations.	Passed
15.	Uninitialized storage pointers.	Passed
16.	Arithmetic accuracy.	Passed
17.	Design Logic.	Passed
18.	Safe Open Zeppelin contracts implementation and usage.	Passed
19.	Incorrect Naming State Variable	Passed
20.	Too old version	Low issue

Severity Definitions

Risk Level	Description
Critical	Critical vulnerabilities are usually straightforward to exploit and can lead to assets loss or data manipulations.
High	High-level vulnerabilities are difficult to exploit; however, they also have a significant impact on smart contract execution, e.g., public access to crucial functions
Medium	Medium-level vulnerabilities are important to fix; however, they can't lead to assets loss or data manipulations.
Low	Low-level vulnerabilities are mostly related to outdated, unused, etc. code snippets that can't have a significant impact on execution.

Security Issues

Critical Severity Issues

No critical severity issue found.

High Severity Issues

No high severity issue found.

Medium Severity Issues

One medium severity issue found.

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.

  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.

- **Recommendation**

  Check that the excluded array length is not too big

Low Severity Issues

One low severity issue found.

1. Old compiler version

- **Description**

  Contract has been deployed using too old solidity version.

- **Recommendation**

  It is advisable to deploy contract using any of the latest version of solidity.

Centralization

Owner privileges :

- METAZOON TOKEN Contract:
    - Owner Can Change The Tax And Liquidity Fee.
    - Owner Can Change The Maximum Transaction Amount.
    - Owner Can Exclude From The Fee.
    - Owner Can Claim All Tokens From Contract Balance.
    - Owner Can Change Numtokensselltoaddtoliquidity.
    - Owner Can Lock And Unlock. By The Way, Using These Functions The Owner Could Leave As Owner Even After The Ownership Was Renounced.

This smart contract has some functions which can be executed by the admin (Owner) only. If the admin wallet private key would be compromised, then it would create trouble, as smart contract ownership has not been renounced.

- renounceOwnership
- transferOwnership
- lock
- excludeFromReward
- includeinReward
- excludeFromFee
- includeinFee
- settaxFeePercent
- setliquidityFeePercent
- setNumTokensSellToAddToLiquidity
- setMaxTxPercent
- setSwapAndLiquifyEnabled
- claimTokens

Conclusion

Smart contract contains low and medium severity issues! The further transfer and operations with the fund raised are not related to this particular contract.

Liquidity locking details NOT provided by the team.

HackSafe 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.