



Smart Contract Security Audit Report

Archie Neko

January 2023

Security Status



www.hacksafe.io



Audit Details



Audited project

Archie Neko



Deployer address

0x088b1fb7919fe8e25377590ac759d4cd73378400



Client contacts

Archie Neko



Blockchain

Ethereum



Website

<https://www.archieneko.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) (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 Archie Neko to perform an audit of smart contracts:

- <https://etherscan.io/token/0xFE5F69dfa2d4501E78078266F6d430c079098f90#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 understood 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.

Contract Details

Token contract details for 27.01.2023

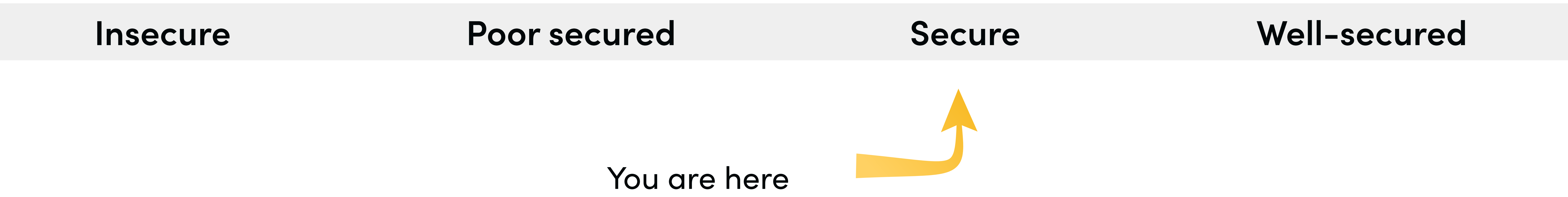
Token Type	: DEFI
Contract name	: ArhcieNeko
Contract address	: 0xFE5F69dfa2d4501E78078266F6d430c079098f90
Total supply	: 10,000,000,000,000,000,000,000
Token ticker	: ARCHIE
Decimals	: 9
Token Holders	: 1,038
Transactions count	: 4,232
Compiler version	: v0.8.9+commit.e5eed63a
Contract deployer address	: 0x088b1fb7919fe8e25377590ac759d4cd73378400
Owner address	: 0x0f7B55e157f91528f1eE480727785F68ffDF8627

Social profiles

Twitter profile	: https://twitter.com/archieneko
Telegram profile	: https://t.me/archieneko_universal

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.



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.

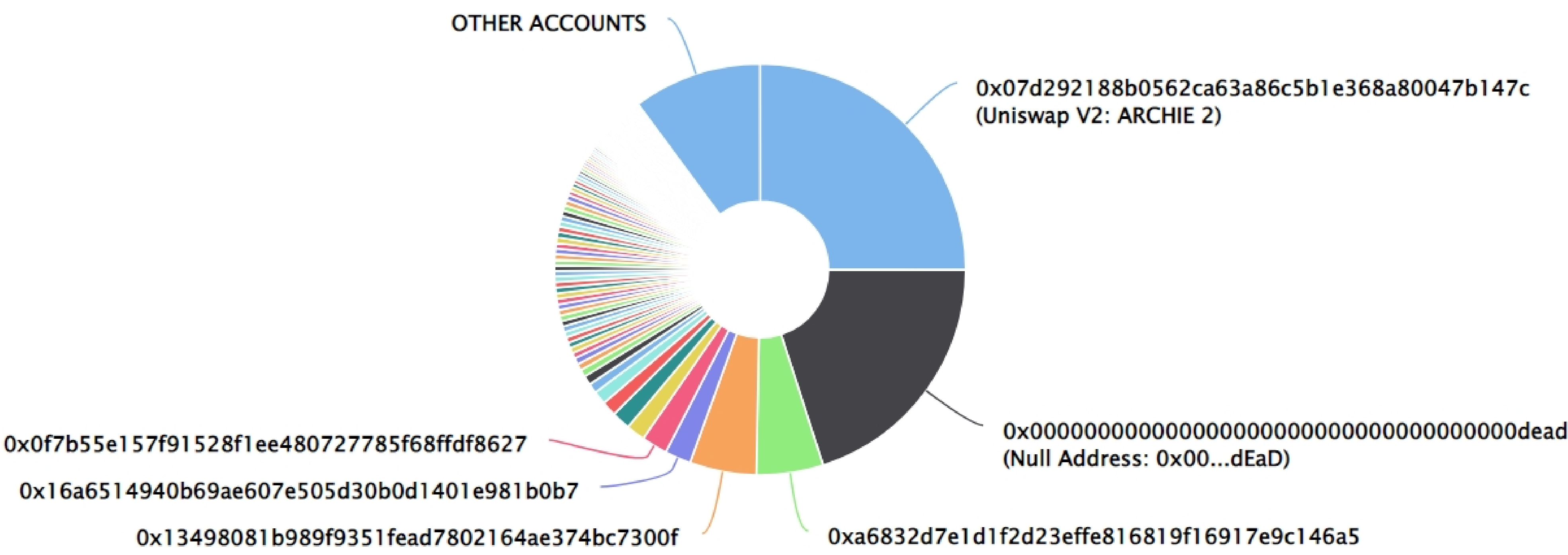
We found 0 critical, 0 high, 1 medium and 0 low.

Archie Neko Token Distribution

💡 The top 100 holders collectively own 89.87% (8,987,366,442,108,750,000,000.00 Tokens) of Archie Neko



💡 Token Total Supply: 10,000,000,000,000,000,000.00 Token | Total Token Holders: 1,038

Archie Neko Top 100 Token Holders
Source: Etherscan.io



Archie Neko Top 20 Token Holders

(A total of 8,987,366,442,108,750,000,000.00 tokens held by the top 100 accounts from the total supply of 10,000,000,000,000,000,000.00 token)

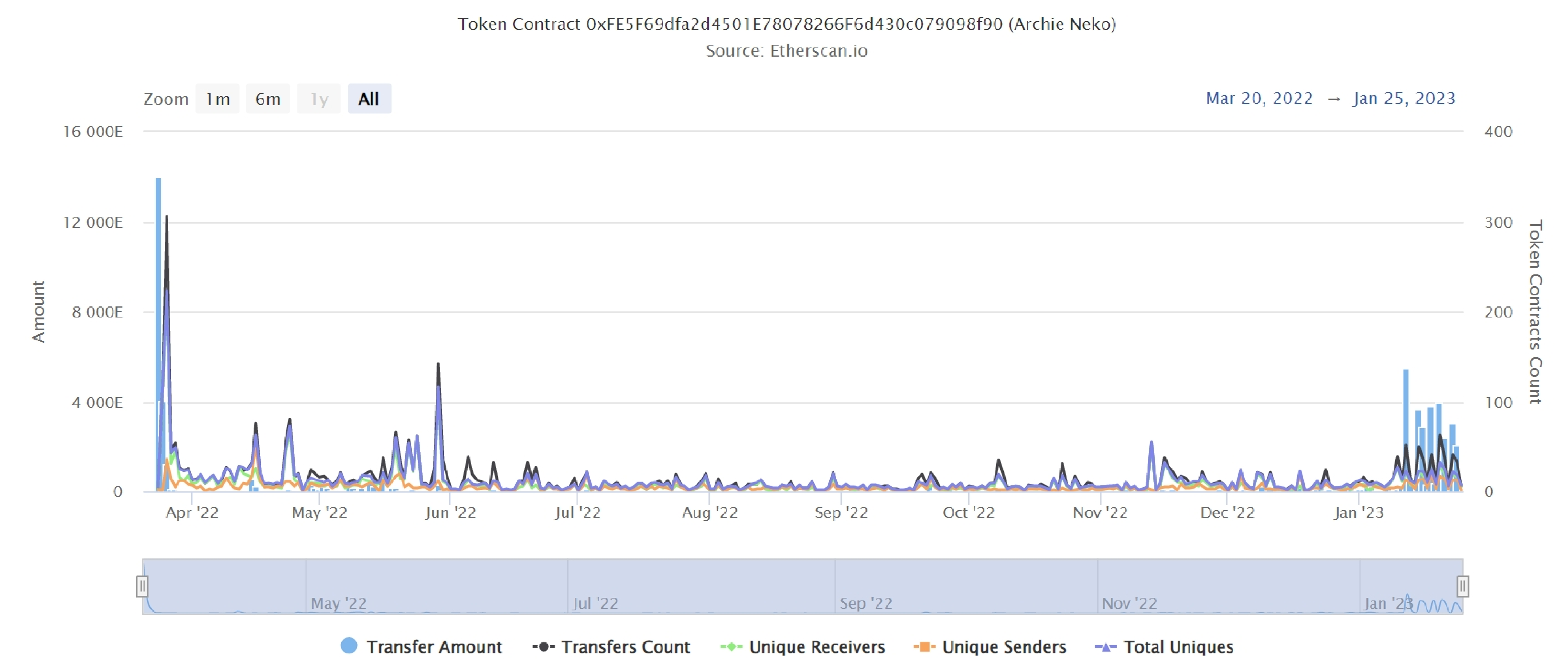
Rank	Address	Quantity (Token)	Percentage
1	 Uniswap V2: ARCHIE 2	2,504,905,035,360,070,000,000.524906076	25.0491%
2	Null Address: 0x00...dEaD	2,000,000,000,000,000,000	20.0000%
3	0xa6832d7e1d1f2d23effe816819f16917e9c146a5	525,356,168,232,531,000,000.097096087	5.2536%
4	0x13498081b989f9351fead7802164ae374bc7300f	525,356,168,232,531,000,000.097096087	5.2536%
5	 0x16a6514940b69ae607e505d30b0d1401e981b0b7	205,718,117,944,027,000,000.447612586	2.0572%
6	0x0f7b55e157f91528f1ee480727785f68ffdf8627	203,143,151,061,485,000,000.692155285	2.0314%
7	0xf628a5b7f19a46f4e1bdfceec4585130b2db2201	148,179,676,717,096,000,000.078529501	1.4818%
8	0x40eb459af0ac9252f4395482aa4ad9260c8f83f6	146,049,501,156,447,000,000.833283188	1.4605%
9	0x3b503d47db896c1888be0ff580495ed7c6ed9958	117,003,148,923,711,000,000.969615821	1.1700%
10	0xea308c85d039c295ab1d07256eab8fd840306555	104,783,762,333,496,000,000.176153877	1.0478%
11	0xc05001d951a52122691302acdf445894643af8da	75,768,636,511,201,800,000.582979295	0.7577%
12	0x03dddade62b9bfbdbf9b9b15989c1cde5f4008283	73,144,789,523,595,200,000.274597207	0.7314%
13	0x516e80ee68eb729d17d15973ca9b61d0d51a6d63	57,865,541,794,332,300,000.219392197	0.5787%
14	0x08a1dc2b2cbd7d2fd92ee3621adcbba95fbdde8c	51,279,351,474,579,600,000.769344628	0.5128%
15	0x6e076a5664489fa0924650d50d2927652ae59693	50,572,839,238,599,100,000.463413737	0.5057%
16	0xa21c8e00bbcd6833c14172177f065739ec7e90e	43,779,680,686,044,300,000.252640176	0.4378%
17	0x45b160c0447a96338c96fd0306b71ad09c08f74c	43,779,680,686,044,300,000.252640176	0.4378%
18	0x442db0c12aa5f66766a3186f2e3acf8132305078	43,779,680,686,044,300,000.252640176	0.4378%
19	0x13861bef1fabcd90355b9b8be6109e202aeb81618	43,779,680,686,044,300,000.252640176	0.4378%
20	0xfc04b8494655471ec2dac1c73c01f158d7929683	43,779,680,686,044,300,000.252640176	0.4378%

Archie Neko Token Distribution

Archie Neko Contract overview

Time Series: Token Contract Overview

Thu 24, Mar 2022 - Wed 25, Jan 2023



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]` <constructor> `#`
- `- [Pub]` owner
- `- [Pub]` renounceOwnership `#`
 - modifiers: onlyOwner
- `- [Pub]` transferOwnership `#`
 - modifiers: onlyOwner

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 #

+ ArhcieNeko (Context, IERC20, Ownable)

- [Pub] <Constructor> #
- [Pub] name
- [Pub] symbol
- [Pub] decimals
- [Pub] totalSupply
- [Pub] balanceOf
- [Pub] transfer #
- [Pub] allowance
- [Pub] approve #
- [Pvt] _approve #
- [Pub] transferFrom #

Contract functions details

- [Pub] increaseAllowance #
- [Pub] decreaseAllowance #
- [Ext] burn #
- [Int] _burnTokens #
- [Int] _transfer #
- [Pub] _getRateTokenAndETH
- [Int] swapAndLiquify #
 - modifiers: lockTheSwap
- [Int] swapTokensForEth #
- [Int] addLiquidity #
- [Int] _tokenTransfer #
- [Int] removeAllTax #
- [Int] setTaxBuyTax #
- [Int] setTaxSellTax #
- [Pvt] _transferStandard #
- [Pvt] _transferToExcluded #
- [Pvt] _transferFromExcluded #
- [Pvt] _transferBothExcluded #
- [Pvt] _takeLiquidity #
- [Pvt] _takeMarketing #
- [Pvt] _takeTreasury #
- [Pvt] _takeFoundation #
- [Pvt] _reflectTax #
- [Pvt] _getValues
- [Pvt] _getTValues
- [Pvt] _getRValues
- [Pvt] _getRate
- [Pvt] _getCurrentSupply
- [Pvt] calculateReflectionTax
- [Pvt] calculateTotalTax
- [Pub] tokenFromReflection
- [Pub] reflectionFromToken
- [Pub] isExcludedFromReward
- [Pub] totalFees
- [Pub] deliver #
- [Pub] isExcludedFromFee
- [Pub] gettransactionDataAmount
- [Pub] gettransactionDataTime

Contract functions details

- [Pub] getTransactionDataIsLocked
- [Pub] getTransactionDataLockedTime
- [Pub] getTransactionDataLockedPeriod
- [Pvt] transferToAddressETH #
- [Pub] excludeFromReward #
 - modifiers: onlyOwner
- [Ext] includeInReward #
 - modifiers: onlyOwner
- [Pub] excludeFromFee #
 - modifiers: onlyOwner
- [Pub] includeInFee #
 - modifiers: onlyOwner
- [Ext] setBuyTaxPercent #
 - modifiers: onlyOwner
- [Ext] setSellTaxPercent #
 - modifiers: onlyOwner
- [Ext] setMarketingAddress #
 - modifiers: onlyOwner
- [Ext] setTreasuryAddress #
 - modifiers: onlyOwner
- [Ext] setFoundationAddress #
 - modifiers: onlyOwner
- [Pub] setSwapAndLiquifyEnabled #
 - modifiers: onlyOwner
- [Ext] setEnableTrading #
 - modifiers: onlyOwner
- [Ext] setNumTokensSellToAddToLiquidityPercentageAndmaxwalletamount #
 - modifiers: onlyOwner
- [Ext] setDailymaxTxAmountAndmaxTxAmount #
 - modifiers: onlyOwner
- [Ext] setAutomatedMarketMakerPairs #
 - modifiers: onlyOwner
- [Ext] lockAccount #
 - modifiers: onlyOwner
- [Ext] unlockAccount #
 - modifiers: onlyOwner
- [Ext] SetAccountMaxWalletLimit #
 - modifiers: onlyOwner

Contract functions details

- [Ext] airdrop #
 - modifiers: onlyOwner
- [Int] airdropInternal #
- [Ext] airdropArray #
 - modifiers: onlyOwner
- [Ext] <Fallback > (\$)

(\$)= 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	Passed

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.

- **Issue:**

The function `airdropArray()` uses the loop to airdrop tokens amounts list to addresses list. Function will be aborted with `OUT_OF_GAS` exception if there will be a long lists.

- **Recommendation**

Check that the array's length is not too big.

✔ Low Severity Issues

No low severity issue found.

Centralization

Owner Privileges

- Deepmaze Coin Contract:
 - Owner can exclude addresses from the fees.
 - Owner can change all fees.
 - Owner can change fee receivers addresses.
 - Owner can enable/disable swap and liquify.
 - Owner can enable trading.
 - Owner can change numTokensSellToAddToLiquidity and _maxwalletamount.
 - Owner can change _dailymaxTxAmount and _maxTxAmount.
 - Owner can mark addresses as automatedMarketMakerPairs.
 - Owner can lock/unlock users accounts.
 - Owner can exclude from max transaction amount.
 - Owner can airdrop tokens.

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.

Conclusion

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

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.