Techrate1 Techrate

TechRate

AUDIT COMPANY

Smart Contract Security Audit

TechRate

November, 2021

Audit Details

Audited project

JUNGLE BOOK CRYPTO

Deployer address

0x0725780e2bdfc8f8c743c022237acff80c5519a1

Client contacts:

JUNGLE BOOK CRYPTO team

Blockchain

Binance Smart Chain

Project website:

Not provided by JUNGLE BOOK CRYPTO team

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 JUNGLE BOOK CRYPTO to perform an audit of smart contracts:

https://bscscan.com/address/0xD10c40eae9B675EEAAF20C84E5D274aBC109281F#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

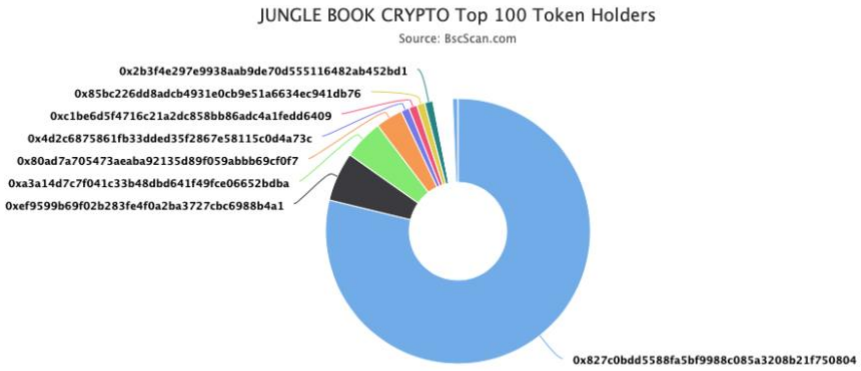
Token contract details for 12.11.2021

Contract name	JUNGLE BOOK CRYPTO
Contract address	0xD10c40eae9B675EEAAF20C84E5D274aBC109281F
Total supply	999,945,767,010,526.639
Token ticker	JBC
Decimals	18
Token holders	238
Transactions count	957
Top 100 holders dominance	99.34%
Liquidity fee	30
Burn fee	30
JBCHub fee	20
Pancake V2 pair	0x4e6ad3066089992d035c5831912a2dfbafac4752
Contract deployer address	0x0725780e2bdfc8f8c743c022237acff80c5519a1
Contract's current owner address	0x827c0bdd5588fa5bf9988c085a3208b21f750804

JUNGLE BOOK CRYPTO Token Distribution

The top 100 holders collectively own 99.34%
(993,318,298,038,927.00 Tokens) of JUNGLE BOOK CRYPTO

Token Total Supply: 999,945,767,010,526.64 Token | Total Token Holders: 238

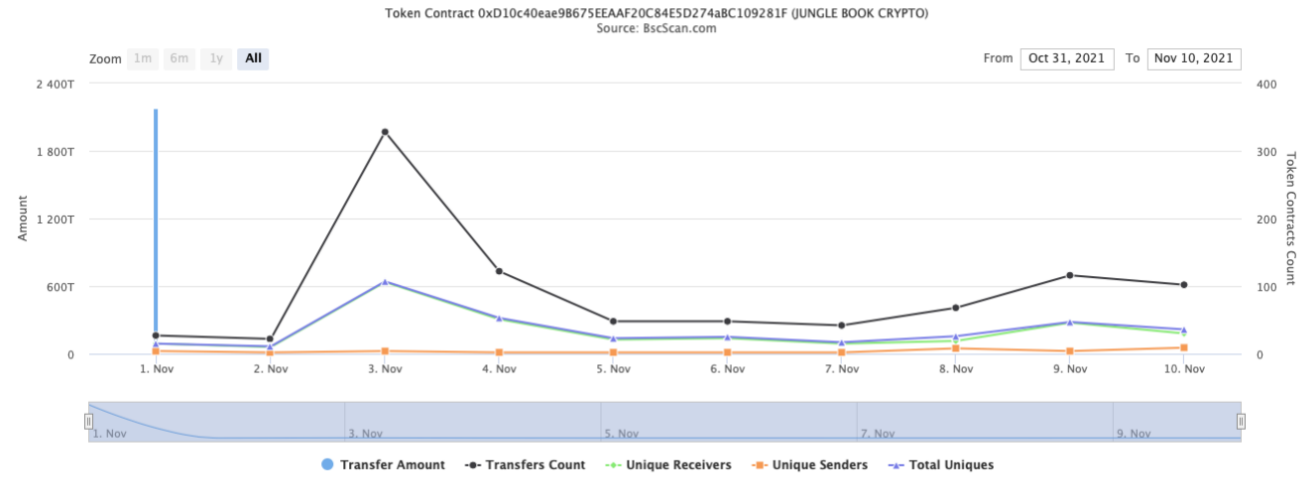


(A total of 993,318,298,038,927.00 tokens held by the top 100 accounts from the total supply of 999,945,767,010,526.64 token)

JUNGLE BOOK CRYPTO Contract Interaction Details

Time Series: Token Contract Overview

Mon 1, Nov 2021 - Wed 10, Nov 2021



JUNGLE BOOK CRYPTO Top 10 Token Holders

Rank	Address	Quantity (Token)	Percentage
1	0x827c0bdd5588fa5bf9988c085a3208b21f750804	787,736,656,452,282.1	78.7779%
2	0xef9599b69f02b283fe4f0a2ba3727cbc6988b4a1	60,000,000,000,000	6.0003%
3	0xa3a14d7c7f041c33b48dbd641f49fce06652bdba	48,724,750,087,568.72701815	4.8727%
4	0x80ad7a705473aeaba92135d89f059abbb69cf0f7	33,000,000,000,000	3.3002%
5	0x4d2c6875861fb33dded35f2867e58115c0d4a73c	10,000,000,000,000	1.0001%
6	0xc1be6d5f4716c21a2dc858bb86adc4a1fedd6409	10,000,000,000,000	1.0001%
7	0x85bc226dd8adcb4931e0cb9e51a6634ec941db76	10,000,000,000,000	1.0001%
8	0x2b3f4e297e9938aab9de70d555116482ab452bd1	10,000,000,000,000	1.0001%
9	0x0c6cbcb37f32298e2490b9cd8ea6abaf50b494a7	754,091,469,121.379997	0.0754%
10	0xaac6c485b57626a92b3801ad1f3c34e2f8951186	379,048,000,919.619996	0.0379%

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

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

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

+ [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 #

+ BEP20 (Context, IBEP20)
  - [Pub] <Constructor> #
  - [Pub] Owner
  - [Pub] transferOwnership #
    - modifiers: onlyOwner
  - [Pub] name
  - [Pub] symbol
  - [Pub] decimals
  - [Pub] totalSupply
  - [Pub] balanceOf

  - [Pub] transfer #
  - [Pub] allowance
  - [Pub] approve #
  - [Pub] transferFrom #
  - [Pub] excludeFromFee #
    - modifiers: onlyOwner
  - [Pub] includeFromFee #
    - modifiers: onlyOwner
  - [Pub] setSwapAndLiquifyEnabled #
    - modifiers: onlyOwner
  - [Ext] <Fallback> ($)
  - [Prv] _approve #
  - [Prv] _transfer #
  - [Prv] swapAndLiquify #
    - modifiers: lockTheSwap
  - [Prv] swapTokensForEth #
  - [Prv] addLiquidity #
  - [Int] _tokenTransfer #
  - [Int] transferFee
  - [Int] charityTransfer #
  - [Int] liquidityTransfer #
  - [Int] jbcHubTransfer #
  - [Int] referralTransfer #
  - [Int] _mint #
  - [Int] _burn #
  - [Pub] burn #
    - modifiers: onlyOwner
  - [Pub] setThresholdLimit #
    - modifiers: onlyOwner
  - [Pub] setBurnFee #
    - modifiers: onlyOwner
  - [Pub] lockTime #
    - modifiers: onlyOwner
  - [Pub] setTeamMember #
    - modifiers: onlyOwner
  - [Pub] start #
    - modifiers: onlyOwner
  - [Pub] setCharityFee #
    - modifiers: onlyOwner
  - [Pub] setJBChubFee #
    - modifiers: onlyOwner
  - [Pub] setreferralFee #
    - modifiers: onlyOwner
  - [Pub] setliquidityFee #
    - modifiers: onlyOwner
  - [Int] _beforeTokenTransfer #
  - [Int] _afterTokenTransfer #

+ JungleBookCrypto (BEP20)
  - [Pub] <Constructor> #
    - modifiers: BEP20

($) = 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.	Passed
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

No low severity issues found.

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

- Owner can include in and exclude from the fee.
- Owner can enable / disable swap and liquify.
- Owner can burn.
- Owner can update threshold limit.
- Owner can change the burn fee.
- Owner can lock transfer for any address.
- Owner can change liquidity time.
- Owner can change fees.

Conclusion

Smart contracts do not contain high severity issues! Liquidity pair contract's security is not checked due to out of scope.

Liquidity locking details NOT provided by the team.

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.