PapaExchange

Not just a bunch of Dads

Papa Exchange Audit for

ZomBear

Audit Details

ZomBear

Blockckain - Binance Smart Chain

Website - zombear.monster

Auditor's - Papa Audit Team

Date Issued

Oct. 12th 2022

DISCLAIMER

PapaExchange LLP audits and reports should not be considered as a form of project's "advertisement" and does not cover any interaction and assessment from "project's contract" to "external contracts" such as Pancakeswap or similar.

PapaExchange LLP does not provide any warranty on its released reports.
PapaExchange LLP should not be used as a decision to invest into an audited project please do your own research. PapaExchange LLP provides transparent reports to all its "clients" and to its "clients participants" and will not claim any guarantee of bug-free code within its Smart Contract.

PapaExchange LLP presence is to analyze, audit and assess the client's smart contract's code.
Each company or project shall be liable for its own security flaws and functionalities.

Scope of Work & Background

The main focus of this report/audit, is to document an accurate assessment of the condition of the smart contract and whether it has any security flaws in the implementation of the contract.
MoopToken team agreed and provided us with the files that needed to be tested (Through Github, Bscscan, files, etc.). PapaExchange will be focusing on contract issues and functionalities along with the projects claims from smart contract to their website, whitepaper and repository where available, which has been provided by the project.
Code is reviewed manually and with the use of software using industry best practices.

Background

PapaExchange was commissioned by ZomBear to perform an audit of smart contract:
● Contract Address 0x0049255E1f7847cBD71E32269D834A83274Fe729
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.

Developers Token Description

The Zombear token ($ZBEAR) is the latest treasury project launched by CryptoCravers LLC. This project builds on lessons learned from the last 3 bear projects: the main BEAR project that is now three months old, and two sequential surrogate projects (Ursa Moon and Bear Cub). Both of which have been successfully completed and treasury disbursed to its holders.

Social Media Links

Twitter: https://twitter.com/ZomBearToken

Telegram: http://T.me/zombeartoken

Facebook: N/A

Discord: N/A

Contracts details

(ZomBear Contract details for Oct. 10th, 2022)

Contract/Project name: ZomBear

Description: Utility and Reward Token

Compiler version: 0.8.17+

Contract address: 0x0049255E1f7847cBD71E32269D834A83274Fe729

Total supply: 2,000,000

Token ticker: ZBEAR

Decimals: 9

Token holders at time of report: 60

Transaction count at time of report: 583

Top 100 holders dominance: 100%

Contract deployer address: 0xDB7c3964f1b88874d570313a68bb1Db4a44a78c5
Contract's current owner address: 0xDB7c3964f1b88874d570313a68bb1Db4a44a78c5

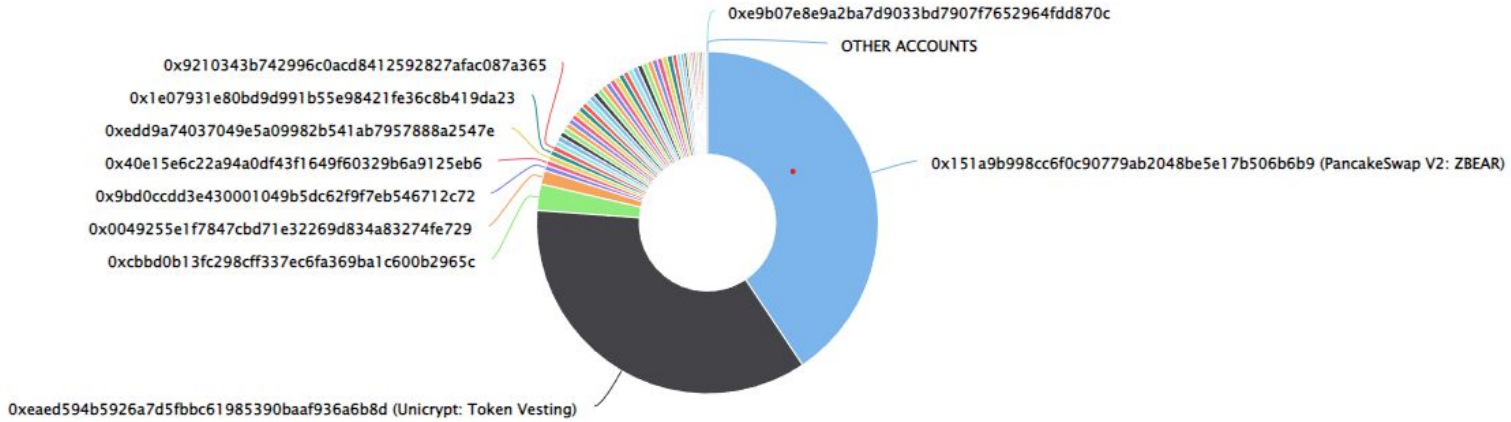
LP Lock: Mudra 2 weeks

KYC done by FUDDOX (through previous project)

Launch Type Fair

ZomBear Top 100 Token Holders

Source: BscScan.com



(A total of 2,000,000.00 tokens held by the top 100 accounts from the total supply of 2,000,000.00 token)

ZomBear LP TOKEN HOLDER

1. 0xae7e6cabad8d80f0b4e1c4dde2a5db7201ef1252 (Mudra - 100%)

Owner Contract write functions details

Owner privileges:

Ownership has NOT been renounced. The owner has privileges and has authority to make some changes now.

Current Fees: • Buy: 12% • Sell: 12% • Owner cannot change fees above 23%.

All Write Functions of Contract that can be adjusted after the contract is deployed.

1. addMaxWalletExempt
2. addTaxExempt
3. Approve
4. changeLiqOnBuy
5. changeLiqOnSell
6. changeLiqSwapAt
7. changeLiqWallet
8. changeMarketingOnBuy
9. changeMarketingOnSell
10. changeMarketingSwapAt
11. changeMarketingWallet
12. changeMaxTxPercent
13. changeMaxWalletSize
14. changeSwapOnSell
15. changeTaxesOnBuy
16. changeTaxesOnSell
17. changeTreasurySwapAt
18. changeTreasuryTaxOnBuy
19. changeTreasuryTaxOnSell
20. changeTreasuryWallet
21. manageBotWallets
22. removeBotWallet
23. removeMaxWalletExempt
24. removeTaxExempt
25. renounceBotListAbility
26. renounceOwnership
27. transfer
28. transferFrom
29. transferOwnership
30. updateLiqCount
31. updateMarketingCount
32. updateTreasuryCount
33. withdrawContractBNB
34. withdrawContractTokens

SWC Registry: Smart Contract Weakness/Vulnerabilities

SWC-136	Unencrypted Private Data On-Chain	PASSED
SWC-135	Code With No Effects	PASSED
SWC-134	Message call with hardcoded gas amount	PASSED
SWC-133	Hash Collisions with Multiple Variable Length Arguments	PASSED
SWC-132	Unexpected Ether balance	PASSED
SWC-131	Presence of unused variables	PASSED
SWC-130	Right-To-Left-Override control character (U+202E)	PASSED
SWC-129	Typographical Error	PASSED

SWC-128	DoS With Block Gas Limit	PASSED
SWC-127	Arbitrary Jump with Function Type Variable	PASSED
SWC-126	Insufficient Gas Griefing	PASSED
SWC-125	Incorrect Inheritance Order	PASSED
SWC-124	Write to Arbitrary Storage Location	PASSED
SWC-123	Requirement Violation	PASSED
SWC-122	Lack of Proper Signature Verification	PASSED
SWC-119	Shadowing State Variables	PASSED

SWC-118	Incorrect Constructor Name	PASSED
SWC-120	Weak Sources of Randomness from Chain Attributes	PASSED
SWC-117	Signature Malleability	PASSED
SWC-116	Block values as a proxy for time	PASSED
SWC-115	Authorization through tx.origin	PASSED
SWC-114	Transaction Order Dependence	PASSED
SWC-121	Missing Protection against Signature Replay Attacks	PASSED
SWC-113	DoS with Failed Call	PASSED

SWC-112	Delegatecall to Untrusted Callee	PASSED
SWC-111	Use of Deprecated Solidity Functions	PASSED
SWC-110	Assert Violation	PASSED
SWC-109	Uninitialized Storage Pointer	PASSED
SWC-108	State Variable Default Visibility	PASSED
SWC-107	Reentrancy	PASSED
SWC-106	Unprotected SELFDESTRUCT Instruction	PASSED
SWC-105	Unprotected Ether Withdrawal	PASSED

SWC-104	Unchecked Call Return Value	PASSED
SWC-103	Floating Pragma	LOW ISSUE
SWC-102	Outdated Compiler Version	PASSED
SWC-101	Integer Overflow and Underflow	PASSED

Issue Checking

Manual code review is satisfactory.

CLOSING NOTES

Whilst there are limitless ownable callable functions that have the potential to be dangerous, they are not overtly so. Trust in the team would mitigate many of these risks. Please make sure you do your own research. If in doubt please contact the project team.

Always make sure to inspect all values and variables.
This includes, but is not limited to: • Ownership • Proper Ownership Renouncement (if any) • Taxes • Transaction/Wallet Limits • Token Distributions • Timelocks • Liquidity Locks • Any other owner-adjustable settings or variables.

OVERALL ASSESSMENT

SATISFACTORY