OXSCANS

MetaZero

AI Generated at 03:19 PM, UTC

February 22, 2024

OVERVIEW

This audit has been perpared for 'MetaZero' to review the main aspects of the project to help investors make an informative decision during their research process

You will find a summarized review of the following key points:

- **Contract's source code**
- **Owner wallets**
- **Tokenomics**
- **Team transparency and goals**
- **Website's age, code, security and UX**
- **Whitepaper and roadmap**
- **Social media and online presence**

Table of Content

General Information

MetaZero

MetaZero is creating a Synthetic Liquidity Layer Protocol for Cross-chain (Omnichain) Tokenization of Gaming Real World Assets (RWAs).

Name	**MetaZero**
Category	**Real World Assets (RWA)** **Ethereum Ecosystem**
Info	Website Telegram Bot Docs Twitter

General Information

Tokenomics

Ticker	0X328A268B191EF593B72498A9E8A481C086EB21BE
Network	Ethereum
Contract Address	0x328a268b191ef593b72498a9e8a481c086eb21be

General Analysis

Audit Review Process

1. Testing the smart contracts against both common and uncommon vulnerabilities
2. Assessing the codebase to ensure compliance with current best practices and industry standards
3. Ensuring contract logic meets the specifications and intentions of the client
4. Cross referencing contract structure and implementation against similar smart contracts produced by industry leaders
5. Thorough line-byline AI review of the entire codebase by industry

Token Transfer Stats

Transactions (Latest Mine Block)	Token holders	Compiler
1	3647	v0.8.22

Smart Contract Stats

Functions	Events	Constructor
67	13	1

Detail Analysis

Threat Level

● High — Issues on this level are critical to the smart contract's performace/functionality and should be fixed before moving to a live enviroment

● Medium — Issues on this level are critical to the smart contract's performace/functionality and should be fixed before moving to a live enviroment

● Low — Issues on this level are minor details and warning that can remain unfixed

● Informational — Informational level is to offer suggestions for improvement of efficacy or secruity for fratures with risk free factor

Threat Level

● High	**6** threats found
● Medium	**0** threats found
● Low	**1** threats found
● Informational	**1** threats found

Detail Analysis

Vulnerability Check 🟢 14 Passed 🔴 7 Fail

- 🟢 Arbitrary Jump/Storage Write
- 🟢 Centralization of Control
- 🟢 Compiler Issues
- 🟢 Delegate Call to Untrusted Contract
- 🔴 Dependence on Predictable Variables
- 🟢 Ether/Token Theft
- 🟢 Flash Loans
- 🔴 Front Running
- 🟢 Improper Events
- 🔴 Improper Authorization Scheme
- 🟢 Integer Over/Underflow
- 🔴 Logical Issues
- 🟢 Oracle Issues
- 🟢 Outdated Compiler Version
- 🔴 Race Conditions
- 🔴 Reentrancy
- 🟢 Signature Issues
- 🟢 Sybil Attack
- 🟢 Unbounded Loops
- 🟢 Unused Code
- 🔴 Overall Contract Safety

Detail Analysis

Detail Analysis

🟢 14 Passed 🔴 7 Fail

CATEGORY	STATUS	NOTES
Arbitrary Jump/Storage Write	🟢	The contract does not perform any low-level calls that could result in arbitrary jumps or storage writes.
Centralization of Control	🟢	No risk of centralization
Compiler Issues	🟢	The contract is compiled with Solidity version 0.8.19, which is a recent and stable version.
Delegate Call to Untrusted Contract	🟢	No delegatecall to untrusted contracts is present in the contract code.
Dependence on Predictable Variables	🔴	The contract relies on block numbers for setting taxes, which can be predicted by miners.

Detail Analysis

Detail Analysis 🟢 14 Passed 🔴 7 Fail

CATEGORY	STATUS	NOTES
Ether/Token Theft	🟢	No functions are exposed that could lead to Ether or token theft without proper authorization.
Flash Loans	🟢	The contract does not interact with flash loans.
Front Running	🔴	Public functions like 'swapAndSend' could potentially be front-run by miners or bots.
Improper Events	🟢	All state-changing functions correctly emit events.
Improper Authorization Scheme	🔴	The contract uses a single owner for authorization, which could be improved by using a multi-signature scheme or decentralized governance.
Integer Over/Underflow	🟢	The contract uses Solidity 0.8.x, which has built-in overflow/underflow protection.

Detail Analysis

Detail Analysis

🟢 14 Passed 🔴 7 Fail

CATEGORY	STATUS	NOTES
Logical Issues	🔴	The contract has logical issues related to the dynamic tax system based on block numbers, which could be exploited by miners.
Oracle Issues	🟢	The contract does not use external oracles.
Outdated Compiler Version	🟢	Uses a recent and stable version of the Solidity compiler.
Race Conditions	🔴	The contract's functions are not protected against reentrancy attacks, which could lead to race conditions.
Reentrancy	🔴	The contract lacks reentrancy protection for functions such as 'swapAndSend'.
Signature Issues	🟢	The contract does not use message signatures that could be vulnerable.

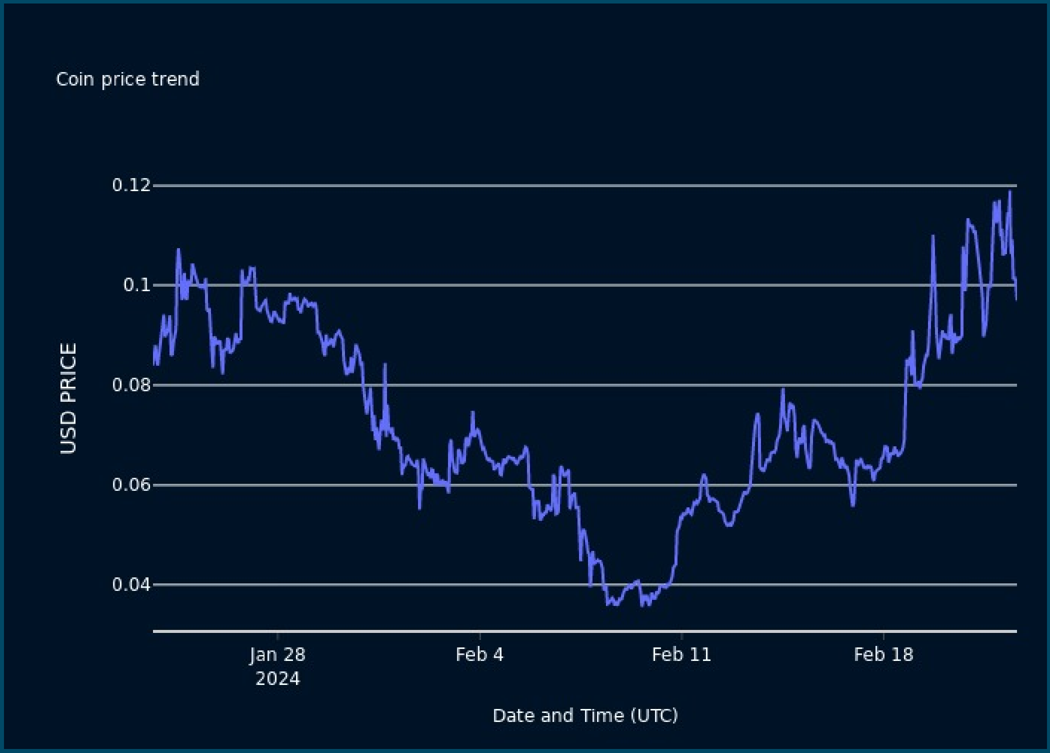
Detail Analysis

Detail Analysis

🟢 14 Passed 🔴 7 Fail

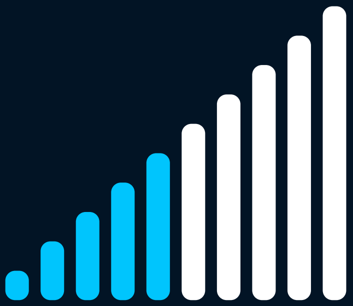
CATEGORY	STATUS	NOTES
Sybil Attack	🟢	The contract is not susceptible to Sybil attacks.
Unbounded Loops	🟢	The contract does not contain any unbounded loops that could lead to denial of service.
Unused Code	🟢	No significant amount of unused code is present in the contract.
Overall Contract Safety	🔴	The contract has several critical issues related to centralization, predictable variables, authorization scheme, and lack of reentrancy protection that could affect overall safety.

Market Analysis



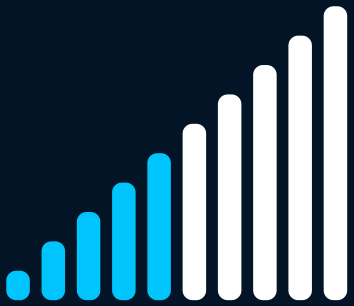
Score

Total Audit Score



5/10

Security Score



5/10

Legal Disclaimer

0xscans operates as an automated system for smart contract due diligence, acknowledging the possibility of bugs or vulnerabilities impacting token values. We do not hold specific obligations regarding your trading outcomes or the utilization of audit content. Users release 0xscans from any liability associated with content obtained through the tool.

AI generated by 0xscans AI technology

Chat with us

Telegram

For more information. Visit below:

Twitter

Github