



OXSCANS

# DecentraNode

AI Generated at 04:17 PM, UTC

March 01, 2024

## OVERVIEW

This audit has been prepared for 'DecentraNode' 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**

---

**1 General Info**

**2 General Analysis**

**3 Vulnerability check**

**4 Threat Analysis**

**5 Risks & Recommendations**

**6 Conclusions**

**7 Disclaimer**

# General Information

## DecentraNode

Name

DecentraNode

Category

Ethereum Ecosystem

Info

[Website](#)

[Telegram Bot](#)

# General Information

## Tokenomics

Ticker 0X014337B35167B3711195361BB85259009E50A8A4

Network Ethereum

Contract Address 0x014337b35167b3711195361Bb85259009e50a8a4

# 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)



1

Token holders



1019

Compiler



v0.8.23

## Smart Contract Stats

Functions



19

Events



4

Constructor



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 secuirty for fratures with risk free factor

## Threat Level

● High	0 threats found
● Medium	4 threats found
● Low	2 threats found
● Informational	2 threats found

## Detail Analysis

### Vulnerability Check



15 Passed



6 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



15 Passed



6 Fail

CATEGORY	STATUS	NOTES
Arbitrary Jump/Storage Write		No arbitrary jumps or storage writes detected, standard ERC20 and Ownable patterns used.
Centralization of Control		No risk of centralization as the owner is a dead address, eliminating the risk of owner's arbitrary control.
Compiler Issues		Compiled with a recent Solidity version (0.8.23) without known compiler issues.
Delegate Call to Untrusted Contract		No delegate calls to external contracts present in the contract.
Dependence on Predictable Variables		No critical dependence on variables like block.timestamp or block.number for crucial contract logic.

# Detail Analysis

## Detail Analysis



15 Passed



6 Fail

CATEGORY	STATUS	NOTES
Ether/Token Theft		Standard ERC20 transfer mechanisms, no functions that could lead to Ether or token theft.
Flash Loans		Flash loan attack vectors not applicable, no external calls or token price dependencies.
Front Running		Some functions might be susceptible to front-running due to public visibility and transfer mechanics, although no direct financial risk observed.
Improper Events		All external state-changing functions emit appropriate events.
Improper Authorization Scheme		No risk of improper authorization as the owner is a dead address, eliminating the risk of owner's arbitrary control.
Integer Over/Underflow		Solidity 0.8.23 inherently protects against integer overflow and underflow.

# Detail Analysis

## Detail Analysis



15 Passed



6 Fail

CATEGORY	STATUS	NOTES
Logical Issues		Logic appears sound, but some custom functions like 'openTrading' and tax-related functions need careful review to ensure they behave as intended.
Oracle Issues		No external oracles or dependencies on off-chain data.
Outdated Compiler Version		Compiled with a recent and stable version of Solidity.
Race Conditions		Potential for race conditions in functions like 'transfer' and 'approve', common in ERC20 tokens.
Reentrancy		No external calls that could lead to reentrancy attacks.
Signature Issues		No signature-based functionalities in the contract.

# Detail Analysis

## Detail Analysis



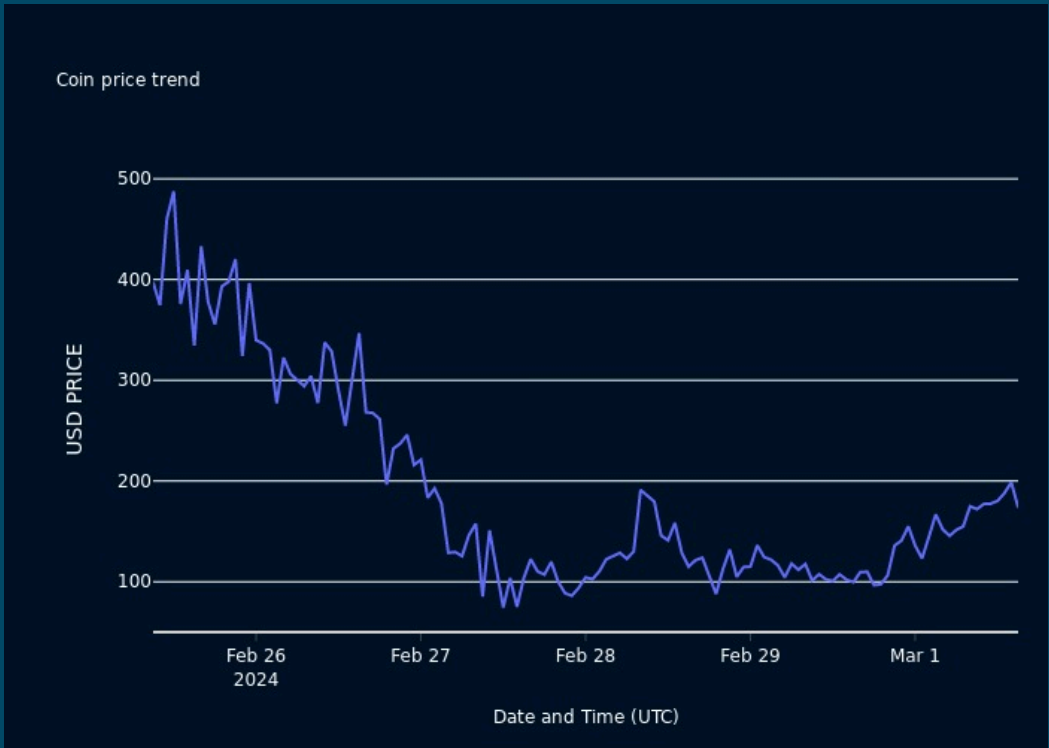
15 Passed



6 Fail

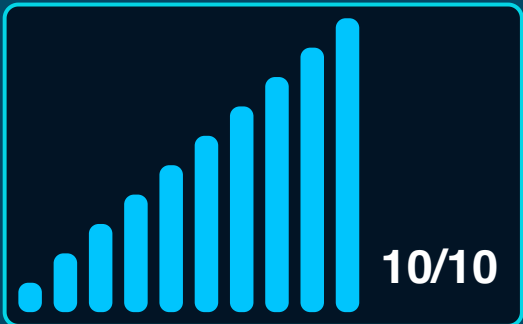
CATEGORY	STATUS	NOTES
Sybil Attack		While not directly vulnerable, ERC20 tokens can potentially be affected by sybil attacks in broader ecosystem.
Unbounded Loops		No unbounded loops that could lead to gas limit issues.
Unused Code		Some code paths are not used in current contract logic but do not pose a risk.
Overall Contract Safety		While the contract follows common ERC20 and Ownable patterns, potential logical issues warrant caution.

# Market Analysis

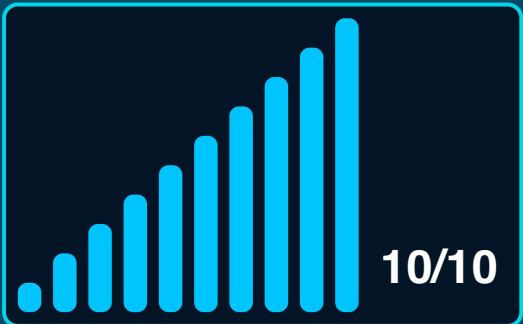


## Score

### Total Audit Score



### Security Score





## 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**