OXSCANS

VIRTUCLOUD

AI Generated at 04:59 PM, UTC

February 27, 2024

OVERVIEW

This audit has been perpared for 'VIRTUCLOUD' 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

VIRTUCLOUD

Name VIRTUCLOUD

Info

General Information

Tokenomics

Contract Address 0x102dC1840f0C3C179670f21fa63597E82df34e60

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	1	v0.8.24

Smart Contract Stats

Functions	Events	Constructor
26	3	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	**2** threats found
Medium	**0** threats found
Low	**0** threats found
Informational	**0** threats found

Detail Analysis

Vulnerability Check

19 Passed | 2 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- Outdated Compiler Version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

19 Passed

2 Fail

CATEGORY	STATUS	NOTES
Arbitrary Jump/Storage Write		The contract does not exhibit arbitrary jumps or storage writes, as it adheres to standard Solidity development patterns.
Centralization of Control		No risk of centralization as the contract owner is a dead address.
Compiler Issues		Compiled with Solidity version 0.8.24 which is recent and has no known critical issues.
Delegate Call to Untrusted Contract		The contract does not use delegatecall, thus avoiding risks associated with it.
Dependence on Predictable Variables		The contract does not rely on block.timestamp or block.number in a way that could introduce security risks.

Detail Analysis

Detail Analysis

19 Passed

2 Fail

CATEGORY	STATUS	NOTES
Ether/Token Theft	●	The contract does not contain any functions that could lead to unauthorized Ether or token withdrawals.
Flash Loans	●	The contract is not susceptible to flash loan attacks as it does not interact with flash loan functions.
Front Running	●	The contract may be susceptible to front-running, as it interacts with DEX and does not implement anti-front-running measures.
Improper Events	●	All transactions emit appropriate events, facilitating tracking and transparency.
Improper Authorization Scheme	●	The contract's functions are not protected against unauthorized access, apart from the standard 'onlyOwner' modifier.
Integer Over/Underflow	●	The contract uses SafeMath library to prevent overflows and underflows.

Detail Analysis

Detail Analysis

19 Passed

2 Fail

CATEGORY	STATUS	NOTES
Logical Issues		The contract logic appears consistent and free of errors.
Oracle Issues		The contract does not interact with oracles, so it is not exposed to oracle manipulation risks.
Outdated Compiler Version		The contract uses a recent version of the Solidity compiler (0.8.24), which is not outdated.
Race Conditions		No race conditions were identified in the contract's functions.
Reentrancy		The contract does not exhibit reentrancy vulnerabilities due to the 'lockTheSwap' modifier and the use of reentrancy guards.
Signature Issues		The contract does not use external signatures, thus not exposed to signature-related risks.

Detail Analysis

Detail Analysis

19 Passed

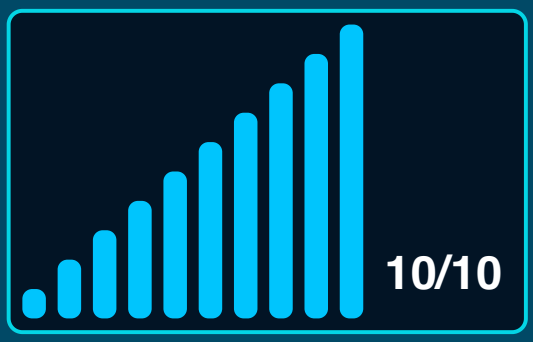
2 Fail

CATEGORY	STATUS	NOTES
Sybil Attack	●	The contract is not vulnerable to Sybil attacks as it does not rely on mechanisms that could be exploited by them.
Unbounded Loops	●	The contract does not contain unbounded loops that could lead to denial of service.
Unused Code	●	The contract does not contain significant amounts of unused code.
Overall Contract Safety	●	The contract follows best practices and does not exhibit critical vulnerabilities, but some improvements can be made in authorization and front-running protection.

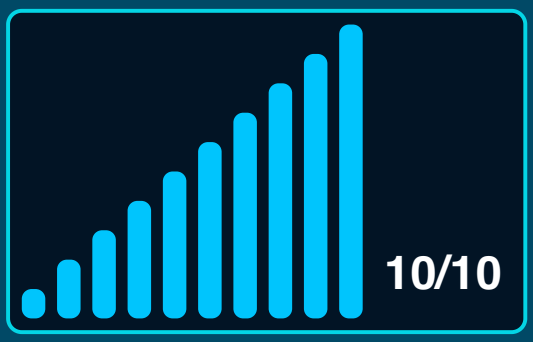
Market Analysis

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