OXSCANS

dappAI

AI Generated at 11:08 PM, UTC

February 23, 2024

OVERVIEW

This audit has been perpared for 'dappAI' 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

dappAI

Name **dappAI**

Info

General Information

Tokenomics

Contract Address 0xbf72ee725f9b06dc564324774801acebad061946

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	468	v0.8.20

Smart Contract Stats

Functions	Events	Constructor
52	10	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	**0** threats found
Medium	**0** threats found
Low	**0** threats found
Informational	**0** threats found

Detail Analysis

Vulnerability Check

21 Passed

0 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- Outdated Compiler Version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

21 Passed

0 Fail

CATEGORY	STATUS	NOTES
Arbitrary Jump/Storage Write		This category is mostly relevant for low-level assembly code, which is not present in this contract.
Centralization of Control		No risk of centralization as the contract owner is a dead address, eliminating the risk of a single point of control.
Compiler Issues		The contract is compiled with a specific and recent compiler version (v0.8.20), reducing compiler-related risks.
Delegate Call to Untrusted Contract		There are no delegate calls to external contracts, so this risk is not applicable.
Dependence on Predictable Variables		The contract does not rely on variables like block.timestamp or block.number in a security-critical way.

Detail Analysis

Detail Analysis

21 Passed

0 Fail

CATEGORY	STATUS	NOTES
Ether/Token Theft		Standard ERC20 implementation without functions that transfer Ether or tokens without proper authorization.
Flash Loans		This contract does not interact with flash loans or lending pools, so this risk is not applicable.
Front Running		No evident vulnerabilities to front-running due to the nature of the implemented functions and their standard behavior.
Improper Events		All external and state-changing functions emit appropriate events.
Improper Authorization Scheme		The contract uses a standard authorization scheme with roles for different functionalities.
Integer Over/Underflow		Uses Solidity 0.8.x which has built-in overflow/underflow protection.

Detail Analysis

Detail Analysis

21 Passed

0 Fail

CATEGORY	STATUS	NOTES
Logical Issues		No logical issues detected in the contract's implementation of the ERC20 standard and additional functionalities.
Oracle Issues		The contract does not use any external oracles, so this category is not applicable.
Outdated Compiler Version		Compiler version v0.8.20 is used, which is recent and considered secure.
Race Conditions		There are no functions or patterns in the contract that could lead to race conditions.
Reentrancy		No reentrancy vulnerabilities detected as state changes occur before external calls.
Signature Issues		The contract does not use Ethereum signatures (ECDSA), so this risk is not applicable.

Detail Analysis

Detail Analysis

21 Passed

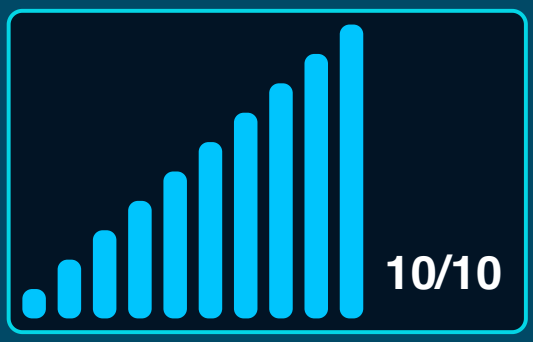
0 Fail

CATEGORY	STATUS	NOTES
Sybil Attack		The concept of a Sybil attack does not directly apply to the functionality of this contract.
Unbounded Loops		No unbounded loops that could lead to gas limit issues or denial of service.
Unused Code		The contract does not contain significant amounts of unused or redundant code.
Overall Contract Safety		The contract follows standard ERC20 practices with additional minting functionality, posing no critical safety concerns.

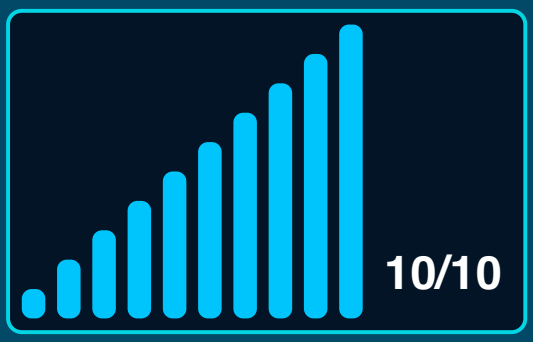
Market Analysis

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