Audit Report
Generated by X Auditor AI

Connect With Us

Website: xauditorai.org

Telegram: t.me/xauditoraichannel

Token Detail

Token Name	Dat Boi
Contract Address	0x7Ca4abDdDa71C61C172b670997130a1e4da32079
Token Symbol	DATBOI
Contract Owner	0x0000000000000000000000000000000000000000
Holders	161
Buy Tax	0%
Sell Tax	0%
is Contract Verified	Verified
is Proxy Contract	No
is Honeypot	No
Anti-Whale Function	Yes
Mintable Function	No
Fake Renounce	No
Hidden Owner	No
Blacklist Function	No
Whitelist Function	Yes
Trading Cooldown Function	Yes
selfDestruct Function	No
Transfer Pauseable	No
Owner Can Change Taxes	No
Owner Can Change Balance	No

Ignore some function return Yes if contract renounced and Fake Renounce and/or Hidden Owner is return No

Automated Audit Report

Solidity assert violation (SWC-110)

Severity: *Low*

A user-provided assertion failed with the message 'Panic(0x11)'.

Integer overflow/underflow (SWC-101)

Severity: *PASSED*

Potential weak source of randomness (SWC-120)

Severity: *Low*

The environment variable "block.number" looks like it might be used as a source of randomness. Note that the values of variables like coinbase, gaslimit, block number and timestamp are predictable and can be manipulated by a malicious miner. Also keep in mind that attackers know hashes of earlier blocks. Don't use any of those environment variables as sources of randomness and be aware that use of these variables introduces a certain level of trust into miners.

Uninitialized Storage Variables (SWC-109)

Severity: *PASSED*

Unprotect Withdraw ETH (SWC-105)

Severity: *PASSED*

Loop Over Unbounded Data Structure (SWC-128)

Severity: *PASSED*

Outdated compiler version (SWC-102)

Severity: *PASSED*

Unused State/Local Variable (SWC-131)

Severity: *PASSED*

Deprecated Global Variables/Function (SWC-111)

Severity: *PASSED*

State Variable Visibility (SWC-108)

Severity: *PASSED*

AI Audit Report

Here are some vulnerabilities and potential solutions for the provided smart contract:

1. **Missing Access Control in `_transfer` Function**:

- The `_transfer` function should include access control to ensure that only authorized users can execute transfers.

2. **Potential Reentrancy Attack in `swapBack` Function**:

- The `swapBack` function calls an external contract (`marketingWallet`) and then updates the transaction timestamp. This can potentially lead to a reentrancy attack.

- Solution: Ensure that external calls are made last in the function to prevent reentrancy attacks. Use the Checks-Effects-Interactions pattern.

3. **Uniswap Functionality**:

- Ensure that the Uniswap functions are properly implemented and secure to prevent attacks or vulnerabilities related to token swapping.

4. **Non-Standard Token Behavior**:

- The token's behavior is non-standard and may not be suitable for all use cases. Consider following ERC20 standards for better compatibility and security.

5. **Insufficient Testing and Audit**:

- The contract may have other vulnerabilities that require further testing and auditing by security experts.

6. **Hardcoded Addresses**:

- The use of hardcoded addresses like `0x7a250d5630B4cF539739dF2C5dAcb4c659F2488D` can be risky as they may change in the future. Consider making these addresses configurable.

Make sure to thoroughly review, test, and potentially refactor the contract to enhance its security and reliability. Additionally, consider conducting a full security audit by professionals to identify and mitigate any potential vulnerabilities.

Disclaimer

This audit report is provided for informational purposes only and is not intended to be used as a basis for financial or investment decisions. The findings, interpretations, and conclusions expressed in this Report are based on the information available at the time of the audit and are subject to change without notice.

While every effort has been made to ensure the accuracy and completeness of the analysis, X Auditor AI does not guarantee the correctness, reliability, or completeness of the Report. The smart contract code is subject to inherent risks, including but not limited to coding errors, vulnerabilities, and unforeseen interactions with other smart contracts or blockchain protocols.

X Auditor AI is not liable for any direct, indirect, incidental, special, or consequential damages arising out of the use of this Report. It is the responsibility of the smart contract owner and users to conduct their own due diligence and assess the risks associated with the smart contract.

This Report does not constitute an endorsement or recommendation of the smart contract or its associated project. X Auditor AI does not assume any responsibility for the use or interpretation of the information contained in this Report.

By using this Report, you acknowledge and agree to the terms and conditions outlined in this disclaimer.