STAYSAFU AUDIT

SECURITY ASSESMENT: FEBRUARY 6th, 2022

CLASSICDOGE

TABLE OF CONTENTS

SUMMARY

*This report has been prepared for **ClassicDoge (XDOGE)** to discover issues and vulnerabilities in the source code of the **ClassicDoge** project as well as any contract dependencies that were not part of an officially recognized library.*

The audit is based on the code of the following BSC smartcontract:

0xb68a34756d8a92ccc821effa03d802846594b64e

*A comprehensive examination has been performed, utilizing Static Analysis, Manual Review, and **ClassicDoge** Deployment techniques. The auditing process pays special attention to the following considerations:*

- Testing the smart contracts against both common and uncommon attack vectors
- Assessing the codebase to ensure compliance with current best practices and industry standards
- Ensuring contract logic meets the specifications and intentions of the client
- Cross referencing contract structure and implementation against similar smart contracts produced by industry leaders
- Thorough line-by-line manual review of the entire codebase by industry experts

OVERVIEW

VULNERABILITY SUMMARY

UNDERSTANDING

The **ClassicDoge** Protocol is a decentralized finance (**DeFi**) token deployed on the Binance smart chain (**BSC**).

ClassicDoge doesn’t employ any fee or auto-burn feature when buying or selling tokens.

PRIVILEGED FUNCTIONS

The contract contains the following privileged functions that are restricted by the **onlyOwner** or **hasMintPermission** modifier.
They are used to modify the contract configurations and addresses attributes. We grouped these functions below :

OWNERSHIP MANAGEMENT

-transferOwnership

-renounceOwnership

PAUSE MANAGEMENT

-pause

-unpause

TRADING MANAGEMENT

-mintAndFreeze

-mint

-finishMinting

OWNERSHIP

Here is a non-exhaustive list of what the smart-contract owner can and cannot do.

Feature	Able to modify / to do	Details
Transaction fees	**No**	
Max transaction	**No**	
Blacklist	**No**	
Whitelist	**No**	
Mint	**Yes**	
Renounce ownership	**Yes**	
Transfer ownership	**Yes**	
Pause/ Unpause	**Yes**	

FINDINGS

Variables of the same name

Severity : Minor

Both **paused** (line 603) and **PAUSED** (line 675) have the same name. The multiplicity of variables with the same name is quite problematic in terms of readability in general. Moreover, the two variables don’t even have the same value, which is a lack of consistency.

Use of block.timestamp for comparison

Severity: Minor

The value of **block.timestamp** can be manipulated by the block’s miner. This is a security problem since **block.timestamp** is used when exchanging token to release frozen tokens. This problem can be avoided by not using **block.timestamp.**

Unlocked compiler version

Severity: Minor

ClassicDoge's contract does not have locked compiler versions, meaning a range of compiler versions can be used. This can lead to differing bytecodes being produced depending on the compiler version, which can create confusion when debugging as bugs may be specific to a specific compiler version(s).

To rectify this, we recommend setting the compiler to a single version, the lowest version tested to be compatible with the code, an example of this change can be seen below.

*0.4.23 is an old version of the compiler, we recommend to use a higher version (at best above 0.8.0)

Before	After
pragma solidity ^0.4.23;	pragma solidity 0.4.23*;

No threshold for minting function

Severity : Medium

Minting function is not subject to a threshold. The owner (in case of a hack would be an attacker) can mint an infinite token amount, leading to a price crash for example. We strongly advice adding a threshold and a cooldown to avoid this problem.

Centralization of major privileges

Severity : Medium

The owner of the smart-contract has major privileges over it (the owner can pause the contract and mint an infinite amount of token). This can be a problem, and we recommend at least to use a multi-sig wallet for the owner address, and at best to establish a community governance protocol to avoid such centralization.

Unused and useless variables/constants

Severity : N/A

some variables or constants are unnecessary or ambiguous, such as **TARGET_USER** or **CONTINUE_MINTING**. We advice you delete them.

Conclusion

No major issue has been found in the **ClassicDoge** smart-contract. The findings we reported are low severity issues, and are common to the majority of rewards smart-contracts. The overall security of the smart-contract is very good, the only points that should be improved is the minting function’s threshold, centralization of privileges and the contract code’s abidance to best practices.

DISCLAIMER

This report is subject to the terms and conditions (including without limitation, description of services, confidentiality, disclaimer and limitation of liability) set forth in the Services Agreement, or the scope of services, and terms and conditions provided to the Company in connection with the Agreement.

This report provided in connection with the Services set forth in the Agreement shall be used by the Company only to the extent permitted under the terms and conditions set forth in the Agreement.

This report may not be transmitted, disclosed, referred to or relied upon by any person for any purposes without **StaySAFU**'s prior written consent. This report is not, nor should be considered, an “endorsement” or “disapproval” of any particular project or team. This report is not, nor should be considered, an indication of the economics or value of any “product” or “asset” created by any team or project that contracts **StaySAFU** to perform a security assessment.

This report does not provide any warranty or guarantee regarding the absolute bug-free nature of the technology analyzed, nor do they provide any indication of the technologies proprietors, business, business model or legal compliance. This report should not be used in any way to make decisions around investment or involvement with any particular project.

This report in no way provides investment advice, nor should be leveraged as investment advice of any sort. This report represents an extensive assessing process intending to help our customers increase the quality of their code while reducing the high level of risk presented by cryptographic tokens and blockchain technology.
Blockchain technology and cryptographic assets present a
high level of ongoing risk.

StaySAFU's position is that each company and individual are responsible for their own due diligence and continuous security. **StaySAFU**'s goal is to help reduce the attack vectors and the high level of variance associated with utilizing new and consistently changing technologies, and in no way claims any guarantee of security or functionality of the technology we agree to analyze.