STAYSAFU AUDIT

July 26TH, 2022

2G Carbon Coin

TABLE OF CONTENTS

AUDIT SUMMARY

This report was written for 2G Carbon Coin ($2GCC) in order to find flaws and vulnerabilities in the 2G Carbon Coin project's source code, as well as any contract dependencies that weren't part of an officially recognized library.

A comprehensive examination has been performed, utilizing Static Analysis, Manual Review, and 2G Carbon Coin 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
- Through line-by-line manual review of the entire codebase by industry expert

AUDIT OVERVIEW

PROJECT SUMMARY

Project name	2G Carbon Coin
Description	Carbon Coin brings carbon on-chain, unlocking the speed and scale of crypto markets to protect the earth's natural carbon sinks and finance planet-saving projects. [2G Carbon Coin: Team]
Platform	Binance Smartchain
Language	Solidity
Codebase	https://bscscan.com/token/0x1a515bf4e35aa2df67109281de6b3b00ec37675e

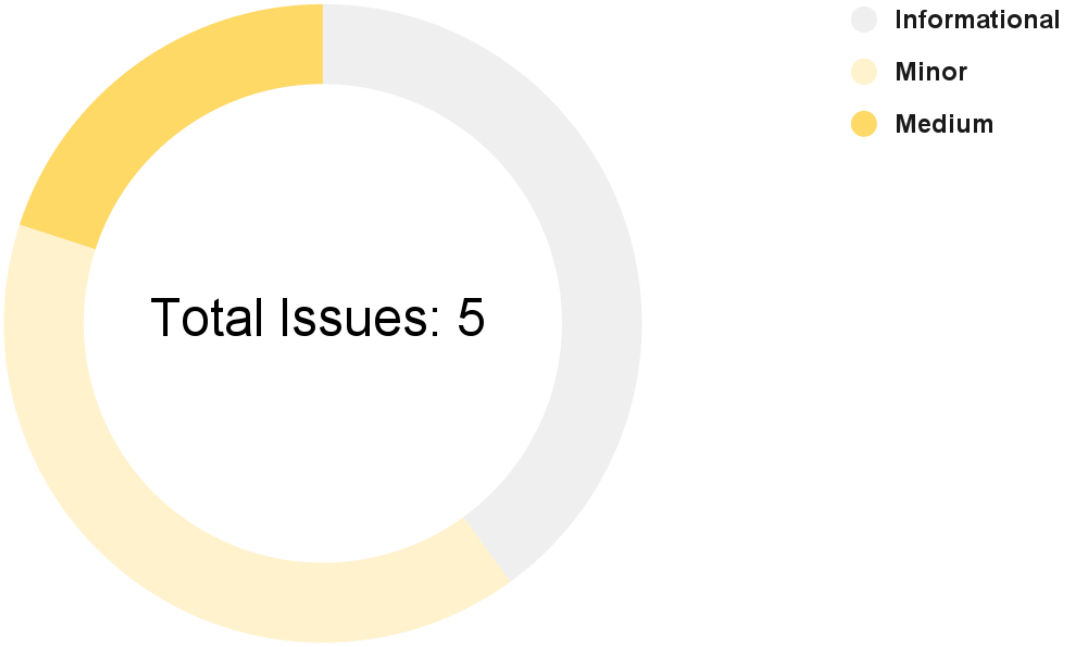
FINDINGS SUMMARY

Vulnerability	Total
Critical	0
Major	0
Medium	1
Minor	2
Informational	2

EXECUTIVE SUMMARY

There have been no major or critical issues related to the codebase and all findings listed here range from informational to medium. The medium security issues are the dependence on a decentralized exchange platform, centralization of privileges, and missing threshold checks.

AUDIT FINDINGS



Code	Title	Severity
EXT-1	External protocol dependencies	Medium
GAS-1	Unoptimized function type	Minor
COMP-1	Unfixed version of compiler	Minor
FUNC-1	Unused functions	Informational
GAS-2	Overly long error messages	Informational

EXT-1 | Dependence to an external protocol

Description

The contract serves as an underlying entity to interact with third party PancakeSwap protocols. The scope of the audit would treat this third party entity as black box and assume it is fully functional. However in the real world, third parties may be compromised and may have led to assets lost or stolen.

Recommendation

We encourage the team to constantly monitor the security level of the entire PancakeSwap project, as the security of the token is highly dependent on the security of the decentralized exchange platform.

GAS-1 | Unoptimized function type

Description

Throughout 2G_Carbon_Coin's contracts some functions are of type public although they are never called within the contract. External functions require significantly less gas to call. Such found functions are listed below:

- renounceOwnership -> Line 175
- transferOwnership -> Line 183
- name -> Line 491
- symbol -> Line 499
- decimals -> Line 516
- totalSupply -> Line 523
- balanceOf -> Line 530
- transfer -> Line 548
- allowance -> Line 561
- approve -> Line 578
- transferFrom -> Line 601
- increaseAllowance -> Line 630
- decreaseAllowance -> Line 657

Recommendation

We recommend reviewing each of the functions listed above and where possible switch their type from public to external.

COMP-1 | Unfixed version of compiler

Description

2G Carbon Coin token'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).

Recommendation

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

```
pragma solidity 0.8.9;
```

FUNC-1 | Unused functions

Description

Multiple functions within 2G Carbon Coin's contract are defined as private or internal but are never called within the contract. This wastes contract space as there is a maximum size a contract can have. Functions found with this issue have been listed below:

- _msgData -> Line 115
- div -> Line 334
- div -> Line 390
- mod -> Line 350
- mod -> Line 416
- mul -> Line 320
- sub -> Line 306
- tryAdd -> Line 221
- tryDiv -> Line 263
- tryMod -> Line 275
- tryMul -> Line 246
- trySub -> Line 234
- _burn -> Line 735
- _setupDecimals -> Line 780

Recommendation

We recommend removing these functions from the contract.

GAS-2 | Overly long error messages

Description

The smart contract has some error messages that are too long. The industry standards specify error messages must have a maximum length of 32 bytes. We recommend having the short error messages within 32 bytes to optimize gas costs. Require statements with this issue have been listed below:

- line -> 164
- line -> 184
- line -> 692
- line -> 693
- line -> 736
- line -> 766
- line -> 767

Recommendation

We recommend shortening these error messages to be 32 characters or less in length.

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 fun.