macro

# Seven Seas A-5

Security Audit

April 22, 2024
Version 1.0.0

Presented by 0xMacro

## Table of Contents

## Introduction

This document includes the results of the security audit for Seven Seas's smart contract code as found in the section titled ‘Source Code’. The security audit was performed by the Macro security team on April 19th 2024.

The purpose of this audit is to review the source code of certain Seven Seas Solidity contracts, and provide feedback on the design, architecture, and quality of the source code with an emphasis on validating the correctness and security of the software in its entirety.

**Disclaimer:** While Macro’s review is comprehensive and has surfaced some changes that should be made to the source code, this audit should not solely be relied upon for security, as no single audit is guaranteed to catch all possible bugs.

## Overall Assessment

The following is an aggregation of issues found by the Macro Audit team:

| Severity | Count | Acknowledged | Won't Do | Addressed |
|---|---|---|---|---|
| Informational | 1 | 1 | - | - |

Seven Seas was quick to respond to these issues.

## Specification

Our understanding of the specification was based on the following sources:

- Discussions with the Seven Seas team.
- Available documentation in the repository.

## Trust Model, Assumptions, and Accepted Risks (TMAAR)

Trust Assumptions:

In addition to the trust assumptions mentioned in SevenSeas-4, this update allows interacting with Into the Block contracts, and holding positions with them. Each is intended to be setup with the vault as the owner, but will likely set Executors on deployment of each of Into the Block's position managers. These executors are bots managed by Into the Block, and have the ability to manage positions that have been setup by the vault, but only the vaults can set approvals and withdraw assets. There is a trust assumption that these set executors will manage the vault's assets as intended. As mentioned in I-1, there is also a trust assumption that contracts deployed are the same as those provided for this review, as the contracts are unverified and private.

## Source Code

The following source code was reviewed during the audit:

- **Repository:** boring-vault
- **Commit Hash:** `db54a948fbce3bc7a59574407ff36fb376117f4d`

Specifically, we audited the following contracts within this repository.

| Contract | SHA256 |
| --- | --- |
| src/base/DecodersAndSanitizers/Protocols/ITB/ITBPositionDecoderAndSanitizer.sol | `cca2554a697a30179ef16e87d768154c23627fd11bcdb6f7f5ef91eccea483d7` |
| src/base/DecodersAndSanitizers/Protocols/ITB/aave/AaveDecoderAndSanitizer.sol | `62f8744a6f1603eded0c66bec5a4a61f1e12a65ef184d3542ac0fc200bf42ed4` |
| src/base/DecodersAndSanitizers/Protocols/ITB/common/BoringDecoderAndSanitizer.sol | `8abc0e9ed3ff91ef2a7bcb3644ebc81f398617aabd5d2cef9c6f309ba7770b36` |
| src/base/DecodersAndSanitizers/Protocols/ITB/common/ITBContractDecoderAndSanitizer.sol | `60b43c345e1c45e3f64aae1dd05575d1644c98689255d271f96119b7dd78fe1a` |
| src/base/DecodersAndSanitizers/Protocols/ITB/common/Ownable2StepDecoderAndSanitizer.sol | `e2ac84785f681b6b0707293ea85e4a25a66304cbb43ac6790ca73a64c4056725` |
| src/base/DecodersAndSanitizers/Protocols/ITB/common/WithdrawableDecoderAndSanitizer.sol | `fd0dd86da91e6431b662541f68e97be7c0c5643f026153d21ed5ed0f229c445d` |
| src/base/DecodersAndSanitizers/Protocols/ITB/curve_and_convex/ConvexDecoderAndSanitizer.sol | `293163a971f30b8abb7c1f5108901c09719420e59f340a684b16c731d5b2846b` |

| Contract | SHA256 |
| --- | --- |
| src/base/DecodersAndSanitizers/Protocols/ITB/curve_and_convex/CurveAndConvexDecoderAndSanitizer.sol | `0c455a09c93c388c093f5ca8eac53f3887dfc16169fb3e76b87b58eea90f696f` |
| src/base/DecodersAndSanitizers/Protocols/ITB/curve_and_convex/CurveNoConfigDecoderAndSanitizer.sol | `2593eb0b799ce09129265e73aae870060a66fa5a1b0687254e3b22e0a8a0ddfa` |
| src/base/DecodersAndSanitizers/Protocols/ITB/gearbox/GearboxDecoderAndSanitizer.sol | `31c2383034cdbd706ab30fca8e64b4a97d236454fbf72d074d84bebb6704f270` |

**Note:** This document contains an audit solely of the Solidity contracts listed above. Specifically, the audit pertains only to the contracts themselves, and does not pertain to any other programs or scripts, including deployment scripts.

## Issue Descriptions and Recommendations

Click on an issue to jump to it, or scroll down to see them all.

## Security Level Reference

We quantify issues in three parts:

1. The high/medium/low/spec-breaking **impact** of the issue:
    - How bad things can get (for a vulnerability)
    - The significance of an improvement (for a code quality issue)
    - The amount of gas saved (for a gas optimization)
2. The high/medium/low **likelihood** of the issue:
    - How likely is the issue to occur (for a vulnerability)
3. The overall critical/high/medium/low **severity** of the issue.

This third part – the severity level – is a summary of how much consideration the client should give to fixing the issue. We assign severity according to the table of guidelines below:

| Severity | Description |
| --- | --- |
| (C-x) Critical | We recommend the client **must** fix the issue, no matter what, because not fixing would mean **significant funds/assets WILL be lost.** |
| (H-x) High | We recommend the client **must** address the issue, no matter what, because not fixing would be very bad, *or* some funds/assets will be lost, *or* the code's behavior is against the provided spec. |
| (M-x) Medium | We recommend the client to **seriously consider** fixing the issue, as the implications of not fixing the issue are severe enough to impact the project significantly, albiet not in an existential manner. |
| (L-x) Low | The risk is small, unlikely, or may not relevant to the project in a meaningful way.<br><br>Whether or not the project wants to develop a fix is up to the goals and needs of the project. |
| (Q-x) Code Quality | The issue identified does not pose any obvious risk, but fixing could improve overall code quality, on-chain composability, developer ergonomics, or even certain aspects of protocol design. |
| (I-x) Informational | Warnings and things to keep in mind when operating the protocol. No immediate action required. |
| (G-x) Gas Optimizations | The presented optimization suggestion would save an amount of gas significant enough, in our opinion, to be worth the development cost of implementing it. |

# Issue Details

## I-1 Into the Block's contracts are unverified and private

| TOPIC | STATUS | IMPACT |
| --- | --- | --- |
| Informational | Acknowledged | Informational ✳ |

As mentioned in the trust assumptions, vaults that hold positions with Into the Blocks contracts. These contracts are private, but were reviewed within this audit, . The contracts that a Boring Vault is setup to interact with are position manager contracts that vary based on the protocol it interacts with. This audit covered sanitizers and a tree that allow for vaults to setup into the block positions for Aave, GearBox, and Convex-Curve, with the vault having ownership of the position manager, having the sole ability to withdraw assets and set approvals. Executor addresses are set in the construction of each manager, and they have the permission to manage the position, ie. stake and unstake assets, in the respective protocols.

RESPONSE BY SEVEN SEAS

> IntoTheBlock has used these contracts in production for years, and is a very reputable team. The contracts were also reviewed by the 7Seas team and no issues were found.

## Disclaimer

Macro makes no warranties, either express, implied, statutory, or otherwise, with respect to the services or deliverables provided in this report, and Macro specifically disclaims all implied warranties of merchantability, fitness for a particular purpose, noninfringement and those arising from a course of dealing, usage or trade with respect thereto, and all such warranties are hereby excluded to the fullest extent permitted by law.

Macro will not be liable for any lost profits, business, contracts, revenue, goodwill, production, anticipated savings, loss of data, or costs of procurement of substitute goods or services or for any claim or demand by any other party. In no event will Macro be liable for consequential, incidental, special, indirect, or exemplary damages arising out of this agreement or any work statement, however caused and (to the fullest extent permitted by law) under any theory of liability (including negligence), even if Macro has been advised of the possibility of such damages.

The scope of this report and review is limited to a review of only the code presented by the Seven Seas team and only the source code Macro notes as being within the scope of Macro's review within this report. This report does not include an audit of the deployment scripts used to deploy the Solidity contracts in the repository corresponding to this audit. Specifically, for the avoidance of doubt, this report does not constitute investment advice, is not intended to be relied upon as investment advice, is not an endorsement of this project or team, and it is not a guarantee as to the absolute security of the project. In this report you may through hypertext or other computer links, gain access to websites operated by persons other than Macro. Such hyperlinks are provided for your reference and convenience only, and are the exclusive responsibility of such websites' owners. You agree that Macro is not responsible for the content or operation of such websites, and that Macro shall have no liability to your or any other person or entity for the use of third party websites. Macro assumes no responsibility for the use of third party software and shall have no liability whatsoever to any person or entity for the accuracy or completeness of any outcome generated by such software.