Beethoven X

Deployment Review

February 4, 2022

Prepared for:

Daniel Mk

Beethoven X

Mr. Kind Human

Beethoven X

Prepared by:

Simone Monica

About Trail of Bits

Founded in 2012 and headquartered in New York, Trail of Bits provides technical security assessment and advisory services to some of the world's most targeted organizations. We combine high-end security research with a real-world attacker mentality to reduce risk and fortify code. With 80+ employees around the globe, we've helped secure critical software elements that support billions of end users, including Kubernetes and the Linux kernel.

We maintain an exhaustive list of publications at https://github.com/trailofbits/publications, with links to papers, presentations, public audit reports, and podcast appearances.

In recent years, Trail of Bits consultants have showcased cutting-edge research through presentations at CanSecWest, HCSS, Devcon, Empire Hacking, GrrCon, LangSec, NorthSec, the O'Reilly Security Conference, PyCon, REcon, Security BSides, and SummerCon.

We specialize in software testing and code review projects, supporting client organizations in the technology, defense, and finance industries, as well as government entities. Notable clients include HashiCorp, Google, Microsoft, Western Digital, and Zoom.

Trail of Bits also operates a center of excellence with regard to blockchain security. Notable projects include audits of Algorand, Bitcoin SV, Chainlink, Compound, Ethereum 2.0, MakerDAO, Matic, Uniswap, Web3, and Zcash.

To keep up to date with our latest news and announcements, please follow @trailofbits on Twitter and explore our public repositories at https://github.com/trailofbits. To engage us directly, visit our "Contact" page at https://www.trailofbits.com/contact, or email us at info@trailofbits.com.

Trail of Bits, Inc.
228 Park Ave S #80688
New York, NY 10003
https://www.trailofbits.com
info@trailofbits.com

Notices and Remarks

Copyright and Distribution

Test Coverage Disclaimer

All activities undertaken by Trail of Bits in association with this project were performed in accordance with a statement of work and mutually agreed upon project plan.

Security assessment projects are time-boxed and often reliant on information that may be provided by a client, its affiliates, or its partners. As such, the findings documented in this report should not be considered a comprehensive list of security issues, flaws, or defects in the target system or codebase.

Table of Contents

Executive Summary

Overview

Beethoven X engaged Trail of Bits to review the deployment of its smart contracts. From January 31 to February 4, 2022, one consultant conducted a review of the client-provided source code, with one person-week of effort. Details of the project's timeline and test targets are provided in subsequent sections of this report.

Project Scope

Our testing efforts were focused on the identification of flaws related to a misconfigured deployment that could result in a compromise of the target system; however, the implementations of the contracts were out of scope. We also sought to confirm that the deployed contracts are a correct Balancer V2 fork.

Summary of Findings

The review did not uncover any deployment or configuration issues that could impact the system. However, we found that the lack of an incident response plan was problematic (see Appendix A). We also raised concerns regarding the dynamic fee pool setting on the UI, which could be a front-running opportunity for a malicious pool owner. Finally, the version of Gnosis Safe in use was not up to date with the latest release at the time of the review.

Project Summary

Contact Information

The following managers were associated with this project:

Dan Guido, Account Manager
dan@trailofbits.com

Sam Greenup, Project Manager
sam.greenup@trailofbits.com

The following engineer was associated with this project:

Simone Monica, Consultant
simone.monica@trailofbits.com

Project Timeline

The significant events and milestones of the project are listed below.

Date	Event
January 28, 2022	Pre-project kickoff call
February 4, 2022	Delivery of report draft
February 4, 2022	Report readout meeting
February 9, 2022	Delivery of final report

Project Targets

The engagement involved a review of the deployed targets listed below.

Beethoven X—Deployment

Repository	https://github.com/beethovenxfi/balancer-v2-monorepo
Branches	`beethovenx-deployments-latest`
	`beethovenx-authorizer-helpers`
Versions	`bbc50d6cb9945f324d86f0fd22ccf91e0dd0e64b`
	`9772f894d1f34619341846d5c074437996201c08`
Type	Solidity
Platform	Ethereum

The following are the deployed contract addresses in the Fantom network:

`vault`	`0x20dd72Ed959b6147912C2e529F0a0C651c33c9ce`
`protocolFeesCollector`	`0xC6920d3a369E7c8BD1A22DbE385e11d1F7aF948F`
`authorizer`	`0x974D3FF709D84Ba44cde3257C0B5B0b14C081Ce9`
`weightedPoolFactory`	`0x60467cb225092cE0c989361934311175f437Cf53`
`weightedPool2TokensFactory`	`0x92b377187bcCC6556FceD2f1e6DAd65850C20630`
`stablePoolFactory`	`0x55df810876354Fc3e249f701Dd78DeDE57991F8D`
`liquidityBootstrappingPoolFactory`	`0x458368B3724B5a1c1057A00b28eB03FEb5b64968`
`metaStablePoolFactory`	`0x70b55Af71B29c5Ca7e67bD1995250364C4bE5554`

Summary of Recommendations

Trail of Bits recommends that Beethoven X take the following actions:

- Develop a detailed incident response plan to ensure that the Beethoven X team can promptly address any issues that arise (Appendix A).
- Develop a monitoring system to track the contracts' behavior (Appendix A).
- For pools with dynamic fees whose owners are not Beethoven's multisignature wallet, provide a warning in the UI regarding the risks of these pools.
- Update Gnosis Safe to the latest version (1.3).

A. Incident Response Recommendations

In this section, we provide recommendations around the formulation of an incident response plan.

- **Identify who (either specific people or roles) is responsible for carrying out the mitigations (deploying smart contracts, pausing contracts, upgrading the front end, etc.).**
    - Specifying these roles will strengthen the incident response plan and ease the execution of mitigating actions when necessary.
- **Document internal processes for situations in which a deployed remediation does not work or introduces a new bug.**
    - Consider adding a fallback scenario that describes an action plan in the event of a failed remediation.
- **Clearly describe the intended process of contract deployment.**
- **Consider whether and under what circumstances Beethoven X will make affected users whole after certain issues occur.**
    - Such issues could include an individual or aggregate loss, a loss resulting from user error, a contract flaw, and a third-party contract flaw.
- **Document how Beethoven X plans keep up to date on new issues, both to inform future development and to secure the deployment toolchain and the external on-chain and off-chain services that the system relies on.**
    - For each language and component, describe the noteworthy sources for vulnerability news. Subscribe to updates for each source. Consider creating a special private Discord channel with a bot that will post the latest vulnerability news; this will help the team keep track of updates all in one place. Also consider assigning specific team members to keep track of the vulnerability news of a specific component of the system.
- **Consider scenarios involving issues that would indirectly affect the system.**
- **Determine when and how the team would reach out to and onboard external parties (auditors, affected users, other protocol developers, etc.).**
    - Some issues may require collaboration with external parties to efficiently remediate them.

- **Define contract behavior that is considered abnormal for off-chain monitoring.**
    - Consider adding more resilient solutions for detection and mitigation, especially in terms of specific alternate endpoints and queries for different data as well as status pages and support contacts for affected services.
- **Combine issues and determine whether new detection and mitigation scenarios are needed.**
- **Perform periodic dry runs of specific scenarios in the incident response plan to find gaps and opportunities for improvement and to develop muscle memory.**
    - Document the intervals at which the team should perform dry runs of the various scenarios. For scenarios that are more likely to happen, perform dry runs more regularly. Create a template to be filled in after a dry run to describe the improvements that need to be made to the incident response plan.