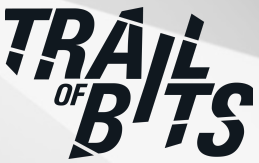

Ethereum Foundation EIP-4788 Smart Contract

Security Assessment (Summary Report)

September 27, 2023

Prepared for:
Tim Beiko
Ethereum Foundation

Prepared by: **Tarun Bansal**

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 100+ 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 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 a result, 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.

Trail of Bits uses automated testing techniques to rapidly test the controls and security properties of software. These techniques augment our manual security review work, but each has its limitations: for example, a tool may not generate a random edge case that violates a property or may not fully complete its analysis during the allotted time. Their use is also limited by the time and resource constraints of a project.

Table of Contents

Project Summary

Contact Information

The following project manager was associated with this project:

Jeff Braswell, Project Manager
jeff.braswell@@trailofbits.com

The following engineering director was associated with this project:

Josselin Feist, Engineering Director, Blockchain
josselin.feist@trailofbits.com

The following consultant was associated with this project:

Tarun Bansal, Consultant
tarun.bansal@trailofbits.com

Project Timeline

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

Date	Event
September 8, 2023	Pre-project kickoff call
September 15, 2023	Delivery of report draft
September 19, 2023	Delivery of the summary report
September 21, 2023	Report readout meeting
September 27, 2023	Delivery of summary report

Executive Summary

Ethereum Foundation engaged Trail of Bits to review the security of the EIP-4788 smart contract (commit e469656 of https://github.com/ethereum/EIPs). The EIP-4788 specification includes the bytecode of a smart contract to store beacon chain block root hashes in a ring buffer. This smart contract has only two functions to get and set a pair of a timestamp and a beacon chain block root hash and functions as an oracle for other smart contracts that need to validate the consensus layer state.

A team of one consultant conducted the review from September 11 to September 15, 2023, for a total of one engineer-week of effort. With full access to source code and documentation, we performed static and dynamic testing of the target bytecode using automated and manual processes.

GOALS AND COVERAGE: The engagement was scoped to provide a security assessment of the EIP-4788 smart contract bytecode. Specifically, we sought to answer the following non-exhaustive list of questions:

- Does the smart contract bytecode meet the functionality described in the EIP-4788 specification?
- Do the set and get functions safely enable storage and querying of timestamps and beacon roots?
- Are all inputs and system parameters properly validated?
- Can the set function write at an unexpected storage slot?
- Can the get function fetch an incorrect storage slot?
- Are the functions reverting only in the expected situations?

We manually reviewed the entire codebase for common EVM bytecode flaws, such as memory manipulation bugs, incorrect opcode arguments order, storage slot manipulation bugs, ABI encoding and decoding issues; and other smart contract flaws, such as access control issues. We tested the codebase for the following edge cases to ensure correct behavior of the code:

- Fetching the block root for the 0 value of the timestamp
- Fetching the block root for the timestamp values that are too far in the past
- Fetching the block root for the timestamp values that are too far in the future

- Setting the block root for the timestamp values covering the whole ring buffer length
- Setting the block root for the timestamp values covering twice the whole ring buffer length

LIMITATIONS. Because of the time-boxed nature of testing work, it is common to encounter coverage limitations. During this project, we did not review all of EIP-4788, such as how the beacon roots are passed from the consensus layer to the execution layer, or client implementations of the EIP-4788, including their interactions with the smart contract, as they were out of scope for this engagement.

REVIEW SUMMARY. The smart contract bytecode is simple, with limited functionality and functions, as specified in the EIP-4788. The bytecode included in the EIP-4788 specification is compiled from the ETK code at https://github.com/lightclient/4788asm. The ETK code includes inline comments to help understand the code and its expected behavior. The audit did not uncover any significant flaws or defects that could impact system confidentiality, integrity, or availability.

RECOMMENDATIONS. The smart contract code works as expected. However, the Ethereum Foundation team could benefit from a reference Solidity implementation of the smart contract and running a differential fuzzing campaign against this reference implementation to capture any divergence from the expected behavior introduced by the low-level manipulations.