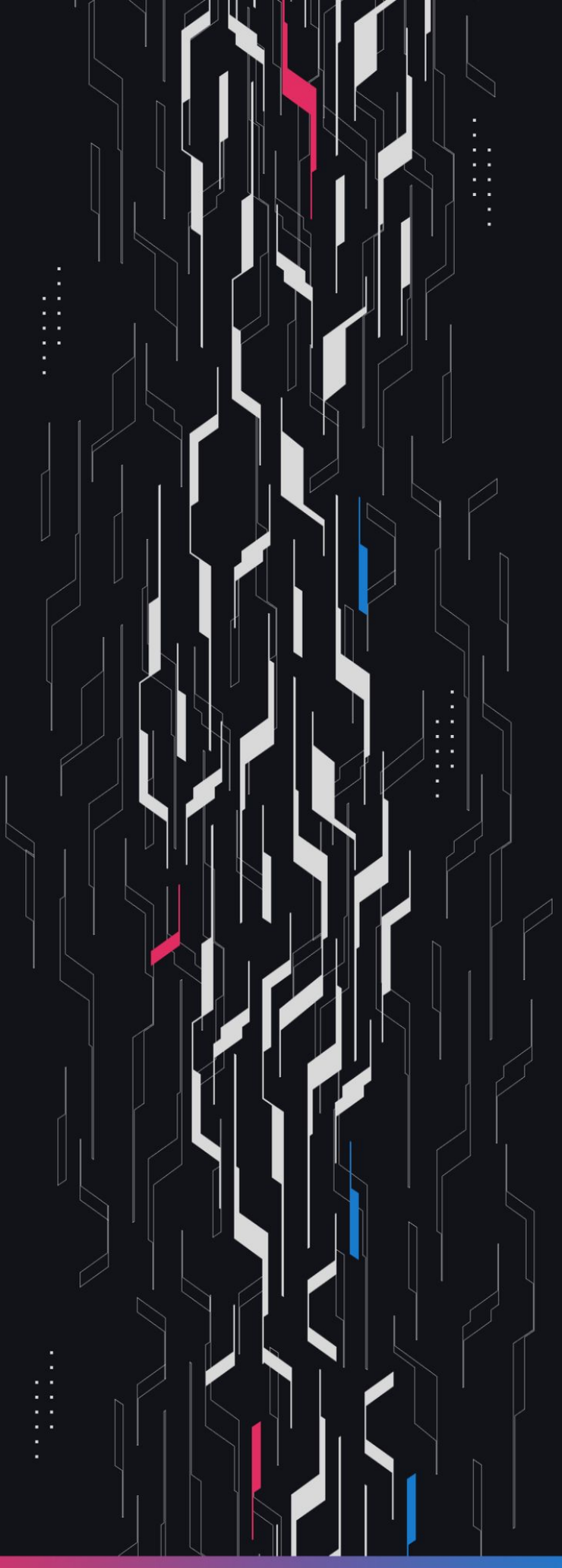
GA GUARDIAN

Baseline

Migrator Updates

Security Assessment

February 25th, 2025



Summary

Audit Firm Guardian

Prepared By Owen Thurm, Daniel Gelfand

Client Firm Baseline

Final Report Date February 25, 2025

Audit Summary

Baseline engaged Guardian to review the security of their Migrator Updates. From the 23rd of November to the 28th of November, a team of 2 auditors reviewed the source code in scope. All findings have been recorded in the following report.

Issues Detected Throughout the engagement 1 High/Critical issues were uncovered and promptly remediated by the Baseline team.

For a detailed understanding of risk severity, source code vulnerability, and potential attack vectors, refer to the complete audit report below.

🔗 Blockchain network: **Blast**

✅ Verify the authenticity of this report on Guardian's GitHub: https://github.com/guardianaudits

📊 Code coverage & PoC test suite: https://github.com/GuardianAudits/Baseline-Perps

Table of Contents

Project Overview

Project Summary

Project Name	Baseline
Language	Solidity
Codebase	https://github.com/0xBaseline/baseline-v2
Commit(s)	Initial commit: d49ca8452c44f630c0527370c24378d64d9e8e3d Final commit: 7fce600e314aa51f99a8e5b85c032a71456ef2f8

Audit Summary

Delivery Date	February 25, 2025
Audit Methodology	Static Analysis, Manual Review, Test Suite, Contract Fuzzing

Vulnerability Summary

Vulnerability Level	Total	Pending	Declined	Acknowledged	Partially Resolved	Resolved
Critical	0	0	0	0	0	0
High	1	0	0	0	0	1
Medium	0	0	0	0	0	0
Low	6	0	0	1	0	5

Audit Scope & Methodology

Vulnerability Classifications

Severity	Impact: *High*	Impact: *Medium*	Impact: *Low*
Likelihood: *High*	🔴 Critical	🟠 High	🟡 Medium
Likelihood: *Medium*	🟠 High	🟡 Medium	🟢 Low
Likelihood: *Low*	🟡 Medium	🟢 Low	🟢 Low

Impact

High Significant loss of assets in the protocol, significant harm to a group of users, or a core functionality of the protocol is disrupted.

Medium A small amount of funds can be lost or ancillary functionality of the protocol is affected. The user or protocol may experience reduced or delayed receipt of intended funds.

Low Can lead to any unexpected behavior with some of the protocol's functionalities that is notable but does not meet the criteria for a higher severity.

Likelihood

High The attack is possible with reasonable assumptions that mimic on-chain conditions, and the cost of the attack is relatively low compared to the amount gained or the disruption to the protocol.

Medium An attack vector that is only possible in uncommon cases or requires a large amount of capital to exercise relative to the amount gained or the disruption to the protocol.

Low Unlikely to ever occur in production.

Audit Scope & Methodology

Methodology

Guardian is the ultimate standard for Smart Contract security. An engagement with Guardian entails the following:

- Two competing teams of Guardian security researchers performing an independent review.
- A dedicated fuzzing engineer to construct a comprehensive stateful fuzzing suite for the project.
- An engagement lead security researcher coordinating the 2 teams, performing their own analysis, relaying findings to the client, and orchestrating the testing/verification efforts.

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. Comprehensive written tests as a part of a code coverage testing suite.
- Contract fuzzing for increased attack resilience.

Findings & Resolutions

ID	Title	Category	Severity	Status
H-01	Sweep DoS	DoS	High	Resolved
L-01	Loans Unfairly Extended By Migration	Gaming	Low	Acknowledged
L-02	Typo	Typo	Low	Resolved
L-03	Unnecessary Keycode	Optimization	Low	Resolved
L-04	Missing Invariant Check	Suggestion	Low	Resolved
L-05	Lacking Event Emission	Events	Low	Resolved
L-06	Inconsistent Discovery Width	Warning	Low	Resolved

H-01 | Sweep DoS

Category	Severity	Location	Status
DoS	High	MarketMaking.sol	Resolved

Description

In function `drop`, the liquidity added to the discovery range is `liqMulWad(threshold, 1e18 + getLiquiditySpread())` while the liquidity added to the anchor range is `liquidityA`.

The `threshold` can be smaller than the anchor's liquidity, ultimately allowing the discovery's liquidity to be thinner than the anchor's liquidity.

Because the invariant `discovery liquidity ≥ anchor liquidity` has been broken, function `sweep` can be DoS'd due to underflow when performing `oldDiscovery.liquidity - liquidityA`, preventing a core market making functionality from being usable.

Recommendation

Minimize `liquidityA` between the threshold and the old anchor liquidity, as done in `sweep` and `slide`.

Resolution

Baseline Team: The issue was resolved in commit 4d8fd4f.

L-01 | Loans Unfairly Extended By Migration

Category	Severity	Location	Status
Gaming	Low	CREDTMigrator.sol	Acknowledged

Description

The `CREDTMigrator` contract allows users to migrate their loans with a discrete expiry to a `LOOPS` position which effectively resets their expiry in that the account will decay at the funding rate.

Therefore a user may game this by migrating their `CREDT` position right before their discrete expiry is hit and enjoy the entire period that their `LOOPS` position loan exists.

Recommendation

Be aware of this potential gaming, if it is undesired consider adding a penalty for users who have already lived out a large portion of the `CREDT` loans.

Resolution

Baseline Team: Acknowledged.

L-02 | Typo

Category	Severity	Location	Status
Typo	Low	CREDTMigrator.sol: 13	Resolved

Description

The `CRETMigrator` contract has a typo where the `CREDT` is missing the D.

Recommendation

Consider renaming the `CRETMigrator` contract to `CREDTMigrator`.

Resolution

Baseline Team: The issue was resolved in commit 4d8fd4f.

L-03 | Unnecessary Keycode

Category	Severity	Location	Status
Optimization	Low	CREDTMigrator.sol: 48	Resolved

Description

In the `requestPermissions` function the `BPOOL_KEYCODE` is declared and unused.

Recommendation

Remove the `BPOOL_KEYCODE` variable.

Resolution

Baseline Team: The issue was resolved in commit 4d8fd4f.

L-04 | Missing Invariant Check

Category	Severity	Location	Status
Suggestion	Low	CREDTMigrator.sol: 57	Resolved

Description

In the `migrate` function currently there is no capacity invariant check at the end of the migration to prevent a breaking of the capacity invariant.

Currently there is no identified invalidation of this invariant due to the migration, however out of an abundance of caution it may be best to add this validation at the end of the `migrate` function.

Recommendation

Consider adding the capacity validation at the end of the `migrate` function.

Resolution

Baseline Team: The issue was resolved in commit 4d8fd4f.

L-05 | Lacking Event Emission

Category	Severity	Location	Status
Events	Low	CREDTMigrator.sol: 57	Resolved

Description

The `migrate` function performs several operations of repaying and opening a new LOOPS position, however no events are emitted for this operation.

Recommendation

Consider emitting a migration event in the `migrate` function.

Resolution

Baseline Team: The issue was resolved in commit 4d8fd4f.

L-06 | Inconsistent Discovery Width

Category	Severity	Location	Status
Warning	Low	Global	Resolved

Description

When launching the protocol within `BaselineInit.launch` the `DISCOVERY_WIDTH` is 350, but within the `MarketMaking` contract the `DISCOVERY_WIDTH` is 100.

This will cause a stepwise decrease in the discovery range after a `MarketMaking` operation, which may be unexpected from the protocol's perspective if the range's size is expected to remain constant.

Recommendation

Document this behavior or keep the widths consistent.

Resolution

Baseline Team: The issue was resolved in commit e1bc337.

Disclaimer

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 Guardian 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. Guardian’s position is that each company and individual are responsible for their own due diligence and continuous security. Guardian’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.

The assessment services provided by Guardian is subject to dependencies and under continuing development. You agree that your access and/or use, including but not limited to any services, reports, and materials, will be at your sole risk on an as-is, where-is, and as-available basis. Cryptographic tokens are emergent technologies and carry with them high levels of technical risk and uncertainty. The assessment reports could include false positives, false negatives, and other unpredictable results. The services may access, and depend upon, multiple layers of third-parties.

Notice that smart contracts deployed on the blockchain are not resistant from internal/external exploit. Notice that active smart contract owner privileges constitute an elevated impact to any smart contract’s safety and security. Therefore, Guardian does not guarantee the explicit security of the audited smart contract, regardless of the verdict.

About Guardian Audits

Founded in 2022 by DeFi experts, Guardian Audits is a leading audit firm in the DeFi smart contract space. With every audit report, Guardian Audits upholds best-in-class security while achieving our mission to relentlessly secure DeFi.

To learn more, visit https://guardianaudits.com

To view our audit portfolio, visit https://github.com/guardianaudits

To book an audit, message https://t.me/guardianaudits