Exactly

Smart Contract Audit

coinspect

Exactly

Exactly RewardsController

March 2023

Smart Contract Audit

V230404

Prepared for Exactly • April 2023

1. Executive Summary

In **March 2023, Exactly** engaged Coinspect to perform a source code review of **Exactly RewardsController**. The objective of the project was to evaluate recently introduced changes and fixes to the `RewardsController` contract.

This report joins a timeline of correlative security audits along with issue IDs, highlighting multiple distinct bugs within the same scope. However, no issues were identified during the assessment:

High Risk	Medium Risk	Low Risk
Open **0**	Open **0**	Open **0**
Fixed 0	Fixed 0	Fixed 0
Reported 0	Reported 0	Reported 0

2. Assessment and Scope

The audit started on **March 30 2023** and was conducted on the **origin/coverage/protocol and origin/fix/missing-rewards-calls** branches of the git repository at `github.com/exactly/protocol` as of commits of the `RewardsController.sol` file:

- `78471e69c04308b13dec1b1a40a20ca409424004`
- `055816d7b567122081ce3d5a6be37ad4f6a06c50`

As for commit `78471e69c04308b13dec1b1a40a20ca409424004`, the main change introduced in `RewardsController.sol` is how the distribution factor is calculated inside the core `PreviewAllocation` function now considering the difference between the start and end times of the distribution. Also, several variables were refactored in memory structs.

Commit `055816d7b567122081ce3d5a6be37ad4f6a06c50` modifies the calculation of the timespan used when calculating the undistributed amount in `RewardsController.PreviewAllocation` if the update is performed after the end of the distribution. It now uses the `deltaTime` input parameter instead of using the current timestamp and the last update.

3. Disclaimer

The information presented in this document is provided "as is" and without warranty. The present security audit does not cover any off-chain systems or frontends that communicate with the contracts, nor the general operational security of the organization that developed the code.