Ambisafe

Nereus NXUSD Staking One Tier Contracts Code Audit and Verification by Ambisafe Inc.

November, 2022

Oleksii Matiiasevych

1. **Introduction.** Nereus Finance. requested Ambisafe to perform a code audit of the Liquidator and related contracts. The contracts in question can be identified by the following git commit hash:

   ```
   9098e68abe7235174c53b3cdc86e3d37493295f9
   ```

   The scope of the audit is NXUSDV2, ProtocolAddressesProvider, TokenOracle, NXUSDStakingOneTier and NXUSDStakingOneTierCalculation contracts.

   During the initial code audit, Nereus Finance team applied a number of updates which can be identified by the following git commit hash:

   ```
   ba008bab100412575ae0131306f4553641ae867f
   ```

   Additional verification was performed after that.

2. **Disclaimer.** The code audit makes no statements or warranties about utility of the code, safety of the code, suitability of the business model, regulatory regime for the business model, or any other statements about fitness of the contracts for any specific purpose, or their bugfree status. The code audit documentation below is for internal management discussion purposes only and should not be used or relied upon by external parties without the express written consent of Ambisafe.

3. **Executive Summary.** There are **no** known compiler bugs for the specified compiler version (0.8.10), that might affect the contracts' logic. There were 0 critical, 0 major, 0 minor, 5 informational and optimizational issues identified in the initial version of the contracts. All findings were resolved and presented below for historical purposes.

4. **Critical Bugs and Vulnerabilities.** No critical bugs or vulnerabilities were found.

5. Initial Line By Line Review. Fixed Findings.

5.1. NXUSDStakingOneTier, line 176. Note, the **unstake()** function if used for full exit will unnecessary clear the **historyUserRewards[msg.sender]** effectively wiping the information on how much the user earned in the past.

5.2. NXUSDStakingOneTier, line 218. Note, the **_calculateRewards()** function uses a leftover notation of **Tier2** while there is only one tier now.

5.3. NXUSDStakingOneTier, line 295. Optimization, the **logarithm()** function is not used.

5.4. NXUSDStakingOneTierCalculation, line 18. Optimization, the **ONE_HUNDRED_PERCENT** constant is not used.

5.5. NXUSDStakingOneTierCalculation, line 76. Optimization, the **logarithm()** function is not used.

Oleksii Matiiasevych