our shielding · Your smart contracts, our shielding · Your smart c

shieldify

Berally Staking

SECURITY REVIEW

Date: 6 February 2025

CONTENTS

1. About Shieldify

Positioned as the first hybrid Web3 Security company, Shieldify shakes things up with a unique subscription-based auditing model that entitles the customer to unlimited audits within its duration, as well as top-notch service quality thanks to a disruptive 6-layered security approach. The company works with very well-established researchers in the space and has secured multiple millions in TVL across protocols, also can audit codebases written in Solidity, Rust, Go, Vyper, Move and Cairo.

Learn more about us at shieldify.org.

2. Disclaimer

This security review does not guarantee bulletproof protection against a hack or exploit. Smart contracts are a novel technological feat with many known and unknown risks. The protocol, which this report is intended for, indemnifies Shieldify Security against any responsibility for any misbehavior, bugs, or exploits affecting the audited code during any part of the project's life cycle. It is also pivotal to acknowledge that modifications made to the audited code, including fixes for the issues described in this report, may introduce new problems and necessitate additional auditing.

3. About Berally - Staking

Pass is a social token on Berally that grants users access to private group chats and crowdfunding vaults. It functions similarly to Friendtech's key but integrates a unique Proof-of-Liquidity (POL) mechanism powered by Berachain's design. This means that every BERA spent on purchasing a Pass is automatically staked into the Berachain reward vault, earning POL rewards in BGT tokens.

4. Risk Classification

Severity	Impact: High	Impact: Medium	Impact: Low
Likelihood: High	Critical	High	Medium
Likelihood: Medium	High	Medium	Low
Likelihood: Low	Medium	Low	Low

4.1 Impact

- **High** - results in a significant risk for the protocol's overall well-being. Affects all or most users
- **Medium** - results in a non-critical risk for the protocol affects all or only a subset of users, but is still unacceptable
- **Low** - losses will be limited but bearable - and covers vectors similar to griefing attacks that can be easily repaired

4.2 Likelihood

- **High** - almost certain to happen and highly lucrative for execution by malicious actors
- **Medium** - still relatively likely, although only conditionally possible
- **Low** - requires a unique set of circumstances and poses non-lucrative cost-of-execution to rewards ratio for the actor

5. Security Review Summary

The security review lasted 2 days with a total of 64 hours dedicated by 4 researchers from the Shieldify team.

Overall, the code is well-written. The audit report identified one low-severity vulnerability and a single informational recommendation. They primarily stem from an incorrect assumption and a suggestion for a documentation improvement.

The Berally team has done a great job with the development and has been highly responsive to the Shieldify research team's inquiries and promptly implemented all recommendations.

5.1 Protocol Summary

Project Name	**Berally - Staking**
Repository	smartcontract-staking
Type of Project	DeFi, Staking
Audit Timeline	3 days
Review Commit Hash	73ec1b6ee0b2d9d8fc645529cf3ce10d00792422
Fixes Review Commit Hash	4a82bfa5978e5b1f6bc266d64b787d7874cf03e5

5.2 Scope

The following smart contracts were in the scope of the security review:

File	nSLOC
Staking.sol	323
IStaking.sol	41
Total	**364**

6. Findings Summary

The following number of issues have been identified, sorted by their severity:

- **Low** issues: **1**
- **Informational** issues: **1**

ID	Title	Severity	Status
[L-01]	Incorrect Assumption That USD Tokens Exist In a 1 to 1 Ratio	Low	Fixed
[I-01]	Documentation Discusses Distribution and Claims of Honey Rewards Rather Than USD Tokens	Info	Fixed

7. Findings

[L-01] Incorrect Assumption That USD Tokens Exist In a 1 to 1 Ratio

Severity

Low Risk

Description

The `migrateUsdToken` incorrectly assumes that all USD-pegged tokens have the same price. The function gets the `usdToken` 's balance and transfers the same balance of the `newUsdToken` to the contract.

Location of Affected Code

File: Staking.sol

```
function migrateUsdToken(address newUsdToken) external onlyOwner {
    require(
        usdToken != newUsdToken,
        "new USD token can't be the same as old one."
    );
@>      uint256 usdBalance = IERC20(usdToken).balanceOf(address(this));
    SafeERC20.safeTransferFrom(
        IERC20(newUsdToken),
        msg.sender,
        address(this),
@>          usdBalance
    );
@>      SafeERC20.safeTransfer(IERC20(usdToken), msg.sender, usdBalance);
    usdToken = newUsdToken;
}
```

Impact

Since different usd pegged tokens (or other tokens) have different prices, situations can arise in which the `newUsdToken` is worth comparatively less (or more) than the `usdToken`. As a result, the new rewards being rewards being migrated may be worth less (or more) to the users. This can also occur if a non-usd token is set as `newUsdToken`.

Recommendation

A potential solution to this involves incorporating an oracle with which price can be compared, and the amount of `newUsdToken` to transfer in will be calculated.

Team Response

Fixed.

[I-01] Documentation Discusses Distribution and Claims of Honey Rewards Rather Than USD Tokens

Severity

Informational Risk

Description

The documentation discusses the distribution and claims of Honey rewards. The contract deals with USD tokens instead.

Location of Affected Code

File: Staking.sol

```
> ### 4. Distribute rewards
> The entry point \codex{distributeRewards()} in the \codex{Staking}
  contract allocates a certain amount of HONEY as a reward for staking.
  The entire reward pool is distributed among all users currently
  staking.

> ### 5. Claim rewards
> The entry point \codex{claimRewards()} in the \codex{Staking} contract
  allows users to withdraw their allocated HONEY rewards, which have
  been distributed based on the staking duration and amount
```

Recommendation

Update docs to reflect USD token instead.

Team Response

Fixed.

, our shielding · Your smart contracts, our shielding · Your smart c

shieldify

Thank you!