SHERLOCK

Security Review For Lazy Bear

Collaborative Audit Prepared For:	**Lazy Bear**
Lead Security Expert(s):	**KupiaSec** **oot2k**
Date Audited:	**April 29 - May 1, 2025**
Final Commit:	**32c2b0d**

Introduction

TBA

Scope

Repository: cdivot/lazy-bear-core

Audited Commit: e0996aec59049e0b513af090f6aac40e2637a11d

Final Commit: 32c2b0d61a7fe5a346fbcfd07c08753e48a58dc6

Files:

- contracts/LazyBearRiver.sol

Final Commit Hash

32c2b0d61a7fe5a346fbcfd07c08753e48a58dc6

Findings

Each issue has an assigned severity:

- Medium issues are security vulnerabilities that may not be directly exploitable or may require certain conditions in order to be exploited. All major issues should be addressed.
- High issues are directly exploitable security vulnerabilities that need to be fixed.
- Low/Info issues are non-exploitable, informational findings that do not pose a security risk or impact the system's integrity. These issues are typically cosmetic or related to compliance requirements, and are not considered a priority for remediation.

Issues Found

High	Medium	Low/Info
0	0	2

Issues Not Fixed and Not Acknowledged

High	Medium	Low/Info
0	0	0

Issue L-1: In the _updateEcosystem, the elapsed epochs should be calculated using getEpochFromTimestamp.

Source: https://github.com/sherlock-audit/2025-04-lazy-bear/issues/2

Summary

This discrepancy arises because ecosystem updates are based on epochs, while the `lastUpdateTime` relies on the current time. The truncated time is ignored. By exploiting this truncated time, malicious users can get more fish from the protocol.

Vulnerability Detail

If the `updateEcosystem` function is called every 11.9 hours and every 6 hours, the ecosystem updates remain the same. However, in the first case, users receive nearly double the rewards compared to the second case with the same ecosystem.

Impact

Users could get more fishes than normal.

Tool Used

Manual Review

Recommendation

https://github.com/sherlock-audit/2025-04-lazy-bear/blob/main/lazy-bear-core/contracts/LazyBearRiver.sol#L233

```
    function _updateEcosystem() internal returns (bool extinction) {
-       uint256 timeElapsed = block.timestamp - lastUpdateTime;
        // Calculate number of epochs that have passed
-      uint256 epochs = timeElapsed / EPOCH_LENGTH;
+     uint256 epochs = getEpochFromTimestamp(block.timestamp) -
 ↪  getEpochFromTimestamp(lastUpdateTime);
        if (epochs == 0) return false;
        ...
        lastUpdateTime = block.timestamp;
        ...
    }
```

Issue L-2: In the _updateEcosystem, the extinction Times should be calculated rather than using the current time.

Source: https://github.com/sherlock-audit/2025-04-lazy-bear/issues/3

This issue has been acknowledged by the team but won't be fixed at this time.

Summary

Even if the pool reaches the extinction state, if the function is triggered one epoch or one day later, users will receive additional rewards corresponding to that duration.

Vulnerability Detail

Transactions may be delayed due to congestion on the blockchain. In such cases, users would receive extra rewards.

Impact

Users may receive more rewards than they should.

Tool Used

Manual Review

Recommendation

https://github.com/sherlock-audit/2025-04-lazy-bear/blob/main/lazy-bear-core/contracts/LazyBearRiver.sol#L257 It is advisable to calculate and assign the extinction time instead of using the current time.

Disclaimers

Sherlock does not provide guarantees nor warranties relating to the security of the project.

Usage of all smart contract software is at the respective users’ sole risk and is the users’ responsibility.