1inch farming audit

22 Feb 2022, Igor Gulamov

Introduction

Igor Gulamov conducted the audit of 1inch farming smart contracts.

This review was performed by an independent reviewer under fixed rate.

Scope

Solidity contracts from 1inch/farming.

Issues

We found no critical or major issues.

We consider commit 2e57d43e9667d6a7599baa02e04fd4b2d6daa9b8 as a safe version from the informational security point of view.

Warnings

https://github.com/1inch/farming/blob/d191cc14b526e2eab09c266580b84116b2630da8/contracts/accounting/FarmAccounting.sol#L7

`Info` is 256bit sized. It is more efficient to implement it as `uint256` or `bytes32` or pass it in memory in view functions and reduce the number of `SLOAD` opcodes.

Fixed in 2e57d43e9667d6a7599baa02e04fd4b2d6daa9b8.

https://github.com/1inch/farming/blob/d191cc14b526e2eab09c266580b84116b2630da8/contracts/accounting/UserAccounting.sol#L29

https://github.com/1inch/farming/blob/d191cc14b526e2eab09c266580b84116b2630da8/contracts/accounting/UserAccounting.sol#L39

https://github.com/1inch/farming/blob/d191cc14b526e2eab09c266580b84116b2630da8/contracts/accounting/UserAccounting.sol#L49

https://github.com/1inch/farming/blob/d191cc14b526e2eab09c266580b84116b2630da8/contracts/accounting/UserAccounting.sol#L52

`SafeCast` is not auto-implemented in Solidity 0.8. We propose using `SafeCast`.

Will not fix.

https://github.com/1inch/farming/blob/d191cc14b526e2eab09c266580b84116b2630da8/contracts/accounting/UserAccounting.sol#L43

This function could be simplified:

1. logical xor for booleans is the same as not equal operator
2. correction could be computed only once

Fixed in 2e57d43e9667d6a7599baa02e04fd4b2d6daa9b8.

https://github.com/1inch/farming/blob/d191cc14b526e2eab09c266580b84116b2630da8/contracts/ERC20Farmable.sol#L74

The current algorithm is $O(n^2)$ complexity and could be optimized up to $O(n \log n)$, or up to $O(n)$ (in helped offchain sort case).

Will not fix. O(n) or O(n log n) solutions also come with a huge constant overhead. And as we expect most of cases be with 0-5 farms, this overhead will cost more than current implementation.

Comments

https://github.com/1inch/farming/blob/d191cc14b526e2eab09c266580b84116b2630da8/contracts/BaseFarm.sol#L33

We propose moving external calls to the end of the function.

Will not fix.

https://github.com/1inch/farming/blob/d191cc14b526e2eab09c266580b84116b2630da8/contracts/ERC20Farmable.sol#L118

We recommend rewriting the function because `view` function cannot emit events.

Fixed. We switched to the external-call style of logging 1e84bbdea85dd1bb3d6a23aad39f4b423f0016f0.

Severity Terms

Comment

Comment issues are generally subjective in nature, or potentially deal with topics like "best practices" or "readability". Comment issues in general will not indicate an actual problem or bug in code.

The maintainers should use their own judgment as to whether addressing these issues improves the codebase.

Warning

Warning issues are generally objective in nature but do not represent actual bugs or security problems.

These issues should be addressed unless there is a clear reason not to.

Major

Major issues will be things like bugs or security vulnerabilities. These issues may not be directly exploitable, or may require a certain condition to arise in order to be exploited.

Left unaddressed these issues are highly likely to cause problems with the operation of the contract or lead to a situation which allows the system to be exploited in some way.

Critical

Critical issues are directly exploitable bugs or security vulnerabilities.

Left unaddressed these issues are highly likely or guaranteed to cause major problems or potentially a full failure in the operations of the contract.