G_0 Group

Security Review of

hvGNO

May 2022

hvGNO / May 2022

Files in scope

Following solidity files:

https://github.com/hundred-finance/hundred-dao/blob/7df827501282cb7de568cce9f2b05d9a0f0aa344/contracts/gnosisBonds/veGNO.sol

Current status

All issues have been fixed by the developer. There are no known issues in the relevant contracts in https://github.com/hundred-finance/hundred-dao/blob/4c9dc59b93739f25c3a718559d1143f3a7187e40/contracts/gnosisBonds/veGNO.sol

Issues

1. burnedBalances do not accumulate, leading to incorrect vesting schedule

type: incorrect implementation / severity: major

In `redeem` function statement `burnedBalances[_msgSender()] = gnoAmount_;` should be replaced by `burnedBalances[_msgSender()] += gnoAmount_;` otherwise the contract will "forget" the amount of tokens the user has already redeemed, allowing him to redeem the remaining balance faster.

status - fixed

Issue has been fixed and is no longer present in

https://github.com/hundred-finance/hundred-dao/blob/4c9dc59b93739f25c3a718559d1143f3a7187e40/contracts/gnosisBonds/veGNO.sol

Notes

- `(endTime_ - halfTime_)` on `line 65` can be simplified to just `YEAR / 2`
- Since `endTime_` is effectively unused after simplifiaction mentioned above and `YEAR` is always divided by `2`, it might be better to store half a year as a constant
- `unlockStartTime` is only ever used to calculate `halfTime_` so `halfTime_` can be stored directly isntead
- `gno` and `unlockStartTime` can be stored as `immutable` to save gas
- `whenNotPaused` and `whenPaused` on `pause` and `unPause` respectively is redundant, since it’s already on the internal functions that are called
- rounding can lead to small `burnedBalances` losses on transfer, leading to slightly earlier withdrawals