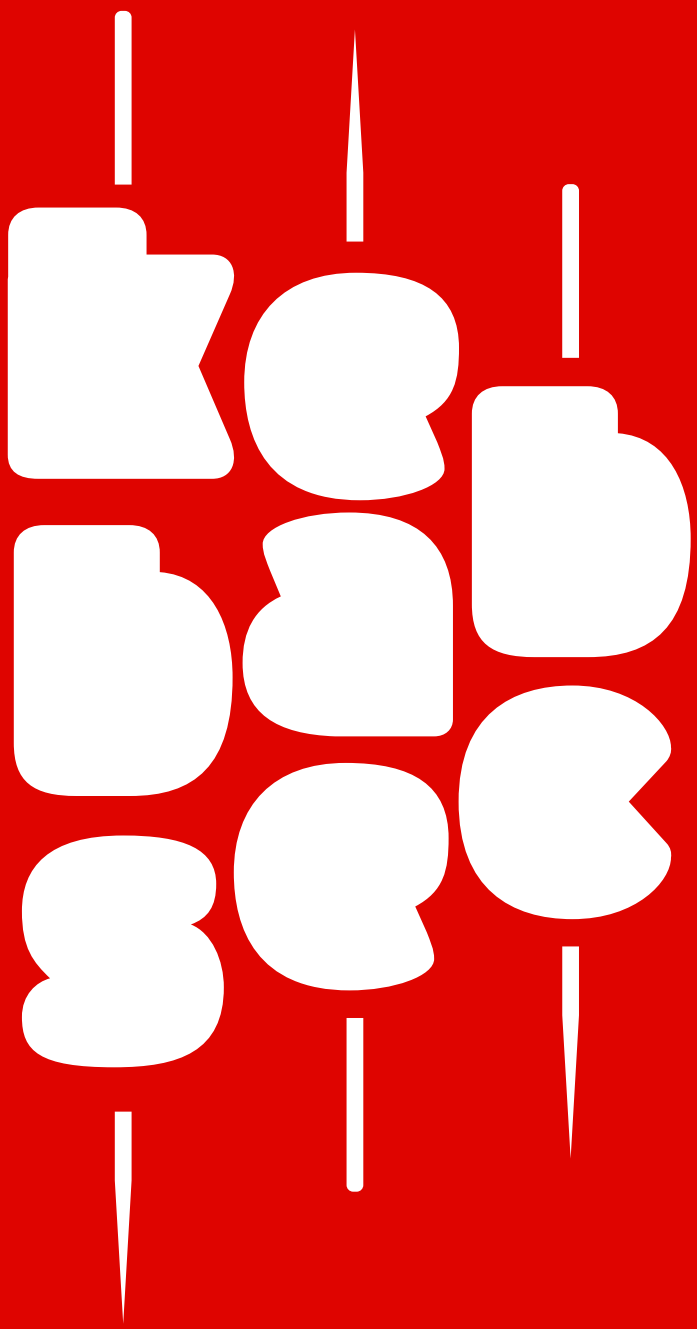

Contents

Introduction & Scope

This audit looks into `MinerEth.sol` with the commit hash 781aa0347092b08224b8b806ff2155ddd98b52db. This audit was conducted by kebabsec members okkothejawa , FlameHorizon and elcid.

Note: This report does not provide any guarantee or warranty of security for the project.

Findings:

1. [LOW] Reverting _mine can lead to potential temporary Denial Of Service
2. [INFORMATIONAL] `ClaimRewards` event emits when no rewards are accured

1. [LOW] Reverting _mine can lead to potential temporary Denial Of Service

Context: MinerETH.sol#L143-L144

Severity: Low likelihood, Medium impact

Description: This contract's functionality requires users to be able to withdraw their principal at all times, but in a low likelihood scenario of Compound III (Comet) having bad debt, making it partially insolvent, `_mine` would assert that the total principal can be withdrawn, or it would revert entirely. In small severity cases in which Comet might lose a small percentage of funds and a big portion of them are still withdrawable, users would not be able to.

Changes made to mitigate issue: Initially, the proposed and implemented solution was to add a parameter `shouldMine` which would allow users to withdraw without mining rewards.

After consideration from Brrito, the change was reverted, since requiring `_mine` to be called prior to withdrawal deterred attackers from taking advantage of Comet's rounding.

A final change was implemented which resolved the DoS vector: `_mine` was updated to conditionally redeposit the contract's ETH balance (if it was slightly less from Comet rounding), or swap interest (if any) for the reward token.

2. [INFORMATIONAL] `ClaimRewards` event emits when no rewards are accured

Context: MinerETH.sol#L235

Description: In `claimRewards`, the event `ClaimRewards` will always emit, even when no rewards are claimed. This is completely safe, but an alternative option would be to move the event into the if `(rewards != 0)` block to save gas for users.

Changes made to mitigate issue: This issue is informational. After deliberation between Brrito and kebabsec, Brrito reached the conclusion that keeping the contract event emissions consistent (i.e. events emitted for every relevant method call), was preferred over the minor gas savings, since it's highly improbable that a tokenholder would have 0 rewards to claim (due to `_mine` being called and rewards accruing every block).