Electisec Euler EVK PR#288 - setLTV() fix Review

Review Resources:

- EVK PR#288

Auditors:

- HHK
- Panda

Table of Contents

Review Summary

Euler PR review

This review covers pull request #288 to Euler's evk (Euler vault kit) codebase that fixes an issue related to the `setLTV()` function.

The PR of the Euler EVK Repo was reviewed over 1 day. Two auditors performed the code review on February 4th, 2025. The review was limited to the latest pull request commit 233afad9155a296016201663b0628ce18cdbd680.

Scope

The scope of the review consisted of the following contracts at the specific commit:

```
src/EVault/modules/Governance.sol
```

After the findings were presented to the Euler team, fixes were made and included in the existing PR or as part of new PRs.

This review is a code review to identify potential vulnerabilities in the code. The reviewers did not investigate security practices or operational security and assumed that privileged accounts could be trusted. The reviewers did not evaluate the security of the code relative to a standard or specification. The review may not have identified all potential attack vectors or areas of vulnerability.

Electisec and the auditors make no warranties regarding the security of the code and do not warrant that the code is free from defects. Electisec and the auditors do not represent nor imply to third parties that the code has been audited nor that the code is free from defects. By deploying or using the code, Euler and users of the contracts agree to use the code at their own risk.

Findings Explanation

Findings are broken down into sections by their respective impact:

- Critical, High, Medium, Low impact

  - These are findings that range from attacks that may cause loss of funds, impact control/ownership of the contracts, or cause any unintended consequences/actions that are outside the scope of the requirements.
- Gas savings
  - Findings that can improve the gas efficiency of the contracts.
- Informational
  - Findings including recommendations and best practices.

Critical Findings

None.

High Findings

None.

Medium Findings

None.

Low Findings

None.

Gas Savings Findings

None.

Informational Findings

1. Informational - Add a `bool getQuote` parameter to the function

Technical Details

The PR#288 removes the `getQuote()` call from the `setLTV()` function.

This is because it can cause issues with pull-based oracles, and also, when an emergency happens, the borrow LTV must be set to 0 but not the liquidation LTV.

This check was in the first place to block vault managers from adding a nested vault of the liability vault as collateral because the non-reentrancy check will revert when the vault tries to price it.

It is improbable that a vault manager adds such a vault, and thus, it was judged better to document it and remove the check simply.

One possible alternative could be to leave the check but add a `bool getQuote` parameter to the function, which would be checked before querying the price.

When using a pull-based oracle or in case of emergency, the vault manager can set it to false to bypass the check, but in other cases, the vault manager can leave it to true to protect himself from adding an incorrect vault.

Impact

Informational.

Recommendation

Consider reverting the change and adding a `bool getQuote` parameter to determine if the function should get a quote.

Developer Response

Acknowledged.

Final Remarks

Auditors support Euler’s decision to remove the `getQuote()` call from the `setLTV()` function, as the complexities and frictions it introduces for vault managers outweigh its benefits. Clear documentation from Euler and responsible management by vault operators should be sufficient to prevent vaults from mistakenly using a nested version of themselves as collateral.