certora

Security Assessment Report

aave

Aave CAPO Precision Update

September 2025

Prepared for:
Aave DAO

Code developed by:
BORED GHOSTS DEVELOPING

Table of contents

Project Summary

Project Scope

Project Name	Repository (link)	Commit Hash	Platform
CAPO Precision Update	https://github.com/bgd-labs/aave-capo.git	PR#103	EVM

Project Overview

This document describes the manual code review findings of Aave CAPO Precision Update. The following contract list is included in our scope:

- `aave-capo/src/contracts/PriceCapAdapterBase.sol`
- `aave-capo/src/contracts/PriceCapAdapterStable.sol`

The work was undertaken from **25.09.2025** to **29.09.2025**. During this time, Certora's security researchers performed a manual audit of all the Solidity contracts and discovered several bugs in the codebase, which are summarized in the subsequent section.

Protocol Overview

Aave relies on oracle adapters to supply safe, reliable asset prices to its risk layers. CAPOs (Capped Price Oracles) are a safeguard that enforce conservative upper bounds on asset prices. They sit between Chainlink (or other price feeds) and the protocol's price consumers, adding governance-controlled caps and growth constraints.

This PR provides updates to the implementations for future CAPO deployments: `PriceCapAdapterBase` now adds a scaling factor for improved calculation precision and tightens snapshot timestamp and parameter validation. `PriceCapAdapterStable` (the stable-asset adapter) introduces an absolute maximum cap, enforces stronger input checks, and improves handling of non-positive aggregator values.

Findings Summary

The table below summarizes the findings of the review, including type and severity details.

Severity	Discovered	Confirmed	Fixed
Critical	-	-	-
High	-	-	-
Medium	-	-	-
Low	-	-	-
Informational	2	2	2
Total	2	2	2

Severity Matrix

Impact	High	Medium	High	Critical
	Medium	Low	Medium	High
	Low	Low	Low	Medium
		Low	Medium	High
		Likelihood		

Detailed Findings

ID	Title	Severity	Status
I-01	decimals not immutable / setter missing	Informational	Fixed
I-02	Negative aggregator answers may return negative cap value	Informational	Fixed

Informational Issues

I-01. `decimals` not immutable / setter missing

File: `src/contracts/PriceCapAdapterStable.sol`

Description

The `decimals` state variable is set once in the constructor from the aggregator (`decimals = ASSET_TO_USD_AGGREGATOR.decimals();`) but is declared as a mutable `uint8` rather than `immutable`. This is a minor gas/intent mismatch and leaves the contract shape slightly ambiguous (it appears writable even though it is never updated).

Recommendation

1. Declare `decimals` as `uint8 public immutable decimals;` to save a small amount of gas and make the contract intent explicit.
2. If on-chain mutability is required by design, add an explicit admin-protected setter and document the rationale.

Status: Fixed in d995761.

I-02. Negative aggregator answers may return negative cap value

File: `src/contracts/PriceCapAdapterStable.sol`

Description

`latestAnswer()` first compares `basePrice` and `priceCap` and returns `priceCap` when `basePrice > priceCap`. If both `basePrice` and `_priceCap` are negative and `basePrice > _priceCap` (e.g. `basePrice = -1`, `_priceCap = -2`), the function will return a negative `priceCap` before reaching the subsequent `basePrice <= 0` guard. This can expose negative prices to callers in edge or mocked environments and is surprising from a robustness perspective.

Recommendation

Reorder the checks to first handle non-positive aggregator responses, for example:

```javascript
int256 basePrice = ASSET_TO_USD_AGGREGATOR.latestAnswer();

if (basePrice <= 0) return 0;

int256 priceCap = _priceCap;
if (basePrice > priceCap) return priceCap;

return basePrice;
```

This ensures negative or zero feed answers are normalized to `0` and prevents negative prices leaking to consumers. Add a unit test simulating negative aggregator responses to validate behavior.

Status: Fixed in 1ec82ab.

Disclaimer

Even though we hope this information is helpful, we provide no warranty of any kind, explicit or implied. The contents of this report should not be construed as a complete guarantee that the contract is secure in all dimensions. In no event shall Certora or any of its employees be liable for any claim, damages, or other liability, whether in an action of contract, tort, or otherwise, arising from, out of, or in connection with the results reported here.

About Certora

Certora is a Web3 security company that provides industry-leading formal verification tools and smart contract audits. Certora's flagship security product, Certora Prover, is a unique SaaS product that automatically locates even the most rare & hard-to-find bugs on your smart contracts or mathematically proves their absence. The Certora Prover plugs into your standard deployment pipeline. It is helpful for smart contract developers and security researchers during auditing and bug bounties.

Certora also provides services such as auditing, formal verification projects, and incident response.