



Aave V3 sUSD Verification and Listing Stewards Audit

Scope

The scope of the assessment is the contracts `MultiCollateralSynth.sol` (`sUSD`) which was formally verified, and `AaveV3OptimismEnableCollateralSteward.sol` which was manually audited due to the fact that the implementation involves only calls to functions with concrete values. The latter contract defines `sUSD` configuration on the Aave V3 platform on Optimism.

As part of our audit, the abstract contract `StewardBase.sol` was inspected as well.

These contracts had 2 security engineers and 1 security researcher reviewing the code in detail.

The verification of the token was finished on the 28th of June, reviewing the [deployed contract on Optimism](#).

The audit was finished on the 1st of August, reviewing commit `4b929bae` of the `sUSD` listing steward.

Contract Overview

As part of our continuous formal verification for Aave, we've inspected the `sUSD` token for security issues and non-trivial features. You can see the results in our [Aave dashboard](#).

The audited contract's purpose is configuring `sUSD` as a collateral token on the Aave V3 platform on the Avalanche network.

The audited contract `AaveV3OptimismEnableCollateralSteward.sol` implements the configuration in 3 steps:

1. Setting a SupplyCap for `sUSD` - [see in code](#).

2. Configuring the parameters for `sUSD` as a collateral asset - [see in code](#).

3. Setting a new interest rate strategy for `sUSD` - [see in code](#).

Audit Goals

During the review of the code, the following checks have been performed:

StewardBase

1. All AAVE roles are correctly declared in the method `getAllAaveRoles` in their `bytes32` form.

AaveV3OptimismEnableCollateralSteward

1. All addresses of external contracts that are used match the existing contracts on the Optimism network.

2. Decimals of all the parameters (`LTV` , `LIQ_THRESHOLD` , `LIQ_BONUS`) match the AAVE standard for those parameters.

3. The asset is being configured as collateral with proper relations between the LTV, Threshold, and Liquidation bonus.

Privileges

4. `AaveV3OptimismEnableCollateralSteward` has the necessary roles to execute `updateSUSDConfig()` without reverting.

5. The roles are renounced from the contract at the end of execution.

Findings And Recommendations

Recommendation

Issue:	Verify that <code>AaveV3OptimismEnableCollateralSteward</code> has the necessary privileges
Description:	Many function calls in the listing process require the configuration contract to have a <code>Risk Admin</code> role in order to be successfully executed. It is important to remember granting those roles prior to calling <code>updateSUSDConfig()</code> .
Recommendation:	Add a dedicated require condition at the beginning of <code>updateSUSDConfig()</code> in order to give a clearer error message in case of such a revert.

Informational - Non-Standard Behavior:

1. The token does not decrease the allowance in `transferFrom()` if it's set to `max uint256`.
2. The zero address might get a non-zero balance as a result of calling `issue()` (only callable by system contracts).
3. sUSD has methods that change the total supply by design.
4. sUSD has methods that burn tokens by design.
5. sUSD is mintable by design.
6. Any user's balance can be decreased by `burn` (which is only callable by system contracts).

You can see the full results in our [AAVE dashboard](#).

Conclusions

StewardBase

1. All AAVE roles specified in the contracts are indeed a direct hashing of the `ACLManager` roles using keccak256.

AaveV3OptimismEnableCollateralSteward

1. All addresses specified in the contract match existing relevant contracts on the Optimism blockchain.
2. The assigned decimals in the contract match the AAVE standard.
3. The relations between the LTV, Threshold, and Liquidation bonus were configured reasonably.

Privileges

4. The roles of the contract are given externally. Therefore, the executor should delegate the necessary privileges before trying to execute.
5. At the end of the process the contract renounces its privileges in a correct manner.

Disclaimer

We hope that this information is useful, but 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.

