



Security Assessment Report



Agora Whitelabel Update

October 2025

Prepared for Agora Team

Table of content

Project Summary.....	3
Project Scope.....	3
Project Overview.....	3
Findings Summary.....	4
Severity Matrix.....	4
Detailed Findings.....	5
Low Severity Issues.....	6
L-01. Hardcoded AUSD address can be difficult to match on certain chains.....	6
Informational Issues.....	7
I-01. Use of Simple Ownable Instead of 2-Step Ownership.....	7
I-02. totalAssets() returns unrecoverable funds.....	8
Disclaimer.....	9
About Certora.....	9

Project Summary

Project Scope

Project Name	Repository (link)	Commit Hash	Platform
Agora Whitelabel	https://github.com/amphora-atlas/whitelabel-evm-dev	d1e17ff (initial commit) 2763e4d (fix commit)	Solidity

Project Overview

This document describes the manual code review findings of **Certora Agora Whitelabel**. The following contract list is included in our scope:

- src/contracts/Whitelabel.sol
- src/contracts/TransparentUpgradeableProxy.sol
- src/contracts/TwoStepProxyAdmin.sol

The work was undertaken from **September 29, 2025**, to **October 13, 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 and Scope Overview

Agora issues AUD, a fiat-backed stablecoin fully collateralized by USD reserves. AUD supports on-chain transfers across multiple networks using LayerZero for cross-chain interoperability.

This review focuses on the Whitelabel module, an upgradeable ERC-4626 vault that wraps AUD to provide 1:1 deposits and withdrawals with whitelisted tokens while enforcing AUD's freeze and blacklist checks on transfers, mints, and burns; along with its supporting upgradeability components: **TwoStepProxyAdmin**, a safer **ProxyAdmin** variant requiring two-step ownership transfers, and a forked **TransparentUpgradeableProxy** configured to use that admin.

Findings Summary

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

Severity	Discovered	Confirmed	Fixed
Critical	-	-	-
High	-	-	-
Medium	-	-	-
Low	1	1	-
Informational	2	2	2
Total	3	3	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
L-01	Hardcoded AUSD address can be difficult to match on certain chains	Low	Acknowledged
I-01	Use of Simple Ownable Instead of 2-Step Ownership	Informational	Fixed
I-02	totalAssets() returns unrecoverable funds	Informational	Fixed

Low Severity Issues

L-01. Hardcoded AUSD address can be difficult to match on certain chains

Severity: **Low**

Impact: **Low**

Likelihood: **Low**

Files: Whitelabel.sol

Status: Acknowledged

Description:

The whitelabel contract contains the **AUSD** token address in a hardcoded constant variable in the contract, which cannot be altered. The issue is that on certain EVM chains, it can be difficult to deploy the **AUSD** contract to the same address. On most EVM chains, contracts are deployed to the same addresses, provided the nonce is the same.

However, for certain chains like zkSync, deployment addresses can be different. If deployed using the create opcode, zksync uses [two](#) different nonces, and if deploying using create2, zksync uses a different address [derivation](#) method, which can make it difficult to match the address to other EVM chains.

Recommendations: If such chains are to be targeted, consider making the **AUSD** variable a configurable value.

Customer's response: Won't fix. AUSD is always deployed to the same known address on every EVM chain. Where not possible, we'll update the address in the code.

Informational Issues

I-01. Use of Simple Ownable Instead of 2-Step Ownership

File: Whitelabel.sol

Description

The `Whitelabel.sol` contract uses `OwnableUpgradeable` instead of `Ownable2StepUpgradeable`, allowing immediate ownership transfers without confirmation.

Impact

If the owner accidentally sets an incorrect address (especially a dead address) during ownership transfer, there's no way to recover the ownership. This also creates inconsistent security with other main contracts in the codebase that use `Ownable2Step`.

Recommendations

Replace `OwnableUpgradeable` with `Ownable2StepUpgradeable`.

Customer's response: Fixed following recommendations in [PR](#).

Fix Review: Fix looks good.

I-02. totalAssets() returns unrecoverable funds

File: Whitelabel.sol

Description

The totalAssets function is inherited from the `ERC4626Upgradeable` contract, and it returns the contract balance of the underlying token. However, since the `_convertToShares` and `_convertToAssets` functions have been overridden to a 1:1 ratio, these funds could actually be inaccessible. Normally, for `ERC4626` vaults, there is an inherent assumption that burning `totalSupply` amount of shares will return `totalAssets` amount of assets, outside of rounding errors.

However, this isn't the case here, since `totalAssets` is never used in the calculations as it is hardcoded to a 1:1 ratio. Thus, any funds donated to the whitelabel contract will show up under `totalAssets` but will actually not be recoverable.

Recommendations

Consider changing `totalAssets` to return the `totalSupply` instead, to reflect the actual amount of recoverable funds.

Also, consider adding a recovery function to remove any excess `AUSD` stuck in the whitelabel contract.

Customer's response: `totalAssets` now calculated from `totalSupply*price` in [PR](#). Recovery function not added, acknowledged as a future upgrade.

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.