PUBLIC

Code Assessment

of the Sulu Extensions XVI Smart Contracts

April 29, 2024

Produced for

by

Contents

1 Executive Summary

Dear all,

Thank you for trusting us to help Avantgarde Finance with this security audit. Our executive summary provides an overview of subjects covered in our audit of the latest reviewed contracts of Sulu Extensions XVI according to Scope to support you in forming an opinion on their security risks.

Avantgarde Finance implements a new Enzyme external position to integrate with Morpho Blue, an immutable overcollateralized lending protocol.

The most critical subjects covered in our audit are functional correctness, asset solvency and the integration in Sulu. Security regarding all the aforementioned subjects is high.

In summary, we find that the codebase provides a high level of security.

It is important to note that security audits are time-boxed and cannot uncover all vulnerabilities. They complement but don't replace other vital measures to secure a project.

The following sections will give an overview of the system, our methodology, the issues uncovered and how they have been addressed. We are happy to receive questions and feedback to improve our service.

Sincerely yours,

ChainSecurity

1.1 Overview of the Findings

Below we provide a brief numerical overview of the findings and how they have been addressed.

Critical-Severity Findings	0
High-Severity Findings	0
Medium-Severity Findings	1
• Code Corrected	1
Low-Severity Findings	0

2 Assessment Overview

In this section, we briefly describe the overall structure and scope of the engagement, including the code commit which is referenced throughout this report.

2.1 Scope

The assessment was performed on the source code files inside the Enzyme Protocol repository based on the documentation files. The table below indicates the code versions relevant to this report and when they were received.

V	Date	Commit Hash	Note
1	22 April 2024	80f5257078b7f1b9283df170ae1113f5ed1607f8	Initial Version
2	25 April 2024	b6eae53ccda0bc5a7e8d21bbfb5a90c8eadf26eb	After Intermediate Report

For the solidity smart contracts, the compiler version `0.8.19` was chosen.

The scope for the MorphoBlue external position is:

```
contracts/external-interfaces/contracts/IMorphoBlue.sol
contracts/release/extensions/external-position-manager/external-positions/morpho-blue/IMorphoBluePosition.sol
contracts/release/extensions/external-position-manager/external-positions/morpho-blue/MorphoBluePositionDataDecoder.sol
contracts/release/extensions/external-position-manager/external-positions/morpho-blue/MorphoBluePositionLib.sol
contracts/release/extensions/external-position-manager/external-positions/morpho-blue/MorphoBluePositionParser.sol
contracts/release/extensions/external-position-manager/external-positions/morpho-blue/bases/MorphoBluePositionLibBase1.sol
contracts/utils/0.8.19/Bytes32ArrayLib.sol
```

2.1.1 Excluded from scope

Any contracts inside the repository that are not mentioned in `Scope` are not part of this assessment. All external libraries, and imports are assumed to behave correctly according to their high-level specification, without unexpected side effects. Notably, this includes MorphoBalancesLib.

More derived usage of template contracts and mixins is not in scope, with the exception of the more derived contracts in scope.

The selection of deployment parameters is not in scope.

The correctness of external systems is not in scope.

Tests and deployment scripts are excluded from the scope.

2.2 System Overview

This system overview describes the initially received version (Version 1) of the contracts as defined in the Assessment Overview.

Furthermore, in the findings section, we have added a version icon to each of the findings to increase the readability of the report.

Avantgarde Finance implements a new external position for integrating with Morpho Blue.

2.2.1 External Position: Morpho Blue

Morpho Blue is an immutable overcollateralized lending protocol with many individual lending markets. This external position enables interactions with markets whitelisted by Enzyme's governance. Technically Enzyme's UintListRegistry registry is used to manage the list. For a detailed description of how external positions work, please refer to the main audit report of Sulu.

The external position implements the following actions to interact with whitelisted markets:

1. `Lend`: Supplies the specified amount of loan asset for the specified `marketId` (`MorphoBlue.supply()`). The assets are pulled from the vault.
2. `Redeem`: Redeems the specified amount of supply shares from the specified `marketId` (`MorphBlue.withdraw()`). The shares are converted to assets and deposited to the vault.
3. `AddCollateral`: Supplies the specified amount of collateral asset for the specified `marketId` (`MorphoBlue.supplyCollateral()`). The assets are taken from the vault.
4. `RemoveCollateral`: Removes the specified amount of collateral asset for the specified `marketId` (`MorphoBlue.withdrawCollateral()`) The assets are deposited to the vault.
5. `Borrow`: Borrows the specified amount of loan assets for the specified `marketID` (`MorphoBlue.borrow()`). The position must have sufficient collateral locked previously. The borrowed assets are transferred to the vault.
6. `Repay`: Allows to repay previously borrowed assets and interests accrued (`MorphoBlue.repay()`). The assets are taken from the vault. If the repayment amount is set to `type(uint256).max`, the external position will calculate and repay all borrowed assets of the position.

This position takes on debt (the borrowed assets which must be repaid to free the collateral). `getDebtAssets()` iterates over all active markets this external position has and aggregates the borrowed assets.

`getManagedAssets()` evaluates the managed assets of the position. It iterates over all active markets and aggregates the assets supplied for lending as well as the assets supplied as collateral. Improving positions in Morpho Blue is permissionless, anyone may add collateral or supply loan tokens on behalf of an external position's position in Morpho Blue. Such a market may not be tracked yet and hence not included in `getManagedAsset()`. Outstanding debt can similarly be repaid directly through Morpho Blue.

A public getter `getMarketIds()` is provided which returns the array of active markets of this external position.

2.2.2 Changes in Version 2

Evaluating the value of the External Position (`getManagedAssets()`) now reverts if a tracked market is no longer whitelisted. This blocks the fund valuation (`calcGav()`) and actions depending on it. Fund managers are expected to promptly unwind these positions to allow `getManagedAssets()` to function again. To facilitate this, unwinding actions (`Redeem`, `RemoveCollateral` and `Repay`) are no longer limited to whitelisted markets.

This behavior of the EP aligns with the behavior of previously supported assets: The fund valuation (`calcGav()`) fails until these assets have been traded away.

In cases where it is impossible to unwind a problematic Morpho Blue position held by the External Position (e.g., due to issues at the Morpho Blue market), all other positions held by this External Position must be unwound, and the External Position deactivated.

2.2.3 Roles and Trust Model

Please refer to the main audit report and the extension audit reports for a general trust model of Sulu.

In general, we assume Enzyme (including the whitelisted Morpho Blue markets) only interacts with normal ERC-20 tokens that do not have multiple entry points, callbacks, fees-on-transfer, or other special behaviors.

Fund owners and asset managers are generally fully trusted for a fund. However, their powers can be limited through the fund's settings. The funds' settings/policies are assumed to be set up correctly for the intended configuration/usage.

Governance is fully trusted and expected to not only behave honestly but also to fully understand the systems Enzyme is interacting with. Notably the Governance whitelists the markets that the Morpho Blue external positions are allowed to interact with. These markets must be evaluated carefully before being whitelisted: Generally only markets with assets supported by Enzyme should be added. Morpho Blue markets, while being immutable and permissionless, may block certain operations or behave unexpectedly if the oracle used by the market reverts or returns stale values. Hence the supported markets should be monitored continuously.

External systems (notably Morpho Blue) are expected to work correctly and as expected.

3 Limitations and use of report

Security assessments cannot uncover all existing vulnerabilities; even an assessment in which no vulnerabilities are found is not a guarantee of a secure system. However, code assessments enable the discovery of vulnerabilities that were overlooked during development and areas where additional security measures are necessary. In most cases, applications are either fully protected against a certain type of attack, or they are completely unprotected against it. Some of the issues may affect the entire application, while some lack protection only in certain areas. This is why we carry out a source code assessment aimed at determining all locations that need to be fixed. Within the customer-determined time frame, ChainSecurity has performed an assessment in order to discover as many vulnerabilities as possible.

The focus of our assessment was limited to the code parts defined in the engagement letter. We assessed whether the project follows the provided specifications. These assessments are based on the provided threat model and trust assumptions. We draw attention to the fact that due to inherent limitations in any software development process and software product, an inherent risk exists that even major failures or malfunctions can remain undetected. Further uncertainties exist in any software product or application used during the development, which itself cannot be free from any error or failures. These preconditions can have an impact on the system's code and/or functions and/or operation. We did not assess the underlying third-party infrastructure which adds further inherent risks as we rely on the correct execution of the included third-party technology stack itself. Report readers should also take into account that over the life cycle of any software, changes to the product itself or to the environment in which it is operated can have an impact leading to operational behaviors other than those initially determined in the business specification.

4 Terminology

For the purpose of this assessment, we adopt the following terminology. To classify the severity of our findings, we determine the likelihood and impact (according to the CVSS risk rating methodology).

- *Likelihood* represents the likelihood of a finding to be triggered or exploited in practice
- *Impact* specifies the technical and business-related consequences of a finding
- *Severity* is derived based on the likelihood and the impact

We categorize the findings into four distinct categories, depending on their severity. These severities are derived from the likelihood and the impact using the following table, following a standard risk assessment procedure.

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

As seen in the table above, findings that have both a high likelihood and a high impact are classified as critical. Intuitively, such findings are likely to be triggered and cause significant disruption. Overall, the severity correlates with the associated risk. However, every finding's risk should always be closely checked, regardless of severity.

5 Findings

In this section, we describe any open findings. Findings that have been resolved have been moved to the Resolved Findings section. The findings are split into these different categories:

- Design: Architectural shortcomings and design inefficiencies

Below we provide a numerical overview of the identified findings, split up by their severity.

Critical-Severity Findings	0
High-Severity Findings	0
Medium-Severity Findings	0
Low-Severity Findings	0

6 Resolved Findings

Here, we list findings that have been resolved during the course of the engagement. Their categories are explained in the Findings section.

Below we provide a numerical overview of the identified findings, split up by their severity.

Critical-Severity Findings	0
High-Severity Findings	0
Medium-Severity Findings	1

- Reducing Position Supported Only in Whitelisted Markets Code Corrected

Low-Severity Findings	0

6.1 Reducing Position Supported Only in Whitelisted Markets

Design Medium Version 1 Code Corrected

CS-SUL16-001

The external position allows interaction with whitelisted Morpho Blue markets only. This implies that if a market is delisted, the vault will no longer be able to interact with it in any way. In particular, the functions `__redeem`, `__removeCollateral`, and `__repay` will revert when called, preventing the fund manager from unwinding the position. As a consequence, if some assets were lent or supplied as collateral, they will remain locked in the delisted Morpho Blue market.

Code corrected:

Avantgarde Finance has corrected the code in Version 2 by removing the `__validateMarketId` function call from the functions `__redeem`, `__removeCollateral`, and `__repay`.

7 Notes

We leverage this section to highlight further findings that are not necessarily issues. The mentioned topics serve to clarify or support the report, but do not require an immediate modification inside the project. Instead, they should raise awareness in order to improve the overall understanding.

7.1 Enzyme Pricefeed Vs Morpho Blue Market Oracle

Note Version 1

Fund manager must be aware that there can be a discrepancy between the Enzyme pricefeed for an asset and the oracle used by the Morpho Blue market for the same asset. It is crucial to use the correct oracle, for example, assessing the health of a position in Morpho Blue should be based on its market oracle, not the Enzyme price feeds.

7.2 External Position With Unhealthy Morpho Blue Position

Note Version 1

Unhealthy positions in Morpho Blue are up for liquidation. Liquidations can liquidate up to 100% of the positions `borrowShares` in exchange for the corresponding value in collateral, along with a liquidation incentive which is proportional to the value being liquidated. As a result, after the liquidation, the positions value reduced by the liquidation incentive.

Hence upon detection of an unhealthy Morpho Blue position, shareholders of the fund may be incentivized to exit the fund before the liquidation takes place using `redeemSharesForSpecificAssets()` in an attempt to to avoid taking the loss which is then carried by the remaining shareholders.

7.3 Risks of Repay in Delisted Markets

Note Version 2

Action `repay` is no longer restricted to whitelisted markets only in order to facilitate unwinding of positions in delisted markets.

Repaying borrowed assets may be necessary to release locked collateral, but it does not guarantee that the collateral can actually be retrieved. For instance, repaying can work when the market oracle reverts (a potential reason why a market could have been delisted), but withdrawing the collateral with `__withdrawCollateral` would not be possible since the health check queries the oracle.

Furthermore, if the magic value is specified, action repay computes the amount of borrow assets based on the current borrow share price, which is computed involving the IRM. As a consequence, a bad IRM could in theory inflate the share price and pull more assets than expected from the vault. Note that if the magic value is not specified, it could happen that the borrow shares repaid are less than the amount expected.

On markets that have been delisted, the Fund Manager must use this function with great care.