Pendle

MarketV6 Security Review

HickupHH3

26 September 2025

Contents

1 Introduction

The purpose of this audit is to review the introduction of a new upgradable factory and a slightly modified PT / YT / Market contract, with the goal of being able to change the market implementation in the future without changing the MarketFactory.

For `Market`, a new function exposing the reentrancy status was introduced. The LP oracle library takes advantage of this change.

1.1 Audit Scope

The scope consisted of the contracts modified in the `pendle-core-v2` repository at the `ngfam/upgradable-factory` branch at commit hash `c34284095a2c800d9e3797e4d8ae5affc1fda3f0`. The main files reviewed were:

- `PendleMarketFactoryV6Upg.sol`
- `PendleMarketV6.sol`
- `PendlePrincipalToken.sol`
- `PendleYieldContractFactory.sol` -> `PendleYieldContractFactoryUpg`
- `PendleYieldToken.sol`
- `PendleGaugeControllerBaseUpg.sol`
- `PendleGaugeControllerMainchainUpg.sol`
- `PendleGaugeControllerSidechainUpg.sol`
- `PendleLpOracleLib.sol`

with other minor changes made to interfaces and helpers.

1.2 Audit Timeline

The audit was conducted on **26th September**.

1.3 Fix Review

Minor changes were introduced and reviewed on **18th October**.

1.4 Auditors Involved

HickupHH3

2 Risk Assessment Classification

There are 4 possible levels used to assess a vulnerability, with a separate section for gas optimizations.

High

Directly exploitable vulnerabilities with medium / high likelihood of loss of user funds, or contract functionality.

Resolving these issues are crucial to ensure the security and functionality of the contracts.

Medium

Vulnerabilities that relies on external dependencies / conditions to be met. Potentially leads to a loss of funds or functionality (eg. denial of service).

Resolving these issues are recommended to avoid undesired consequences.

Low

Issues arising from deviant behaviour than expected, but has no / little bearing from a security standpoint.

Informational

Issues that relate to security best practices recommendations, grammatical or styling errors, suggestions for variable/function name improvements etc. These issues are subjective and can be addressed based on the client's discretion.

While these issues may not directly affect the contract's functionality or security, addressing them can improve code readability, maintainability, and overall quality.

Gas Optimizations

Suggested changes to the codebase that will help reduce deployment or runtime gas costs, or to reduce the bytecode size should the limit be reached.

3 Findings Summary

Severity	No. of issues
High	0
Medium	0
Low	0
Informational	2
Gas Optimizations	0
Total	2

3.1 [Info] Library usage can be simplified

Context

PendleYieldContractFactoryUpg.sol#L122-L123

PendleMarketV6.sol#L96-L103

Details

The `StringLib` library prefix is redundant because of the `using StringLib for string;` and `using StringLib for StringLib.slice;` lines.

Mitigation

```
- return LP_NAME_PREF.toSlice().concat(
-     StringLib.stripPrefixSlice(
-         IPPrincipalToken(_PT).name(), PT_NAME_PREF
-     )
- );
+ return LP_NAME_PREF.toSlice().concat(
+     IPPrincipalToken(_PT).name().stripPrefixSlice(PT_NAME_PREF)
+ );

- return LP_SYMBOL_PREF.toSlice().concat(
-     StringLib.stripPrefixSlice(
-         IPPrincipalToken(_PT).symbol(), PT_SYMBOL_PREF
-     )
- );
+ return LP_SYMBOL_PREF.toSlice().concat(
+     IPPrincipalToken(_PT).symbol().stripPrefixSlice(PT_SYMBOL_PREF)
+ );
```

Response

Fixed at commit 238468a.

Status

Fixed.

3.2 [Info] Out-of-place comment

Context

IPMarketFactory.sol#L5

Details

An out-of-place comment was introduced that has no relation to the line below it.

Mitigation

Removed the referenced line.

Response

Fixed at commit 238468a.

Status

Fixed.

4 Disclaimer

The audit report provided reflects a thorough review conducted to the best of my ability. However, it is important to note that the time-boxing nature of the review and available resources may prevent the discovery of all potential security vulnerabilities. As such, this audit does not guarantee the absence of undiscovered vulnerabilities.

Furthermore, please be aware that the security review was conducted on a specific commit of the codebase, as indicated. Any subsequent modifications made to the code will necessitate a new security review to ensure comprehensive coverage.

Note that the contracts used in production and expected deployment values may defer significantly from what was reviewed.

To ensure a robust evaluation of the codebase, it is highly recommended to engage multiple auditors and firms, particularly for large and complex projects. The involvement of multiple perspectives can provide additional insights and potential missed vulnerabilities.

Please consider these factors when assessing the audit report and making decisions related to the security and reliability of the smart contracts. The security review is not an endorsement of the project or its team, and should not be treated as such.