OpenZeppelin | security

Compound - Contango Integration Audit

May 2, 2024

Table of Contents

Summary

Type	DeFi
Timeline	From 2024-03-18 To 2024-04-05
Languages	Solidity

Total Issues	3 (0 resolved)
Critical Severity Issues	0 (0 resolved)
High Severity Issues	0 (0 resolved)
Medium Severity Issues	0 (0 resolved)
Low Severity Issues	0 (0 resolved)
Notes & Additional Information	3 (0 resolved)

Scope

We audited the contango-xyz/core-v2 repository at commit 3abfea9.

In scope were the following files::

```
src
└── moneymarkets
    ├── compound
    │   ├── CompoundMoneyMarket.sol
    │   ├── CompoundMoneyMarketView.sol
    │   └── CompoundReverseLookup.sol
    ├── comet
    │   ├── CometMoneyMarket.sol
    │   ├── CometMoneyMarketView.sol
    │   └── CometReverseLookup.sol
    ├── BaseMoneyMarketView.sol
    └── BaseMoneyMarket.sol
```

System Overview

Contango is a protocol offering on-chain perpetual futures contracts. To achieve leverage, Contango is utiizing lending protocols such as Compound V2 and Comet. The audited contracts are implementations of their general money market interface, allowing Contango to access Compound's money markets in a standardized manner. Those contracts contain functionality for borrowing and lending the base asset, as well as providing and withdrawing collateral.

Security Model and Trust Assumptions

As Contango is a third party protocol utilizing Compound's functionality, there is no additional risk incurred by the Compound protocol.

Notes & Additional Information

N-01 Missing documentation

Given the project's complicated logic that interfaces with various lending protocols, it is crucial to have clear specifications for each function. Currently, the project lacks comprehensive documentation for the majority of its features. Implementing NatSpec documentation is vital to clarify points mentioned below:

- The intended use of each function's inputs
- The expected outputs from each function
- Assumptions made for each functionality
- The reason behind design choices in the functionalities

Although the absence of such documentation does not affect the protocol's practical applications, it is essential for the auditing process and for the ongoing management of the project.

Consider adding NatSpec documentation for all features, addressing the points listed above.

Update: *Acknowledged, not resolved. The Contango team stated:*

> *will document the generic interface, but won't block the release on that*

N-02 `UniswapAnchoredView` deprecation

UniswapAnchoredView might be disabled on Compound V2 and be replaced with a normal price feed. While this does not change the functionality or break the code, the naming might become outdated after this change.

Consider using a price feed interface instead which only requires the `getUnderlyingPrice` functionality as expected by the `Comptroller`.

Update: *Acknowledged, not resolved. The Contango team stated:*

> *CompoundV2 has barely been used by anyone so far, so when/if that becomes a problem we'll make the choice of changing the oracle or simply delist that money market, as it seems to be on its way out anyway. The main reason to delay this is that atm we use some internals of that oracle on the View contract to lookup cTokens, so we'd need to wait until the replacement is there to be able to test with it.*

N-03 Direct invocation of userCollateral array not necessary

To query Contango's collateral balance on Comet, at several places in `CometMoneyMarket.sol` and `CometMoneyMarketView.sol`, Comet's array `userCollateral` is directly read:

- line 43 of CometMoneyMarket
- line 63 of CometMoneyMarket
- line 40 of CometMoneyMarketView

However, `CometExt.sol` provides a dedicated function `collateralBalanceOf` for reading collateral balances.

Consider using the function provided by the Comet extension to improve readability of the code.

Update: *Acknowledged, not resolved.*

Conclusion

During this audit, we reviewed the Contango protocol's integration with Compound V2 and Comet in detail. Our investigation included various components of the protocol, taking into account the structure of both Compound V2 and Comet. The codebase appears to be functioning appropriately; however, due to the project's lack of detailed specifications and the audit's focus being confined to the integration code, we cannot fully confirm the validity of the functionalities.

In conclusion, while the code appears operational, we could not confirm that its functionality matches the specifications, due to the lack of documentation. We strongly recommend developing comprehensive documentation for the protocol to ensure that all elements are functioning as expected.