Contango Perpetual Option Review

October 28, 2024

Prepared for Contango

Conducted by:

Vara Bandaru (s3v3ru5)

Richie Humphrey (devtooligan)

About the Contango Perpetual Option Review

Contango builds perps by automating a looping strategy, also known as recursive borrowing and lending. This is achieved using spot and money markets.

The perpetual option contract for "oTANGO" a perpetual option token for Contango's TANGO token. It allows holders to purchase at a variable discount.

About Offbeat Security

Offbeat Security Labs, LLC is a boutique security company providing unique security solutions for complex and novel crypto projects. Our mission is to elevate the blockchain security landscape through invention and collaboration.

Summary & Scope

The token (https://github.com/contango-xyz/core-v2-private/tree/82b845deedec39f008bc2596301e1076392695f6) folder of the `core-v2-private` repo was reviewed at commit 82b845deedec39f008bc2596301e1076392695f6.

The following **2 contracts** were in scope:

- src/token/ContangoToken.sol
- src/token/ContangoPerpetualOption.sol

Summary of Findings

Identifier	Title	Severity	Fixed
L-01	Mint function lacks access control	Low	Fixed in b16c5b4 (https://github.com/contango-xyz/core-v2-private/commit/b16c5b428886a7edc97f3599ec7ac310ef5a9e6d)
I-01	Unnecessary use of flashloan	Informational	Fixed in b16c5b4 (https://github.com/contango-xyz/core-v2-private/commit/b16c5b428886a7edc97f3599ec7ac310ef5a9e6d)

Overview

ContangoToken

This contract inherits OpenZeppelin's ERC20Permit, ERC20FlashMint, and Ownable contracts which were not in scope for this review. The native flash loan functionality is added as a convenience because not all markets have flashloans. The contract adds a public `burn` function and a permissioned `mint` which allows the owner to mint up to the `MAX_SUPPLY` cap of 1,000,000,000e18.

We did not note any security issues with the `ContangoToken` contract.

ContangoPerpetualOptionToken

The `oTANGO` tokens are created by the treasury by funding `TANGO` tokens. The `oTANGO` options will be distributed and the holders can exercise these options to purchase `TANGO` at a discount from market.

The contract code is loosely based on Bunni's OptionsToken (https://github.dev/timeless-fi/options-token/blob/main/src/OptionsToken.sol). It does not make use of storage for critical operations outside of standard ERC20 functionality and has no permissioned functions.

The contract does not contain a high degree of complexity. The discount calculations and oracle interactions are straightforward.

Discount calculation

The discount starts when the token price exceeds 0.045 and increases logarithmically. It is capped at the maximum discount rate of 0.75 when the price of the underlying `TANGO` reaches 1.0.

```
// discount formula
A = MAX_DISCOUNT / (ln(START_FLAT) - ln(TANGO_SEED_PRICE));
B = -A * ln(TANGO_SEED_PRICE);
discount = tangoPrice_ < START_FLAT ? A * ln(tangoPrice_) + B : MAX_DISCOUNT;
```

We modeled this in Desmos for a visual representation of the discount (https://www.desmos.com/calculator/girjzjhjne) and the net price (https://www.desmos.com/calculator/kecciyecjw).

DIA Oracle

The `ContangoPerpetualOption` uses the DIA oracle and the price retrieval logic is based on the `DiaOracleV2` adapter created by Silo Finance (https://silopedia.silo.finance/), a lending protocol on Arbitrum who uses DIA for long tail assets.

The project team has reported that the oracle uses off-chain TWAP observations, which makes it difficult to manipulate prices. In contrast to on-chain observations, which are vulnerable to front-running and sandwich attacks, off-chain data cannot be as easily predicted or intercepted since the timing of observations is unknown.

Code Assumptions

Assumptions noted:

- The correct treasury and oracle addresses are passed as deployment arguments.
- The DIA oracle will not suffer any permanent or long term outages.
- The oracle cannot be manipulated to create profitable opportunities.

Detailed Findings

Low Findings

[L-01] Mint function lacks access control

The `mint()` function is intended to be called by the contract owner to mint option tokens in exchange for transferring the underlying token 1:1 to the contract. This function is a public function which can be called by anyone as long as they approve the contract for transfer of the underlying.

When a non-owner user calls this either due to a misunderstanding or by mistake, they will lose the amount of the exercise price they need to buy their tokens back. Furthermore, there is no valid use case for a non-owner to call this function.

Note, this would be less likely to happen because the caller would also have to approve the underlying `TANGO` tokens in advance.

Recommendation

Consider renaming this function and adding a warning in the comment.

Informational Findings

[I-01] Unnecessary use of flashloan

The `ContangoPerpetualOption` contract inherits OpenZeppelin's `ERC20FlashMint` which adds flash loan capability to the token.

The project team confirmed that there is no valid use case for this functionality. While we did not identify any direct attack vectors or security issues related to this, it is unclear what effects this could have if the token were integrated with other DeFi protocols. Furthermore, it adds unecessary bloat to the bytecode size without clear benefit.

Recommendation

Remove the `ERC20FlashMint` dependency.