Mitigation review

Project: Olas Lockbox v2 (after Cantina campaign)
Commit: 0c652b0528dc522f92b7191862897cbbe8f159f9
Start Date: 2024-03-08

Scope of mitigation measures to review:

- PR 13 - Adding vulnerabilities doc for lockbox2 specified by the Cantina audit findings.
- PR 14 - Addressing sandwich attack fix in the deposit() function
- PR 15 - Addressing small Cantina audit findings

Reaction to suggestions of this mitigation review:
Commit: 8b61218c4cbfaad05689f9e9f303239ec14d4918
Review Date: 2024-03-15
Scope: PR 18 - Addressing external audit findings

1. Sandwich attack mitigation

After introduction of a liquidity_amount parameter in the deposit() function, it is ensured the a user receives the expected liquidity_amount of bridged tokens while only spending token_max_a of SOL and token_max_b of OLAS in the worst-case.
As a consequence of those in- and output constraints, a user is **sufficiently** protected from a sandwich attack.

Furthermore, the direct use of liquidity_amount, token_max_a and token_max_b with the underlying Whirlpool program led to obsolete code, which was previously used for the computation of the liquidity & token b amounts, that facilitated the sandwich attack in the first place.
Consequently, the get_liquidity_from_token_a() function became obsolete too and was removed and therefore also resolved the Division before multiplication in liquidity_lockbox::get_liquidity_from_token_a(...) #50 issue.

In addition, the approval of any unused SOL & OLAS tokens (not all of token_max_a/token_max_b used) is revoked (set to 0).

OK ✓

2. Improvements according to Low and Informational findings

1. Hardcoded PROGRAM_ID was replaced with the implicitly available ID constant.
   Instances: #1 & #2
   Recommendation: Use liquidity_lockbox::ID to improve readability/clarity.
   OK ✓
2. Length verification of position account now returning / reverting with error code instead of using assert!() macro.
   Instance: #1
   OK ✓
3. Moved ownership checks of lower/upper tick arrays from function body to function account context constraints (of deposit() and withdraw() methods).
   Instances: from #1 & #2 to #3 & #4
   OK ✓
4. Error handling improvement: Return / revert with error code instead of using panic!() macro.
   Instances: #1 [overflow handling, deposit()] & #2 [underflow handling, withdraw()]
   OK ✓
5. Implemented address check to ensure that the pda_position_account is exactly the one in the lockbox account.
   Instances: #1 & #2
   OK ✓
6. Use actually declared token account in corresponding Anchor constraint (instead of elsewhere declared token mint).
   It's still ensured that the token mint is exactly the one in the lockbox account.
   Instances: #1 & #2
   OK ✓
7. Removed unused system_program and rent from function account context.
   Instance: #1
   OK ✓

3. Further concerns

1. Wrong comment over deposit() method: User deposits SOL & OLAS tokens, not an NFT.
   Instance: #1
   Fixed: #1
   OK ✓
2. Inconsistent declaration of position account.
   Instances: #1 vs. #2
   Recommendation: Add has_one = position_mint to #1.
   Fixed: #1; also at initialization and explicitly added position_mint account with correct address & supply checks
   OK ✓
3. Issue Attacker can frontrun lockbox initialization to provide own fee token accounts #52 is unmitigated.
   Acknowledged by the team: By design the contract has no ownership access, so we assume that the initialization is correctly done by the deployer.
   OK ✓

4. Out-of-scope concerns

Not part of Lockbox v2 but still part of PR 15.

1. Anchor's account close() function, see source code, is not doing exactly the same as the previous code.
   After defunding, instead of zeroing the account's data and overwriting it with the CLOSED_ACCOUNT_DISCRIMINATOR, the account is now assigned to the system_program and then its size is reallocated to 0.
   Acknowledged by the team: here
   OK ✓
2. The signer account is declared *read-only* (missing Anchor mutable constraint) which is technically incorrect since it's modified/written when closing a position.
   Fixed: here
   OK ✓