CoinFabrik

AlexGo Audit

Launchpad, Yield Vault and Collateral Rebalancing Pool

April 2022

By CoinFabrik

Introduction

CoinFabrik was asked to audit the contracts for the AlexGo project. First we will provide a summary of our discoveries and then we will show the details of our findings.

Scope

The contracts audited are from the https://github.com/alexgo-io/alex-v1 git repository. The audit is based on the commit `e268fd53370be3a271625bd45523ae07cb1239ac`.

The audited contracts are:

- `clarity/contracts/pool/alex-launchpad-v1-1.clar`: Launchpad contract for IDO creation.
- `clarity/contracts/pool/collateral-rebalancing-pool.clar`: Contract for collateral-token pool creation.
- `clarity/contracts/pool/yield-vault-alex.clar`: Vault for yield tokens.

The scope of the audit is limited to those files. No other files in this repository were audited. Its dependencies are assumed to work according to their documentation. Also, no tests were reviewed for this audit.

Analyses

Without being limited to them, the audit process included the following analyses:

- Arithmetic errors
- Race conditions
- Misuse of block timestamps
- Denial of service attacks
- Excessive gas usage
- Needlessly complex code and contract interactions
- Poor or nonexistent error handling
- Insufficient validation of the input parameters
- Centralization and upgradeability
- Weak authentication

Summary of Findings

We found a medium issue and a minor issue. Also, two enhancements were proposed.

The two issues were acknowledged.

Security Issues

ID	Title	Severity	Status
ME-01	Launchpad: Full Distribution not Guaranteed	Medium	Acknowledged
MI-01	Launchpad: Tokens Locked due to Incomplete Distribution	Minor	Acknowledged

Privileged Roles

These are the privileged roles that we identified on each of the audited contracts.

alex-launchpad-v1-1

Owner

The owner is the only role which can create ticket pools, required to start an IDO. It is also capable of providing tickets to a pool, calling the claiming and refunding function without waiting for the grace period to end, and transferring the entire balance of a specific token to the owner.

Approved Operator

This role can also provide tickets to a pool and call the claiming and refunding functions without waiting for the grace period to end.

IDO Owner

The IDO owners can add tickets to their pool and call the claiming and refunding functions without waiting for the grace period to end.

Security Issues Found

Severity Classification

Security risks are classified as follows:

- **Critical:** These are issues that we manage to exploit. They compromise the system seriously. They must be fixed **immediately**.
- **Medium:** These are potentially exploitable issues. Even though we did not manage to exploit them or their impact is not clear, they might represent a security risk in the near future. We suggest fixing them **as soon as possible**.
- **Minor:** These issues represent problems that are relatively small or difficult to take advantage of but can be exploited in combination with other issues. These kinds of issues do not block deployments in production environments. They should be taken into account and be fixed **when possible**.

Issues Status

An issue detected by this audit can have four distinct statuses:

- **Unresolved**: The issue has not been resolved.
- **Acknowledged**: The issue remains in the code but is a result of an intentional decision.
- **Resolved**: Adjusted program implementation to eliminate the risk.
- **Partially resolved**: Adjusted program implementation to eliminate part of the risk. The other part remains in the code but is a result of an intentional decision.
- **Mitigated**: Implemented actions to minimize the impact or likelihood of the risk

Critical Severity Issues

No issues found.

Medium Severity Issues

ME-01 Launchpad: Full Distribution not Guaranteed

Location:

- `clarity/contracts/pool/alex-launchpad-v1-1.clar:258-287`

Launchpad randomness relies on moving positions along a chain of tickets, where the distance of each step is determined by a random number generator, and for each position where it stops, that ticket is determined as a winner. Each random number is the result of a greater number modulus a predefined `max_step`. This predefined value is general for each IDO and its formula (simplified for this explanation) is:

$$max\_step = \frac{3 \cdot registered}{2 \cdot winners}$$

Where `registered` is the number of tickets registered in the IDO by the players, and `winners` is the amount of winner tickets the distribution can have.

As a consequence, full distribution is only guaranteed when $\frac{registered}{winners} \leq \frac{2}{3}$ because `max_step` results in one or less than one, and therefore every player wins.

On the other hand, if that relation is not satisfied, fewer tickets might be distributed. As a clear example, if there were only one winner ticket available and only one ticket registered, `max_step` would be equal to 1.5. The player would win if the position after the step falls between 0 and 1. Therefore, this player has a probability of winning of just 66.67% in a single-player IDO.

Recommendation

If there are remaining tickets, a new round needs to be initiated, with the `max_step` value adjusted to this amount.

Status

Acknowledged. The recommendation will be incorporated into the following IDOs.

Minor Severity Issues

MI-01 Launchpad: Tokens Locked due to Incomplete Distribution

Location:

- `clarity/contracts/pool/alex-launchpad-v1-1.clar:258-287`

If the tokens are not distributed because of the issue described in ME-01, they will be locked in the contract unless the contract owner calls `transfer-all-to-owner()`, which will transfer the tokens back to the owner.

Recommendation

If ME-01 remains unresolved, the IDO owner should be able to extract those tokens.

Status

Acknowledged. The development team considered the conditions of this issue unlikely to be satisfied.

Enhancements

These items do not represent a security risk. They are best practices that we suggest implementing.

Table

ID	Title	Status
EN-01	Launchpad: Unnecessary Assertions	Not implemented
EN-02	Yield Vault: Misleading Response	Not implemented

Details

EN-01 Launchpad: Unnecessary Assertions

Location:

- `clarity/contracts/pool/alex-launchpad-v1-1.clar:269,353`

A block's VRF seed can be taken from a block already mined, a past block. The seed used for the IDOs is from the block after the registration ends. However, the assertions in lines 269 and 353 checks $blockHeight \geq registrationEndHeight$, while it can also fail because the seed cannot be taken if current block height is at registration end or the block after it.

Moreover, the assertion in `claim-process()` will never fail because `get-last-claim-walk-position()` is called before, and it will revert if the seed is not available.

Status

Not implemented. The corresponding fixes are planned to be implemented in the following iterations.

EN-02 Yield Vault: Misleading Response

Location:

- `clarity/contracts/pool/yield-vault-alex.clar:161`

If `activated` is equal to false, the response will be `(ok true)`, even when only the rewards are claimed, and they are not staked.

Status

Not implemented. The corresponding fixes are planned to be implemented in the following iterations.

Other Considerations

The considerations stated in this section are not right or wrong. We do not suggest any action to fix them. But we consider that they may be of interest for other stakeholders of the project, including users of the audited contracts, owners or project investors.

Centralization

Launchpad's claiming and refunding functions are public and do not have restrictions to be called by the users, when a grace period is passed. However, the claiming functions require a list of winners ordered in sequence as an input. As a consequence, if the off-chain task cannot be run, users will need to figure out the sequence and pay the cost of, at least, the claiming of every user before them.

The refunding functions need the claiming function to be called before because refund is only available when it is confirmed on-chain that the ticket is not a winner ticket.

Moreover, launchpad contract owner can transfer all the tokens in the contract's balance.

Changelog

- 2022-04-08 – Initial report based on commit `e268fd53370be3a271625bd45523ae07cb1239ac`.
- 2022-04-19 – Final report based on feedback provided by the development team.

Disclaimer: This audit report is not a security warranty, investment advice, or an approval of the AlexGo project since CoinFabrik has not reviewed its platform. Moreover, it does not provide a smart contract code faultlessness guarantee.