Kanalabs aggregator Smart Contract

Audit Report

contact@bitslab.xyz

https://twitter.com/movebit_

Thu Sep 07 2023

Kanalabs aggregator Smart Contract Audit Report

1 Executive Summary

1.1 Project Information

Description	A swap aggregator
Type	DeFi
Auditors	MoveBit
Timeline	Tue Sep 05 2023 - Thu Sep 07 2023
Languages	Move
Platform	Sui
Methods	Architecture Review, Unit Testing, Manual Review
Source Code	https://github.com/kanalabs/aggregator-contracts
Commits	92184676d5c3046312439f7b0f746aacdfdaaddc 9bf7f5583aeed88b4763799f30f3f08ac09936d0

1.2 Files in Scope

The following are the SHA1 hashes of the original reviewed files.

ID	File	SHA-1 Hash
MOV	aggregator/sui/Move.toml	54f4d5fdb46bb58e648b2d7d9fbd1c7c27783d8a
UTI	aggregator/sui/sources/utils.move	d563f3a86ba11a44c280b2c5f473b5db2740e27f
SAG	aggregator/sui/sources/swap_aggregator.move	ec74c5bac3120f753902de4f900d442dbad53b5e

1.3 Issue Statistic

Item	Count	Fixed	Acknowledged
Total	2	2	0
Informational	0	0	0
Minor	2	2	0
Medium	0	0	0
Major	0	0	0
Critical	0	0	0

1.4 MoveBit Audit Breakdown

MoveBit aims to assess repositories for security-related issues, code quality, and compliance with specifications and best practices. Possible issues our team looked for included (but are not limited to):

- Transaction-ordering dependence
- Timestamp dependence
- Integer overflow/underflow by bit operations
- Number of rounding errors
- Denial of service / logical oversights
- Access control
- Centralization of power
- Business logic contradicting the specification
- Code clones, functionality duplication
- Gas usage
- Arbitrary token minting
- Unchecked CALL Return Values
- The flow of capability
- Witness Type

1.5 Methodology

The security team adopted the **"Testing and Automated Analysis"**, **"Code Review"** and **"Formal Verification"** strategy to perform a complete security test on the code in a way that is closest to the real attack. The main entrance and scope of security testing are stated in the conventions in the "Audit Objective", which can expand to contexts beyond the scope according to the actual testing needs. The main types of this security audit include:

(1) Testing and Automated Analysis

Items to check: state consistency / failure rollback / unit testing / value overflows / parameter verification / unhandled errors / boundary checking / coding specifications.

(2) Code Review

The code scope is illustrated in section 1.2.

(3) Formal Verification

Perform formal verification for key functions with the Move Prover.

(4) Audit Process

- Carry out relevant security tests on the testnet or the mainnet;
- If there are any questions during the audit process, communicate with the code owner in time. The code owners should actively cooperate (this might include providing the latest stable source code, relevant deployment scripts or methods, transaction signature scripts, exchange docking schemes, etc.);
- The necessary information during the audit process will be well documented for both the audit team and the code owner in a timely manner.

2 Summary

This report has been commissioned by Kanalabs to identify any potential issues and vulnerabilities in the source code of the aggregator smart contract, as well as any contract dependencies that were not part of an officially recognized library. In this audit, we have utilized various techniques, including manual code review and static analysis, to identify potential vulnerabilities and security issues.

During the audit, we identified 2 issues of varying severity, listed below.

ID	Title	Severity	Status
SAG-1	Redundant Abilities of Event Object	Minor	Fixed
SAG-2	Unused Constant	Minor	Fixed

3 Participant Process

Here are the relevant actors with their respective abilities within the aggregator Smart Contract :

User

- User can initiate swap coins through `initiate_swap()`.
- User can terminate swap coins through `terminate_swap()`.
- User can swap `X` coins to `Y` coins through `cetus_swap_x_to_y()`.
- User can swap `Y` coins to `X` coins through `cetus_swap_y_to_x()`.

4 Findings

SAG-1 Redundant Abilities of Event Object

Severity: Minor

Status: Fixed

Code Location:

aggregator/sui/sources/swap_aggregator.move#24,30

Descriptions:

Only `copy` and `drop` abilities are needed for event objects, so the `store` is redundant.

Suggestion:

It is recommended to modify the abilities of `InitiateSwapEvent` and `TerminateSwapEvent`.

Resolution:

The client followed our suggestion and fixed this issue.

SAG-2 Unused Constant

Severity: Minor

Status: Fixed

Code Location:

aggregator/sui/sources/swap_aggregator.move#13,14

Descriptions:

The constant `CETUX_DEX` and `TURBOS_DEX` are not used in the contract.

Suggestion:

It is recommended to remove the unused constants if there's no further use.

Resolution:

The client followed our suggestion and fixed this issue.

Appendix 1

Issue Level

- **Informational** issues are often recommendations to improve the style of the code or to optimize code that does not affect the overall functionality.
- **Minor** issues are general suggestions relevant to best practices and readability. They don't post any direct risk. Developers are encouraged to fix them.
- **Medium** issues are non-exploitable problems and not security vulnerabilities. They should be fixed unless there is a specific reason not to.
- **Major** issues are security vulnerabilities. They put a portion of users' sensitive information at risk, and often are not directly exploitable. All major issues should be fixed.
- **Critical** issues are directly exploitable security vulnerabilities. They put users' sensitive information at risk. All critical issues should be fixed.

Issue Status

- **Fixed:** The issue has been resolved.
- **Partially Fixed:** The issue has been partially resolved.
- **Acknowledged:** The issue has been acknowledged by the code owner, and the code owner confirms it's as designed, and decides to keep it.

Appendix 2

Disclaimer

This report is based on the scope of materials and documents provided, with a limited review at the time provided. Results may not be complete and do not include all vulnerabilities. The review and this report are provided on an as-is, where-is, and as-available basis. You agree that your access and/or use, including but not limited to any associated services, products, protocols, platforms, content, and materials, will be at your own risk. A report does not imply an endorsement of any particular project or team, nor does it guarantee its security. These reports should not be relied upon in any way by any third party, including for the purpose of making any decision to buy or sell products, services, or any other assets. TO THE FULLEST EXTENT PERMITTED BY LAW, WE DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, IN CONNECTION WITH THIS REPORT, ITS CONTENT, RELATED SERVICES AND PRODUCTS, AND YOUR USE, INCLUDING BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NOT INFRINGEMENT.