MOVEBIT

Vimverse
Smart Contract
Audit Report

contact@movebit.xyz

https://twitter.com/movebit_

05/17/2023

Vimverse Smart Contract Audit Report

1 Executive Summary

1.1 Project Information

Description	Vimverse is an innovative platform built on the Sui Network leveraging the potential of enhanced decentralized reserve currency protocol, providing multiple opportunities and value across various ecosystems.
Type	DeFi
Auditors	MoveBit
Timeline	May 10, 2023 – May 17, 2023
Languages	Move
Platform	Sui
Methods	Architecture Review, Unit Testing, Manual Review
Source Code	https://github.com/vimverse/vimverse-protocol-move

Commits	1a8fd1b83a1f1ac7b86e504db1e7946196623dc8 379063bde0d861bc58fee9265ddc96be2c7b6f29 bf01ff65857dcc15542db58b4bff499410765ffb

1.2 Files in Scope

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

ID	Files	SHA-1 Hash
BD4	vimverse-protocol-move/sources/bond_depository44.move	795874325eb24f224f84b5c00f36fe4815333b2e
MATH	vimverse-protocol-move/sources/math.move	b93d6892acdaa71492b525848ab6b0b0bccb7552
STD	vimverse-protocol-move/sources/staking_distributor.move	35d45816e59df064598be780dba2ad32a29ca8f0
STK	vimverse-protocol-move/sources/staking.move	7712b337896019b0542c70f5c53e36c00479d353
SVM	vimverse-protocol-move/sources/svim.move	b51d0bb825578e67391ca1866b1e651ad1c4df08
TSY	vimverse-protocol-move/sources/treasury.move	5bf85917545a96b9471abc42f3a961ddf88a491b
VIM	vimverse-protocol-move/sources/vim.move	b25d58859677bef5566a07e59e9b8cc600de2ca7

1.3 Issue Statistic

Item	Count	Fixed	Acknowledged

Total	7	7	
Informational			
Minor	4	4	
Medium			
Major	2	2	
Critical	1	1	

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 **Vimverse** to identify any potential issues and vulnerabilities in the source code of the **Vimverse-protocol** 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 **7** issues of varying severity, listed below.

ID	Title	Severity	Status

BD4-01	Incorrect Calculation	Critical	Fixed
VIM-02	Centralization Risk	Major	Fixed
BD4-03	Incorrect Judgment	Major	Fixed
VIM-04	Gas Optimization	Minor	Fixed
BD4-05	Unused Constant	Minor	Fixed
VIM-06	Unused Function	Minor	Fixed
BD4-07	Missing Emit Event	Minor	Fixed

3 Participant Process

Here are the relevant actors with their respective abilities within the `Vimverse-protocol` Smart Contract：

Admin

- Admin can initialize `Terms<T>` and `BondDepository44<T>` through `init_bond()`.
- Admin can initialize `staking.config` through `initialize()`.
- Admin can set the deposit token through `set_deposit_token()`.
- Admin can modify the `StakingDistributorConfig` 's `rate` and `adjust` through `set_rate()` and `set_adjust()`.
- Admin can modify the total_reserves of `treasury` through `set_total_reserves()`.
- Admin can modify the member variables of `Terms` through `set_bond_terms()`、`set_adjust()`、`set_dao()`、`set_pause()`.
- Admin can set the manager of the `DepoistToken` through `set_mamager()`.
- Admin can pause `Staking` through `set_pause()`.
- Admin can create a new `TreasuryLock` through `new_lock()`.

Manager

- Manager can withdraw the `DepoistToken` from `Treasury` through `withdraw()` and `manage()`.

User

- User can deposit and redeem through `deposit()` and `redeem()`.

- User can stake and unstake `Coin<SVIM>` through `stake()` and `unstake()`.
- User can rebase the `Staking` through `rebase()`.

4 Findings

BD4-01 Incorrect Calculation

Severity: Critical

Status: Fixed
Code Location: sources/bond_depository44.move#L301.
Descriptions: In the `else` branch of the `redeem<T>()` function, the update of the `bond.payout` quantity should be subtracted from the payout quantity for this redemption, instead of `bond.payout = bond.payout + payout`.
Suggestion: It is recommended to change the update of `bond.payout` quantity to `bond.payout = bond.payout - payout`.

Resolution: The client followed our suggestion and fixed this issue.

VIM-02 Centralization Risk

Severity: Major

Status: Fixed
Code Location: sources/vim.move#L64, L71.
Descriptions: The administrator can mint any number of tokens to any address, and there is a risk of centralization.
Suggestion: It is recommended that the privileged accounts use multi-signature accounts.

Resolution: The client followed our suggestion and fixed this issue.

BD4-03 Incorrect Judgment

Severity: Major

Status: Fixed
Code Location: sources/bond_depository44.move#L285.
Descriptions: In the function `redeem<T>()`, the conditional judgment `assert!(table::cont`

`ains(&bond_depositor.user_bond_info, recipient) == false, ENoBond)` is incorrect and will cause the function to fail to execute properly.
Suggestion: It is recommended to modify the conditional judgment to `assert! (table::contains(&bond_depositor.user_bond_info, recipient), ENoBond)` .

Resolution: The client followed our suggestion and fixed this issue.

VIM-04 Gas Optimization

Severity: Minor

Status: Fixed
Code Location: sources/vim.move#L56, L59, L66.
Descriptions: In the `assert!(vec_set::contains(&lock.mint_authorities, &id) == false, EAlreadyExist)` statement, using `==` to judge the Boolean value will increase the gas consumption.
Suggestion: The modification suggestion is as follows: `assert!(!vec_set::contains(&lock.mint_authorities, &id), EAlreadyExist)` .

Resolution: The client followed our suggestion and fixed this issue.

BD4-05 Unused Constant

Severity: Minor

Status: Fixed
Code Location: sources/bond_depository44.move#L84, sources/svim.move#L60.
Descriptions: The constant variables `EInitialized` , `ENotEnough` in `bond_depository44.move` and `svim.move` are not used.
Suggestion: It is recommended that unused constant variables should be removed.

Resolution: The client followed our suggestion and fixed this issue.

VIM-06 Unused Function

Severity: Minor

Status: Fixed
Code Location: sources/vim.move#L90.
Descriptions: There is an unused function of `treasury_cap<T>()` .

Suggestion: It is recommended to delete the unused function.

Resolution: The client followed our suggestion and fixed this issue.

BD4-07 Missing Emit Event

Severity: Minor

Status: Fixed

Code Location: sources/bond_depository44.move#L157, L179, L193, L201.

Descriptions: The smart contract lacks appropriate events for monitoring sensitive operations, which could make it difficult to track important actions or detect potential issues.

Suggestion: It is recommended to emit events for these functions.

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.