Navi Perp Vault

Audit Report

contact@bitslab.xyz

https://twitter.com/movebit_

Fri Feb 28 2025

Navi Perp Vault Audit Report

1 Executive Summary

1.1 Project Information

Description	It is a token deposit and withdrawal system intended for centralized systems.
Type	DeFi
Auditors	MoveBit
Timeline	Thu Feb 06 2025 - Wed Feb 12 2025
Languages	Move
Platform	Sui
Methods	Architecture Review, Unit Testing, Manual Review
Source Code	https://github.com/naviprotocol/perp-vault-contract
Commits	11a53eafdda0778ab422e591a375a06b11959588 446ed9168040a8f50f7a35878587f91b3794a37d

1.2 Files in Scope

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

ID	File	SHA-1 Hash
DMA	sources/dvault_manage.move	25ddecf9a999a32ea56d36ec1b94554d2e2ac1b3
UEN	sources/user_entry.move	715df01448ce797b486cccc3bb8df2c2197822ac
AVA	sources/active_vault.move	e072abb4359298c9a01bbe918cc094726978823c
SVA	sources/secure_vault.move	2766a5be5c353d719cfcd9c58eb20aa987ad9dbe

1.3 Issue Statistic

Item	Count	Fixed	Acknowledged
Total	5	4	1
Informational	0	0	0
Minor	3	3	0
Medium	1	1	0
Major	1	0	1
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(Optional)

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 Navi to identify any potential issues and vulnerabilities in the source code of the Navi Perp Vault 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 5 issues of varying severity, listed below.

ID	Title	Severity	Status
AVA-1	Centralization Risk	Major	Acknowledged
AVA-2	Missing Re - initialization of `config.signers` in `reset_signers()` Method	Medium	Fixed
AVA-3	Minor Version Comparison Adjustment in `version_migration()` Method	Minor	Fixed
AVA-4	Minor Enhancement Suggestion for `withdraw()` Method Signature	Minor	Fixed
DMA-1	Lack of Events Emit	Minor	Fixed

3 Participant Process

Here are the relevant actors with their respective abilities within the Navi Perp Vault Smart Contract :

Owner

- Owner can call `create_owner_cap` to create new OwnerCap.
- Owner can call `create_pause_cap` to create pause control capability.
- Owner can call `add_signer` / `remove_signer` / `reset_signers` to manage signers.
- Owner can call `set_pause` to pause contract operations.
- Owner can call `freeze_pause_cap` to freeze pause capabilities.
- Owner can call `version_migration` to perform version upgrades.
- Owner can call `create_vault` to create new vault instances.
- Owner can call `admin_withdraw_secure_vault` to directly withdraw from secure vault.
- Owner can call `set_operator_epoch_max_amount` to configure withdrawal limits.
- Owner can call `operator_withdraw_secure_vault` (with operator cap) for secure withdrawals.

User

- Users can call `deposit` to deposit funds into active vault.
- Users can call `withdraw` to withdraw funds with signature verification.

4 Findings

AVA-1 Centralization Risk

Severity: Major

Status: Acknowledged

Code Location:

sources/active_vault.move;

sources/secure_vault.move

Descriptions:

Centralization risk was identified in the smart contract:

- The owner can update the config of the vaults.
- The owner can withdraw assets from active vaults and secure vaults.

Suggestion:

It is recommended that measures be taken to reduce the risk of centralization, such as a multi-signature mechanism.

Resolution:

NAVI's comment: The Active vault is protected by multiple independent signers, while the Secure vault's capabilities are managed by NAVI team's multi-signature system.

AVA-2 Missing Re - initialization of `config.signers` in `reset_signers()` Method

Severity: Medium

Status: Fixed

Code Location:

sources/active_vault.move#289

Descriptions:

In the `reset_signers()` method at line #289 of the `active_vault.move` file, the `config.signers` field is not re - initialized. When this method is called to reset the signers, the new signers are simply pushed onto the existing `config.signers` vector. This can lead to the presence of old and potentially invalid signers in the list, which may cause unexpected behavior in the contract's signature verification and authorization processes.

Suggestion:

Before adding the new signers to the `config.signers` vector, clear the existing contents of the vector.

Resolution:

This issue has been fixed. The client has adopted our suggestions.

AVA-3 Minor Version Comparison Adjustment in `version_migration()` Method

Severity: Minor

Status: Fixed

Code Location:

sources/active_vault.move#317

Descriptions:

In the `active_vault.move` file at line #317, within the `version_migration()` method, there is a minor issue with the version comparison. Currently, the comparison is set as `config.version <= VERSION`. To ensure more accurate version migration and avoid potential incorrect migrations when the version is equal, it should be changed to `config.version < VERSION`.

Suggestion:

It is recommended to replace the existing `config.version <= VERSION` with `config.version < VERSION`. This adjustment will make the version migration logic more precise and reliable.

Resolution:

This issue has been fixed. The client has adopted our suggestions.

AVA-4 Minor Enhancement Suggestion for `withdraw()` Method Signature

Severity: Minor

Status: Fixed

Code Location:

sources/active_vault.move#207

Descriptions:

In the `active_vault.move` file at line #207, for the `withdraw()` method, adding a restriction `signatures.length() <= config.signers.length()` to the method signature can enhance the method's integrity.

Suggestion:

Add the condition `assert!(signatures.length() <= config.signers.length(), "Number of signatures exceeds number of configured signers");` at the beginning of the method.

Resolution:

This issue has been fixed. The client has adopted our suggestions.

DMA-1 Lack of Events Emit

Severity: Minor

Status: Fixed

Code Location:

sources/dvault_manage.move;

sources/active_vault.move;

sources/secure_vault.move

Descriptions:

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

Suggestion:

It is recommended to emit events for the function.

Resolution:

This issue has been fixed. The client has adopted our suggestions.

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.