Mango Network

Audit Report

contact@bitslab.xyz

https://twitter.com/movebit_

Thu Feb 06 2025

Mango Network Audit Report

1 Executive Summary

1.1 Project Information

Description	Layer 1 blockchains powered by Move enable the construction of a transactional, fully on-chain infrastructure network, allowing developers to build super DApps effortlessly.
Type	L1
Auditors	MoveBit
Timeline	Mon Dec 30 2024 - Fri Jan 24 2025
Languages	Move, Rust, Solidity
Platform	Others
Methods	Architecture Review, Unit Testing, Manual Review
Source Code	https://github.com/movebit/ClientProjects
Commits	e5d999cca86d5a50d871984bffa152f8e2a35a42 fe248da19d671f035d66609458d4c2a03570e886

1.2 Files in Scope

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

ID	File	SHA-1 Hash
MOV52	crates/mgo-framework/packages/mgo-inscription/Move.toml	df87b36a41a18672975c18bd6e4f082629f76531
COI1	crates/mgo-framework/packages/mgo-inscription/sources/coinscription.move	b3a068ecf04ed9a2b91816688afd72454783ffd4
SVG	crates/mgo-framework/packages/mgo-inscription/sources/svg.move	515dc40e68fd29d1a0ea74bb5e28506a73fe2bef
INS	crates/mgo-framework/packages/mgo-inscription/sources/inscription.move	6a7e82f4307cdf27477b3e0ea5281c08cda49998
SUT	crates/mgo-framework/packages/mgo-inscription/sources/string_util.move	0bd843e54f20f829e5fdd69cd730208dbf4ab9a5
SIN	crates/mgo-framework/packages/mgo-inscription/sources/singlescription.move	41228f958f3fcc77cab4a83b87c63ac0384bd245

1.3 Issue Statistic

Item	Count	Fixed	Acknowledged
Total	2	2	0
Informational	0	0	0
Minor	1	1	0
Medium	1	1	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(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 Mango to identify any potential issues and vulnerabilities in the source code of the Mango Network 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
SUT-1	Multiple Issues in `sting_util.move` Handling Empty Strings	Medium	Fixed
SVG-1	Potential XSS Risks Caused by SVG Image	Minor	Fixed

3 Participant Process

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

Mango Chain User

- Users can create Mango Accounts.
- Users can send transactions.

Full Node

- Mango Full nodes validate blockchain activities, including transactions, checkpoints, and epoch changes. Each Full node stores and services the queries for the blockchain state and history.

Validator

- Validators on the Mango network run special nodes and have additional tasks and responsibilities beyond those of Full node operators.Validators can vote on the proposal.

`mgo-inscription` **module**

- 3th-party contracts or users can create and share `TickRecord` through `coinscription_new_tick()` .
- 3th-party contracts or users can mint coin throuth `coinscription_mint()`
- 3th-party contracts or users can transfer coin through `coinscription_transfer()`
- 3th-party contracts or users can split coin through `coinscription_split()`
- 3th-party contracts or users can merge coin through `coinscription_merge()`
- 3th-party contracts or users can burn coin through `coinscription_burn()`
- 3th-party contracts or users can crate NFT copyright through `singlescription_new_copyright()`
- 3th-party contracts or users can mint NFT through `singlescription_mint()`
- 3th-party contracts or users can transfer NFT through `singlescription_transfer`
- 3th-party contracts or users can burn NFT through `singlescription_burn`

Wallet App

- Wallet App can create mnemonic phrases, generate public and private key pairs, send transactions, and view transactions.

4 Findings

SUT-1 Multiple Issues in `sting_util.move` Handling Empty Strings

Severity: Medium

Status: Fixed

Code Location:

crates/mgo-framework/packages/mgo-inscription/sources/string_util.move#1-163

Descriptions:

The `sting_util.move` is a system library for handling strings.

In the `index_of` function, when both `input` and `search` are empty strings, it returns (false, 0), but it should return (true, 0).

In the `contains_any` function, when both `input` and `chars` are empty strings, it returns false, but it should return true.

Suggestion:

1. It is recommended to modify the handling logic for the above special cases.
2. It is recommended to set the access permissions for all functions in this module to "public(friend)" to prevent other contracts from calling them.

Resolution:

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

SVG-1 Potential XSS Risks Caused by SVG Image

Severity: Minor

Status: Fixed

Code Location:

crates/mgo-framework/packages/mgo-inscription/sources/svg.move#16

Descriptions:

The `generate_coinscription_svg` and `generate_singlescription_svg` functions are used to assemble a SVG image.

1. Since the `svg.move` module is a system library, third-party contract developers are likely to call the functions from this module.
2. When assembling the SVG image, XSS injection is not considered.
3. If a third-party contract calls this SVG library, an attacker could insert JavaScript code into the SVG, which may run in the front-end of a DApp.

Suggestion:

It is recommended to set the access permissions for all functions in this module to "public(friend)" to prevent other contracts from calling them.

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.