AUDITMOS

Secure your space

LUNAVERSE

audit / code review report

February 22, 2022

TABLE OF CONTENTS

contact@auditmos.com

AUDITMOS

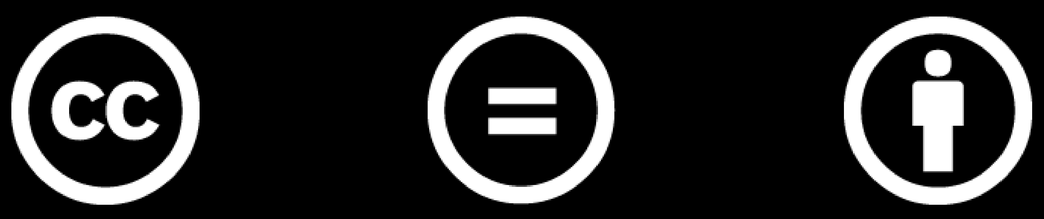
Secure your space

LUNAVERSE

audit / code review report

February 22, 2022

LICENSE



contact@auditmos.com

AUDITMOS

Secure your space

LUNAVERSE

audit / code review report

February 22, 2022

DISCLAIMER

contact@auditmos.com

AUDITMOS

Secure your space

LUNAVERSE

audit / code review report

February 22, 2022

APPROACH AND METHODOLOGY

PURPOSE

1. Determine the correct operation of the protocol, according to the design specification.
2. Identify possible vulnerabilities that could be exploited by an attacker.
3. Detect errors in the smart contract that could lead to unexpected behavior.
4. Analyze whether best practices were followed during development.
5. Make recommendations to improve security and code readability.

CODEBASE

Repository	https://github.com/Lunaverse-Official/smart-contracts/tree/main/Contracts/contracts
Branch	main
Commit hash	fda6fc8daa26a039c2bb4d3e4cb819f138a4feeb

METHODOLOGY

1. Reading the available documentation and understanding the code.
2. Doing automated code analysis and reviewing dependencies.
3. Checking manually source code line by line for security vulnerabilities.
4. Following guildlines and recommendations.
5. Preparing this report.

contact@auditmos.com

AUDITMOS

Secure your space

LUNAVERSE

audit / code review report

February 22, 2022

DESCRIPTION

Issues Categories:

Severity	Description
CRITICAL	vulnerability that can lead to loss of funds, failure to recover blocked funds, or catastrophic denial of service.
HIGH	vulnerability that can lead to incorrect contract state or unpredictable operation of the contract.
MEDIUM	failure to adhere to best practices, incorrect usage of primitives, without major impact on security.
LOW	recommendations or potential optimizations which can lead to better user experience or readability.

Each issue can be in the following state:

State	Description
PENDING	still waiting for resolving
ACKNOWLEDGED	know but not planned to resolve for some reasons
RESOLVED	fixed and deployed

contact@auditmos.com

AUDITMOS

Secure your space

LUNAVERSE

audit / code review report

February 22, 2022

AUDIT SCOPE

1. getting to know the project	✓
2. research into architecture	✓
3. manual code read	✓
4. check of permissions	✓
5. identify common Rust vulnerabilities	✓
6. test coverage	✓
7. static analysis	✓

contact@auditmos.com

AUDITMOS

Secure your space

LUNAVERSE

audit / code review report

February 22, 2022

FINDINGS

Finding	Severity	Status
#1 - withdrawing funds from staking is not possible	MEDIUM	RESOLVED
#2 - improve tests code coverage	LOW	RESOLVED
#3 - cover with test all private methods and check if unauthorized exception is thrown	LOW	RESOLVED

contact@auditmos.com

AUDITMOS

Secure your space

LUNAVERSE

audit / code review report

February 22, 2022

#1 - WITHDRAWING FUNDS FROM STAKING IS NOT POSSIBLE

Severity	Status
MEDIUM	RESOLVED

Current implementation of staking smart contract not allow to withdraw collected fees.

Staking contract is flexible and allow configure fees in easy way but there is a lack of function which will allow contract admin to withdraw these fees.

RECOMMENDATION

It is recommended to add a function which should be available only for admin and will allow to withdraw collected fees to external wallet.

Sample implementation:

ExecuteMsg::FeesWithdraw { amount } => fees_withdraw(deps, env, info, amount)

PROOF OF SOURCE

https://github.com/Lunaverse-Official/smart-contracts/blob/main/Contracts/contracts/contracts/staking/src/contract.rs#L115

contact@auditmos.com

LUNAVERSE

audit / code review report

February 22, 2022

#2 - IMPROVE TESTS CODE COVERAGE

Severity	Status
LOW	RESOLVED

Test Coverage is an important indicator of software quality and an essential part of software maintenance. It helps in evaluating the effectiveness of testing by providing data on different coverage items. It is a useful tool for finding untested parts of a code base. Test coverage is also called code coverage in certain cases.

Test coverage can help in monitoring the quality of testing and assist in directing the test generators to create test cases that cover areas that have not been tested. It helps in determining a quantitative measure of Test coverage, which is an indirect measure of quality and identifies redundant test cases that do not increase coverage.

RECOMMENDATION

It is highly recommended to test all of the functions and have high ratio of test coverage. It is recommended to use code coverage reporting tool for the Cargo build system for example cargo-tarpaulin.

PROOF OF SOURCE

https://github.com/Lunaverse-Official/smart-contracts/tree/main/Contracts/contracts/contracts/staking/src/testing

https://github.com/Lunaverse-Official/smart-contracts/tree/main/Contracts/contracts/contracts/vesting/src/testing

AUDITMOS

Secure your space

LUNAVERSE

audit / code review report

February 22, 2022

#3 - COVER WITH TEST ALL PRIVATE METHODS AND CHECK IF UNAUTHORIZED EXCEPTION IS THROWN

Severity	Status
LOW	RESOLVED

In the current contract implementation not all functions with restricted access to contract owner are covered by tests. It might lead to some unpredictable behaviour during this function call by some address which is not owner address.

RECOMMENDATION

It is highly recommended to test all of the functions which should have restriceted access and/or should be called only by contract owner.

PROOF OF SOURCE

https://github.com/Lunaverse-Official/smart-contracts/tree/main/Contracts/contracts/contracts/staking/src/testing

https://github.com/Lunaverse-Official/smart-contracts/tree/main/Contracts/contracts/contracts/vesting/src/testing

contact@auditmos.com

auditmos.com

Secure your space

contact@auditmos.com