AUDITMOS

Secure your space

LOOP

audit / code review report

October 10, 2021

TABLE OF CONTENTS

contact@auditmos.com

AUDITMOS

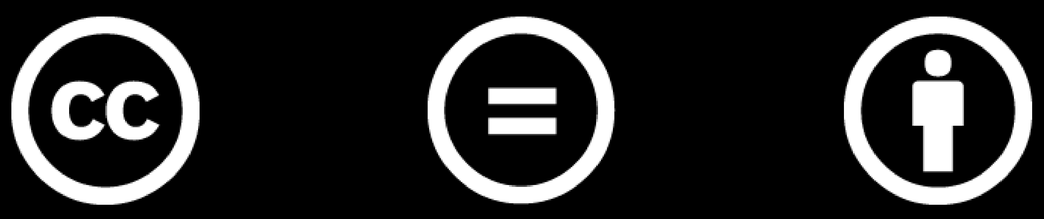
Secure your space

LOOP

audit / code review report

October 10, 2021

LICENSE



contact@auditmos.com

AUDITMOS

Secure your space

LOOP

audit / code review report

October 10, 2021

DISCLAIMER

contact@auditmos.com

AUDITMOS

Secure your space

LOOP

audit / code review report

October 10, 2021

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/Loop-Protocol/Loop_protocol_col5/tree/staking-contract/contracts/loopswap_staking
Branch	staking-contract
Commit hash	b7936d39e392634294ec67b2e9811ea8c2b844f8

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.

AUDITMOS

Secure your space

LOOP

audit / code review report

October 10, 2021

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

LOOP

audit / code review report

October 10, 2021

FINDINGS

Finding	Severity	Status
#1 - Length of key in stakeable_info incorrect	HIGH	RESOLVED
#2 - execute_distribute function is available to everyone	HIGH	RESOLVED
#3 - function always returns success	LOW	RESOLVED
#4 - id not checked in reply function	LOW	RESOLVED
#5 - typo in response	LOW	RESOLVED

contact@auditmos.com

AUDITMOS

Secure your space

LOOP

audit / code review report

October 10, 2021

Severity	Status
HIGH	RESOLVED

#1 - LENGTH OF KEY IN STAKEABLE_INFO INCORRECT

It seems that there is a bug in item key as 9 is not the length of stakeable_info

Current code:

```
pub const STAKEABLE_INFO: Item<StakeableInfoRaw> =
Item::new("\u{0}\u{9}stakeable_info");
```

RECOMMENDATION

Length prefixed key is required only in older version of Terra blockchain (and to keep backward compatibility), if contracts were not deployed to Columbus-4, it is recommended to remove length prefix completely, if migration to Columbus-5 is needed with that code, it is suggested to use to_length_prefixed.

Current version of the code is not backward compatible with Columbus-4.

```
use cosmwasm_storage::to_length_prefixed;
Item::new(to_length_prefixed(b"stakeable_info"));
```

PROOF OF SOURCE

https://github.com/Loop-Protocol/Loop_protocol_col5/blob/b7936d39e392634294ec67b2e9811ea8c2b844f8/contracts/loopswap_staking/src/state.rs#L21

contact@auditmos.com

AUDITMOS

Secure your space

LOOP

audit / code review report

October 10, 2021

#2 - EXECUTE_DISTRIBUTE FUNCTION IS AVAILABLE TO EVERYONE

Severity	Status
HIGH	RESOLVED

In execute_distribute you there is no check who should be able to call this function, so everyone is able to distribute tokens.

RECOMMENDATION

If such behaviour is not intended it is highly recommended to use some kind of authorization pattern.

PROOF OF SOURCE

https://github.com/Loop-Protocol/Loop_protocol_col5/blob/b7936d39e392634294ec67b2e9811ea8c2b844f8/contracts/loopswap_staking/src/contract.rs#L295

contact@auditmos.com

LOOP

audit / code review report

October 10, 2021

#3 - FUNCTION ALWAYS RETURNS SUCCESS

This function always returns success even if user tries to cheat and withdraw more tokens that possible.

Severity	Status
LOW	RESOLVED

```
if amount <= lp_provided {
```

RECOMMENDATION

It is not a security issue.

PROOF OF SOURCE

https://github.com/Loop-Protocol/Loop_protocol_col5/blob/b7936d39e392634294ec67b2e9811ea8c2b844f8/contracts/loopswap_staking/src/contract.rs#L230

AUDITMOS

Secure your space

LOOP

audit / code review report

October 10, 2021

#4 - ID NOT CHECKED IN REPLY FUNCTION

Reply Id is not checked.

Severity	Status
LOW	RESOLVED

RECOMMENDATION

It is recommended to confirm that Id is correct.

PROOF OF SOURCE

https://github.com/Loop-Protocol/Loop_protocol_col5/blob/b7936d39e392634294ec67b2e9811ea8c2b844f8/contracts/loopswap_staking/src/contract.rs#L430

contact@auditmos.com

AUDITMOS

Secure your space

LOOP

audit / code review report

October 10, 2021

#5 - TYPO IN RESPONSE

There is a typo in reponse object.

Severity	Status
LOW	RESOLVED

```
Ok(Response::new().add_attribute("last_distributed", "last_distributed"))
```

RECOMMENDATION

Use

```
Ok(Response::new().add_attribute("last_distributed", last_distributed))
```

PROOF OF SOURCE

https://github.com/Loop-Protocol/Loop_protocol_col5/blob/b7936d39e392634294ec67b2e9811ea8c2b844f8/contracts/loopswap_staking/src/contract.rs#L322

contact@auditmos.com

auditmos.com

Secure your space

contact@auditmos.com