code4rena

Moxie

Smart Contract Security Assessment

Version 1.0

Audit dates: Jul 11 — Jul 12, 2024

Audited by: Alex the Entreprenerd
Hickuphh3

Contents

1. Introduction

1.1 About Zenith

Zenith is an offering by Code4rena that provides consultative audits from the very best security researchers in the space. We focus on crafting a tailored security team specifically for the needs of your codebase.

Learn more about us at https://code4rena.com/zenith.

1.2 Disclaimer

This report reflects an analysis conducted within a defined scope and time frame, based on provided materials and documentation. It does not encompass all possible vulnerabilities and should not be considered exhaustive.

The review and accompanying report are presented on an "as-is" and "as-available" basis, without any express or implied warranties.

Furthermore, this report neither endorses any specific project or team nor assures the complete security of the project.

1.3 Risk Classification

SEVERITY LEVEL	IMPACT: HIGH	IMPACT: MEDIUM	IMPACT: LOW
Likelihood: High	Critical	High	Medium
Likelihood: Medium	High	Medium	Low
Likelihood: Low	Medium	Low	Low

2. Executive Summary

2.1 About Moxie

Moxie is an orchestration of several smart contracts that can be executed via Frames, Actions and Apps/Clients. It represents foundational technology that anyone can use to add economic incentives to their Farcaster experience.

2.2 Scope

Repository	moxie-protocol/contracts/tree/feat/token-vesting
Commit Hash	f9763f923a51b70f54a8d08a65171cd4f37fb6c5

2.3 Audit Timeline

DATE	EVENT
Jul 11, 2024	Audit start
Jul 12, 2024	Audit end
Nov 28, 2024	Report published

2.4 Issues Found

SEVERITY	COUNT
Critical Risk	0
High Risk	0
Medium Risk	0
Low Risk	1
Informational	0
Total Issues	1

3. Findings Summary

ID	DESCRIPTION	STATUS
L-1	Delta logic could be sidestepped if address that can pull tokens is added	Resolved

4. Findings

4.1 Low Risk

A total of 1 low risk findings were identified.

[L-1] Delta logic could be sidestepped if address that can pull tokens is added

Severity: Low	Status: Resolved

Impact

```
        require(_tokenDestinations.add(_dst), "Destination already
added");
```

Risk of sidestep if a contract that can pull tokens is added

Explanation

Since the `fallback` uses a delta balances

```
    uint256 diff = oldBalance.sub(newBalance);
    usedAmount = usedAmount.add(diff);
```

Pulling tokens that are approved will completely sidestep this mechanism

And will allow moving all tokens before they are vested

NOTE

This will be safe when using `MoxieBondingCurve` and `SubjectFactory` and `EasyAuction` as the only `_tokenDestinations`

Additional instances

The tokenManager could also offer the same vector

```
        address subjectToken = tokenManager.tokens(_subject);
```

Moxie: Acknowledged