PancakeSwap OFT

Audit

Presented by:

OtterSec contact@osec.io

Robert Chen r@osec.io

Shiva Shankar sh1v@osec.io

Contents

Appendices

01 | Executive Summary

Overview

PancakeSwap engaged OtterSec to perform an assessment of the `oft` program. This assessment was conducted between December 5th and December 12th, 2022.

Critical vulnerabilities were communicated to the team prior to the delivery of the report to speed up remediation. After delivering our audit report, we worked closely with the team to streamline patches and confirm remediation. We delivered final confirmation of the patches December 20th, 2022.

Over the course of this audit engagement, we produced 2 findings total, making minor recommendations around discrepancies in token supply tracking.

Overall, the Pancake Swap team was responsive to feedback and great to work with.

02 | Scope

The source code was delivered to us in a git repository at github.com/chefcooper/aptos-contracts/ and github.com/pancakeswap/cake-oft. This audit was performed against commit `c6f896c` and `567b315` respectively.

A brief description of the programs is as follows.

Name	Description
cake-oft	EVM OFT built on top of LayerZero
pancake-cake-oft	Aptos OFT built on top of LayerZero

03 | General Findings

Here we present a discussion of general findings during our audit. While these findings do not present an immediate security impact, they represent antipatterns and could lead to security issues in the future.

ID	Description
OS-OFT-SUG-00	Track supply limits via u128
OS-OFT-SUG-01	Potential overreliance on admin-gated withdrawals

OS-OFT-SUG-00 | u128 Token Supply

Description

Aptos token standard allows for tokens with supply greater than u64. If you want to support this behavior, it could make sense to change the supply limits to u128s.

Note that the underlying LayerZero implementation will only allow for u64 transfers at a time.

Remediation

Change supply limits to u128.

OS-OFT-SUG-01 | Centralization Risk

Description

The current implementation uses `fallbackWithdraw` to handle dropped messages on the Aptos side. This is used to prevent messages from blocking the LayerZero pipe on the Aptos side. While acceptable, this pattern exposes a potentially significant attack surface if the operator key was compromised.

We've seen recent similar compromises with various protocols and thought this was worth raising.

Remediation

Ensure proper access control on the operator key.

A | Vulnerability Rating Scale

We rated our findings according to the following scale. Vulnerabilities have immediate security implications. Informational findings can be found in the General Findings section.

Critical

Vulnerabilities that immediately lead to loss of user funds with minimal preconditions

Examples:

- Misconfigured authority or access control validation
- Improperly designed economic incentives leading to loss of funds

High

Vulnerabilities that could lead to loss of user funds but are potentially difficult to exploit.

Examples:

- Loss of funds requiring specific victim interactions
- Exploitation involving high capital requirement with respect to payout

Medium

Vulnerabilities that could lead to denial of service scenarios or degraded usability.

Examples:

- Malicious input that causes computational limit exhaustion
- Forced exceptions in normal user flow

Low

Low probability vulnerabilities which could still be exploitable but require extenuating circumstances or undue risk.

Examples:

- Oracle manipulation with large capital requirements and multiple transactions

Informational

Best practices to mitigate future security risks. These are classified as general findings.

Examples:

- Explicit assertion of critical internal invariants
- Improved input validation