Zellic

LayerZero OFT Wrapper

Smart Contract Security Assessment

January 19, 2023

Prepared for:

Ryan Zarick

LayerZero

Prepared by:

Katerina Belotskaya

Zellic Inc.

Contents

About Zellic 2

1 Executive Summary 3

2 Introduction 4

2.1 About LayerZero OFT Wrapper 4

2.2 Methodology 4

2.3 Scope 5

2.4 Project Overview 5

2.5 Project Timeline 6

3 Detailed Findings 7

3.1 The `sendOFT` function call can be blocked 7

4 Audit Results 9

4.1 Disclaimers 9

About Zellic

Zellic was founded in 2020 by a team of blockchain specialists with more than a decade of combined industry experience. We are leading experts in smart contracts and Web3 development, cryptography, web security, and reverse engineering. Before Zellic, we founded perfect blue, the top competitive hacking team in the world. Since then, our team has won countless cybersecurity contests and blockchain security events.

Zellic aims to treat clients on a case-by-case basis and to consider their individual, unique concerns and business needs. Our goal is to see the long-term success of our partners rather than simply provide a list of present security issues. Similarly, we strive to adapt to our partners' timelines and to be as available as possible. To keep up with our latest endeavors and research, check out our website zellic.io or follow @zellic_io on Twitter. If you are interested in partnering with Zellic, please contact us at hello@zellic.io.

1 Executive Summary

Zellic conducted an audit for LayerZero from January 18th to January 19th, 2023 on the scoped contracts and discovered 1 low severity finding. Fortunately, no critical issues were found. We applaud LayerZero for their attention to detail and diligence in maintaining incredibly high code quality standards in the development of LayerZero OFT Wrapper.

Zellic thoroughly reviewed the LayerZero OFT Wrapper codebase to find any technical issues outlined in the Methodology section of this document.

Our general overview of the code is that it was very well-organized and structured. The code coverage is high and tests are included for the vast majority of the functions. The documentation was extensive. The code was easy to comprehend and intuitive.

Breakdown of Finding Impacts

Impact Level	Count
Critical	0
High	0
Medium	0
Low	1
Informational	0

2 Introduction

2.1 About LayerZero OFT Wrapper

LayerZero OFT Wrapper is a wrapper for OmnichainFungibleToken. The contract allows flexible configuring of the fee value for various oft token contracts.

2.2 Methodology

During a security assessment, Zellic works through standard phases of security auditing including both automated testing and manual review. These processes can vary significantly per engagement, but the majority of the time is spent on a thorough manual review of the entire scope.

Alongside a variety of tools and analyzers used on an as-needed basis, Zellic focuses primarily on the following classes of security and reliability issues:

Basic coding mistakes. Many critical vulnerabilities in the past have been caused by simple, surface-level mistakes that could have easily been caught ahead of time by code review. Depending on the engagement, we may also employ sophisticated analyzers such as model checkers, theorem provers, fuzzers, and so on as necessary. We also perform a cursory review of the code to familiarize ourselves with the contracts.

Business logic errors. Business logic is the heart of any smart contract application. We examine the specifications and designs for inconsistencies, flaws, and weaknesses that create opportunities for abuse. For example, these include problems like unrealistic tokenomics or dangerous arbitrage opportunities. To the best of our abilities, time permitting, we also review the contract logic to ensure that the code implements the expected functionality as specified in the platform's design documents.

Integration risks. Several well-known exploits have not been the result of any bug within the contract itself; rather, they are an unintended consequence of the contract's interaction with the broader DeFi ecosystem. Time permitting, we review the contracts' external interactions and summarize the associated risks: for example, flash loan attacks, oracle price manipulation, MEV/sandwich attacks, and so on.

Code maturity. We look for potential improvements in the code base in general. We look for violations of industry best practices and guidelines and code quality standards. We also provide suggestions for possible optimizations, such as gas optimization, upgradeability weaknesses, centralization risks, and so on.

For each finding, Zellic assigns it an impact rating based on its severity and likelihood.

There is no hard-and-fast formula for calculating a finding's impact. Instead, we assign it on a case-by-case basis based on our judgment and experience. Both the severity and likelihood of an issue affect its impact. For instance, a highly severe issue's impact may be attenuated by a low likelihood. We assign the following impact ratings (ordered by importance): Critical, High, Medium, Low, and Informational.

Zellic organizes its reports such that the most important findings come first in the document, rather than being strictly ordered on impact alone. Thus, we may sometimes emphasize an "Informational" finding higher than a "Low" finding. The key distinction is that although certain findings may have the same impact rating, their *importance* may differ. This varies based on various soft factors, like our clients' threat models, their business needs, and so on. We aim to provide useful and actionable advice to our partners considering their long-term goals, rather than a simple list of security issues at present.

2.3 Scope

The engagement involved a review of the following targets:

LayerZero OFT Wrapper Contracts

Repository	https://github.com/LayerZero-labs/oft-wrapper
Versions	`e14285703001e1acb51dfe3d6abb665d80fb58e4`
Programs	• OFTWrapper • IOFTWrapper
Type	Solidity
Platform	EVM-compatible

2.4 Project Overview

Zellic was contracted to perform a security assessment with one consultants for a total of two person-day. The assessment was conducted over the course of two calendar days.

Contact Information

The following project managers were associated with the engagement:

Jasraj Bedi, Co-founder
jazzy@zellic.io

The following consultants were engaged to conduct the assessment:

Katerina Belotskaia, Engineer
kate@zellic.io

2.5 Project Timeline

The key dates of the engagement are detailed below.

January 18, 2023	Start of primary review period
January 19, 2023	End of primary review period

3 Detailed Findings

3.1 The `sendOFT` function call can be blocked

- **Target**: OFTWrapper
- **Category**: Coding Mistakes
- **Likelihood**: Low
- **Severity**: Low
- **Impact**: Low

Description

The contract owner can set any `bps` value of the variables `defaultBps` and the `oftBps[_oft]` in the range from 0 to the maximum `BPS_DENOMINATOR` inclusive. But during the `sendOFT` function call, the `getAmountAndFees` function will check that the final `bps` value is less than `BPS_DENOMINATOR` and revert the transaction if it equals or more.

```
function getAmountAndFees(
        address _oft,
        uint256 _amount,
        uint256 _callerBps
    )
        public
        view
        override
        returns (
            uint256 amount,
            uint256 wrapperFee,
            uint256 callerFee
        )
    {
        uint256 wrapperBps;

        if (oftBps[_oft] == MAX_UINT) {
            wrapperBps = 0;
        } else if (oftBps[_oft] > 0) {
            wrapperBps = oftBps[_oft];
        } else {
            wrapperBps = defaultBps;
        }
        require(wrapperBps + _callerBps < BPS_DENOMINATOR, "OFTWrapper:
    Fee bps exceeds 100%");
```

```
        ...
    }
```

Impact

In case if the contract owner sets the `defaultBps` to the maximum `BPS_DENOMINATOR` value, the `sendOFT` function call will be blocked for all unassigned `_oft` addresses. Also if the maximum `oftBps` value is set for a specific `_oft` address, the `sendOFT` function call with this `_oft` address will be reverted.

Recommendations

Set a limit for the `defaultBps` and `oftBps[_oft]` values strictly less than the `BPS_DENOMINATOR` value.

Remediation

This issue was fixed by LayerZero in commit f11289a5.

4 Audit Results

At the time of our audit, the code was not deployed to mainnet evm.

During our audit, we discovered 1 low risk findings. LayerZero acknowledged this finding and implemented fix.

4.1 Disclaimers

This assessment does not provide any warranties about finding all possible issues within its scope; in other words, the evaluation results do not guarantee the absence of any subsequent issues. Zellic, of course, also cannot make guarantees about any additional code added to the assessed project after the audit version of our assessment. Furthermore, because a single assessment can never be considered comprehensive, we always recommend multiple independent assessments paired with a bug bounty program.

For each finding, Zellic provides a recommended solution. All code in these recommendations are intended to convey how an issue may be resolved (i.e., the idea), but they may not be tested or functional code.

Finally, the contents of this assessment report are for informational purposes only; do not construe any information in this report as legal, tax, investment, or financial advice. Nothing contained in this report constitutes a solicitation or endorsement of a project by Zellic.