

Code Audit

for



Streamflow
FINANCE

Project Information

Project	
Mission	Code audit
Client	Streamflow Finance
Start Date	02/08/2022
End Date	02/23/2022

Document Revision			
Version	Date	Details	Authors
1.0	02/22/2022	Document creation	Thibault MARBOUD Théo LEJEUNE
1.1	02/23/2022	Peer review	Xavier BRUNI
2.0	03/02/2022	Public version	Xavier BRUNI

Table of Contents

Project Information	2
Overview	4
Mission Context	4
Mission Scope	4
Project Summary	4
Synthesis	6
Vulnerabilities summary	6
Vulnerabilities & issues table	7
Identified vulnerabilities	7
Identified vulnerabilities	8
Faulty authority validation	8
Conclusion	10

Overview

Mission Context

The purpose of the mission was to perform a code audit to discover issues and vulnerabilities in the mission scope. Comprehensive testing has been performed utilizing automated and manual testing techniques.

Mission Scope

As defined with Streamflow Finance before the mission, the scope of this assessment was two Solana programs. This report only concerns the Partner oracle program, another one has been edited for the Protocol program. The code source was supplied through the following GitHub repositories:

- <https://github.com/streamflow-finance/protocol> [7fc6d89](#) (main)
- <https://github.com/streamflow-finance/partner-oracle> / [50a145e](#) (master)

OPCODES engineers were due to strictly respect the perimeter agreed with Streamflow Finance as well as respect ethical hacking behavior.

At the end of the audit, Streamflow Finance patched multiple reported vulnerabilities. These patches have been reviewed by OPCODES up to commit [06484cc](#). The issues table list the status of each vulnerability (Fixed / Accepted risk / Rejected) and the vulnerability write-up contains a link to the resolution commit when applicable.

Note: OPCODES engineers audited the main branch of the protocol repository and master branch of the partner oracle repository.

Project Summary

As a reminder, streamflow is building a token vesting library/application on Solana. It provides all the logic necessary to lock SPL tokens and distribute them to a specific recipient within a given timeframe.

Streamflow works with a set of partners that act as referrer and earn a fee for every stream created via them. Not everyone can be a partner and the fee share they get from the stream creation can be different for each partner. The Partner Oracle is the program used to manage partners. It is used to add, remove, and set the share of fees each partner is eligible for.

Protocol program interacts with the partner oracle account to fetch the partner data and distribute the fees.

The partner oracle program does not have any tests. OPCODES strongly encourage Streamflow to create tests for the partner oracle program.

The application is developed without a framework, but Streamflow added a wrapper using Anchor framework. This allows for easier testing and frontend integration using the tools provided by Anchor ecosystem. If possible, it should be considered to switch completely the program to Anchor. From a security point of view, some vulnerabilities could have been avoided with Anchor.






Synthesis

Security Level: GOOD

The overall security level is considered as good. OPCODES investigation only produced 1 major severity result which has been fixed.




The major vulnerability concerns a faulty validation of the calling authority. This issue allowed anyone to add or remove partners and to tweak the fees.

Vulnerabilities summary

Total vulnerabilities	1
 Critical	0
 Major	1
 Medium	0
 Minor	0
 Informational	0

Vulnerabilities & issues table

Identified vulnerabilities

Ref	Vulnerability title	Severity	Remediation effort	Status
#1	Faulty authority validation	 Major	 Low	 Fixed

Identified vulnerabilities

Faulty authority validation

Severity	Remediation effort	Status
Major	Low	Fixed

Description

Only Streamflow' administrators (authority) must be allowed to add, remove, and update partners data. However, a faulty logical test makes it possible for an attacker to change the partners data without having access to the administrators' private key.

An attacker can pass the authority account without signing the transaction and the authority verification will succeed and return false, so it will not error out.

src/entrypoint.rs – L29 / L71 / L127

```
if !authority.is_signer && authority.key != &authority_pub {
    return Err(ProgramError::MissingRequiredSignature);
}
```

Scope

Partner Oracle

Risk

An attacker might add, remove, or make change to streamflow partners.

Remediation

The check should be using a logical OR instead of a logical AND


```
if !authority.is_signer || authority.key != &authority_pub {  
    return Err(ProgramError::MissingRequiredSignature);  
}
```

Fix

This issue has been discussed and fixed with commit [06484cc](#) during the audit.

Conclusion

The Partner Oracle program only contains 3 very simple instructions to create, insert/modify and remove partners. The number of attack vector is low since each instruction can only be called by Streamflow' administrators.

The major vulnerability regarding the faulty authority validation has already been patched during the audit.

We advise Streamflow to create test cases for the Partner Oracle program as there is no coverage currently.

In the future, it should be considered to switch the program to Anchor which reduces the number of account checks required and improves code readability.