< Back to articles

Helio Protocol Audit Summary

Andrea Nováková // AUDITS, SOLANA MAY 20, 2022

Helio engaged Ackee Blockchain to review and audit their protocol between **May 16 and 20, 2022**. The entire audit process was conducted with a total time commitment of **5 engineering days**. We now publish a summary of our results.

METHODOLOGY

First, we took the time to understand the entire Helio platform.

Reviewing the specifications, sources, and instructions provided to us is essential to ensure we understand the project's size, scope, and functionality. This is followed by a detailed manual code review, which is the process of reading the source code line by line to identify potential vulnerabilities.

When the code review is complete, we run client's tests to ensure the system works as expected and potentially write missing unit or fuzzy tests using our testing framework Trdelnik. We also deploy programs locally and try to attack and break the system.

SCOPE

We audited commit `8a6b1a20551cde8cff68d55e43baa5524692e82c` of the `heliofi/helio-protocol` repository.

During the security review of the *helio-protocol* program, we paid particular attention to the following questions:

- Is the correctness of the program ensured (does it correctly implement the project goals)?
- Do the program correctly use dependencies or other programs they rely on (e.g., SPL dependencies)?
- Is the code vulnerable to economic attacks?

FINDINGS

Here we present our findings.

Critical severity

C1: *withdraw_payment* and *cancel_payment* instructions will not work after the pay stream ends

C2: Possibility of stealing tokens from escrow token account

C3: Possibility of stuck tokens

C4: Using the same struct for SOL payments as for token payments results in the possibility of a tokens lock attack

High severity

No high severity issues were found.

Medium severity

M1: Hanging *payment_token_account*(s)

Low severity

L1: Using *find_program_address* instead of *create_program_address*

Warning severity

No warning severity issues were found.

Informational severity

I1: *PaymentAccount* struct has unused fields

I2: Unnecessary mutable modifier

CONCLUSION

Our review resulted in **8 findings** ranging from *Informational* to *Critical* severity. Four of these issues were critical, causing either the lockup of assets or the possibility of stealing them.

The issues C1 and C2 were reported to Helio immediately upon discovery in the separate revision of this document (pre-audit version 0.1), even though the Helio protocol was not yet live.

We recommended Helio:

- to address all reported issues;
- another full audit once the issues are fixed (the reason is that we devoted a lot of time on exploits due to many critical issues);
- to follow Rust and Solana's best practices.

Update: On **July 22, 2022**, Helio provided an updated codebase that acknowledged and fixed all reported issues. During the fix review process, we found three additional informational issues.

I3: Anchor version mismatch

I4: Impossible to build and test with a newer anchor version

I5: A missing *CHECK* doc comment

All of them have been reported to the Helio team.

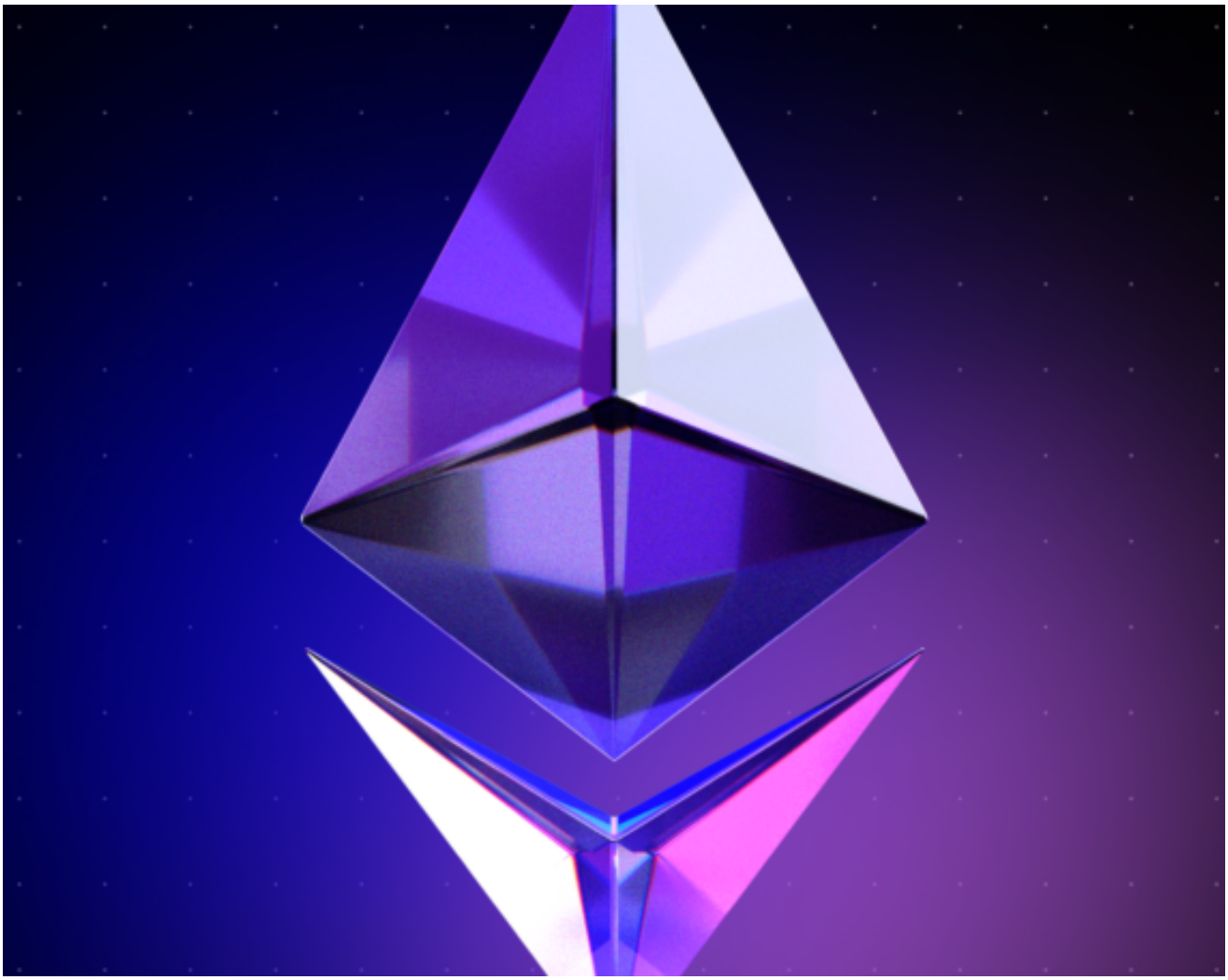
Ackee Blockchain's full *Helio protocol* audit report with a more detailed description of all findings and recommendations can be found here.

We were delighted to audit **Helio** and look forward to working with them again.

Audit Blockchain Security Solana

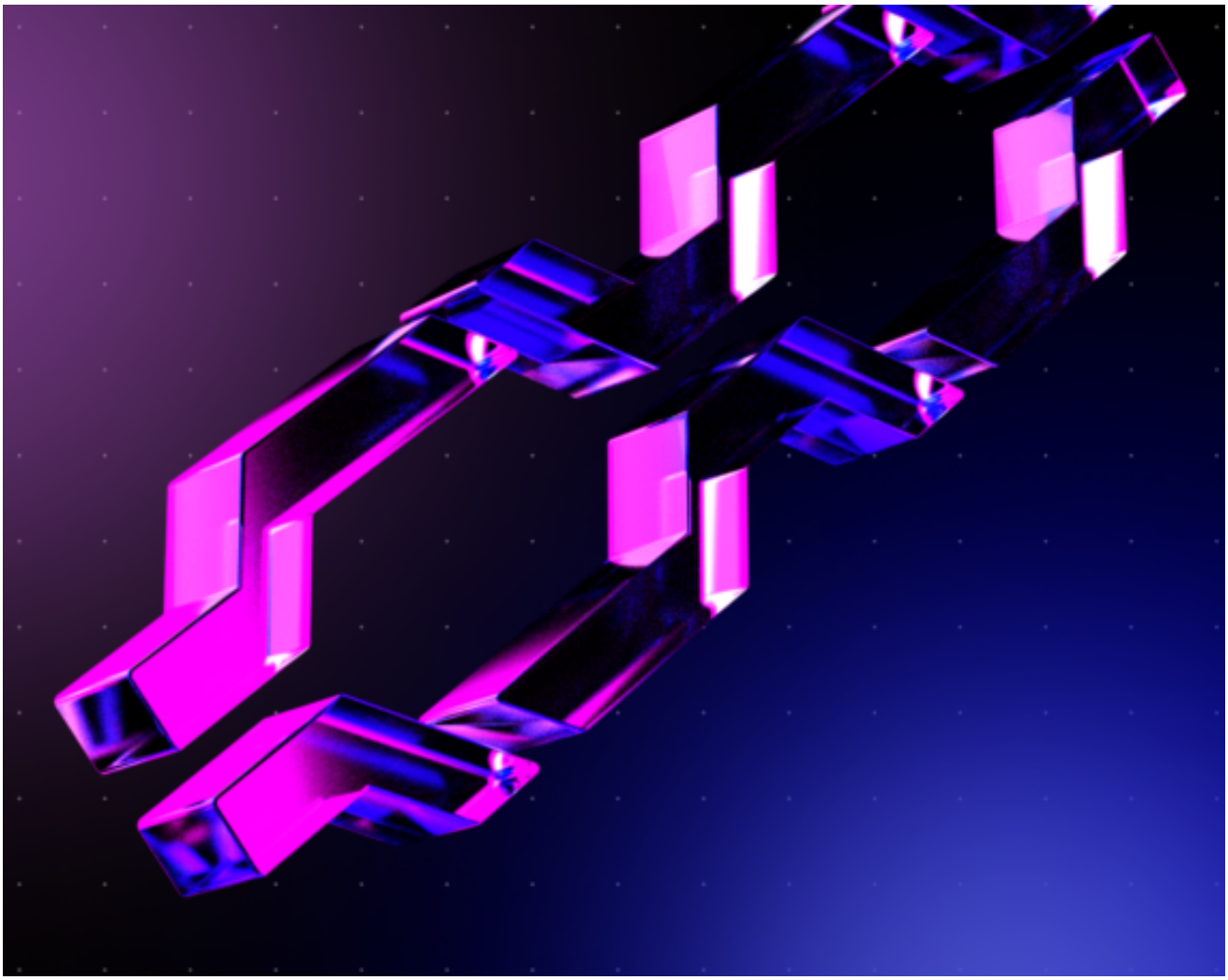
Andrea Nováková

You May Also Like



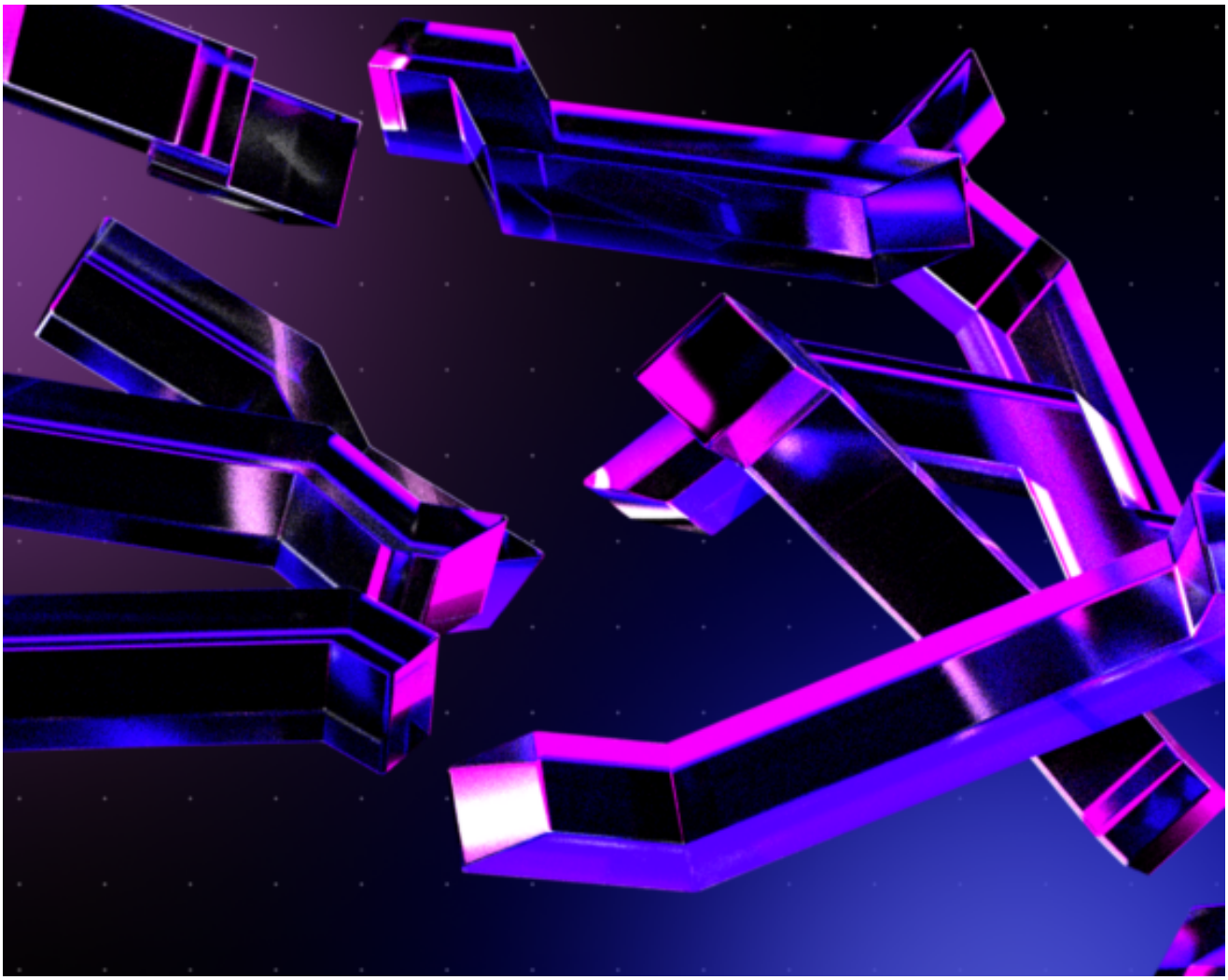
// EDUCATION, ETHEREUM APRIL 27, 2025

Ethereum's Pectra Upgrade: Security Implications and Insights



// EDUCATION, EXPLOITS, HACKS, SOLIDITY, WAKE APRIL 25, 2025

Reentrancy Attack in ERC-777



// EDUCATION, EXPLOITS, HACKS, SOLIDITY, WAKE APRIL 11, 2025

Flash Loan Reentrancy Attack