

A Unified Framework for Formal Verification of Vyper Bytecode using Halmos

Lucas Goiriz



Index

- \$ whoami
- The importance of Diversity
- The Curve Finance Hack
- Compiler bugs: first time?
- Halmos: Leveraging existing tests for formal verification
- A naive integration of Halmos and Vyper
- Demo? Demo!

\$ whoami



Lucas Goiriz | luksgrin | Bronicle

- PhD Student (Applied Mathematics to Biosciences) at I2SysBio (CSIC)
- JSR at Spearbit Labs
- Secureum Alumni/Volunteer



X @Cryptonicle1

GitHub luksgrin



The importance of Diversity

Execution clients

- Nethermind, Erigon, Geth, Akula, Besu, Reth...

Consensus clients

- Lighthouse, Lodestar, Nimbus, Prism, Teku...

Development frameworks

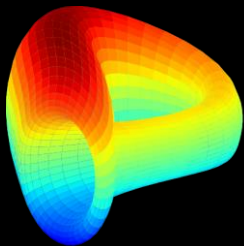
- Foundry, Hardhat, Brownie, Truffle†, ApeWorkx...

EVM languages

- Solidity, Yul/Yul+, Vyper, Huff, Fe...



The Curve Finance Hack



July 30, 2023

- \$70M estimated losses
- Largest MEV reward blocks EVER



eric.eth ✓
@econoar · Seguir



Today has produced some of the largest MEV reward blocks in Ethereum's history.

Slot 6,992,273: 584 ETH

Slot 6,993,342: 345 ETH

Slot 6,992,050: 247 ETH

Slot 6,993,346: 51 ETH

12:24 a. m. · 31 jul. 2023

The Curve Finance Hack



Cause: failure in Vyper's reentrancy locks

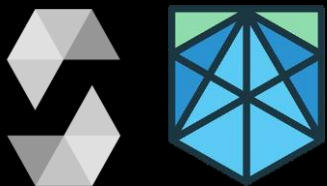
- Versions 0.2.15, 0.2.16 and 0.3.0 of the Vyper compiler.

```
#pragma version 0.2.15

@external
@nonreentrant("lock")
def dummyRawCall(to: address, data: bytes32) -> bytes32:
    res: Bytes[32] = raw_call(
        to,
        concat(
            method_id("evilFunc(bytes32)"),
            data
        ),
        max_outsize=32

    return convert(res, bytes32)
```

Compiler bugs: first time?



Symbolic testing revealed solidity compiler bugs:

- The Solidity Compiler silently disrupts storage
- Overly optimistic optimizer

Certora Prover recently added support for Vyper

Halmos: Leveraging existing tests for formal verification

- Halmos automatically integrates on Foundry projects' tests
- Currently only supports Solidity

But it is possible making it work with Vyper via
a couple hacks

A naive integration of Halmos and Vyper



Ingredients:


- Vyper compiler
- Foundry
- Foundry tests written in Solidity
- Snekmate utils*
- Latest Halmos version





A naive integration of Halmos and Vyper


add `ffi` cheatcode #185

 Merged daejunpark merged 15 commits into `a16z:main` from `luksgryn:main`  on Sep 12

 Conversation 12

 Commits 15

 Checks 91

 Files changed 10



luksgryn commented on Aug 18 • edited ▾

Contributor ...

This PR contains the implementation of Foundry's `function ffi(string[] calldata) external returns (bytes memory);` cheatcode with some helper functions and some code refactoring. See `ffi`'s [documentation here](#).

Dependencies added

- Python's native `subprocess` module

Demo? Demo!



Special thanks to



- Daejun Park



- Karmacoma-eth

Thank you! Gn!