TRAIL OF BITS

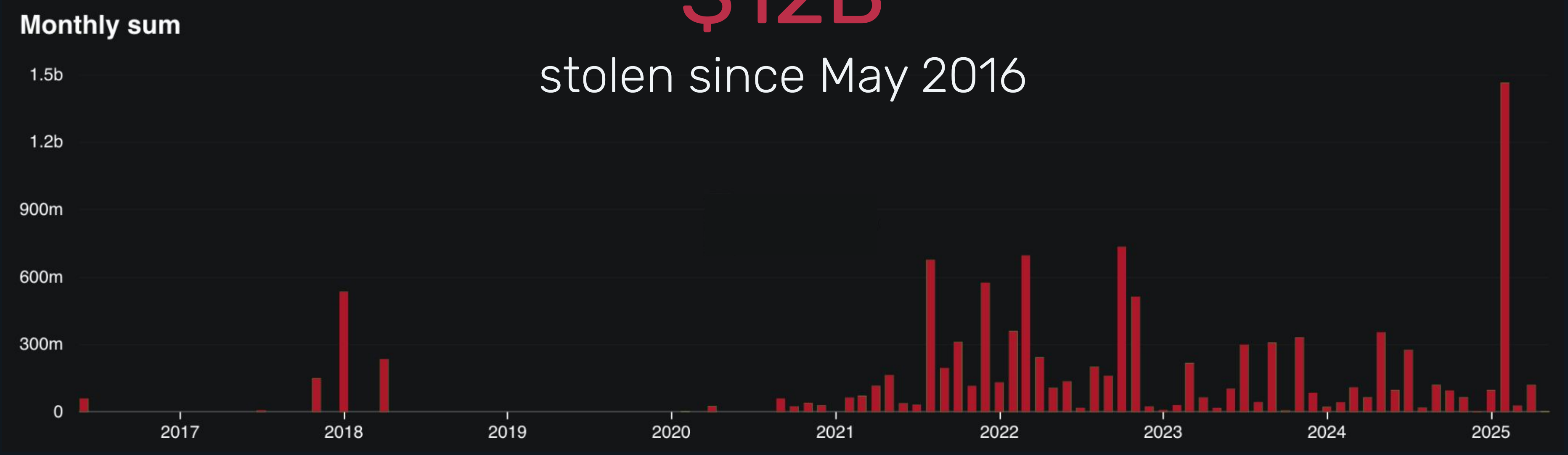
How to become a smart contract auditor

Who am I?

- nisedo (@nisedo_)
- Founded Soliditors, the 🇫🇷 Web3Sec community
- Blockchain Security Engineer at Trail of Bits
  - We help developers build safer software
  - R&D focused: Slither, Medusa, Echidna, solc-select, …

Why become a smart contract auditor?

Web3 security is broken 🩸



$12B

stolen since May 2016

DefiLlama.com

Web3Sec vs Web2Sec 🛡️

🥷 **Web2 hack**

↓

🗂️ **data theft**

↓

🌚 **dark web**

↓

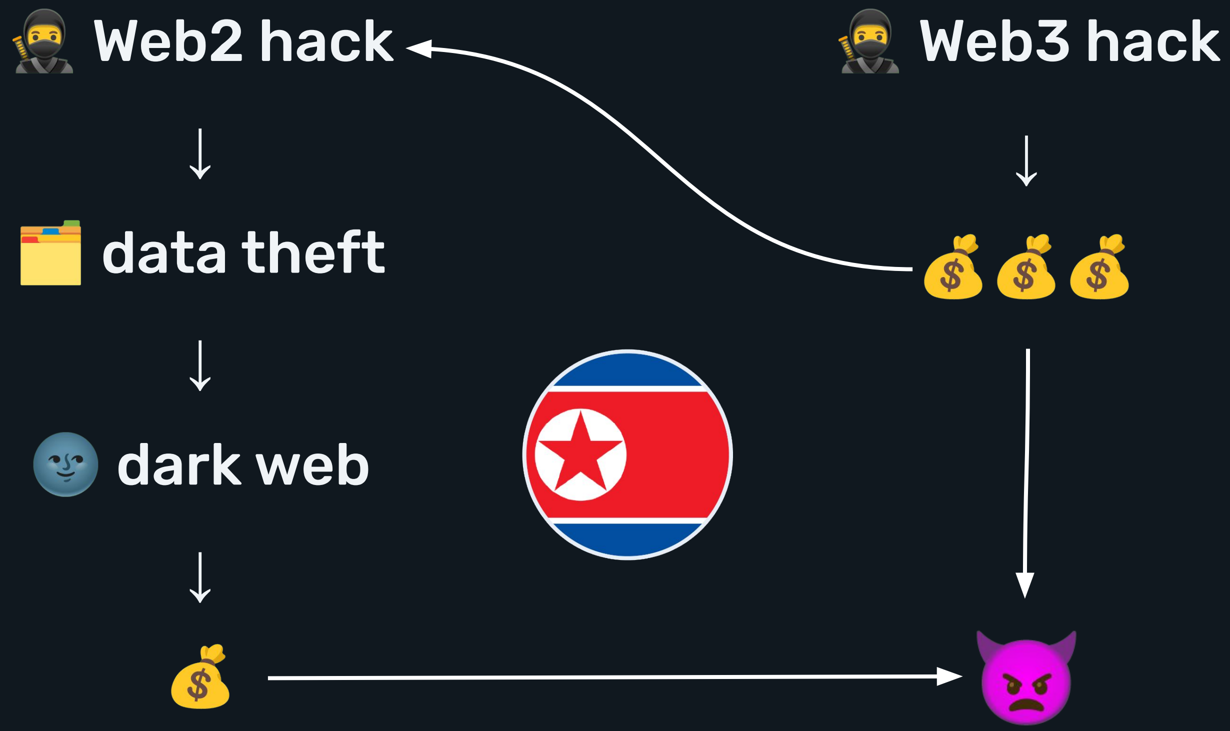
💰

🥷 **Web3 hack**

↓

💰💰💰

Our work actually matters 🛡️



🥷 Web2 hack

↓

🗂️ data theft

↓

🌚 dark web

↓

💰

🥷 Web3 hack

↓

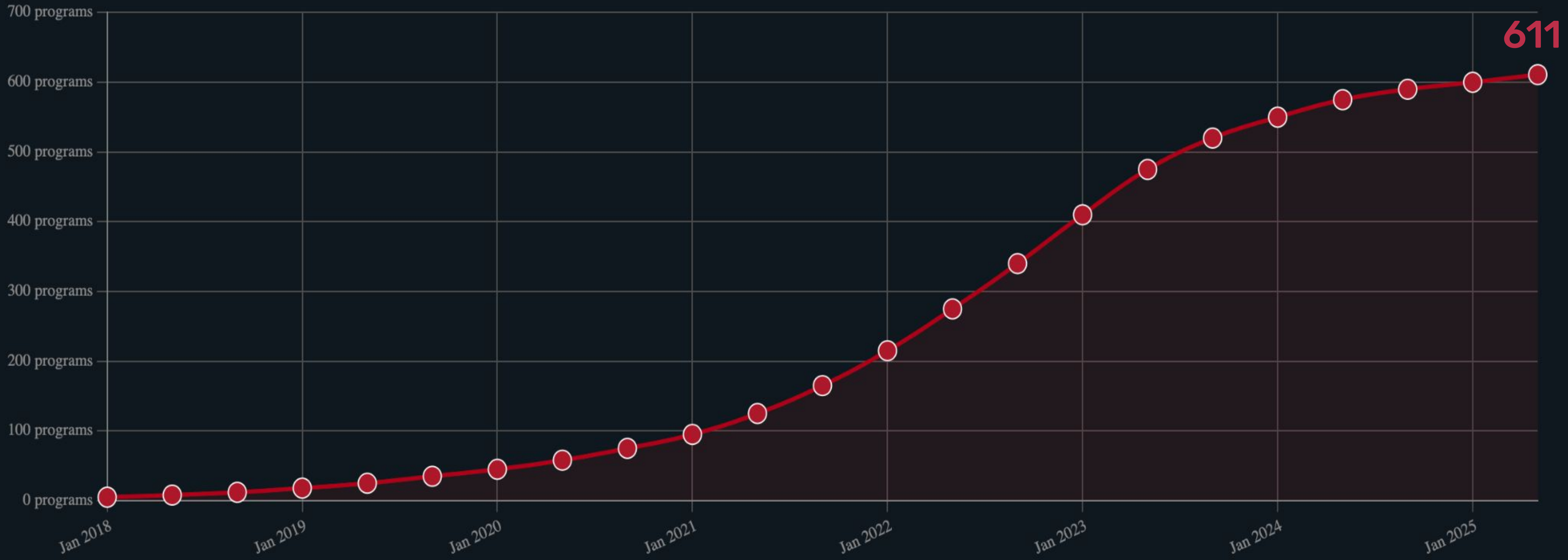
💰💰💰

😈

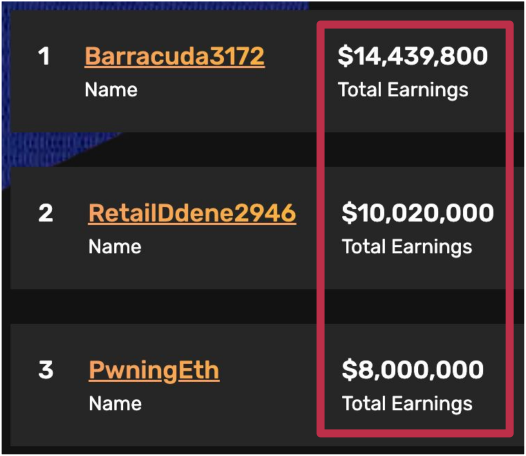
Growing demand 📈

Web3 Bug Bounty Programs

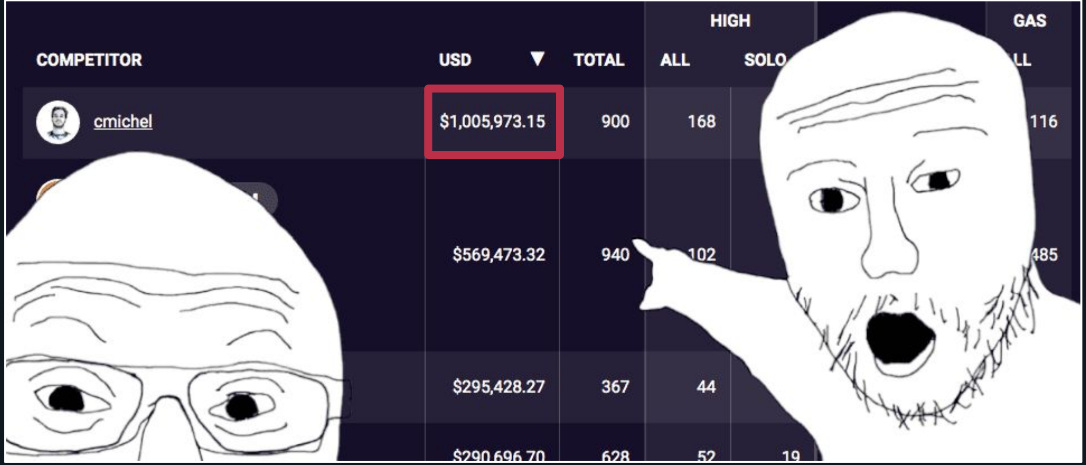
Bug bounty growth across platforms (2018-2025)



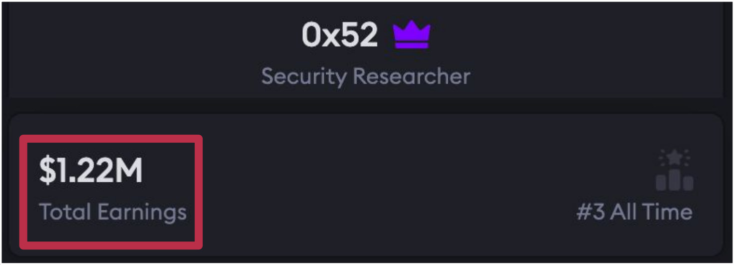
Money, Money, Money 💰



1	Barracuda3172 Name	$14,439,800 Total Earnings
2	RetailDdene2946 Name	$10,020,000 Total Earnings
3	PwningEth Name	$8,000,000 Total Earnings



			HIGH		GAS
COMPETITOR	USD ▼	TOTAL	ALL	SOLO	LL
cmichel	$1,005,973.15	900	168		116
	$569,473.32	940	102		485
	$295,428.27	367	44		
	$290,696.70	628	52	19	



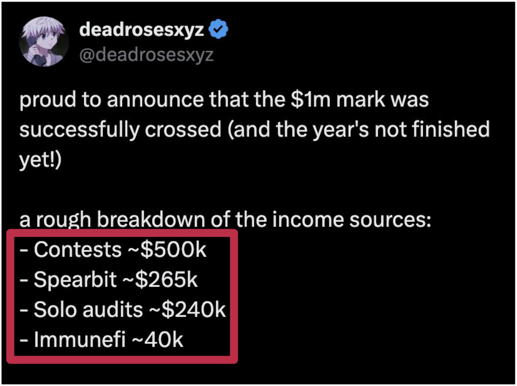
0x52 👑

Security Researcher

$1.22M

Total Earnings

#3 All Time



deadrosesxyz

@deadrosesxyz

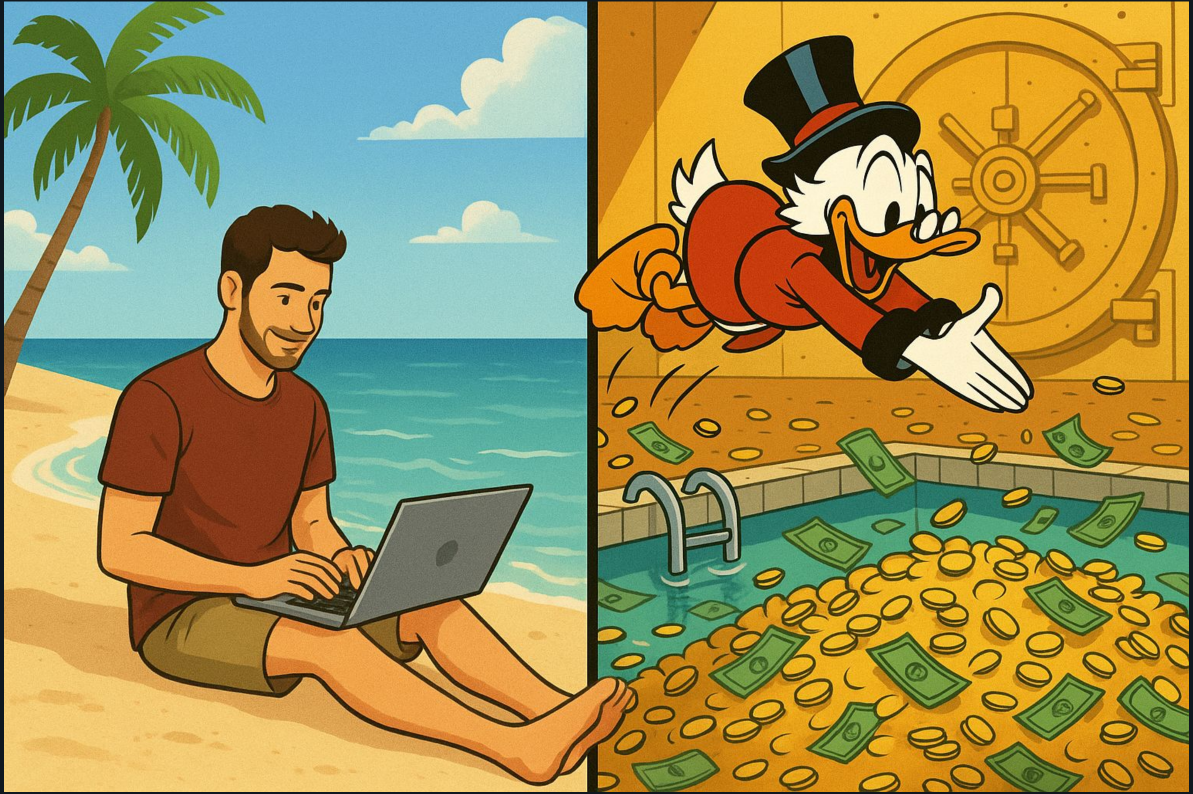
proud to announce that the $1m mark was successfully crossed (and the year's not finished yet!)

a rough breakdown of the income sources:

- Contests ~$500k
- Spearbit ~$265k
- Solo audits ~$240k
- Immunefi ~40k

But...

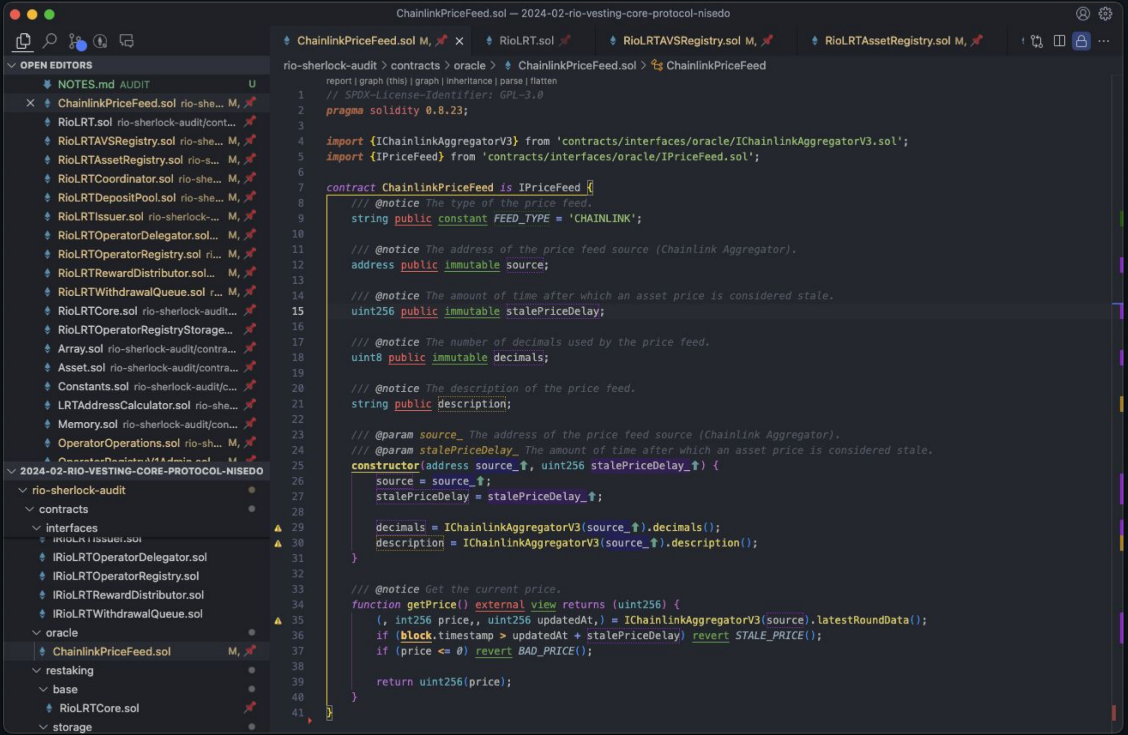
Expectation



Reality

What do smart contract auditors do?

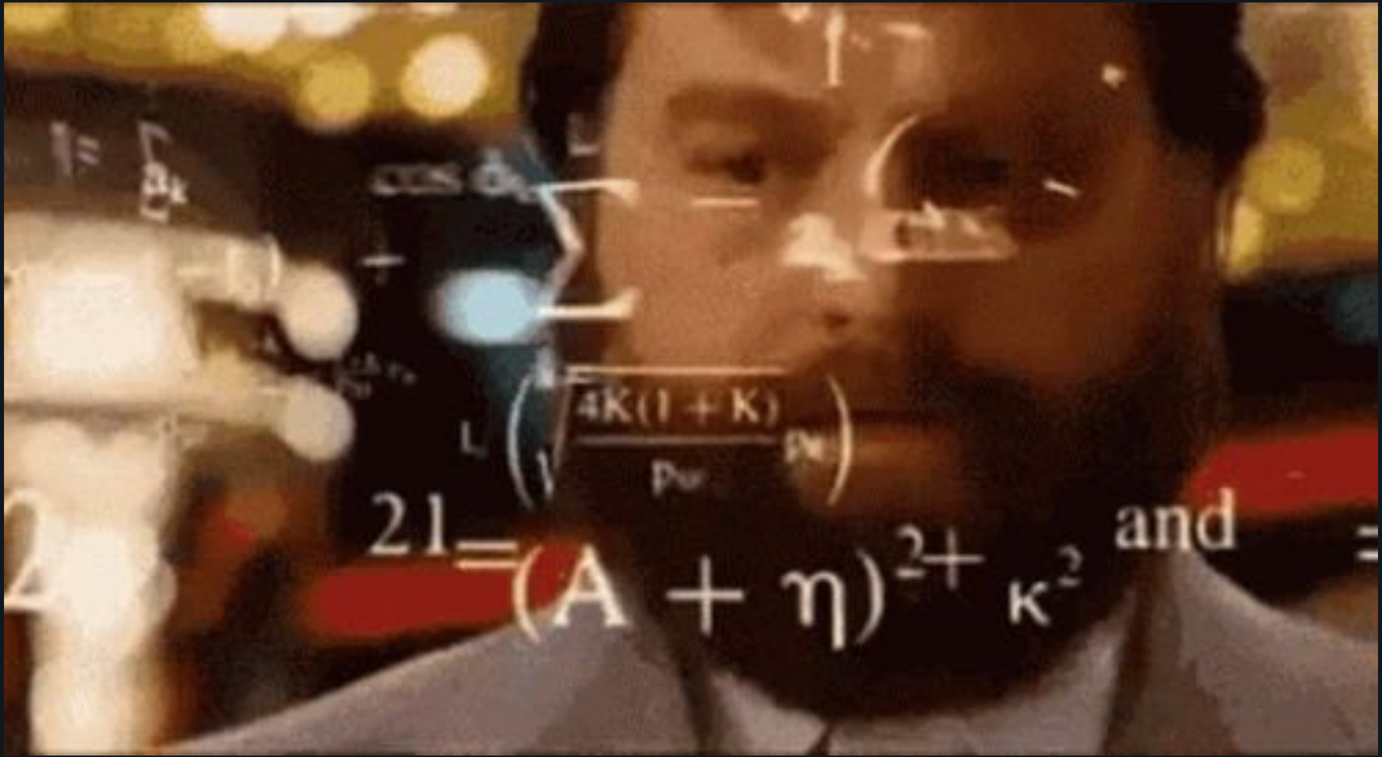
We find bugs… but how?


ChainlinkPriceFeed.sol — 2024-02-rio-vesting-core-protocol-nisedo
OPEN EDITORS
NOTES.md AUDIT U
ChainlinkPriceFeed.sol rio-she... M,
RioLRT.sol rio-sherlock-audit/cont...
RioLRTAVSRegistry.sol rio-she... M,
RioLRTAssetRegistry.sol rio-s... M,
RioLRTCoordinator.sol rio-she... M,
RioLRTDepositPool.sol rio-she... M,
RioLRTIssuer.sol rio-sherlock-... M,
RioLRTOperatorDelegator.sol... M,
RioLRTOperatorRegistry.sol ri... M,
RioLRTRewardDistributor.sol... M,
RioLRTWithdrawalQueue.sol r... M,
RioLRTCore.sol rio-sherlock-audit...
RioLRTOperatorRegistryStorage...
Array.sol rio-sherlock-audit/contra...
Asset.sol rio-sherlock-audit/contra...
Constants.sol rio-sherlock-audit/c...
LRTAddressCalculator.sol rio-she...
Memory.sol rio-sherlock-audit/con...
OperatorOperations.sol rio-sh... M,
2024-02-RIO-VESTING-CORE-PROTOCOL-NISEDO
rio-sherlock-audit
contracts
interfaces
IRioLRTOperatorDelegator.sol
IRioLRTOperatorRegistry.sol
IRioLRTRewardDistributor.sol
IRioLRTWithdrawalQueue.sol
oracle
ChainlinkPriceFeed.sol M,
restaking
base
RioLRTCore.sol
storage
ChainlinkPriceFeed.sol M,
RioLRT.sol
RioLRTAVSRegistry.sol M,
RioLRTAssetRegistry.sol M,
rio-sherlock-audit > contracts > oracle > ChainlinkPriceFeed.sol > ChainlinkPriceFeed
report | graph (this) | graph | inheritance | parse | flatten

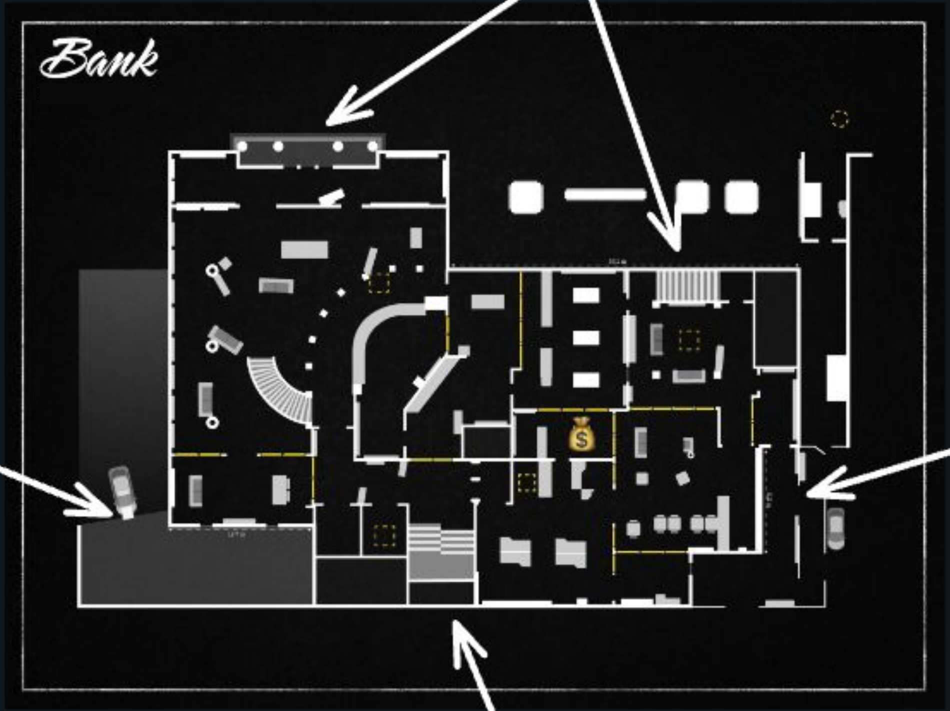
```solidity
// SPDX-License-Identifier: GPL-3.0
pragma solidity 0.8.23;

import {IChainlinkAggregatorV3} from 'contracts/interfaces/oracle/IChainlinkAggregatorV3.sol';
import {IPriceFeed} from 'contracts/interfaces/oracle/IPriceFeed.sol';

contract ChainlinkPriceFeed is IPriceFeed {
    /// @notice The type of the price feed.
    string public constant FEED_TYPE = 'CHAINLINK';

    /// @notice The address of the price feed source (Chainlink Aggregator).
    address public immutable source;

    /// @notice The amount of time after which an asset price is considered stale.
    uint256 public immutable stalePriceDelay;

    /// @notice The number of decimals used by the price feed.
    uint8 public immutable decimals;

    /// @notice The description of the price feed.
    string public description;

    /// @param source_ The address of the price feed source (Chainlink Aggregator).
    /// @param stalePriceDelay_ The amount of time after which an asset price is considered stale.
    constructor(address source_, uint256 stalePriceDelay_) {
        source = source_;
        stalePriceDelay = stalePriceDelay_;

        decimals = IChainlinkAggregatorV3(source_).decimals();
        description = IChainlinkAggregatorV3(source_).description();
    }

    /// @notice Get the current price.
    function getPrice() external view returns (uint256) {
        (, int256 price,, uint256 updatedAt,) = IChainlinkAggregatorV3(source).latestRoundData();
        if (block.timestamp > updatedAt + stalePriceDelay) revert STALE_PRICE();
        if (price <= 0) revert BAD_PRICE();

        return uint256(price);
    }
}
```

We get a giant puzzle

We stare at code



And we piece it together



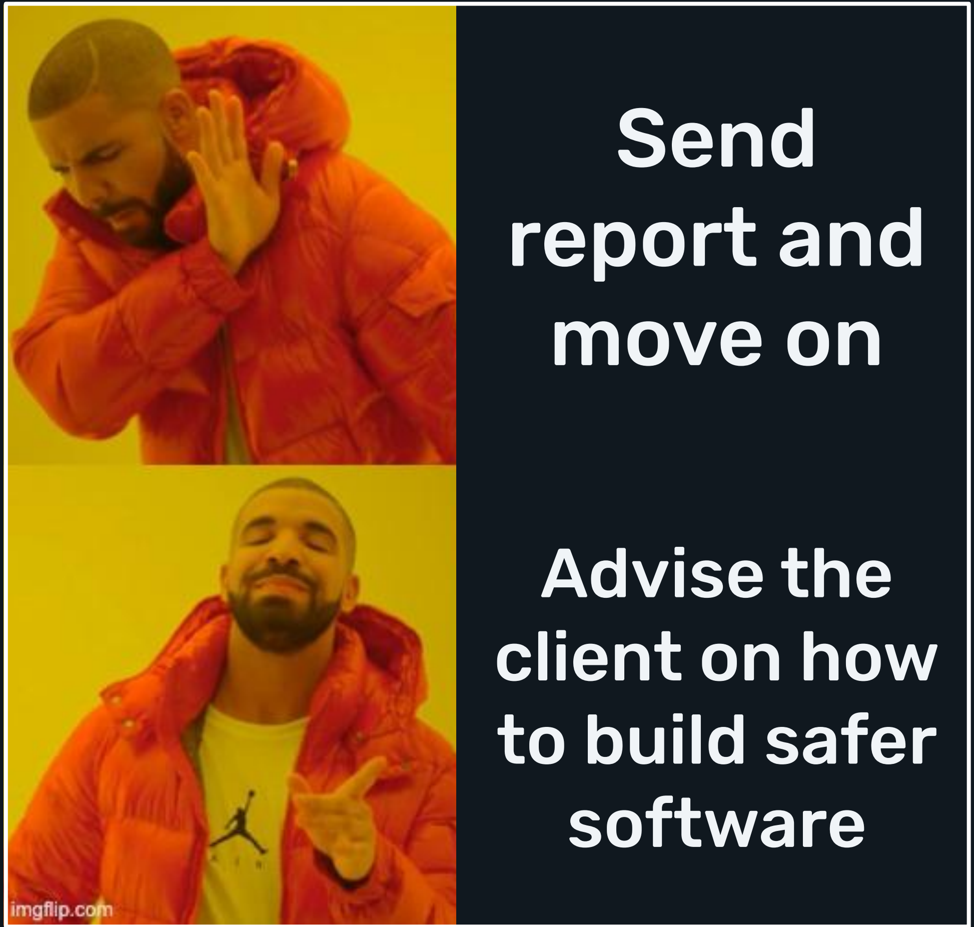
Camera can be avoided here

Code can be bruteforced

No guard here

Open window here

We’re advisors

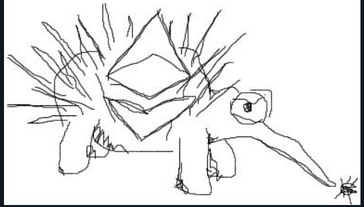


Armed to the teeth 🏹

Framework

Hardhat

Fuzzing



Medusa

Static Analysis

SLITHER

Semgrep

aderyn

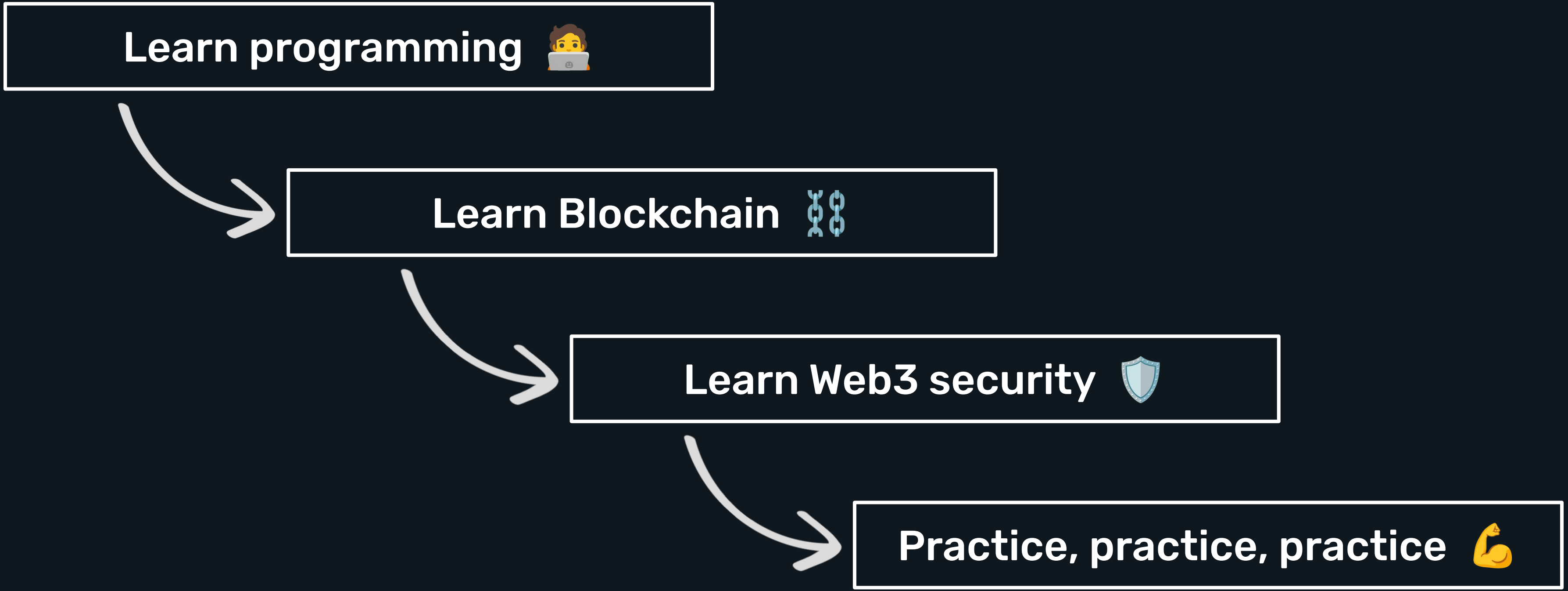
Formal Verification

Halmos

kontrol

How to become a smart contract auditor?

How to become a smart contract auditor? 🎓



Learn programming 🧑‍💻

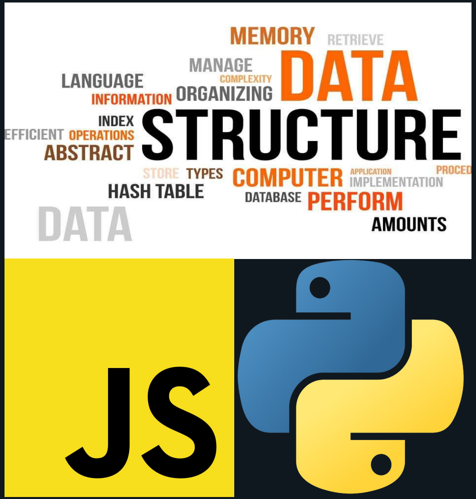
Learn Blockchain ⛓️

Learn Web3 security 🛡️

Practice, practice, practice 💪

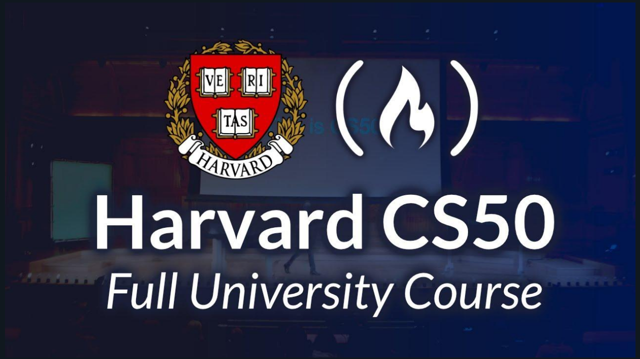
1. Learn programming 🧑‍💻

Computer Science



→

Harvard CS50



2. Learn Blockchain ⛓️

EVM

Solidity

→

Cyfrin Updraft

RareSkills

3. Learn Web3 security 🛡️

Solve CTFs 🚩

The Ethernaut

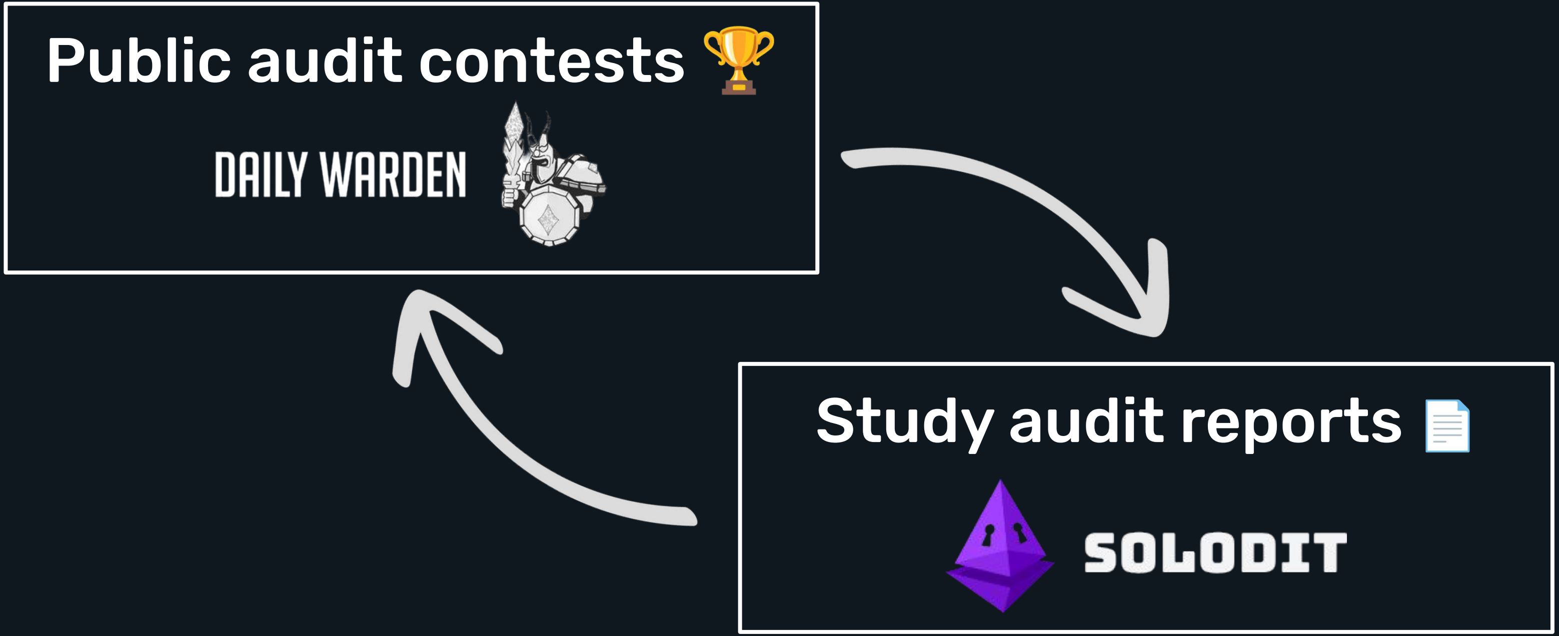
$DAMN VULNERABLE DEFI

ONLYPWNER

Study audit reports 📄

SOLODIT

4. Practice, practice, practice 💪



Public audit contests 🏆

DAILY WARDEN

Study audit reports 📄

SOLODIT

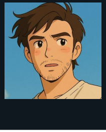
And then what?

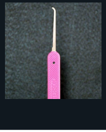
TRAIL OF BITS
EXPERIENCED SMART CONTRACT AUDITOR
imgflip.

The different types of audits 🔎

Audit contests 🏆

Private audits 🕵️





Bug bounties 🏹

Audit firms 🏢

Web3 needs you 🫵

The end