Soliditors



Le métier d'Auditeur de Smart Contracts

iMeet by iExec

& DeFi France

La sécurité dans le Web3 🛡️

🥷 hack Web2 ➡ vol de données ➡ dark web ➡ 💰

🥷 hack Web3 ➡ 💰💰💰

Code immuable 📝

Industrie immature 🐣

Manque de crédibilité 🤡

Les hacks dans le Web3

Total Value Hacked (USD)

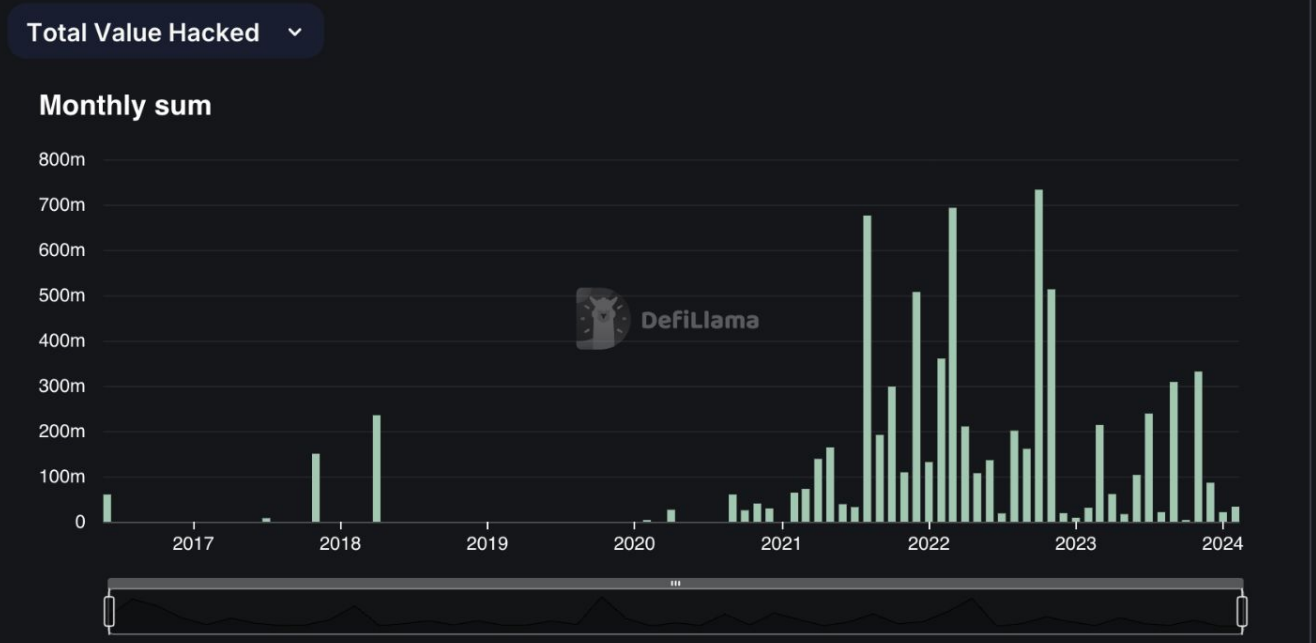
$7.68b

Total Value Hacked in DeFi (USD)

$5.83b

Total Value Hacked in Bridges (USD)

$2.83b



Les principaux vecteurs d'attaque en 2023 ⚔️

1. 🔑 **Clés privées volées**	243 000 000 $	16 attaques
2. 🎰 **Manipulation des récompenses**	200 000 000 $	27 attaques
3. 🔄 **Manipulation de l'oracle de prix**	146 000 000 $	52 attaques
4. 🚫 **Contrôle d'accès aux fonctions insuffisant**	17 000 000 $	25 attaques
5. 💡 **Erreur de logique**	17 000 000 $	9 attaques
6. ⚙️ **Mauvaise configuration**	17 000 000 $	3 attaques
7. 📐 **Validation des paramètres de fonction**	15 000 000 $	9 attaques
8. 🔒 **Réentrance en lecture seule**	14 000 000 $	7 attaques
9. 🔄 **Réentrance**	7 500 000 $	10 attaques
10. 🏛️ **Gouvernance**	7 000 000 $	8 attaques

Blockthreat.io, updraft.cyfrin.io

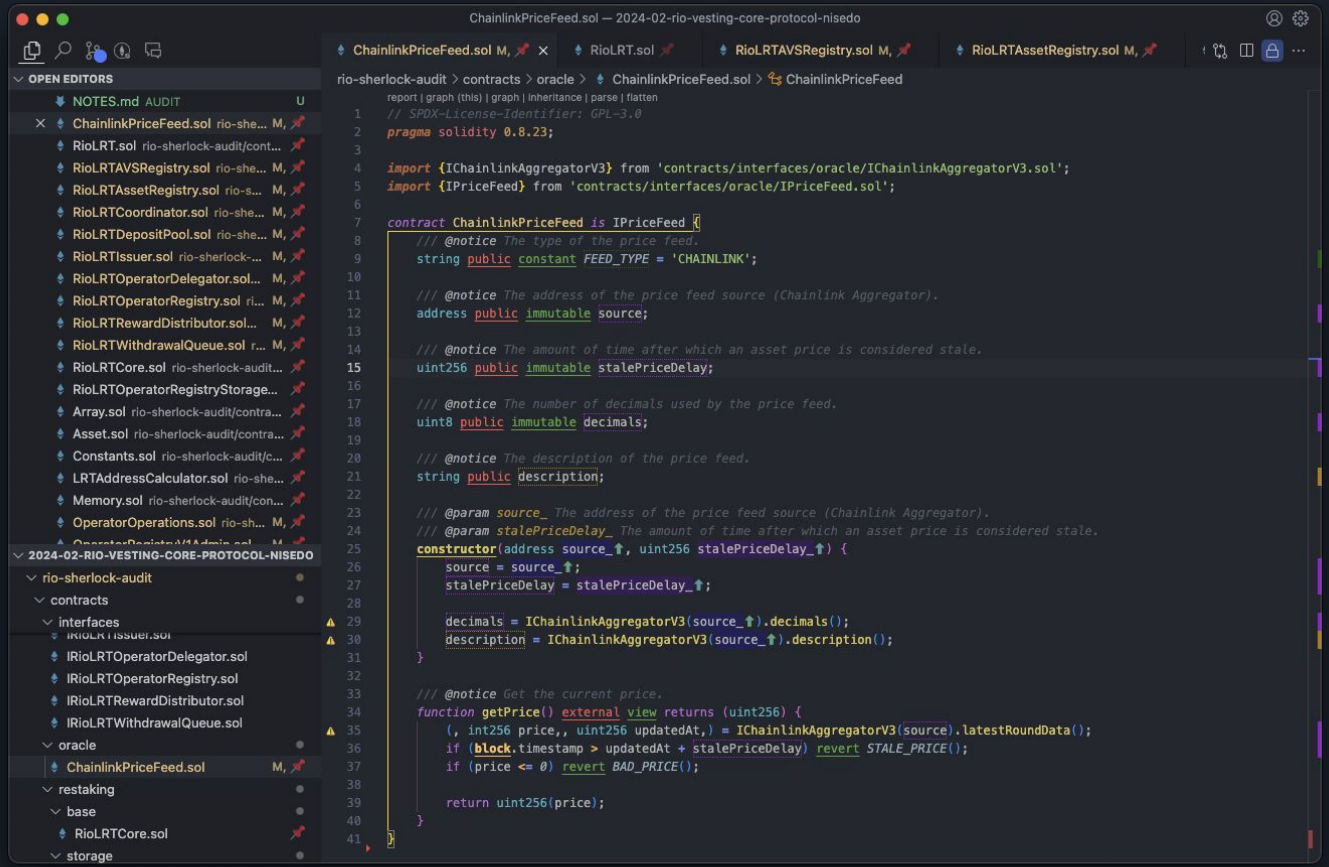
Le métier d’Auditeur de Smart Contracts 🔍

1️⃣ Lire du code 👩‍💻

2️⃣ Identifier des bugs 🐛

3️⃣ Écrire un rapport d’audit 📝

1. Lire du code (Solidity / Rust / Go...) 🧑‍💻



2. Identifier des bugs / vulnérabilités 🐛

```
1 contract DepositFunds {
2     mapping(address => uint) public balances;
3
4     function deposit() public payable {
5         balances[msg.sender] += msg.value;
6     }
7
8     function withdraw() public {
9         uint bal = balances[msg.sender];
10        require(bal > 0);
11
12        (bool sent, ) = msg.sender.call{value: bal}("");
13        require(sent, "Failed to send Ether");
14
15        balances[msg.sender] = 0;
16    }
17 }
```

3. Écrire un rapport d'audit 📝

3.2.5 Reentrancy Safeguards Needed In `GClaimManager`

Severity: *Medium Risk*

Context: `GClaimManager.sol#L77-104`

Situation: In function `exit()` of `GClaimManager.sol`, the burning on the tokens is done at the end.

This is not according to the Checks-Effects-Interactions pattern and no `non-Reentrant` modifier is used.

Reentrancy is possible with a malicious adapter contract and is potential in other situations as well.

```
function exit(address adapter, uint48 maturity, uint256 uBal) external {
    ...
    // External call
    uint256 collected = Claim(claim).collect();
    ...
    // External call
    uint256 tBal = uBal.fdiv(gclaims[claim].totalSupply(), total);
    ...
    // External call
    ERC20(Adapter(adapter).target()).safeTransfer(msg.sender, tBal);
    ...
    // External call
    ERC20(claim).safeTransfer(msg.sender, uBal);
    // Burn the user's gclaims
    // Completed at the very end with potential reentrancy above
    gclaims[claim].burn(msg.sender, uBal);
    ...
}
```

Recommendation: To improve security, add a `nonReentrant` modifier on functions of `GClaimManager` that do external calls.

Note: `GClaimManager` will most likely be deprecated. This issue is included for completeness.

Armé comme jamais



SLITHER

Semgrep

Hardhat

aderyn

Formal Verification

Fuzzing



Testing

Static Analysis

Examining code for errors or vulnerabilities without executing it

Slither, Aderyn, Semgrep, 4naly3er

Dynamic Testing

Unit Testing

Process in which the smallest testable parts of an application are individually scrutinized for proper operation

Foundry, Hardhat

Fuzz Testing

Testing a program by inputting random data to find bugs

Stateless Fuzzing

A type of fuzz testing that doesn't consider the program's state or history

Foundry, Echidna, Diligence Fuzzing, Medusa, ItyFuzz

Stateful Fuzzing / Invariant Testing

A type of fuzz testing that considers the program's state and previous interactions, checking that certain conditions always hold true throughout the code execution

Foundry, Echidna, Diligence Fuzzing, Medusa, ItyFuzz

Formal Verification

The act of proving or disproving a given property of a system using a mathematical model

Certora

Symbolic Execution / Symbolic Testing

Converting the code to a set of mathematical expressions to explore different paths in a program

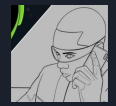
Halmos, Mythril, Manticore, hevm, EthBMC, Solidity SMTChecker, Scribble, ItyFuzz, Kontrol

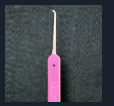
Les différents types d'audits 🔍

*En **solo** ou en **team***

Concours d'audit public 🏆

Audits privés 🕵️



Bug bounties 🏹

Cabinets d'audit 🏢

Pourquoi devenir Auditeur de Smart Contracts ? 💰

zkSync Era Code4rena contest results



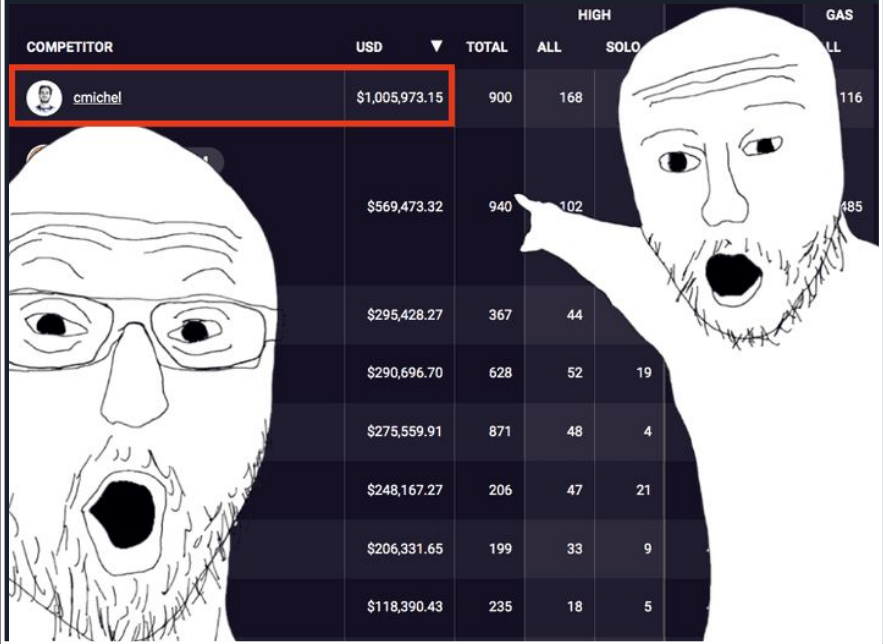
liveactionllama | C4 01/17/2024 10:58 PM

Top 5 Awards for **zkSync Era**:

$502,041.99 USDC » @xuwinnie
$157,696.85 USDC » @ChainLight
$140,480.81 USDC » @Audittens
$38,573.19 USDC » @minhtrng
$25,342.88 USDC » @erebus

Code4rena leaderboard

COMPETITOR	USD ▼	TOTAL	HIGH ALL	HIGH SOLO	GAS
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	
	$275,559.91	871	48	4	
	$248,167.27	206	47	21	
	$206,331.65	199	33	9	
	$118,390.43	235	18	5	



Immunefi leaderboard

	Name	Whitehat Score	Total Earnings	Paid Reports
1	Barracuda3172	958	$13,210,000	5
2	RetailDdene2946	723	$10,020,000	2
3	PwningEth	590	$8,000,000	8
4	GothicShanon89238	257	$3,181,150	15
5	Bobface	253	$3,391,900	8

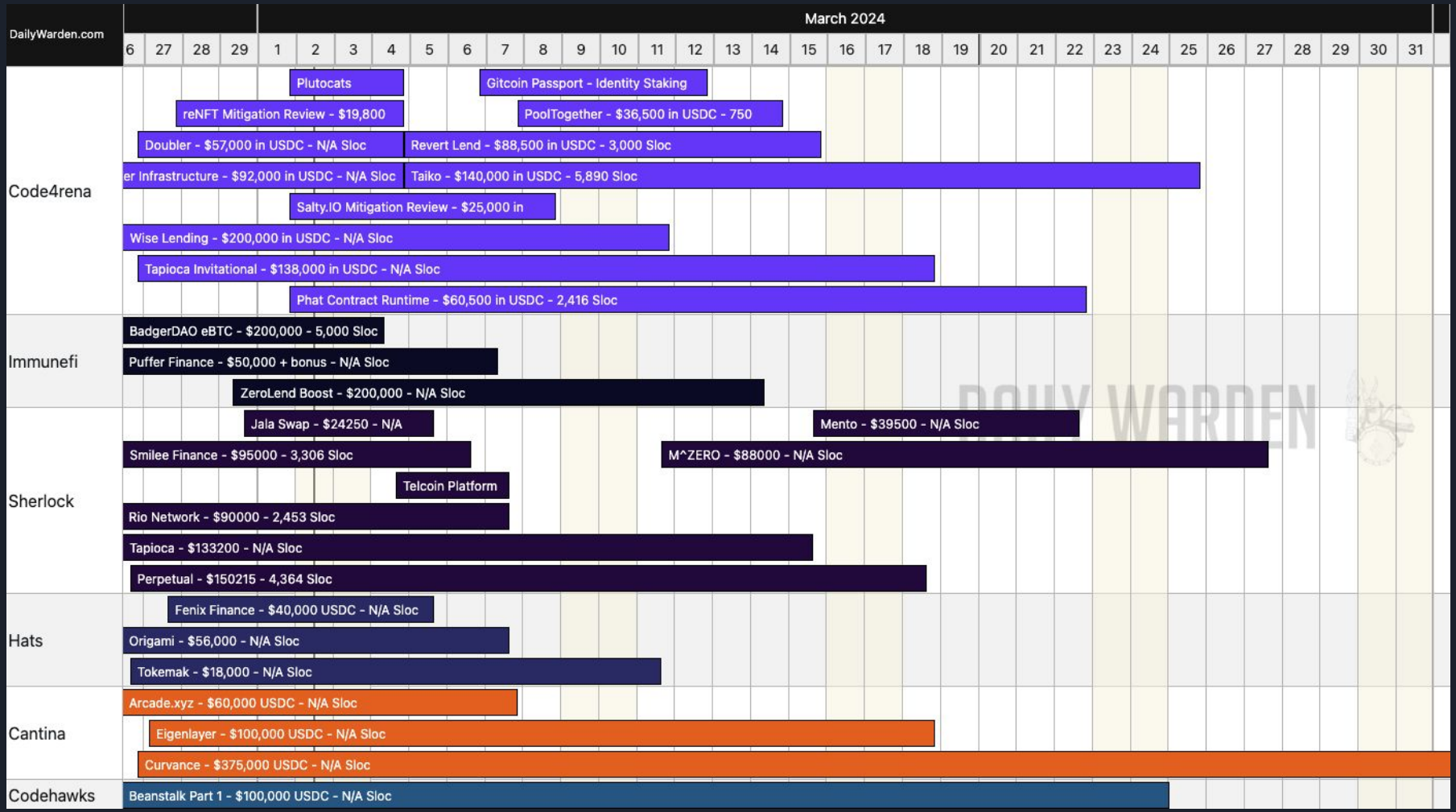
Pourquoi devenir Auditeur de Smart Contracts ? 🕵️

Sécuriser le Web3 🛡️

Méritocratique 💪

Défi intellectuel 🧠

Demande croissante 📈



dailywarden.com

Comment devenir Auditeur de Smart Contrats ? 👩‍🎓

1️⃣ Apprendre la programmation 👩‍💻

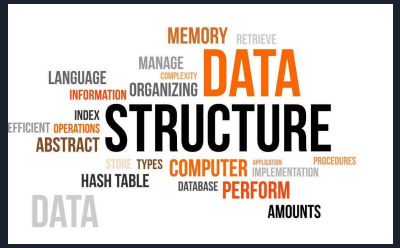
2️⃣ Apprendre la Blockchain ⛓️

3️⃣ Apprendre la sécurité Web3 🛡️

4️⃣ Pratiquer, pratiquer, pratiquer 💪

1. Apprendre la programmation 🧑‍💻

Informatique



JavaScript / Python

JS

Harvard CS50



2. Apprendre la Blockchain ⛓️

EVM



Solidity



Cyfrin Updraft



3. Apprendre la sécurité Web3 🛡️

Cyfrin Updraft 🎓



CTF 🚩

The Ethernaut

Damn Vulnerable DeFi



ONLYPWNER

Rapports d'audit 📄



SOLODIT

4. Pratiquer, pratiquer, pratiquer 💪

Concours d'audit public 🏆



Étudier le rapport d'audit 📄



SOLODIT

Soliditors

La communauté et le podcast francophone des passionnés et professionnels de la sécurité Web3 ⛓️🛡️🇫🇷

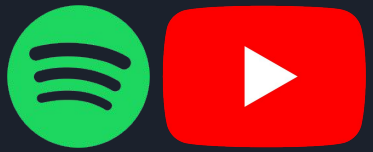
Août 2023 🐣



140 💃🕺



Podcast 🎙️🇫🇷



DevConnect 2023 Istanbul 🇹🇷



EthCC 2024 Bruxelles 🇧🇪



Devcon 2024 Bangkok 🇹🇭



Nous suivre

