# Ethereum France Soliditors





Les Amis d'Ethereum France #17

**Sécurité et audit de smart contracts**

# La sécurité dans le Web3

🥷 hack Web2 ➡ données ➡ dark web ➡ 💰

🥷 hack Web3 ➡ 💰💰💰

Code immuable
📝

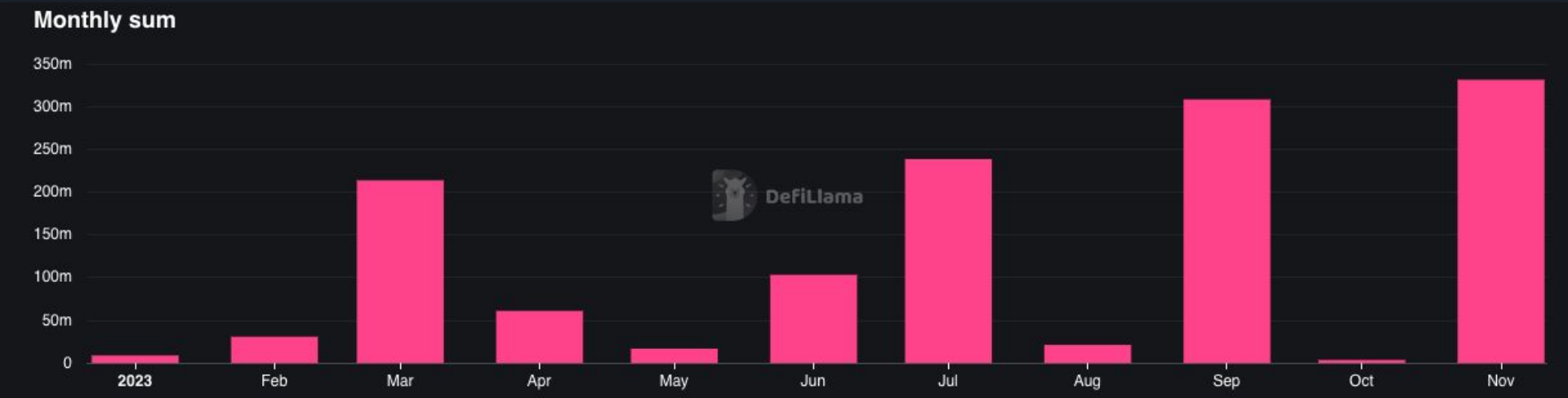
Industrie immature
🐣

Manque de crédibilité
🤡

# Les hacks Web3 en 2023

Valeur totale hack depuis **2016**
**7,5 Md $**

Valeur totale hack en **2023**
**1,3 Md $**



# Les 10 principaux vecteurs d'attaque DeFi 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*

# Soliditors

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

**Août 2023** 🐣




**110** 💃🕺

**Meetups mensuels**

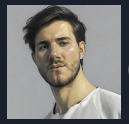
**DevConnect 2023 Istanbul**

**EthCC[7] 2024 Bruxelles** 🇧🇪

# Programme des intervenants



**Lil.eth** De Pentester Web2 à Auditeur Web3



**Cergyk** Whitehatting dans le Web3 : qui veut sauver des millions ?



**Beirao** Le framework pour un protocole sécurisé



**Merlin** Comment sécuriser un protocole DeFi ?



# De Pentester Web2 à Auditeur Web3

# Du Web2 au Web3



**Pentester**

Thales – 2 ans



**SOC Analyst**

Airbus – 6 mois




**VOC Analyst**

Orange CyberDefense – 1 an et demi



**Smart Contract Auditor**

Home - Juin 2023 - ∞

# Quels sont les enjeux d’une faille/d’un hack ?

## Web2

**Argent = Extérieur au système**

**Image de marque**

**Intégrité/Confidentialité des données**

**Identité**

## Web3

**Argent = partie intégrante du système**

**Clap de fin ⇒ Rekt Leaderboard**

**Argent des utilisateurs, du protocol**

**Anonymat**

# Patching vs Preventing

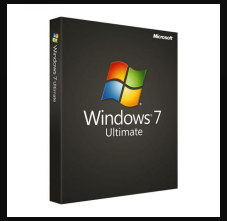


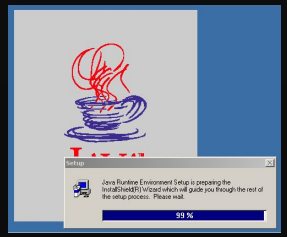
Web2 Security Expert



Web3 Security Expert

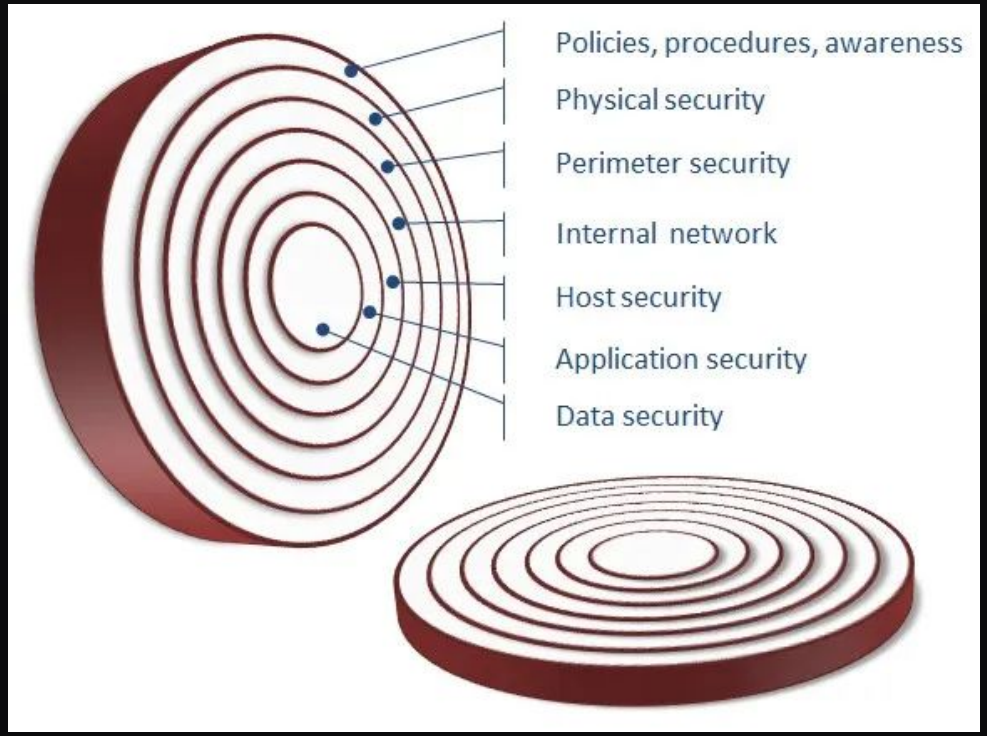
# Patching : pourquoi ça fonctionne comme ça en web2 ?









# Prévention : pourquoi ça fonctionne comme ça en web3 ?



# Connaissances Web2 VS Connaissances Web3

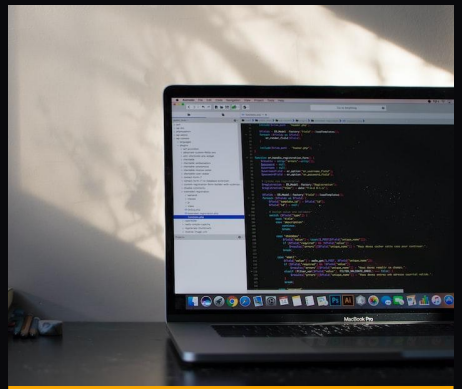
Peu de connaissances dans beaucoup d’endroits

VS

Beaucoup de connaissances dans peu d’endroits



# Connaissances Auditeur Smart Contract



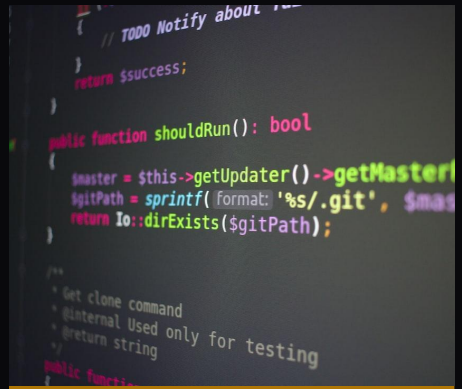
**Solidity**

Langage orienté objet, heritage, librairies , ...



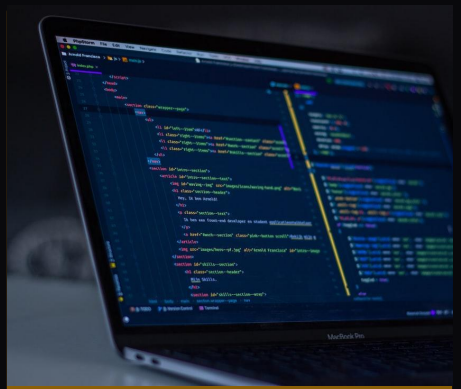
**EVM**

Machine d'état, External Owned account, Contrat, bytecode, GAS, sender, msg.value , ...



**Finance**

Emprunts, liquidités,Options, staking, prêts  ROI, effet de levier ...

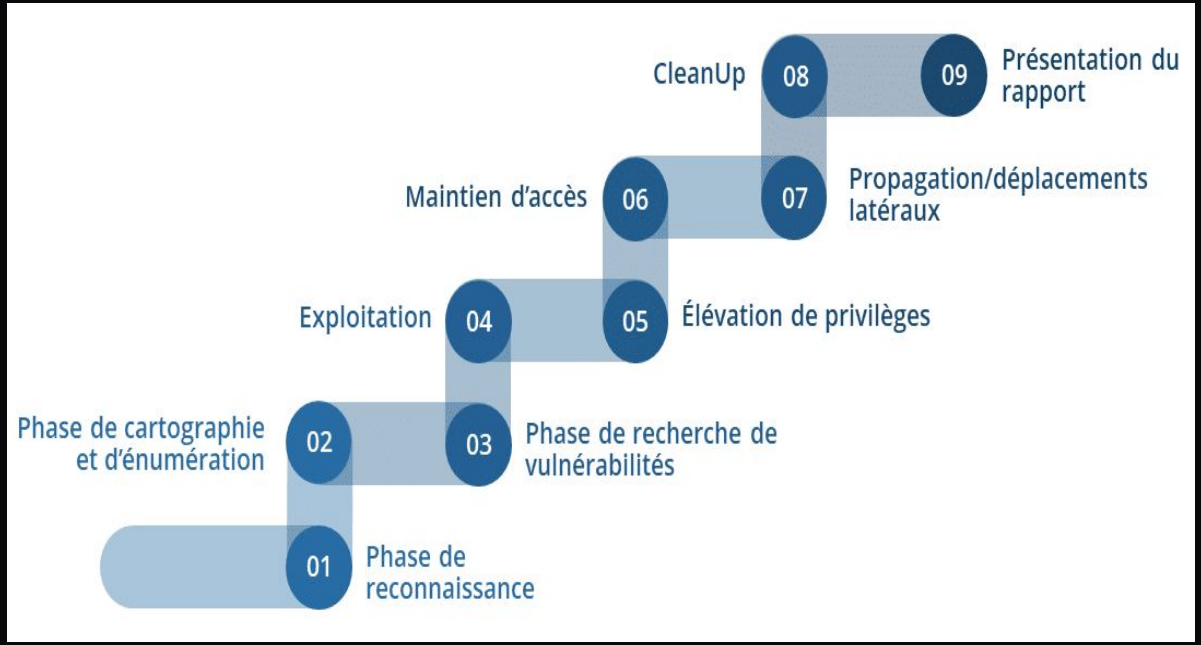


**Maths**

AMM, nombres binaires, hexadecimal ,cryptography , hashs

# Pentester Web2 : quel focus ?

## Étapes d'un Pentest




## Outils Pentest




Et beaucoup d'autres ...

# Auditeur Web3 : quel focus ?

Étapes d’un audit de smart contract

Compréhension

Compromission

Génération du rapport

Controle des patchs

Principaux Outils :







Et quelques d’autres ...

# Conclusion



# Whitehat dans le Web3

Qui veut sauver des millions ?

# About me

- Independent Smart contract auditor
- Ex-Web2 software eng (8 years)
- Building upgradehub.xyz
- ZK enjoyoor

twitter: @cergyk1337

# Spécificités des smart-contracts

Les contrats sont :

- Déployés une fois
- Difficiles à changer (upgrades)
- Code public
- Intéractions publiques
- Intéroperables



# What is MEV? (Baby don't hurt me)

# L'acronyme MEV c'est ?

a/ Maximal extractable value

b/ Miner extractable value

c/ Malicious exploit vulnerability

d/ Medical emergency vehicle

# L'acronyme MEV c'est ?

a/ Maximal extractable value (*post-merge)

b/ Miner extractable value

c/ Malicious exploit vulnerability

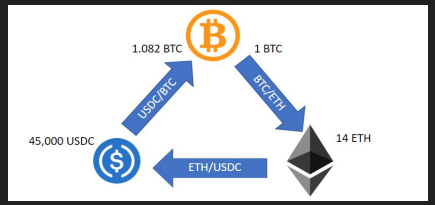
d/ Medical emergency vehicle

# Formes multiples

- Arbitrage sur les DEX

- Liquidations

- Sandwich









# Generalized MEV bots

- Simulation des tx dans le mempool

- Remplacement de certaines variables (receiver address)

Rapport avec la sécurité ?

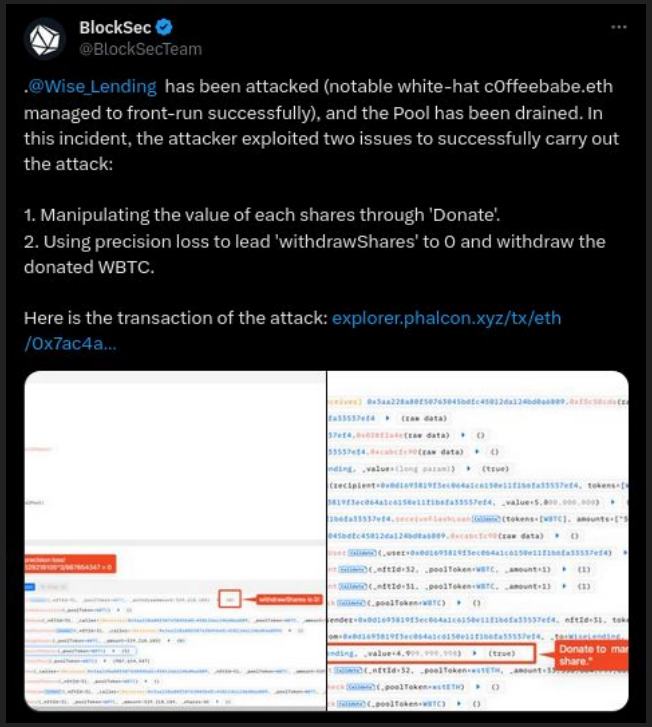
- Front-run de hacks



# Hack front-run

- Blocksec
- c0ffeebabe.eth






A smart contract audit firm called BlockSec was the first to notify Saddle. Interestingly, BlockSec was able to rescue $3.8 million from the exploiters with an "internal system" that can detect and front-run hacking incidents using off-chain arbitrage bots called flashbots, it told The Block in a Twitter message.

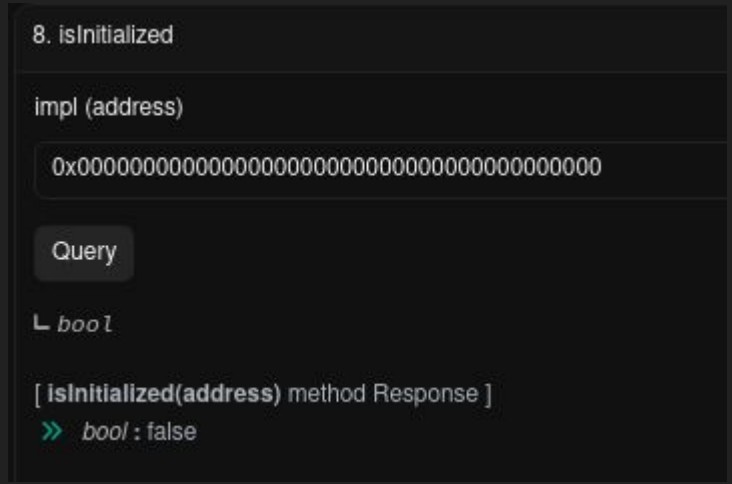
# Outils / OSINT

# Block explorer

Wormhole vulnerability

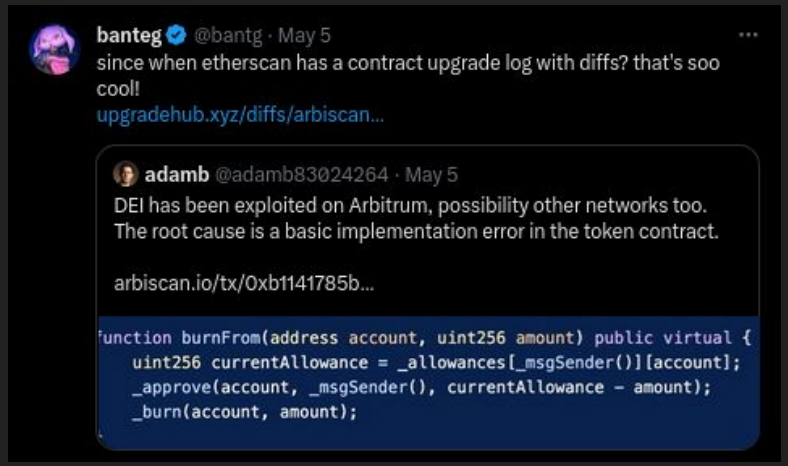
**Bounty historique de 10 000 000$**

Variable non initialisée (visible sur etherscan.io) :




Au-delà du code, le déploiement aussi est crucial

# Upgradehub.xyz

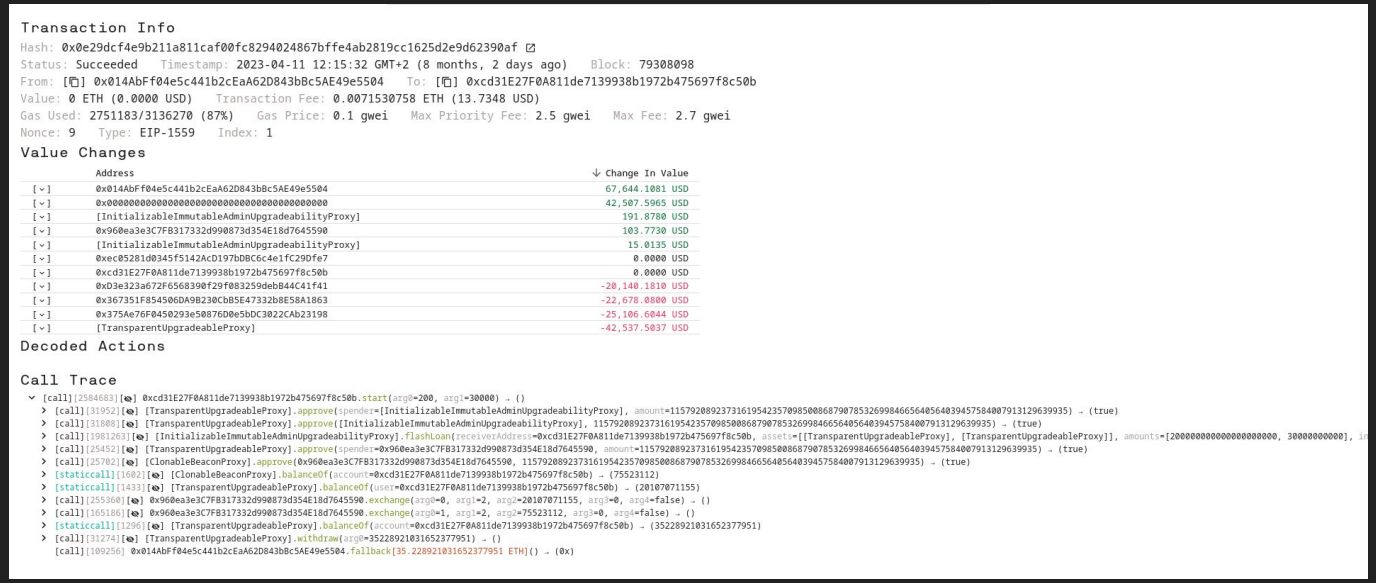



```
315
316     function burn(uint256 amount) public virtual {
317         _burn(_msgSender(), amount);
318     }
319
320     function burnFrom(address account, uint256 amount) public virtual {
321         uint256 currentAllowance = _allowances[_msgSender()][account];
322         _approve(account, _msgSender(), currentAllowance - amount);
323         _burn(account, amount);
324     }
325
326     function _transfer(
```

Permet de voir les changements de code sur des contrats capables de mises à jour (upgradeables)

# Transaction tracers (forensic)

Openchain.xyz / Phalcon / Tenderly :



Transaction Info

Hash: 0x0e29dcf4e9b211a811caf00fc8294024867bffe4ab2819cc1625d2e9d62390af

Status: Succeeded  Timestamp: 2023-04-11 12:15:32 GMT+2 (8 months, 2 days ago)  Block: 79308098

From: [] 0x014AbFf04e5c441b2cEaA62D843bBc5AE49e5504  To: [] 0xcd31E27F0A811de7139938b1972b475697f8c50b

Value: 0 ETH (0.0000 USD)  Transaction Fee: 0.0071530758 ETH (13.7348 USD)

Gas Used: 2751183/3136270 (87%)  Gas Price: 0.1 gwei  Max Priority Fee: 2.5 gwei  Max Fee: 2.7 gwei

Nonce: 9  Type: EIP-1559  Index: 1

Value Changes

| | Address | ↓ Change In Value |
|---|---|---|
| [~] | 0x014AbFf04e5c441b2cEaA62D843bBc5AE49e5504 | 67,644.1081 USD |
| [~] | 0x0000000000000000000000000000000000000000 | 42,507.5965 USD |
| [~] | [InitializableImmutableAdminUpgradeabilityProxy] | 191.8780 USD |
| [~] | 0x960ea3e3C7FB317332d990873d354E18d7645590 | 103.7730 USD |
| [~] | [InitializableImmutableAdminUpgradeabilityProxy] | 15.0135 USD |
| [~] | 0xec05281d0345f5142AcD197bDBC6c4e1fC29Dfe7 | 0.0000 USD |
| [~] | 0xcd31E27F0A811de7139938b1972b475697f8c50b | 0.0000 USD |
| [~] | 0xD3e323a672F6568390f29f083259debB44C41f41 | -20,140.1810 USD |
| [~] | 0x367351F854506DA9B230CbB5E47332b8E58A1863 | -22,678.0800 USD |
| [~] | 0x375Ae76F0450293e50876D0e5bDC3022CAb23198 | -25,106.6044 USD |
| [~] | [TransparentUpgradeableProxy] | -42,537.5037 USD |

Decoded Actions

Call Trace

[call][2584683][] 0xcd31E27F0A811de7139938b1972b475697f8c50b.start(arg0=200, arg1=30000) → ()

[call][31952][] [TransparentUpgradeableProxy].approve(spender=[InitializableImmutableAdminUpgradeabilityProxy], amount=1157920892373161954235709850086879078532699846656405640394575840079131296399 35) → (true)

[call][31808][] [TransparentUpgradeableProxy].approve([InitializableImmutableAdminUpgradeabilityProxy], 1157920892373161954235709850086879078532699846656405640394575840079131296399 35) → (true)

[call][1981263][] [InitializableImmutableAdminUpgradeabilityProxy].flashLoan(receiverAddress=0xcd31E27F0A811de7139938b1972b475697f8c50b, assets=[[TransparentUpgradeableProxy], [TransparentUpgradeableProxy]], amounts=[2000000000000000000, 3000000000], in

[call][25452][] [TransparentUpgradeableProxy].approve(spender=0x960ea3e3C7FB317332d990873d354E18d7645590, amount=1157920892373161954235709850086879078532699846656405640394575840079131296399 35) → (true)

[call][25702][] [ClonableBeaconProxy].approve(0x960ea3e3C7FB317332d990873d354E18d7645590, 1157920892373161954235709850086879078532699846656405640394575840079131296399 35) → (true)

[staticcall][1602][] [ClonableBeaconProxy].balanceOf(account=0xcd31E27F0A811de7139938b1972b475697f8c50b) → (75523112)

[staticcall][1433][] [TransparentUpgradeableProxy].balanceOf(user=0xcd31E27F0A811de7139938b1972b475697f8c50b) → (20107071155)

[call][255360][] 0x960ea3e3C7FB317332d990873d354E18d7645590.exchange(arg0=0, arg1=2, arg2=20107071155, arg3=0, arg4=false) → ()

[call][165186][] 0x960ea3e3C7FB317332d990873d354E18d7645590.exchange(arg0=1, arg1=2, arg2=75523112, arg3=0, arg4=false) → ()

[staticcall][1296][] [TransparentUpgradeableProxy].balanceOf(account=0xcd31E27F0A811de7139938b1972b475697f8c50b) → (3522892103165237795 1)

[call][31274][] [TransparentUpgradeableProxy].withdraw(arg0=3522892103165237795 1) → ()

[call][109256] 0x014AbFf04e5c441b2cEaA62D843bBc5AE49e5504.fallback[35.2289210316523779 51 ETH]() → (0x)

# Divulgation

# Je tombe sur une faille mettant à risque des fonds

a) J'agis au plus vite pour mettre les fonds en sécurité

b) J'avertis le protocole au plus vite afin qu'ils puissent réagir

c) Je vérifie la faille en effectuant une transaction sur le testnet

d) Je poste sur twitter pour alerter la communauté

# Je tombe sur une faille mettant à risque des fonds

a) J'agis au plus vite pour mettre les fonds en sécurité

b) J'avertis le protocole au plus vite afin qu'ils puissent réagir

c) Je vérifie la faille en effectuant une transaction sur le testnet

d) Je poste sur twitter pour alerter la communauté

# Prise de contact

Responsible disclosure program

-  Immunefi

- Email dédié (security@company.xyz)

Sinon

- Directement sur discord/telegram

# Rapport

Rapport :

- Scénario de la vulnérabilité

- Sévérité (conditions, impact)

PoC (Proof of Concept) :

- Script qui exploite la vulnérabilité sur un état copié (fork) de la Blockchain

# SEAL 911

Alliance de whitehats/protocols



# Mitigation

# Mécanismes de protection

- Pause
- Emergency withdraw
- Allowlist & Blocklist
- Rate limiters
- Settlement timelocks
- Frontrunning

Credit : Openzeppelin blog

# Conclusion

# Le framework pour une dApp sécurisée

Beirao, 18 décembre 2023

# Présentation



@0xBeirao

# Motivations

## The Framework to help you securing your protocol

Beirao
@0xBeirao

TAGS
BLOG FRAMEWORK

PREVIOUS ARTICLE
#1 Solo Review: Frak.id swap feature

NEXT ARTICLE
The ultimate security checklist

← Back to the blog



- Difficile de se retrouver dans toutes les solutions disponibles sur le marché
- Difficile de savoir dans quel ordre faire les choses
- Difficile de savoir quoi prioriser en fonction du budget

# Déployer des smart contracts c'est très cher

## SaaS*

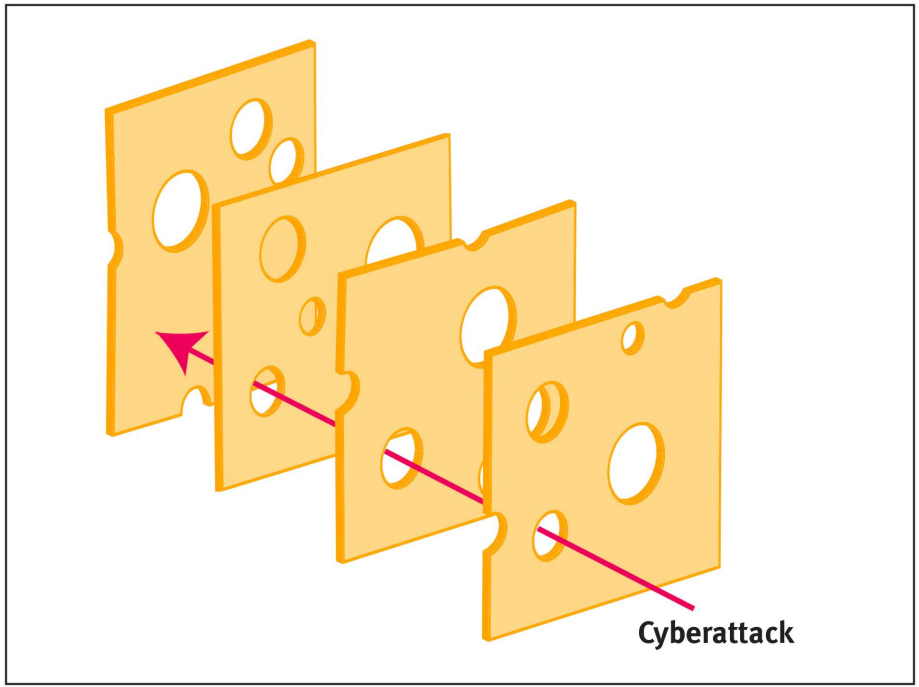
- Développer

## Dapp

- Développer (**Complètement**)
- **Tester**
- **Auditer**
- **Avoir un programme de bug bounty**

* : Pour un MVP

# Couches de sécurité

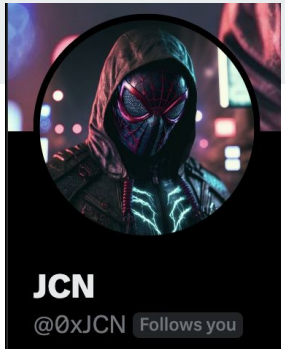


# Tout part de la team

On veut :

- un protocole “secure by design”
- du code propre et documenté
- des devs qui s’y connaissent en sécurité
- des reviews internes
- 100% de coverage
- l’utilisation de fuzzing/vérification formelle => identification des invariants




# Optimisation de gas

**Vous devez faire cette étape si:**

- Déploiement sur Ethereum mainnet
- Vous avez beaucoup de moyen

**Les solutions:**

- Firm
- Solo gas review

**Prix**: entre 2k$ et 10k$ (pour 1000 lignes de code)

# Consulting




**Conseillé de faire cette étape**

**Les solutions:**

- Solo reviews
- Traditional security firms
- OG trad firms

**Prix**: entre 3k$ et 160k$

# Fuzzing/Formal verification

**Vous devez faire cette étape si:**

- Vous avez beaucoup de moyen

**Les solutions:**

- Certora
- Solo auditor

**Prix**: entre 80k$ et 160k$

# Contests

**Très conseillé de faire un contest sauf si le code doit rester confidentiel**

**Les solutions:**

- Code4rena
- Sherlock
- Cantina
- CodeHawks
- Hats finance

**Prix**: entre 50k$ et 140k$

# Decentralized OGs

**Vous devez faire cette étapes si:**

- il vous reste de l'argent

**Les solutions:**

- Spearbit
- Trust security

**Prix**: entre 40k$ et 160k$

# Bug bounty

**Vous devez faire cette étapes**

**Les solutions:**

- Immunefi
- Hats Finance
- Solo

**Prix**: 10% de la TVL pour un critique



# Sécuriser 1000 lines = [50k, 500k] $

# Que faire si je n'ai pas de tune? (< 50k$)

=> Full solo reviews







(Invitational sur code4rena)

# Il ne reste plus qu'à préparer le pire

Vérifier le bytecode on-chain

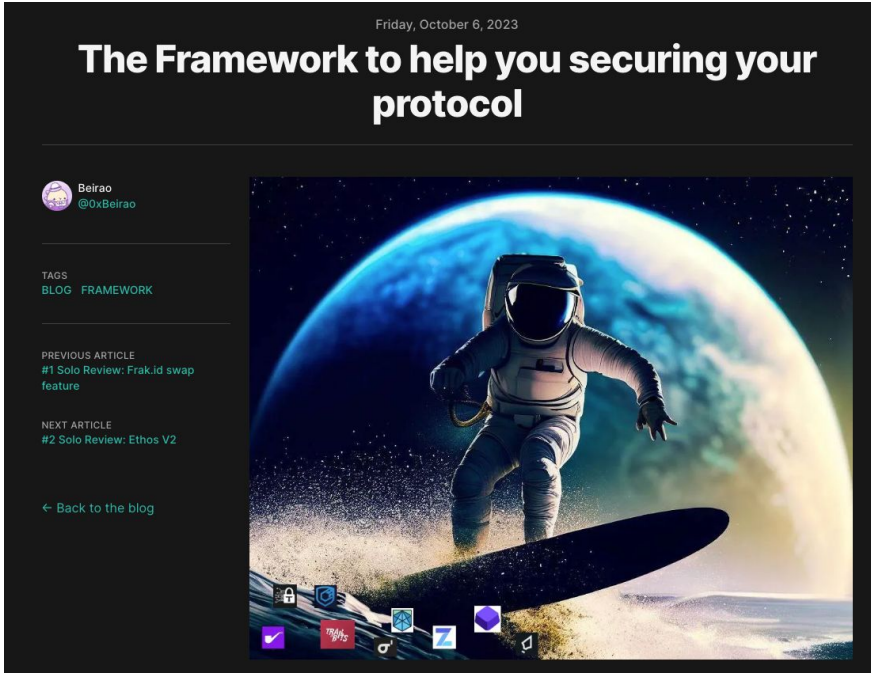
Monitorer la blockchain

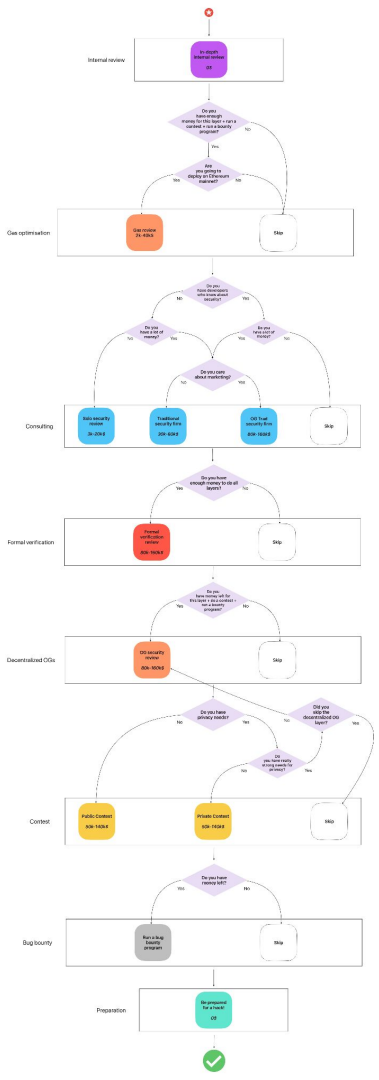
Établir des procédures en cas de hack

Rédiger à l'avance des messages à envoyer aux hackers

Prier

# Mon site : beirao.xyz



# Merci de votre attention




Mon site : **beirao.xyz**

