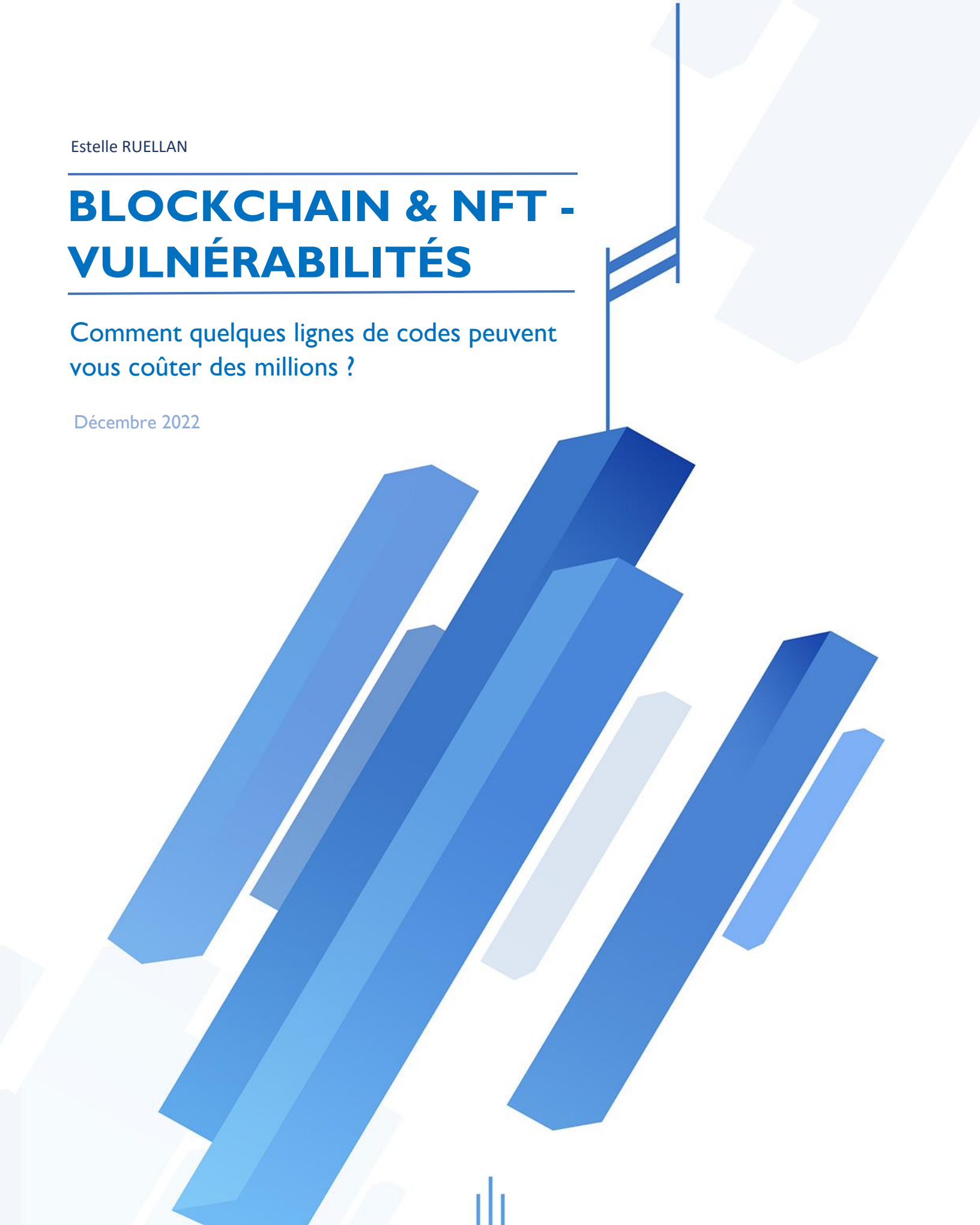
Estelle RUELLAN

BLOCKCHAIN & NFT - VULNÉRABILITÉS

Comment quelques lignes de codes peuvent vous coûter des millions ?

Décembre 2022



RÉSUMÉ EXÉCUTIF

L'arrivée des contrats intelligents au sein de la blockchain a permis le développement de nombreux produits financiers décentralisés (DeFi). Les premiers NFT ont vu le jour en 2014(Futura Sciences., 2022, 12 juin), mais c'est seulement en 2017 lorsqu'Ethereum a pris de l'ampleur que les NFT sont rapidement devenus l'un des produits DeFi les plus populaires. En 2021, le marché des NFT a été estimé à plus de 10 milliards de dollars américains. Un tel afflux de capitaux suscite beaucoup d'attention, tant bienveillante que malveillante. Malheureusement, le marché des NFT n'étant encore qu'à ses débuts, il existe de nombreuses vulnérabilités au sein de l'écosystème NFT. Vulnérabilités qui ont été très couteuses ou qui pourraient se révéler l'être si jamais elles sont exploitées. Ainsi, ce rapport fait un état des lieux des principales failles et vulnérabilités présentes au sein de l'écosystème de la blockchain et des NFT.

En premier lieu, il est question des vulnérabilités liées à la blockchain. Ce sont les contrats intelligents qui constituent le gros des vulnérabilités de cette section. Les nombreuses erreurs logiques dans la construction (programmation) des contrats rendent possible la quasi-totalité des vulnérabilités recensées. Les contrats étant disponibles sur la blockchain, prêter une plus grande attention aux détails des contrats avant de les utiliser réduirait significativement les chances d'attaque. Ensuite, les risques associés aux services d'oracle (*oracle issue*) peuvent être facilement réduit en augmentant la vigilance lors de la sélection du service. Un service d'oracle décentralisé est à privilégier.

En deuxième lieu, l'économie des NFT, les failles des plateformes de trading et les attaques connues sont recensées. Les plateformes de trading sont des facilitateurs d'attaque. Les attaques recensées usurpent l'identité des plateformes afin d'amener l'utilisateur à donner l'accès à son *wallet* à l'attaquant ou à signer des transactions malicieuses. Une vigilance accrue des utilisateurs lors de la signature de transaction ou la réception de mails douteux est préconisée contre ce type d'attaque. Les paramètres et régulations des plateformes sont également problématiques. Enfin, la permission de nombreuses extensions dans la création de NFT ou encore la mauvaise conception du système de mise en vente de NFT ont donné lieu à de nombreuses attaques. De tels paramètres méritent une plus grande réflexion lors de leur implémentation au sein des plateformes.

TABLE DES MATIÈRES

INTRODUCTION

Depuis la naissance de la blockchain, un nombre grandissant d'individus se sont engagés dans la conception et le développement de divers produits financiers décentralisés (DeFi). L'intégration du contrat intelligent au sein de la blockchain n'a fait que renforcer l'engouement entourant la blockchain et ses dérivés. Effectivement, l'avènement des contrats intelligents a permis d'élargir le champ des possibles dans le monde cryptographique. Parmi toutes utilisations qui ont été découvertes aux contrats intelligents, il semblerait que le jeton non fongible (NFT) soit devenu l'un des produits DeFi les plus populaires.

Le NFT diffère des crypto-monnaies classiques telles que Bitcoin par ses caractéristiques intrinsèques. Le bitcoin est une cryptomonnaie « standard », c'est-à-dire que tous les *coins* sont équivalents et indiscernables, ils sont fongibles. En revanche, un NFT est unique et ne peut être échangé à l'identique, il est donc non fongible. Cette propriété intrinsèque rend les NFT appropriés dans l'identification de quelque chose ou de quelqu'un de manière unique. Plus précisément, en utilisant des NFT sur des contrats intelligents, un créateur peut facilement prouver l'existence et la propriété d'actifs numériques sous forme de vidéos, d'images, d'arts, de billets d'événement, etc.

Ces dernières années, les NFT ont suscité une attention remarquable au sein des communautés industrielles et scientifiques. En 2021, le marché des NFT a atteint une valeur totale estimée à environ 11 milliards USD (ELLIPTIC., 2022). En date du 30 novembre, le volume moyen de transactions en 24 heures sur le marché NFT est de 1 785 291 784 USD[1], tandis que le volume moyen de transactions en 24 heures de l'ensemble du marché des cryptomonnaies est de 60 663 648 285 USD. Au mois de novembre, près de 543 000 ventes de NFT ont eu lieu pour un total de 397 millions USD[2].

Bien que les NFT aient un potentiel important d'impact sur les marchés décentralisés actuels et les futures opportunités commerciales, les technologies NFT n'en sont encore qu'à leurs débuts. En tant que marché naissant, un audit et une réglementation solides et efficaces n'ont pas suivi le rythme de l'afflux rapide et massif de capitaux. Ce flux massif de capitaux a attiré et attire encore beaucoup d'attentions malveillantes.

" Wherever there is power, greed, and money, there is corruption."

(Ken Poirot)

Ainsi, ce rapport a pour but de présenter les principales failles du système qui une fois exploitées, ont pu couter très cher à certains. Ce rapport fera donc l'état des lieux des failles et vulnérabilités existantes au sein de l'écosystème des NFT. Tout d'abord, les fondements techniques de la blockchain nécessaires à la compréhension du rapport seront

[1] https://www.coingecko.com/en/categories/non-fungible-tokens-nft
[2] https://nonfungible.com/market-tracker

expliqués. Ensuite, les vulnérabilités liées à la blockchain seront exposées. À la suite de cela, l'économie des NFT ainsi que ses failles et les attaques existantes seront discutées.

1 - LES FONDEMENTS TECHNIQUES

Cette section détaille les fondements techniques nécessaires à la compréhension du fonctionnement de la blockchain et des NFT qui sont essentiels pour la suite du rapport.

1.1 - BLOCKCHAIN

La blockchain a été initialement proposée par Nakamoto (Nakamoto., 2008) comme alternative aux systèmes centralisés. La blockchain est un type de base de données partagée qui diffère d'une base de données typique dans la manière dont elle stocke les informations. Les blockchains stockent les données dans des blocs qui sont ensuite liés entre eux et protégés via des protocoles cryptographiques. Les blocs ont certaines capacités de stockage. Au fur et à mesure que de nouvelles données arrivent, elles sont entrées dans un nouveau bloc. Une fois que le bloc est rempli de données, il est chaîné/lié au bloc précédent formant une chaîne de données chronologique connue sous le nom de blockchain. La blockchain la plus utilisée pour les NFT est Ethereum, mais c'est loin d'être la seule.

Différents blockchains supportant les NFT

De nombreuses autres blockchains sont impliquées dans les NFT. Binance Chain (BNB), Solana, World Wide Asset Exchange (WAX), Flow, Tezos, Cardana ou encore Polygon font parties de celles qui touchent aux NFT.

Mécanismes de consensus

Chaque blockchain possède son mécanisme de consensus. Ce mécanisme est essentiel au bon fonctionnement d'une blockchain puisque c'est lui qui est en charge de la vérification et validation des blocks qui constituent la blockchain. Le mécanisme de consensus permet donc au réseau de se mettre d'accord sur une version unique de l'histoire qu'est la blockchain. Par exemple, alors que Bitcoin repose sur un mécanisme nommé le *Proof-of-Work* (PoW), Ethereum repose sur le *Proof-of-Stake* (PoS).

Chaque mécanisme de consensus possède son fonctionnement qui lui est propre. La prochaine section s'attelle à expliquer les différents mécanismes de consensus existants.

Tableau 1:Blockchain et leur mécanisme de consensus respectif

BLOCKCHAIN	MÉCANISME DE CONSENSUS
Binance Chain (BNB)	PoSA
Bitcoin	PoW
Cardana	PoS
Ethereum	PoS
Flow	PoS
Polygon	PoS
Solana	PoS + PoH
Tezos	PoS
Worldwide Asset Exchange (WAX)	PoS

La plupart des blockchains supportant les NFT fonctionnant selon le *Proof-of-Stake*, c'est sur ce dernier que l'analyse sera développée.

PoW

Le *Proof-of-Work* fonctionne avec un système de mineurs. Les mineurs sont en compétition dans la résolution d'un puzzle cryptographique. Le premier arrivant résoudre le puzzle ajoutera (*minera*) le nouveau block à la blockchain et obtiendra la récompense qui était à la clé : un nombre de cryptomonnaie (BTC ou ETH par exemple). N'importe qui peut devenir un mineur tant qu'il possède une puissance de calcul suffisante, soit des ordinateurs assez puissants, pour résoudre un puzzle. Bien évidemment, plus un mineur a de puissance de calcul, plus il a de chance de résoudre le puzzle en premier et donc obtenir la récompense y étant associée. C'est donc la résolution du puzzle qui sert de preuve du travail fournit, d'où l'appellation *Proof-of-Work* (Ethereum., 2022, 26 septembre).

PoS

Dans le *Proof-of-Work*, les mineurs prouvent qu'ils ont un capital à risque en dépensant de l'énergie et de la puissance de calcul pour miner un nouveau bloc. Dans le *Proof-of-Stake*, les validateurs mettent en jeu du capital sous la forme d'ETH dans un contrat intelligent sur Ethereum. Cet ETH jalonné agit alors comme une garantie qui peut être détruite si le validateur se comporte de manière malhonnête ou paresseuse. Le validateur est alors chargé de vérifier que les nouveaux blocs propagés sur le réseau sont valides et occasionnellement de créer et de propager eux-mêmes de nouveaux blocs (Ethereum., 2022, 3 novembre).

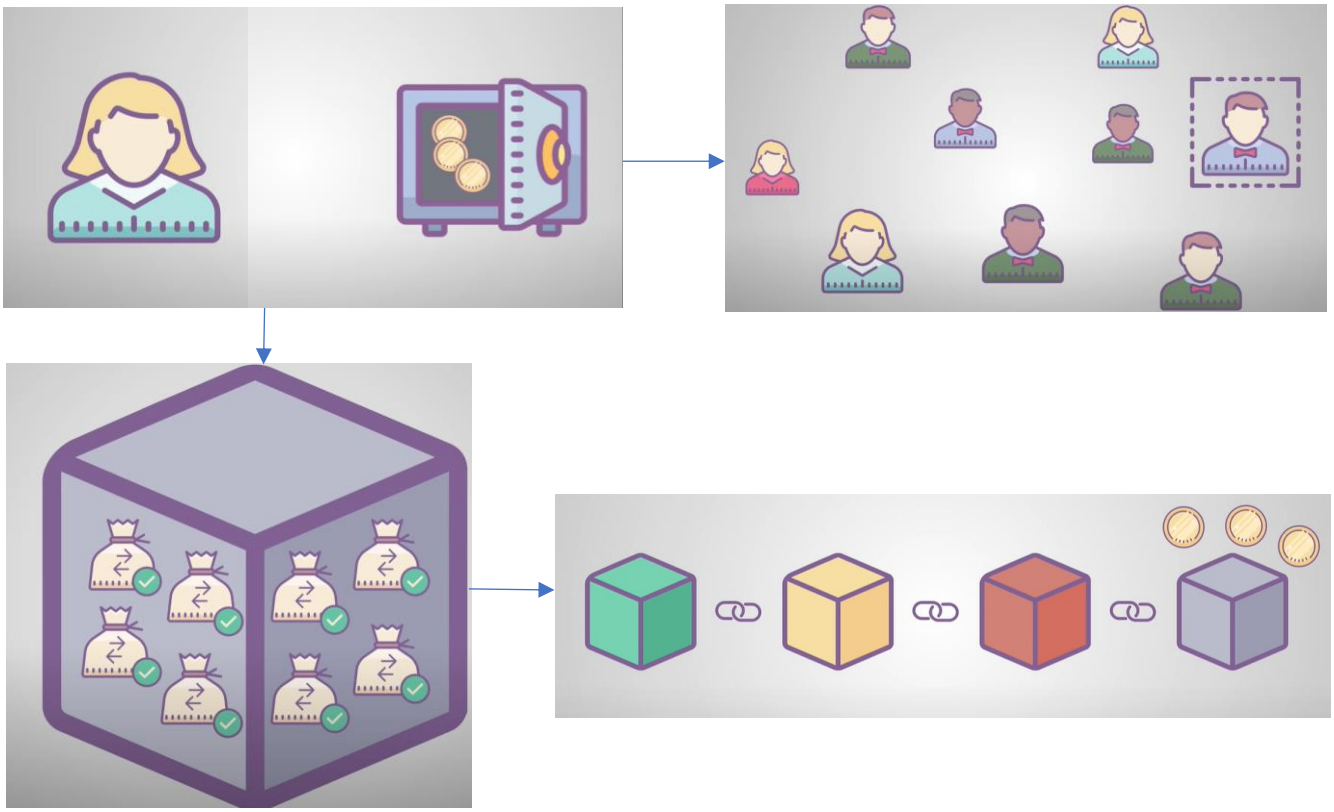


Figure 2: Schéma de fonctionnement du PoS

Le PoS fonctionne selon un processus d'élection dans lequel un nœud est choisi au hasard[3] pour valider le bloc suivant. Les validateurs ne sont pas choisis complètement au hasard ; pour devenir un validateur, un nœud doit déposer un certain nombre d'ETH en tant que mise. Un peu comme un dépôt de garantie. La taille de la mise détermine la chance qu'un validateur soit choisi pour forger le bloc suivant. Une fois choisi, un validateur vérifiera si toutes les transactions du bloc sont effectivement valides et si tout est vérifié, le nœud signera le bloc et l'ajoutera à la blockchain. En récompense, le validateur reçoit les frais associés aux transactions qui sont dans le bloc (Ethereum., 2022, 3 novembre).

Mais alors comment peut-on faire confiance aux validateurs du réseau ? C'est là que le jalonnement entre en jeu. Les validateurs perdront une partie de leur mise s'ils approuvent des transactions frauduleuses. Tant que l'enjeu est supérieur à ce que le validateur tire des

[3] Les validateurs sont choisis de manière pseudo aléatoire en fonction : du montant mis en jeu, l'âge de la mise et la santé du nœud.

frais de transaction, cela l'incite à effectuer correctement son travail. Parce que s'il ne le fait pas, il perdra plus qu'il ne gagnerait. C'est une motivation financière.

Adresses et transactions

L'adresse et la transaction sont des concepts essentiels de la blockchain. Une adresse blockchain est un identifiant unique permettant à un utilisateur d'envoyer et de recevoir des actifs. De la sorte, une adresse est similaire à un compte bancaire. Elle se compose d'un nombre fixe de caractères alphanumériques générés à partir d'une paire de clé publique et de clé privée. Pour transférer des NFT, le propriétaire doit prouver qu'il est en possession de la clé privée correspondante et envoyer les actifs à une ou plusieurs autres adresses avec une signature numérique correcte. Cette opération simple est généralement effectuée à l'aide d'un portefeuille (*wallet*) de crypto-monnaie et est représentée comme l'envoi d'une transaction par le biais des contrats intelligents (*smart contract*) (Ruan., 2022).

Smart contracts

Un contrat intelligent est un contrat qui utilise un langage informatique au lieu d'un langage juridique pour enregistrer les termes et conditions d'un contrat. Celui-ci est automatiquement exécuté par le programme en question. Déployé sur un bloc, le contrat est public, n'importe qui ayant accès à la blockchain peut en consulter le contenu. Un contrat possède également les propriétés d'inviolabilité ainsi que le fonctionnement perpétuel de la blockchain. Par rapport aux contrats traditionnels, les contrats intelligents sont fiables, sécurisés, efficaces et ne nécessitent aucun arbitrage par un tiers (Ethereum., 2022, 1 septembre).

En résumé, les contrats intelligents sont des programmes informatiques qui vivent sur la blockchain. Ils peuvent s'exécuter automatiquement. Vous pouvez suivre leurs transactions, prédire comment ils agissent et même les utiliser sous un pseudonyme. Mais à quoi servent-ils ? Les contrats intelligents peuvent faire essentiellement tout ce que font les autres programmes informatiques. Ils peuvent effectuer des calculs, créer de la monnaie, stocker des données, créer des NFT, envoyer des communications et même générer des graphiques. Voici quelques exemples d'utilisation de ces contrats :

- *Stablecoins*
- Créer et distribuer des NFT
- Un bureau de change automatique et ouvert 24/7
- Une police d'assurance autonome

Les *Non Fungible Tokens* ou NFT

Le NFT est un type particulier de jeton sur une blockchain. NFT fait généralement référence à un contrat intelligent émis sur la base de la norme Ethereum ERC-721. Plus simplement,

un NFT est une propriété non fongible, c'est-à-dire qu'il est unique et ne peut être interchangé, contrairement à un bitcoin par exemple, qui lui peut être interchangé ou même divisé en plus petites unités. Un NFT est une propriété d'actif(s) numérique(s) indivisible, irremplaçable, non interchangeable, unique et vérifiable qui est basée sur la blockchain.

Le transfert de la propriété d'un NFT se fait par le biais de contrats intelligents. Ces derniers enregistrent l'ensemble du processus de transfert sur et via la blockchain. Il existe actuellement deux normes de contrat intelligent pour NFT : ERC-721 et ERC-1155 (Ruan., 2022).

ERC-721

L'ERC-721 est le standard pour les contrats intelligents NFT. L'ERC-721 [4]définit le contenu minimum qu'un contrat intelligent doit implémenter afin de permettre la gestion, la propriété et l'échange de NFT. Il n'impose pas de norme pour les métadonnées de ces derniers ni ne limite l'ajout de fonctions supplémentaires (Ethereum., 2022, 15 août).

Le code ci-dessous liste les évènements essentiels à la compréhension des fonctionnalités du standard ERC-721, soit le transfert et l'approbation.

Tableau 2:extrait de code du ERC-721 (https://erc721.org/).

```
1 event Transfer(address indexed _from, address indexed _to, uint256
    indexed _tokenId);
2 event Approval(address indexed _owner, address indexed _approved,
    uint256 indexed_tokenId);
3 event ApprovalForAll(address indexed _owner, address indexed _operator,
    bool_approved);
```

La fonction *Transfer()*, comme son nom l'indique, permet de transférer la propriété d'un NFT de l'adresse *_from* à l'adresse *_to*. Les fonctions d'approbation, *Approuval()* et *ApprouvalForAll(),* laisse le propriétaire *_owner* définir/approuver une adresse comme opérateur *_operator* d'un ou de tous ses NFT. L'opérateur aura ensuite les droits de déplacer les NFT approuvés vers un compte tiers.

L'ERC-721 [5]possède ainsi plusieurs fonctions permettant de transférer des NFT d'un compte à un autre, ou d'obtenir le solde actuel d'un compte en NFT, le nom du propriétaire d'un NFT spécifique et le nombre total de NFT disponibles sur le réseau. En plus de celles-ci, il en existe d'autres pour, par exemple, approuver que des jetons provenant d'un compte soient déplacés par un compte tiers.

[4] https://erc721.org/
[5] https://erc721.org/

ERC-1155

L'idée derrière la norme ERC-1155 est de créer un standard de contrats intelligents pouvant représenter et contrôler de multiples types de tokens (fongibles et non-fongibles). Un jeton unique dans l'ERC-1155 est un NFT. Contrairement à ERC-721, les fonctions de transfert et d'approbation dans ERC-1155 sont capables de traiter plusieurs tokens au sein d'une seule transaction (Ethereum., 2022, 15 août). Le code ci-dessous illustre la définition de la fonction *safeBatchTranferFrom()* dans ERC-1155 (https://eips.ethereum.org/EIPS/eip-1155).

```
function safeBatchTransferFrom(
    address _from,
    address _to,
    uint256[] calldata _ids,
    uint256[] calldata _values,
    bytes calldata _data
) external;
```

Par exemple, en donnant ids= [1, 2, 5] et values= [10, 33, 6], les transferts résultants de la fonction seront : transférez 10 jetons avec l'id 1, 33 jetons avec l'id 2 et enfin 6 jetons avec l'id 5 de *_from* à *_to*.

Oracle

Les oracles sont des flux de données qui importent des données, de sources externes à la blockchain (*off-chain*), sur la blockchain pour les mettre à disposition des contrats intelligents. Le système d'Oracle est nécessaire car les contrats intelligents exécutés sur Ethereum ou autres blockchains ne peuvent pas accéder aux informations stockées en dehors de leur réseau interne (Ethereum., 2022, 8 novembre).

Les oracles permettent donc aux contrats intelligents d'obtenir les informations externes nécessaires à leur exécution. Supposons qu'Élijah parie 5 ETH sur le vainqueur de la coupe du monde de football 2022. Dans ce cas, le contrat régissant le pari a besoin d'un oracle pour confirmer les résultats de la coupe du monde 2022 et déterminer si Élijah est éligible pour un paiement.

Ainsi, les oracles étendent la valeur des applications décentralisées en rendant exécutable une plus grande variété de contrats. Par exemple, les marchés de prédiction décentralisés s'appuient sur des oracles pour fournir des informations sur les résultats avec lesquels ils peuvent valider les prédictions des utilisateurs.

Voici des exemples de données recueillies par les oracles : • Gagnants de la loterie • Catastrophes naturelles et mesure des risques • Prix et taux de change des actifs réels/crypto • Données statiques • Données dynamiques (par exemple, mesures de temps) • Conditions météorologiques • Événements politiques • Événements sportifs • Informations de géolocalisation et de traçabilité • Les accidents • Événements dans d'autres blockchains

2 - VULNÉRABILITÉS PROPRES AUX BLOCKCHAINS

2.1 - LES MÉCANISMES DE CONSENSUS – THE 51% ATTACK

L'attaque des 51% est une préoccupante lorsque le PoS est utilisé, mais il est peu probable qu'elle se produise. Sous PoW, une attaque à 51% se produit lorsqu'une entité contrôle plus de 50% de la puissance minière ou des mineurs d'un réseau et utilise cette majorité pour modifier la blockchain ou approuver des transactions frauduleuses (Ethereum., 2022, 3 novembre).

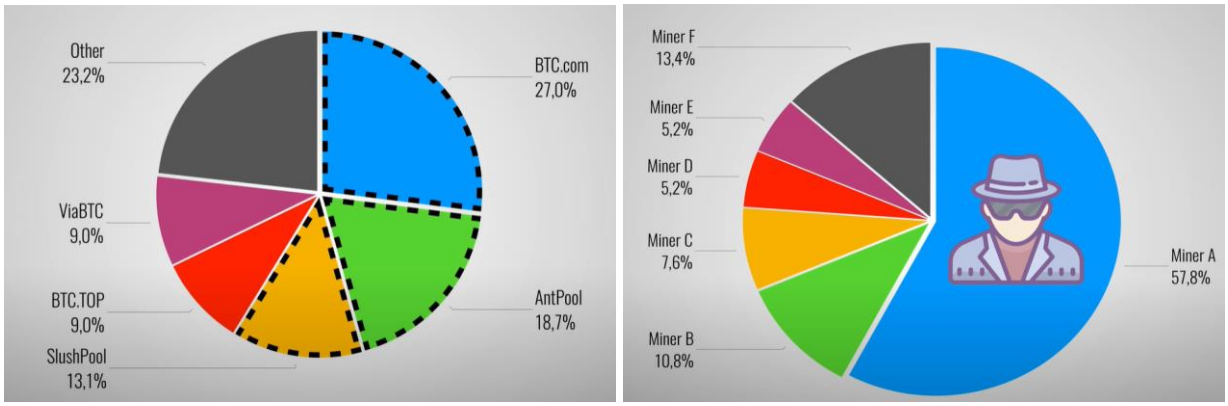


Figure 3: Illustration du principe de l'attaque des 51%

Dans PoS, un groupe ou un individu devrait posséder 51% de la cryptomonnaie mise en jeu pour lancer une attaque. Toutefois, il est extrêmement onéreux de contrôler 51 % de la cryptomonnaie jalonnée. Ainsi, le PoS est relativement sécuritaire sur le plan crypto-économique, car un attaquant tentant de prendre le contrôle de la chaîne devrait prendre le risque de détruire une quantité massive de cryptomonnaie (Jmcook.eth., 2022, 15 avril; Ethereum., 2022, 3 novembre).

Dans l'ensemble, le risque d'attaque est relativement faible. En effet, le montant risqué sur une attaque des 51% est monumental et la perte de celui-ci sert à dissuader la majorité des attaques de ce type. La couche incitative de « la carotte et le bâton » protège contre

la plupart des malversations. La probabilité qu'une telle attaque soit rentable pour l'attaquant est suffisamment faible pour être un moyen de dissuasion efficace.

2.2 - VULNERABILITIES IN SMART CONTRACT

Comme mentionné précédemment, les NFT sont créés en utilisant un contrat intelligent. Par conséquent, les attaques visant les contrats intelligents représentent une première faille directe des NFT. Les *reantrancy attacks*, *integer overflow* et *access control flaws* sont des exemples d'attaques connues. Cependant, selon Ruan (2022) La faille la plus courante des contrats intelligents NFT ne serait pas très sophistiquée, mais résiderait simplement dans le mauvais ordre logique du code constituant le contrat.

Wrong Logical Order of Code

CryptoPunks, l'un des projets NFT les plus populaires de l'histoire, a été affecté par un bug en juin 2017 qui empêchait l'ETH d'entrer dans le portefeuille du vendeur. Plus en détails, lors de la vente d'un CryptoPunk l'ETH payé par l'acheteur lui était retourné, tandis que le NFT était bel et bien transféré à son adresse (Corbella,. 2021, 28 septembre; Ruan., 2022). Plus simplement, l'erreur contenue dans le contrat a permis à l'attaquant d'acheter un Crypto Punk et de récupérer l'argent du contrat, soit d'obtenir un CryptoPunk gratuitement.

La figure *CryptoPunks Code* ci-dessous contient le code de l'ancien contrat Cryptopunks. Il est possible de voir l'erreur dans l'ordre logique du code qui est à l'origine de la vulnérabilité.

```
1  struct Offer {
2          bool isForSale;
3          uint punkIndex;
4          address seller;
5          uint minValue;          // in ether
6          address onlySellTo;     // specify to sell only to a specific
               person
7      }
8  function punkNoLongerForSale(uint punkIndex) {
9      if (punkIndexToAddress[punkIndex] != msg.sender) throw;
10     punksOfferedForSale[punkIndex] = Offer(false, punkIndex, msg.sender
           , 0, 0x0);
11     PunkNoLongerForSale(punkIndex);
12 }
13 function buyPunk(uint punkIndex) payable {
15     punkNoLongerForSale(punkIndex);
16     pendingWithdrawals[offer.seller] += msg.value;
17     PunkBought(punkIndex, msg.value, offer.seller, msg.sender);
18 }
```

Figure 4: CryptoPunks Code

Dans la fonction *buyPunk()*, le *punkNoLongerForSale()* est appelé **avant** d'envoyer l'ETH, soit le paiement, au vendeur. L'ordre des fonctions est primordial puisque c'est dans *punkNoLongerForSale()*, que le statut de l'acheteur est changé à celui de propriétaire et donc de (nouveau) vendeur de ce NFT; et c'est dans *buyPunk()* que le paiement est envoyé au vendeur. Ainsi, le paiement effectué par l'acheteur du NFT, qui est désormais reconnu comme le nouveau vendeur de ce même NFT, est retourné à l'acheteur au lieu du vendeur originel du NFT.

Reentrancy Attack

Une attaque *reantrancy* exploite un défaut d'ordre logique dans le code d'un contrat intelligent. Une attaque par réentrance implique deux contrats intelligents. Un contrat vulnérable et un contrat d'acteur malicieux. Le contrat de l'attaquant appelle à plusieurs reprises la fonction d'origine du contrat vulnérable avant la fin de son exécution afin de drainer les fonds (Crypto Market Pool., 2021, 27 août).

Par analogie, disons que le contrat vulnérable est une banque et le contrat malicieux est un voleur. Pour faire un retrait à la banque, le voleur a besoin de faire un dépôt de au moins 1 ETH pour pouvoir faire un retrait. En effet, un solde positif sur un compte est nécessaire pour pouvoir retirer. Une fois le dépôt fait, le voleur devient éligible et demande à la banque de faire un retrait de 1ETH de suite après son dépôt (la fonction *withdraw()* est appelée). Dès lors que le voleur a reçu 1 ETH, il relance la fonction de retrait en continue, ne laissant pas l'occasion à la banque de mettre à jour le solde du compte du voleur. En effet, pendant que la fonction de la banque est appelée de manière récursive, la banque n'exécutera pas sa dernière ligne de code qui aurait mis à jour le montant du compte du voleur à 0. De cette façon, le voleur peut continuer à retirer jusqu'à vider les fonds de la banque car aux yeux de celle-ci, le voleur a toujours un solde supérieur à 0.

Un exemple simpliste de contrat vulnérable :

```
function():
    checkbalance
    sendfunds
    updatebalance
```

Le contrat devrait être rédigé dans l'ordre suivant. Cela empêcherait un voleur de l'appeler de manière récursive afin de délayer la mise à jour du solde (*updatebalance*).

```
function():
    checkbalance
    updatebalance
    sendfunds
```

La plus célèbre attaque de réentrance est celle de DAO qui a causé une perte de près 60 millions de dollars américains en 2017 (Cryptopedia Staff., 2022, 16 mars). Malheureusement, de nombreuse autre attaque ont suivi :

Tableau 3: Sommaire des reantrancy attacks recensées sur https://defiyield.app/rekt-database.

VICTIME	DATE DE L'ATTAQUE	MONTANT
Uniswap/Lendf.Me	Avril 2020	$25 millions
BurgerSwap	Mai 2021	$7.2 millions
SURGEBNB	Août 2021	$4 millions
CREAM FINANCE	Août 2021	$18.8 millions
Siren protocol	Septembre 2021	$3.5 millions
Grim Finance	decembre 2021	$40 millions
Rari Fuse	Avril 2022	$80 millions
Ownly	Mai 2022	$19 219
OMNI	Juillet 2022	$1 430 000
n00dleswap	Octobre 2022	$31 096

Le cas de OMNI

OMNI est l'un des plus intéressants pour le sujet de cette recherche. De la même façon qu'un acteur malicieux peut faire des retraits d'ETH d'un contrat, il est possible de faire des retraits de NFT dans certains autres contrats/blockchains; c'est le cas d'OMNI.

OMNI[6] est une plate-forme de financement NFT qui prête de la crypto-monnaie en échange de NFT jalonnés. Cette plate-forme permet aux utilisateurs de miser des jetons NFT pour recevoir des jetons fongibles, par ex. $ETH. Afin de mener à bien l'attaque, l'auteur a d'abord déposé des NFT de la collection Doodles en garantie d'un prêt d'WETH (wrapped ETH). Une fois le prêt garanti, l'attaquant a exploité une vulnérabilité afin

[6] https://defiyield.app/rekt-database

d'exécuter une *reantrancy attack* en retirant tous les NFT Doodles déposés en garantie par l'attaquant à l'exception d'un NFT.

L'attaquant a ensuite utilisé le montant WETH emprunté pour acheter plus de NFT avant de liquider la position du prêt. La position créditrice est liquidée parce que la valeur du NFT restant de la garantie initiale est insuffisante pour couvrir la position de la dette ; le Doodle NFT restant de la garantie est donc restitué à l'attaquant.

Ainsi, l'acteur malicieux a réussi à s'en tirer avec l'argent du prêt ($WETH), utilisé pour acheter de nouveaux NFT, ainsi que les Doodles qu'il avait déposés en garantie. Les fruits de l'attaque ont ensuite été retiré via Tornado.cash, un service mixte qui offusque l'origine des fonds[7].

Access Control Flaws et *Hack a Wallet*

L'*Access Control Flaw* est éponyme et en soit très simpliste. Une attaque utilisant ce mode opératoire exploite un défaut dans la configuration des accès aux données et fonctions octroyés aux utilisateurs. Un attaquant peut également se procurer ou subtiliser les accès d'un utilisateur ayant un accès privilégié pour ensuite s'en servir à sa guise. C'est ce que l'on désigne ici par *Hack a Wallet*.

Les créateurs du projet FlippazOne ont créé un contrat intelligent NFT, qui est également un contrat d'enchères. Le contrat comportait une vulnérabilité grave dans la fonction *ownerWithdrawAllTo()*. En effet, celle-ci ne comporte aucune vérification de l'adresse du propriétaire, ce qui permet à quiconque de retirer tous les fonds du contrat en appelant la fonction pour les envoyer vers son adresse personnelle. Près de $7000 ont été dérobé de la sorte[8].

Lympo est un cas extrêmement couteux de *Hack a Wallet*. Le 10 janvier 2022, des pirates ont réussi à accéder au hot wallet opérationnel de Lympo et ont volé un total d'environ 165,2 millions de LMT, soit près de $18,5 millions[9].

Rug & Pull – L'art du scam au grand jour

Le *Rug & Pull* est l'une des escroqueries les plus courantes sur le marché de l'investissement en crypto-monnaie. Le dessein est simple : le développeur du projet amène les investisseurs à acheter en grande quantité par le biais de promotions marketing afin de maximiser la valeur de sa collection. Puis, il s'enfuit directement avec l'argent sous prétexte d'abandonner le projet. Les acheteurs sont incapables de vendre leur NFT fraichement acquis, ce qui fait chuter son prix en peu de temps. Dans ce type d'escroquerie, le profit revient entièrement au créateur du NFT (Das., 2022; Ruan., 2022).

[7] https://defiyield.app/rekt-database
[8] https://defiyield.app/rekt-database
[9] https://defiyield.app/rekt-database

Un tel dessein est rendu possible par le manque d'attention des investisseurs/acheteurs aux détails du contrat de la collection promue. En effet, bien que les contrats soient entièrement ouverts au public sur la blockchain et parfois même publiés sur des répertoires de code (Github), la plupart des investisseurs ne se pencheront jamais sur le contrat intelligent d'une collection qu'ils convoitent; soit par manque de compétences techniques ou par pur manque d'intérêt. Ces manques rendent les investisseurs/acheteurs vulnérables aux escroqueries telles que R&P.

Integer Overflow

En termes simples, l'*overflow* est une situation où uint (*unasigned integer*) atteint sa taille en octets. Ensuite, le prochain élément ajouté renverra la valeur minimale. Disons que nous avons un uint8, qui ne peut stocker que 8 bits. Cela signifie que le plus grand nombre que nous pouvons stocker est le binaire 11111111 (en décimal 2^8 - 1 = 255).

```
1    solde uint8 = 255 ;
2    solde++ ;
```

Si l'on exécute le code ci-dessus, le "solde" sera égal à 0. Ceci est un exemple simple d'*overlfow*. Si l'on ajoute 1 au binaire 11111111, qui est le nombre maximal atteignable avec un unint8, il se réinitialise à 00000000, soit la valeur minimale d'un unint8 (Ethereum Blockchain Developer., 2022, 10 avril).

Dans le cas d'un *underflow*, si l'on soustrait 1 à un uint8 qui a la valeur 0 (valeur minimale), au lieu que le résultat soit de -1, cela changera sa valeur en 255 (valeur maximale).

Cette vulnérabilité permet donc à une personne d'en envoyer plus d'argent que ce qu'elle ne possède et se retrouver avec plus d'argent que ce qu'elle n'avait à l'origine. De fait, l'*overflow/underflow* permet au destinataire des fonds de recevoir plus d'argent que l'expéditeur n'en avait réellement.

Voici un exemple de transaction utilisant un *overflow*. Disons que le nombre maximum de ETH est régi par un nombre à 5 chiffres.

Tableau 4 : Scénario de comparaison de situation d'Overflow

Pas de *Overflow*	*Overflow*
A possède 10 000 ETH dans son compte A envoie à 200 individus 499 ETH chacun, Le total envoyé est de 200*499 = 99 800 ETH	A possède 10 000 ETH dans son compte A envoie à 200 individus 500 ETH chacun, Le total envoyé est de 100 000 ETH
Le total envoyé respecte la limite de 5 chiffres. Il n'y a pas d'*overflow*, la transaction est simplement bloquée.	Ici, le total envoyé dépasse 5 chiffres. Il y a *overflow*. Le total devient alors 0.

Finalement, A est capable d'envoyer une somme supérieure à son solde d'origine. Créant ainsi de la richesse à partir d'une vulnérabilité. Une personne pourrait également se servir de l'*overflow* afin de faire tomber le solde d'un compte à zéro, en lui envoyant une somme qui ferait dépasser le solde du récipiendaire la limite des 5 chiffres. L'exploitation de cette vulnérabilité permet donc d'envoyer plus d'argent que ce que A ne possède.

Les versions de Solidity antérieures à 0.8.x étaient vulnérables aux attaques d'*overflow/underflow*. Dans Solidity 0.8, le compilateur est sensé se charger automatiquement de vérifier les tentatives *d'overflow et* d'*underflow*. Mais que se passe-t-il si vous voulez tout de même arriver à un *overflow/underflow* ? Il existe un nouveau bloc "*unchecked*" dans lequel vous pouvez envelopper vos variables. Grace à celui-ci, le contrat aura le même comportement que dans les versions précédentes de Solidity. Cette découverte date d'avril 2022 et n'a, pour le moment et à la connaissance de l'auteure, pas reçu de correction (Ethereum Blockchain Developer., 2022, 10 avril).

Enfin, la vulnérabilité a également été identifiée en 2018 dans les contrats intelligents d'Ethereum (basés sur le protocole ERC20) mais a été corrigé depuis (Mix., 2018, 25 avril).

Hypothèses

Plusieurs hypothèses apparaissent naturellement. Tout d'abord, serait-il possible d'exploiter de l'*over/underflow* avec des NFT ? C'est à dire d'envoyer un ou tout une collection de NFT par le biais d'une vulnérabilité d'*overflow* alors que l'individu n'en possède qu'un ou pas du tout ?

Ensuite, est-il possible de faire du typosquatting de *smart contract* ? Les contrats intelligents ayant des identifiants, un acteur mal intentionné pourrait vouloir imiter l'identifiant ainsi que le contenu du contrat en apparence, tout en y rajoutant quelques lignes de code malicieuses. Des utilisateurs non avertis ou naïfs se serviraient ainsi de ce contrat pensant être en sécurité et se feraient donc piéger.

Enfin, à l'instar de n'importe quelle plateforme ou logiciel informatique, il est possible qu'une mise à jour (au niveau du protocole d'une blockchain) amène les anciens contrats à se comporter différemment que ce à quoi ils sont destinés.

2.3 - THE ORACLE ISSUE

Comme expliqué plus haut, les oracles permettent de fournir aux contrats intelligents les données *off-chain* dont ils ont besoin pour leur exécution (Ethereum., 2022, 8 novembre). Cependant, qu'arrive-t-il lorsqu'un oracle fournit des données corrompues, inexactes ou encore inexistantes ? Vous l'aurez deviné, l'utilisation même d'oracle représente une vulnérabilité de taille au sein de la blockchain.

Les oracles servent d'interface entre la blockchain et le monde réel. Cependant, en tant qu'entités centralisées, elles réintroduisent le concept de confiance et de point de défaillance unique dans un environnement décentralisé, mettant en péril cette même

décentralisation (Caldarelli et Ellul., 2021). Ce paradoxe est connu sous le nom de "*Oracle Issue*"

Il est essentiel de s'assurer que les données d'un oracle sont correctes, sinon l'exécution d'un contrat intelligent produira des résultats erronés (Ethereum., 2022, 8 novembre). Mais cela amène deux préoccupations de taille :

- La validité et la falsifiabilité des données - Comment peut-on être sûr que les données sont valides ?
- La disponibilité et l'actualisation des données - Comment s'assurer que ces données soient toujours disponibles et mises à jour régulièrement ?

La validité et la falsifiabilité des données sont deux des préoccupations majeures dans l'*Oracle Issue*. Un hacker pourrait entrer dans un système Oracle et altérer/corrompre les données. Encore plus simplement, un individu ayant accès à l'oracle pourrait manipuler les données à son avantage. Il est donc essentiel que les oracles soient cybersécurisés.

La disponibilité peut être compromise de nombreuses façons. Celle qui figure en tête de liste est l'arrêt volontaire ou involontaire du service Oracle. En effet, si le fournisseur décide de désactiver le service ou si un hacker pirate le composant hors chaîne de l'oracle, un contrat intelligent risque d'être victime d'une attaque par déni de service (DoS).

L'actualisation des données tient du ressort de l'oracle lui-même. Si l'oracle n'est pas capable de fournir des données d'actualité, de nombreux *smart contracts* s'en retrouveront impactés. Par exemple, à l'instar des valeurs boursières des FIATs, un oracle fournissant le taux de change entre ETH et USD se doit d'actualiser ses informations très fréquemment au risque de couter plusieurs milliers, voire millions à de nombreux individus (Gupta et Kumar., 2022).

Enfin, il est tout à fait possible qu'un Oracle présente des erreurs de code ou des bugs entrainant un mauvais fonctionnement du service (Caldarelli et Ellul., 2021).

Le cas de Compound

Compound est un protocole de prêt. En novembre 2020, près de 89 millions de dollars de prêts ont été liquidés sur la plateforme en raison d'un problème d'Oracle[10].

Sur Compound, pour emprunter de la crypto-monnaie, les utilisateurs doivent fournir une garantie qui dépasse le montant qu'ils empruntent. Cela signifie que la garantie est verrouillée dans un contrat intelligent jusqu'à ce qu'une condition de fin soit remplie. Dans certains cas, il peut arriver que la garantie devienne insuffisante et que le prêt soit liquidé en conséquence.

La blockchain ne connaît pas les prix actuels de chaque (crypto)monnaie sur chaque échange, il doit donc l'obtenir quelque part. C'est Ici, que les oracles entrent en jeu. Dans

[10] https://defiyield.app/rekt-database

ce cas, le prix de DAI/USDC est monté à plus de 1,3$ sur Coinbase PRO (l'oracle de tarification de Compound) ; amenant ainsi la blockchain à penser que le prix du *stablecoin* $DAI avait grimpé en flèche.

Figure 5: Screenshot du pic de prix DAI-USDC sur Coinbase PRO

À la réception de telles données, la blockchain a évalué le prix du $DAI comme étant trop haut, rendant ainsi la garantie de nombreux prêts insuffisante. Les prêts en question ont été liquidés, entrainant une perte de $89 millions (Mlinaric, 2021, 14 janvier).

Hypothèse

L'exploitation d'oracle serait possible dans le cadre de NFT. Si un NFT ou sa collection faisaient appel à un service d'oracle pour leur tarification, alors un acteur malicieux pourrait cibler cet oracle dans le but de nuire à la collection de NFT ou d'en faire fluctuer le prix afin de s'en procurer à bas prix.

3 - L'ÉCONOMIE DES NFT

3.1 - LES CONTRATS DE NFT

Selon Ruan (2022), il existe deux types de NFT : le *stand along* NFT et le *share contract* NFT respectivement. Ces types découlent directement de l'existence ainsi que des conditions de leur contrat intelligent.

Stand along NFT

Le premier type de NFT possède un contrat intelligent, publié sur la blockchain, qui lui est propre. Ce contrat enregistre toutes les informations de la collection NFT, allant de l'Identifiant du/des NFT(S), leur statut (*minted* ou non), qui détient le titre de propriété

…etc. Quelques exemples de collections NFT appartenant à cette catégorie sont les CryptoKittie, CryptoPunks ou encore Bored Ape Yacht Club (BAYC).

Share contract NFT

Le deuxième type de NFT correspond à ceux qui n'ont pas leur propre contrat. Les collections et NFT de ce type vont donc partager un contrat avec d'autres collections. Toutes les informations de ces collections sont enregistrées au sein de ce contrat partagé. Ce sont généralement les plateformes de trading de NFT qui créent et maintiennent ces contrats.

En effet, les plateformes telles qu'OpenSea par exemple, fournissent un service de minage de NFT permettant à chaque utilisateur de créer leurs propres NFT. Par exemple, un individu va uploader/téléverser une image sur Opensea et *mint* un NFT, représentant le titre de propriété de l'image, pour ensuite le déposer sur son adresse Ethereum. Le contrat partagé d'OpenSea peut être comparé à une forgerie ouverte au public. Chacun peut y rentrer, créer/forger son NFT puis repartir avec ; c'est pour cette raison que le contrat est nommé « *Shared Storefront* ». Ce contrat étant situé sur la blockchain Ethereum générera un NFT avec un nouvel identifiant et signera le titre de propriété au créateur.

3.2 - LE TRADING DE NFT

Les méthodes de trading de NFT sont définies au sein de leur contrat.

Le trading intégré

Certains contrats NFT possèdent des fonctions de trading intégrées à leur contrat (Ruan., 2022). Ces fonctions permettent à un utilisateur d'échanger son NFT contre de l'Ether (ETH) ou une autre cryptomonnaie en faisant appel à la fonction prévue à cet effet. Ainsi, il suffit de prêter attention au code du contrat d'un NFT pour savoir quels sont les moyens mis disponible pour échanger un NFT. Ces contrats permettent un trading qui est dit intégré. En général, les fonctions permettent de mettre en vente, acheter, mettre aux enchères ou encore proposer un prix pour un NFT.

Cryptopunks est une des collections dont le trading est intégré à son contrat. Les fonctions définies au sein de son contrat sont éponymes : offerPunkForSale(); buyPunk(); enterBidForPunk() et acceptBidForPunk().

```
140     function offerPunkForSale(uint punkIndex, uint minSalePriceInWei) {
156     function buyPunk(uint punkIndex) payable {
195     function enterBidForPunk(uint punkIndex) payable {
211     function acceptBidForPunk(uint punkIndex, uint minPrice) {
```

Figure 6: extrait du contrat intelligent des CryptoPunks (https://github.com/larvalabs/cryptopunks/blob/master/contracts/CryptoPunksMarket.sol)

Le trading extérieur ou tierce

En revanche, certains contrats ne contiennent qu'une fonction de transfert, rendant indisponible l'échange d'un NFT contre l'ETH ou autre cryptomonnaie. Les collections se retrouvant dans ce cas NFT se doivent de faire appel à une tierce partie, soit une plateforme de trading telles qu'OpenSea ou encore NFTrade …etc. Le rôle de ces plateformes est simple, mettre en contact un vendeur et un acheteur, et faire transiter la monnaie d'échange d'une adresse à l'autre. Ce n'est qu'une fois la transaction faite que le vendeur du NFT appellera la fonction de transfert présente dans le contrat du NFT afin de transférer le titre de propriété à l'acheteur.

3.3 - LES PLATEFORMES DE TRADING

Il existe une multitude de plateforme dédiées au trading de NFT mais OpenSea est de loin l'une des plateformes les plus connues dans le domaine. Avec plus d'un million d'utilisateurs enregistrés et une valeur de plus de 13 milliards USD, elle peut être considérée comme l'équivalent d'eBay pour la cryptomonnaie et les NFT. À l'instar d'OpenSea, les plateformes de trading sont principalement utilisées pour acheter et vendre des NFT mais elles permettent également d'en créer (Campbell, S., 2022, 21 novembre).

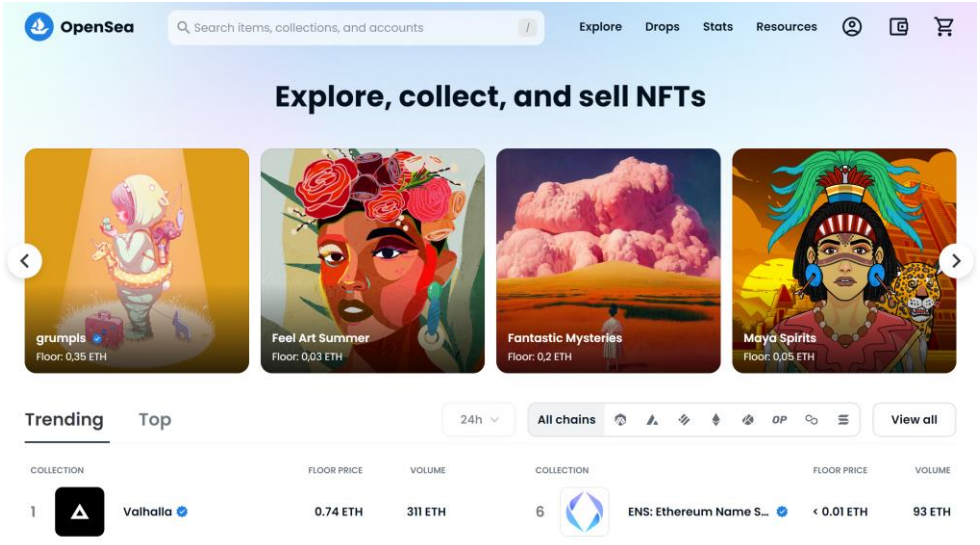


Figure 7:Screenshot d'OpenSea (29 novembre 2022)

4 - VULNÉRABILITÉS ET ATTAQUES CONNUES

Airdrop Phishing Attacks

Comme de nombreuses plateformes de NFT, OpenSea permet à quiconque de créer un NFT et de le vendre sur la plateforme. Le NFT peut être créé à partir de n'importe quel fichier tant qu'il se termine par une des extensions suivantes : JPG, PNG, GIF, SVG, MP4, WEBM, MP3, WAV, OGG, GLB, GLTF. C'est l'une de ces extensions, le SVG, qui est à l'origine de la vulnérabilité présentée dans cette section.

L'attaque de phishing par Airdrop a été introduite pour la première fois par le groupe Check Point Research en octobre 2021 (Barda, D., Zaikin, R. & Vanunu, O., 2021, 13 octobre). La société israélienne de cybersécurité a signalé qu'elle avait trouvé des vulnérabilités dans OpenSea qui auraient pu permettre aux cybercriminels de vendre des NFT malveillants ou de l'art numérique comportant un cheval de Troie (*Trojan*). La vulnérabilité permettait aux acteurs malicieux de créer un NFT à partir d'une image (SVG) comportant un logiciel malveillant et de le Airdrop à une victime.

La victime reçoit alors une notification indiquant qu'un NFT lui a été envoyé à son adresse et va ainsi cliquer sur le NFT et l'ouvrir. Lorsque la victime ouvrira le fichier NFT, le code malveillant sera exécuté en arrière-plan et une série de pop-ups malveillants imitant la plateforme OpenSea se déploiera. Parmi les nombreux pop-ups se trouvera une demande à l'utilisateur de connecter son portefeuille numérique.

Il convient de noter que les pop-ups de signature de portefeuille apparaissent souvent sous la forme d'un avis système et constituent un processus standard adopté par de nombreuses plateformes afin de créer plusieurs activités. Dans ce cas, si les utilisateurs ne lisent pas attentivement les pop-ups, ils pourraient avoir autorisé sans le savoir l'accès à leur compte en pensant que la communication se faisait au nom d'Opensea. En cliquant sur le pop-up, l'utilisateur envoie au pirate, sans le savoir, les informations concernant son *wallet* et lui signe également une sorte de procuration, l'autorisant à réaliser des transactions en son nom. Ainsi, l'attaquant peut simplement transférer tous les actifs numériques du compte de la victime vers son compte. OpenSea a corrigé la faille lorsqu'elle a été portée à l'attention de l'entreprise.

Un scenario similaire s'est produit avec les plateformes MetaMask et Rarible en février et avril 2022. Le dessein était le même, offrir un NFT contenant du code malicieux aux utilisateurs afin qu'ils cliquent dessus pour activer l'attaque. Ces fois-ci, en signant la transaction, les utilisateurs vont, sans le savoir, faire une procuration à l'attaquant, l'autorisant ainsi à gérer leurs NFT (Barda, D., Zaikin, R. & Vanunu, O., 2022, 14 avril).

Ce dessein légèrement différent prend avantage de la fonction *setApprouvalForAll().* Comme expliqué dans la section SMART CONTRACT, NFT a une norme (ERC-721), qui fournit des fonctionnalités de base pour les contrats NFT. Cette norme contient une fonction appelée *setApprovalForAll*(). Cette fonction désigne essentiellement qui est autorisé à contrôler tous vos NFT. La fonction a été créée principalement pour des tiers

comme Rarible ou OpenSea pour faciliter les transferts de NFT/tokens par ces plateformes au nom des utilisateurs. Cette fonction peut être très dangereuse puisqu'elle peut permettre à n'importe qui de contrôler vos NFT si vous l'approuvez auparavant.

Les utilisateurs ne sachant pas toujours exactement quelles autorisations ils accordent en signant une transaction, ils supposent, la plupart du temps, qu'il s'agit de transactions régulières alors qu'en fait, ils fournissent l'accès à leurs propres NFT à un individu malicieux. En cliquant sur le bouton de confirmation, l'attaquant obtient un accès complet à tous les NFT en possession de la victime. L'attaquant peut désormais transférer tous les NFT sur son compte car la victime l'a "autorisé" à le faire.

Old Listing Flaw

En janvier 2022, un problème de taille a attiré énormément d'attention au sein de la communauté d'OpenSea. À cause d'un principe de base de la plateforme, soit de ne pas prendre part dans la gestion des mises en vente des NFT des utilisateurs, des individus pouvaient acheter un ou plusieurs NFT à un prix bien inférieur à celui qui est d'actualité grâce à une ancienne mise en vente (*listing*) de ces derniers (Attalah., 2022, 26 janvier). Par exemple, un individu a réussi à acheter un NFT de la collection Bored Ape Yatch Club (BAYC) pour seulement 23ETH via une ancienne mise en vente datant de juillet 2021 (Irwin., 2022, 24 janvier). Il l'a ensuite revendu, pour 135ETH, soit 5 fois le plus prix d'achat.

L'inscription en vente d'un NFT est enregistrée sur la blockchain après que le vendeur ait confirmé la mise en vente via son portefeuille. Une fois l'inscription confirmée, elle ne peut être annulée que par le vendeur qui signe la transaction d'inscription.

Sur le marché financier traditionnel, la mise en vente ou le *listing* d'une action en bourse ne peut excéder une journée (*a trading day*) et les mises en vente non conclusives seront automatiquement annulées à la fin de la journée lorsque la bourse est fermée. Cependant, sur le marché NFT, les utilisateurs peuvent définir une mise en vente sans délai d'expiration (définir le délai d'expiration sur infini). Ils doivent ainsi annuler la mise en vente manuellement ; tache qu'ils pourraient oublier de faire.

Au lieu de décrire cette faille comme un bug ou un *exploit*, OpenSea défend sa position en expliquant qu'il s'agit d'une caractéristique fondamentale des plateformes de trading : seule la personne qui met en vente un article peut annuler sa mise en vente. OpenSea ne peut interférer dans ce processus. Bien que ce principe soit essentiel, il apparait clair que la conception du système de *listing* est problématique.

Plateform Impersonating Email Phishing Attacks

Pour résoudre le *Old Listing Flaw*, Opensea a annoncé la migration des contrats le 18 février 2022, obligeant leurs utilisateurs à migrer leurs anciennes mises en vente vers le nouveau contrat intelligent d'Opensea sur Ethereum. Suite aux annonces d'Opensea, des hackers ont profité de la situation pour arnaquer les utilisateurs d'Opensea en envoyant

un email copiant le format de l'email officiel d'OpenSea. Le courriel de phishing contenait un lien malveillant et a été envoyé à divers utilisateurs d'Opensea le lendemain de l'annonce (Barda, Zaikin & Vanunu., 2022, 24 février).

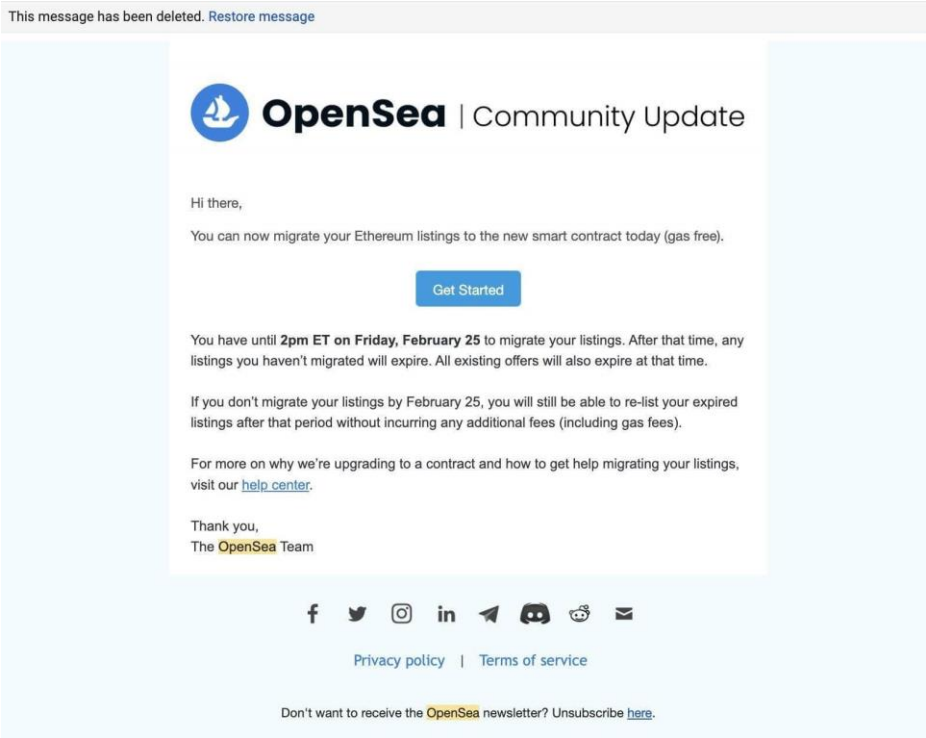


Figure 8: Courriel de phishing utilisé pour l'attaque (https://blog.checkpoint.com/2022/02/20/new-opensea-attack-led-to-theft-of-millions-of-dollars-in-NFT/)

En cliquant sur « *Get Started* », l'utilisateur est redirigé vers un site de phishing lui demandant de signer une transaction qui semblent être destinée à la migration de contrat et qui imite le format d'OpenSea.

Une fois la transaction signée, une requête de transfert est envoyée au contrat du pirate et non celui d'OpenSea. C'est seulement une fois que la requête a transigé par le contrat du pirate qu'elle est transférée au contrat d'OpenSea. À la réception de la requête, le contrat d'OpenSea vérifie les paramètres et exécute la transaction puisqu'elle a été signée et approuvée par la victime au début. Les NFT de la victime sont ainsi transférés à l'adresse du pirate.

En fait, en signant la transaction, la victime définit le pirate comme récipiendaire de la transaction au lieu d'OpenSea. De la sorte, le pirate réussit à subtiliser les fruits de la transaction sans que la victime ne soit au courant.

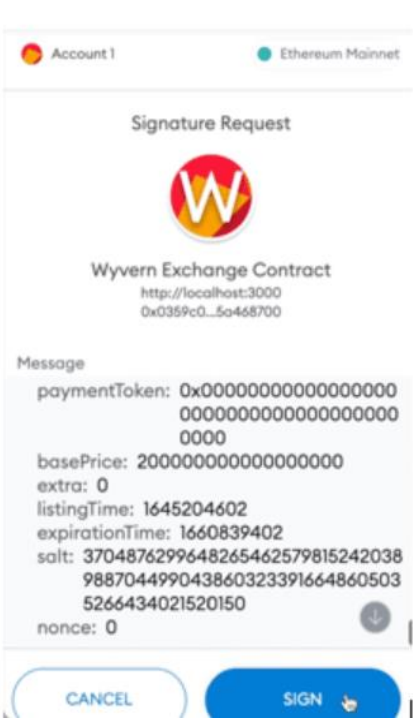


Figure 9: Transaction Originelle OpenSea

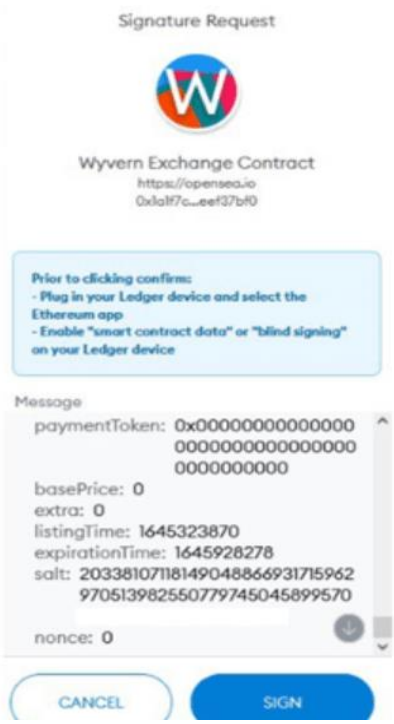


Figure 10 : Transaction Malicieuse

Source : https://twitter.com/isotile/status/1495234649970421760?s=21

CONCLUSION

Compte tenu du rythme effréné de l'innovation, il existe un défi inhérent à l'intégration sécurisée des applications DeFi et des crypto-marchés. Les acteurs malintentionnés savent qu'ils disposent actuellement d'une fenêtre d'attaque fort profitable. Effectivement, l'engouement autour des NFT en tant que nouvel objet d'investissement tendance allié à la latence des mesures de (cyber)sécurité rendent l'environnement NFT propice aux cyberattaques. La communauté de la cybersécurité doit intensifier ses efforts pour aider les technologies pionnières de la blockchain à sécuriser les actifs cryptographiques des consommateurs.

Ce rapport a tenté d'établir un état des lieux des principales failles et vulnérabilités présentes au sein de l'écosystème de la blockchain et des NFT.

En premier lieu, il semble que les contrats intelligents représentent un point de défaillance et d'intérêt malveillant important au sein de la blockchain. Ce sont les erreurs et manipulation du code dans la programmation initiale d'un contrat qui rendent possible la quasi-totalité des vulnérabilités recensées. Étant entièrement composés de code, peu sont ceux qui prennent le temps d'en analyser le contenu avant d'en appeler les services. Les contrats étant disponibles sur la blockchain, prêter une plus grande attention aux détails des contrats avant de les utiliser réduirait significativement les chances d'attaque.

Ensuite, les risques associés aux services d'oracle (*oracle issue*), soit les enjeux de validité, falsifiabilité, disponibilité et actualisation des données constituent la deuxième préoccupation de taille quant à l'environnement de la blockchain. Ces enjeux peuvent être facilement réduit en augmentant la vigilance lors de la sélection du service. Un service d'oracle décentralisé est à privilégier puisqu'ils permettent d'éviter et/ou de mitiger la plupart des attaques visant les oracles centralisés comme le DDOS ou encore la corruption de l'entité responsable du service d'oracle.

Bien que l'attaque des 51% ait été mentionnée au sein de ce rapport, elle ne semble pas poser de réelle menace pour les blockchains supportant les NFT. L'utilisation du PoS par ces dernières leur permet de réduire presqu'à néant les chances de réussite d'une telle attaque grâce au système de *staking*. La risque d'une perte de plusieurs millions d'ETH pour les attaquants constitue une dissuasion de taille.

En deuxième lieu, ce sont les plateformes de trading qui apparaissent comme cibles privilégiées des attaques récentes. Les plateformes de trading sont des facilitateurs d'attaque puisqu'elles servent d'interface entre différents utilisateurs pour des transactions. Ces plateformes représentent également une entité de « confiance » pour bon nombre d'utilisateurs, ce qui rend l'usurpation d'identité de ces plateformes très attrayante dans les desseins malicieux.

Une grande partie des attaques recensées usurpent l'identité des plateformes. Les attaques de phishing jouent sur la confiance qu'octroient les utilisateurs aux plateformes en tant qu'entité de confiance pour amener les utilisateurs à fournir les accès à leur wallet, signer des transactions malicieuses, ou encore octroyer des procurations aux attaquants. Ces attaques étant des attaques de masse, il est important que les plateformes sensibilisent les utilisateurs à ce genre d'attaque. Une vigilance accrue des utilisateurs lors de la signature de transaction ou la réception de mails est préconisée contre ce type d'attaque.

Enfin, les paramètres et régulations des plateformes sont problématiques. En effet, la permission de nombreuses extensions dans la création de NFT ou encore la mauvaise conception du système de mise en vente de NFT sont deux des fonctionnalités qui, une fois exploitées, ont donné naissance à des vecteurs d'attaques inédits. De tels paramètres méritent une plus grande réflexion lors de leur implémentation au sein des plateformes.

RÉFÉRENCES

Attalah, A. (2022, 26 janvier). *Important updates for listing and delisting your NFTs*. OpenSea. https://opensea.io/blog/safety-security/important-updates-for-listing-and-delisting-your-nfts/

Barda, D., Zaikin, R. & Vanunu, O. (2021, 13 octobre). *Check Point Research Prevents Theft of Crypto Wallets on OpenSea, the World's Largest NFT Marketplace*. CheckPoint Research. https://research.checkpoint.com/2021/check-point-research-prevents-theft-of-crypto-wallets-on-opensea-the-worlds-largest-nft-marketplace/

Barda, D., Zaikin, R. & Vanunu, O. (2022, 14 avril). *Check Point Research detects Vulnerability in the Rarible NFT Marketplace, Preventing Risk of Account Takeover and Cryptocurrency Theft.* CheckPoint Research. https://research.checkpoint.com/2022/check-point-research-detects-vulnerability-in-the-rarible-nft-marketplace-preventing-risk-of-account-take-over-and-cryptocurrency-theft/

Barda, D., Zaikin, R. & Vanunu, O. (2022, 24 février). *New OpenSea attack led to theft of millions of dollars in NFTs*. CheckPointResearch. https://blog.checkpoint.com/2022/02/20/new-opensea-attack-led-to-theft-of-millions-of-dollars-in-nfts/

Caldarelli, G. (2020). Understanding the blockchain oracle problem: A call for action. Information, 11(11), 509.

Caldarelli, G., & Ellul, J. (2021). The blockchain oracle problem in decentralized finance—A multivocal approach. Applied Sciences, 11(16), 7572.

Campbell, S. (2022, 21 novembre). *OpenSea Statistics 2022: How Many Users Does OpenSea Have?* TheSmallBusinessBlog. https://thesmallbusinessblog.net/opensea-statistics/#:~:text=OpenSea%20has%20more%20than%201,number%2015%20on%20the%20list

Corbella, A. (2021, 28 septembre). *Are NFTs safe? Uncovering NFT Vulnerabilities and Security Concerns.* Valid.Network. https://valid.network/post/are-nfts-safe-uncovering-nft-vulnerabilities-and-security-concerns

Crypto Market Pool. (2021, 27 août). *Reentrancy attack in a Solidity smart contract.* https://cryptomarketpool.com/reentrancy-attack-in-a-solidity-smart-contract/

Cryptopedia Staff. (2022, 16 mars). *What Was The DAO?.* Gemini. https://www.gemini.com/cryptopedia/the-dao-hack-makerdao

Das, D., Bose, P., Ruaro, N., Kruegel, C., & Vigna, G. (2021). Understanding security Issues in the NFT Ecosystem. arXiv preprint arXiv:2111.08893.

ELLIPTIC., (2022). *Preventing Financial Crime in Cryptoassets.* https://www.elliptic.co/hubfs/Typologies-2022-Preventing%20Financial%20Crime%20in%20Crypto-NH.pdf?hsCtaTracking=459bfdd0-05d4-4c16-ade1-d73d88236fe8%7Ce1df3ed3-2a59-40bc-990d-a11d6a803e24#:~:text=The%20survey%20was%20conducted%20online,financial%20crime%20risk%20from%20cryptoassets.

Ethereum., (2022, 8 novembre). *Oracles.* https://ethereum.org/en/developers/docs/oracles/#top

Ethereum. (2022, 3 novembre). *PROOF-OF-STAKE (POS).* https://ethereum.org/en/developers/docs/consensus-mechanisms/pos/#top

Ethereum. (2022, 1 septembre). *INTRODUCTION TO SMART CONTRACTS.* https://ethereum.org/en/developers/docs/smart-contracts/

Ethereum. (2022, 26 septembre). *PROOF-OF-WORK (POW).* https://ethereum.org/en/developers/docs/consensus-mechanisms/pow/#top

Ethereum. (2022, 15 août). *ERC-1155 MULTI-TOKEN STANDARD.* https://ethereum.org/en/developers/docs/standards/tokens/erc-1155/#safe_transfer_rule

Ethereum. (2022, 15 août). *NORME DE JETON NON FONGIBLE ERC-721. https://ethereum.org/fr/developers/docs/standards/tokens/erc-721/*

Ethereum Blockchain Developer. (2022, 10 avril). *Integer Overflow and Underflow.* https://ethereum-blockchain-developer.com/010-solidity-basics/03-integer-overflow-underflow/

Futura Sciences. (2022, 12 juin). *Non-fungible token : qu'est-ce que c'est ?* https://www.futura-sciences.com/tech/definitions/tech-non-fungible-token-19205/

Gupta, Y., & Kumar, J. (2022). Identifying Security Risks in NFT Platforms. arXiv preprint arXiv:2204.01487.

Irwin, K. (2022, 24 janvier). *OpenSea Exploit Sees Bored Ape Yacht Club NFT Sell For $1,700 in Ethereum.* Decrypt. https://decrypt.co/91076/opensea-exploit-sees-bored-ape-yacht-club-nft-sell-1700

Jmcook.eth. (2022, 15 avril). *Ethereum PoS Attack and Defense.* Mirror. https://mirror.xyz/jmcook.eth/YqHargbVWVNRQqQpVpzrqEQ8IqwNUJDIpwRP7SS5FXs

Mix. (2018, 25 avril). *Ethereum bug causes integer overflow in numerous ERC20 smart contracts [Update]*. TheNextWeb. https://thenextweb.com/news/ethereum-smart-contract-integer-overflow

Mlinaric, N. (2021, 14 janvier). *DeFi Hacks - millions lost in 2020*. Node Factory. https://nodefactory.io/blog/defi-hacks-millions-lost-in-2020/

Nakamoto, S. (2008). Bitcoin: A peer-to-peer electronic cash system. Decentralized Business Review, 21260.

Werner, S. M., Perez, D., Gudgeon, L., Klages-Mundt, A., Harz, D., & Knottenbelt, W. J. (2021). Sok: Decentralized finance (defi). arXiv preprint arXiv:2101.08778.