Unveiling the Paradox of NFT Prosperity

Anonymous Author(s)

ABSTRACT

Unlike fungible tokens (e.g., cryptocurrency), a Non-Fungible Token (NFT) is unique and indivisible. As such, they can be used to authenticate ownership of digital assets (e.g., a photo) in a decentralized fashion. Given that NFTs have generated significant media attention since 2021, we perform a large-scale measurement study of the NFT ecosystem. We collect over 242M transfer logs and over 97M marketplace transactions until Aug 1st, 2023, by far the largest NFT dataset, to the best of our knowledge. We characterize the on-chain behavior of NFTs and their trading across five major marketplaces. We find that, although the NFT ecosystem is growing rapidly, it is driven by a relatively small set of dominant centralized players, with suspicious trades activities, e.g., over 23% of the monetary volume is generated by malicious wash trading and the ecosystem has experienced over 157K cases of NFT arbitrage, with a total sum of over $25M USD profit. Our observations motivate the need for more research efforts in the NFT security analysis.

1 INTRODUCTION

There has been significant media and market attention surrounding Non-Fungible Tokens (NFTs) [6, 17]. These are a kind of cryptographic token that is *unique* and *indivisible*. Each NFT is *one-of-a-kind* and can be used to authenticate ownership of a single digital entity, e.g., a photo. As all exchanges of NFTs are recorded on a blockchain, they can be used to prove the ownership of a particular asset. This simple concept has spurred interest, assisting users to trade non-fungible goods in a decentralized fashion. Yet, many are concerned about the economic risks of NFTs, as their rapid growth [13] has attracted various anecdotal fraudulent attacks.

Although there has been recent work [55] on NFTs *themselves*, we lack answers to (even basic) questions that are associated with *NFT markets*, such as (*i*) How can we systematically collect data from NFT markets? (*ii*) How often are NFTs traded and for what price? (*iii*) Which are the most dominant marketplaces and what role do they play in underpinning the wider ecosystem? (*iv*) Are NFTs subject to price fraud, or other associated types of market manipulation?

To explore these issues, we conduct a comprehensive study of the NFT market ecosystem. Our focus is on the digital tokens themselves (NFTs) and the platforms where people buy and sell them. *First*, we aim to examine the growth of the NFT ecosystem, which includes tracking NFT-related events, the number of participants involved, and how these marketplaces operate — particularly if there are any unfair practices. *Second*, we aim to explore the possibility for market manipulation within the ecosystem. Based on anecdotal reports [22, 32], we strive to systematically understand the severity of this problem.

To achieve these aims, we collect over 242M transfer logs and 97M marketplace trades until Aug 1st, 2023 (§3). After that, we conduct a graph analysis of NFTs, as well as how they are exchanged via NFT marketplaces (§4). We identify preliminary evidence of potential market manipulation, and this inspires us to perform a rigorous analysis of two specific cases (§6): (*i*) *wash trading*, where users repeatedly exchange NFTs between accounts they control to simulate artificial demand; and (*ii*) *arbitrage*, where users strategically sell and buy across marketplaces to exploit fluctuations in price. We find that both are commonplace, with worrying implications: over 23% of the NFT market's monetary volume is fake (generated artificially by wash trading). This raises serious concerns over the sustainability of the NFT market.

We make the following research contributions in this paper:

- *We perform a large-scale graph analysis of the NFT ecosystem.* We gather a dataset covering over 24M NFT smart contracts, 142M NFTs, 242M transfer events and 97M trade events. We expose a growing ecosystem, driven by a relatively small set of dominant players with unhealthy behaviors.
- *We measure the prevalence of wash trading behavior in the NFT ecosystem.* We reveal that NFTs experience significant price manipulation by at least 826 wash trading bots. In total, these bots account for at least over $24*B* USD of history volume growth (over 23%) in the NFT ecosystem.
- *We propose a methodology to detect the arbitrage of NFTs.* Our proposed detection method reveals that over 157K instances of NFT arbitrage exist in the wild, with the profits of over $25*M* USD conducted by 629 accounts. All datasets will be made publicly available.

We will release our results to the research community.

2 BACKGROUND

2.1 Ethereum Primer

Ethereum. *Ethereum* is one of the most popular blockchains. Its key innovation was the introduction of *smart contracts*, and it is the de-facto technology used for NTFs. *Ether (ETH)* is the native cryptocurrency on *Ethereum*, the second largest cryptocurrency after *Bitcoin* [8].

Ethereum Account. *Ethereum accounts* are identified by a fixed-length hash-like address, which can be divided into *external-owned accounts (EOAs)* and *contract-owned accounts (COAs)*. *EOAs* are controlled by users, i.e., anyone with private keys, while the *COAs* are controlled by code stored together with the accounts. An *EOA* is an ordinary account that can transfer tokens, invoke deployed *smart contracts* and store received tokens. Moreover, an *EOA* can deploy a *smart contract* into a *COA* and a *COA* can only send *transactions* in response to receiving *transactions*.

Ethereum Transactions. When a user wants to interact with *Ethereum*, a *transaction* is made through their *EOA* to modify or update the state stored in *Ethereum*.

Etherum Smart Contracts. A *smart contract* consists of code that implements actions using *transactions*. Based on the foundation of *smart contracts*, *ERCs* (Ethereum Request for Comments) have proposed a series of standards for digital tokens in *Ethereum*.

2.2 Digital Token and DeFi

Tokens. Each token belongs to a *token smart contract*, which defines a set of functions used to perform different tasks. One prominent example is *ERC-20*, which is non-unique and divisible [11]. In a token smart contract under the *ERC-20 standard*, all tokens are the same and have the same value.

NFTs. *A Non-Fungible Token (NFT)* is a kind of cryptographic asset implemented on a blockchain. *NFTs* are used to identify content in a digital way. Such content includes paintings, videos or other items in the real world. The ownership of the NFT is recorded via a *transaction* on the blockchain. Thus, theoretically people can verify the ownership.

ERC-721 and ERC-1155. *ERC-721* defines a minimum set of interfaces which a smart contract must implement to manipulate the NFT tokens on *Ethereum*. Each ERC-721 NFT has unique *ID* and identifies one unique piece of content, which means they cannot be divided into smaller units. However, when we need many different kinds of *NFTs* to operate, using *ERC-721* is inefficient since it needs to create many *ERC-721* contracts. To address this, *ERC-1155* was proposed to manage multiple token types in a single *smart contract*. The unique *ID* of a *ERC-1155* smart contract points to a batch of tokens that have the same content. If someone needs to transfer a batch of tokens, they can execute a single *transaction* (rather than multiple ones), which consumes less *gas* (the fee required to conduct a transaction or execute a contract).

Decentralized Exchanges. *Decentralized exchanges (DEXes)* provide peer-to-peer marketplaces for investors who want to trade digital tokens. The *DEXes* have their own smart contracts launched to deal with the events the transactions generate through DEXes.

NFT Secondary Marketplaces. In the NFT ecosystem, the NFT exchanges (aka "*secondary marketplaces*") play the role of *DEXes*. Five top platforms dominate the NFT market: *OpenSea* [16], *LooksRare* [21], *CryptoPunks* [9], *LooksRare* [14], and *Blur* [7]. They each have their own unique official smart contracts that have been launched on *Ethereum*. They also have front-end websites which provide a convenient place for NFT trading.

2.3 The Life Cycle of an NFT

NFT Creation. An NFT smart contract (which normally implements either *ERC-721* or *ERC-1155* tokens) implements all features and functions of one NFT project. After the launch, other participants can perform the "mint" function to create an NFT. Normally, the qualification of minting tokens is sold to the public as a chance to be added to the *whitelist* of the projects' smart contract. The accounts then have the privilege to perform the mint operation and generate a *mint event*, as well as to gain authority over the token. Note, NFT smart contracts on *Ethereum* have an "approve" operation which allows users to grant their privileges on tokens to other accounts. Note that, NFT can also be burned, i.e., destroying it by sending an NFT to an un-spendable address.

NFT Trading. NFTs rely on a *secondary marketplace* for circulation, where token owners can list their NFTs. In a marketplace, the NFTs of a project always appear as a "collection", which is an off-chain concept and can be seen as "brands" in the NFT world. Normally, one smart contract maps to one collection. Optionally, sellers can list their NFTs on multiple marketplaces and users can place *bids* on them. When an *offer* is accepted, the website will automatically invoke their official smart contract to deal with this event, and generate a *swap* event. For full details, we redirect readers to [46].

Table 1: Dataset overview.

Data Type	# Number	Type	# of Transfer Events	marketplace	# of Trade Events
Smart Contract	244,154	Mint	148,500,667	OpenSea	93,128,954
Token (Except ERC-1155)	142,561,997	Burn	917,025	X2Y2	2,264,694
Transfer Event	242,444,962	Swap	93,027,270	CryptoPunks	30,839
Trade Event	97,902,053			LooksRare	620,789
				Blur	1,856,777

3 DATASETS

Token Transfer Dataset. We use *Geth* [26] to download the *Ethereum* ledger. We first synchronize all blocks until Aug. 1st, 2023. We then extract four parts of data from these blocks: *external transactions*, *internal transactions*, *contract information*, and contract calling information. We then trace every NFT contract and extract other information directly from the blockchain. We extract all 242, 444, 962 transfer events.

NFT Secondary Market Trade Dataset. We next compile data covering the trades that take place within marketplaces. Note, a trade is different to a transfer: a *trade* takes place within the smart contract of a secondary marketplace (for a sum of money), whereas a *transfer* is the event that transfers NFT ownership to another account on the first market (i.e. the Ethereum). We start by manually analyzing the smart contracts of five major NFT markets to see how they execute NFT trades: *OpenSea*, *X2Y2*, *CrypotoPunks*, *LooksRare* and *Blur*. These cover over 98.1% of the total trade volume in Ethereum [10]. The specific contract analysis and data collection methods are detailed in Appendix A. We gather 97,902,053 data items in our NFT secondary market trade dataset until Aug. 1st, 2023.

NFT Smart Contracts and NFTs Dataset. To identify all NFT smart contracts and tokens, we simply extract all the *ERC-721* and *ERC-1155* token's transfer events in the external transaction logs. In total, we identify over 244,154 NFT smart contracts. Note, because smart contracts under the *ERC-1155* standard could be called to mint a huge number of tokens at one time, it is meaningless to count the *ERC-1155* tokens. While minting a token, a specific transfer event is generated (on the blockchain) whose transfer from is the *null address*. Thus, we count this type of transfer event and filter out *ERC-1155* transfer events to calculate the number of NFTs. This gives us 142,561,997 NFT tokens in total. To the best of our knowledge, this is the most complete dataset of NFTs available.

Dataset Overview. Table 1 summarizes the data we have collected, consisting of the data type, transfer type and trade marketplace. In total, we have collected over 244,154 NFT smart contracts, 128M NFTs, 242,444,962 transfer events and 97,902,053 marketplace trade events. For analysis, we further divide the transfer events into three types. For those transfer events whose "transfer from address" is the null address, we label them as *mint events*. For those whose "transfer to address" is the burn account [24], we label them as *burn events*. This is where the user removes the tokens from the overall supply (aka "burning"). For the remaining tokens, we label them as *swap events*, whereby an NFT is transferred to another owner.

4 NFT ECOSYSTEM DEVELOPMENT

4.1 Exploration of NFTs Events

We first inspect the activity and usage of NFTs by dissecting the various NFT events recorded.

Mint Events of NFTs. A *mint event* is when a smart contract is used to create a new NFT. Fig. 1(a) presents a time series of the number of *ERC-721* and *ERC-1155* tokens minted. When the *ERC-721* standard was first proposed in 2018, it did not attract much attention. But since the beginning of 2021, the creation of *ERC-721* tokens has become far more frequent, with significant growth. This is primarily driven by the growing use cases of NFTs. The total number of *mint events* of *ERC-721* smart contracts in Jan 2021 hit 96,771, while the number is 4,518,268 in Jan 2022, which has increased over 46 times.

There have also been serious fluctuations during this period. For example, from the middle of Sept 2021, the daily creation rate dropped rapidly, before rebounding again in 2022. Overall, the rate of *ERC-721* tokens creation has been higher than that of *ERC-1155* tokens. Closer inspection further reveals significant peaks. For example, from Oct. 29th, 2019, to Nov. 18th, 2019, the number of mints per day is above 10^5, where it reaches a peak on 2019.11.17 (with over 4.8M mints). We find that the project *Gods Unchained Cards* performs the majority of minting during that period (a digital trading card game). During this period, it minted many cards to satisfy the needs of its players. This phenomena highlights that the behavior of the overall ecosystem can be heavily affected by a single (non malicious) influential smart contract.

We also inspect the distribution of *mint events* across all NFT contracts. Fig. 1(b) and (c) present the number of mint events per contract for *ERC-721* and *ERC-1155* contracts, respectively. 23.1% of *ERC-721* smart contracts only mint one token, and 49.1% of the smart contracts mint no more than 5 tokens (64.4% mint no more than 20 tokens). The characteristics of *ERC-721* contracts are similar with *ERC-1155* contracts, although overall *ERC-1155* contracts tend to mint more tokens. The respective percentages for *ERC-1155* are 35.9%, 61.1%, and 74.8%. Thus, a small number of smart contracts mint the majority of NFTs: The top 10% of contracts mint 90.57% of all tokens. This raises serious concerns about the the true level of decentralization in the ecosystem, as the removal of a small number of stakeholders would remove the majority of "creativity".

Swap Events of NFTs. To explore how active these tokens are, we next look the number of swap events for each token. Recall, a *swap* event is where the ownership of an NFT is transferred to another. Fig. 2(a) presents a time series distribution of the number of token *swap events*. We see that *swap events* became frequent in the beginning of 2021 and have grown by 5581% since (Jan 2021 – Sept 2022). Much like the token mint timeline, the curve fluctuates heavily and the swap rate of *ERC-1155* tokens is less than *ERC-721* tokens for the same reason discussed above.

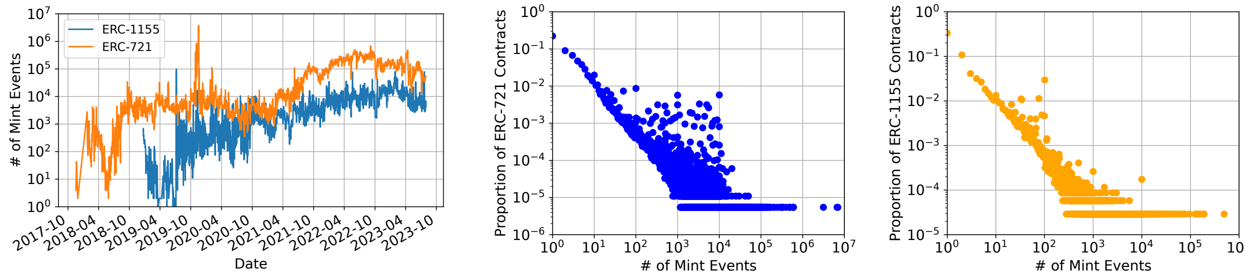
Fig. 2(b) and Fig. 2(c) present the distribution of *swap events* per contract for *ERC-721* and *ERC-1155* contracts, respectively. We observe a large range among the number of *swap events*. Whereas most tokens are transferred a small number of times, we observe an elite that experience extremely heavy circulation. Only the top 1% have been transferred over 20 times. Consequently, we observe a long-tail of highly undesirable NFTs. 73.1% of *ERC-721* NFTs have never been transferred (77.2% for *ERC-1155* tokens); and 98.9% (98.4%) of them have fewer than 5 swap events. This suggests that the majority of NFTs are rather undesirable and experience little market activity.

Burn Events of NFTs. Finally, we inspect the number of *burn events* for NFTs. A *burn event* is where an NFT is deleted from the supply. As shown in Table 1, we identify 917,025 *burn events*. There are only 12,652 (4.96% of the total) smart contracts that have one or more burn events. This is perhaps surprising as it is not clear why one would "burn" an NFT. To understand the reasons, we manually investigate 100 NFT projects that have *burn events*, and observe the following reasons. First, some projects burn for corner-case reasons. To highlight this we take the example of the *OpenSea Shared Storefront* smart contract, which has the huge number of burn events (33,982). It is the official contract from *OpenSea*, an NFT marketplace: It does not only support one collection, but many (in fact, it allows users to mint their own NFTs). Thus, the contract burns NFTs that are removed from the market, e.g., because they are reported to be scams. Second, *ERC-1155* NFT projects appear to burn their NFT tokens to reduce the total supply. For example, we check the *ERC-1155* NFT project *PAGE* [27] that has the second largest number of *burn events* (29,045). Unlike *ERC-721* tokens, the *contract address* and *token ID* belong to a set of tokens with the same price. In this case the *ERC-1155* tokens are therefore practically the same as traditional cryptocurrency tokens (i.e., ERC-20 tokens). Burning them can therefore reduce supply, thereby increasing their price. Third, since there are many NFTs airdropped to other accounts like *spam emails*, *EOAs* also burn the tokens by themselves, to avoid accidentally clicking on a fraudulent link.

4.2 Exploration of Participants

We next explore *who* drives the above NFT events (i.e., the accounts). We first define a weighted directed graph, the *transfer account graph*, i.e., $TAG = (V, E, w)$, where V is a set of accounts, E is a set of edges, and w is a set of integers indicating the number of transfers between two different accounts. There are 8,189,043 nodes (i.e., accounts in the NFT ecosystem) with 242,444,962 edges (i.e. transfer events). Note, we include the "null" account from which all new NFTs are initially transferred. To generalize this, Fig. 3(a) and (b) show the in and out degree distributions. As expected, the distributions are highly skewed. As with prior analysis, we observe a long tail — 40.8% of accounts have an in-degree of 1, with 35.4%, having an out-degree of 1. Just 12.8% have an in-degree over 20 (85.8% for out-degree). This suggests significant centralization in the production of NFTs.

To better understand these influential accounts, Table 4 and Table 5 of the Appendix B list the top five accounts, as measured by in and out-degree. In total, these five accounts cover 3.06% of in-degree and 64.94% of out-degree, respectively. The discrepancy is because the mint events generate a transfer events whose "from address" is null (see §3). Thus, the null address has an out degree of 148,500,667 (61.25% of the total out degree), which reveals *the low liquidity of NFTs*. Beyond the null account, we further conjecture that other accounts with very high degrees might be automated. By searching these accounts, we observe a number of automated services (see Tables 4 and 5 of the Appendix). For example, the *Ethereum*

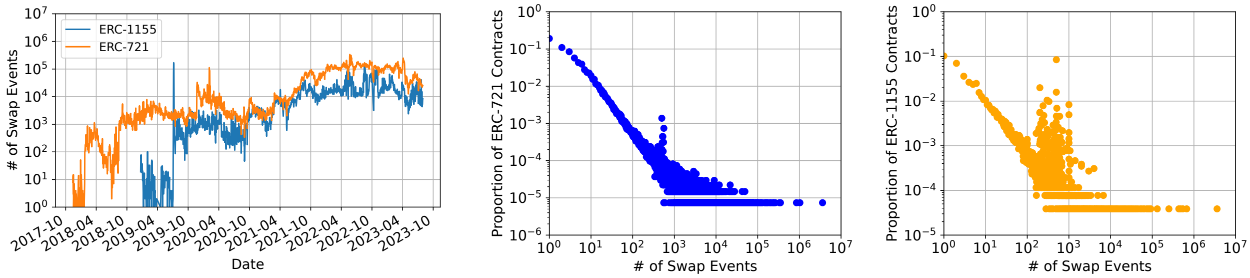


(a) The distribution of daily *mint events.*

(b) *Mint events* per *ERC-721* contract.

(c) *Mint events* per *ERC-1155* contract.

Figure 1: Graphs of *ERC-721* and *ERC-1155* of *mint events.*

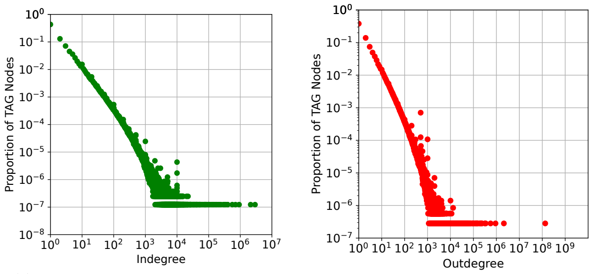


(a) The distribution of daily *swap events.*

(b) *Swap events* per *ERC-721* token.

(c) *Swap events* per *ERC-1155* token.

Figure 2: Graphs of *ERC-721* and *ERC-1155* of *swap events.*



(a) Account *in degree* distribution.

(b) Account *out degree* distribution.

Figure 3: Overview of the *transfer account graph* (TAG).

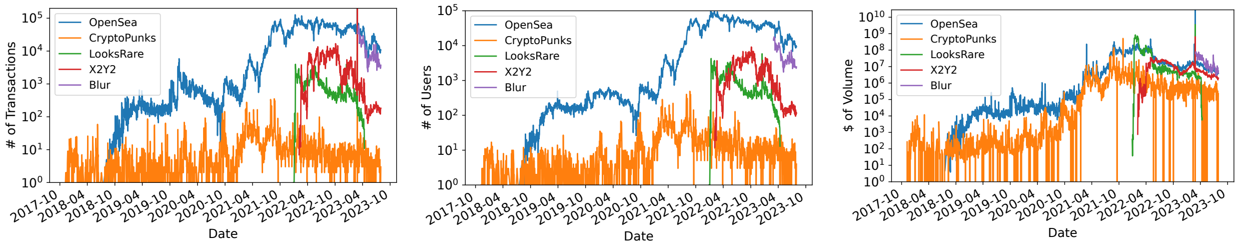
Name Service (ENS) is a naming system based on the Ethereum blockchain, which maps human-readable names (e.g., *alice.eth*) to machine-readable identifiers. Another example is *MetaWin*, which aims to provide a community-oriented brand by investing in opportunities centered around NFTs. These accounts are Dapps on Ethereum, providing different function to the NFT ecosystem. Importantly, we do not see *any* personal trading accounts attaining this large number of transfer events.

4.3 Exploration of Marketplaces

Marketplaces Overview. Recall that, the marketplaces we measure (*OpenSea*, *X2Y2*, *CrypotoPunks*, *LooksRare* and *Blur*) cover 98.1% of total trade volume in *Ethereum* in 2022 [10]. Fig. 4 presents their number of users, cumulative NFT price (i.e., volume), and transactions.

OpenSea is the most successful marketplace (across all three metrics). *OpenSea* and *CryptoPunks* are the longest running NFT marketplaces. *LooksRare* and *X2Y2* were launched later in 2022, but also have stable daily users, transactions and a large price volume. However, they are collapsing now. *Blur*, as a new market, has significant growth in 2023. After NFTs became popular in 2021, the sum price within *CryptoPunks* rapidly increased in value and held a high daily cumulative price volume (almost higher than *OpenSea*), yet only had an average of just 1,654 transactions and 1,924 users. This surprising observation is explained by the nature of the *CryptoPunks* marketplace. It was launched in 2017 with 10,000-pixel images, also called "The first non-fungible token" [45]. This small set of NFTs gained significant attention, resulting in high price trades amongst a small number of individuals. *LooksRare* has far fewer transactions on average, but occasionally outstrips *OpenSea*, with around 10^3 daily transactions and 10^3 users. Closer inspection reveals that this might be attributable to market manipulation. Specifically, *LooksRare* has its own *ERC-20* tokens to reward users based on the number of trades performed on their platform. This incentivizes fake transactions, where users exchange NFTs frequently simply to earn rewards. This observation inspires us to explore this form of NFT price manipulation in §6.1.

Collection Price. A collection is similar to a "brand", consisting of multiple tokens minted from the same smart contract. We next evaluate the value of every token using their last trade price. In total, there are 54,277 (22.23% of the total) NFT smart contracts in the market, and the sum market cap is $20B USD. The majority of NFT collections are surprisingly expensive: the average is $383,660.07 USD. The most expensive collection is an astonishing around $1.4B



(a) # of daily transactions.

(b) # of daily active users.

(c) # of daily volume.

Figure 4: A comparison of five marketplaces.

USD, with an average of $0.172B USD among the top 100 collections. Only 15.99% collections have a price under $10 USD. We observe a notable set of middle-priced NFTs though: 37.62% exceed $1000 USD. Thus, although many people may think that digital collections seldom have market value, these results suggest otherwise. For context, Table 6 in the Appendix B summarizes the top *collections* that have a value over $700*M* USD.

However, we find suspicious behaviors within the top collections. Specifically, *Meebits* is one of the most valuable collections in the NFT ecosystem. By inspecting its transactions, we observe that there are 1,655 trades with a price over $1M USD, and 152 trades with a price over $10M USD. Intuitively, these prices are suspiciously high and closer inspection reveals that they are in fact traded by the same small group of users, which also drives us into §6.1.

User Wealth. We finally inspect the overall wealth of *users*. We treat the last trade price of each NFT as its value. We identify 1,989,109 *accounts* (users) who hold NFTs. Table 7 in the Appendix shows the top *users* who hold a value of over $\$10^8$ USD. We identify four addresses that have a sum value over $\$10^9$ USD and they hold the wealth of over $1.48B USD, which is 7.4% of the total. The top 10% of the holders hold 86.71% of all NFT wealth, with a value of $18B USD. This suggests we are witnessing a consolidation of wealth in the hands of a small minority.

It is difficult to identify *who* these accounts are, however, we do find evidence that some are not authentic. For example, the top user 0xa9 [31] bought 21 tokens in LooksRare whose price is more than $1,000,000 USD during Jan 20th – Feb 10th, 2022. These NFTs belong to the first top collection Meebits, and the third top collection Loot. We conjecture that this is a suspicious activity. We therefore check the trade and find the seller is 0x35 [30], who is also listed in Table 7 of the Appendix B. During the same period, 0x35 simultaneously sells tokens to 0xa9 with a price of more than $10M USD. We find that these accounts buy each others' tokens at a high price, artificially inflating their listed value, which is assumed as a kind of price manipulation and will be discussed further in §6.1.

We have also observed that certain users engage in a large number of trade activities within secondary markets. However, the amount of wealth these users possess remains remarkably small. For instance, 0xc3 [29], who has executed 87,055 trades in secondary markets, yet the wallet still does not hold any value. We are particularly curious about this type of phenomenon, which motivates us to explore further in §6.2.

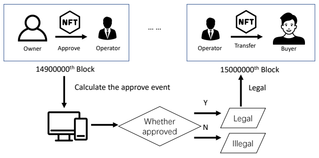


Figure 5: The mechanism of sleepmint.

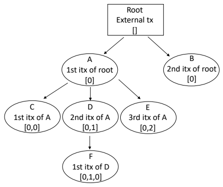


Figure 6: An example *transaction tree.*

Summary of NFT Measurement *The NFT ecosystem became popular in the middle of 2021, with significant and fluctuating growth since then. Dominant projects and NFT holders can trigger huge fluctuations in the NFT ecosystem.*

5 THE BAD: NFT SECURITY ISSUES

JT: todo update.

To measure the attacks of NFTs, we first resort to *REKT* [25] (a popular database of DeFi scams, hacks and exploits) to collect known NFT related attacks. We collect all the reported NFT attacks and after comparing with the first paper of studying the NFT security issues [46], we found a gap and identify two attacks that have not been explored but reported by a few media outlets have reported attacks that target NFT projects [1, 3, 15] [1]: (*i*) *Sleepmint*: the most special one, since a novel access control attacks that *is explicitly related to the feature of NFT*; and (*ii*) *NFT whitelist reentrancy attacks*, which is a *totally different* kind of reentrancy attack, but *to the best of our knowledge, no one has measured it in-the-wild before.*

5.1 Sleepmint: Novel Access Control Attack

Sleepminting is when someone first mints NFTs to the wallets of another famous person, then transfers ownership back to the scammers themselves (without the famous person knowing it happened), via a backdoor (a trick widely used by attackers to circumvent normal access control procedures [4]).

This creates the appearance that this account authentically minted an NFT to themselves, and then sent that NFT to the scammer. The scammer can make use of that information on-chain and declare

[1] Since the two security issues are harmful to participants, we label them as "The Bad".

```
function transferFrom(address from, address to, uint256 tokenId) public
↪  virtual override {
    require(_isApprovedOrOwner(_msgSender(), tokenId), "ERC721: transfer
    ↪  caller is not owner nor approved");
    _transfer(from, to, tokenId);
}
function _isApprovedOrOwner(address spender, uint256 tokenId) internal view
↪  virtual returns (bool) {
    require(_exists(tokenId), "ERC721: operator query for nonexistent
    ↪  token");
    address owner = ERC721C.ownerOf(tokenId);
    return (spender == owner  getApproved(tokenId) == spender
    ↪  isApprovedForAll(owner, spender)  _superOperators[_msgSender()]);
}
```

Listing 1: An example of sleepmint backdoor.

that the NFT was once owned by a famous account. Thus, the scammer can sell it for a higher price.

To achieve that goal, the attack is divided into three steps. (*i*) A sleepmint backdoor is hidden in the smart contract owned by the scammer, and used to mint the NFT; (*ii*) The scammer uses the contract to mint the NFT to someone else; and (*iii*) The scammer operates the token without the approval of the owner, and takes ownership of the token via the backdoor. To perform a sleepmint, one must withdraw the token from the destination account without permission. Thus, we can detect the transfer event without permission and identify contracts that have a sleepmint backdoor.

Case Study. Listing 1 shows an example of an NFT project, *Social BEES University (SBB)*, which has a sleepmint backdoor. When a token of *SBB* is about to be transferred, the function *transferFrom* is called. In this function, it must check the authority of the operator to find out whether they can transfer the token. However, in the function *isApprovedOrOwner*, the check is bypassed if one of four circumstances is satisfied. One of them is whether this operator is the one of the *SuperOperators*, which is an obvious sleepmint backdoor.

Detection Method. Our detection technique for sleepmint attacks is shown in Fig. 5. As mentioned in §2.3, the “*approve*” mechanism in NFT contracts allows one user to approve its NFTs to another operator, and the approved operator can handle those approved NFTs just like the original owner. We therefore focus on NFT transfer events with unapproved operators. The operation of a single NFT approval will emit an approval event, and the topics in the event include the approved user, the approved NFT’s *token ID* and the *approval state* (true or false). The *ApprovalForAll* event is quiet similar, but the *token ID* is replaced with the address of the approved operator, since this event indicates the operation of approving all of one’s NFT to another operator. Since the approve-related event has an approval state which can be overwritten by future approval operation, we must check the approval state of an NFT by the latest *Approval* or *ApprovalForAll* event before the transfer of this NFT happens. We therefore parse all blocks before Apr. 1st, 2023, and record all *Approval* and *ApprovalForAll* events with their transaction index in the block.

We filter any transfer events whose operator is the owner, and retain those whose operator is not the owner. After that, for each NFT transfer event in a transaction, we fetch all the previous approve events related to the NFT. We then re-calculate the approval state of this token. After that, we verify whether the operator of this transfer event is approved by the owner of the NFT. If it is approved, this is a legal transfer event. If it is not, there is a hidden way for the operator to transfer this NFT, i.e., a sleepmint backdoor. All the smart contracts that have unapproved transfer events with an *EOA* operator have sleepmint backdoors. Since the results are detected by transactions that has happened, the results have no false positive cases for such backdoor.

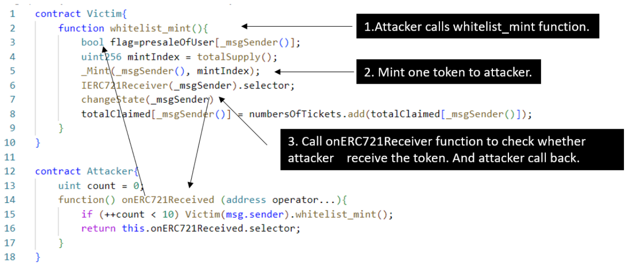
Results. We identify 99 smart contracts that have sleepmint backdoors, and 2,832 sleepmint NFTs. Among them, 944 NFTs have entered the market and been sold at least once. For example, *Beeple* [5, 23], a famous artist, has been reported to be a target [2]. The smart contract [28] that hides sleepmint backdoor has four sleepmint NFTs detected by our approach. Beyond this well-known attack, we identify 98 new sleepmint smart contracts that are not known to the community yet. To estimate the cost, we count the number of sleepmint NFTs that have been sold at least once, and calculate the profit as the total price of these NFTs. The overall profit of the tokens from the top-5 sleepmint collections is $53,997.48 USD, confirming that this attack is highly lucrative.

Comparison of Previous Study. We note that Guidi et al. [47, 48] once investigated this issue, though their definition for sleepmint is overly broad. Specifically, if an NFT was minted by $A_{attacker}$ to $A_{defender}$, or it was approved for $A_{attacker}$ by $A_{attacker}$ but owned by $A_{defender}$, or it was transferred by $A_{attacker}$ but owned by $A_{defender}$, Guidi et al. labeled it as sleepmint. Since they do not catch the key point of sleepmint (the unapproved transfer via backdoor) and have a broad definition of sleepmint, they detect over 1.3M attacks without verification. The number is far more than our results of 2,832 attacks, which shows previous detection results for this issue were mostly composed of false positives. While Guidi et al. *only* claim these events *are connected with sleepmint*, our detection *precisely measures the malicious sleepmint attacks in the wild, which revises this knowledge.*

5.2 NFT Whitelist Reentrancy Attack

An NFT whitelist reentrancy attack is rather different to other Web3 reentrancy attacks [19]. Web3 reentrancy attacks usually happen during on transferring funds or manipulating internal variables. In contrast,the NFT whitelist reentrancy attack is caused by the special whitelist mechanism of NFTs. Here, if someone wants to mint a token from the smart contract, they should pay for the mint authority and then be listed on the whitelist of the smart contracts. Thus, the attacker might be able to reuse the single mint authority to mint multiple tokens without paying extra fees in NFT creation process (see §2.3).

Case Study. Fig. 7 shows an example of NFT whitelist reentrancy. First, the attacker buys one change of mint authority. They call the *whitelist_mint* function of the victim, which can bypass the mint authority check. After that, the contract mints one token to the attacker. Next, the smart contract tries to check whether the attacker receives the token and calls the *ERC721Receiver* function launched on the account of attackers. The attacker launches a malicious function and calls back the *whitelist_mint*, while their mint authority has not been modified so the smart contract will still mint the token to the attacker. After the end of reentrancy, the state on



```
contract Victim{
    function whitelist_mint(){
        bool flag=presaleOfUser[_msgSender()];
        uint256 mintIndex = totalSupply();
        _Mint(_msgSender(), mintIndex);
        IERC721Receiver(_msgSender).selector;
        changeState(_msgSender)
        totalClaimed[_msgSender()] = numbersOfTickets.add(totalClaimed[_msgSender()]);
    }
}

contract Attacker{
    uint count = 0;
    function() onERC721Received (address operator...){
        if (++count < 10) Victim(msg.sender).whitelist_mint();
        return this.onERC721Received.selector;
    }
}
```

1.Attacker calls whitelist_mint function.

2. Mint one token to attacker.

3. Call onERC721Receiver function to check whether attacker receive the token. And attacker call back.

Figure 7: An example of NFT reentrancy attack.

the whitelist will be changed, even though the attacker has already minted many tokens.

Detection Method. The reentrancy risk in NFT contracts is introduced by the *OnReceived* callback functions. Thus, we make use of internal transactions collected by *Geth* in §**??**, including the "from address", "to address", calldata, and the trace of the internal transactions. To make the narrative clearer, we define the *transaction tree* for each of the transaction described as follow. We set the external transaction as *root*, then organize all the internal transactions as the *children* of that root. Thus, every node can be presented as a *identical list*, whose element is the index in the siblings of every node from root to this node. Fig. 6 shows one example of a transaction tree, where *tx* refers to transaction and *itx* refers to internal transaction. Each node, shown as an ellipse in Fig. 6, evokes an internal transactions. According to our definition, the root shown as a square in Fig. 6 (i.e., the external transaction) is represented as []. Each node is then represented as its path from the root, e.g., node *F* can be presented as a node list of [0, 1, 0]. We use this data structure to detect attacks.

First, we transform all NFT transactions into the above tree representations. Second, we calculate the *node list* for every node to represent every internal or external transactions. Third, if there is an external transaction which is included in more than one node list (which means the node could be reached via multiple paths from its *root*), we consider that this specific transaction has the scope to become a whitelist reentrancy attack. Finally, to ensure the multiple *node lists* of this node are caused by the *OnReceived function*, we traverse the tree and check whether it contains the *OnReceived function*. If so, this can create a loop and confirms a whitelist reentrancy attack.

Results. Using the above technique, we identify 35 whitelist reentrancy attacks, involving 6 vulnerable smart contracts (see Table **??**). Note, only one NFT project (*HypeBears*) has been reported to have a whitelist reentrancy vulnerability [20], the other five have not been revealed yet. We have reported the issues to the NFT developers. By counting the number of whitelist reentrancies, we find that 1,244 NFTs have been minted without the permission of the smart contract. 869 NFTs (69.86% of the total) have been published in the market and have a first sold price sum of $115,580.91. We define reentrancy mint events as the mint events that happen in a whitelist reentrancy transaction; we define the loss of collections as the first sold price of tokens from that collection; and define profit as the first sold price of tokens that the attacker gains. Table **??** shows the top-5 attackers that gain the largest profit. The profit reaches over $25K USD for the top attacker. This confirms that reentrancy attacks are both prominent and highly lucrative, exaggerating the real market worth of the NFT ecosystem.

> **Answer to RQ2** *We identify 2,832 sleepmint tokens and 35 NFT whitelist reentrancy attacks in the wild. Over 90% of the sleepmint and 83% of whitelist reentrancy attacked contracts have not been flagged in the community. Our work fills in the gap of insufficient research of NFT security issues.*

6 NFT MARKET MANIPULATION

The previous section has identified preliminary evidence of two kinds of market manipulation [18]. We next deep dive into two types of market manipulation: (*i*) wash trading, and (*ii*) NFT arbitrage.

6.1 NFT Wash Trading

Wash trading occurs when a set of accounts buy and sell the same assets multiple times in a short period, to deceive other (normal) market participants about an asset's price.

Pilot Study. Our prior analysis of NFT markets (see §4.3) provides evidence of this type of malicious behavior (e.g., market rules of *LooksRare* and fake trades of *Meebits*). This motivates us to conduct a pilot study. To inspect the patterns of wash trading, we define a *seller, buyer* pair, which can be represented as a triplet: <*seller, buyer, weight*>. Because the *Meebits* NFTs are sold in the *LooksRare* marketplace, we select the top 50 seller-buyer pairs according to their sum trade frequency, and represent them in a chord diagram, as shown in Fig. 8. The different color blocks represent different buyers and sellers; and the width of the arrows represents the trade frequency. There are clearly seller-buyer pairs who exchange a large number of NFTs. We also find a non-negligible number of exchanges where the two-way flow of assets are very similar — a classic sign of wash trading. Via manual inspection, we confidently identify 31 users who are almost certainly performing wash trading. From this, we identify three kinds of patterns (motifs), as shown in Fig. 9. *Motif 1:* Wash trading can happen between two users, whereby they buy and sell tokens with each other. *Motif 2:* Wash trading can happen between many pairs of accounts, with a single central user. *Motif 3:* Wash trading can occur in a cycle (i.e., a minimum of three users).

Detection Approach. The current methods for detecting wash trading [40, 46, 50, 53, 56, 57, 60] rely on a few basic patterns. However, we aim to design a novel way to identify more wash trading patterns. Specifically, our method estimates the minimum number of wash trading bots and then makes an effort to filter out cases where bots are not involved in wash trading. Based on the observation, we design an automated approach to uncover the wash trading activities in the ecosystem. Note that the three kinds of motifs we identified in the pilot study are the most simple ones that may not cover all sophisticated wash trading behaviors in the wild. Thus, in contrast to existing works that rely solely on the summarized patterns of wash trading activities, we seek to uncover the wash traders behind them, and then reveal their diverse wash trading behaviors.

To achieve this, we first apply a general heuristic method to flag suspicious trading activities, based on which we label the bots

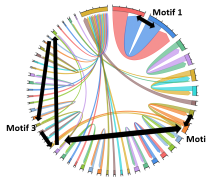


Figure 8: The top-50 pairs of LooksRare.

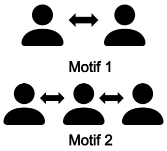


Figure 9: Three kinds of wash trading motifs.

that perform wash trading Based on the wash trading patterns observed in the manually identified bots, we further cluster the trades performed by the bots. Thus, we distinguish normal trades from potentially malicious ones. Specifically, our approach can be divided into four steps.

Step 1: Selecting Suspicious Trading Pairs. We first define a triple <*seller, buyer, weight*>, where the weight represents the number of trades between two users. Next, we filter any pairs whose sellers or users are the *official address*. For the remaining pairs, we observe that 98.75% have under five trades, which we did not observe abnormal behaviors by sampling 100 such pairs. Thus, we inspect the remaining pairs that have at least five trades as suspicious ones. We notice that all the wash trading pairs trade intensely during certain short periods (usually within 1 day). Thus, we extract all that have performed their trades within a 48 hour time window. *Step 1 finds 482,274 suspicious trading pairs.*

Step 2: Heuristic Detection. Based on the above pairs, we search for all cases of the three *Motifs* discussed in the pilot study. Note that this step may involve false positives, however, the issue will be alleviated in our following step.

Motif 1: Wash trading between two users. The first motif is where two users exchange NFTs directly between each other. To detect these from our suspicious set of users, we first compute the volume of reciprocal trades between each pair. This is modeled as a quintuple: <*user1, user2, to weight, from weight*>. If the user pairs are wash trading, the balance of trade between the two users should be approximately *equal*. Thus, we exclude any pairs where there is a over 10% difference between the incoming/outgoing trade flow. The remaining set are assumed to be wash traders.[2]

Motif 2: Wash trading with central users. The second motif is where a central user trades with many other users, as shown in Fig. 9. Each individual trade therefore looks similar to *Motif 1*, with a single central high degree user. In fact, these are the users who appear many times in the results of our *Motif 1* analysis. Thus, we identify *Motif 2* users by extracting all users identified more than once in *Motif 1*.

Motif 3: Wash trading cycle. The third form of wash trading is a cycle, containing at least three nodes. To extract all such cases, we generate a directed graph of sellers and buyers using the marketplace dataset. We then extract all the simple cycles that exist within the suspicious pairs, described in *Step 1*. Also, for the same reason, we calculate all the simple cycles in the directed graph and again filter any where the absolute differential value of trade frequency between each pair is over 10%.

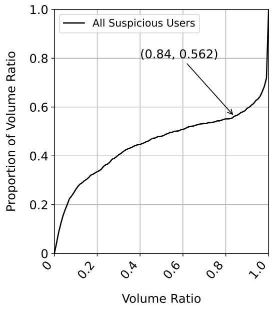


Figure 10: CDF of *volume ratios* for per user.

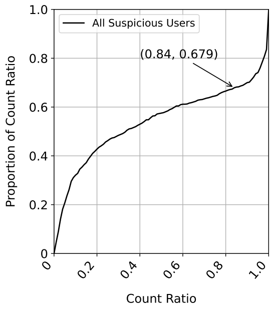


Figure 11: CDF of *count ratios* for per user.

Step 2 flags 454,537 suspicious trades according to the three trading motifs from 246,295 trading pairs in Step 1, associated with 15,148 users.

Step 3: Labelling Wash Trading Bots. The previous step is quite straightforward, yet it may contain false positives (based on a fixed threshold), and it may not cover the advanced tactics used by wash traders. Thus, we next seek to identify the wash trading bots accurately from the result of *Step 2*, and further expand our *wash trading motifs* by analyzing all their trading activities the motifs did not cover.

Specifically, we introduce two metrics to label the bots. For this, we sum up the *wash trading volume* from the 15,148 users detected in *Stage 2*. Then, we sum up the *total trading volume* of each of those users (using the marketplace dataset). After that, we calculate the ratio between those two numbers for every user, termed the *volume ratio*. Similarly, we calculate the ratio between the *wash trading count* and *total trading count* as *count ratio*. We argue that, since *wash trading bots* primarily perform wash trading, they should have either the *volume ratio* or *count ratio* near 1. If *either of* the two ratios are over a certain threshold for a specific user, we assume the user is a wash trading bot and all the trades performed by that specific account are wash trading. We next try to determine a suitable threshold. Fig. 10 and Fig. 11 are the cumulative distribution graphs of *volume ratio* and *count ratio*. The curves for both graphs increase slowly while *volume ratio* or *count ratio* is around 0.5, and rapidly increase as the *volume ratio* or *count ratio* nears 1. After 0.84 for *volume ratio* and *count ratio*, the curves increase rapidly, indicating any user above this threshold has a high possibility to be a wash trading bot. Note, the threshold is not 100%, as these bots are confirmed to have other kinds of wash trading behaviors.

Thus, we heuristically set the thresholds as 0.84 for *volume ratio* and *count ratio*. Among the 15,148 suspicious users detected in *Step 2*, we therefore label 826 bots as wash trading bots. Even if we make slight adjustments to the thresholds for *volume ratio* or *count ratio*, the identification of users in this step remains relatively consistent, which validates our choice of thresholds. To validate our heuristics, we manually check 100 of the 826 bots by sampling their trade activities, and confirm that they are all wash trading bots, which can ensure that we can get a lower-bound analysis of the issue. *Step 3 finds 826 bots from 15,148 users labelled in Step 2, flagging 85,516 suspicious trades with $24,876,390,650.34 USD trading volume.*

[2] Note, the results of *Motif 1* naturally overlap with *Motif 2*. The results from *Motif 2* is the subset of those from *Motif 1*. We therefore remove the results from *Motif 2* and retain them for *Motif 1*.

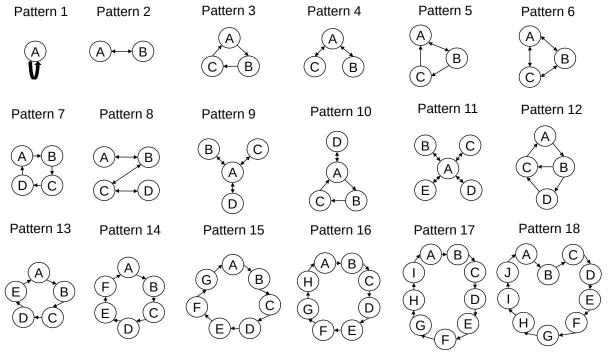


Figure 12: Summary of wash trading patterns.

Step 4: Clustering. Note that not all trades carried out by a bot necessarily involve wash trading. To filter out transactions that are not related to wash trading, we rely on the identified wash trading patterns within these bots and group together the wash trading activities within them. Consequently, we proceed to expand the trading patterns of both the trades found in *Step 2* and the newly flagged trades. This results in a comprehensive representation of wash trading behaviors, as illustrated in Fig. 12. The types of discovered patterns are beyond the scope of previous research [40, 46, 50, 53, 56, 57, 60], underscoring the effectiveness of our approach in uncovering new patterns. Based on the discovered patterns, we proceed with clustering to identify and exclude non-malicious trades. *Step 4 flags 60,971 wash trades from 85,516 trades labelled in Step 3, with $24,775,694,029.02 USD trading volume performed by 826 bots.*

Results. We flag 60,971 wash trades performed by 826 bots. These actions constitute a remarkable $24,775,694,029.02 USD, which means that at least 23.03% of NFT activity on secondary market is created by wash trading. Table 2 summarizes the breakdown of wash trading across all five marketplaces, and presents the top-8 NFT collections that have the largest wash trading volume.

Blur, as a marketplace that get popular in 2023, also have wash trading. Therefore, wash trading is a consistent problem within the NFT ecosystem. There are also notable differences across the marketplaces. Both *CryptoPunks* and *OpenSea* have only a few wash traders, whereas the vast majority takes place on *LooksRare* (over $22B USD). To explain this, we turn to the *LooksRare* official documentation [51]: "*all collections now generate trading rewards. No minimum volume required - you earn LOOKS every time your buy or sell an NFT on LooksRare, from any collection!*". This is a likely explanation for the large volume of wash trading, as users only pay a small trade fee to gain LOOKS token as rewards. This mirrors our prior observation, showing wash trading is common in *LooksRare*: From the 122 collections, exhibit 20,945 wash trading behaviors with over $22B USD fake history trading volume.

6.2 NFT Collection Offer Arbitrage

Cyclic arbitrage of fungible tokens [59] occurs because the exchange rates between different pairs of tokens in *DEXes* are not always perfectly in sync, opening up arbitrage possibilities for cyclic trading. In some countries, digital arbitrage activities may be regulated or restricted, particularly in financial markets such as currency or stock trading.[3] We therefore conjecture that arbitrage might also happen in the NFT ecosystem. Here, we refer to cycle arbitrage in traditional cryptocurrencies as *traditional e-arbitrage*, and arbitrage in the NFT ecosystem as *NFT-arbitrage*.

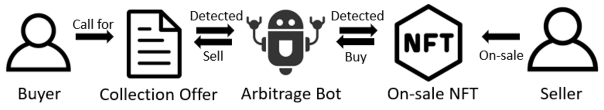


Figure 13: The flow of NFT arbitrage.

Overview of NFT-Arbitrage. Compared to *traditional e-arbitrage*, the *unique* characteristics of NFTs open up the possibility of arbitrage in a different way. Figure 13 shows the general process of NFT-arbitrage. Unlike traditional e-arbitrage, arbitrage of NFTs always begins with a *collection offer*. A collection offer is like a "wanted" for any NFT in a specific collection. In *OpenSea*, *WETH (Wrapped Ether)* is needed to make a collection offer. After raising the offer, it is shown in the OpenSea official website and the user needs to wait for the echo. *X2Y2* and *LooksRare* also have approximately the same process. To successfully perform NFT-arbitrages, three conditions must be met: (*i*) A collection offer must be raised by someone else; (*ii*) An NFT from a target collection must be listed for sale on the market; and (*iii*) The output (collection offer price) must outweigh the input (gas fees, handling fee and purchase fee). Arbitrage bots therefore must monitor the collection offers posted on marketplaces. If these three conditions are satisfied, the bot will automatically buy the token listed on the market and sell it to the collection offer. Note, to avoid undesirable changes in price, the buy and sell actions must take place within a single smart contract transaction.

Detection Method. In NFT arbitrage, the buy-and-sell actions should be completed within one transaction. This inspires us to design an effective detection method. We refer to the trade dataset as T. All the users involved in the trades are in set U. Every trade in T consists of the seller, buyer and other information. If T_1 and T_2 match the following five criteria, we label it as arbitrage: (*i*) The two trades happen in a single transaction, i.e., $T_1.transaction\_hash = T_2.transaction\_hash$. (*ii*) The token of the trade is the same, i.e., $T_1. < contract\_address, token\_id > = T_2. < contract\_address, token\_id >$. (*iii*) If the type of the token is *ERC-1155*, the amount of tokens in these two trade should be the same, i.e., $T_1.amount = T_2.amount$. (*iv*) The price of the first trade should be less than the second one, i.e., $T_1.price < T_2.price$. (*v*) To avoid including false positives by wash trading (see §6.1), $T_1.seller! = T_2.buyer$ and $T_1.seller! = T_1.buyer$ and $T_2.seller! = T_2.buyer$ If all five criteria are fulfilled, we regard this trade (pair) as arbitrage.

Results. Through the above methodology, we identify 629 users who exhibit arbitrage behavior. These users perform 157,302 cases of arbitrage. We define the *arbitrage profit* as the sale price minus the the bot purchasing price; and the *arbitrage volume* as the price that the sale price plus the bot purchasing price. These arbitrages sum up to a profit of $25,310,982.22 USD and a volume of

[3] We consider this a type of market manipulation. However, there are differing opinions on to what extent this constitutes market manipulation vs. strategic trading.

Table 2: Summary of wash trades we identified. The column "$ of Wash Trades" is the total history volume generated from wash trades. The column "$ of All Trades" is the total history volume generated from all the trades.

	Name or Address	# of Wash trades	$ of Wash trades	$ of All trades	% of Fake history volumn
Marketplace	LooksRare	20,945	22,230,486,364.41	31,473,916,119.27	70.63%
	X2Y2	11,765	2,059,696,277.77	5,920,282,010.60	34.79%
	OpenSea	22,766	453,034,260.52	64,231,558,049.82	0.71%
	Blur	5,489	31,187,981.66	3,219,154,421.63	0.97%
	CryptoPunks	6	1,289,144.65	2,702,620,665.80	0.04%
Collection	Terraforms (TERRAFORMS)	10,884	11,674,819,866.45	12,320,656,847.36	94.75%
	Meebits	7,720	7,071,806,358.50	10,061,077,548.79	70.29%
	dotdotdot (dotdotdot)	1,727	1,838,298,518.38	2,724,498,012.57	67.47%
	More Loot (MLOOT)	1361	1,451,415,137.95	4,880,660,670.93	29.73%
	Loot (LOOT)	616	600,663,668.40	1,009,972,739.01	59.47%
	Audioglyphs (AG)	738	377,160,076.54	380,729,286.42	99.06%
	CATGIRL ACADEMIA (CAT)	422	339,172,692.24	339,515,284.64	99.85%
	CryptoPhunksV2 (PHUNK)	125	275,645,653.57	285,390,139.01	96.56%

$186,188,047.24 USD. There are 38,819 cases of cross-marketplace arbitrage and 118,483 times of same-marketplace arbitrage. Table 8 (in Appendix) summarizes the top-5 arbitrage bots, each of which has gained a profit of over $800K USD. That said, 80.4% of the bots perform arbitrage fewer than 20 times, indicating that a small set of bots gain the majority of profits via arbitrage. There are 5,443 collections that have been arbitraged, and the average number of cases per collection is 28.90. Interestingly, we observe that some of the arbitraged collections also appear among the most valuable collections, e.g., *OpenSea Shared Storefront* and *Otherdeed*. This is intuitive because of the demand for NFTs from popular collections, i.e., the more offers are raised, creating more potential for arbitrage.

Summary of NFT Market Manipulation *Wash trading and NFT arbitrage both take place, affecting billions of dollars on market. At least 23% of NFT market trading is fake, generated by 826 bots. 157,302 NFT arbitrage cases are performed by 629 bots, with profits of over* $25*M USD.*

7 RELATED WORK

Research on NFTs. Wang et al. [58] study the technical component of NFTs, explaining their design and properties. They also discuss potential security issues. However, their conclusions are based on the design of NFTs or individual cases; they lack an empirical investigation into the overall NFT ecosystem, unlike this paper. Ante et al. [34] study 14 top collections of NFTs, as well as the relationship between NFTs and Ethereum by evaluating the exchange rate and other economic factors. They only focus on several large NFT projects. There are also some researchers who focus on the usage of NFTs [35–37, 39, 42, 43, 54]. However, none of these works provide a systematic overview of the NFT ecosystem from both an on-chain data and market view.

Crypto Market Manipulation. There have been works identifying price manipulation on blockchains. Cong et al. [44] study wash trading with fungible cryptocurrencies. Rug pull schemes have also been detected by Mazorra [52], Xia [61] and Huang [49]. Prior studies explore price manipulation behavior on *Ethereum* or other chains from different angles, such as wash trading [40, 46, 50, 53, 56, 57, 60], crypto rug pulls [52, 61], crypto arbitrage [33, 38, 41, 59]. Our approach can identify more patterns of wash trading than previous work and ensure the lower bound of bots. We also automatically detect the arbitrage within *NFTs*, which is different from existed detection of arbitrage within fungible tokens.

8 LIMITATION

Our study carries certain limitations. Addressing these are the foundation of our future work. First, we only track five major NFT markets. Since these markets have a complicated design, manual efforts are still a necessary part, which means we may miss some cases of misbehavior in smaller markets. That said, these markets account for most of the trading volume and will likely reflect most trading (mis)behaviors. Second, our detection for wash trading is simple, and may miss certain cases such as wash trading bots with a low impact. However, we emphasize that we are able to find more patterns than prior works [40, 46, 50, 53, 56, 57, 60].

9 CONCLUSION

This paper has conducted the first large-scale analysis of the NFT ecosystem from both an on-chain and market view. Based on datasets of both NFT transactions and trades on major marketplaces, we have looked at various dimensions. We have shown that the ecosystem is subject to substantial market manipulation, and over 23% of NFT market volume is generated artificially. Arbitrage also takes place in NFT ecosystem, bringing over $25*M* USD profits for arbitrager. Our exploration suggests that the governance of NFTs needs to be improved, and it is urgent for the research community to propose effective countermeasures to address NFT issues.

REFERENCES

[1] The gray market: How a brazen hack of that $69 million beeple revealed the true vulnerability of the nft market (and other insights). https://news.artnet.com/opinion/sleepminting-nftheft-monsieur-personne-1960744, 2021.

[2] How to sleepmint nft tokens. https://kf106.medium.com/how-to-sleepmint-nft-tokens-bc347dc148f2, 2021.

[3] What is sleepminting and will it ruin nft provenance? https://timdaub.github.io/2021/04/22/nft-sleepminting-beeple-provenance/, 2021.

[4] Backdoor (computing). https://en.wikipedia.org/wiki/Backdoor_(computing), 2022.

[5] Beeple homepage. https://www.beeple-crap.com/, 2022.

[6] Bitcoin, ethereum price level sluggish while nft sales volume surges. https://forkast.news/bitcoin-ethereum-struggle-nft-sales-volume-explodes/, 2022.

[7] Blur: Nft marketplace for pro traders. https://blur.io, 2022.

[8] Coinmarketcap. https://coinmarketcap.com/, 2022.

[9] Cryptopunks. https://cryptopunks.app/, 2022.

[10] Dappradar marketplace. https://dappradar.com/nft/marketplaces/protocol/ethereum, 2022.

[11] Difference disclosed: Erc20 vs. erc721. https://www.blockchain-council.org/ethereum/erc20-vs-erc721/, 2022.

[12] Ethplorer api. https://github.com/EverexIO/Ethplorer/wiki/Ethplorer-API, 2022.

[13] Global non-fungible token (nft) market size to reach usd 20 billion by 2028 | blueweave consulting. https://www.globenewswire.com/news-release/2022/09/12/2514295/0/en/Global-Non-Fungible-Token-NFT-Market-Size-to-Reach-USD-20-billion-by-2028-BlueWeave-Consulting.html, 2022.

[14] Looksrare - nft marketplace. https://looksrare.org/, 2022.

[15] Nft protocol omni suffers reentrancy attack, loses 1,300 eth in testing funds. https://beincrypto.com/nft-protocol-omni-reentrancy-attack-loses-1300-eth-testing-funds/, 2022.

[16] Opensea: Buy crypto collectibles, cryptokitties, decentraland, and more on ethereum. https://opensea.io/, 2022.

[17] Trading in nfts spiked 21,000% to more than $17 billion in 2021, report says. https://www.cnbc.com/2022/03/10/trading-in-nfts-spiked-21000percent-to-top-17-billion-in-2021-report.html, 2022.

[18] What is market manipulation in cryptocurrency? https://www.binance.com/en/blog/fiat/what-is-market-manipulation-in-cryptocurrency-421499824684902912, 2022.

[19] What is reentrancy? – reentrancy smart contract example. https://moralis.io/what-is-reentrancy-reentrancy-smart-contract-example/, 2022.

[20] When "safemint" becomes unsafe: Lessons from the hypebears security incident. https://medium.com/my-blockchain-development-daily-journey/%E7%95%B6-safemint-%E8%AE%8A%E5%BE%97%E4%B8%8D%E5%AE%89%E5%85%A8%E6%99%82-hypebears-%E5%AE%89%E5%85%A8%E4%BA%8B%E4%BB%B6%E7%9A%84%E6%95%99%E8%A8%93-e2b44c4c0c63, 2022.

[21] X2y2 marketplace - x2y2.io. https://x2y2.io/, 2022.

[22] 2tread the market. https://www.outlookindia.com/business/2-of-non-fungible-token-trades-globally-are-manipulated-how-to-safely-tread-the-market-news-219849, 2023.

[23] Beeple account. https://etherscan.io/address/0xc6b0562605d35ee710138402b878ffe6f2e23807, 2023.

[24] Burn account. https://etherscan.io/address/0x000000000000000000000000000000000000dead, 2023.

[25] De.fi. https://defiyield.app/rekt-database/, 2023.

[26] Geth. https://geth.ethereum.org/, 2023.

[27] Page. 0xa7206d878c5c3871826dfdb42191c49b1d11f466, 2023.

[28] Sleepmint contract. https://etherscan.io/address/0x5fbbacf00ef20193a301a5ba20acf04765fb6dac, 2023.

[29] Suspicious account. https://etherscan.io/address/0xc34349fbedd527215aae19b2e4626254ec29a13d, 2023.

[30] Suspicious user. 0x035d0ca92152d1fea18240d6c67c2adfe0cca287c, 2023.

[31] Top user. 0xa99a76dddbb9678bc33f39919bc76d279c680c89, 2023.

[32] Why market manipulation is so rampant in nfts. https://www.enchant.com/nft-market-manipulation, 2023.

[33] Guillermo Angeris, Hsien-Tang Kao, Rei Chiang, Charlie Noyes, and Tarun Chitra. An analysis of uniswap markets. 2021.

[34] Lennart Ante. Non-fungible token (nft) markets on the ethereum blockchain: Temporal development, cointegration and interrelations. *Economics of Innovation and New Technology*, pages 1–19, 2022.

[35] Mustafa Bal and Caitlin Ner. Nftracer: a non-fungible token tracking proof-of-concept using hyperledger fabric. *arXiv preprint arXiv:1905.04795*, 2019.

[36] Hong Bao and David Roubaud. Recent development in fintech: Non-fungible token, 2021.

[37] Hong Bao and David Roubaud. Non-fungible token: A systematic review and research agenda. *Journal of Risk and Financial Management*, 15(5):215, 2022.

[38] Jan Arvid Berg, Robin Fritsch, Lioba Heimbach, and Roger Wattenhofer. An empirical study of market inefficiencies in uniswap and sushiswap. *arXiv preprint arXiv:2203.07774*, 2022.

[39] SAMUEL J BOLTON and JOSEPH R CORA. Virtual equivalents of real objects (veros): A type of non-fungible token (nft) that can help fund the 3d digitization of natural history collections. *Megataxa*, 6(2):93–95, 2021.

[40] Gianluca Bonifazi, Francesco Cauteruccio, Enrico Corradini, Michele Marchetti, Daniele Montella, Simone Scarponi, Domenico Ursino, and Luca Virgili. Performing wash trading on nfts: Is the game worth the candle? *Big Data and Cognitive Computing*, 7(1):38, 2023.

[41] Naratorn Boonpeam, Warodom Werapun, and Tanakorn Karode. The arbitrage system on decentralized exchanges. In *2021 18th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON)*, pages 768–771. IEEE, 2021.

[42] Yanto Chandra. Non-fungible token-enabled entrepreneurship: A conceptual framework. *Journal of Business Venturing Insights*, 18:e00323, 2022.

[43] Ferdinando Chiacchio, Diego D'Urso, Ludovica Maria Oliveri, Alessia Spitaleri, Concetto Spampinato, and Daniela Giordano. A non-fungible token solution for the track and trace of pharmaceutical supply chain. *Applied Sciences*, 12(8):4019, 2022.

[44] Lin William Cong, Xi Li, Ke Tang, and Yang Yang. Crypto wash trading. *arXiv preprint arXiv:2108.10984*, 2021.

[45] Cryptopunks No.1 cryptopunks–the first "non-fungible token", 2022. https://dappradar.com/ethereum/collectibles/cryptopunks.

[46] Dipanjan Das, Priyanka Bose, Nicola Ruaro, Christopher Kruegel, and Giovanni Vigna. Understanding security issues in the nft ecosystem. In *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security*, pages 667–681, 2022.

[47] Barbara Guidi and Andrea Michienzi. Sleepminting, the brand new frontier of non fungible tokens fraud. In *Proceedings of the 2022 ACM Conference on Information Technology for Social Good*, pages 75–81, 2022.

[48] Barbara Guidi and Andrea Michienzi. Delving nft vulnerabilities, a sleepminting prevention system. *Multimedia Tools and Applications*, pages 1–20, 2023.

[49] Jintao Huang, Ningyu He, Kai Ma, Jiang Xiao, and Haoyu Wang. A deep dive into nft rug pulls. *arXiv preprint arXiv:2305.06108*, 2023.

[50] Massimo La Morgia, Alessandro Mei, Alberto Maria Mongardini, and Eugenio Nerio Nemmi. Nft wash trading in the ethereum blockchain. *arXiv preprint arXiv:2212.01225*, 2022.

[51] LooksRare. What are trading rewards?, August 2022.

[52] Bruno Mazorra, Victor Adan, and Vanesa Daza. Do not rug on me: Leveraging machine learning techniques for automated scam detection. *Mathematics*, 10(6):949, 2022.

[53] Sven Serneels. Detecting wash trading for nonfungible tokens. *Finance Research Letters*, 52:103374, 2023.

[54] Sakib Shahriar and Kadhim Hayawi. Nftgan: Non-fungible token art generation using generative adversarial networks. In *2022 7th International Conference on Machine Learning Technologies (ICMLT)*, pages 255–259, 2022.

[55] Yixiang Tan, Zhiying Wu, Jieli Liu, Jiajing Wu, Zibin Zheng, and Ting Chen. Bubble or not: Measurements, analyses, and findings on the ethereum erc721 and erc1155 non-fungible token ecosystem. *arXiv preprint arXiv:2301.01991*, 2023.

[56] Syed Ahzam Tariq and Imtiaz Sifat. Suspicious trading in nonfungible tokens (nfts): Evidence from wash trading. *Available at SSRN 4097642*, 2022.

[57] Victor von Wachter, Johannes Rude Jensen, Ferdinand Regner, and Omri Ross. Nft wash trading: Quantifying suspicious behaviour in nft markets. *arXiv preprint arXiv:2202.03866*, 2022.

[58] Qin Wang, Rujia Li, Qi Wang, and Shiping Chen. Non-fungible token (nft): Overview, evaluation, opportunities and challenges. *arXiv preprint arXiv:2105.07447*, 2021.

[59] Ye Wang, Yan Chen, Shuiguang Deng, and Roger Wattenhofer. Cyclic arbitrage in decentralized exchange markets. *arXiv preprint arXiv:2105.02784*, 2021.

[60] Xiaolin Wen, Yong Wang, Xuanwu Yue, Feida Zhu, and Min Zhu. Nftdisk: Visual detection of wash trading in nft markets. *arXiv preprint arXiv:2302.05863*, 2023.

[61] Pengcheng Xia, Haoyu Wang, Bingyu Gao, Weihang Su, Zhou Yu, Xiapu Luo, Chao Zhang, Xusheng Xiao, and Guoai Xu. Trade or trick? detecting and characterizing scam tokens on uniswap decentralized exchange. *Proceedings of the ACM on Measurement and Analysis of Computing Systems*, 5(3):1–26, 2021.

APPENDIX

A DETAILS OF COLLECTING SECONDARY MARKET DATASET

First, we manually inspect all the *external functions* or *public functions* in the smart contracts to find functions that directly handle trading-related information. The smart contracts emit an event when the trade process completes. We thus check the event declarations emitted by these contracts, and find several events containing information related to NFT trades. All *official smart contracts* and *relative events* of marketplaces that are taken into consideration are listed in Table 3. To automate the process, we must map the raw data in the logs to useful trading information. Thus, we take the aforementioned external and public functions as the entries of these market smart contracts, and go through the execution path in which an NFT trade can successfully complete and emit the corresponding events. We do this to help understand each field of the logged data in these trading-related events. With this insight, we manually construct a mapping between *trading information* and *on-chain log data* to help us parse the remaining data in the logs. Finally, the extracted trading information consists of the contract address, token id, buyer's address, seller's address, currency address and currency amount. We use *Ethplorer* [12] to obtain the daily average exchange rate (to USD) of all encountered cryptocurrency tokens. We compile this data for all trades within the four marketplaces.

Table 3: Smart contracts and addresses about the top-five NFT secondary markets.

	Relative Segment Name	Relative Address
OpenSea	Seaport Address (V1)	0x00000000006cee72100d161c57ada5bb2be1ca79
	Seaport Address (V2)	0x00000000006c3852cbef3e08e8df289169ede581
	Seaport Address (V2)	0x00000000006c3852cbef3e08e8df289169ede581
	Seaport Address (V3)	0x0000000000000006c7676171937c444f6bde3d6282
	Seaport Address (V4)	0x0000000000000ad24e80fd803c6ac37206a45f15
	Seaport Address (V5)	0x00000000000001ad428e4906ae43d8f9852d0dd6
	Seaport Address (V6)	0x000000000000000adc04c56bf30ac9d3c0aaf14dc
	Wywern Address (V1)	0x7be8076f4ea4a4ad08075c2508e481d6c946d12b
	Wywern Address (V2)	0x7f268357a8c2552623316e2562d90e642bb538e5
X2Y2LooksRare	LooksRare Address	0x59728544b08ab483533076417fbbb2fd0b17ce3a
	TakerAsk Event	0x68cd251d4d267c6e2034ff0088b990352b97b2002c0476587d0c4da889c11330
	TakerBid Event	0x95fb6205e23ff6bda16a2d1dba56b9ad7c783f67c96fa149785052f47696f2be
	X2Y2 Address	0x74312363e45dcaba76c59ec49a7aa8a65a67eed3
	Inventory Event	0x3cbb63f144840e5b1b0a38a7c19211d2e89de4d7c5faf8b2d3c1776c302d1d33
	Profit Event	0xe2c49856b032c255ae7e325d18109bc4e22a2804e2e49a017ec0f59f19cd447b
Blur	Blur Marketplace 1	0x000000000000ad05ccc4f10045630fb830b95127
	Blur Marketplace 2	0x39da41747a83aee658334415666f3ef92dd0d541
	Blur Marketplace 3	0xb2ecfe4e4d61f8790bbb9de2d1259b9e2410cea5
CryptoPunks	CryptoPunks Address	0xb47e3cd837ddf8e4c57f05d70ab865de6e193bbb
	PunkBought Event	0x58e5d5a525e3b40bc15abaa38b5882678db1ee68befd2f60bafe3a7fd06db9e3

B EXTREME CASES IN NFT ECOSYSTEM

As discussed in §4.1, we list the top in-degree accounts in Table 4, top out-degree accounts in Table 5, the most valuable collections in Table 6, the wealthiest users in Table 7. The top arbitrage bots that perform arbitrage with a profit of over $800K.

Table 4: Top five indegree accounts.

Account address	Indegree	Identity
0x0000000000000000000000000000000000000000	3,462,665	Official Account
0x283af0b28c62c092c9727f1ee09c02ca627eb7f5	2,160,818	ENS
0x000000000000000000000000000000000000dead	917,025	Marketplace
0x83c8f28c26bf6aaca652df1dbbe0e1b56f8baba2	916,057	Official Account
0x39da41747a83aee658334415666f3ef92dd0d541	746,025	Marketplace

Table 5: Top five outdegree accounts.

Account address	Outdegree	Identity
0x0000000000000000000000000000000000000000	148,500,667	Official account
0x283af0b28c62c092c9727f1ee09c02ca627eb7f5	2,160,811	Ethereum Name Service (ENS)
0x83c8f28c26bf6aaca652df1dbbe0e1b56f8baba2	915,920	Marketplace
0x39da41747a83aee658334415666f3ef92dd0d541	745,931	Marketplace
0x6109dd117aa5486605fc85e040ab00163a75c662	342,806	ENS: Wallet

Table 6: Top five collections that have largest value.

Collection address	Value	Name
0xb47e3cd837ddf8e4c57f05d70ab865de6e193bbb	1,434,932,716.61	CRYPTOPUNKS
0xbc4ca0eda7647a8ab7c2061c2e118a18a936f13d	1,237,039,866.62	BoredApeYachtClub
0x7bd29408f11d2bfc23c34f18275bbf23bb716bc7	723,464,798.77	Meebits
0x495f947276749ce646f68ac8c248420045cb7b5e	722,008,516.34	OpenSea Shared Storefront
0xa7d8d9ef8d8ce8992df33d8b8cf4aebabd5bd270	717,895,694.75	Art Blocks

Table 7: Top users that hold the largest value of NFTs.

User account address	Total value(USD)
0xa99a76dddbb9678bc33f39919bc76d279c680c89	592,586,076.20
0x9b5a5c5800c91af9c965b3bf06ad29caa6d00f9b	511,029,067.58
0x73ec85489681da69fb52d8b25aee0091eb2925ce	211,809,146.96
0x83c8f28c26bf6aaca652df1dbbe0e1b56f8baba2	165,675,922.31
0x35d0ca92152d1fea18240d6c67c2adfe0cca287c	46,622,000.73

Table 8: Top-5 bots that perform arbitrage with a profit of over $800K.

Bot address	**# of Arbitrage times**	**$ of Arbitrage profits**	**$ of Arbitrage volume**
0x8f44e22ac221cc25a46289d1c307d4f34a4dd6c2	9,248	5,741,249.36	9,253,846.63
0x9e9346e082d445f08fab1758984a31648c89241a	1566	2,114,383.24	7,789,423.87
0x553eea17185e5ae6bb72f9528a4c3fc1a844b859	986	1,268,150.30	6,485,947.10
0xc34349fbedd527215aae19b2e4626254ec29a13d	43,446	1,262,516.60	68,810,679.29
0x6b58007b960016b2f559dbfd809ac4dcb1febdfe	717	821,175.15	4,000,063.41

C ADVICE FOR COMMUNITY

Our findings are of key importance to the stakeholders in the NFT community. (*i*) *The governance of the NFTs:* The ecosystem has witnessed significant growth. However, considering that market manipulation and security issues are prevalent, the governance of NFTs needs to be improved. We believe a platform for evaluating NFT tokens is needed to mitigate the impact of market manipulation and security issues. The platform can adopt techniques in this work for monitoring the trades and contracts to identify wash trading, arbitrage or other security issues. Our detection techniques can be further embedded in services like markets and wallets and act as reminders for investors when they try to interact with potential high-risk NFTs. (*ii*) *NFT Creators:* The official NFT creators should be aware of potential market manipulation. It is their responsibility to actively search, understand and identify these risks. After the launch of their projects, they should regularly publish security bulletins to remind users. (*iii*) *Investors:* For NFT investors, the awareness of potential risks on NFTs should be improved. Rather than just searching for high-value or over-hyped NFTs, they should rely on trusted sources to investigate the trading history of their potential purchases. They also need to perform research on the developers behind the projects to check whether they have a bad reputation in prior projects.