Dissecting Payload-based Transaction Phishing on Ethereum

Zhuo Chen
Zhejiang University
hypothesiser.hypo@zju.edu.cn

Yufeng Hu
Zhejiang University
yufenghu@zju.edu.cn

Bowen He
Zhejiang University
bowen_os@zju.edu.cn

Dong Luo
Zhejiang University
22321053@zju.edu.cn

Lei Wu †
Zhejiang University
lei_wu@zju.edu.cn

Yajin Zhou * †
Zhejiang University
yajin_zhou@zju.edu.cn

***Abstract*—In recent years, a more advanced form of phishing has arisen on Ethereum, surpassing early-stage, simple transaction phishing. This new form, which we refer to as *payload-based* transaction phishing (PTXPHISH), manipulates smart contract interactions through the execution of malicious payloads to deceive users. PTXPHISH has rapidly emerged as a significant threat, leading to incidents that caused losses exceeding $70 million in 2023 reports. Despite its substantial impact, no previous studies have systematically explored PTXPHISH.**

In this paper, we present the first comprehensive study of the PTXPHISH on Ethereum. Firstly, we conduct a long-term data collection and put considerable effort into establishing the first ground-truth PTXPHISH dataset, consisting of 5,000 phishing transactions. Based on the dataset, we dissect PTXPHISH, categorizing phishing tactics into four primary categories and eleven sub-categories. Secondly, we propose a rule-based multi-dimensional detection approach to identify PTXPHISH, achieving an F1-score of over 99% and processing each block in an average of 390 ms. Finally, we conduct a large-scale detection spanning 300 days and discover a total of 130,637 phishing transactions on Ethereum, resulting in losses exceeding $341.9 million. Our in-depth analysis of these phishing transactions yielded valuable and insightful findings. Scammers consume approximately 13.4 ETH daily, which accounts for 12.5% of the total Ethereum gas, to propagate address poisoning scams. Additionally, our analysis reveals patterns in the cash-out process employed by phishing scammers, and we find that the top five phishing organizations are responsible for 40.7% of all losses.

Furthermore, our work has made significant contributions to mitigating real-world threats. We have reported 1,726 phishing addresses to the community, accounting for 42.7% of total community contributions during the same period. Additionally, we have sent 2,539 on-chain alert messages, assisting 1,980 victims. This research serves as a valuable reference in combating the emerging PTXPHISH and safeguarding users' assets.

* Corresponding Author.
† These authors are also affiliated at Key Laboratory of Blockchain and Cyberspace Governance of Zhejiang Province.

Network and Distributed System Security (NDSS) Symposium 2025
23 - 28 February 2025, San Diego, CA, USA
ISBN 979-8-9894372-8-3
https://dx.doi.org/10.14722/ndss.2025.23311
www.ndss-symposium.org

TABLE I: Differences between simple transaction phishing and PTXPHISH. ✔ means the scam has the feature, ✗ means the scam does not.

Phishing category		Feature		
		U [1]	T [2]	C [3]
Simple transaction phishing	Direct-transfer	✗	✗	✗
	Fake token purchase	✗	✗	✔
Payload-based transaction phishing (PTXPHISH)	Ice phish	✔	✔	✗
	NFT order	✔	✔	✗
	Address poison	✔	✔	✔
	Payable function	✔	✔	✔

[1][U] **Web3 unique phishing tactics.** Web3 unique phishing tactics arise from the EVM's design, leveraging smart contracts semantics for phishing rather than simply transferring funds.
[2][T] **Malicious transaction payload.** Malicious transaction payload refers to transactions with malicious input data executing specific phishing-related smart contracts.
[3][C] **Malicious contracts deployed by scammers.**

I. INTRODUCTION

The rapid growth of Decentralized Finance (DeFi) on Ethereum has led to a significant rise in phishing scams. As users actively participate in the DeFi ecosystem, engaging in activities such as purchasing tokens like NFTs and conducting transactions on Ethereum, phishing attempts have adapted to specifically target users' crypto assets. Unlike traditional phishing scams that focus on privacy or financial information [1], [2], [3], [4], [5], Ethereum phishing is inherently tied to transactions. Therefore, we refer to this type of phishing as *transaction phishing* in this paper.

In the early stages, transaction phishing attempts are relatively straightforward, relying on traditional tactics to deceive users. Ethereum transactions are used as a new means of carrying out these scams, rather than being the primary lure for victims. Scammers may initiate transfer transactions through websites to steal victims' crypto assets [6], [7], or entice victims to purchase fake assets [8], [9] via websites or crypto wallets. Various mitigation proposals have been suggested to address such threats, such as the detection of phishing websites to limit their widespread [10], [7], and the prediction of address risk scores based on fund flow relationship [11], [12].

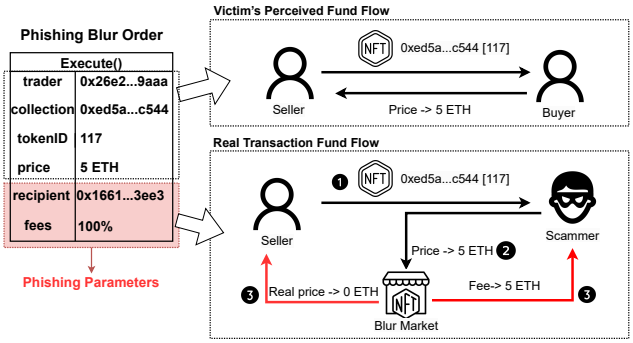

Fig. 1: A PTXPHISH example that leverages Blur order transaction semantics. From the perspective of the NFT seller, it seems as if a regular buyer is purchasing the NFT for 5 ETH. However, the scammer cleverly sets the `fees` parameter to 100% and designates himself as the `recipient`. In reality, the seller sends the NFT to the scammer ❶, and the scammer sends 5 ETH to Blur first ❷. But due to the 100% fees, Blur redirects the 5 ETH (calculated as *price * fees*) back to the scammer, who is the designated fee recipient, and sends the remaining 0 ETH (calculated as *price * (1-fees)*) to the seller ❸. As a result, the scammers appropriate the victims' NFT without making any payment.

However, with the continuous evolution of phishing tactics, more sophisticated scams are emerging that exploit complex on-chain semantics. These sophisticated scams involve scammers crafting transactions or messages [1] that *manipulate smart contract interactions through the execution of malicious payloads* to deceive users. These payloads can either be embedded within the malicious smart contracts deployed by the scammers or executed by benign smart contracts used as the executor. In this paper, we refer to these scams as *Payload-based Transaction Phishing* (PTXPHISH). Table I provides a summary of the differences between the aforementioned simple transaction phishing and PTXPHISH, with a further categorization of PTXPHISH discussed in Section III-B.

Figure 1 provides a PTXPHISH example of a malicious payload executed by a benign smart contract. The scammer manipulates the semantics of the Blur [2] order transactions to deceive the victim. The intricate transaction semantics make it difficult for users to understand the role of each parameter in the `calldata`. Consequently, victims, especially those lacking domain knowledge, may perceive that they are engaging in transactions with a reputable NFT market, while remaining unaware of the concealed malicious behavior within the transaction's parameters (*e.g.*, `fees` in Figure 1). This lack of awareness leads victims to place blind trust in the scammer, ultimately allowing the scammer to successfully appropriate the victim's NFT without making any payment. Additionally, the propagation process and tricks are detailed in Section III-B1.

PTXPHISH has been increasingly prevalent in the recent two years. For example, a significant number of PTXPHISH incidents were reported from November 2022 to July 2023 [13], [14], [15], [16], [17], [18], [19], [20], [21], [22], [23], [24], [25], resulting in cumulative financial losses exceeding $70 million. One particular PTXPHISH incident stands out, causing a loss of $24 million and ranking among the top ten blockchain attack incidents of 2023 [26]. Unfortunately, existing countermeasures have not effectively addressed PTXPHISH, as it exploits transaction semantics in carrying out its scams. Consequently, there is an urgent need to propose an effective detection method to combat PTXPHISH.

Unfortunately, despite the significant threat posed by this emerging type of phishing, the understanding of PTXPHISH is limited. Only a few studies, such as a recent work [9], have measured a fraction of PTXPHISH. The focus of this particular study [9] is primarily on visual scams that exploit wallet mistakes, without considering comprehensive contract code. However, it is crucial to perform in-depth contract code analysis to detect transactions that employ sophisticated fraudulent techniques. To the best of our knowledge, no systematic study of PTXPHISH has been conducted to date.

This work. In this paper, we present the first comprehensive study that dissects PTXPHISH on Ethereum. We first characterize PTXPHISH and then propose an effective detection approach to combat these scams. Furthermore, we conduct large-scale and long-term detection and measurement, providing valuable insights into this emerging form of phishing. Our research aims to contribute to the community's understanding and mitigation of such threats.

Specifically, we first conduct extensive data collection and build up the first ground-truth PTXPHISH dataset. Based on this dataset, we propose an in-depth analysis of the processes and tactics used in phishing scams (see Section III). This involves classifying the current PTXPHISH tactics into four main categories and eleven sub-categories. Drawing from the insights gained through this analysis, we then identify key features of PTXPHISH and propose a rule-based multi-dimensional detection approach accordingly. The effectiveness of this approach in identifying potential PTXPHISH transactions is demonstrated through a thorough evaluation, achieving over 99% F1-score and processing each block in 390 ms on average. (see Section IV). Lastly, we conduct a large-scale detection and perform an extensive analysis of PTXPHISH from three perspectives (see Section V):

- *The transactions*: we delve into phishing transactions, examining the extent of funds lost and providing detailed insights into the characteristics of each category.
- *The scammers*: we categorize scammer addresses into three types based on their behaviors: *cashiers*, *fund aggregators*, and *depositors*. Additionally, we propose an algorithm based on *cash-out* patterns, which utilizes the relationship between funds and address types to identify and track scammer organizations.
- *The victims*: we scrutinize the profiles of victims, including their address behavior features and the remedial actions they took after falling victim to phishing scams.

Our findings. In this study, we provide valuable insights into the characteristics of PTXPHISH. Our analysis of PTXPHISH transactions reveals the increasing prevalence of this type of phishing. From December 31, 2022, to October 27, 2023, the frequency of PTXPHISH escalated, resulting in significant

[1] The signed messages are initially dispatched to the scammer, who subsequently broadcasts them to the blockchain.

[2] Blur.io is one of the top NFT marketplaces on Ethereum.

economic damage exceeding $341.9 million across 130,637 transactions. Notably, approximately 4.97% of `approve` transactions and 46.22% of `permit` transactions are identified as phishing transactions. Our investigations suggest that the NFT markets prove ineffective in preventing the sale of stolen NFTs, with the majority of valuable NFTs being cashed out through platforms such as Blur (61.78%) and OpenSea (21.97%). Remarkably, scammers spent over 13.4 ETH per day in gas fees to send address poison transactions, accounting for 12.5% of the total Ethereum gas usage.

Additionally, our observations indicate a high level of organization among scammers in their cash-out process. By leveraging our cash-out pattern-based algorithm, we successfully identify the current phishing organizations. Interestingly, the top five phishing organizations are responsible for 40.7% of the total losses. Regarding the victims, our findings reveal that nearly half of them (40.38%) do not take remedial measures after incurring losses.

Contributions. Our study makes the following contributions:

- **Anatomy of PTXPHISH.** Through extensive data collection and long-term on-chain monitoring, we systematically analyze the PTXPHISH process and categorize its tactics (Section III).
- **First PTXPHISH open-source dataset.** We build the first ground-truth PTXPHISH dataset, which encompasses a comprehensive collection of **5,000** phishing transactions alongside **13,557** legitimate transactions. We release it to the community [3].
- **PTXPHISH transaction detection approach.** We propose a rule-based multi-dimensional detection approach that effectively and efficiently identifies phishing transactions, achieving an F1-score of over **99%** on both the ground-truth dataset and real-world Ethereum transactions from May 1, 2023, to Jun 1, 2023. The average processing time per block is only **390 ms**.
- **In-depth analysis of PTXPHISH.** We conduct a large-scale detection and perform an in-depth analysis of PTXPHISH to provide insightful findings from three perspectives: PTXPHISH transactions (Section V-A), PTXPHISH scammers (Section V-B), and PTXPHISH victims (Section V-C).
- **Contribution to mitigating real-world threats.** We help mitigate this emerging threat. Specifically, we have reported 1,726 phishing addresses to the community, accounting for **42.7%** of the total community contributions in the same period. Moreover, we have sent **2,539** on-chain alert messages, assisting **1,980** victims. The community has acknowledged and recognized our efforts to combat phishing attempts and protect individuals from these threats.

II. BACKGROUND

A. Ethereum Blockchain

Ethereum is a public blockchain-based distributed computing platform and operating system featuring scripting functionality. The Ethereum blockchain [27] is the most prominent framework for smart contracts [28].

Address. In Ethereum, the account can be divided into two types: externally owned account (EOA) and contract account (CA). The EOA is created by using the public-private keys and is controlled by the entity in possession of the private key. On the other hand, the CA is created by the EOA through contract creation transactions. The functionality of CA is controlled by its deployed code instead of an entity. What's more, the CA relies on EOA to execute its functions.

Transaction. During the operation of Ethereum, users can interact with other users and contracts through sending transactions. A transaction is a signed message to be sent from an EOA to another account, which carries the following information: to (receiver), from (message sender), value (the amount of native token, *i.e.*, ETH in Ethereum), data (the input for a contract call), *etc*. In particular, when a transaction sets its *to* field to be empty, Ethereum regards it as a transaction that creates a contract with its data field being the bytecode of the contract. In the end, transactions will be verified by all chain clients and be written onto the blockchain.

B. Decentralized Finance (DeFi)

Decentralized Finance (DeFi) is an emerging model for organizing and enabling cryptocurrency-based transactions [28]. In Ethereum, DeFi is built on top of multiple smart contracts, giving rise to projects such as lending, trading, and marketing [29], [30], [31].

Token. In Ethereum chains, tokens are digital assets. Unlike native cryptocurrency (*i.e.*, ETH in Ethereum), tokens are implemented using specialized smart contracts. There are two main types of tokens: fungible and non-fungible.

Fungible tokens, which are homogeneous and interchangeable, mostly conform to the same interface standard. These tokens serve as a complement to the native currency, playing the role of a more flexible secondary currency within the DeFi ecosystem. In contrast, non-fungible tokens (NFT) conform to a different type of interface, such as ERC-721/1155 in ETH. These tokens are identified with unique `_tokenID`, representing a digital asset such as ENS domains or pictures.

ERC-20/721 are currently the most widely used standards for token implementation on Ethereum. The standard interface defines a set of API methods that a token contract needs to implement. Some important API methods relevant to our study are listed in Figure 7 (in the appendix). The `approve` method approves the `_spender` as the operator of the token (msg.receiver) with `_value` (ERC-20) or `_tokenId` (ERC-721). In ERC-721, the `setApprovalForAll` method can either add or remove the address *_operator* from/to the set of the operators authorized by the msg.sender. The spender can call the `transferFrom` method to transfer the token (within the approve `_value` in ERC-20, or the same `_tokenId` in ERC-721) from the current owner's `_from` address to the `_to` address.

NFT Marketplaces. NFT marketplaces are decentralized application (dApp) platforms where NFTs are traded. Typically, there are two main components of an NFT marketplace: a user-facing web interface and a collection of smart contracts that interact with the blockchain. Users interact with the web app, which in turn sends transactions to the smart contracts. To facilitate these transactions, these marketplaces have implemented many methods to help users place orders, make purchases, and transfer NFTs in batches.

[3] https://github.com/HypoopyH/PTXPhish

III. ANATOMY OF PTXPHISH

In this section, we first describe our data collection process of the PTXPHISH dataset. We then analyze phishing tactics and categorize current phishing scams. Finally, we evaluate the coverage and effectiveness of our anatomy.

A. Data Collection of PTXPHISH

Currently, there is no centralized source of information dedicated to PTXPHISH, and public information sources are diverse. To address this gap, we have created the first ground-truth phishing transaction dataset. Specifically, our dataset was established through the following steps:

- **Collecting public reports.** We gathered public reports from two sources, *i.e.*, the phishing complaints made by victims on social media and the phishing blogs reported by the security community [32], [33], [34], [35], [36]. By querying keywords related to *phishing*, *scam*, and *drainer*, we identified relevant websites. Our information collection lasted for three months and resulted in 101 public phishing complaints and reports.
- **Reviewing public reports.** Due to the diverse sources of phishing reports, these phishing reports are in different formats and lack authoritative verification. To ensure the accuracy of the dataset, a manual review was conducted by two security experts. They analyzed the transaction data, logs, tokens transferred, and transaction call traces. A consensus was reached by the two experts to label a transaction as phishing. During the review process, we recorded scammers' and victims' addresses, transaction parameters, and transaction hashes to standardize the data format.
- **Expanding from the historical phishing data.** To increase the number of phishing transactions, we reviewed the transaction history of the scammers' addresses collected from the public reports. Random historical transactions were selected from each scammer's address, with an additional 50 transactions chosen for each phishing address [4]. The extended transactions underwent manual review as in the previous step. Notably, through our data extension and manual reviews, we have found some hidden phishing scams and provided several first-of-its-kind reports of new scams (*i.e.*, Blur free buy order, dust value poison).

By doing so, we have established the first ground-truth PTXPHISH dataset, which consists of 5,000 phishing transactions. The PTXPHISH dataset is further categorized into different phishing categories (see Section III-B), including 2,569 *ice phishing* transactions, 609 *NFT order* transactions, 226 *address poisoning* transactions, and 1,596 *payable function* transactions. The detailed information can be found in Table IX in the appendix due to the page limit.

Furthermore, we built a benign dataset for comparison by collecting transactions from two distinct sources:

- Top 50 DeBank [5] Key Opinion Leaders (KOL). These influential users significantly impact the investment community and have a large following, which bolsters the credibility of their transactions.
- Top 10 DeFi Protocol Developers [6]. These high-level developers are prominent in the DeFi space, and their widely used contracts underscore the legitimacy of their transactions.

To ensure comprehensive representation and maintain a balanced sample size, we randomly selected 200 transactions for each user [7]. In total, we gathered 13,557 benign transactions.

B. Categorization of PTXPHISH

Based on the ground-truth dataset, our analysis reveals that scammers employ two distinct strategies: *(i)* Abusing legitimate contracts; and *(ii)* Exploiting phishing contracts, as depicted in Figure 2. In the following, we delve into the details of these strategies, including their progress and specific tactics.

1) Abusing legitimate contract: As depicted in Figure 2, abusing legitimate smart contracts involves three steps. We provide a thorough description of them in the following:

- *Step I: Scammer abuses legitimate contracts to construct phishing transactions.* At first, the scammer analyzes well-known DeFi projects' contracts (*e.g.*, ERC-20 token contracts, NFT market contracts, and Uniswap contracts) and their functions. Subsequently, based on some functions of these contracts, the scammer constructs a set of transactions with malicious semantics. Although the interaction targets of these transactions are legitimate contracts, their actual behavior will cause phishing scams.
- *Step II: Scammer spreads phishing transactions through websites.* Generally, the scammer conceals phishing transactions within fraudulent websites and promotes them on social media platforms such as Twitter, Telegram, Instagram, and Discord. When visiting fake websites, victims would connect their wallets and be asked to sign a transaction [8]. Unfortunately, victims only understand that they are interacting with authorized contracts but are unaware of the real consequences of the phishing transactions, leading them to place blind trust in the phishing transactions.
- *Step III: Victims sign phishing transactions and lose assets.* Once the victim signs and submits the transaction to the Ethereum client, the legitimate contract will be executed, transferring/authorizing the victim's assets to the scammer.

The essence of abusing existing legitimate contracts involves deceiving victims by making them believe the transactions conducted with authoritative contracts are legitimate. Therefore, based on the types and methods of the exploited legitimate contracts, we can divide them into two categories: *ice phishing* and *market order* scams.

Scam Category I: Ice phishing scam. The ice phishing scam exploits the `approve` function in token contract (see Section II). Token owners can call the `approve` function to give an address the right to control a certain amount of their tokens.

However, the interface does not impose any limitations on the spender. Specifically, *(i)* the spender can be any address, no matter whether it is a Contract Account (CA) or an Externally

[4]Our investigation suggests that a threshold of 50 is typically enough to cover the majority of phishing techniques, see Appendix B for details.

[5]A well-known website for tracking Web3 portfolio [37].

[6]Based on DefiLlama [38], a top site for DeFi project rankings.

[7]Addresses with fewer than 200 transactions were included in full.

[8]Since the user's address is different, the phishing website will adjust the phishing transaction request according to the user's address.

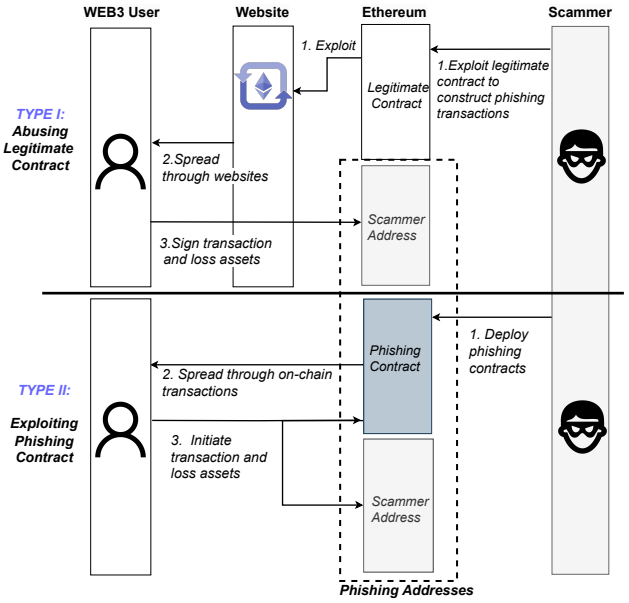

Fig. 2: Anatomy of PTXPHISH. According to the strategies, PTXPHISH is divided into two types: (i) Abusing legitimate contracts. (ii) Exploiting phishing contracts.

Owned Account (EOA); *(ii)* the spender has the ability to transfer approved amount of tokens to any other addresses. In other words, if the spender is an EOA address, it can arbitrarily transfer the owner's assets to any address without the owner's consent. There are three specific sub-categories:

- ◇ *I-A: Approve.* Targeting victims' ERC-20 tokens, the scammer constructs phishing transactions with `approve` (ERC-20 standard interface) and `increaseAllowance` (optional ERC-20 interface) to lure victims to sign.
- ◇ *I-B: Permit.* The `permit` function performs the same role as the `approve` function but allows for off-chain signing. Exploiting this feature, the scammer creates off-chain ERC20 `permit` messages and lures victims into signing them. The scammer then submits the `permit` transaction to Ethereum.
- ◇ *I-C: SetApproveForAll.* Turning to NFTs, the scammer exploits the `setApproveForAll` function of NFT collections, which can approve an entire NFT collection to an address within a single transaction.

Scam Category II: NFT order scam. NFT order scams specifically target popular NFTs owned by victims. Since the majority of users manage and trade their NFTs through dedicated NFT markets such as OpenSea [39] and Blur [40], scammers abuse existing NFT market contracts to construct deceptive transactions.

Due to the lack of unified interfaces, NFT markets have implemented their own market order contracts. These contracts are highly complex, making it challenging for users to comprehend the corresponding transactions. Even wallets are only able to display raw data without providing clear explanations. Consequently, we have observed three commonly employed tactics in these scams.

- ◇ *II-A: Bulk transfer.* Aiming to simplify the process of transferring multiple NFTs to a recipient address, OpenSea introduced a convenient function called `bulkTransfer`. Regrettably, scammers exploit this function by surreptitiously replacing the intended `recipient` address with their own, thereby diverting the NFTs to their control.
- ◇ *II-B: Proxy upgrade.* In the early stage, OpenSea implemented a proxy contract to streamline the trading process for its users. By default, this proxy contract initially grants operator rights over the user's NFTs. Exploiting this feature, scammers deceive users into signing a `proxy upgrade` transaction, which replaces the proxy contract's implementation with a scammer-controlled contract. As a result, the scammers gain ownership of the proxy contract [41], thereby enabling them to steal the user's NFTs through the manipulated proxy contract.
- ◇ *II-C: Free buy order.* In contrast to traditional centralized markets, NFT markets utilize a combination of front-end web pages and smart contracts [42]. Specifically, users use the front-end interface to sign an off-chain message that describes their order details, including the floor price and the trade time window. Upon matching the order, the market automatically completes the remaining details, such as the recipient and the final price.
  Exploiting this design, scammers construct transactions with malicious parameters that ultimately result in the loss of the NFT owner. As illustrated in Figure 1, the occurrence of a free buy order is caused by a malicious 100% `fees` parameter intentionally set by the scammer.

```
1  // transferFrom 0 token => _value = 0
2  function transferFrom(address _from, address _to,
       uint _value){
3  // default _allowance is 0 => _allowance = 0
4  var _allowance = allowed[_from][msg.sender];
5  // _allowance - _value = 0 => pass check
6  if (_allowance < MAX_UINT) {
7      allowed[_from][msg.sender] = _allowance.sub(
           _value);}
8      ...
9  // _from transfer 0 token to _to
10 Transfer(_from, _to, _value);}
```

Fig. 3: Simplified `transferFrom` function of the USDT token. Any zero value transfer between two addresses is permitted.

2) Exploiting phishing contracts: As depicted in Figure 2, the process of exploiting malicious contracts deployed by scammers involves three steps. We provide a detailed description of each step below:

- ○ *Step I: Scammer deploys phishing contracts.* In this kind of scam, scammers begin by deploying one or a group of contracts with different functionalities. Common malicious contracts include fake token, broadcoast, and trap contracts.
- ○ *Step II: Phishing contracts spread fake information to victims through transactions.* In contrast to spreading scams through websites, scammers employ broadcast contracts to spread fake information to users through on-chain transactions. These transactions are specially designed to contain false information that can contaminate users' wallets. For example, they can poison users' transaction records or airdrop tokens with fake information.

```
contract SecurityUpdates {
address private owner;
constructor() { owner = msg.sender }
function withdraw() public payable {
 require(msg.sender==owner, "Bro? Are you idiot?")
 payable (msg.sender).transfer(address(this).
     balance)
}
function SecurityUpdate() public payable {
 if (msg.value > 0) payable(owner).transfer(
     address(this).balance)
}}
```

Fig. 4: An example of a malicious `SecurityUpdate` function. This fraudulent implementation has the `payable` modifier to receive the victim's native tokens. When victims attempt to withdraw their funds, the scammer will mock them.

- *Step III: Victims believe the fake information, initiate transactions and lose assets.* The victims believe the information appeared in their wallets and initiate transactions to phishing addresses. Unfortunately, these user-initiated transactions lead to the loss of their assets.

According to the different contracts the scammers deployed, we can divide them into two categories: *address poisoning*, and *payable function* scam.

Scam Category III: Address poisoning scam. The address poisoning scam is a distinct type of scam within the blockchain ecosystem. Its primary objective is to create fake transactions between fake addresses and user addresses actively. By doing so, the scammer effectively contaminates the user's transaction records with these fake addresses [9].

For ease of understanding, we show a famous address poisoning scam example [43]. Initially, Binance sent 10 million USDT to a legitimate deposit address (*0xa7B4BAC8f0f9692e56750aEFB5f6cB5516E90570*). After monitoring this transfer, the scammer creates a counterfeit address (*0xa7Bf48749D2E4aA29e3209879956b9bAa9E90570*) that has the same GUI (`0xa7B4...0570`) in various wallets. And then, the scammer `transferFrom` 10 million fake USDT from Binance to the fake address. This action leaves a record of the fake address in Binance's transfer history. In a crucial misstep, Binance mistakenly believed the fake address in transfer history and transferred another 20 million USDT to the fake address in transaction [10]. This mistaken behavior results in the loss of funds.

The scam exploits the victims who mistakenly believe that all the history records are initiated by themselves. Specifically, there are three sub-categories, as follows:

- *III-A: Zero value transfer.* The interface of ERC-20 tokens specifies that the spender can only transfer tokens within the approved amount. However, by default, the approved amount is set to zero. Exploiting this default behavior, scammers can invoke the `transferFrom` function to transfer zero tokens from the victim's address to a fraudulent address, as depicted in Figure 3. Even though no tokens are transferred, this action leaves a transfer record in the victim's transaction history, potentially misleading the victim.
- *III-B: Fake token transfer.* The scammers deploy fake tokens with the same name/symbol as authoritative tokens. What's more, the scammers remove `allowance` check so they can call `transferFrom` to transfer any fake tokens from the victim's address to a fake address. By doing so, scammers can leave fake addresses in the victim's transfer history.
- *III-C: Dust value transfer.* The scammer sends a small number of authoritative tokens from a fake address to the victim's address. This leaves the fake address in the victim's transfer history. Since they are tiny amounts, they are called dust value transfer.

Scam Category IV: Payable function scam. Due to the absence of an auditing mechanism on the blockchain, the functionality of smart contract functions may not comply with interface protocols but may instead be determined by the smart contract developer.

For better understanding, we show a concrete example in Figure 4. The scammer poses as a legitimate project and deceives victims into believing this is a standard interface. However, the malicious `SecurityUpdate` function accepts the victims' native tokens (via the `payable` modifier), while the `withdraw` function permits only the "owner" of the contract (typically the scammer) to withdraw the tokens. The victim will incur losses upon calling this function with native tokens.

Specifically, there are two major sub-categories, as follows:

- *IV-A: Airdrop function.* Airdrops are common in DeFi [44]. Scammers exploit users' greed and pretend to be an airdrop project. They first airdrop fake tokens to victims and lure victims into calling standard airdrop interfaces, such as the `Claim`, `ClaimReward`, `ClaimRewards`. After victims call the function, they steal victims' native tokens.
- *IV-B: Wallet function.* Most users use wallets to manage their addresses. The scammer pretends to be the user's wallet and sends a message, asking the user to call functions similar to the wallet's functionality and steal their native token. For example, the `SecurityUpdate` function pretends to be a wallet update, and the `ConnectWallet` function pretends to be a wallet connection.

From the analysis of PTXPHISH described earlier, it is evident that malicious payloads are employed as fraudulent tactics, leading to notable distinctions between the content of on-chain transactions in comparison to benign transactions. Additionally, the diverse nature of scam techniques allows for the differentiation of each category based on the transaction content associated with specific techniques. Extracting key features from these distinctions forms the foundation for the detection approach outlined in Section IV.

Finding #1: PTXPHISH employs malicious payloads as fraudulent tactics, leading to notable distinctions from benign transactions. Moreover, these PTXPHISH transactions can be accurately classified into sub-categories based on various techniques utilized.

[9] The full length of an address is 20, so the GUI of wallets commonly omits a part of the address, causing similar addresses to display the same.

[10] Transaction hash: 0x08255ca0e42a872559437141fa46980e66d907f7668922467d67515b1ebb4b7f

TABLE II: Comparison of our categorization with Etherscan phishing address nametag.

Our Categorization		Etherscan's NameTag	
Type	number (#)	Type	number (#)
Ice phishing scam	363	-	
NFT order scam	80	-	
Address poisoning scam [1]	4,166	Address poisoning scam	3,561
Payable function scam	15	-	
Unknown	506	Unknown	1,569

[1][*] The address poisoning label indicates a potentially harmful address that could lead to poisoning-related losses, rather than losses that have already occurred.

C. Evaluation of PTXPHISH Anatomy

To ensure the coverage and effectiveness of PTXPHISH anatomy, we evaluate our classification by comparing it to well-known phishing labels. Specifically, we choose to utilize the Etherscan [11] `Fake_Phishing` nametags, which are the largest publicly available source of phishing nametags. However, during our investigation, we encountered certain issues with the data fetched from Etherscan. For instance, we found that addresses belonging to the `Hacker` subcategory were separate and distinct from phishing and should not be included under `Fake_Phishing`. Additionally, the `Fake token` subcategory represented simple transaction phishing, which fell outside the scope of our study. Consequently, we excluded these addresses from our analysis. As a result, there are two effective subcategories provided by Etherscan for PTXPHISH:

- **Address poisoning scam.** The description states the address related to address poisoning scams, *e.g.*, *"This address may be attempting to impersonate a similar-looking address"* and *"Zero Value Token Transfer Phishing"*.
- **Unknown.** The description lacks a specific reason, *e.g.*, *"involved with a phishing campaign"*, and *"involved in suspicious activities"*.

Accordingly, we collected a total of 5,130 addresses along with their corresponding phishing labels from May 10, 2023, to July 20, 2023. The quantities of each nametag type are presented in Table II. Comparing our classification to Etherscan, we achieved a more comprehensive coverage and broader inclusion of phishing labels. Our classification encompassed four types, providing a coverage rate of 91.2%, with only 9.8% of addresses labeled as `Unknown`. For the remaining 506 unknown addresses, we conducted additional manual analysis. Some of these labels were assigned because the addresses had been identified as phishing addresses on other EVM-compatible chains, even though they were not phishing on Ethereum. Others, according to a multi-chain search conducted by DeBank [37], were found to be completely empty addresses. Since no recorded phishing transactions were associated with these addresses on Ethereum, we were unable to classify them using the available data.

In summary, our anatomy achieves a better coverage of addresses with valid transactions, allowing for a comprehensive analysis of the phishing landscape on Ethereum.

[11]The entity information has been verified by Etherscan.

IV. DETECTION OF PTXPHISH

In this section, we first introduce the key features for PTXPHISH detection based on the previous analysis. We then propose a rule-based detection approach and evaluate its effectiveness using the ground-truth dataset.

A. Key features for PTXPHISH detection

Drawing from the insights gained through the categorization (see Table III-B), we extract four key features for phishing transaction detection:

- *Contract code called by the transaction (**Code**)*. For transactions involving contracts, we capture the relevant contract code, including the bytecode, .sol files, and ABI files (if the contract is open source).
- *Transaction input data (**InputData**)*. The input data of a transaction is composed of the hash of the function and its corresponding parameter arguments. We parse the input data based on the ABI file of the called contract, allowing us to extract specific function and parameter information [12].
- *Transaction-related addresses (**Address**)*. We collect all addresses involved in the transaction, including the caller, the callee of the transaction, and the addresses parsed from the parameter information.
- *Transaction history (**History**)*. For the $tx.from.address$ of the transaction (*i.e.*, `msg.sender`), we collect their transaction history, which includes all transactions related to these addresses.

B. Rule-based PTXPHISH detection approach

By integrating these features, we propose a rule-based multi-dimensional detection approach. This approach involves employing customized detection methods for each category, utilizing specific detection features to ensure high accuracy. The detailed detection rules are outlined in Table III. In the following, we elaborate on the detection methods for each phishing category:

- **Ice Phishing Scam Detection**. This type of scam tactic abuses legitimate contracts. Firstly, we collect a list of authorized addresses (called *Authorized_List*) obtained from Etherscan, including those associated with decentralized exchanges (DEX) and DeFi projects. Specifically, we use the Etherscan label cloud page [13], search for the top 100 DeFi project ranking on DefiLlma [38], and record the searched addresses within these label groups. Next, we set up the prerequisites rule of ice phishing scam: when we encounter a transaction that involves valuable fund transfers and notice a discrepancy between the $tx.from.address$ and $transfer.from.address$, we conduct further analysis on the $tx.from.address$. If the target address is unauthorized (not in the *Authorized_List*) and transfers all existing funds in the $transfer.from.address$, do we classify the transaction as *I: Ice Phishing scam*.
  To recognize each sub-category, we proceed to gather the transaction history of the $tx.from.address$ and $transfer.from.address$. Based on the various transaction

[12]In the case of phishing that abuses legitimate contracts, it is essential to note that the legitimate contracts are typically open-source.

[13]Etherscan label cloud https://etherscan.io/tokens/label

TABLE III: PTXPHISH detection rules. The prerequisite serves as the condition criterion for a broad category (*i.e.*, I: Ice Phishing). Upon fulfillment of prerequisites, the detection is further subdivided into respective rules based on sub-categories (*i.e.*. I-A: Approve).

Phishing Category		Rules
I: Ice Phishing	Prerequisites	tx has transfer & tx.from $\neq$ transfer.from & tx.from $\notin$ Authorized_List & transfer.value = transfer.from.value
	I-A: Approve	$\exists$ ['approve','increaseAllowance'] in transfer.from.History, authorized_address = tx.from
	I-B: Permit	$\exists$ ['permit','permit2'] in transfer.from.History, authorized_address = tx.from
	I-C: setApproveForAll	$\exists$ ['setApproveForAll'] in transfer.from.History, authorized_address = tx.from
II: NFT Order	Prerequisites	tx.to.Address $\in$ NFTMarkt
	II-A: Bulk transfer	param.func = 'bulkTransfer' & param.recepient $\neq$ tx.from
	II-B: Proxy upgrade	param = 'upgradeto' & tx.from.Address $\neq$ param.owner
	II-C: Free buy order	param.price = 0 param.fees = 100% param.recipient $\neq$ param.offerer
III: Address Poisoning	Prerequisites	tx has transfer & $\exists$ transfer' in tx.from.History.transfer transfer'.to.address = transfer.to.address & $\exists$ transfer" in tx.from.History.transfer transfer".to.address $\approx$ transfer'.to.address & transfer".value >0
	III-A: Zero value	transfer'.value = 0
	III-B: Fake token	transfer'.token $\in$ fake token
	III-C: Dust value	transfer'.value <0.01
IV: Payable Function	Prerequisites	tx.value >0 & close_source(tx.to.Code) & tx.log = null
	IV-A: Airdrop function	tx.InputData.funcsig $\in$ Airdrop
	IV-B: Wallet function	tx.InputData.funcsig $\in$ Wallet

types identified from the transaction history, we categorize the transaction into subcategories such as *I-A: approve*, *I-B: permit*, or *I:C setApproveForAll*.

- **NFT Order Scam Detection**. This type of scam tactic abuses legitimate contracts. First, we apply a prerequisite to isolate transactions based on the transaction callee $tx.to.address$. In this study, we only focus on the addresses that belong to the famous NFT markets (*e.g.*, Opensea, Blur, X2Y2) [14]. Then, combining the contract Code and ABI file, we parse the transaction Input Data to get the *parameters*. When the parameters meet the function `bulkTransfer` and the recipient is not the transaction $tx.from.address$, we label them as the *II-A: bulk transfer scam*. Seamless, if the parameters meet the function `upgradeTo`, we check whether the owner is the $tx.from.address$ to judge if it is a *II-B: proxy upgrade scam*.
  Turn to free buy order scam, we mainly focus on the conditions given by the seller, including NFT price, receipt address, and tips. *(i)* the seller signs a sales order where the NFT price is \$0, *i.e.*, without `collection` in Seaport 1.1 `fullfilAdvancedOrder`. *(ii)* the seller gives an incredibly high fee to the buyer, *i.e.*, the 100% `fees` in Blur `execute`. It results in the same result of zero buy, see Figure 1. *(iii)* the order recipient is not the NFT seller, *i.e.*, the seller gives his `WETH` to the buyer in Blur `execute`. When a transaction exhibits any of these abnormal behaviors, we classify it as a *II-C: free buy order scam*.
- **Address Poisoning Scam Detection**. Address poisoning scams adhere to a prerequisite, regardless of the specific deceptive techniques employed (*i.e.*, fake token, zero value, or dust transfer): *(i)* When the victim sends a phishing transaction with a transfer (*i.e.*, tx has transfer). The fake address already exists in historical transactions. (*i.e.*, (*i.e.*, $\exists$ transfer' in $tx.from.History$, $transfer'.to.address$ = $transfer.to.address$)) *(ii)* Before the scammer imitates a fraudulent transfer record from victim to a fake similar address, it is essential that the address has already sent valuable tokens to the genuine address, where the fake address is highly similar to the genuine address (*i.e.*, $\exists$ transfer" in $tx.from.History.transfer$, $transfer''.to.address \approx transfer'.to.address$ & $transfer''.transfer.value >0$). Specifically, when we observe that the first 4 bits and the last 4 bits of two addresses are identical, we consider these addresses to exhibit a high degree of similarity. After we encounter a PTXPHISH transfer, we finally conduct preliminary matching of transactions with suspicious transfer behavior, *i.e.*, zero value transfer, fake token transfer, and dust value transfer.
- **Payable Function Scam Detection**. The payable function scam relies on masquerading as innocuous function names to lure victims. After we observe many famous DeFi projects' functions, we observe a pattern: *(i)* most functions are open-source. *(i)* most functions are not `payable`, which means they can not receive users' native tokens. *(iii)* most functions have implementation logic that is not empty.
  Inspired by that, we first collect function signatures with sensitive names from Ethereum 4byte Signature Database [45], such as claim, claimRewards, and Claim. Based on their function name, we separate the function signatures into *Airdrop* and *Wallet* classes. Next, we establish the prerequisites for our detection approach: we consider only valuable transactions ($tx.value >0$) that have no associated transaction logs ($tx.log$ = null). In such cases, we attempt to retrieve the contract source code. If the source code is inaccessible (*i.e.*., closed-source), we classify the transaction as an *IV: Payable Function* scam and further classify the sub-categories (*i.e.*, *IV:A Airdrop function*, *IV:B Wallet function*) based on the corresponding function signatures.

C. Evaluation of PTXPHISH detection approach

We have implemented a prototype to evaluate our detection approach. First, to expedite the collection of Ethereum transaction information, we set up a local Ethereum archive node following the methodology described by Feng et al. [46]. Additionally, to speed up the history data collection, we accelerated the historical transaction replay process by following the techniques outlined by Wu et al. [47]. Finally, we implemented our aforementioned detection rules using Golang.

Besides the prototype implementation, we collected two datasets, *i.e.* the ground-truth dataset (see Section III-A), and a large-scale dataset consisting of Ethereum transactions from May 1, 2023, to Jun 1, 2023. The large-scale dataset includes 210,000 blocks with 30,976,209 transactions. In the following

[14]In detail, the Seaport 1.1, Seaport 1.2, Seaport 1.3, Seaport 1.4, Blur.io Marketplace, Blur.io Marketplace 2.0, Opensea Helper, Opensea Factory

TABLE IV: Accuracy evaluation of the detection approach.

Category	Number (#)	TP/FP/FN
Ground-truth		
Benign	**13557**	**13555/1/2**
PTXPHISH	**5000**	**4999/2/1**
Ice phishing	2569	2568/0/1
NFT order	609	609/0/0
Address poisoning	226	226/0/0
Payable function	1596	1596/2/0
Large-scale		
PTXPHISH	-	**12050/84/6** [1]

[1] For false negatives, we manually reviewed not detected as PTXPHISH transactions but initiated by addresses labeled as `Fake_Phishing` by Etherscan within the same timeframe.

sub-sections, we will first use the ground-truth dataset to assess the accuracy of our approach. After that, we will apply our approach to the large-scale dataset to evaluate its real-world accuracy and efficiency.

1) Accuracy Evaluation: For the accuracy evaluation on the ground-truth dataset, we conducted separate accuracy assessments for each phishing category, as shown in the table IV. The table demonstrates that our detection approach achieves remarkably high accuracy on the ground-truth dataset, with an overall F1-score over 99.9% (only 2 FPs in payable function and 1 FN in ice phishing).

For the large-scale dataset, we detected 12,050 PTXPHISH transactions. To evaluate false positives (FPs), our research team manually reviewed these transactions using the process described in Section III-A. However, manually evaluating false negatives (FNs) in the same manner was impractical due to the large volume of transactions. Therefore, for transactions not detected as PTXPHISH, we collected their initiating addresses to check if they were flagged as `Fake_Phishing` by Etherscan within the same timeframe. We then manually reviewed transactions initiated by addresses labeled as `Fake_Phishing`. If a transaction was confirmed to be phishing, it was classified as an FN. The results are summarized in Table IV: 84 transactions were identified as FP (4 in ice phishing and 80 in misleading), and 6 transactions were identified as FN (all in NFT order), resulting in an overall F1-score of 99.6%.

Additionally, we conducted a manual analysis to clarify instances of false detection cases. For ice phishing, the majority of FPs resulted from victims approving transactions to themselves and invoking the `transferFrom` function. This rare behavior closely mimicked phishing activities and could not be distinguished by our detection approach. FPs related to the payable function were attributed to specialized Miner Extractable Value (MEV) bots that employed payable functions without logical functionalities. These MEV bots had off-chain information beyond our knowledge, leading to FP occurrences.

Regarding FNs, most were observed in ice phishing and NFT orders. In ice phishing, FNs resulted from scammers leveraging decentralized exchanges (DEX) to convert victims' funds into alternative tokens, with the phishing address as the recipient. The complex contract semantics of these swaps disrupted the flow of funds, leading to FNs. In NFT orders, FNs occurred because some scammers used extremely low prices (*e.g.*, 1 wei) to perform free order tricks instead of 0 value, resulting in detection failures.

TABLE V: Efficiency evaluation of the detection approach. T/B means the time consumption per block.

Ethereum	Detection Approach		
Ave.T/B	Ave. T/B	Median T/B	Max T/B
12,000 ms	390 ms	362 ms	3,553 ms

These special cases will be discussed further in Section VI.

2) Efficiency Evaluation: To evaluate the efficiency of our detection approach, we use real Ethereum blocks to calculate time consumption. In Ethereum, the fundamental unit of packaging is the block, which contains multiple transactions. The average block production time in Ethereum is 12 seconds (12,000 ms). As shown in TableV, our approach exhibits high efficiency, with an average time consumption of just 390 ms per block, a median of 362 ms per block, and a maximum of 3,553 ms.

For a more detailed view, we present a time consumption graph in Figure 8 in the appendix. Our approach consistently consumes significantly less time than the block production time, even for blocks with the maximum time (which are rare, making the average time consumption a more reliable metric). Therefore, our approach meets the requirements for real-time performance and has been integrated into Forta, a well-known real-time anti-phishing platform (detailed in Section VI).

V. Large scale detection in the real world

Given the demonstrated effectiveness of the proposed detection approach in the previous section, we can now apply this approach to detect real-world threats. Specifically, we conduct the detection on the Ethereum blockchain, covering the period from block *16,304,348* to block *18,440,040*. This corresponds to a timeframe of 300 days, spanning from December 31, 2022, to October 27, 2023. During this period, our detection approach identifies a total of **130,637** PTXPHISH transactions, as detailed in Table VI.

Building upon the detection results, we proceed to perform a comprehensive analysis in various aspects. In Section V-A, we delve into an analysis of the PTXPHISH transactions themselves. Section V-B focuses on examining the characteristics and behaviors of PTXPHISH scammers, while Section V-C explores the experiences and impact on victims of such scams. Lastly, in Section V-D, we present the valuable action we have provided to help combat and mitigate the risks posed by these real-world threats.

A. *Analyzing* PTXPHISH *Transactions*

To analyze the PTXPHISH transactions, we present our analysis from multiple perspectives. First, We examine the economic losses caused by phishing and their relationship with time changes. Secondly, we analyze the characteristics and performance of different phishing categories.

The economic losses caused by PTXPHISH. Considering the diversity of asset types and price fluctuations, it is important

TABLE VI: Detected PTXPHISH and losses in 300 days.

Phishing Category	Number (#)	Loss ($)	Average ($)
Ice phishing	47,762	201,880,314	4,226.8
NFT order	14,999	57,495,168	3,833.3
Address poisoning	1,050	64,042,825	60,993.2
Payable Function	66,826	18,527,500	277.3
Total	**130,637**	**341,945,807**	**2,617.5**

to explain the principles for calculating losses. To ensure that our calculations are as realistic as possible, the prices of all ERC-20 tokens and NFTs are chosen as the price when the phishing transaction occurs. Specifically, the price of ERC20 is determined by the price oracle [15]. To NFTs, there is currently no way to determine the price of a specific NFT, we use the floor price marked on OpenSea instead.

We summarize the detailed phishing transactions and corresponding losses based on their phishing category in Table VI. In total, PTXPHISH caused a total loss of $341,945,807 during 300 days. Among them, ice phishing has the highest proportion, accounting for $201,880,314 (59.04%). Address poisoning is the second highest, with a total profit of $64,042,825 accounting for 18.73%. Market order scams generate a total profit of $57,495,168, accounting for 16.81%. Finally, payable function scams generate $18,527,500, accounting for 5.42%. Interestingly, when we calculate the average losses, we observe variations in the profit strategies employed by phishing scams. For example, the number of address poisoning scams is relatively small (only 1,050 cases), yet they yield a significant individual loss of $60,993 per transaction. In contrast, payable function scams have the highest occurrence rate (66,826 cases), but the individual transaction loss is only $273.

We conclude the graph of PTXPHISH by data and corresponding losses in Figure 5, from which we can see that PTXPHISH has existed for a long time since early 2023, without being effectively solved, and as time goes by, the losses are still increasing. It can be seen that phishing is an increasingly and continuously serious social problem, which further highlights the value of our work. Especially, from March 22 to 24, the losses amount reached over $30 million. After investigating the dates of these extreme cases, we find that Arbitrum airdrops [48] occurred on March 23, 2023. Unfortunately, such campaigns often result in great phishing success. In the later stage, we find two extremely high losses, *i.e.*, $20M from the address poisoning attack suffered by Binance and $2.4M losses from the ice phishing of victim 0x13e382dfe53207E9ce2eeEab330F69da2794179E. To examine the evolution and emergence process of elaborate scams, we conducted a separate study on the active periods of various scams in the early months of 2023, as illustrated in Figure 9 in the appendix. From the active period of different phishing sub-categories, it is evident that these phishing methods are constantly evolving and improving. For instance, zero value transfer poisoning was already been active on December 25, 2022, as an early phishing method. However, with the emergence of new variants of address poisoning scams, the first

[15]In this study, we only focus on Top tokens, *i.e.*, ETH, USDT, USDC, DAI, WETH, stETH, WBTC, BUSD.

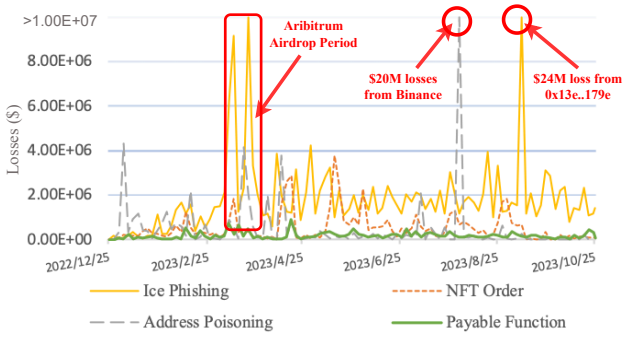

Fig. 5: Variation of PTXPHISH losses over time.

successful phishing transaction of dust poisoning appeared on March 7, 2023, while the first successful fake token poisoning appeared on March 16, 2023. The time interval shows that scammers continue to innovate fishing methods.

Finding #2: PTXPHISH has become a threatening cybercrime, yielding profits exceeding $341.9 million during a 300-day observation period. To make the most profit, PTXPHISH employ different strategies. Payable function scams are numerous with small profits per transaction. In contrast, address poisoning scams are fewer in number but can generate significant profits in a single instance.

The characteristics of each PTXPHISH category. To better understand the characteristics of PTXPHISH tricks, we delve into each trick respectively.

- **Ice Phishing Scam.** To better understand the prevalence of ice phishing scams, we conduct an analysis of the total number of `approve` and `permit` transactions during the same period. Our findings reveal that, out of the token contracts we examined, there are a total of 4,207,423 successful `approve` transactions. Among these, 209,318 transactions are identified as phishing approves, accounting for 4.97% of the total number. Even more concerning, we discover that out of the 13,877 successful `permit` transactions, 6,414 transactions are identified as phishing permits, accounting for a staggering 46.22% of the total number. These alarming numbers show that the `approve` and `permit` functions are abused by phishing scams. Based on our speculation, these functions are favored by phishers due to their hidden and efficient ability to transfer funds ownership.
- **NFT Order Scam.** We analyze the movement of stolen NFT assets. In total, there are 61,838 stolen NFTs, of which 16,442 NFTs have been transferred after being stolen (only 26.6%) until November 30, 2023. It indicates the poor liquidation of NFT assets. In addition, we tracked the transfer events of these NFTs and identified the NFT markets in which these NFTs are sold. Finally, the movements of stolen NFTs are summarized in Table XI in the appendix. From the table, we find that most scammers (62.22%) directly sell the NFTs to the market using the cashier address, while a small portion (17.85%) transfers NFTs to fund aggregators for selling. Among the stolen NFTs, we observed that most of the NFTs were sold through Blur (61.78%), followed by OpenSea (21.97%), X2Y2 (8.32%), and LooksRare (7.83%). In summary, we conclude that these NFT marketplaces do

not effectively prevent the sale of stolen NFTs, and over 80% of stolen NFTs are sold through the markets.

- **Address Poison Scam.** To poison the victims' transaction history, scammers will actively initiate attack transactions. During the observation phase, we discovered a total of 888,744 address poisoning attack transactions, resulting in a total of 3,132,607 addresses being affected. This long-term and extensive scam method poses a significant threat to the security of all addresses.
  The scammer needs to pay the gas fee for their attack transactions. We calculate and find that the gas fee consumed by the scammers was 4023.3 ETH over a period of 300 days (13.4 ETH daily). Additionally, $60,509 tokens were used for dust transfers. According to Etherscan [49], the daily gas consumption is around 107.5 ETH, which means that the gas fee consumed by address poisoning attack transactions accounts for 12.5% of all gas fees on the entire Ethereum.
- **Payable Function Scam.** We conduct an analysis of various functions used in payable function scams to determine their respective proportions (see Table X in the appendix). The total loss resulting from these scams exceeds $18 million. We observe two distinct types based on their functionalities: *Airdrop* accounted for 74.2% of the total losses (e.g., `Claim`/`claim`), while *Wallet* accounted for 25.8% (e.g., `SecurityUpdate`). These findings indicate that victims of this specific phishing attack are primarily motivated by greed, as they aim to profit from potential gains associated with accepting airdrops. Unfortunately, their funds are ultimately stolen through deceptive profit-generating mechanisms employed by scammers. It is crucial to note that a minority of victims lack a fundamental understanding of blockchain technology and mistakenly perceive these interactions as standard wallet operations. As a result, they unknowingly make payments and become prey to these phishing scams.

Finding #3: PTXPHISH is extremely rampant and has impacted ecosystem of Ethereum. For example, 4.97% `approve` transactions and 46.22% `permit` transactions are identified as phishing transactions. Scammers consume about 4023.3 ETH as transaction fees (13.4 ETH daily) to spread the address poisoning scams, which account for 12.5% of the total Ethereum gas fees.

B. Analyzing PTXPHISH *Scammer*

In this section, we analyze the PTXPHISH scammer and focus on their fund flow during the cash-out process, *i.e.*, the money transfer pattern and scammer address organization. After reviewing scams that occurred over six months, we find a special money cash-out pattern, and categorize the behavior of scammer addresses into the following three types:

- *Cashiers.* The Cashier addresses are responsible for directly obtaining funds from victims.
- *Fund Aggregators.* The fund aggregator addresses are responsible for aggregating the profit funds from multiple cashier addresses [16]. The fund aggregators may also be involved with multiple DeFi protocols, such as token swaps in decentralized exchanges (DEXes).

[16]During our analysis, we found that fund aggregators always receive funds from more than 3 cashier addresses.

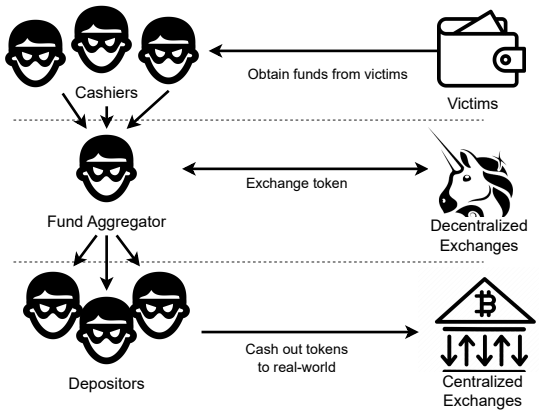

Fig. 6: Scammer organization during the cash-out process.

- *Depositors.* The depositor addresses are responsible for depositing on-chain assets to centralized exchanges (CEXes).

We illustrate the money cash-out pattern and the address organization in Figure 6. First, the cashier addresses obtain funds from victims. Then, multiple cashiers transfer their funds to the fund aggregator. The fund aggregator may exchange the tokens into fiat currencies (*e.g.*, USDT, USDC) or Ether. Finally, the fund aggregator transfers the funds to multiple depositor addresses, which cash out the profits by CEXes. Additionally, to escape the regulation from CEX and security companies, the fund aggregator addresses will change occasionally, resulting in a cashier address transferring money to different fund aggregators. Due to the complex DeFi semantics and large transaction volume (over 3 billion transfers until June, 2023), the money flow graphs (MFG) of blockchain are complex and over-weight for analysis [50], [51]. However, based on the cash-out pattern, we propose a lightweight organization discovery algorithm based on their fund flow relationships and scammer roles, and show the algorithm process as follows.

- *Step S1: Locate the Cashier.* First, we collected a set of cashier addresses from PTXPHISH transactions that were directly exposed and identified as recipients of stolen funds.
- *Step S2: outgoing Transfer Expansion.* We trace the outgoing fund transfer of the cashier addresses and record the destination addresses. To make the outgoing fund transfer more reliable, we analyze several famous DEXes (*e.g.*, Uniswap, Sushiswap), and remove redundant edges caused by DEX interaction. What's more, we prune transfers with a small value (less than $100).
- *Step S3: Expansion Address Categorization.* After getting the outgoing transfer destination addresses, we further categorize these addresses through behavior features: (i) if the destination address is in the CEX whitelist (the CEX whitelist is collected from Etherscan), the address is labeled as the CEX address. (ii) if there are over 3 cashiers with the same outgoing destination address, we label the destination address as the fund aggregator address; (iii) if the address does not fall into either of the above categories, we label it as an unknown address.
- *Step S4: Repeat Expansion & Categorization.* For the remaining unknown addresses, we further trace their outgoing transfers like step S2. And perform address categorizations like step S3. In this study, we repeat 3 times in total.

TABLE VII: Top 5 scammer organizations and profits.

Rank	Famous Address	Total Profits ($)	Percentage
#1	-	60,149,219	17.6%
#2	Fake_Phishing186943 Fake_Phishing186944 [52]	31,628,798	9.25%
#3	Fake_Phishing179050 (Sha zhu pan [53])	17,854,190	5.2%
#4	VenomDrainer [13]	16,351,857	4.8%
#5	InfernoDrainer [54] AngelDrainer [55]	13,196,846	3.9%

We show our algorithm process in Figure 11 in the appendix due to the page limit. In total, from the detected PTXPHISH transaction, we identified 121 scammer organizations with the same fund aggregators. We show the top 5 scammer organization in Table VII. From the table, we can observe that among the highest-ranked organizations, there are several well-known scam addresses (*i.e.*, `Fake_Phishing186944` [52], `Fake_Phishing179050` [53]) and scam drainers (*i.e.*, VenomDrainer [13], InfernoDrainer [54], and AngelDrainer [55]) exposed by the media. These top organizations account for 40.7% of all phishing scam revenue, making them a serious problem that needs to be addressed.

The findings from our cash-out pattern analysis indicate that our proposed scam organization is widely adopted within the current landscape of on-chain scam organizations. Nevertheless, our proposed pattern has certain limitations when it comes to centralized services such as underground money laundering service, which will lead to some false correlations. However, our proposed cash-out pattern can serve as an inspiration for future research endeavors that aim to uncover more fraudulent addresses by exploiting address correlations.

Finding #4: Phishing addresses are highly organized during the cash-out process, with different roles such as cashier, fund aggregator, and depositor. Based on the cash-out pattern, we find that the top five phishing organizations account for 40.7% PTXPHISH losses.

C. Analyzing PTXPHISH Victims

In this section, we analyze the phishing victims. Specifically, we conduct research on the victim's behavior profile and remedial measures after being phished.

The victim behavior profile. Aim to identify user characteristics that are vulnerable to phishing scams. We collected victim addresses from all phishing transactions and recorded the transactions actively initiated by these addresses. To better demonstrate the behavior of victim addresses, we present two dimensions in Figure 10 in the appendix, *i.e.*, the victims' transaction volumes and corresponding losses, and the proportion of transaction types. From the analysis of the figure, it is evident that the majority of victims have fewer than 1,000 transactions. Interestingly, in incidents involving large amounts (over $100k), victim transactions are predominantly concentrated at less than 50. This statistic implies that experienced users with higher transaction amounts exhibit a greater awareness of phishing prevention. Furthermore, our findings reveal that 99% of the victims had been engaged in DeFi activities, with 20% of them specifically involved in NFT transactions. In contrast, only 1% of the victims were found to be engaged in simple Ethereum transfers. This data further solidifies the notion that this new phishing technique predominantly targets DeFi users.

TABLE VIII: Overview of community contributed phishing addresses.

Label Source	**Our Reports**	Blockmage [56]	Tayvano [57]
Number (#/%)	**1726 (42.7%)**	559 (13.8%)	530 (13.1%)
Label Source	AnciliaInc [58]	ZachXBT [59]	Others
Number (#/%)	499 (12.3%)	416 (10.3%)	314 (7.8%)

The victim remedial measure. We mainly focus on the victims of ice phishing, as this type of fraud has ongoing harm until the victim uses the `revoke` function to cancel the phishing approval. According to our observations, after being ice phished, victims mainly exhibit the following three behaviors: *(i)* revoking the phishing approval; (*(ii)*) transferring all assets to other addresses and abandoning the victim address; *(iii)* taking no remedial measures. Out of the randomly selected 5,000 victims (in table XIII in the appendix), only 1,316 addresses (26.32%) chose to revoke the phishing approval, while 1,665 addresses (33.3%) transferred all funds to other addresses, abandoning the previous address. However, a concerning 2,019 addresses (40.38%) did not take any remedial measures, leaving them vulnerable to further attacks and potential financial loss. This indicates that many victims have no idea how to take remedial measures. The vast majority of victims (73.68%) did not take the most effective measure of revoking the phishing approval, but instead chose to transfer funds, which is more time-consuming and expensive.

Finding #5: The majority of victims (99%) are actively involved in DeFi, including NFT transactions. However, a significant portion of these victims (40.38%) lack awareness of the necessary steps to take for implementing remedial measures after experiencing a phishing attack.

D. Contributing to the Community

To further assist users in mitigating threats, we actively contribute to the community by submitting identified phishing addresses to *Etherscan*, which is the largest and de-facto standard blockchain explorer on Ethereum. It offers a *nametag* mechanism that allows trustworthy third parties to label various types of addresses. This practice is widely adopted by the community, including security companies and community sleuths, to combat phishing scams. During the period from December 31, 2022, to October 27, 2023, we contributed a total of **1,726** phishing addresses. Among all the community contributors, our phishing address labels [17] accounted for **42.7%** of the total, as shown in Table VIII.

In addition to providing phishing address labels to the community, we have made other efforts to assist users. Firstly, we proactively send on-chain messages directly to victims to alert

[17]Etherscan only records the label source of the first submission.

them about phishing attempts. Our process involves monitoring the Ethereum pending pool for any suspicious transactions. Upon identifying a phishing transaction in the pending pool, we promptly send a transaction to the victim containing alert information. By receiving our alert transactions, victims are empowered to take proactive measures and prevent phishing losses. During the specified period, we have successfully sent a total of **2,539** on-chain alert messages, providing assistance to **1,980** victims. Additionally, we contribute to anti-phishing efforts by providing phishing reports as online educational resources. These reports have been visited by a significant number of users, with a total visit count of **18,585** based on our internal records for that period. This effectively raises awareness and promotes anti-phishing initiatives.

As a result of our efforts, we have received expressions of gratitude in the form of on-chain transactions and tweets. We take pride in the acknowledgment and appreciation we have received from Etherscan and other members of the community. Their recognition validates our commitment to combatting phishing attempts and protecting individuals from these threats.

VI. Discussion

Our study performs the first empirical study of PTXPhish. Although our focus is primarily on Ethereum, our approach can be easily applied to other EVM-compatible blockchains (*e.g.*, BNB smart chain and Polygon Mainnet). In the following, we will discuss details related to the anatomy, corner cases, and anti-phishing tools/platforms.

Anatomy of PTXPhish. In this study, we categorize the current phishing scams into four categories. However, as discussed in Section V-A, scammers are continuously developing new methods. Therefore, the categorization presented in this study reflects the current state of phishing techniques. Future advancements in phishing methods may necessitate adjustments to this categorization.

Corner Case of Detection. In some extreme theoretical scenarios, our detection approach may produce inaccurate results, such as self-approvals or closed-source MEV bot (see Section IV-C). Other potential corner cases might include situations where a drainer executes a `transferFrom` but leaves some funds with the victim.

These cases are counter-intuitive, as we assume that all on-chain behaviors are driven by rational actors seeking to maximize their benefits. However, behaviors like self-approvals or leaving funds behind lead to unnecessary losses or wasted gas fees, making them relatively rare. Consequently, while our detection approach may not cover all extreme theoretical cases, it remains suitable and effective for real-world applications.

Anti-Phishing Tools/Platforms. Many security companies have developed anti-phishing tools/platforms to combat the prevalence of phishing scams. We list prominent anti-phishing tools/platforms in Table XII in the appendix, and categorize them based on their approaches. Current anti-phishing tools (*e.g.*, AegisWeb3, Pocket Universe) primarily use transaction pre-execution to predict fund changes and implement blacklists for receiving address detection. In contrast, our detection approach adopts a rule-based strategy based on on-chain information. This unique approach complements existing tools and enhances their security coverage. Indeed, **our detection approach has been integrated into Forta**, a leading scam detection platform, establishing us as a primary partner.

VII. Related Work

A. Security Issues on Ethereum

Since its inception, Ethereum has faced numerous security issues. The security issues have evolved with the development of the platform. The academic community has shown great concern for the security of Ethereum, with many research [60], [61], [12] efforts dedicated to addressing its security challenges. Xia et al. [8] perform the first analysis on the fake ERC-20 tokens, and leverage AI to perform fake token detection. Chen et al. [62] conduct analysis on smart contracts and propose a method to find the security issues by comparing historical versions. Liu et al. [63] focus on the permission bugs in the DeFi project, and propose a prototype detection system. Su et al. [64] measure the DeFi attacks and propose a detection algorithm. Das et al. [42] perform an in-depth analysis of the NFT ecosystem, and raise several security issues.

B. Phishing Analysis

Research into analyzing phishing behaviors have been evolving for years. For traditional Web2 phishing, several studies [3], [65], [66], [5] have analyzed phishing behaviors and characteristics. Web3 phishing, while similar to traditional Web2 phishing, extends beyond websites and leverages cryptocurrency as a payment method [7]. He et al. [10] and Li et al. [7] have proposed website-based phishing detection systems and conducted analyses of phishing websites.

In addition to traditional phishing scams, Ivanov et al. [67] were the first to highlight scams exploiting misleading EVM features, *i.e.*, address manipulation and Unicode attacks. Ye et al. [9] focused on phishing that involves misleading information on the wallet UI (including token symbols, wallet addresses, and smart contract function names), though their study was limited to zero-value transfers and fake `claim` functions. Kim et al. [68] focused on NFT scams and developed a detection model using features like price differences, time duration, and transfer relations. Li et al. [69] collected illicit addresses from the Blockchain Intelligence Group and employed machine learning techniques to predict these addresses.

Our study distinguishes itself from related research in the following aspects: *(i) Different target & motivation.* To our knowledge, our study is the first to provide a comprehensive analysis of PTXPhish. We aim to thoroughly investigate this new form of phishing, which may include subclasses of previous phishing tactics such as "setApproveForAll" in NFT phishing and "zero value transfer" in address poisoning. *(ii) Different detection method & capability.* Previous research primarily relies on past fund flows, which may overlook/delay the detection of newly created phishing addresses. In contrast, our rule-based detection method allows for **real-time** identification of phishing transactions and addresses.

VIII. Conclusion

This paper presents the first comprehensive study of PTXPhish on the Ethereum. First, we conducted a long-term data

collection to establish the first ground-truth PTXPHISH dataset consisting of 5,000 phishing transactions. Then we dissected PTXPHISH, categorizing phishing tactics into four primary categories and eleven sub-categories. Second, we proposed a rule-based multi-dimensional detection approach to identify phishing transactions, achieving over 99% F1-score. Finally, we conducted an in-depth analysis of the large-scale detection results to offer insightful findings. Our analysis revealed that PTXPHISH resulted in losses exceeding $341.9 million within a 300-day period. Scammers expended approximately 13.4 ETH daily, which accounted for 12.5% of the total Ethereum gas fees, in spreading address poisoning scams. Notably, the top five phishing organizations were responsible for 40.7% of the total losses. Furthermore, our work made significant contributions to the community. We reported a total of 1,726 phishing addresses, accounting for 42.7% of the total community contributions during the same period. Additionally, we sent 2,539 on-chain alert messages, providing assistance to 1,980 victims of phishing attacks.

ACKNOWLEDGMENT

We thank all anonymous reviewers for their invaluable comments. This work is partially supported by the National Key R&D Program of China (No. 2022YFE0113200), the National Natural Science Foundation of China (NSFC) under Grant 62172360, U21A20464, and U21A20467. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of funding agencies.

REFERENCES

[1] H. Bijmans, T. Booij, A. Schwedersky, A. Nedgabat, and R. van Wegberg, "Catching phishers by their bait: Investigating the dutch phishing landscape through phishing kit detection," in *30th USENIX Security Symposium (USENIX Security 21)*, 2021, pp. 3757–3774.

[2] T. Kim, N. Park, J. Hong, and S.-W. Kim, "Phishing url detection: A network-based approach robust to evasion," in *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security*, 2022, pp. 1769–1782.

[3] A. Oest, P. Zhang, B. Wardman, E. Nunes, J. Burgis, A. Zand, K. Thomas, A. Doupé, and G.-J. Ahn, "Sunrise to sunset: Analyzing the end-to-end life cycle and effectiveness of phishing attacks at scale," in *29th {USENIX} Security Symposium ({USENIX} Security 20)*, 2020.

[4] A. Oest, Y. Safei, A. Doupé, G.-J. Ahn, B. Wardman, and G. Warner, "Inside a phisher's mind: Understanding the anti-phishing ecosystem through phishing kit analysis," in *2018 APWG Symposium on Electronic Crime Research (eCrime)*. IEEE, 2018, pp. 1–12.

[5] A. Oest, Y. Safaei, P. Zhang, B. Wardman, K. Tyers, Y. Shoshitaishvili, and A. Doupé, "{PhishTime}: Continuous longitudinal measurement of the effectiveness of anti-phishing blacklists," in *29th USENIX Security Symposium (USENIX Security 20)*, 2020, pp. 379–396.

[6] "A brazen online attack targets v.i.p. twitter users in a bitcoin scam," Feb. 2022. [Online]. Available: https://www.nytimes.com/2020/07/15/technology/twitter-hack-bill-gates-elon-musk.html

[7] X. Li, A. Yepuri, and N. Nikiforakis, "Double and nothing: Understanding and detecting cryptocurrency giveaway scams," in *Network and Distributed Systems Security (NDSS) Symposium*, 2023.

[8] P. Xia, H. Wang, B. Gao, W. Su, Z. Yu, X. Luo, C. Zhang, X. Xiao, and G. Xu, "Trade or trick? detecting and characterizing scam tokens on uniswap decentralized exchange," *Proceedings of the ACM on Measurement and Analysis of Computing Systems*, vol. 5, no. 3, pp. 1–26, 2021.

[9] G. Ye, M. Wu, G. Hong, and M. Yang, "Revealing and analyzing the visual scams of cryptocurrency wallets," in *Proceedings of the ACM Turing Award Celebration Conference-China 2023*, 2023, pp. 148–149.

[10] B. He, Y. Chen, Z. Chen, X. Hu, Y. Hu, L. Wu, R. Chang, H. Wang, and Y. Zhou, "Txphishscope: Towards detecting and understanding transaction-based phishing on ethereum," in *Proceedings of the 2023 ACM SIGSAC Conference on Computer and Communications Security*, 2023, pp. 120–134.

[11] W. Chen, X. Guo, Z. Chen, Z. Zheng, and Y. Lu, "Phishing scam detection on ethereum: Towards financial security for blockchain ecosystem." in *IJCAI*, vol. 7, 2020, pp. 4456–4462.

[12] S. Li, G. Gou, C. Liu, C. Hou, Z. Li, and G. Xiong, "Ttagn: Temporal transaction aggregation graph network for ethereum phishing scams detection," in *Proceedings of the ACM Web Conference 2022*, 2022, pp. 661–669.

[13] "Venom drainer analysis report," 2023. [Online]. Available: https://twitter.com/realScamSniffer/status/1642813130454765568

[14] "The btc approve phish twitter report," 2023. [Online]. Available: https://twitter.com/MetaSleuth/status/1638812482021228544

[15] "10,000,000 $ scammer analysis report," online, 2023. [Online]. Available: https://twitter.com/MetaSleuth/status/1643901208116224000

[16] "Opensea phishing scam news," online, 2023. [Online]. Available: https://www.lowyat.net/2022/266445/opensea-phishing-attack-loss-of-nfts-641-ethereum/

[17] "approve phishing scam report," 2023. [Online]. Available: https://twitter.com/realScamSniffer/status/1658362378243956736

[18] "approve phishing scam report 2," 2023. [Online]. Available: https://twitter.com/realScamSniffer/status/1655376202335653888

[19] "permit phishing scam report," 2023. [Online]. Available: https://twitter.com/realScamSniffer/status/1662401670649892864

[20] "permit phishing scam report2," 2023. [Online]. Available: https://twitter.com/MetaSleuth/status/1655865135217647616

[21] "dust transfer phishing scam report," 2023. [Online]. Available: https://twitter.com/MetaSleuth/status/1635856097067294722

[22] "empty function phishing scam report," 2023. [Online]. Available: https://twitter.com/realScamSniffer/status/1647780640346243073

[23] "empty function phishing scam report 2," 2023. [Online]. Available: https://twitter.com/realScamSniffer/status/1638832010948128768

[24] "bulktransfer phishing scam report," 2023. [Online]. Available: https://twitter.com/realScamSniffer/status/1644149450481467392

[25] "Claim contract phish analysis report," online, 2023. [Online]. Available: https://twitter.com/realScamSniffer/status/1678985330236506113

[26] "Top blockchain attack losses in 2023," online, 2023. [Online]. Available: https://twitter.com/PeckShieldAlert/status/1706256486991863833?s=20

[27] D. Vuji, D. Jagodic, and S. Randic, "Blockchain technology, bitcoin, and ethereum: A brief overview," in *2018 17th international symposium infoteh-jahorina (infoteh)*. IEEE, 2018, pp. 1–6.

[28] S. Wu, D. Wang, J. He, Y. Zhou, L. Wu, X. Yuan, Q. He, and K. Ren, "Defiranger: Detecting price manipulation attacks on defi applications," *arXiv preprint arXiv:2104.15068*, 2021.

[29] L. Zhou, K. Qin, A. Cully, B. Livshits, and A. Gervais, "On the just-in-time discovery of profit-generating transactions in defi protocols," in *2021 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2021, pp. 919–936.

[30] D. Wang, S. Wu, Z. Lin, L. Wu, X. Yuan, Y. Zhou, H. Wang, and K. Ren, "Towards a first step to understand flash loan and its applications in defi ecosystem," in *Proceedings of the Ninth International Workshop on Security in Blockchain and Cloud Computing*, 2021, pp. 23–28.

[31] S. Werner, D. Perez, L. Gudgeon, A. Klages-Mundt, D. Harz, and W. Knottenbelt, "Sok: Decentralized finance (defi)," in *Proceedings of the 4th ACM Conference on Advances in Financial Technologies*, 2022, pp. 30–46.

[32] "Forta blog homepage," Jan. 2023. [Online]. Available: https://forta.org/blog

[33] "Blocksec blog homepage," Jan. 2023. [Online]. Available: https://blocksec.com/blog

[34] "Peckshiled blog homepage," Jan. 2023. [Online]. Available: https://twitter.com/peckshield

[35] "Metasleuth blog homepage," Jan. 2023. [Online]. Available: https://metasleuth.io/blog

[36] "Scamsniffer blog homepage," Jan. 2023. [Online]. Available: https://drops.scamsniffer.io/

[37] D. T. KOL, "debank kol," Jan. 2023. [Online]. Available: https://debank.com/ranking

[38] difillama, "Defillama," Sep. 2024. [Online]. Available: https://defillama.com/

[39] "Opensea maekrt," 2023. [Online]. Available: https://opensea.io

[40] "Blur maekrt," 2023. [Online]. Available: https://blur.io

[41] W. E. Bodell III, S. Meisami, and Y. Duan, "Proxy hunting: Understanding and characterizing proxy-based upgradeable smart contracts in blockchains," in *32nd USENIX Security Symposium (USENIX Security 23)*, 2023, pp. 1829–1846.

[42] D. Das, P. Bose, N. Ruaro, C. Kruegel, and G. Vigna, "Understanding security issues in the nft ecosystem," in *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security*, 2022, pp. 667–681.

[43] "The huge loss from binance caused by address poison," online, 2023. [Online]. Available: https://twitter.com/tayvano_/status/1686418992599273472

[44] "Airdrop in blockchain," 2023. [Online]. Available: https://ckfinance.medium.com/defi-airdrop-b84c1396d70a

[45] beaconcha.in, "Ethereum 4byte signature database," Sep. 2023. [Online]. Available: https://www.4byte.directory/

[46] H. Feng, Y. Hu, Y. Kou, R. Li, J. Zhu, L. Wu, and Y. Zhou, "{SlimArchive}: A lightweight architecture for ethereum archive nodes," in *2024 USENIX Annual Technical Conference (USENIX ATC 24)*, 2024, pp. 1257–1272.

[47] S. Wu, L. Wu, Y. Zhou, R. Li, Z. Wang, X. Luo, C. Wang, and K. Ren, "Time-travel investigation: Toward building a scalable attack detection framework on ethereum," *ACM Transactions on Software Engineering and Methodology (TOSEM)*, vol. 31, no. 3, pp. 1–33, 2022.

[48] "Arbitrum 3.23 airdrop twitter," Feb. 2023. [Online]. Available: https://twitter.com/ArbinuCoin/status/1636375881051705347

[49] "Etherscan," 2023. [Online]. Available: https://etherscan.io/

[50] H. Kalodner, M. Möser, K. Lee, S. Goldfeder, M. Plattner, A. Chator, and A. Narayanan, "{BlockSci}: Design and applications of a blockchain analysis platform," in *29th USENIX Security Symposium (USENIX Security 20)*, 2020, pp. 2721–2738.

[51] L. Wu, Y. Hu, Y. Zhou, H. Wang, X. Luo, Z. Wang, F. Zhang, and K. Ren, "Towards understanding and demystifying bitcoin mixing services," in *Proceedings of the Web Conference 2021*, 2021, pp. 33–44.

[52] "The largest single transaction phishing scam in 2023," online, 2023. [Online]. Available: https://coinmarketcap.com/community/articles/64fa1ed60eddf61668875586/

[53] "The analysis of shazhupan scam," online, 2023. [Online]. Available: https://www.coinbase.com/blog/security-psa-sha-zhu-pan-pig-butchering-investment-scams

[54] "Inferno drainer analysis report," 2023. [Online]. Available: https://decrypt.co/140877/inferno-drainer-scam-scammer-phishing-crypto-nfts

[55] "Angel drainer," online, 2023. [Online]. Available: https://coinmarketcap.com/community/articles/65ce34e1fbed1659a6837ef5

[56] "Blockmage," 2023. [Online]. Available: https://www.blockmage.dev/home

[57] "Tay," 2023. [Online]. Available: https://twitter.com/tayvano_

[58] "Ancilia," 2023. [Online]. Available: https://www.ancilia.ai/

[59] "Zach," 2023. [Online]. Available: https://twitter.com/zachxbt

[60] W. Chen, Z. Zheng, J. Cui, E. Ngai, P. Zheng, and Y. Zhou, "Detecting ponzi schemes on ethereum: Towards healthier blockchain technology," in *Proceedings of the 2018 world wide web conference*, 2018, pp. 1409–1418.

[61] H. Hu, Q. Bai, and Y. Xu, "Scsguard: Deep scam detection for ethereum smart contracts," in *IEEE INFOCOM 2022-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*. IEEE, 2022, pp. 1–6.

[62] J. Chen, "Finding ethereum smart contracts security issues by comparing history versions," in *Proceedings of the 35th IEEE/ACM International Conference on Automated Software Engineering*, 2020, pp. 1382–1384.

[63] Y. Liu, Y. Li, S.-W. Lin, and C. Artho, "Finding permission bugs in smart contracts with role mining," in *Proceedings of the 31st ACM SIGSOFT International Symposium on Software Testing and Analysis*, 2022, pp. 716–727.

[64] L. Su, X. Shen, X. Du, X. Liao, X. Wang, L. Xing, and B. Liu, "Evil under the sun: understanding and discovering attacks on ethereum decentralized applications," in *30th USENIX Security Symposium (USENIX Security 21)*, 2021, pp. 1307–1324.

[65] D. Lain, K. Kostiainen, and S. Čapkun, "Phishing in organizations: Findings from a large-scale and long-term study," in *2022 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2022, pp. 842–859.

[66] Y. Lin, R. Liu, D. M. Divakaran, J. Y. Ng, Q. Z. Chan, Y. Lu, Y. Si, F. Zhang, and J. S. Dong, "Phishpedia: A hybrid deep learning based approach to visually identify phishing webpages," in *30th USENIX Security Symposium (USENIX Security 21)*, 2021, pp. 3793–3810.

[67] N. Ivanov, J. Lou, T. Chen, J. Li, and Q. Yan, "Targeting the weakest link: Social engineering attacks in ethereum smart contracts," in *Proceedings of the 2021 ACM asia conference on computer and communications security*, 2021, pp. 787–801.

[68] H. Kim, J. Cui, E. Jang, C. Lee, Y. Lee, J.-W. Chung, and S. Shin, "Drainclog: Detecting rogue accounts with illegally-obtained nfts using classifiers learned on graphs," *arXiv preprint arXiv:2301.13577*, 2023.

[69] J. Li, F. Baldimtsi, J. P. Brandao, M. Kugler, R. Hulays, E. Showers, Z. Ali, and J. Chang, "Measuring illicit activity in defi: The case of ethereum," in *Financial Cryptography and Data Security. FC 2021 International Workshops: CoDecFin, DeFi, VOTING, and WTSC, Virtual Event, March 5, 2021, Revised Selected Papers 25*. Springer, 2021, pp. 197–203.

[70] "proxyupdate phishing scam report," 2023. [Online]. Available: https://twitter.com/MetaSleuth/status/1658320175647834113

[71] "Blur.io zero buy scam," 2023. [Online]. Available: https://twitter.com/MetaSleuth/status/1633318417938939905

APPENDIX

The appendix contains charts and figures mentioned in the main text but not displayed due to space constraints.

A. Important ERC-20/ERC-721 interface

```
// ERC-20
approve(address _spender, uint256 _value)
transferFrom(address _from, address _to, uint256 _value)
// ERC-721
approve(address _spender,uint256 _tokenId)
setApprovalForAll(address _operator, bool _approved)
transferFrom(address _from, address _to, uint256 tokenId)
```

Fig. 7: Important ERC-20/ERC-721 interface.

B. Decision on expanding the number of phishing transactions.

Due to the extensive transactions associated with the addresses, manually verifying all historical transactions is impractical. Consequently, we employed a sampling method to obtain historical data. This approach involves a trade-off: a larger sample size significantly increases manual effort, while a smaller sample may result in insufficient coverage.

We analyzed the number of transactions per address and determined the median count to be 43.5. To balance adequate coverage with manageable effort, we chose a threshold of 50 transactions. Detailed information about the addresses is available at: https://github.com/HypoopyH/PTXPhish.

C. Detailed ground-truth dataset of PTXPHISH

We have established the first ground-truth PTXPHISH dataset, consisting of 5,000 phishing transactions. The dataset is categorized into various phishing categories, including 2,569 *ice phishing* transactions, 609 *NFT order* transactions, 226 *address poisoning* transactions, and 1,596 *payable function* transactions. Detailed information can be found in Table IX.

D. Efficiency evaluation of detection approach.

The figure 8 shows our detection approach time consumption, which is mentioned in Section IV-C. The average block production time in Ethereum is 12s (12,000 ms). Our approach is highly efficient, with an average time consumption of only 390 ms per block, a median time consumption of 362 ms per block, and a max time of 3,553 ms.

E. Popular signatures of payable function phishing scams

Table X described in Section V-A, details popular signatures of payable function phishing scams. These scams have resulted in total losses resulting exceeding $18 million. We observe two distinct types based on their functionalities: Airdrop scams account for 74.2% of the total losses (e.g., `Claim/claim`), while Wallet scams account for 25.8% (e.g., `SecurityUpdate`).

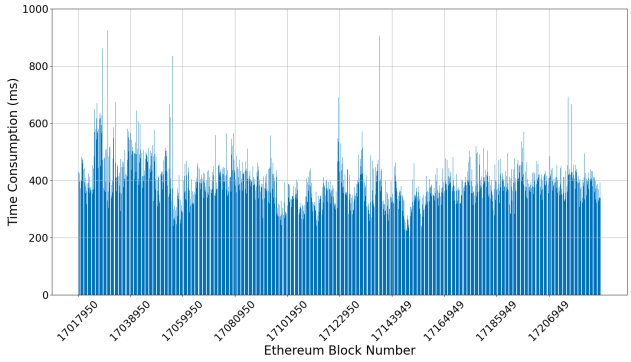

Fig. 8: The efficiency evaluation of detection methods. Due to the extended duration, the time value is the average time consumption calculated for every ten blocks as a group.

F. Heatmap of PTXPHISH by date and corresponding losses in the early stage

Figure 9, described in Section V-A shows the heatmap of PTXPHISH by date and corresponding losses. The data indicates that phishing methods are continually evolving and improving. For example, zero value transfer poisoning emerged as an early phishing method on December 25, 2022. However, new variants of address poisoning scams began to appear, with the first successful dust poisoning transaction on March 7, 2023, and the first successful fake token poisoning on March 16, 2023. This timeline highlights the ongoing innovation in phishing techniques.

G. Stolen NFTs cash-out markets

Table XI, described in Section V-A, presents data on stolen NFTs cash-out markets. The table reveals that the majority of scammers (62.22%) directly sell the NFTs to the market using the cashier address, while a smaller portion (17.85%) transfers NFTs to fund aggregators for selling. Among the stolen NFTs, most were sold through Blur (61.78%), followed by OpenSea (21.97%), X2Y2 (8.32%), and LooksRare (7.83%).

H. Victim behavior profile

Figure 10, mentioned in Section V-C, illustrates the victim behavior profile. The data shows that the majority of victims have conducted fewer than 1,000 transactions. Notably, in incidents involving large amounts (over $100k), the victim transactions are predominantly fewer than 50. This suggests that experienced users, who handle higher transaction amounts, are generally more aware of phishing prevention.

I. Scammer organization discovery algorithm process

Figure 11, described in Section V-B, outlines the scammer organization discovery algorithm process. In step S1, we found 5,350 cashier addresses, of which 1,210 had no outgoing transfers. In step S2, we identified 4,384 outgoing destination addresses with transfer value exceeding 100$. In step S3, we categorized these addresses into 2,307 destination fund aggregators and 260 depositors based on their behavior. In step S4, we repeated the outgoing transfer expansion & categorization process for the remaining 1,817 addresses.

TABLE IX: Detailed ground-truth dataset of PTXPHISH.

Category			Target Assests	Spread Method	Our Findings [1]	Dataset Num
Exploiting legitimate contract	Ice phishing	Approve	ERC20 token	Website	-	1247
		Permit	ERC20 token		-	814
		SetApproveForAll	NFT		-	508
	NFT order	Bulk transfer	NFT		-	37
		Proxy upgrade	NFT		✔ [70]	108
		Free buy order	NFT & ERC20 token		✔ [71] [2]	464
Deploying phishing contract	Address poisoning	Zero value transfer	ERC20 token	Transaction	-	104
		Fake token transfer	ERC20 token		-	100
		Dust value transfer	ERC20 token		✔ [21]	22
	Payable Function	Airdrop function	ETH		-	788
		Wallet function	ETH		-	808
Benign Transaction			-	-	-	13557
Total			-	-	-	18555

[1] We were the first to discover and report the new fishing tricks.
[2] We were the first to discover the free buy order scam targeting the Blur.io market.

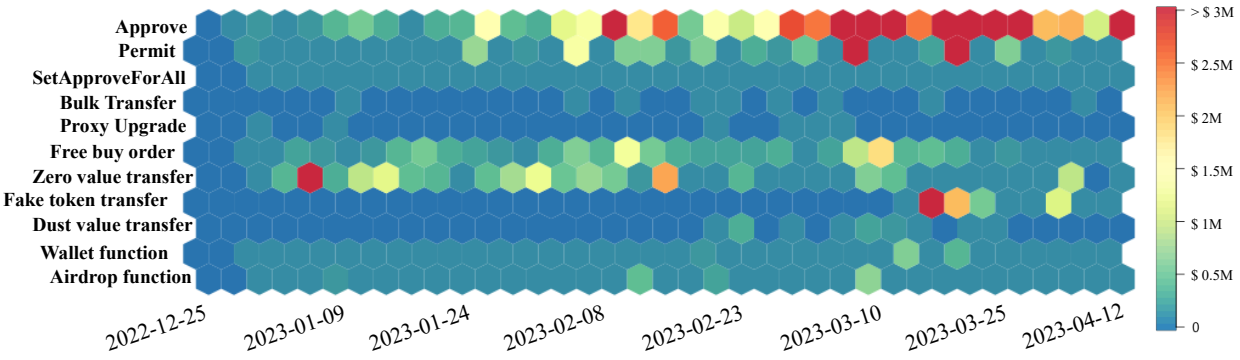

Fig. 9: Heatmap of PTXPHISH by date and corresponding losses in the early stage.

TABLE X: Popular signatures of payable function phishing scams.

Function		Function Signature	Loss ($)
Wallet	SecurityUpdate	0x5fba79f5 0xaf347b61	3,275,537
	ConnectWallet	0x62929a1e	166,302
	NetworkMerge	0x9c9316c5	913,114
	pay	0x1b9265b8	419,270
Airdrop	claim/Claim	0x4e71d92d 0x3158952e 0xaad3ec96 0x0c7ef932	9,717,171
	claimReward	0xb88a802f 0x79372f9a 0xaf7ec6cb 0x63e32091	507,850
	claimRewards	0xef5cfb8c	3,456,764
	receiveETH	0x4185f8eb	71,492
Total		-	18,527,500

TABLE XI: Stolen NFTs cash-out markets.

Seller	Markt				Total
	Blur	Opensea	LooksRare	X2Y2	
Cashier	6,654	1,694	944	939	10,231
Fund aggregator	1,487	1,201	88	158	2,934
Total	8,141	2,895	1,032	1,097	13,165

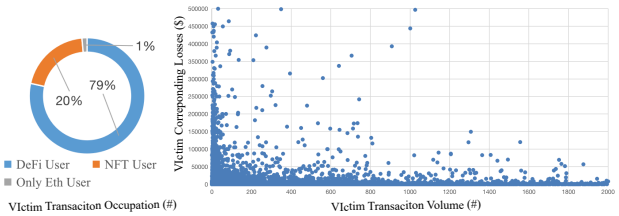

Fig. 10: The victim behavior profile. Figure 1 is the proportion of victims' transaction types, and Figure 2 is the victims' transaction volume and corresponding losses.

J. Details of existing anti-phishing tools/platforms

Table XII, described in Section VI, provides details on existing anti-phishing tools/platforms.

K. Remedial behavior of ice phishing victims

Table XIII, described in Section V-C, details the remedial behavior of ice phishing victims. Among the randomly selected 5,000 victims, only 1,316 addresses (26.32%) chose to revoke the phishing approval, while 1,665 addresses (33.3%) transferred all funds to other addresses, abandoning the compromised address. However, a concerning 2,019 addresses

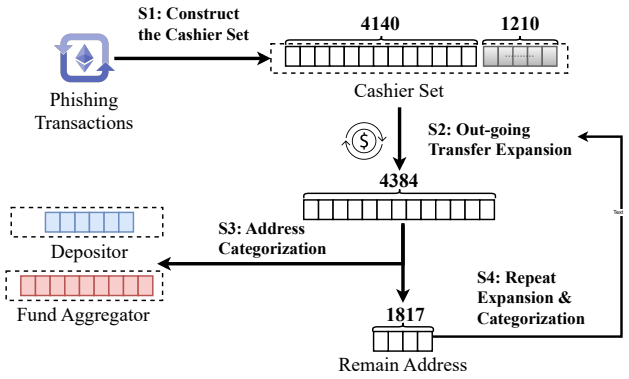

Fig. 11: Scammer organization discovery algorithm process.

TABLE XII: Details of existing anti-phishing tools/platforms. ✔means the tool/platform leverages the feature, ✗means the tool/platform does not.

Category	Tool/Platform	Blacklist Label	Pre-execution	User Number
Website Blocking	AegisWeb3	✔	✗	100,000+
	MetaShield	✔	✗	1,000+
	ScamSniffer	✔	✔	40,000+
Transaction Pre-execution	Pocket Universe	✔	✔	100,000+
	Stelo	✔	✔	9,000+
Transaction Blocking	Forta Scam Bot	✔	✗	- [1]
Remedial Tool	Revoke.cash	✔	✗	60,000+
	MetaSleuth.io	✔	✗	3,500+

[1] "-" Represents that the number of users cannot be known.

TABLE XIII: Remedial behavior of ice phishing victims.

Remedial measure	Number (#)	Proportion (%)
Revoke	1,316	26.32%
Transfer assets	1,665	33.3%
No remedial measure	2,019	40.38%
Total	**5,000**	**100.0%**

(40.38%) did not take any remedial measures, leaving them vulnerable to further attacks and potential financial loss. This suggests that the majority of victims unaware of how to effectively address phishing incidents.