Examining Financially-aware Adversaries

Taro Tsuchiya

February 6, 2026

Software and Societal Systems Department
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213

Thesis Committee:
Professor Nicolas Christin, Carnegie Mellon University (Chair)
Professor Giulia Fanti, Carnegie Mellon University
Professor Ariel Zetlin-Jones, Carnegie Mellon University
Professor Arthur Gervais, University College London

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy.

Abstract

Traditional banking systems are heavily regulated, requiring days to pass an identification check or process transactions. With technological advancement (e.g., blockchain, trading apps), users can trade pseudonymously or copy others' trading strategies online. Users can even perform tasks traditionally reserved for governments or banks: minting financial assets or verifying transactions themselves. This change, so-called "democratization" in finance, offers new opportunities for users, but also introduces a new attack surface. This proposal will introduce "financially-aware adversaries"—the computer security attackers operating within online financial systems. I will postulate that those adversaries primarily repurpose traditional computer security tactics (e.g., DoS, domain spoofing, Sybil), but incorporate financial information (e.g., asset prices, transaction history) that is typically irrelevant or unavailable in traditional computer security. In this proposal, I will present three completed works and one proposed study that identify such attacks, quantify their damage at scale, and analyze attack strategies. The first study formalized a novel denial-of-service (DoS) attack on blockchain peer-to-peer (P2P) networks, which exploits nodes' financial incentive to reduce transaction validation latency. The second study examined a new phishing scheme on blockchain wallets, "address poisoning," where attackers spoof the victim's recipient to misdirect the victim's assets to the attacker. The third study identified malicious financial traders online who post spam and manipulate reputation through Sybil (i.e., fake) accounts. In the proposed study, I will investigate various attacks that manipulate the trading signals on meme coin markets. These findings will better inform defenses and moderation efforts to combat such adversaries.

Chapter 1

Introduction

About a decade ago, financial services were simple for end users. Large financial institutions (e.g., banks, brokerages, governments) intermediated most of the financial activities, while users had little control over their assets: depositing money in a bank and occasionally trading stocks through certified brokers. Large institutions also typically handled security threats (e.g., denial of service, front-running, fraud/money laundering), leaving users little incentive or ability to protect themselves.

Recent technological advancements have changed the dynamics between users and financial institutions. Most notably, blockchain technology, empowered by many computer science primitives, has enabled users to access financial services without identity information (i.e., pseudonymously), send assets promptly across borders, and create their financial applications through smart contracts, a programmable code on blockchains. Moreover, the introduction of social media platforms and mobile trading applications has lowered the barrier to trading assets on their own. For instance, users can find trending assets on social media, copy financial influencers' trading strategies, and create new tokens with a few clicks on their mobile phones. Those changes have increased the accessibility of financial products and expanded capabilities for end users, typically characterized as "democratization" in finance [12, 13, 39**?**]. However, this transformation simultaneously brings new challenges from computer security domains.

I will postulate the thesis statement below.

> This thesis introduces "financially-aware adversaries"—the computer security attackers operating within online financial systems. Through large-scale measurements and simulations, I demonstrate that those adversaries are pervasive and inflict significant costs on society. Moreover, I illustrate that these adversaries combine traditional computer security tactics with financial information—from macro-level market conditions to micro-level user trading data—that is typically irrelevant or unavailable in traditional computer security literature. These findings will better inform defense and moderation efforts to combat such adversaries.

To verify this statement, I will present three completed and one ongoing study that investigate financially-aware adversaries. In particular, I perform large-scale measurements or simulations to identify such attacks (or quantify potential damage), analyze their strategies, and inform better defense and moderation.

Blockchain P2P networks (§2.1, completed) On blockchains, users can run a node in a peer-to-

peer (P2P) network and validate transactions. I formalize a new DoS (denial-of-service) attack targeting a P2P node—a "blockchain amplification attack." Given that the financial market provides value for low latency (e.g., arbitrage), some nodes skip transaction validations to save a few milliseconds of processing time. This behavior makes them vulnerable to accepting a flood of invalid transactions. My work quantifies the damage of this attack through mathematical modeling, topology inference (from 2.5 billion data points), and local network simulation.

Self-custody wallets (§2.2, completed) On blockchains, users can set up a self-custody wallet to have full control over their assets, eliminating risks from intermediaries. However, they face usability challenges—dealing with non-human-readable wallet addresses. My work examines a new phishing scheme, "address poisoning." Attackers generate a similar-looking address to the victim's recipient, send a tiny amount to get it added to the victim's history, and hope the victim sends their assets by mistake. I implement a detection algorithm to scan two years of Ethereum and Binance Smart Chain (BSC) and identify over 270 million attack attempts and approximately 83 million USD losses. I also uncover large attacker groups and find some groups that disproportionately target rich and active accounts.

Financial social media platforms (§2.3, completed) Users can browse websites or social media for information, such as trending assets or influencer picks, but are also increasingly exposed to harmful content online. I collect ten years of financial social media data (e.g., TradingView) and reveal that some accounts frequently distribute spam, post toxic comments, and manipulate their reputation through fake accounts (i.e., Sybils). While those accounts exhibit similar characteristics to those in traditional cybercrime (e.g., a closely-knit community), the level of misbehavior increases for certain assets when the market faces turmoil.

Meme coin marketplaces (Ch. 3, proposed) Anyone can easily mint new financial assets through smart contracts, which, however, makes them vulnerable to market manipulation. My ongoing work will study the meme coin marketplaces on Solana and provide evidence of how attackers create multiple blockchain wallets (e.g., Sybils), and trade among them to manipulate prices for profits. In the past, most manipulation had happened in the centralized exchanges, which were typically closed-source. It will be the first to investigate trade market manipulation tactics at scale using account-level trading data.

To connect to the thesis statement above, those four studies aim to show that those attackers perform computer security attacks (e.g., DoS, address poisoning, Sybil accounts), but they tailor their strategies based on financial information. The attackers incorporate (or are influenced by) macro-level market information, such as asset type, price, liquidity, or micro-level account information, such as victims' transaction history and balances. That information is typically considered irrelevant or unavailable in computer security. These findings will inform users, platforms, and regulators to better design their defense and moderation strategies. For instance, wallet providers can alert users, especially when users transact with high-volume, or social media platforms can primarily moderate content that is associated with highly volatile assets.

For the completed works in Ch. 2, I will only highlight the key analysis results that directly support my thesis statement. Ch. 3 will describe the proposed work and provide the timeline for

the remaining work.

1.1 Thesis scope

This section defines some important terms to better position my thesis. First, "democratization in finance" is not a new concept. The term was often used interchangeably with the term "financialization," and introduced in the 1990s with the premise to increase accessibility in financial services and promote financial literacy for all households [12]. There exist several definitions and interpretations. Shiller [40] defines as reducing economic inequality that cannot be explained by effort and talent, and famously quotes as "bring the advantages enjoyed by the clients of Wall Street to the customers of Wal-Mart." Fink [13] recently defines democratization as increasing the accessibility of financial products and bringing new users into the financial system, and highlights the importance of tokenization. This thesis focuses on the role of new digital technologies, namely blockchain, social media, and mobile trading apps, which give users new capabilities that are traditionally reserved for financial institutions. Notably, each study corresponds to the user's new capability and the corresponding system that enables it: 1) validating transactions (on the blockchain P2P network), 2) having full control over the assets (via blockchain self-custody wallets), 3) creating trading strategies (on financial social media platforms), and 4) minting new assets (through smart contracts and meme coin launchpads).

Next, I will define traditional computer security threats in order to better contextualize financially-aware adversaries. Following the Commission of the European Communication [35], I refer to these threats as one of three categories: 1) online fraud or forgery (e.g., spoofing), 2) the publication of illegal contents (e.g., hate speech, non-consensual image sharing), and 3) crimes unique to electronic networks (e.g., DoS, hacking). While Soska [41] argues that the defenders can consider the economic rationality of attackers, this thesis will explore financial information as an additional information source to develop a better defense.

Regarding the societal "cost" from these attacks, my focus is primarily on direct costs [3, 4]. Direct costs include victims' financial losses (e.g., payment for attackers) or resources spent on attacks (e.g., blockchain transactions, blockchain network bandwidth). While I try to include indirect costs (e.g., user/platform reputation damage) or defense costs (e.g., defense implementation costs, platform moderation costs), they are generally difficult to quantify.

Chapter 2

Completed Works

This chapter presents a brief summary of three completed studies. Each section generally follows the same structure: introduce attacks, quantify attack damage, and analyze attack strategies.

2.1 Attacks on P2P networks

Validating transactions is a fundamental operation in any financial system to prevent double-spending and ensure consistency. While banks and payment processors traditionally perform this operation, blockchain systems distribute this task to thousands of nodes in a peer-to-peer (P2P) network and keep the shared ledger simultaneously updated. However, without proper incentives, P2P nodes could deviate from the intended protocol. This chapter will describe a summary of my published paper at SIGMETRICS 2025 [47] that introduces a novel denial-of-service (DoS) attack on blockchain P2P networks (i.e., computer security aspects). I will illustrate that the attack is only feasible because the nodes skip the transaction validation process when the financial market provides value for low latency (i.e., financial aspects).

2.1.1 Introduction

In "Flash Boys" [27], Lewis famously describes how, in the traditional stock market, lower network latency to stock exchange matching engines provides such a competitive advantage that trading firms have started to physically shorten network links – using, e.g., dedicated fiber – and routinely optimize network configuration to reduce latency. Could the same type of optimizations apply to decentralized finance (DeFi), i.e., in the context of blockchain P2P networks? In principle, yes: the concept of Extractable Value (MEV/BEV) [8, 36], i.e., maximizing profits by reordering transactions, implies that centralized business entities have an incentive to reduce latency to deliver pending transactions. Being the fastest to get transactions – even if just by milliseconds – can attract users, bots, and validators who can profit from arbitrage, front- or back-running, and sandwich attacks.

As a case in point, I recently notice that certain *modified nodes* in the Ethereum network tailor their configurations, sidestep transaction validation processes to shorten latency, which

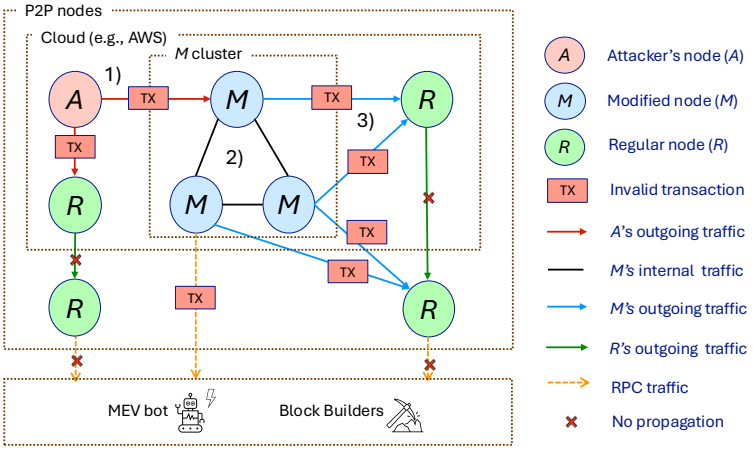

Figure 2.1: Overview of the Blockchain Amplification Attack.

introduce invalid transactions that are not supposed to propagate in the network. Unfortunately, these optimizations open the door to a *Blockchain Amplification Attack*.

In this attack, Figure 2.1, 1) the adversary A sends an invalid transaction (denoted by the red arrow) to a modified node M, that 2) shares it with other modified nodes (black arrows), which will 3) collectively further propagate it (blue arrows) to the rest of the network, amplifying the invalid transaction on the network. The attack motivations are 1) to degrade the quality of the service provided by modified nodes, 2) to inflict traffic costs on modified nodes, and 3) to disrupt all nodes and users in the *entire* network. .

2.1.2 Data

I use publicly available data (RPC nodes, txpool providers) from Flashbots [18], as well as measurements through our customized nodes deployed in a P2P network.

Flashbots' public data: Flashbots has been providing publicly available txpool (a memory buffer that stores pending transactions) information every day [18]. I use this data from Sep. 1st, 2023, to Jan. 11th, 2024. Flashbots collects a set of pending transactions from 1) RPC providers and generic nodes – Infura, Alchemy, A-pool, Flashbots' local node, Mempoolguru, and 2) the infrastructure txpool providers – bloXroute, Chainbound, Eden.

My measurement data: To infer the network topology and estimate attack impact, I modify Geth (i.e., Ethereum node client) to store all messages received in the P2P network layer between Sep. 1st, 2023, and Jan. 25th, 2024. I deploy two nodes in the network to avoid a single point of failure and increase the coverage and robustness of our estimates. Those monitoring nodes enable us to observe transactions *before* they enter the txpool to capture all transaction messages (including invalid ones), and record the message types, the origin node IDs, and the timestamp. This is critical for estimating the number of peer connections for each node, and identifying modified nodes within the P2P network. To further profile each node, I gather peer information such as node names, software, network IDs, compatible protocols, and IP addresses using the `admin_peerEvents` API in our node. In total, I capture about 2,493,695,017 unique messages

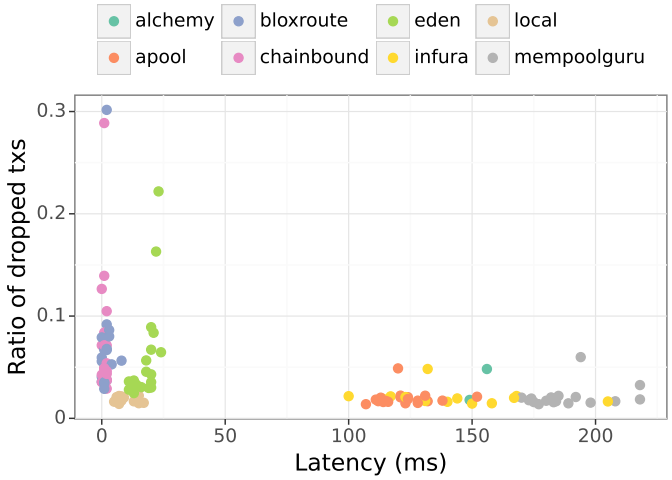

Figure 2.2: Comparing the txpool of centralized entities.

from 36,815 peers in the aforementioned timeframe.

2.1.3 Main results

I will describe three main results: 1) the attack in the wild, 2) modeling amplification, and 3) security-latency trade-off.

Attack in the wild

I show that the attack is feasible and (a variant) is observed in the wild. Based on Flashbots' data, I find that some services deliver transactions much faster than others, but also propagate a significant number of transactions that are not confirmed (i.e., dropped transactions). Figure 2.2 illustrates the bi-modal representation of latency vs. accuracy. The x-axis is the median latency (from the fastest source) from all the transactions, whereas the y-axis is the ratio of dropped transactions. Each data point represents one week of data for each service. Services such as bloXroute, Chainbound, and Eden invest in infrastructure to optimize their propagation flow, resulting in potential latency reduction. However, those "fast" services fail to filter out transactions that are not supposed to persist in the network. Indeed, many of the dropped transactions are invalid: 1) insufficient balance, 2) past nonce, or 3) duplicate (e.g., featuring only a slight change in the data field). I have identified 2,591 instances from 345 Ethereum addresses that target those services and spammed invalid transactions (i.e., sending more than 100 transactions in 12 seconds, with 95% failure rate). This result motivates us to model amplification.

Model amplification

I will precisely estimate the *amplification factor* to show how the attack scales up given the attacker's input. The previous literature uses two metrics to quantify attack effectiveness: Traffic

Amplification Factor (TAF) [25] and Economic Amplification Factor (EAF) [50]. In my scenario, TAF is the ratio of outgoing traffic generated by the attacker to the modified nodes, which corresponds to the original red arrow and the sum of the blue arrows in Figure 2.1, respectively. EAF is the amplification in economic damage, defined as $TAF \cdot \frac{p_{victim}}{p_{attacker}}$ where p is the price per outgoing traffic for a victim and an attacker, respectively. While I defer the detailed mathematical modeling and model parameter estimation to the original paper, I will highlight two important aspects here: 1) estimating the number of peer connections and 2) identifying modified nodes.

First, I propose a new method for estimating the number of active connections for each peer, x_i. Although studies on blockchain P2P network topology exist [28, 55], the key novelty is to estimate *active* peer connections on the main Ethereum network without submitting transactions (i.e., mediating ethical concerns and research costs). I leverage the fact that, for every transaction, a Geth node with x peer connections *uniformly randomly* chooses $\sqrt{x}$ nodes to broadcast a full transaction, while sending just announces (hashes) to the rest. With the assumption about the binomial distribution, I can reverse-engineer the number of peers based on the ratio of those two messages. The large dataset (2.5 billion messages in §2.1.2) helps to reduce the variance in our estimate. In total, I estimate the number of peer connections for 6,005 nodes, with a mean and median of 41 and 31, respectively.

Second, I estimate the ratio of the modified nodes from the set of peers my nodes connect. I label peers that forward invalid transactions to us, which suggests a txpool mechanism was modified. To derive the ratio of modified nodes, I look at the set of nodes in our active connections and check whether any of those are from modified nodes. Both of our nodes maintain approximately 15 connections with modified nodes over time; dividing the total number of connections at each time gives us an average ratio of about 1.5% (175 unique modified nodes in total). Notably, 174 out of 175 peers (1 not available) share an identical git commit hash (in a node name), which is not present in their client's public GitHub repository. This suggests that those nodes are likely from the same entity and ensures that an invalid transaction reaches out to all modified nodes.

All in all, I conclude that the attacker can amplify the original traffic by 3,638 times (TAF). I calculate EAF based on the data transfer cost of AWS, which is used by many services for node deployment. The attacker can strategically deploy its node in the cloud service co-located with modified nodes, to reduce the cost to 0–20 USD/TB (AWS US East). However, modified nodes are organically connected to the rest of the P2P network (90 USD/TB until 10TB). The resulting price discrepancy yields an EAF of 13,827.

Trade-ff discussion

I discuss whether modified nodes should skip transaction validations for financial benefits when they face the risk of amplification attacks. According to my simulation in my original paper, a modified node can relay transactions about 1 millisecond faster without transaction validation to their customers (e.g., MEV searchers). To approximate the economic value of latency, we refer to recent work [49] that examines the impact of block bid timing on MEV profits in Ethereum. If the MEV searchers or block proposers reduce latency by x milliseconds, they could gain $\$0.05x$ extra per block, which is *up to* $\$10{,}800x$ per month. On the other hand, I also estimate potential traffic costs incurred by the amplification attack to be up to $89,000 per node. The financial benefits from lower latency may not necessarily justify the potential attack damages.

2.2 Attacks on self-custody wallets

While traditional banks typically hold users' assets, on blockchain, users can have full control over their assets through self-custody wallets. Although these self-custody wallets mediate counterparty risks (e.g., when banks become insolvent), it poses significant usability challenges for managing cryptographic keys and the corresponding wallet addresses (e.g., hexadecimal or base58 strings). This chapter will describe a summary of my published paper at USENIX Security 2025 [46] that introduces a new phishing scheme targeting blockchain wallets. I will present that the attackers brute-force private keys (sometimes using GPUs) and generate a similar-looking address to spoof the recipient (i.e., computer security aspects). One analysis will demonstrate that the attackers disproportionately target accounts with high balances and frequent/large transactions (i.e., financial aspects).

2.2.1 Introduction

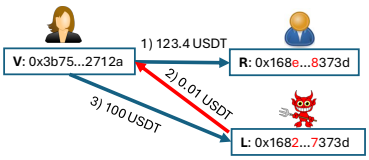

Figure 2.3: Tiny

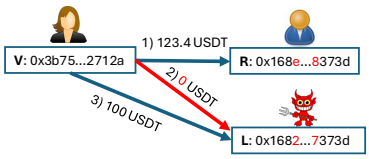

Figure 2.4: Zero-value

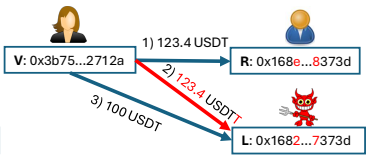

Figure 2.5: Counterfeit

Figure 2.6: Three types of poisoning transfers (tiny, zero-value, and counterfeit token transfers).

Most modern blockchains rely on "wallets" for monetary transfers. Wallet addresses are derived from cryptographic public keys and are often represented by long, hard-to-memorize strings. However, most currency or token transfers require users to manually input or select the recipient's wallet address. A common practice is to copy and paste or select addresses with which one has previously interacted. This introduces a new attack, *blockchain address poisoning*.

The process of the attack is as follows. The attacker first monitors blockchain transactions and identifies asset (ETH or ERC-20 tokens) transfer events. When such transfers are observed, the attacker identifies an intended recipient wallet, R, and generates a corresponding lookalike address L that matches the first and last characters of R (through a brute-force search).

The attacker then submits to the chain phishing transactions that interact with the victim V using a lookalike address L, typically within a short period of time. The attacker can poison the victim through three types of token transfers: sending a small amount—*tiny transfers*, a zero amount—*zero-value transfers*, and counterfeit tokens that resemble the original one—*counterfeit token transfers*. Figure 2.6 depicts these variations. When the victim later attempts to send assets to R, they could get confused and mistakenly send funds to L instead. This confusion typically arises through some combination of UI/UX limitations and the victim failing to carefully check the recipient. A common vector is for a victim to transfer funds to an address that is prepopulated by their wallet software in a list of addresses with which they had recent interactions. In contrast to traditional bank transfers, blockchain transactions are irreversible, making fund recovery challenging and thereby exacerbating the severity of any mistakes.

2.2.2 Main results

I will first introduce the detection algorithm, describe the damage, and two analyses that are particularly relevant to my thesis statement.

Detection

I design a detection algorithm that scans blockchain ledgers efficiently.
Step 1: Identifying potential victims. For block n, I collect the list of addresses $\mathcal{V}$ (potential victims) who have sent the major stablecoins and the addresses $\mathcal{R}$ (intended addresses) that have received these tokens.
Step 2: Identifying possible poisoning transfers. For each potential victim V, I collect all transfers associated in the next 20 minutes, with the assumption that the attackers conduct poisoning transfers right after the original transfer.
Step 3: Extracting lookalike addresses. For every address that has interacted with a potential victim in Step 2, I calculate the similarity score with each of the intended addresses in Step 1. The similarity score (a, b) is the number of consecutive matched digits in the prefix and suffix of two addresses. I define a lookalike address L as one for which $a \geq 3$ and $b \geq 4$ to avoid natural collisions.
Step 4: Categorizing transfers. When a lookalike address L interact with the victim V, I categorize the transfers I have collected into intended transfers, three poisoning transfers (tiny, zero-value, counterfeit token), and final payoff transfers, based on the timing of the transfer (i.e., block timestamp), the set of senders/receivers, the type of tokens, and the transferred amount. Payoff transfers are defined as transfers from the victim to the lookalike address with a non-zero amount.

I evaluate my detector by 1) the external dataset (100% precision, 97.2% recall), 2) analyzing its false negatives, 3) manually verifying the large phished cases, 4) checking the smart contracts used for those attacks, and 5) comparing detection outcomes with different parameters (e.g., window size and similarity threshold).

Detection results

I apply my detection algorithm over the period of Jul. 1st, 2022, to Jun. 30th, 2024, for both Ethereum and BSC. I present the summary statistics for poisoning transfers in Table 2.1.

Combining both chains, I identify over 270M attack attempts (i.e., at least 13 times more than the previous efforts: 21M [54] and 14M [17]) that target over 17M victims. Out of those, attackers successfully received 6,633 transfers, adding up to over 83.8M USD in ill-gotten gains (mostly from Ethereum). This attack appears to be one of the largest and most widely-targeted cryptocurrency phishing schemes observed in the wild [15, 20, 29, 31, 33, 51] at the time of publication.

There are a few interesting observations. First, poisoning transactions consume a non-negligible amount of blockchain resources. On the most active day (Feb. 18th, 2023) on Ethereum, I capture 362,934 poisoning transfers (≈50 transfers per block). Those attack transactions consume 6.6% of total gas used that day, producing wasteful state on the chain and increasing network operating cost. Second, the number of attacks is significantly larger on BSC

Table 2.1: Attack summary statistics

	Blocks	Transactions	All poisoning transfers	Tiny transfers	Zero-value transfers	Counterfeit-token transfers	Victim addresses	Lookalike addresses	Attack contracts	Counterfeit tokens
Ethereum	5,154,722	1,691,529	17,365,954	308,881	7,185,298	9,871,775	1,330,948	6,492,215	3,480	6,280
BSC	20,906,918	16,505,215	252,703,515	3,651,015	140,556,905	108,495,595	16,107,774	43,644,433	406	710
Total	26,061,640	18,196,744	270,069,469	3,959,896	147,742,203	118,367,370	17,438,722	50,136,648	3,892	6,990

than on Ethereum, totaling over 252M poisoning transfers in 17M transactions. The result suggests a higher attack prevalence in chains with lower transaction fees, leading to a significant clutter in UIs and degradation in user experience. Third, this attack is generalizable and scalable across chains. For any EVM-compatible chains, users can use the same wallet addresses across different chains. Attackers can reduce computation especially as the most active accounts frequently hold (ERC-20 and BEP-20) tokens across chains. I observe that attackers reuse 16,903 lookalike addresses and target the same 107,542 victims across two chains.

Attack entities

Following traditional cybercrime literature [42], I aim to link attack instances to extrapolate attack groups and analyze strategies, profitability, and infrastructure. Attack instances include all blockchain addresses that are involved in the poisoning transfers.

Attackers typically optimize their operations by bundling several transfers into a single transaction and reusing addresses and contracts. This behavior allows us to link attack transfers through the "guilt-by-association" heuristic [51]. At a high level, I use three heuristics to cluster attack instances. I assume that 1) two poisoning transfers in the same transaction belongs to the same attacker, 2) an attacker does not specify other attackers' lookalike addresses (to successfully extract payoffs), and 3) an attacker does not share a private key of a wallet (that pays transaction fees). In addition, I carefully removed some "copying bots" (e.g., MEV bots appeared in § 2.1) that copy other attackers poisoning transactions, which could erroneously merge attack groups. Using these heuristics, I identify 49 groups (with more than one lookalike address) in total. I validate the robustness clustering results by 1) verifying the temporal consistency and 2) finding the reuse of their smart contract (byte)codes. Notably, the top groups have more than a million lookalike addresses and conduct millions of poisoning transfers, showing the large-scale operations of those attackers. In my original paper, I further show that most large groups are highly profitable despite their millions of dollars of investment (e.g., transaction fees). On the other hand, smalle groups sometimes suffer from losses due to the high variability of phished amounts and competition against larger groups.

Lookalike addresses

To deceive victims, one goal of the attacker is to generate as similar lookalike addresses as possible (i.e., matching the first and the last few characters). Due to the irreversible nature of wallet addresses, attackers must brute-force private keys to generate lookalike addresses. I observe a substantial disparity in generating capability across groups; the largest group matches 20 digits, while others generate around 14 digits at most. To understand the difference, I empirically

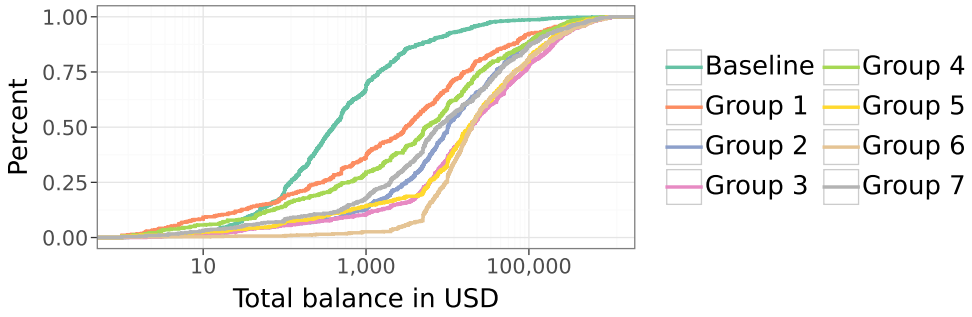

Figure 2.7: CDF of targeted victims' stablecoin balance for the top 7 groups (and the baseline). x-axis is in logscale.

estimate how many wallet addresses can be generated per second with different software implementations (naive, optimized) and hardware types (CPUs, GPUs). The results show that GPUs vastly outperform CPUs, while software optimization provides a relatively marginal improvement. This result indicates that the largest group uses GPUs to generate lookalike addresses because it is practically infeasible to achieve this result by optimizing CPU-based implementation.

Targeted population

Attackers appear to select their targets in advance because the attack typically happens right after the target's original transfer (mostly within 20 minutes). I further investigate whether attackers disproportionately target certain populations compared to the average stablecoin user.

To make the attack more profitable, I conjecture that attackers are more likely to target rich addresses. As different attack groups may employ different strategies, I look at victims on a group-by-group basis on Ethereum. Specifically, I randomly sample 1,000 victims from each of the top seven attack groups (based on the number of lookalike addresses employed). For each victim, I retrieve the sum of their three stablecoins (USDT, USDC, DAI) balance just before when the victim receives the first poisoning transfer from each attack group. I also create a baseline group by randomly sampling 1,000 USDT users and looking up their balance (for three stablecoins) before the most recent transfer. Figure 2.7 illustrates the cumulative distribution function (CDF) of the total stablecoin balance for the victims in each of the seven attack groups and the baseline USDT users. All attack groups tend to target addresses that have significantly more stablecoins than regular USDT users. Groups 6 particularly target rich accounts given that most of the victims own more than 10,000 USD at the time of the attack. Targeting rich accounts seems successful since I detect 83 payoff transfers with more than 100,000 USD. I conduct a similar analysis for the accounts with large transfers or many transactions and find that attackers also disproportionately target those accounts.

As one mediation strategy, I also developed a real-time detection system called Toxin Tagger[1] that identifies poisoning transfers live and broadcasts results on social media.

[1] `https://cryptotrade.cylab.cmu.edu/poisoning/`

2.3 Attacks on financial social media

Users nowadays can not only consult certified financial brokers to manage their funds, but also follow (or even directly copy) trading strategies from financial influencers on social media or online platforms. Despite this increased accessibility, users are conflicted with more ambiguous signals about whom to trust online. This section will describe a summary of my published paper at The Web Conference 2023 [45]. I will show that some malicious accounts creates a set of Sybil accounts in order to manipulate the platform metrics (i.e., computer security aspects) while the level of their misbehavior appears to be correlated with the asset types and price movements (i.e., financial aspects).

2.3.1 Introduction

Over the past decade, individual trading behaviors have experienced a marked change. Individual investors have increasingly relied on social media and other online outlets to 1) show off their profits in hopes of becoming "financial influencers," and 2) ask for financial advice from high-performing traders (i.e., social/copy trading [10, 23]). A similar movement has emerged in the traditional stock market, as discussed in online forums such as the `r/wallstreetbets` [37] (WSB) "sub-Reddit." For instance, in 2021, users of WSB self-organized to purchase large numbers of shares from GameStop (GME) in an alleged attempt to "short squeeze" hedge funds. These activities have been facilitated by the rise of user-friendly financial apps such as Robinhood, lowering the barrier-to-entry to start trading. However, this increased accessibility has been at the expense of a similar increase in online misbehavior. New investors have become the target of various types of attacks, including spam, fraud, and misleading financial advice.

According to cases reported to the Federal Trade Commission, from Jan. 2021 to Mar. 2022, fraud starting from ads/messages on social media has reached USD 1.1 billion, with 40% paid through cryptocurrencies [14]. Despite increasing calls for regulating such malicious behavior, little is known about the types of misbehavior, their prevalence, and potential mitigations.

TradingView[2] is an online platform where traders analyze price charts, post ideas about particular assets, and create trading strategies. It is reportedly the largest trading communication website, with 30M monthly users [43] in 2022. The website also presents social features and encourages investors to interact with each other; for instance, users can disclose their personal information, social media handles, and follow other accounts. My study is the first to characterize the world's largest financial communication platform by leveraging 10 years of users' profiles, activity, and financial interests, and uncover various types of online misbehavior.

2.3.2 Data

I performed account-based snowball sampling, starting with seed users and recursively expanding through their following and follower list [16]. In my original paper, I show that the dataset collection procedure is robust to the number of collection hops and the selection of seed accounts. Overall, I obtained 2,756,809 users (205,842 active; 2,550,967 inactive). User data include user

[2] `https://www.tradingview.com`

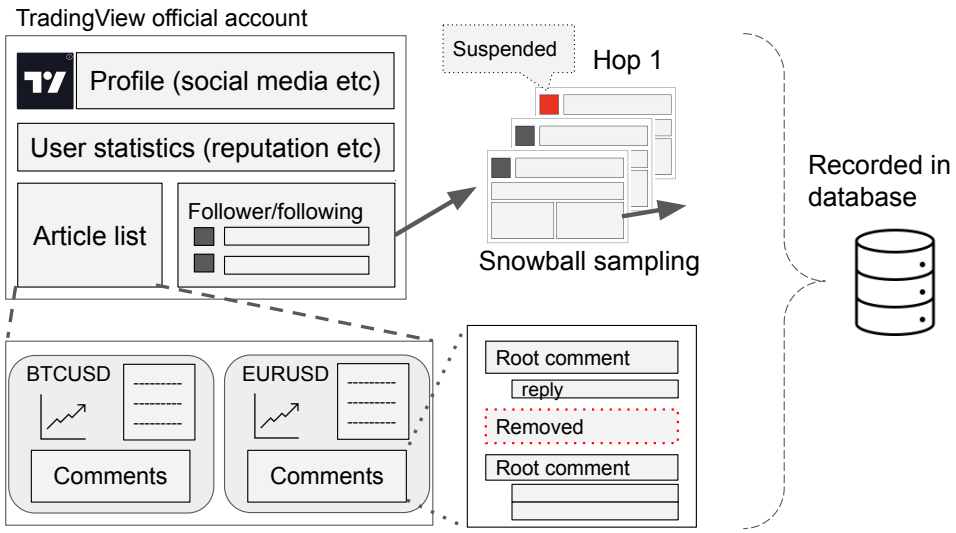

Figure 2.8: The overall data collection architecture (user profiles through snowball sampling, articles, and comments)

plan (free/pro users), number of followers, published articles, reputation score, and social media handles (if listed). For active users (who post at least one article), the data also include their registration date and the user data for all of their followers and followees. I find 3,981 permanently suspended accounts, which account for 1.93% of active users.

For each user, I also collected all of their article posts and comments, resulting in 4,181,673 article posts and 5,273,351 comments from Sep. 5th, 2011, and Aug. 8th, 2022. All articles include the asset symbol being discussed, which can be used to infer the financial interests of the user. While users cannot delete or edit their articles and comments, content moderators can suppress those that violate their policies. I find 16,735 comments, or 0.32% of the total, have been removed.

2.3.3 Major results

I will first characterize the types of misbehavior on the platform, then show the behavior of suspended accounts (i.e., Sybils) and their relation to financial markets.

Misbehavior types

To characterize the types of online abuse across removed comments on TradingView, I randomly subsample a set of 500 removed comments and employ a qualitative coding approach (see details in the original paper). Two of the authors independently coded 500 comments (based on content, without context) using the predefined set of categories from Kumar et al's analysis of Reddit toxicity [24]. Additionally, I define a Spam category that captures comments which contain URLs or invitations to other social media platforms (e.g., WhatsApp, Facebook, Telegram, etc.). Lastly, an "Undefined" category denotes comments for which the coders could not identify the reason for removal.

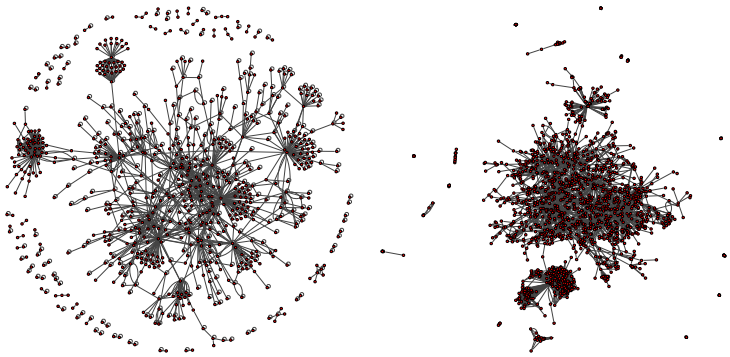

Figure 2.9: The comment (left) and follower (right) networks of suspended users among themselves.

Coders computed their inter-rater agreement using Cohen's Kappa [7] $\kappa = 0.734$, which indicates substantial agreement. The coders then met to discuss, resolve differences, and ultimately agreed on the following breakdown: 30.8% toxic comments, 35.8% spam, and 33.4% undefined. The breakdown of toxic comments is: Insult (25.2%), Identity Attack (1.8%), Call to Leave (0.2%), Threat (0.6%), Sexual Aggression (0.4%), and Identity Misrepresentation (0.0%). Most of the toxic comments seem to originate as an attack on trading ideas and ability, with some variation on the type of insult (call to leave/suspend or attacking someone's identity based on user profiles). For spam comments, I observed invites to external trading websites or social media. These comments often start as related to the article but quickly proceed to exhort users to visit a URL (often with the use of URL shorteners). With regards to the Undefined category, the coders found many comments that seemed to be related to reputation manipulation. However, since the coders could not conclusively rule, without context, whether these comments were harvesting reputation, they were labeled as Undefined.

Network of suspended accounts

Guided by the Sybil account research on other platforms [1, 5, 6, 48, 53], I will illustrate the high level interaction between suspended users. In Figure 2.9 (left), nodes represent suspended users, and edges exist if one suspended user comments on another suspended user's article. While suspended accounts are less than 2% of the total active accounts, there is a giant component, where many of the suspended users are interacting, within which tight account clusters exist. I observe a similar behavior in the follower-followee network in Figure 2.9 (right). This behavior is consistent with that of users coordinating to increase their reputation or the popularity of their articles. Another experiment in the paper shows that suspended accounts tend to interact with accounts having closer registration dates, implying the possibility of mass account creation.

Misbehavior and financial market

I quantify abuse on the platform through the number of removed comments. I am interested in examining what types of assets have often been the targets of online abuse, based on the symbols that have been attached to all the articles and the comments. Some assets attract a

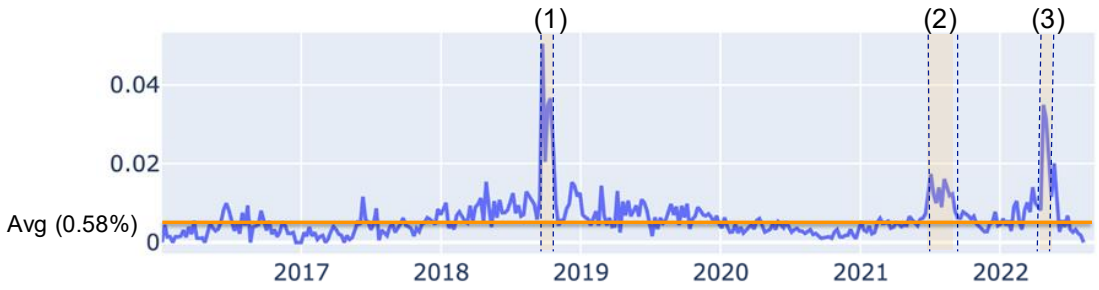

Figure 2.10: The ratio of articles with removed comments. (1)-(3) corresponds to the spike explained in the main text.

disproportionately large number of malicious activities. For instance, nearly 2.84% of articles on "AMC" contain at least one removed comment, which is five times larger than the average. In general, the most abusive assets seem to be often featured on social media, including meme stocks ("AMC", "GME"), DeFi/NFT related coins ("AAVE", "ALICEUSDT"), the LUNA stable coin meltdown ("LUNA"), Bitcoin-related ("BTCUSD", "BTCUSDT", "XBT", "BSVUSD") and vaccine-related ("MRNA"). Monitoring assets that trend as "meme assets" could lead to a more efficient social media content moderation process.

Furthermore, I investigate how the abuse of the platform changes over time, since 2016. In Figure 2.10 shows the ratio of articles with removed comments, Though the ratio seems to be relatively stable over time, there are spikes around (1) Sep. 23rd to Oct. 21st, 2018, (2) Jul. 4th to Sep. 12th, 2021, and (3) Apr. 17th to Jun. 5th, 2022. I manually investigate each spike to come up with explanations. For the first spike, Ripple ("XRP") had a relatively high removal rate; it experienced a huge price turmoil, which may have triggered some of the abusive behavior, but the root cause is still unclear. For the second spike, "AMC" has a remarkably high toxic rate (6.85%), which corresponds to the second price spike from Aug. to Sep. 2021. The third spike can be attributed to the crash in Terra's LUNA/UST, where most cryptocurrencies crashed; "BTCUSDT" and "LUNAUSDT" got particularly abusive. The price fluctuation seems to have some association with the abuse of the platform.

Another major contribution of this paper is to assess the platform's content moderation effort. I demonstrate that the number of removed comments, the number of moderated articles, and the difference in registration date (i.e., a proxy of fake accounts) are correlated with account suspension likelihood. However, those violations do not equally lead to suspension between free and pro users (those with paid subscriptions), questioning the fairness of the platforms' content moderation effort.

Chapter 3

Proposed work and timeline

Most assets are typically issued by large entities, including governments (e.g., fiat currencies) and corporations (e.g., stocks). With smart contracts, a programmable code on blockchains, users can easily create their own tokens. For instance, users can create a meme coin (e.g., as an internet joke) through a few clicks on a platform like `pump.fun` with nearly zero cost. This chapter will propose a study that investigates various manipulative behavior on meme coin markets. I aim to illustrate that attackers create many wallet addresses or spoof existing tokens (i.e., computer security aspects) and manipulate the trading signals, such as price, volume, token distribution, and token authenticity, to deceive other traders for profits (i.e., financial aspects).

3.1 Introduction

A pump and dump (P&D) is a form of price manipulation where an attacker creates an artificial price increase ("pump") and sells their assets at the peak price to make profits ("dump"). The success of the attack relies on manipulating the signals (e.g., price, trading volume) that traders perceive and tricking them into buying the assets. While this scheme has long observed in traditional stock markets, it has become prevalent in cryptocurrency markets [30, 52]. In the last few years, attackers have targeted listed cryptocurrencies on centralized exchanges (e.g., Binance) and utilized social media platforms (e.g., Telegram, Discord) for coordination and advertisement [32, 34]. Attacks were executed on the exchanges' orderbooks (i.e., off-chain) because the transaction fees were too high and confirmation times were too slow for price manipulation. However, a more recent blockchain like Solana provides nearly zero-transaction fees and fast confirmation times, making it an ideal environment for market manipulation. In particular, an attacker can easily create their own tokens (i.e., not necessarily restricted to the listed tokens), create many blockchain wallet addresses (i.e., "Sybil" accounts), and create artificial ("organic"-like) trading volume. Simultaneously, the emergence of meme coin platforms (e.g., `pump.fun`) has allowed any users to easily create tokens and "enjoy" trading on a user-friendly interface. Indeed, `pump.fun` attracted over 82 billion USD trading volume [9] and launched more than 15 million tokens [21] since its inception in early 2024. This change provides a unique opportunity for researchers to publicly observe all trading activities at the account-level (unlike closed-source centralized exchanges) and uncover the manipulation strategies in the wild. While

there exist some web articles describing the P&D prevalence on pump.fun [2, 38], this is the first large-scale academic study to uncover various attacks that manipulate trading signals (e.g., price, trading volume, token distribution, token authenticity).

3.2 Data

I will first focus on pump.fun platform, the pioneer in meme coin marketplaces that still covers 95% of "graduated" coins (exceeding a certain threshold in market cap) [11]. I and my collaborators will collect two types of data: 1) on-chain data from the Solana blockchain and 2) off-chain data from the pump.fun website. For on-chain data, I will prepare the list of all tokens minted on pump.fun. The data include signatures (i.e., transaction hash), token addresses, token creator addresses, and block timestamp. To identify the manipulative behavior, I also plan to collect all transfers associated with each token. For off-chain data, I will use pump.fun's APIs to collect coin information (e.g., coin name, symbol, description, related social media links) and comments on each coin's page.

3.3 Attack example

I will present one example *Thunder* token[1] to illustrate how attackers manipulate the price. Figure 3.1 shows its price chart on Nov. 1st, 2024. The price gradually increases with some bumps, and suddenly drops to nearly zero. I identified five types of accounts that were involved in the manipulation: 1) a master, 2) a token creator, 3) a coordinator, 4) "hodlers," and 5) wash traders. Given that an account needs SOL (native currency on Solana) to initiate transactions, I define account A as a "funder" of account B if A is the first one to send SOL to B[2]. I assume that accounts A and B are from the same entity unless A is a well-known centralized entity (e.g., exchanges). Figure 3.2 illustrates the funding relationship among those accounts. The attack proceeds as follows:

1. The master (DjzH6fx...) creates (funds) a token creator (Hkno...) and a coordinator (84U8...). The token creator creates a Thunder token.
2. The coordinator generates a set of hodlers (e.g., 3kVK...) and wash traders (e.g., 6xaY...).
3. Hodlers buy Thunder tokens in the beginning and hold them until the end (no trade in between).
4. Wash traders buy and sell Thunder tokens repeatedly to create artificial trade volume. The drops in price are partly due to those wash trades.
5. Eventually, all Sybils (i.e., hodlers and wash traders) send (remaining) Thunder tokens to the coordinator, who sells them at once to realize profits.
6. All accounts send the remaining SOL back to the master.
7. A master creates another coordinator to conduct similar attacks on different tokens.

[1] 68G5UB7mw5twJDus9T48ipmNE2jxuujqBNLrfKJwpump
[2] This definition is used in most online chain scanners such as solscan.io or intel.arkm.com

Figure 3.1: Price chart of Thunder token on Nov. 1st, 2024.

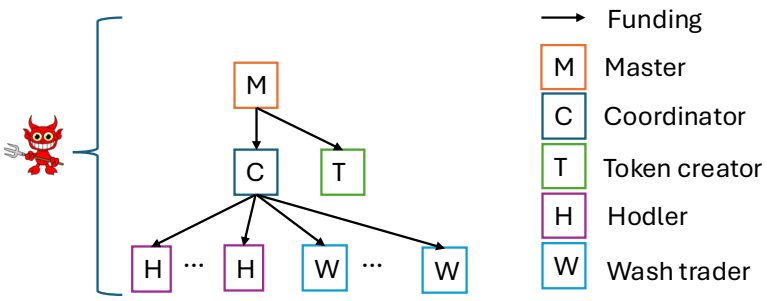

Figure 3.2: Funding relationship among accounts that traded Thunder token.

The scheme slightly differs across tokens, but the overall structure remains the same; a set of accounts funded (or controlled) by the same address buys and sells the same token. In particular, the success of this attack depends on other accounts that do not belong to the master and are deceived into buying Thunder tokens with an artificial signal. There are a few additional techniques to create such signals.

First, attackers distribute tokens to many Sybil accounts to create a wider ownership distribution because many external websites (e.g., DEX scanners) flag the tokens with higher concentration. Second, attackers can use the "bundling" technique, which allows them to immediately execute multiple buy (or sell) orders in one transaction (before the price changes). Third, attackers can not only manipulate on-chain trade signals but also utilize social signals. For instance, attackers can post comments on `pump.fun`'s website from multiple (on-chain) accounts to increase social engagement.

I will also explore more sophisticated attacks. First, to evade detection, attackers can create Sybils through funding twice (i.e., Sybils' funders are all different) or generate Sybil accounts directly from centralized exchanges or mixing services. Indeed, I find a service that helps users to create many Sybil accounts from mixing services. Second, some attackers forcefully "graduate" tokens on `pump.fun`. Token "graduation" refers to an event when a token meets certain criteria (e.g., a high market cap), gets listed on external exchanges (both in centralized and decentralized exchanges), and typically attracts more investors (i.e., hence higher price). Attackers can push the price up to the graduation threshold and sell their tokens after the listings.

3.4 Research questions and proposed methods

I have the following research questions.

1. *Can I identify the signs of market manipulation or wash trading?* As described in §3.3, I will develop heuristics to identify suspicious trading activities. For instance, I will retrieve the funding address of each account and identify the case where a set of accounts with the same funder trade the same token. For each manipulation, I will try to quantify the profits made by the attackers (by grouping accounts and calculating the change in SOL balance). I will also explore the variants of those attacks (as described above) and develop the detection system accordingly.
2. *Can we detect that behavior at a large scale?* Downloading two years of `pump.fun`'s token transfers is practically infeasible, with a single day of transaction data exceeding 50GB. To address this challenge, I plan to strategically select coins without downloading all coin transaction data. I plan to construct a classifier that predicts vulnerable coins likely to experience manipulation, only based on coin metadata (e.g., token issuer, token name and description, social media links). As a ground truth, I will randomly pick some coins and download their entire transfer history to get the list of manipulated coins. This experiment will build upon the previous works that predict coin manipulation before it occurs [19, 22, 26, 44].
3. *Can I find any other attacks or misbehavior?* In addition to market manipulation and wash trading, I will explore other attack types. For instance, there are many copy tokens to the original one, often with a slight alteration, typosquatting (e.g., hundreds of Trump coins). I will also analyze the tokens that have been suspended by the platforms to identify potential harmful contents (e.g., hate, misinformation, nudity, violence), similar to § 2.3.
4. *Can we find anything unique about this market?* There are a few special traders in meme coin markets that have not been observed in any other financial markets. I am interested in analyzing their behavior and their involvement in the above attacks. Anecdotal evidence suggests that there exist "professional token creators" who create hundreds of meme coins every day. I will investigate their strategies (e.g., how do they choose what types of tokens to create?) and profits over time. Another interesting actor is the token sniper bot that instantly buys ("snipes") newly launched tokens. I aim to uncover their strategies (e.g., how do they choose the token to buy? To what extent does the latency matter to performance?), and their evolution over time (e.g., are they gradually getting faster at buying tokens due to the competition?). The final example is the Telegram bots that enable users to trade tokens through the Telegram interface. As described in § 2.2, setting up blockchain wallets poses a significant usability challenge for users. Some Telegram bots (e.g., BoNKbot, Trojan bot) generate custody wallets (i.e., keep the user's private keys) and offer various services (e.g., execute trades, perform copy trading, or provide low-latency nodes to connect). By leveraging the users' transaction data that pay fees to those services on-chain, I plan to identify all Telegram bot users and analyze their behavior compared to normal users.

3.5 Proposed Timeline

The proposed timeline for the remaining work on this thesis is as follows.

- **February 2026** Propose thesis.
- **February - April 2026** Perform data collection and analysis for the meme coin manipulation study.
- **May - August 2026** Summer internship or work on other projects (e.g., cybercrime on Telegram), potentially with REUSE students.
- **September - October 2026** Conduct further analysis on the meme coin manipulation study.
- **November 2026** Write thesis.
- **December 2026** Defend thesis.

Bibliography

[1] Muhammad Al-Qurishi, Majed Alrubaian, Sk Md Mizanur Rahman, Atif Alamri, and Mohammad Mehedi Hassan. A prediction system of sybil attack in social network using deep-regression model. *Future Generation Computer Systems*, 87:743–753, 2018. 2.3.3

[2] Pine Analytics. Exit liquidity machines. `https://x.com/PineAnalytics/status/1914301091337953734`, 2025. Accessed May 10th, 2025. 3.1

[3] Ross Anderson, Chris Barton, Rainer Böhme, Richard Clayton, Michel JG Van Eeten, Michael Levi, Tyler Moore, and Stefan Savage. Measuring the cost of cybercrime. In *The economics of information security and privacy*, pages 265–300. Springer, 2013. 1.1

[4] Ross Anderson, Chris Barton, Rainer Bölme, Richard Clayton, Carlos Ganán, Tom Grasso, Michael Levi, Tyler Moore, and Marie Vasek. Measuring the changing cost of cybercrime. 2019. 1.1

[5] Qiang Cao, Michael Sirivianos, Xiaowei Yang, and Tiago Pregueiro. Aiding the detection of fake accounts in large scale social online services. In *9th USENIX Symposium on Networked Systems Design and Implementation (NSDI 12)*, pages 197–210, 2012. 2.3.3

[6] Meta Transparency Center. Inauthentic behavior. `https://transparency.fb.com/policies/community-standards/inauthentic-behavior/`, 2022. Accessed Sep. 29th, 2022. 2.3.3

[7] Jacob Cohen. A coefficient of agreement for nominal scales. *Educational and psychological measurement*, 20(1):37–46, 1960. 2.3.3

[8] Philip Daian, Steven Goldfeder, Tyler Kell, Yunqi Li, Xueyuan Zhao, Iddo Bentov, Lorenz Breidenbach, and Ari Juels. Flash boys 2.0: Frontrunning in decentralized exchanges, miner extractable value, and consensus instability. In *2020 IEEE Symposium on Security and Privacy (SP)*, pages 910–927. IEEE, 2020. 2.1.1

[9] DeFiLlama. pump.fun. `https://defillama.com/protocol/dexs/pump.fun`, 2025. Accessed Jan. 15th, 2026. 3.1

[10] Philipp Doering, Sascha Neumann, and Stephan Paul. A primer on social trading networks–institutional aspects and empirical evidenc. In *EFMA annual meetings*, 2015. 2.3.1

[11] Decentralized Dog. Pumpfun controls 95% of token graduation market. `https://coinmarketcap.com/academy/article/pumpfun-controls-95percent-of-token-graduation-market`, 2025. Accessed Jan. 15th, 2026. 3.2

[12] Ismail Erturk, Julie Froud, Sukhdev Johal, Adam Leaver, and Karel Williams. The democratization of finance? promises, outcomes and conditions. *Review of international political economy*, 14(4):553–575, 2007. 1, 1.1

[13] Larry Fink. The democratization of investing: Expanding prosperity in more places, for more people. https://corpgov.law.harvard.edu/2025/04/14/the-democratization-of-investing-expanding-prosperity-in-more-places-fo Apr 2025. Accessed Jan. 21st, 2026. 1, 1.1

[14] Emma Fletcher. Reports show scammers cashing in on crypto craze. https://www.ftc.gov/news-events/data-visualizations/data-spotlight/2022/06/reports-show-scammers-cashing-crypto-craze#crypto1, Jul 2022. 2.3.1

[15] Bingyu Gao, Haoyu Wang, Pengcheng Xia, Siwei Wu, Yajin Zhou, Xiapu Luo, and Gareth Tyson. Tracking counterfeit cryptocurrency end-to-end. *Proceedings of the ACM on Measurement and Analysis of Computing Systems*, 4(3):1–28, 2020. 2.2.2

[16] Leo A Goodman. Snowball sampling. *The annals of mathematical statistics*, pages 148–170, 1961. 2.3.2

[17] Shixuan Guan and Kai Li. Characterizing ethereum address poisoning attack. In *Proceedings of the 2024 on ACM SIGSAC Conference on Computer and Communications Security*, pages 986–1000, 2024. 2.2.2

[18] Chris Hager. Mempool dumpster. https://mempool-dumpster.flashbots.net/index.html, Aug 2023. Accessed Nov. 29th, 2023. 2.1.2

[19] JT Hamrick, Farhang Rouhi, Arghya Mukherjee, Amir Feder, Neil Gandal, Tyler Moore, and Marie Vasek. An examination of the cryptocurrency pump-and-dump ecosystem. *Information Processing & Management*, 58(4):102506, 2021. 2

[20] Bowen He, Yuan Chen, Zhuo Chen, Xiaohui Hu, Yufeng Hu, Lei Wu, Rui Chang, Haoyu Wang, and Yajin Zhou. Txphishscope: Towards detecting and understanding transaction-based phishing on ethereum. In *Proceedings of the 2023 ACM SIGSAC Conference on Computer and Communications Security*, pages 120–134, 2023. 2.2.2

[21] jhackworth. Pump.fun. https://dune.com/jhackworth/pumpfun, 2025. Accessed Jan. 15th, 2026. 3.1

[22] Josh Kamps and Bennett Kleinberg. To the moon: defining and detecting cryptocurrency pump-and-dumps. *Crime Science*, 7(1):1–18, 2018. 2

[23] Daisuke Kawai, Kyle Soska, Bryan Routledge, Ariel Zetlin-Jones, and Nicolas Christin. Stranger danger? investor behavior and incentives on cryptocurrency copy-trading platforms. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems*, pages 1–20, 2024. 2.3.1

[24] Deepak Kumar, Jeff Hancock, Kurt Thomas, and Zakir Durumeric. Understanding longitudinal behaviors of toxic accounts on reddit. *arXiv preprint arXiv:2209.02533*, 2022. 2.3.3

[25] Sanjeev Kumar. Smurf-based distributed denial of service (ddos) attack amplification in in-

ternet. In *Second International Conference on Internet Monitoring and Protection (ICIMP 2007)*, pages 25–25. IEEE, 2007. 2.1.3

[26] Massimo La Morgia, Alessandro Mei, Francesco Sassi, and Julinda Stefa. The doge of wall street: Analysis and detection of pump and dump cryptocurrency manipulations. *ACM Transactions on Internet Technology*, 23(1):1–28, 2023. 2

[27] Michael Lewis. *Flash boys: a Wall Street revolt*. WW Norton & Company, 2014. 2.1.1

[28] Kai Li, Yuzhe Tang, Jiaqi Chen, Yibo Wang, and Xianghong Liu. Toposhot: uncovering ethereum's network topology leveraging replacement transactions. In *Proceedings of the 21st ACM Internet Measurement Conference*, pages 302–319, 2021. 2.1.3

[29] Kai Li, Darren Lee, and Shixuan Guan. Understanding the cryptocurrency free giveaway scam disseminated on twitter lists. In *2023 IEEE International Conference on Blockchain (Blockchain)*, pages 9–16. IEEE, 2023. 2.2.2

[30] Tao Li, Donghwa Shin, and Baolian Wang. Cryptocurrency pump-and-dump schemes. *Journal of Financial and Quantitative Analysis*, pages 1–59, 2018. 3.1

[31] Xigao Li, Anurag Yepuri, and Nick Nikiforakis. Double and nothing: Understanding and detecting cryptocurrency giveaway scams. In *Proceedings of the Network and Distributed System Security Symposium (NDSS)*, 2023. 2.2.2

[32] Mehrnoosh Mirtaheri, Sami Abu-El-Haija, Fred Morstatter, Greg Ver Steeg, and Aram Galstyan. Identifying and analyzing cryptocurrency manipulations in social media. *IEEE Transactions on Computational Social Systems*, 8(3):607–617, 2021. 3.1

[33] Muhammad Muzammil, Zhengyu Wu, Lalith Harisha, Brian Kondracki, and Nick Nikiforakis. Typosquatting 3.0: Characterizing squatting in blockchain naming systems. 2024. 2.2.2

[34] Leonardo Nizzoli, Serena Tardelli, Marco Avvenuti, Stefano Cresci, Maurizio Tesconi, and Emilio Ferrara. Charting the landscape of online cryptocurrency manipulation. *IEEE access*, 8:113230–113245, 2020. 3.1

[35] Commition of the European Communities. Towards a general policy on the fight against cyber crime. `https://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=COM:2007:0267:FIN:EN:PDF`, 2007. Accessed Jan. 22nd, 2026. 1.1

[36] Kaihua Qin, Liyi Zhou, and Arthur Gervais. Quantifying blockchain extractable value: How dark is the forest? In *2022 IEEE Symposium on Security and Privacy (SP)*, pages 198–214. IEEE, 2022. 2.1.1

[37] Reddit. /r/wallstreetbets. `https://www.reddit.com/r/wallstreetbets/`, Accessed 2022-10-12. 2.3.1

[38] Solidus Labs Research. The 2025 rug pull report: Rug pulls and pump-and-dumps on solana. `https://www.soliduslabs.com/reports/solana-rug-pulls-pump-dumps-crypto-compliance`, 2025. Accessed May 10th, 2025. 3.1

[39] Bryan Routledge and Ariel Zetlin-Jones. The financialization of cryptocurrencies ibm supply chain and blockchain blog. `https://www.ibm.com/blogs/blockchain/`

2021/05/the-financialization-of-cryptocurrencies/, May 2021. Accessed May. 21st, 2021. 1

[40] Robert J Shiller. The new financial order: Risk in the 21st century. 2003. 1.1

[41] Kyle Soska. *Security defender advantages via economically rational adversary modeling*. PhD thesis, Carnegie Mellon University, 2021. 1.1

[42] Janos Szurdi, Balazs Kocso, Gabor Cseh, Jonathan Spring, Mark Felegyhazi, and Chris Kanich. The long {"Taile"} of typosquatting domain names. In *23rd USENIX Security Symposium (USENIX Security 14)*, pages 191–206, 2014. 2.2.2

[43] TradingView. Advertise on tradingview. https://www.tradingview.com/advertising-info/, Oct 2022. Accessed Oct. 6th, 2022. 2.3.1

[44] Taro Tsuchiya. Profitability of cryptocurrency pump and dump schemes. *Digital Finance*, 3(2):149–167, 2021. 2

[45] Taro Tsuchiya, Alejandro Cuevas, Thomas Magelinski, and Nicolas Christin. Misbehavior and account suspension in an online financial communication platform. In *Proceedings of the ACM Web Conference 2023, Austin, TX, USA*, 2023. 2.3

[46] Taro Tsuchiya, Jin-Dong Dong, Kyle Soska, and Nicolas Christin. Blockchain address poisoning. In *Proceedings of the 34th USENIX Security Symposium (USENIX Security'25), Seattle, WA*, 2025. 2.2

[47] Taro Tsuchiya, Liyi Zhou, Kaihua Qin, Arthur Gervais, and Nicolas Christin. Blockchain amplification attack. In *Proceedings of the 2025 ACM SIGMETRICS Conference, Stony Brook, NY*, 2025. 2.1

[48] Bimal Viswanath, Ansley Post, Krishna P Gummadi, and Alan Mislove. An analysis of social network-based sybil defenses. *ACM SIGCOMM Computer Communication Review*, 40(4):363–374, 2010. 2.3.3

[49] Anton Wahrstätter, Liyi Zhou, Kaihua Qin, Davor Svetinovic, and Arthur Gervais. Time to bribe: Measuring block construction market. *arXiv preprint arXiv:2305.16468*, 2023. 2.1.3

[50] Huangxin Wang, Zhonghua Xi, Fei Li, and Songqing Chen. Abusing public third-party services for edos attacks. In *10th USENIX Workshop on Offensive Technologies (WOOT 16)*, 2016. 2.1.3

[51] Pengcheng Xia, Haoyu Wang, Bingyu Gao, Weihang Su, Zhou Yu, Xiapu Luo, Chao Zhang, Xusheng Xiao, and Guoai Xu. Trade or trick? detecting and characterizing scam tokens on uniswap decentralized exchange. *Proceedings of the ACM on Measurement and Analysis of Computing Systems*, 5(3):1–26, 2021. 2.2.2, 2.2.2

[52] Jiahua Xu and Benjamin Livshits. The anatomy of a cryptocurrency {Pump-and-Dump} scheme. In *28th USENIX Security Symposium (USENIX Security 19)*, pages 1609–1625, 2019. 3.1

[53] Zhi Yang, Christo Wilson, Xiao Wang, Tingting Gao, Ben Y Zhao, and Yafei Dai. Uncovering social network sybils in the wild. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 8(1):1–29, 2014. 2.3.3

[54] Guoyi Ye, Geng Hong, Yuan Zhang, and Min Yang. Interface illusions: Uncovering the rise of visual scams in cryptocurrency wallets. 2024. 2.2.2

[55] Chonghe Zhao, Yipeng Zhou, Shengli Zhang, Taotao Wang, Quan Z Sheng, and Song Guo. Dethna: Accurate ethereum network topology discovery with marked transactions. *arXiv preprint arXiv:2402.03881*, 2024. 2.1.3