On how zero-knowledge proof blockchain mixers improve, and worsen user privacy

Wang, Zhipeng; Chaliasos, Stefanos; Qin, Kaihua; Zhou, Liyi; Gao, Lifeng; Berrang, Pascal; Livshits, Ben; Gervais, Arthur

On How Zero-Knowledge Proof Blockchain Mixers Improve, and Worsen User Privacy

Zhipeng Wang*, Stefanos Chaliasos*, Kaihua Qin*, Liyi Zhou*,
Lifeng Gao*, Pascal Berrang†, Ben Livshits*, and Arthur Gervais*
*Imperial College London, United Kingdom
†University of Birmingham, United Kingdom

***Abstract*—One of the most prominent and widely-used blockchain privacy solutions are zero-knowledge proof (ZKP) *mixers* operating on top of smart contract-enabled blockchains. ZKP mixers typically advertise their level of privacy through a so-called anonymity set size, similar to k-anonymity, where a user hides among a set of k other users.**

In reality, however, these anonymity set claims are mostly inaccurate, as we find through empirical measurements of the currently most active ZKP mixers. We propose five heuristics that, in combination, can increase the probability that an adversary links a withdrawer to the correct depositor on average by 51.94% (108.63%) on the most popular Ethereum (ETH) and Binance Smart Chain (BSC) mixer, respectively. Our empirical evidence is hence also the first to suggest a differing privacy-predilection of users on ETH and BSC. We further identify 105 Decentralized Finance (DeFi) attackers leveraging ZKP mixers as the initial funds and to deposit attack revenue (e.g., from phishing scams, hacking centralized exchanges, and blockchain project attacks).

State-of-the-art mixers are moreover tightly intertwined with the growing DeFi ecosystem by offering "anonymity mining" (AM) incentives, i.e., mixer users receive monetary rewards for mixing coins. However, contrary to the claims of related work, we find that AM does not always contribute to improving the quality of an anonymity set size of a mixer, because AM tends to attract privacy-ignorant users naively reusing addresses.

I. Introduction

It is well-known that non-privacy focussed permissionless blockchains, such as Bitcoin, offer pseudonymity rather than anonymity [15], [25]. Every blockchain transaction discloses the transferred amount, time, transaction fees, and user addresses. While privacy-preserving blockchains [33], [42], [27] successfully protect their users' privacy, retrofitting a blockchain with privacy has proven challenging and remains an active research area [32], [23], [22], [47], [26], [43], [39], [40], [31]. The solution space can be broadly divided into *(i)* privacy-by-design blockchains and *(ii)* add-on privacy solutions, which are retrofitted, e.g., as a decentralized application (dApps) on top of non-privacy-preserving blockchains.

One of the most widely-used add-on privacy solutions, undoubtedly inspired by Zerocash [42], are zero-knowledge proof-based (ZKP) mixers, where users deposit a *fixed* amount of coins into a pool and later withdraw these coins to a new address [6], [9], [7], [8], [1]. If used carefully, such decentralized mixer should break the *linkability* between a deposit and a new withdrawal address. ZKP mixers are dApps on smart contract-enabled blockchains (e.g., ETH and BSC). One of the most active ZKP mixers, Tornado.Cash reports an anonymity set size of $12{,}189$ for its largest pool (i.e., 1 `ETH` pool), by counting the unique deposit addresses on November 1st, 2021. This anonymity set suggests that, given a withdrawal transaction, the corresponding depositor can be hidden among the $12{,}189$ addresses.

ZKP mixers typically focus only on the linkability of addresses on the blockchain layer and leave the remaining operational and privacy-relevant decisions to the user. Users are therefore entrusted to follow the best privacy practices, e.g., no address/wallet re-use, clean browser cache/cookies, preventing browser fingerprinting, VPN/proxy services, etc.

In this work, we however find that the mixer privacy depends on several factors, such as the anonymity set of the mixer pool and crucially, the behavior of other pool users. We perform an empirical analysis of the most active mixers by analyzing the mixer usage over time, the behavior of depositors, withdrawers, and the pre- and post-mixer flow of funds. We then present five heuristics to accurately quantify the anonymity set when considering the user behavior, such as deposit and withdrawal addresses, the asset flow, etc. We attempt to validate the heuristics by synthesizing a candidate ground truth dataset from privacy-exposing side-channels. Furthermore, we identify how adversaries launder coins through ZKP mixers. Finally, we analyze the impact of Anonymity Mining (AM) [30] on a mixer's privacy and discover counter-intuitive results.

We summarize our contributions as follows:

1. Empirical Analysis of Existing Mixers: We empirically analyze the usage of the two most popular ZKP mixers, Tornado.Cash (TC) and Typhoon.Network (TN). We find that at least 18 malicious addresses directly withdraw `ETH` from TC as adversarial source of funds, while 87 malicious addresses deposit 372.1M USD (4.1% of the total deposit volume) into TC, with the likely attempt to hide their traces. We find that the average deposit volume of malicious addresses is $10.7\times$ larger than the average deposit volume of a TC user.

2. Measuring the Anonymity Set Size: We propose five heuristics leveraging on-chain data to derive a more accurate mixer pool anonymity set size, than the naive enumeration of its deposit addresses. Combining heuristics proves powerful, as our evaluation shows that the probability that an adversary links a withdrawer to the correct depositor rises on

average by 51.94% (108.63%) on Tornado.Cash (on ETH) and Typhoon.Network (on BSC) respectively. We are hence the first to provide quantitative evidence that may indicate a user behavior difference w.r.t. privacy on two different blockchains. Our results may also support the hypothesis that the biggest anonymity set continues to attract privacy-aware users, similar to how liquidity attracts liquidity in financial exchanges.

3. Anonymity Mining's Impact on Privacy: We are the first to study and empirically evaluate the impact of AM in ZKP mixers. Contrary to the claims of related work [30], we find that AM does not always increase a mixer's anonymity set size quality, because AM appears to attract privacy-ignorant users, primarily interested in mining rewards. After applying our first heuristic measuring the TC pools' anonymity set, we find that the relative increase of the probability that an adversary links a withdrawer to the correct depositor rises from 7.00% (before AM launch) to 13.50% (after AM launch) on average.

4. Heuristic Validation: We extract three orthogonal privacy-exposing side-channels that help to validate our heuristics based on data from, *(i)* airdrops [48], *(ii)* the Ethereum Name Service and *(iii)* the DeFi explorer Debank. From these side-channels, we build a candidate ground truth dataset, which we plan to open source, and which allows validating privacy-exposing heuristics. Given our reproducible dataset, we find that our heuristics yield an average F1 score of 0.55.

The remainder of the paper is organized as follows. Section II provides an overview of blockchain and existing DeFi mixers. We outline our system and threat model along with the considered privacy metrics in Section III. In Section IV we present our empirical measurements of ZKP mixer pools. In Section V, we provide our heuristics to measure the realistic anonymity set size. We study the novel phenomena of incentivized mixer pools in Section VI. In Section VII, we attempt to gather candidate ground truth data to validate our linking results in mixers. We discuss the implications of our work in Section VIII. We outline related works in Section IX and conclude the paper in Section X.

II. Background

In this section, we outline the required background for mixer pools on non-privacy-preserving blockchains.

A. Blockchain and Smart Contracts

Permissionless blockchains act as a distributed ledger on top of a peer-to-peer (P2P) network [36]. Smart contracts are quasi Turing-complete programs that typically execute within a virtual machine and allow users to construct various applications [49]. For instance, Decentralized Finance is a financial ecosystem that runs autonomously on smart-contracts-enabled blockchains. The total locked value in DeFi has reached over 93B USD at the time of writing. Many DeFi applications are inspired and mirrored by traditional centralized finance systems, such as asset exchanges, lending and borrowing platforms, margin trading systems, and derivatives. A blockchain transaction can be used to transfer blockchain tokens or to trigger the execution of smart contract functions. The sender of a transaction usually pays for the cost of the entire smart contract execution caused by that transaction. For a more thorough background on blockchains and smart contracts, we refer the interested reader to the relevant surveys [21], [16].

TABLE I: Comparison of ZKP mixers on ETH, BSC, and Polygon on November 1st, 2021.

Mixers	Chain	TVL (USD)	Total USD deposited	# Pools	# depositors	# withdrawers
TC [6]	ETH	1.11B	9.12B	19	23,529	32,860
TP [7]	ETH	678K	96.58M	24	1,217	1,174
TN [8]	BSC	0.86M	35.26M	13	4,811	6,318
	Polygon	546	3,683	4	31	41
Cyclone [1]	ETH	4.00M	70.94M	5	91	79
	BSC	2.36M	41.22M	5	550	524
	Polygon	15K	32.80K	3	11	7

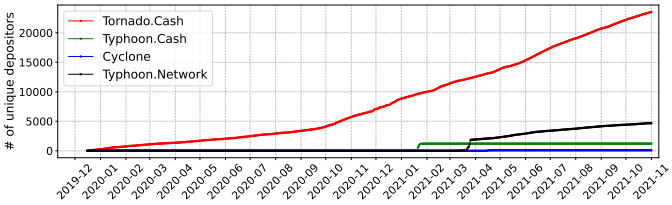


Fig. 1: Number of unique mixer depositors over time. 23,529 (79.75%) and 4,842 (16.41%) depositors appear in Tornado.Cash and Typhoon.Network, respectively.

B. Mixing Services for DeFi

Mixing services allow users to mix their coins with other users in an effort to break the linkability of addresses (i.e., whether two addresses belong to the same entity). The literature features various proposals of mixing service designs, which can be centralized [22], [47], [26], [43] or governed by smart contracts [6], [30], [9], [7].

As DeFi adoption increases and all transactions, balances, senders and recipients are public, the demand for privacy in the DeFi ecosystem has led to the launch of ZKP mixers. To date, the largest mixing service on Ethereum is Tornado.Cash [6]. TC is an autonomous and decentralized ZKP mixer, launched in December 2019. TC operates four `ETH` pools (i.e., 0.1, 1, 10 and 100 `ETH` pool) which support the deposit and withdrawal of a fixed amount of `ETH`. When a user deposits a fixed amount of `ETH` into a TC pool, the user should safely backup a deposit *note*; to withdraw, the user should provide the deposit *note*, which needs to be verified by the TC smart contract. TC also supports the mixing of other tokens (e.g., `DAI`, `USDC`, `USDT`, etc.), but most users appear to be mixing `ETH`. The total `ETH` deposited in TC reached over 2.05M `ETH` (9.11B USD)[1] at the time of writing.

AMR [30] is a new mixer design similar to TC, but additionally rewards its users for their participation in the system. Such incentivization of paying rewards is similar to the currently popular liquidity mining, also called "DeFi farming",

[1]We adopt the prices of coins on CoinMarketCap on November 1st, 2021 to convert them to USD, e.g., 1 `ETH` = 4,300 USD, 1 `BNB` = 530 USD.

TABLE II: System Model Definitions

Name	Definition	Eq.
coin transfer	$\mathsf{tr} = (\mathtt{bn}, \mathtt{from}, \mathtt{to}, \mathsf{amt}, \mathtt{coin})$ where $\mathtt{from} \xrightarrow{\mathtt{coin}} \mathtt{to}$	(1)
coin flow	$\mathcal{F} = (\mathsf{tr}_1, ..., \mathsf{tr}_n)$ where $\mathsf{tr}_{i-1}.\mathtt{to} = \mathsf{tr}_i.\mathtt{from}$ and $\mathsf{tr}_{i-1}.\mathtt{bn} \leq \mathsf{tr}_i.\mathtt{bn}$	(2)
Link	$\textsc{Link}(\mathsf{a}_1, \mathsf{a}_2) = 1 \Leftrightarrow \mathsf{a}_1$ is linked to a_2	(3)
Cluster	$\mathcal{C} = \{\mathsf{a}_1, .., \mathsf{a}_n\}$, $\forall \mathsf{a}_i \in \mathcal{C}$, $\exists \mathsf{a}_j \in \mathcal{C} \setminus \{\mathsf{a}_i\}$, satisfies $\textsc{Link}(\mathsf{a}_i, \mathsf{a}_j) = 1$	(4)
Mixer Pool	$\mathbf{P}$	(5)
Fixed currency denomination	p of coin	(6)
Pool depositors	$\mathcal{D}_{\mathbf{P}}(t) = \{\mathbf{d} \mid \mathbf{d}$ deposits coin into $\mathbf{P}$ before $t\}$	(7)
Pool withdrawers	$\mathcal{W}_{\mathbf{P}}(t) = \{\mathbf{w} \mid \mathbf{w}$ withdraws coin from $\mathbf{P}$ before $t\}$	(8)
Address Balance	$\mathbf{bal}_{\mathsf{a}}(t) = \mathbf{u}_{\mathsf{a}}(t) \times p - \mathbf{v}_{\mathsf{a}}(t) \times p$, where $\mathbf{u}_{\mathsf{a}}(t)$ and $\mathbf{v}_{\mathsf{a}}(t)$ are the numbers of a's deposit and withdrawal, respectively.	(9)
Pool State	$\mathbb{S}_{\mathbf{P}}(t) = \{(\mathsf{a}, \mathbf{bal}_{\mathsf{a}}(t)) \mid \mathsf{a} \in \mathcal{D}_{\mathbf{P}}(t) \cup \mathcal{W}_{\mathbf{P}}(t)\}$	(10)
Merge	$\textsc{Merge}(\mathbb{S}_{\mathbf{P}}(t), (\mathsf{a}_1, \mathsf{a}_2)) = \{(\mathsf{a}, \mathbf{bal}_{\mathsf{a}}(t)) \mid \mathsf{a} \in \mathcal{D}_{\mathbf{P}}(t) \cup \mathcal{W}_{\mathbf{P}}(t) \wedge \mathsf{a} \neq \mathsf{a}_1 \wedge \mathsf{a} \neq \mathsf{a}_2\} \cup \{(\mathsf{a}_1, \mathbf{bal}_{\mathsf{a}_1}(t) + \mathbf{bal}_{\mathsf{a}_2}(t))\}$	(11)
Simplified Pool State	$\textsc{Simp}(\mathbb{S}_{\mathbf{P}}(t), \mathcal{S}) = \textsc{Simp}\left(\textsc{Merge}(\mathbb{S}_{\mathbf{P}}(t), (\mathsf{a}_i, \mathsf{a}_{i+1})), \mathcal{S}'\right)$ where S is a set of linked addresses and $\mathcal{S}' = \mathcal{S} \setminus \{(\mathsf{a}_i, \mathsf{a}_{i+1})\}$	(12)
Depositors Extension	$\mathcal{D}_{\mathbf{P}}^{(n)}(t) = \{\mathsf{a} \mid \exists \mathsf{a}_1 \in \mathcal{D}_{\mathbf{P}}^{(n-1)}(t), \mathsf{a} \xrightarrow{\mathtt{coin}} \mathsf{a}_1$ before $t\}$	(13)
Withdrawers Extension	$\mathcal{W}_{\mathbf{P}}^{(n)}(t) = \{\mathsf{a} \mid \exists \mathsf{a}_1 \in \mathcal{W}_{\mathbf{P}}^{(n-1)}(t), \mathsf{a}_1 \xrightarrow{\mathtt{coin}} \mathsf{a}$ before $t\}$	(14)
Observed Anonymity Set	$\mathbf{OAS}_{\mathbf{P}}(t) = \mathcal{D}_{\mathbf{P}}(t)$	(15)
True Anonymity Set	$\mathbf{TAS}_{\mathbf{P}}(t) = \{\mathsf{a} \mid (\mathsf{a}, \mathbf{bal}_{\mathsf{a}}(t)) \in \mathbb{S}_{\mathbf{P}}(t) \wedge \mathbf{bal}_{\mathsf{a}}(t) > 0\}$	(16)
Simplified Anonymity Set	$\mathbf{SAS}_{\mathbf{P}}(t)$	(17)

an attempt to attract more users. More users should translate to a larger anonymity set size, as AMR proclaims. Blender [9] appears to be an instance of an AMR mixer by depositing the users' ETH into Aave (a DeFi lending platform) [13] and then redistributing the interests earned to users. Soon after AMR, TC was updated in December 2020 to support *anonymity mining* [44]. Anonymity mining incentivizes users to keep their deposited ETH in mixer pools for a longer time period. A user can receive rewards through a "shielded liquidity mining protocol" (cf. Section VI).

ZKP mixers such as TC can be implemented on smart contract-enabled blockchains, e.g., Typhoon.Cash (TP) [7] on Ethereum, Typhoon.Network [8] and Cyclone [1] on BSC and Polygon. Table I and Figure 1 provide a comparison of ZKP mixers to date. We observe that TC accumulates the largest total deposited USD and the number of depositors.

III. System and Threat Model

In this section, we outline our system and threat model.

A. System Model

Users have at least one public/private key-pair (corresponding to their *address*), which controls cryptocurrency assets on a permissionless blockchain. To transfer or trade an asset, the user signs a transaction with its private key. Each transaction corresponds to an event with various publicly readable features, such as the time of day and the transaction fees.

We summarize in Table II the definitions of this work to further describe the system model: *(1)* the "movement" of cryptocurrency assets between addresses (Eq. 1–2), *(2)* the linkability of addresses (Eq. 3–4) and *(3)* the notations of a mixer pool (Eq. 5–16).

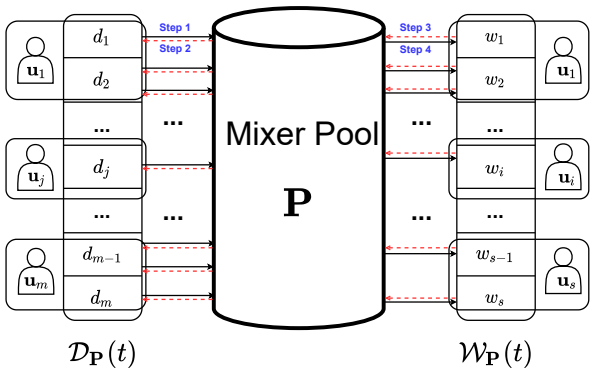


Fig. 2: System Model, where $\mathcal{D}_{\mathbf{P}}(t) = \{d_1, ..., d_m\}$ and $\mathcal{W}_{\mathbf{P}}(t) = \{w_1, ..., w_s\}$. '$\longrightarrow$' represents a transfer of coin, and '$\dashleftarrow$' represents a transfer of *note*. When a user $\mathbf{u}$ deposits coin into pool $\mathbf{P}$ (Step 1), $\mathbf{u}$ receives a *note* from $\mathbf{P}$ (Step 2); To withdraw, $\mathbf{u}$ needs to provide *note* to $\mathbf{P}$ (Step 3) and will receive coin after $\mathbf{P}$ verifies *note* (Step 4). A user $\mathbf{u}$ can control multiple addresses. An address can be used to deposit or withdraw multiple times.

Definition 1: **(Transfer of coin – Eq. 1)** A transfer of coin is a tuple $\mathsf{tr} = (\mathtt{bn}, \mathtt{from}, \mathtt{to}, \mathtt{amt}, \mathtt{coin})$, where bn is the block number (i.e., timestamp), amt is the amount of coin that is transferred from the address from to to.

Definition 2: **(Flow of coin – Eq. 2)** A flow of coin is a chain of transfers of coin between addresses.

Definition 3: **(Link – Eq. 3)** If two addresses a_1 and a_2 belong to the same user, then a_1 and a_2 are linked.

Definition 4: **(Cluster – Eq. 4)** A cluster is a set of mutually-linked addresses.

Definition 5: **(Mixer Pool – Eq. 5–8)** A mixer pool is an aggregation of cryptocurrency assets governed by a smart contract (cf. Figure 2). Users can only deposit and withdraw a specific cryptocurrency coin. To avoid that deposit/withdrawal asset amounts leak privacy, mixer pools typically only accept a fixed currency denomination. A *depositor* is an address to deposit coin into $\mathbf{P}$, and a *withdrawer* is an address to receive coin from $\mathbf{P}$.

The proper use of a mixer pool $\mathbf{P}$, requires choosing one address $\mathsf{a}_{\mathcal{D}}$ to deposit and another ideally unlinkable address $\mathsf{a}_{\mathcal{W}}$ to withdraw.

Definition 6: **(Address Balance – Eq. 9)** The amount of coin that an address holds in a pool at a time t.

Definition 7: **(Pool State – Eq. 10)** The set of tuples constituted by all depositors, withdrawers, and their balances in $\mathbf{P}$, at time t.

A pool $\mathbf{P}$'s state is determined by users' balances. For instance, if d_1 deposits once, d_2 deposits twice, and w_1 withdraws once in a 100 coin pool $\mathbf{P}_{100}$ before time t, then $\mathbf{P}_{100}$'s pool state is $\mathbb{S}_{\mathbf{P}_{100}}(t) = \{(d_1, 100), (d_2, 200), (w_1, -100)\}$.

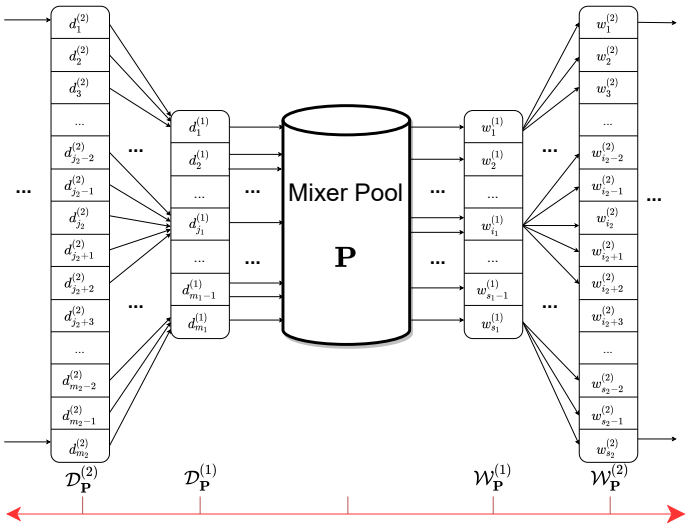


Fig. 3: Extended model of a mixer pool. '$\longrightarrow$' represents a transfer of coin. m_n is the size of the distance-n depositors set $\mathcal{D}_{\mathbf{P}}^{(n)}$, and s_n is the size of the distance-n withdrawers set $\mathcal{W}_{\mathbf{P}}^{(n)}$. $d_{j_n}^{(n)}$ is a depositor in $\mathcal{D}_{\mathbf{P}}^{(n)}$ and $j_n \in [1, m_n]$. $w_{i_n}^{(n)}$ is a withdrawer in $\mathcal{W}_{\mathbf{P}}^{(n)}$ and $i_n \in [1, s_n]$. Here we suppose the time t is fixed and omit it for simplicity.

If there exists a link between a depositor and a withdrawer in a pool $\mathbf{P}$, we can simplify the pools' state (cf. Definition 8). For instance, in the previous example $\mathbf{P}_{100}$, if $\textsc{Link}(d_1, w_1) = 1$, then we can simplify the state as $\textsc{Simp}\left(\mathbb{S}_{\mathbf{P}_{100}}(t), (d_1, w_1)\right) = \{(d_2, 200)\}$.

Definition 8: **(Simplified Pool State – Eq. 12)** Given a pool's state $\mathbb{S}_{\mathbf{P}}$ and a set $\mathcal{S}$ consisting of linked address pairs, we compute the Simplified Pool State by merging the balances of linked addresses.

To help track a user's transfers of coin before and after the user interacts with a mixer pool, we extend the depositor and withdrawer set as follows.

Definition 9: **(Depositors Extension – Eq. 13)** At time t, we let $\mathcal{D}_{\mathbf{P}}(t) = \mathcal{D}_{\mathbf{P}}^{(1)}(t)$ and define the depositors in distance $n (n > 1)$, $\mathcal{D}_{\mathbf{P}}^{(n)}(t)$, as the set of addresses that transfer coin to the addresses in $\mathcal{D}_{\mathbf{P}}^{(n-1)}(t)$.

Definition 10: **(Withdrawers Extension – Eq. 14)** At time t, we let $\mathcal{W}_{\mathbf{P}}(t) = \mathcal{W}_{\mathbf{P}}^{(1)}(t)$ and define the withdrawers in distance $n (n > 1)$, $\mathcal{W}_{\mathbf{P}}^{(n)}(t)$, as the set of addresses that receive coin from the addresses in $\mathcal{W}_{\mathbf{P}}^{(n-1)}(t)$.

Note that based on Definitions 9 and 10, the mixer pool model in Figure 2 can be extended to a model in Figure 3, to cover depositors and withdrawers in longer distances.

B. Privacy Mechanisms

Knowing the set of depositors $\mathcal{D}_{\mathbf{P}}(t)$ and the state of a pool $\mathbf{P}$ at time t, we define the *observed* anonymity set and the *true* anonymity set of the pool $\mathbf{P}$.

Definition 11: **(Observed Anonymity Set – Eq. 15)** At time t, the observed anonymity set $\mathbf{OAS}_{\mathbf{P}}(t)$ of a pool $\mathbf{P}$ is the set of unique addresses used to deposit, i.e., $\mathcal{D}_{\mathbf{P}}(t)$.

Definition 12: **(True Anonymity Set – Eq. 16)** At time t, the true anonymity set $\mathbf{TAS}_{\mathbf{P}}(t)$ of a pool $\mathbf{P}$ is the set of addresses with a positive balance, i.e., the set of depositors that are still in the pool $\mathbf{P}$.

Note that the true anonymity set might not be apparent from observing the blockchain data, because it is the mixer's intention to obfuscate the addresses with positive mixer balances. However, an adversary can leverage on-chain data to compute a more "realistic" anonymity set (cf. Def 13), which can be more representative than $\mathbf{OAS}_{\mathbf{P}}(t)$.

Definition 13: **(Simplified Anonymity Set – Eq. 17)** Given a mixer pool $\mathbf{P}$ at time t, the simplified anonymity set $\mathbf{SAS}_{\mathbf{P}}(t)$ is the set of depositors with a positive balance, which is computed by leveraging on-chain data to simply the pool state. Note that $\mathbf{SAS}_{\mathbf{P}}(t) \subseteq \mathbf{OAS}_{\mathbf{P}}(t)$.

Privacy Metric. The probability that an adversary without prior knowledge links a withdrawer (who withdraws at time t) to the correct depositor is:

$$\mathsf{Adv}_{\mathcal{A}}^{o}(t) = 1/\left|\mathbf{OAS}_{\mathbf{P}}(t)\right| \tag{18}$$

If we assume that the adversary can link a withdrawer w (who withdraws at time t), to a target set of depositors $\mathbf{SAS}_{\mathbf{P}}(t)$, then the probability that the adversary links w to the correct depositor is:

$$\mathsf{Adv}_{\mathcal{A}}^{s}(t) = 1/\left|\mathbf{SAS}_{\mathbf{P}}(t)\right| \tag{19}$$

Using Eq. 18 and Eq 19, we define $\mathsf{R}_{\mathsf{Adv}}$ as the increase of $\mathsf{Adv}_{\mathcal{A}}^{s}(t)$ over $\mathsf{Adv}_{\mathcal{A}}^{o}(t)$, to represent the *relative increase* of the probability that an adversary links a withdrawer to the correct depositor:

$$\mathsf{R}_{\mathsf{Adv}} = \frac{\mathsf{Adv}_{\mathcal{A}}^{s}(t) - \mathsf{Adv}_{\mathcal{A}}^{o}(t)}{\mathsf{Adv}_{\mathcal{A}}^{o}(t)} \tag{20}$$

C. Threat Model: Public L1 Mixer Attacker

In this work, the goal of the adversary is to link the deposit addresses of the same user or entity, with the true withdrawal address(es) of the mixer. Hence, the privacy of a user is quantified with the ability of the adversary to successfully *link* a deposit address with a withdrawal address. We assume that the adversary possesses the following prior knowledge:

Mixer Functionality: We assume that the adversary is fully aware of how the mixer operates, and has access to the source code of the involved smart contract.

Transparent Mixer Input and Output: We assume that the adversary can record the input and output of the mixer, e.g., has access to all the deposit and withdrawal transactions, their addresses, timestamps, etc.

User Trading Activity: We further assume that the adversary has access to the full transaction record of users in DeFi, amounting to the transaction sender, recipient, amounts, involved smart contracts, and DeFi platforms.

TABLE III: Number of deposits and withdrawals in four TC ETH and TN BNB pools on November 1st, 2021.

Pool	# Deposits	# Withdrawals	# Depositors	# Withdrawers
TC 0.1 ETH	17,075	13,821	6,939	7,499
TC 1 ETH	30,290	24,774	9,733	12,189
TC 10 ETH	24,678	23,012	8,865	11,286
TC 100 ETH	18,479	16,718	4,108	5,762
TN 0.1 BNB	6,400	5,947	2,850	3,225
TN 1 BNB	8,417	8,226	2,618	3,064
TN 10 BNB	3,729	3,699	1,265	1,505
TN 50 BNB	417	410	156	203

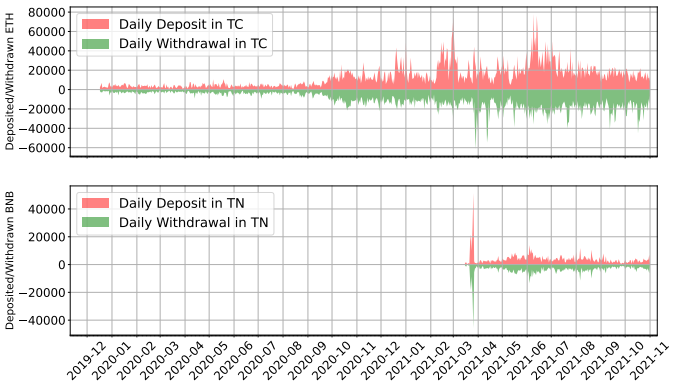


Fig. 4: Daily activity in TC ETH and TN BNB pools.

We further assume that the adversary cannot be a relayer, and cannot monitor the network layer to record depositors' and withdrawers' IP addresses.

Note that a public L1 mixer attacker model is expressly different from an attacker in the context of a permissioned chain or a private chain, or an L2 protocol with data availability limits, such as StarkWare [12]. Note also that the attack model for L2s such as StarkWare, where validators or availability committee members get to see more of execution is different from what is presented here. However, our attacker model can also be applied to an L2 protocol without data availability limits, e.g., Rollups [5].

IV. Empirical Mixer Activity

To gather empirical insights into the activities of existing DeFi mixers, we crawl the deposit, withdrawal events and transactions of the 73 pools on four ZKP mixers: Tornado.Cash, Typhoon.Cash, Typhoon.Network and Cyclone, from December 16th, 2019 (i.e., the inception time of TC) to November 1st, 2021. Figure 1 shows the number of unique depositors in mixers over time. We observe that 96.16% of the mixer users deposit assets into TC and TN, and that the number of TP depositors did not change since February, 2021. Therefore, in the following, we focus on analyzing the two most active mixers, TC and TN.

We analyze the top four active pools in TC (0.1, 1, 10 and 100 ETH pools) and TN (0.1, 1, 10 and 50 BNB pools). For TC, we crawl the deposit and withdrawal events

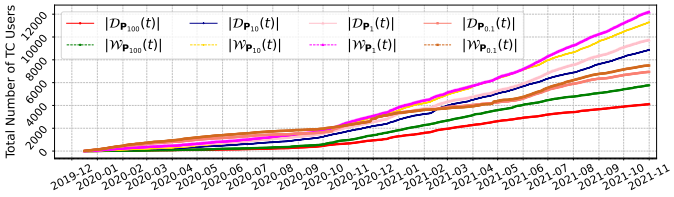


Fig. 5: The number of depositors and withdrawers in TC ETH pools over time. TC 0.1 ETH pool has the most depositors and withdrawers before November 1st, 2021.

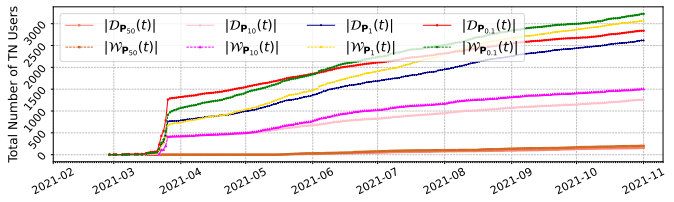


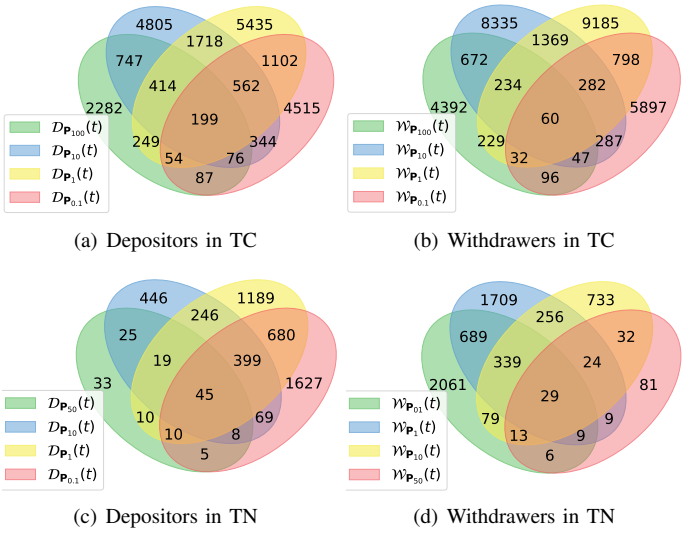
Fig. 6: The number of depositors and withdrawers in TN BNB pools over time.

data from the Ethereum block 9,116,966 (December 16th, 2019) to 13,530,000 (November 1st, 2021). The TC 1 ETH pool is the most active (30,290 deposits and 24,774 withdrawals), while the TC 100 ETH pool has the smallest depositor and withdrawer set (4,108 deposit and 5,762 withdraw addresses). The TC pools accumulate deposits of 2,126,677.5 ETH (9.12B USD). Moreover, from TN's inception at BSC block 5,230,899 (February 27th, 2021) until block 12,271,000 (November 1st, 2021), we find that 4,811 addresses generate 18,963 deposits in the four BNB pools, accumulating 67,197 BNB (35.26M USD).

A. Daily Activity in Mixer Pools

We plot the daily deposited and withdrawn ETH and BNB in TC and TN pools from December 16th, 2019 to November 1st, 2021 in Figure 4. We observe that the graphs of daily deposits and withdrawals seem to be approximately symmetrical.

Figure 5 shows the growth of the number of depositors $|\mathcal{D}_{\mathbf{P}}(t)|$ and withdrawers $|\mathcal{W}_{\mathbf{P}}(t)|$ in the four TC ETH pools over time. We notice that $|\mathcal{D}_{\mathbf{P}_{0.1}}(t)|$ is superior compared to $|\mathcal{D}_{\mathbf{P}_j}(t)|$, where $j \in \{1, 10, 100\}$, before the 1st of October 2020, but $|\mathcal{D}_{\mathbf{P}_j}(t)|$ increases faster than $|\mathcal{D}_{\mathbf{P}_{0.1}}(t)|$ after the 1st of October 2020. We speculate that this is because first-time TC users tend to try the 0.1 ETH pool for testing purposes before attempting other pools. We also observe that $|\mathcal{D}_{\mathbf{P}_p}(t)|$ and $|\mathcal{W}_{\mathbf{P}}(t)|$ grow approximately linearly before July 2020, and faster afterwards. We conjecture that this growth change is likely due to the announcement of the Tornado Fund Launch in June 2020 [45]. Additionally, for TN, we observe that the pools with the smaller denomination have more depositors and withdrawers (cf. Figure 6). Notably, this is similar to the early stages of TC, where users prefer to try the TN 0.1 BNB pool for testing purpose.



(a) Depositors in TC (b) Withdrawers in TC

(c) Depositors in TN (d) Withdrawers in TN

Fig. 7: Venn diagram of depositors and withdrawers in four TC ETH and TN BNB pools, given t on November 1st, 2021. 199 depositors and 60 withdrawers appear in the four TC pools. 45 depositors and 29 withdrawers appear in the four TN pools.

B. Depositors and Withdrawers

The four TC ETH pools contain $|\bigcup_p \mathcal{D}_{\mathbf{P}_p^{\text{ETH}}}(t)| = 22{,}589$ unique depositors and $|\bigcup_p \mathcal{W}_{\mathbf{P}_p^{\text{ETH}}}(t)| = 31{,}915$ unique withdrawers (given $p \in \{0.1, 1, 10, 100\}$), depositing 94.15 ETH (400k USD) and withdrawing 60.41 ETH (260k USD) on average. In each pool, the number of withdrawers is greater than depositors, indicating that a user may adopt multiple addresses to withdraw but fewer address to deposit.

Because a mixer pool only supports a fixed currency denomination, users may utilize multiple pools to mix arbitrary amounts of assets. Figure 7 shows that 199 depositors utilize all four TC pools, and 5,552 (24.58%) deposit in more than one pool. Additionally, 60 users withdraw from all four pools, and 4,106 (12.87%) use more than one pool to withdraw. Likewise, for TN, we observe a slight increase in overlaps on both depositors (31%) and withdrawers (24%) appearing in at least two pools. The overlap of pools may help an adversary to link addresses (cf. Section V-A).

C. Relayers

Relayers help users to withdraw coins from a pool towards a new address by paying for the transaction fees in the native blockchain currency (e.g., ETH and BNB). Relayers in exchange receive a share of the withdrawn coins [6]. As shown in Table IV, we identify 192 unique relayers, and 102 relayers operate on the four TC ETH pools. Furthermore, 68.21% of the withdrawal transactions are performed with the help of a relayer, and more than 92.47% of withdrawers use relayers. Contrarily, although TN aims to add more relayers [46], currently, it only uses a single relayer and substantially fewer users withdraw funds using a relayer (less than 60%).

TABLE IV: Relayers usage in TC ETH and TN BNB pools before November 1st, 2021.

Pool	# relayers	# withdrawals with relayer	# withdrawers using relayer
TC $\mathbf{P}_{0.1}$	139	9,362 (67.74%)	6,974 (93.00%)
TC $\mathbf{P}_{1}$	158	19,739 (79.68%)	11,751 (96.41%)
TC $\mathbf{P}_{10}$	152	20,276 (88.11%)	10,699 (94.80%)
TC $\mathbf{P}_{100}$	141	11,404 (68.21%)	5,328 (92.47%)
# total TC relayers			192
TN $\mathbf{P}_{0.1}$	1	3,438 (57.81%)	1,899 (56.30%)
TN $\mathbf{P}_{1}$	1	5,051 (61.40%)	1,921 (59.07%)
TN $\mathbf{P}_{10}$	1	1,027 (52.10%)	696 (44.30%)
TN $\mathbf{P}_{50}$	1	214 (52.20%)	107 (50.23%)
# total TN relayers			1

Note that when users withdraw funds without a relayer, they may incautiously adopt their deposit address to initiate the withdrawal transaction, revealing that the deposit and withdrawal addresses belong to the same user. Therefore, an adversary can link the addresses to reduce the $\mathbf{OAS}_{\mathbf{P}}(t)$ of the mixer (cf. Section V-A).

D. Coin Flow of ZKP Mixer Pools

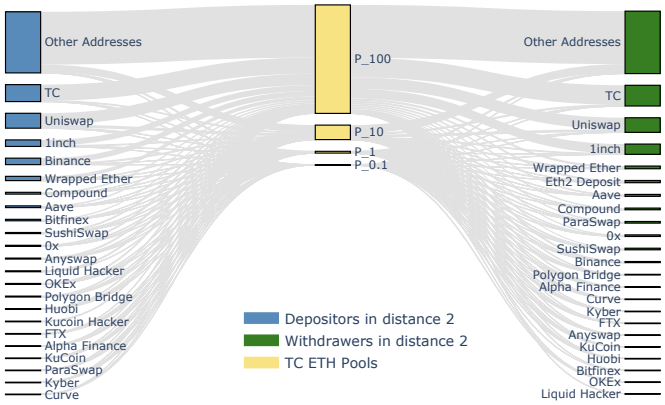


Fig. 8: Coin Flow of TC ETH Pools before block 13,530,000 (November 1st, 2021). The shown bandwidth of each flow represents the magnitude of the aggregate ETH transferred from depositors in distance 2 to the TC pools (via depositors in distance 1), or from TC pools to withdrawers in distance 2 (via withdrawers in distance 1).

In addition to immediate depositors and withdrawers, we are also interested in coins' wider flow to get their origins and destinations. For example, users move their coins from exchanges or DeFi platforms via temporary addresses into and outside of the mixer. Analyzing the flow of coins (cf. Def. 2) could thus reveal information about users' trading activities.

To track where the deposited ETH in TC are transferred from and where the withdrawn ETH are transferred to, we extend our pool model to cover depositors and withdrawers in distance 2 (cf. Def. 9 and 10). We crawl the direct and internal transaction[2] history of the 22,589 TC depositors and 31,915 withdrawers before block 13,530,000.

[2]An internal transaction is a transaction triggered by a smart contract as a result of one or more previous transactions.

For each depositor of $d^{(1)}$ in a TC p ETH pool, we extract *the most recent* transfers of p ETH that $d^{(1)}$ receives before depositing into TC, where we can obtain the depositors in distance 2 that transfer ETH to $d^{(1)}$. Similarly, we obtain the withdrawers in distance 2 by extracting *the most recent* transfers of p ETH that the withdrawers in distance 1 send after withdrawing from TC. Then, we tag the depositors and withdrawers in distance 2 using manually crawled labels from Etherscan. We finally cluster the addresses into different platforms based on their tags.

Figure 8 visualizes the flow of ETH via four TC pools, from the depositors in distance 2 to the withdrawers in distance 2. We observe that the top 10 clusters in distance 2 cover 45.13% of the total deposit volume, and transfers from Uniswap alone amount to 253,674 ETH (11.93% of the total deposit volume). Uniswap is also the most popular DeFi platform to which TC users transfer their withdrawn ETH; i.e., 11.71% of the total withdrawal volume are transferred to Uniswap. We also observe that 13.58% of the total deposit volume are re-deposited into TC.

Cross-chain Mixers Usage. Users on other blockchains (e.g., BSC, Polygon) can also leverage TC to enhance their privacy. To do so, they convert their funds to ETH through decentralized bridges (e.g., Anyswap [10]) and then deposit ETH into TC. Figure 8 shows that 0.79% (16,853.9 ETH) of the TC deposited volume comes from the Anyswap BSC bridge. We find that 61 addresses have transferred funds from BSC through this bridge and deposited ETH into TC. Those preliminary results indicate that due to the largest **OAS** (cf. Table I), TC can attract privacy-seeking users from other blockchains.

E. Malicious Addresses

Because mixers break the linkability between addresses, malicious actors may choose mixers to hide their traces. To gain initial insights into how malicious users adopt TC, we first crawl 6,340 phishing and hack-related addresses from Etherscan and CryptoScamDB[3]. We denote the 6,340 addresses as *malicious addresses* and analyze whether they appear in the set of TC depositors and withdrawers.

Malicious Addresses in Distance 1. We identify 47 malicious addresses, depositing 62,548.2 ETH (2.94%) into TC, and 36 malicious addresses withdrawing 851.7 ETH from the four TC pools. Through manual transaction inspection, we find that 18 addresses withdraw ETH from the TC pools, then launch an attack (e.g., phishing scams, hacking centralized exchanges, etc.), before re-depositing the resulting profit to the mixer (cf. Table V). For instance, the xToken Exploiter withdraws 10 ETH from TC as the source of funds to launch an attack while re-depositing 5,855 ETH into TC.

Malicious Addresses in Distance 2. After extending the coin flow of TC pools, we identify 40 malicious addresses in distance 2 (cf. Def. 9) from a TC pool, transferring 23,986.4 ETH to depositors in distance 1. We investigate two examples as follows: *(1)* the KuCoin Hacker transfers 11,520 ETH to an address 0x34a...c6b in distance 1, then this address deposits all funds into the TC 100 ETH pool. *(2)* the Abyss Hacker transfers 57.1 ETH to three addresses 0x28A...B3d, 0x030...75b and 0xAc8...71e, then the three addresses deposit funds into the TC 10 ETH pools. The three addresses likely belong to the Abyss Hacker.

[3]https://etherscan.io/accounts/label and https://cryptoscamdb.org/scams

TABLE V: Malicious addresses withdrawing initial funds from TC and re-depositing funds into TC.

Address	Labels/References	Total Deposit	Total Withdrawal
0x079...758	Compounder.Finance: Deployer	14,012.0 ETH	13.0 ETH
0x07E...c3e	xToken Exploiter	5,855.0 ETH	10.0 ETH
0xCB3...233	Rari Capital Exploiter	4,004.0 ETH	100.0 ETH
0x56E...628	AFKSystem	3,223.0 ETH	0.1 ETH
0x905...B57	Alpha Homora V2 Exploiter	2,320.0 ETH	10.0 ETH
0xb62...212	Furucombo Hacker	2,000.0 ETH	1.0 ETH
0x1D5...72B	Punk Protocol Exploiter	968.0 ETH	0.2 ETH
0x903...678	KORE Vault: Deployer	600.0 ETH	2.0 ETH
0x8b1...B5B	Fake_Phishing4583	348.4 ETH	10.0 ETH
0x8eD...597	Fake_Phishing4518	300.0 ETH	220.0 ETH
0x30e...83f	Fake_Phishing4166	200.0 ETH	1.0 ETH
0x5Eb...d5F	FinNexus Hacker	153.4 ETH	1.0 ETH
0xdA2...708	Fake_Phishing4946	120.0 ETH	100.0 ETH
0xeBc...C40	Warp.Finance Hacker	100.3 ETH	1.0 ETH
0xE29...4dD	Force Vault Hacker 3	34.3 ETH	15.3 ETH
0x24F...594	Fake_Phishing4640	30.0 ETH	1.0 ETH
0xcE7...3A2	Fake_Phishing4540	21.0 ETH	12.0 ETH
0xEda...113	ChainSwap Hacker	20.0 ETH	10.0 ETH

In total, we find that 87 malicious addresses deposit 86,534.6 ETH (372.1M USD) into TC, i.e., 4.1% of the total deposit volume. The average deposit volume of malicious addresses (i.e., 4.28M USD) is 10.7× larger than the average deposit volume of TC users (i.e., 400k USD).

V. Measuring Anonymity Set Size

In the following, we propose five heuristics to compute a mixer pool's anonymity set size ($\mathbf{SAS}_{\mathbf{P}}^{((1\ldots5))}(t)$), which is more representative than the naive $\mathbf{OAS}_{\mathbf{P}}(t)$. The heuristics leverage on-chain data and insights from our empirical study (Section IV) to link addresses and prune the $\mathbf{OAS}_{\mathbf{P}}(t)$. Note that our heuristics are best-effort methods and subject to known limitations (cf. Section V-D).

A. Heuristics

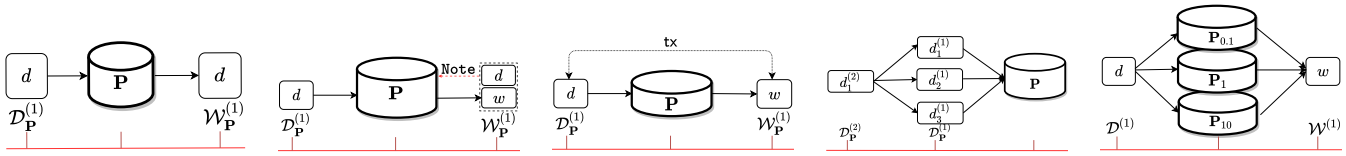
1) $\mathbf{H_1}$ *- Deposit Address Reuse:*

Observation: We observe that an address can be reused to both deposit and withdraw (cf. Section IV), which leaks privacy and is incautious behavior [18], [48].

Heuristic: If an address appears both in the depositor and the withdrawer set, we assume that the deposits and withdrawals of this address are conducted by the same user (cf. Figure 9(a)). Therefore, we apply Equation 9 to compute a depositor's balance. We then extract the depositors with a positive balance to evaluate the anonymity set (cf. Eq. 21).

$$\boxed{\mathbf{SAS}_{\mathbf{P}}^{(1)}(t) = \{\mathsf{a} \mid \mathsf{a} \in \mathcal{D}_{\mathbf{P}}(t) \wedge \mathbf{bal}_{\mathsf{a}}(t) > 0\}} \quad (21)$$

Results: On TC pools, Heuristic 1 reduces the anonymity set by an average of 11.92% from the reported $\mathbf{OAS}_{\mathbf{P}}(t)$ (cf. Table VI). For instance, in the TC 100 ETH pool



(a) Heuristic 1. (b) Heuristic 2. (c) Heuristic 3. (d) Heuristic 4. (e) Heuristic 5.

Fig. 9: Heuristics 1-5: *(a)* **Deposit Address Reuse:** User applies the same address d for deposit and withdrawal in $\mathbf{P}$.
(b) **Improper Withdrawal Sender:** A user adopts a deposit address d to provide a deposit note to $\mathbf{P}$ and adopts another address w to receive the withdrawn `coin`.
(c) **Related Deposit-Withdrawal Address Pair:** A user adopts two addresses d and w to deposit and withdraw in $\mathbf{P}$. d and w are different; however, w transfers (receives) tokens to (from) d through a transaction tx.
(d) **Intermediary Deposit Address:** An address $d_1^{(2)}$ in distance 2 controls 3 intermediary addresses $d_j^{(1)} (j = 1, 2, 3)$ in the distance 1, to deposit `coin` in $\mathbf{P}$.
(e) **Cross-pool Deposit:** A user adopts an address d to deposit 0.1 `coin` in $\mathbf{P}_{0.1}$, 1 `coin` in $\mathbf{P}_1$ and 10 `coin` in $\mathbf{P}_{10}$, and uses address w to withdraw the same amount of `coin` from $\mathbf{P}_{0.1}$, $\mathbf{P}_1$ and $\mathbf{P}_{10}$.

TABLE VI: **Heuristics 1-5** applied to TC `ETH` and TN `BNB` pools on November 1st, 2021. $|\mathbf{SAS}_{\mathbf{P}}^{(n)}(t)|$ represents the Anonymity Set Size after applying Heuristic n. The percentages in the parentheses show the difference between $\mathbf{SAS}_{\mathbf{P}}^{(n)}(t)$ and $\mathbf{OAS}_{\mathbf{P}}(t)$.

Pool	$\lvert\mathbf{OAS}_{\mathbf{P}}(t)\rvert$	$\lvert\mathbf{SAS}_{\mathbf{P}}^{(1)}(t)\rvert$	$\lvert\mathbf{SAS}_{\mathbf{P}}^{(2)}(t)\rvert$	$\lvert\mathbf{SAS}_{\mathbf{P}}^{(3)}(t)\rvert$	$\lvert\mathbf{SAS}_{\mathbf{P}}^{(4)}(t)\rvert$	$\lvert\mathbf{SAS}_{\mathbf{P}}^{(5)}(t)\rvert$
TC 0.1 `ETH`	6,939	6,108 (−11.98%)	6,914 (−0.36%)	5,130 (−26.07%)	6,610 (−4.74%)	6,884 (−0.79%)
TC 1 `ETH`	9,733	8,850 (−9.07%)	9,695 (−0.39%)	7,771 (−20.16%)	9,284 (−4.61%)	9,658 (−0.77%)
TC 10 `ETH`	8,865	7,868 (−11.25%)	8,843 (−0.25%)	7,397 (−16.56%)	8,430 (−4.91%)	8,789 (−0.86%)
TC 100 `ETH`	4,108	3,476 (−15.38%)	4,087 (−0.51%)	3,419 (−16.77%)	3,866 (−5.89%)	4,060 (−1.17%)
TN 0.1 `BNB`	2,843	1,840 (−35.28%)	2,592 (−8.83%)	1,581 (−44, 39%)	2,842 (−0.04%)	2,816 (−0.95%)
TN 1 `BNB`	2,598	1,719 (−33.83%)	2,357 (−9.28%)	1,428 (−45.03%)	2,591 (−0.27%)	2,566 (−1.23%)
TN 10 `BNB`	1,257	777 (−38.19%)	1,190 (−5.33%)	705 (−43.91%)	1,256 (−0.08%)	1,227 (−2.39%)
TN 50 `BNB`	155	147 (−5.16%)	147 (−5.16%)	136 (−12.26%)	155 (0%)	142 (−8.39%)

$\mathbf{P}_{100}$, there are 4,108 unique depositors, but only 3,476 depositors have a positive balance, and therefore contribute to the anonymity set. Consequently, $\mathbf{SAS}_{\mathbf{P}}^{(1)}(t)$ of $\mathbf{P}_{100}$ is 15.38% less than the respective $\mathbf{OAS}_{\mathbf{P}}(t)$ (cf. Table VI). On TN pools, we find that on average 71.89% of the depositors have a positive balance. Henceforth, $\mathbf{SAS}_{\mathbf{P}}^{(1)}(t)$ of TN pools is reduced by an average of 28.11% from $\mathbf{OAS}_{\mathbf{P}}(t)$.

2) $\mathbf{H_2}$ *- Improper Withdrawal Sender:*

Observation: Incautious users maybe adopt a deposit address a_d to issue a withdrawal transaction, while assigning another address a_w to receive the withdrawn funds (cf. Section IV-C). This action infers that a_d and a_w are likely controlled by the same user.

Heuristic: Therefore, we assume that given a depositor-withdrawer pair $(\mathsf{a}_d, \mathsf{a}_w)$ in a pool, where a_d is not a relayer, if a_d generates a withdrawal and assigns a_w to receive the withdrawn coins, then a_d and a_w belong to the same user (cf. Figure 9(b)), i.e., $\textsc{Link}(\mathsf{a}_d, \mathsf{a}_w) = 1$.

Let $\mathcal{S}_{\mathbf{P}}^{\mathrm{nt}}(t)$ be the set of linked address pairs in a pool $\mathbf{P}$. Given $\mathcal{S}_{\mathbf{P}}^{\mathrm{nt}}(t)$, we merge the balance of the linked addresses to simplify the pool state, and then compute the anonymity set (cf. Eq. 22).

$$\mathbf{SAS}_{\mathbf{P}}^{(2)}(t) = \{\mathsf{a} \mid \mathbf{bal}_{\mathsf{a}}(t) > 0 \wedge (\mathsf{a}, \mathbf{bal}_{\mathsf{a}}(t)) \in \textsc{Simp}(\mathbb{S}_{\mathbf{P}}(t), \mathcal{S}_{\mathbf{P}}^{\mathrm{nt}}(t))\} \tag{22}$$

Results: Heuristic 2 reduces the TC pools' $\mathbf{OAS}_{\mathbf{P}}(t)$ by an average of 0.38% (cf. Table VI). For instance, in the TC 100 `ETH` pool, there are 37 depositor-withdrawer pairs; meaning Heuristic 2 reduces the $\mathbf{OAS}_{\mathbf{P}}(t)$ to 4,087. Interestingly, we observe that there are more users (i.e., 766) in TN who do not perform withdrawals correctly. By applying Heuristic 2 to the TN pools, we reduce the $\mathbf{OAS}_{\mathbf{P}}(t)$ by 7.15% on average.

3) $\mathbf{H_3}$ *- Related Deposit-Withdrawal Address Pair:*

Observation: To withdraw coins, users are encouraged to choose a new address with no links to the deposit address. However, we observe that, users may adopt different deposit and withdrawal addresses, which are directly linked through a coin transfer.

Heuristic: We assume that, given two addresses $\mathsf{a}_d \in \mathcal{D}_{\mathbf{P}}(t)$ and $\mathsf{a}_w \in \mathcal{W}_{\mathbf{P}}(t)$, if a_d transferred (received) coins or tokens to (from) a_w before time t, then a_d and a_w are related and under the control of the same user (cf. Figure 9(c)), i.e., $\textsc{Link}(\mathsf{a}_d, \mathsf{a}_w) = 1$.

Let $\mathcal{S}^{\mathtt{tx}}_{\mathbf{P}}(t)$ be the set of related depositor-withdrawer pairs in a pool $\mathbf{P}$. We simplify the pool state and compute the anonymity set (cf. Eq. 23).

$$\mathbf{SAS}^{(3)}_{\mathbf{P}}(t) = \{\mathsf{a} \mid \mathbf{bal}_{\mathsf{a}}(t) > 0 \wedge (\mathsf{a}, \mathbf{bal}_{\mathsf{a}}(t)) \in \textsc{Simp}(\mathbb{S}_{\mathbf{P}}(t), \mathcal{S}^{\mathtt{tx}}_{\mathbf{P}}(t))\} \quad (23)$$

Results: Heuristic 3 reduces the TC pools' $\mathbf{OAS}_{\mathbf{P}}(t)$ by an average of 19.89%. For instance, in the TC 100 ETH pool, there are 1,607 depositor-withdrawer linked address pairs. Hence, $\mathbf{H}_3$ reduces the $\mathbf{OAS}_{\mathbf{P}}(t)$ to 3,419 (i.e., reduced by 16.77%). By applying $\mathbf{H}_3$ to the TN pools, we reduce the $\mathbf{OAS}_{\mathbf{P}}(t)$ by 36.40% on average.

4) $\mathbf{H_4}$ *- Intermediary Deposit Address:*

Observation: From the flow of coins, we observe that there are multiple depositors in distance 1 whose coins are all transferred from the same depositor in distance 2. Hence, these depositors in distance 1 are likely temporary addresses and are only used to transfer funds into a mixer (cf. Figure 9(d)).

Heuristic: We hence assume that given two addresses $d^{(1)} \in \mathcal{D}^{(1)}_{\mathbf{P}}(t)$ and $d^{(2)} \in \mathcal{D}^{(2)}_{\mathbf{P}}(t)$, if all $d^{(1)}$'s coins are transferred from $d^{(2)}$ and $d^{(2)}$ is a user account[4], then $d^{(1)}$ and $d^{(2)}$ belong to the same user, i.e., $\textsc{Link}(d^{(1)}, d^{(2)}) = 1$.

We denote $d^{(1)}$ as an *intermediary deposit address*, $\mathcal{B}^{(1)}_{\mathbf{P}}(t)$ as the set of intermediary deposit address, and $\mathcal{B}^{(2)}_{\mathbf{P}}(t)$ as the set of user accounts in distance 2 who transfer coins to an address in $\mathcal{B}^{(1)}_{\mathbf{P}}(t)$. For each address $d^{(1)}$ in $\mathcal{B}^{(1)}_{\mathbf{P}}(t)$, we replace it by the address in $\mathcal{B}^{(2)}_{\mathbf{P}}(t)$ which transfers coins to $d^{(1)}$. We then compute the pool's anonymity set (cf. Eq. 24).

$$\mathbf{SAS}^{(4)}_{\mathbf{P}}(t) = \{\mathsf{a} \mid \mathbf{bal}_{\mathsf{a}}(t) > 0 \wedge \mathsf{a} \in \mathcal{B}^{(2)}_{\mathbf{P}}(t) \cup \mathcal{D}^{(1)}_{\mathbf{P}}(t) \setminus \mathcal{B}^{(1)}_{\mathbf{P}}(t)\} \quad (24)$$

Results: Heuristic 4 reduces the TC pools' $\mathbf{OAS}_{\mathbf{P}}(t)$ by 5.04% on average (cf. Table VI). For instance, 233 intermediary deposit addresses are controlled by 53 user accounts in $\mathcal{B}^{(2)}_{\mathbf{P}_{100}}(t)$. Hence, $\mathbf{H}_4$ reduces the $\mathbf{OAS}_{\mathbf{P}}(t)$ to 3,866 (i.e., reduced by 5.89%). However, only 14 intermediary deposit addresses appear in the TN BNB pools. Hence, the TN pools' $\mathbf{OAS}_{\mathbf{P}}(t)$ can only be reduced by 0.1% on average.

5) $\mathbf{H_5}$ *- Cross-pool Deposit:*

Observation: Current mixer pools only support the deposit and withdrawal of a *fixed* coin denomination. When a user aims to mix an *arbitrary* amount of coins, the user needs to interact with multiple pools and may not change the respective deposit (or withdrawal) address (cf. Figure 9(e)).

Heuristic: Given a depositor-withdrawer pair $(\mathsf{a}_d, \mathsf{a}_w)$, we assume that a_d and a_w belong to the same user if: *(i)* a_d and a_w are both in m pools where $m > 1$, *(ii)* in each pool, a_d's total deposit amount equals a_w's total withdrawal amount, and *(iii)* for each a_w's withdrawal transaction $\mathtt{tx}^w$, at least one deposit transaction $\mathtt{tx}^d$ is generated earlier than $\mathtt{tx}^w$.

Let $\mathcal{S}^{\mathtt{cu}}$ be the set of address pairs $(\mathsf{a}_d, \mathsf{a}_w)$ that satisfy the above conditions. Given $\mathcal{S}^{\mathtt{cu}}$, we simplify the state of a pool $\mathbf{P}$, and then compute the anonymity set (cf. Eq. 25).

$$\mathbf{SAS}^{(5)}_{\mathbf{P}}(t) = \{\mathsf{a} \mid \mathbf{bal}_{\mathsf{a}}(t) > 0 \wedge (\mathsf{a}, \mathbf{bal}_{\mathsf{a}}(t)) \in \textsc{Simp}(\mathbb{S}_{\mathbf{P}}(t), \mathcal{S}^{\mathtt{cu}}_{\mathbf{P}}(t))\} \quad (25)$$

Results: Heuristic 5 reduces the TC pools' $\mathbf{OAS}_{\mathbf{P}}(t)$ by 0.90% on average. For instance, in the TC 100 ETH pool, 24 depositor-withdrawer address pairs are linked. Hence, $\mathbf{H}_5$ reduces the $\mathbf{OAS}_{\mathbf{P}}(t)$ to 4,060 (i.e., reduced by 1.17%). By applying $\mathbf{H}_5$ to the TN pools, we reduce the $\mathbf{OAS}_{\mathbf{P}}(t)$ by 3.24% on average.

TABLE VII: Heuristic Combinations applied to TC ETH and TN BNB pools on November 1st, 2021. The percentages show the reduction on the Observed Anonymity Set size $|\mathbf{OAS}_{\mathbf{P}}(t)|$ after applying heuristics.

Pool	Heuristic Combinations			
	$\mathbf{H_1 + H_2}$	$\mathbf{H_1 + H_2 + H_3}$	$\mathbf{H_1 + H_2 + H_3 + H_4}$	$\mathbf{H_1 + H_2 + H_3 + H_4 + H_5}$
TC $\mathbf{P}^{\mathrm{ETH}}_{0.1}$	−12.28%	−36.72%	−40.01%	−40.58%
TC $\mathbf{P}^{\mathrm{ETH}}_{1}$	−9.47%	−28.03%	−31.39%	−31.99%
TC $\mathbf{P}^{\mathrm{ETH}}_{10}$	−11.52%	−25.79%	−29.75%	−30.37%
TC $\mathbf{P}^{\mathrm{ETH}}_{100}$	−15.90%	−28.33%	−32.94%	−33.79%
TN $\mathbf{P}^{\mathrm{BNB}}_{0.1}$	−41.54%	−63.45%	−63.52%	−64.16%
TN $\mathbf{P}^{\mathrm{BNB}}_{1}$	−37.49%	−59.31%	−59.64%	−60.55%
TN $\mathbf{P}^{\mathrm{BNB}}_{10}$	−40.73%	−55.29%	−55.45%	−57.12%
TN $\mathbf{P}^{\mathrm{BNB}}_{50}$	−10.32%	−18.71%	−18.71%	−26.45%

Heuristic Combinations. Table VI shows the $\mathbf{SAS}_{\mathbf{P}}(t)$ of mixer pools after applying each heuristic individually, while we can further reduce the $\mathbf{OAS}_{\mathbf{P}}(t)$ by combining two or more heuristics (cf. Table VII). Combining all heuristics yields the largest reduction of $\mathbf{OAS}_{\mathbf{P}}(t)$: after applying Heuristics 1-5 to the TC (TN) pools, an adversary can reduce the the reported $\mathbf{OAS}_{\mathbf{P}}(t)$ on average by 34.18% (52.07%). Therefore, the probability that an adversary links a withdrawer (who withdraws at time t) to the correct depositor rises by 51.94% (108.63%) on average (cf. Eq. 20 and Eq. 26).

$$\begin{aligned} \mathsf{R}_{\mathsf{Adv}} &= \frac{\mathsf{Adv}^{s}_{\mathcal{A}}(t) - \mathsf{Adv}^{o}_{\mathcal{A}}(t)}{\mathsf{Adv}^{o}_{\mathcal{A}}(t)} \\ &= \frac{1/|\mathbf{SAS}_{\mathbf{P}}(t)| - 1/|\mathbf{OAS}_{\mathbf{P}}(t)|}{1/|\mathbf{OAS}_{\mathbf{P}}(t)|} = \frac{1}{\frac{|\mathbf{SAS}_{\mathbf{P}}(t)|}{|\mathbf{OAS}_{\mathbf{P}}(t)|}} - 1 \\ &= \frac{1}{1 - 34.18\%\,(52.07\%)} - 1 = 51.94\%\,(108.63\%) \end{aligned} \quad (26)$$

B. Linking Results

Through Heuristics 2-5, we can link 20,695 TC and 9,127 TN address pairs, which form 4,931 and 1,345 clusters (cf.

[4]We denote a user account as an Externally Owned Account which is not a labeled exchange address, i.e., we exclude contract addresses and exchange addresses that are labeled on Etherscan and Bscsan, such as Binance.

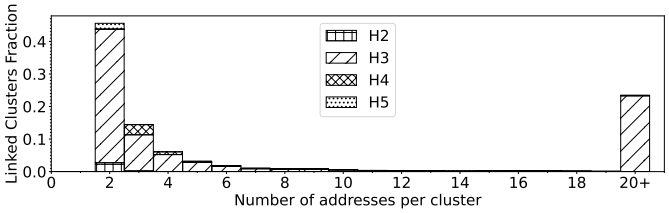


Fig. 10: Number of addresses per TC cluster. 45.57% of the TC 4,931 clusters only have two addresses.

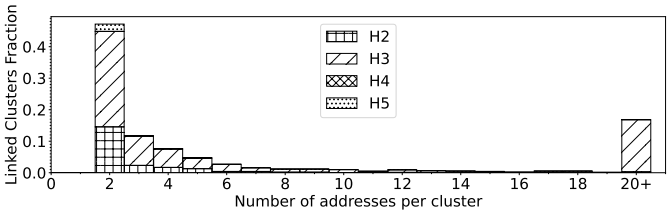


Fig. 11: Number of addresses per TN cluster. 47.18% of the TN 1,345 clusters only have two addresses.

Def. 4), respectively. Figures 10 and 11 visualize the distribution of TC and TN clusters over the number of addresses in a cluster. 2,094 (42.74%) TC and 1,345 (47.18%) TN clusters only have two addresses. Interestingly, we find that the distribution of clusters is similar to previous works on Bitcoin address clustering (e.g., Figure 9(a) in [29]).

C. User Privacy Behavior

Our heuristics appear to function better on the Binance Smart Chain mixer (TN) than on the Ethereum mixer (TC). While our study should be repeated once the other mixers grow on both chains (e.g., Cyclone and TP), our empirical evidence is the first to suggest a differing privacy-focus of users on Ethereum and Binance Smart Chain. One could also argue that privacy-aware users want the best available anonymity set, and will therefore use TC and follow all best practices. As such, a suitable assumption is that anonymity set attracts anonymity set, i.e., the biggest anonymity set will inherently attract more users, and particularly those that worry about privacy (which is analogous to how liquidity attracts liquidity in financial exchanges).

D. Limitations

We would like to point out that our heuristics are best-effort methods and may yield false positives and negatives, a known challenge of related works [15], [48], [38]. For Heuristic 1, it is likely that our assumptions are accurate: When an address a is under control of a user $\mathbf{u}$, a's deposits or withdrawals in a mixer pool should be generated by $\mathbf{u}$. Unfortunately, for Heuristics 2–5, we have no ground truth to verify that two different addresses belong to the same user. Hence, we investigate in Section VII possible side-channels which can form a candidate ground truth dataset from public sources that can be used to validate the results of Heuristics 2–5. Moreover, other potential side-channel information could also be applied to link depositors and withdrawers (e.g., transaction gas price, timestamps, etc.), which could be explored in future work.

VI. Incentivized ZKP Mixer Pools

Spearheaded by the introduction of incentivized mixer pools by AMR [30], we have witnessed a number of real-world mixer pools [9], [7], [44] (cf. Section II-B) introducing rewarding governance tokens through Anonymity Mining. In this section, we analyze how AM affects user privacy.

A. Anonymity Mining in TC Pools

TC incentivizes users to maintain their assets in TC pools through *Anonymity Mining* [44]. Users receive TORN tokens through a so-called shielded liquidity mining protocol consisting of four steps (cf. Figure 12).

(1) Deposit: A user deposits ETH into a TC pool using addresses $\mathbf{addr}_d$, and receives a deposit *note*.

(2) Withdraw: When the user withdraws ETH from a TC pool, the deposit note becomes a *spent note*.

(3) Claim: After withdrawing from a TC pool, the user can submit the *spent note* to the TC pool to claim the Anonymity Points AP. Because AP is determined by the amounts of deposited ETH and the deposited duration (both of which are private information), the user stores AP privately on a shielded account[5].

(4) Swap: A user can convert the shielded AP to public TORN tokens using a dedicated TC Automated Market Maker (AMM) exchange. The user receives the TORN tokens in an address $\mathbf{addr}_r$ that can be different from the user's deposit or withdrawal address.

Equation 27 from TC outlines the amount of AP a user $\mathbf{u}$ is entitled to at time t, where Weight_p is a predefined parameter to calculate a user's AP in various pools[6]. v_p corresponds to the number of withdrawals in the $\mathbf{P}_p$ pool before time t. $t^d_{p,i}$ and $t^w_{p,i}$ are the block numbers of $\mathbf{u}$'s i-th deposit and withdrawal, $0 \leq i \leq v_p$.

$$\mathsf{AP}_{\mathbf{u}}(t) = \sum_{p \in \{0.1,1,10,100\}} \mathsf{Weight}_p \cdot \sum_{i=1}^{v_p} \left(t^w_{p,i} - t^d_{p,i}\right) \quad (27)$$

For instance, let's assume that a user $\mathbf{u}$ deposits twice 1 ETH into $\mathbf{P}_1$ at block 11,476,000 and 11,476,100, and deposits 10 ETH into the $\mathbf{P}_{10}$ pool at block 11,476,000. If $\mathbf{u}$ then withdraws all the deposited funds at block 11,476,200, $\mathbf{u}$'s AP becomes $20 \times (100 + 200) + 400 \times 200 = 86{,}000$.

B. Linking User's Addresses through Anonymity Mining

AM aims to attract users depositing more coins over a longer timeframe. However, AM also increases the required

[5]According to [44], a shielded account is a secret key newly generated by a user, which is used to encrypt and submit claim and withdrawal data without revealing the user's identity. For recoverability, the user encrypts this secret key using his ETH public key and stores the encrypted result on-chain.

[6]Weight_p is predefined as 10, 20, 50 and 400 in TC 0.1, 1, 10, and 100 ETH pools, respectively.

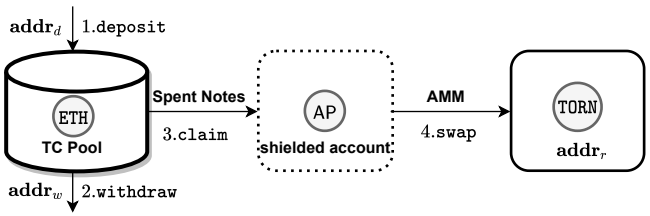


Fig. 12: Overview of the TC Anonymity Mining.

user interactions with mixers (e.g., a claim action to receive rewards), and may thus provoke the leakage of privacy-compromising information. We explore how to link users' withdrawals and deposits by solving Equation 27.

We first identify the addresses that received TORN tokens from TC pools. From block 11,474,710 (December 18th, 2020) to 13,530,000 (November 1st, 2021), we identify 13,751 swap events[7], and find that 1,709 addresses received TORN. We then extract the converted AP value in swap events.

Receive Rewards with Deposit Address. In the following, we demonstrate how re-using a deposit address to receive rewards can deteriorate a user's privacy.

We discover that among the 1,709 addresses receiving TORN, 1,027 are depositors. We extract their deposit time, receiving TORN time, and the converted values of AP. Based on the data, we divide the 1,027 depositors into three categories:

- 1 *deposit*/1 *claim*/1 *pool:* Out of the 1,027 depositors, 224 only deposited *once* in *one* TC pool and only received TORN tokens from AP with *one* transaction. In this case, Equation 27 can be simplified as $\mathsf{AP}_{\mathbf{u}}(t) = \mathsf{Weight}_p \cdot (t^w_{p,1} - t^d_{p,1})$. Because $\mathsf{AP}_{\mathbf{u}}(t)$ and $t^d_{p,1}$ are known, we can resolve the value of $t^w_{p,1}$ and search if there is a withdrawal transaction in block $t^w_{p,1}$. In total, we find the withdrawals for 50 depositors. For the remaining 174 depositors, we speculate that we cannot find their withdrawals because they have likely not yet converted all their AP.
- n *deposits*/1 *claim*/1 *pool:* 194 addresses deposited *more than once* in *one* TC pool but only received TORN *once*. Equation 27 can be simplified as $\mathsf{AP}_{\mathbf{u}}(t) = \mathsf{Weight}_p \cdot \sum_{i=1}^{v_p} (t^w_{p,i} - t^d_{p,i})$. In this case, we find the possible withdrawals for 67 depositors.
- n *deposits*/n *claims*/n *pools:* For the remaining 609 depositors receiving TORN *more than once* or using *multiple* pools, it is challenging to find their withdrawals, because we are not sure if they have claimed all AP and Equation 27 is hard to solve. However, we would suggest TC users avoid reusing deposit addresses to receive TORN. The reason is simple: One conversion of AP for a depositor shows that this depositor has already (partly or entirely) withdrawn the deposits from TC.

[7]TC Reward Swap contract emits Swap (address indexed recipient, uint256 pTORN, uint256 TORN) events, where pTORN is the value of AP.

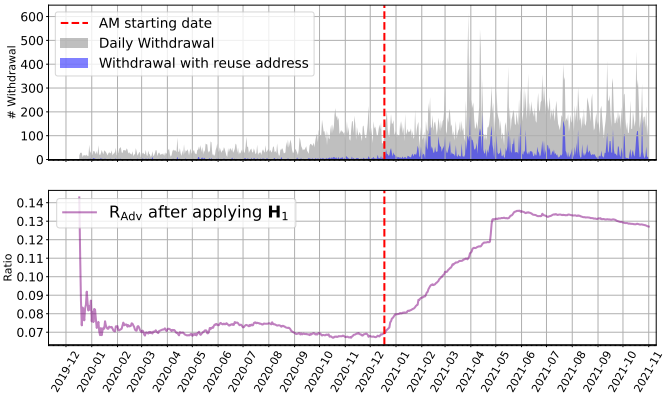


Fig. 13: Daily withdrawal in the four TC ETH pools. The AM launch does not increase the number of daily withdrawals but attracts privacy-ignorant users. Heuristic 1 performs better after AM started, i.e., the relative increase of the probability that an adversary links a withdrawer to the correct depositor (cf. Eq. 20) rises from 7.00% to 13.50%.

In total, we can find the possible withdrawal transactions for 117 (11.39%) re-using addresses, indicating that re-using a deposit address for AM can deteriorate a user's privacy.

C. Does AM Contribute to the Anonymity Set?

As shown in Section VI-B, TC AM users tend to reuse their deposit addresses to receive TORN rewards, ignoring their privacy. To understand whether AM contributes to enlarging a mixer pool's AS, we plot the number of daily withdrawal transactions and extract the withdrawals generated by reusing deposit addresses in the four TC ETH pools (cf. Figure 13).

We observe that the number of daily withdrawals in TC pools is not affected by AM as intended: the number started increasing before AM launch on October 18th, 2020. However, AM does attract more users reusing the deposit addresses to withdraw. Such "reusing depositors" are likely interested in mining TORN, but privacy-ignorant.

Heuristic 1 is specialized in identifying naive users that reuse addresses. We observe that Heuristic 1 performs better after AM started, i.e., the relative increase of the probability that an adversary links a withdrawer to the correct depositor (cf. Eq. 20) rises from 7.00% to 13.50% (cf. Figure 13).

In conclusion, contrary to the claims of related work [30], we find that AM does not always contribute to the mixers' anonymity set size as expected, because it attracts privacy-ignorant users.

VII. Heuristic Validation Attempt

We observe the existence of a variety of publicly available side-channels that may indicate whether two blockchain addresses belong to the same entity. In this section, we expand on three of such side-channels, and then synthesize a candidate ground truth dataset to validate the heuristics presented in Section V.

TABLE VIII: Validation attempt for TC linked address pairs. $\mathcal{S}_{\text{TC}}^{\text{H}_i}$ represents the linked address pairs obtained through Heuristic i. For Heuristic 2, 3, and 5, Test Pairs = depositors in $\mathcal{S}_{\text{GT}}$ (the Candidate Ground Truth data) $\times$ withdrawers in $\mathcal{S}_{\text{GT}}$. For Heuristic 4, Test Pairs = distance-2 depositors in $\mathcal{S}_{\text{GT}}$ $\times$ distance-1 depositors in $\mathcal{S}_{\text{GT}}$.

Candidate Ground Truth $\mathcal{S}_{\text{GT}}$	Heuristics	Test Pairs	$tp = \mathcal{S}_{\text{GT}} \cap \mathcal{S}_{\text{TC}}^{\text{H}_i}$	$tn = \overline{\mathcal{S}_{\text{GT}}} \cap \overline{\mathcal{S}_{\text{TC}}^{\text{H}_i}}$	$fp = \overline{\mathcal{S}_{\text{GT}}} \cap \mathcal{S}_{\text{TC}}^{\text{H}_i}$	$fn = \mathcal{S}_{\text{GT}} \cap \overline{\mathcal{S}_{\text{TC}}^{\text{H}_i}}$	precision	recall	F1
$\mathcal{S}_{\texttt{Airdrop}}$ (35,081)	$\mathbf{H}_2$	931 × 580	2	539,747	4	227	0.33	0.01	0.02
	$\mathbf{H}_3$	931 × 580	229	539,367	384	0	0.37	1.00	0.54
	$\mathbf{H}_5$	931 × 580	0	539,751	0	229	0.00	0.00	0.00
	$\mathbf{H}_2 + \mathbf{H}_3 + \mathbf{H}_5$	931 × 580	229	539,366	385	0	0.37	1.00	0.54
	$\mathbf{H}_4$	710 × 931	2	660,641	3	364	0.40	0.01	0.01
$\mathcal{S}_{\texttt{ENS}}$ (5,105)	$\mathbf{H}_2$	291 × 213	1	61,928	2	52	0.33	0.02	0.04
	$\mathbf{H}_3$	291 × 213	50	61,854	76	3	0.40	0.94	0.56
	$\mathbf{H}_5$	291 × 213	0	61,930	0	53	0.00	0.00	0.00
	$\mathbf{H}_2 + \mathbf{H}_3 + \mathbf{H}_5$	291 × 213	50	61,854	76	3	0.40	0.94	0.56
	$\mathbf{H}_4$	118 × 291	0	34,311	0	27	0.00	0.00	0.00

A. Airdrop Side-Channel

A blockchain airdrop is a form of donation, where a coin is given to a blockchain address without further explicit expectation. Victor et al. [48] present the following privacy-related airdrop approach: if a user receives an airdrop on multiple addresses and aggregates those funds within a short timeframe after the airdrop to one central address, this address can be labeled as the user's primary address. As such the first side-channel we consider is the *Airdrop* approach.

In our evaluation, we consider two particular instances of DeFi airdrops. First, the Uniswap airdrop and second, the 1inch airdrop. To apply Victor's heuristic, we crawl transaction data on the Ethereum network in the first seven days after an airdrop took place.

Results. From the airdrop data we identify a total of 35,081 linked address pairs, denoted as $\mathcal{S}_{\texttt{Airdrop}}$.

B. Ethereum Name Service (ENS) Side-Channel

In the following, we propose two novel approaches to link addresses using ENS [2] data. ENS is a decentralized naming service on Ethereum [2], aiming to map human-readable names (e.g., "alice.eth") to blockchain addresses. Similar to DNS, ENS proposes dot-separated hierarchical domains, and a domain owner can create subdomains (e.g., "foo.alice.eth"). To map a new `name` to an address a, a user registers the `name` with a and sets its expiry time. Users can also transfer the ownership of a name to another address, or assign subdomains to addresses. For a more thorough background on ENS, we refer the reader to related works [2], [4], [51].

Linking Addresses through ENS Usage. To cluster ENS addresses, we provide two approaches:

- *Name Ownership Transfers*: Given two addresses a_1 and a_2, if a_1 transfers the ownership of an ENS `name` to a_2, before `name` expires, and a_1 only transfers its name once, then $\textsc{Link}(\mathsf{a}_1, \mathsf{a}_2) = 1$.
- *Subdomain Assignments*: Given two addresses a_1 and a_2, if a_1 has an ENS `name` and assigns a subdomain of `name` to a_2, then $\textsc{Link}(\mathsf{a}_1, \mathsf{a}_2) = 1$.

Results. To apply the *Name Ownership Transfers* approach, we crawl all (372,756) `Transfer` events of the ENS registry contract until November 1st, 2021. We extract the address pairs $(\mathsf{a}_1, \mathsf{a}_2)$, where a_1 transfers a name to a_2 and a_1 only transfers its name once. This approach can link 4,399 address pairs.

To apply the *Subdomain Assignments* approach, we crawl all (900) `NewOwner` events emitted when a user directly calls the ENS registry contract. We then extract the address pairs $(\mathsf{a}_1, \mathsf{a}_2)$, where a_1 assigns subdomains to a_2. We can identify 725 linked address pairs.

In total, from the ENS data, we can link 5,124 address pairs, denoted as $\mathcal{S}_{\texttt{ENS}}$.

C. Debank Side-Channel

Debank [11] is an online blockchain explorer for tracking DeFi user portfolios. Users can log into Debank through a wallet (e.g., MetaMask) and *follow* other addresses, similar to a social network. We hence assume that a user is unlikely to follow its own addresses and propose the following approach. Note that this is the first side-channel we consider which yields a *negative signal* on whether two addresses are linked.

- *Debank Follower and Following Relationship*: Given two addresses a_1 and a_2, if a_1 follows a_2, or a_1 is followed by a_2 on Debank, then $\textsc{Link}(\mathsf{a}_1, \mathsf{a}_2) \neq 1$.

Results. For each TC depositor and withdrawer address, we crawl the follower and following addresses on Debank before November 1st, 2021, i.e. those Debank addresses that follow the TC address or followed by TC users. Out of $54,504$ TC addresses, we find that $655 + 258 = 913$ addresses (1.8%) have at least one follower or following address on Debank.

Let $\mathcal{S}_{\texttt{Debank}}$ be the set of TC depositor-withdrawer pairs $(\mathsf{a}_d, \mathsf{a}_w)$, where a_d follows a_w, or a_d is followed by a_w on Debank. Our results show that $|\mathcal{S}_{\texttt{Debank}}| = 150$, i.e., 150 depositor-withdrawer pairs have a follower or following relationship.

D. Validation Attempt

In the following, we attempt to validate the heuristics presented in Section V, using $\mathcal{S}_{\texttt{Airdrop}}$, $\mathcal{S}_{\texttt{ENS}}$ and $\mathcal{S}_{\texttt{Debank}}$ as the candidate ground truth data. Note that we can only validate the link of TC address pairs, not the link among deposit and withdrawal transactions. We therefore omit Heuristic 1 from the validation process, as H1 does not link addresses.

1) $\mathcal{S}_{Airdrop}$ + $\mathcal{S}_{ENS}$: Table VIII shows the results of our heuristic validation by applying the side-channels given by $\mathcal{S}_{\texttt{Airdrop}}$ and $\mathcal{S}_{\texttt{ENS}}$. Unfortunately, $\mathbf{H}_2$, $\mathbf{H}_4$, and $\mathbf{H}_5$ appear to perform rather poorly, when compared to $\mathbf{H}_3$. This result appears plausible, when considering that $\mathbf{H}_3$ focuses on asset-transfers, which also applies to the Airdrop and ENS side-channel data. Luckily, heuristic 3 is the most potent heuristic to reduce the anonymity set size, as shown in Table VIII.

2) Airdrop and ENS Side-Channel Intersection: To increase our confidence on the side-channel data, we intersect the candidate ground truth data sources: if an address pair $(\mathsf{a}_1, \mathsf{a}_2)$ are linked both in $\mathcal{S}_{\texttt{Airdrop}}$ and $\mathcal{S}_{\texttt{ENS}}$, then a_1 and a_2 are more likely to be controlled by the same user. Nevertheless, we find that the overlap size of the airdrop and the ENS data consists of only 13 pairs. We hence refrain from applying the intersected side-channel dataset to validate $\mathcal{S}_{\texttt{TC}}$.

3) $\mathcal{S}_{Debank}$: We find that, out of the 150 depositor-withdrawer pairs in $\mathcal{S}_{\texttt{Debank}}$, 34 (23%) pairs are linked through Heuristics 2, 3, and 5. Therefore, if we regard the Debank follower relationship data as the ground truth, then those 34 addresses cannot be owned by the same user; thus, we consider them false positives.

4) Results Summary: In conclusion, by applying the airdrop and ENS side-channels as candidate ground truth datasets, our heuristics can achieve an average F1 score of 0.55 (cf. Table VIII), whereas Heuristic 3 provides the strongest signal. Our results suggest that validating the heuristics presented in Section V is a challenging, but feasible task. Our methodology can be further extended with additional side-channel data to synthesize a larger candidate ground truth dataset (e.g., by crawling Twitter data from testnet wallet validations, additional blockchain explorer labels, etc.).

VIII. Discussion and Implications

We now discuss several implications of our work, and how our findings can help advance the understanding and design of future DeFi mixers.

Towards a More Accurate Anonymity Set Size. The *Observed Anonymity Set* (**OAS** – cf. Eq 15) presented by current ZKP mixers counts the total amount of unique depositors in a specific pool. Although it is challenging to compute the *True Anonymity Set* (cf. Eq 16), in this work, we propose five heuristics (cf. Section V) to calculate more realistic anonymity set sizes based on public on-chain data. Our results show that, on average, a more realistic anonymity set size for a TC pool is reduced by 34.18% from its reported **OAS**. Therefore, we argue that mixers should also report more realistic anonymity sets of their pools on their websites along with the **OAS**. Our findings further suggest that mixers should improve the UI, warning users about privacy-compromising actions.

What Privacy-Impact Yields Anonymity Mining? Anonymity mining attracts privacy-ignorant users primarily interested in mining rewards. As a result, their deposits do not contribute to the anonymity set size as expected (Section VI-C). To enlarge the anonymity set size, AM can be improved by incentivizing only the privacy-aware users, e.g., providing rewards to users that interact with a mixer in a privacy-conscious manner.

Using Mixers via Cross-chain Bridges. Among all ZKP mixers, TC has the largest **OAS** (cf. Table I). Users on other blockchains (e.g., BSC, Polygon) can also leverage TC to enhance their privacy, i.e., users can convert their funds to ETH through decentralized bridges (e.g., Anyswap [10]) and then deposit ETH into TC. As shown in Figure 8, 0.79% and 0.72% of TC deposited volume are from BSC and Polygon bridges. Moreover, the economic cost (e.g., relayer fees, gas price, etc.) may also affect users' choice of mixer. These observations could be the basis for future research endeavors on analyzing cross-blockchain transactions of privacy seeker users.

Tracing Malicious Addresses. Although malicious addresses can adopt DeFi mixers to launder money, achieving unlinkability requires the proper use of mixers. Our heuristics can find the possible withdrawers for 5 (5.74%) out of the 87 labeled malicious addresses (cf. Section IV-E), which enables us to trace their coin flow after withdrawing assets from mixers until centralized exchanges. Furthermore, given an address's registration information on centralized exchanges, we could link the user's identity in the real world. For more details, we refer the reader to our case studies in Appendix A.

IX. Related Work

Mixers on Bitcoin: Mixers were originally applied in anonymous communications [24] and are also applied to enhance Bitcoin users' privacy. Mixcoin [22] and Blindcoin [47] are centralized, trusted mixers that receive BTC from a user's address d and then return BTC to another address w of the same user. CoinJoin [31] allows a user to find other mixing partners to merge multiple transactions, thereby obfuscating the link between senders and recipients. Although the design of CoinJoin [31] is decentralized, its existing implementation, such as Wasabi wallet, remains centralized but non-custodial. CoinShuffle [39], CoinShuffle++ [41], and Xim [20] propose to achieve better anonymity in a decentralized mixer. The ecosystem and mixing mechanisms of Bitcoin mixing service are also studied in recent work [37], [50].

Mixers on smart contract-enabled Blockchain: ZKP mixers are inspired by Zerocash [42] to obfuscate the link between the users' deposit and withdrawal using zero-knowledge proof. Several ZKP mixers attempt to operate on Ethereum, such as Miximus [17] and Hopper [3]. AMR [30] proposes how to reward users for participating in a mixer, and shortly after, Blender [9] implements a mixer with a reward scheme. TC follows by adding anonymity mining as a deposit reward scheme for users [44]. Besides ZKP mixers [6], [7], [8], [1], a notable mixer example that relies on linkable ring signatures and the stealth addresses from Monero [14] is Möbius [32].

Analysis of blockchains privacy: Many researchers have studied privacy on non-privacy-preserving blockchains (e.g., Bitcoin [15], [25], Ethereum [18], [48]), as well as on

privacy-preserving blockchains (e.g., Monero [34], [35], Zerocash [28], [19]). As ZKP mixers are inspired by Zerocash, our heuristics 1 and 4 can also be applied to link shielded and deshielded transactions in Zerocash, and can reduce the anonymity set size of the shielded pool by 69.1% when combining other heuristics presented in [28]. However, the majority of the transactions (i.e., with 65.6% of the withdrawn value) in [28] involve miners or founders, while this paper investigates generic ZKP mixer users, and can be applied to trace malicious addresses. Moreover, recent studies [52] have also shown that users' privacy may be leaked when using cross-chain exchanges.

X. Conclusion

This paper empirically demonstrates that the advertised anonymity set sizes of popular mixers (such as Tornado.Cash) do not represent the true privacy offered to users. We propose a methodology that can increase the probability that an adversary links a withdrawer to the correct depositor on average by 51.94% (108.63%) on TC (on Ethereum) and TN (on Binance Smart Chain) respectively. Worryingly, while previous work suggests that incentivized mixers could improve the offered mixer privacy, we find evidence that speculators are likely to act in a privacy-ignorant manner, deteriorating the overall anonymity set size. We measure that 60.09% (i.e., 1,027 out of 1,709) of the reward claiming users are improperly using the mixer, resulting in *worse* privacy than if they had ignored the mixer reward coins. We hope that our work engenders further research into user-friendly, backward-compatible, and privacy-enhancing anonymity mining solutions.

References

[1] Cyclone. Available at: https://cyclone.xyz/bsc.

[2] Ethereum name service. Available at: https://app.ens.domains/.

[3] Hopper. Available at: https://hoppereth.org/.

[4] Introduction to ethereum name service. Available at: https://docs.ens.domains/.

[5] Layer 2 rollups. Available at: https://ethereum.org/en/developers/docs/scaling/layer-2-rollups/.

[6] Tornado cash. Available at: https://tornado.cash/.

[7] Typhoon.cash. Available at: https://typhoon.cash/.

[8] Typhoon.network. Available at: https://app.typhoon.network/.

[9] Blender, 2020. Available at: https://ourblender.github.io/.

[10] Anyswap. https://anyswap.exchange/, 2021.

[11] Debank. https://debank.com/, 2021.

[12] Starkware, 2021. Available at: https://medium.com/starkware.

[13] Aave. Aave Protocol. https://github.com/aave/aave-protocol, 2020.

[14] Kurt M. Alonso. Zero to Monero: First edition. a technical guide to a private digital currency; for beginners, amateurs, and experts. https://web.getmonero.org/library/Zero-to-Monero-2-0-0.pdf.

[15] Elli Androulaki, Ghassan O Karame, Marc Roeschlin, Tobias Scherer, and Srdjan Capkun. Evaluating user privacy in bitcoin. In *International Conference on Financial Cryptography and Data Security*, pages 34–51. Springer, 2013.

[16] Nicola Atzei, Massimo Bartoletti, and Tiziana Cimoli. A survey of attacks on ethereum smart contracts (sok). In *International conference on principles of security and trust*, pages 164–186. Springer, 2017.

[17] barryWhiteHat. Miximus. Available at: https://github.com/barryWhiteHat/miximus.

[18] Ferenc Béres, István András Seres, András A Benczúr, and Mikerah Quintyne-Collins. Blockchain is watching you: Profiling and deanonymizing ethereum users. *arXiv preprint arXiv:2005.14051*, 2020.

[19] Alex Biryukov, Daniel Feher, and Giuseppe Vitto. Privacy aspects and subliminal channels in zcash. In *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*, pages 1813–1830, 2019.

[20] George Bissias, A Pinar Ozisik, Brian N Levine, and Marc Liberatore. Sybil-resistant mixing for bitcoin. In *Proceedings of the 13th Workshop on Privacy in the Electronic Society*, pages 149–158, 2014.

[21] Joseph Bonneau, Andrew Miller, Jeremy Clark, Arvind Narayanan, Joshua A Kroll, and Edward W Felten. Sok: Research perspectives and challenges for bitcoin and cryptocurrencies. In *2015 IEEE symposium on security and privacy*, pages 104–121. IEEE, 2015.

[22] Joseph Bonneau, Arvind Narayanan, Andrew Miller, Jeremy Clark, Joshua A Kroll, and Edward W Felten. Mixcoin: Anonymity for bitcoin with accountable mixes. In *International Conference on Financial Cryptography and Data Security*, pages 486–504. Springer, 2014.

[23] Benedikt Bünz, Shashank Agrawal, Mahdi Zamani, and Dan Boneh. Zether: Towards privacy in a smart contract world. *IACR Cryptol. ePrint Arch.*, 2019:191, 2019.

[24] David L Chaum. Untraceable electronic mail, return addresses, and digital pseudonyms. *Communications of the ACM*, 24(2):84–90, 1981.

[25] Arthur Gervais, Srdjan Capkun, Ghassan O Karame, and Damian Gruber. On the privacy provisions of bloom filters in lightweight bitcoin clients. In *Computer Security Applications Conference*, pages 326–335, 2014.

[26] Ethan Heilman, Leen Alshenibr, Foteini Baldimtsi, Alessandra Scafuro, and Sharon Goldberg. Tumblebit: An untrusted bitcoin-compatible anonymous payment hub. In *Network and Distributed System Security Symposium*, 2017.

[27] Abraham Hinteregger and Bernhard Haslhofer. An empirical analysis of monero cross-chain traceability. *CoRR*, abs/1812.02808, 2018.

[28] George Kappos, Haaroon Yousaf, Mary Maller, and Sarah Meiklejohn. An empirical analysis of anonymity in zcash. In *27th {USENIX} Security Symposium ({USENIX} Security 18)*, pages 463–477, 2018.

[29] Ghassan O Karame, Elli Androulaki, Marc Roeschlin, Arthur Gervais, and Srdjan Čapkun. Misbehavior in bitcoin: A study of double-spending and accountability. *ACM Transactions on Information and System Security (TISSEC)*, 18(1):2, 2015.

[30] Duc V Le and Arthur Gervais. Amr: Autonomous coin mixer with privacy preserving reward distribution. *ACM Conference on Advances in Financial Technologies (AFT'21)*, 2021.

[31] Greg Maxwell. Coinjoin: Bitcoin privacy for the real world. In *Post on Bitcoin forum*, 2013.

[32] Sarah Meiklejohn and Rebekah Mercer. Möbius: Trustless tumbling for transaction privacy. *Proceedings on Privacy Enhancing Technologies*, 2018(2):105–121, 2018.

[33] Ian Miers, Christina Garman, Matthew Green, and Aviel D Rubin. Zerocoin: Anonymous distributed e-cash from bitcoin. In *2013 IEEE Symposium on Security and Privacy*, pages 397–411. IEEE, 2013.

[34] Andrew Miller, Malte Möser, Kevin Lee, and Arvind Narayanan. An empirical analysis of linkability in the monero blockchain. *arXiv preprint arXiv:1704.04299*, 2017.

[35] Malte Möser, Kyle Soska, Ethan Heilman, Kevin Lee, Henry Heffan, Shashvat Srivastava, Kyle Hogan, Jason Hennessey, Andrew Miller, Arvind Narayanan, et al. An empirical analysis of traceability in the monero blockchain. *Proceedings on Privacy Enhancing Technologies*, 2018(3):143–163, 2018.

[36] Satoshi Nakamoto. Bitcoin: A peer-to-peer electronic cash system, 2008. Available at: https://bitcoin.org/bitcoin.pdf.

[37] Jaswant Pakki, Yan Shoshitaishvili, Ruoyu Wang, Tiffany Bao, and Adam Doupé. Everything you ever wanted to know about bitcoin mixers (but were afraid to ask). In *International Conference on Financial Cryptography and Data Security*, pages 117–146. Springer, 2021.

[38] Matteo Romiti, Friedhelm Victor, Pedro Moreno-Sanchez, Peter Sebastian Nordholt, Bernhard Haslhofer, and Matteo Maffei. Cross-layer deanonymization methods in the lightning protocol. In *International Conference on Financial Cryptography and Data Security*, pages 187–204. Springer, 2021.

[39] Tim Ruffing, Pedro Moreno-Sanchez, and Aniket Kate. Coinshuffle: Practical decentralized coin mixing for bitcoin. In *European Symposium on Research in Computer Security*, pages 345–364. Springer, 2014.

[40] Tim Ruffing, Pedro Moreno-Sanchez, and Aniket Kate. P2P mixing and unlinkable bitcoin transactions. In *Network and Distributed System Security Symposium*, 2017.

[41] Tim Ruffing, Pedro Moreno-Sanchez, and Aniket Kate. P2p mixing and unlinkable bitcoin transactions. In *NDSS*, 2017.

[42] Eli Ben Sasson, Alessandro Chiesa, Christina Garman, Matthew Green, Ian Miers, Eran Tromer, and Madars Virza. Zerocash: Decentralized anonymous payments from bitcoin. In *Symposium on Security and Privacy*, pages 459–474. IEEE, 2014.
[43] Erkan Tairi, Pedro Moreno-Sanchez, and Matteo Maffei. A2l: Anonymous atomic locks for scalability and interoperability in payment channel hubs. Technical report, Cryptology ePrint Archive, Report 2019/589, 2019.
[44] TornadoCash. Tornado.cash governance proposal, 2020. Available at: https://tornado-cash.medium.com/tornado-cash-governance-proposal-a55c5c7d0703.
[45] Tornado.Fund. Decentralizing tornadocash: The launch of tornado fund and the path towards tornadodao. Available at: https://medium.com/@Tornado_Fund.
[46] TyphoonNetwork. Typhoon network – apply as relayer, 2021. Available at: https://docs.typhoon.network/relayers/apply-as-relayer.
[47] Luke Valenta and Brendan Rowan. Blindcoin: Blinded, accountable mixes for bitcoin. In *International Conference on Financial Cryptography and Data Security*, pages 112–126. Springer, 2015.
[48] Friedhelm Victor. Address clustering heuristics for ethereum. In *International Conference on Financial Cryptography and Data Security*, pages 617–633. Springer, 2020.
[49] Gavin Wood. Ethereum: A secure decentralised generalised transaction ledger. *Ethereum project yellow paper*, 151:1–32, 2014.
[50] Lei Wu, Yufeng Hu, Yajin Zhou, Haoyu Wang, Xiapu Luo, Zhi Wang, Fan Zhang, and Kui Ren. Towards understanding and demystifying bitcoin mixing services. In *Proceedings of the Web Conference 2021*, pages 33–44, 2021.
[51] Pengcheng Xia, Haoyu Wang, Zhou Yu, Xinyu Liu, Xiapu Luo, and Guoai Xu. Ethereum name service: the good, the bad, and the ugly. *arXiv preprint arXiv:2104.05185*, 2021.
[52] Haaroon Yousaf, George Kappos, and Sarah Meiklejohn. Tracing transactions across cryptocurrency ledgers. In *28th {USENIX} Security Symposium ({USENIX} Security 19)*, pages 837–850, 2019.

APPENDIX A
TRACING ADVERSARY USERS CASE STUDIES

In the following, we provide two examples in which we apply our linking results in TC to trace malicious addresses.

Example 1: Upbit Hackers. On the 27th November 2019, hackers stolen 342,000 ETH from Upbit, a South-Korean cryptocurrency exchange. As shown in Figure 14, *(1)* A depositor 0xeFf...1A9 receives 1,526.95 ETH from address 0x5a8...857, which obtains the same amount of ETH from four labeled Upbit Hacker addresses. *(2)* 0xeFf...1A9 then deposits 1,524 ETH into TC 1, 10, and 100 ETH pools during block 11,971,221 and 11,972,040. *(3)* From our linking results, we find that 0xD7D...b1f withdraws the same amount from TC during block 11,971,270 and 11,972,098, and then transfers 1,520 ETH to address 0x361...ac7, which finally exchanges all ETH to fiat currency (e.g., USD) on Houbi, a centralized exchange paltform. Given the address's registration information on Huobi, we can link the user's identity in the real world.

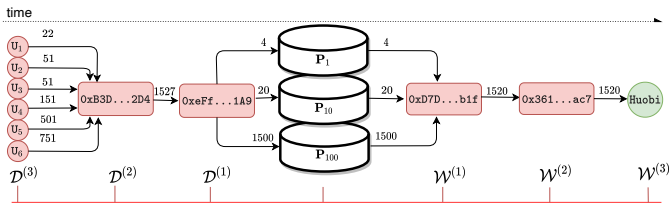


Fig. 14: Example of tracing Upbit Hackers.

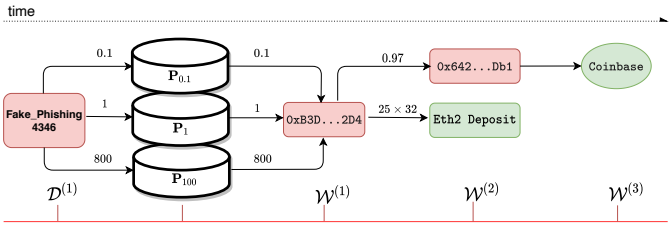


Fig. 15: Example of tracing Tomatos.Finance Fishing addresses.

Example 2: Tomatos.Finance Fishing. 0x917...3e3 is a labeled fishing address (Fake_Phishing4346) which was used to steal users' funds on Tomatos.Finance. We observe that this address leverages TC to launder money. As shown in Figure 15, *(1)* Fake_Phishing4346 deposits 801.1 ETH into TC 0.1, 1, and 100 ETH pools during block 10,944,566 and 10,944,735. *(2)* Through Heuristic 4, we find that the address 0xB3D...2D4 withdraws the same amount of ETH from the three TC pools after block 10,944,735. Then 0xB3D...2D4 is likely linked with Fake_Phishing4346. *(3)* By manually checking the transactions of 0xB3D...2D4, we find that this address transfers 800 ETH to Eth2 Deposit Contract via 25 transactions, and 0.97 ETH to 0x642...Db1, which finally transfers funds to Coinbase, a centralized exchange platform. We can continue to trace the address given its information on Coinbase.