Regulatory Implications of MEV Mitigations

Yan Ji(✉) and James Grimmelmann

Cornell Tech, New York, NY 10044, USA
{yj348,james.grimmelmann}@cornell.edu

Abstract. This paper examines the legal ramifications of Miner / Maximal Extractable Value (MEV), a phenomenon in which some entities (e.g., miners or validators) leverage their positional advantages to generate extra profits on blockchains. In previous work, Barczentewicz *et al.* argued that some MEV extraction techniques could constitute illegal market manipulation under United States securities law, depending on the *publicness* of the victim transactions. While their analysis applies to typical Ethereum and Flashbots implementations, we contend that the rapidly evolving blockchain ecosystem and the emergence of new MEV mitigation measures necessitate a revised test for market-manipulation liability. Our proposal focuses on the principle of respecting the initiating user's *intent*, rather than simply the network status of the transactions. We also identify new enforcement challenges that arise from the decentralization nature of blockchains. By offering a nuanced understanding of the MEV landscape and exploring the legal implications for manipulation liability, this paper contributes to the ongoing discussion on MEV regulation in blockchain ecosystems.

Keywords: Regulation · MEV · DeFi · Blockchain

1 Introduction

"Miner/Maximal Extractable Value (MEV)" is the potential extra profits some entities can make by leveraging their power to insert, omit, or reorder blockchain transactions. MEV concept was introduced in the paper "Flash Boys 2.0," by Daian *et al.* [38], and since then has been a topic of significant discussion and analysis in the blockchain communities given its substantial economic impact. Over $2 billion has been exploited through MEV strategies as of the time of this writing.[1] MEV exploitations result in revenue for validators but sometimes at the expense of ordinary users of these blockchain networks.

In recent work [14], Barczentewicz *et al.* made an initial attempt to analyze the potential legal liabilities associated with MEV extraction. In particular, they classified pending transactions by their publicness and discussed legal liability for extracting MEV from public and private transactions, respectively. They argued that (i) prosecuting cases involving MEV extraction from public transactions is

[1] More than $675 million worth of MEV were exploited in the two years leading up to the Merge on Ethereum [53] and $1.5 billion after the Merge [55].

J. Budurushi et al. (Eds.): FC 2024 Workshops, LNCS 14746, pp. 335–363, 2025.
https://doi.org/10.1007/978-3-031-69231-4_21

likely to be a challenging endeavor, due to the difficulty in proving specific intent and demonstrating price artificiality (ii) for cases involving extracting MEV from private transactions, in contrast, there is greater potential for successful fraud-based manipulation charges, and the demonstration of a breach of trust or duty could offer a clear route towards establishing liability.

Given the nuance in transaction routing traces in practice and the significant differences in legal consequences, Barczentewicz *et al.* emphasized that whether a pending transaction is classified as public or private shouldn't be trivially determined by its purported routing. Instead, they proposed a standard for determining transaction publicness. Specifically, they defined a transaction as public if "an actor, who did not directly receive the transaction from the user who submitted it, can access it in an unencrypted state without undue delay and without any special arrangements with the node that initially received the transaction."

Barczentewicz *et al.*'s legal analysis aligns well with the particular Ethereum and Flashbots implementation they focus on. However, the rapidly evolving nature of the blockchain ecosystem, coupled with the emergence of new MEV extraction techniques and mitigation measures, presents new challenges in establishing liability for market manipulation. The multi-tiered, multi-entity MEV extraction supply chain introduced by Flashbots [50] and the increasing reliance on Trusted Execution Environments (TEEs) [80] introduce additional complexities to the liability landscape. In particular, a classification standard based only on transaction publicness is too simple. It overlooks potential fraud resulting from the abuse of public information and breaches of trust, and does not sufficiently address the complicated MEV landscape of today.

To resolve these issues, we propose a test that respects users' *intentions* rather than focusing solely on the *network status* of transactions. If an actor profits by routing a transaction contrary to the sender's intentions or by ordering transactions using information not intended for MEV extraction, there is likely to be liability of fraud-based manipulation. Recent MEV mitigation efforts also align in spirit with this shift in focus from transaction visibility to users' intent, as seen in slippage limits in Uniswap and intent specification in MEV-Share and SUAVE.

On the other hand, the complexities involved with multiple entities and the inherent permissionless and decentralized nature of blockchains introduce new challenges in enforcement. We note that the gap in meeting the prerequisites for establishing legal liability for MEV extraction largely hinges on the community's stance on whether to rely solely on technological solutions for MEV mitigation or to also embrace legal regulations as complementary measures. From a practical standpoint, the feasibility of implementing in-protocol accountability also significantly influences the challenges associated with prosecution.

In this paper, we first review existing attempts on mitigating MEV in Sect. 2. This backdrop illuminates the complexity of the current MEV landscape. In Sect. 3, we explore the regulatory framework for market manipulation, revisiting Barczentewicz *et al.*'s standard for determining the publicness of transactions and their analysis of market manipulation liability in MEV extraction. Next, we introduce our proposed modification of that standard, examine the challenges in pursuing liability under it, and discuss its implications for the community in Sect. 4. We conclude our discussion in Sect. 5.

We focus on United States law because of the exceptional global reach of U.S. financial regulation. U.S. authorities take the position that the U.S. has jurisdiction over transactions that pass through the U.S. financial system, even none of the end-user parties to the transaction have any other connection to the U.S. [43] But because so many transactions are denominated in dollars, pass through intermediary banks located in the U.S., and/or are cleared through U.S. financial institutions, a high proportion of all global payments are potentially subject to U.S. securities, sanctions, or other financial regulations. While there are limits on the degree to which some of these regimes apply to "foreign" transactions, [36] these are typically regime-specific and others apply broadly.

2 MEV Mitigation Attempts

Various efforts have been made to mitigate rampant MEV exploitation. In this section, we will discuss general-purpose approaches to mitigating MEV.[2] We specifically examine scenarios when these mitigation schemes have failed or may fail (due to design vulnerabilities or engineering bugs), to illustrate the complexity and difficulty in addressing this problem from a technical perspective and to shed light on when regulation may serve as a backup oversight.

2.1 Proposer-Builder Separation

Proposer-Builder Separation (PBS) [83], the increasingly popular design philosophy on Ethereum, aims to divide the tasks of block building and proposing - previously both dominated by validators - into separate stages. The primary motivation behind PBS is to prevent market monopolies, based on the view that MEV is an inherent aspect of DeFi and thus inevitable [37]. In Ethereum 2.0, which employs Proof-of-Stake, there is a risk that MEV exploitation could become dominated by a few validators with specialized expertise in extracting MEV. This could lead to a concentration of capital among these major validators in the long run, which is counter to blockchain decentralization and may introduce severe security issues.

Among various designs of PBS [21,81], the one in use today on Ethereum is MEV-boost [54], developed by Flashbots. MEV-boost introduces three new types of participants: MEV searchers, block builders, and relays. Any entity can participate as a searcher or builder, while there is a higher trust bar for relays. The workflow of MEV-boost is depicted in Fig. 1. Searchers scan the public

[2] We do not consider application-specific solutions [29,33,63,107,108] due to their limited scope [64]. A broad introduction to the prerequisite concepts of blockchains, smart contracts, decentralized finance and MEV is presented in Sect. A.

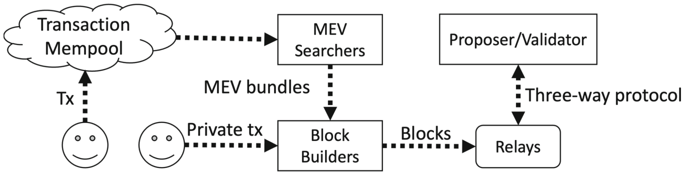

Fig. 1. MEV-boost workflow

mempool for MEV opportunities, create sandwiching bundles[3] and send those bundles to builders. Builders collect transactions from both the mempool and searchers, create blocks with transaction ordering that maximizes their profit, and send those blocks to relays. Next, relays engage in a three-way interactive protocol with the validator chosen to propose the next block. They first send the most profitable block header (without the actual transactions) to the validator. The validator selects the most profitable header, signs it, and sends the signature back to the corresponding relay. Upon receiving the signed block header, the relay broadcasts the full block, including transactions, to both the validator and the entire blockchain network. This allows validators to proceed with the Proof-of-Stake consensus protocol using the newly created block.

The three-way protocol between relays and validators is designed to prevent free-riding by validators. Without it, an adversarial validator who is less capable of exploiting MEV than searchers and builders could replace the sandwiching transactions with their own, following the same strategy developed by others. With relays revealing only the block header but not the transactions, the validator is unable to sandwich any transactions or learn the strategy of searchers and builders. After signing the block header, the validator is committed to proposing this particular block. If the validator deviates from this protocol and proposes a new block, it will be punished by the slashing rules [42], i.e., a proportion of their locked stake will be slashed. This strategy works as expected only if relays do not collude with validators or free ride by themselves, thus trust is needed in relays. In practice, relays are established by reputation [52]; Flashbots, for example, initially dominated the relay market, relaying over 80% of blocks in the early days [90]. This number has since decreased to around 30% [15], indicating a trend towards decentralization in the relays.

MEV-boost aims to abstract MEV extraction to a separate layer, providing all validators with equal opportunities to extract MEV and prevent market monopolies. Despite this goal, as previously mentioned, the dominant position of validators compared to other entities remains largely unchanged, with over 90% of profits eventually going to validators [12]. From another perspective, the situation might seem even worse for ordinary users due to the open MEV auctions. The total MEV extracted in the network depends on the capabilities of MEV seekers. Before the open auction market was introduced, validators had varying capabilities in extracting MEV, and the expected MEV was averaged

[3] Sandwiching is a common practice of extracting MEV by injecting one transaction before the victim one and another after it. For more details, see Sect. A.3.

out. With MEV-boost, however, the MEV extracted in each block is determined by the most skilled player in the network, not depending on the power distribution among validations. A study shows that validators earn more than two times the MEV than before [102]. Concerns have also been raised within the community regarding Flashbots' assumption that MEV is inevitable [87], as well as its overall impact on the ecosystem and alignment with the fundamental principles of decentralization and fairness [70].

MEV-boost was attacked in early April 2023 by an adversarial validator exploiting $20 million from MEV searchers and builders [76]. To mount the attack, the adversary sent several transactions that appeared to offer lucrative MEV opportunities to the public mempool. As anticipated, searchers and builders injected transactions to extract MEV from these transactions. The validator signed the block header from the relay and retrieved the transactions. However, instead of proceeding with the signed block in the consensus protocol, the adversary created a new block, extracting MEV from the injected transactions by MEV seekers, and proposed it to the network. The attack would not have been so smooth without a vulnerability in the Flashbots relay software [16]. Nevertheless, such an *equivocation* attack (equivocating the relayed block) remains possible in today's MEV-boost infrastructure, raising concerns about the effectiveness of the incentive mechanism aimed at enhancing security of Proof-of-Stake consensus in Ethereum 2.0 [84]. Typically, when the MEV opportunities are significant enough and exceed validators' stakes to be slashed for misconduct, there is evident motivation for validators to equivocate. In response to the attack, Flashbots not only patched the software bug, but also added a blacklist of adversarial validators [31]. However, the blacklist functionality was removed just a couple of days later [32]. This incident highlights potential vulnerabilities in the current MEV-boost infrastructure and challenges in PBS for the ecosystem.

2.2 Private Order Flow

To mitigate ordinary users' loss in MEV extraction, in contrast to the goal of PBS, one widely adopted strategy is private order flow (POF), by which users' transactions remain private at the POF provider before being committed on chain. Taking Flashbots Protect [51] as an example, users who prefer not to be a part of the MEV game can send their transactions to Flashbots. These transactions will not be publicly visible in the mempool; instead, they are secretly included in blocks created by Flashbots builders. Flashbots promises not to extract MEV from these transactions and not to disclose their information to any entity until they are included in a block signed by validators. Transactions that might fail and revert the state after execution are also excluded to save users' transaction fees. There are similar proposals for private order flows, such as the OFA design [60], which vary by privacy policies. However, transactions going through Flashbots Protect may experience delays; a higher proportion of Flashbots-built blocks results in shorter latency. Flashbots also does not protect privacy if the blockchain is forked, which is technically unavoidable. Furthermore, Flashbots Protect operates in a centralized manner due to its inherent privacy requirements, which leads to a lack of transparency. There is limited public information available regarding the statistics or details of its policy.[4]

An alternative implementation of POF without centralized trust uses Trusted Execution Environments (TEEs). This approach has been gaining traction, as TEEs offer integrity and privacy guarantees at the hardware level. TEEs are specially designed, isolated computing hardware enclaves that provide two key security guarantees, setting them apart from ordinary computers. First, TEEs ensure that program execution cannot be tampered with by adversaries. In the context of smart contract execution, for instance, users do not need to explicitly verify transactions, as the hardware manufacturer's attestations provide assurance of correct execution. Second, TEEs offer robust privacy protection, ensuring that the internal execution state remains invisible to external parties, including the operating system. To implement anonymous payments using TEEs, for example, users can encrypt transaction details, including sender, receiver and amount, before sending them to the TEEs. Transactions are then decrypted and executed inside the TEE enclaves without revealing any information to a third party. Intel's Software Guard Extensions (SGX) [80] is one example of a TEE that has been adopted for blockchain applications.

By employing TEEs to process POF, MEV exploitation can be mitigated. For instance, Secret Network [93] provides private smart contract execution and native MEV resistance by having validators order and execute transactions within SGX enclaves. Flashbots also recently released a builder implementation inside SGX [61], paving the way for a forthcoming plan for a more complex trustless order flow auction system called SUAVE [48]. The use of SGX in this context has the potential to address the transparency concerns associated with Flashbots Protect, as well as eliminate the need to trust Flashbots. Another recent academic proposal, called PROF [12], leverages SGX for private order flow while remaining compatible with the current MEV-boost infrastructure. PROF allows for seamless integration by providing economic incentives for all types of participants in the MEV ecosystem.

While the ideal security and privacy features of TEEs seem promising, incorporating them into real-world applications with the expected guarantees has proven to be challenging, and many issues can arise. For example, a recent study on Secret Network revealed that all the privacy guarantees it claimed were broken [68]. In particular, transaction details such as sender, receiver, and amount could be observed, leading to straightforward MEV exploitation. This attack could be mounted by any individual validator in the network independently, and it is not dependent on the TEE implementation (the underlying TEE remains intact). Secret Network froze validator registration before patch-

[4] To mitigate the need to trust a centralized entity not to abuse its non-public information, commit-reveal protocols [9,82,89] have been proposed to allow transactions to be ordered in their encrypted form first and then revealed for execution. These protocols trust among validators not to collude and decrypt transactions before the ordering is committed on chain. This strategy may introduce other concerns, such as spamming attacks, which are out of the scope of this paper.

ing the vulnerability. Aside from the intricacies of system deployment, several side-channel attacks[5] against SGX have also been demonstrated, including Spectre [77], Foreshadow [100], and AEPIC [59]. These attacks enable adversaries to breach TEE security guarantees and infer secret states within it.

2.3 Fair Ordering Protocols

Fair ordering protocols offer another possible approach to mitigating MEV extraction [72–74,78,106]. Unlike private order flow, which keeps transactions private from MEV extractors, fair ordering protocols aggregate different orderings from validators and produce an ordering that respects the majority of validators. This prevents a single validator who is responsible for proposing the next block from dictating the transaction ordering. Implementing these proposals requires fundamental changes to the underlying consensus protocol, making them more suitable for application-specific scenarios rather than general-purpose blockchains. For instance, Chainlink [69] is developing Fair Sequencing Service within its oracle network to fairly order transactions off-chain and send them in batches to the blockchain for DeFi applications, and Espresso Sequencer [94] is a shared fair sequencer for Layer 2 rollups.

On the other hand, the effectiveness of fair ordering protocols relies on the assumption that the majority of validators will follow the protocol and propose transaction orderings according to the public policy, e.g., based on a first-come, first-served basis. However, there is no inherent incentive for validators to adhere to the protocol, and accountability is not easily enforceable. If a validator misbehaves, holding them accountable for any unexpected outcomes is challenging. In the context of MEV resistance, when a significant MEV opportunity arises, there is no guarantee that the majority of validators will not collude and exploit it. This presents a potential risk to the effectiveness of fair ordering protocols, as their security is contingent on the assumption that the majority of validators will prioritize fairness[6] over personal gain. To address this issue, future research and development may focus on devising mechanisms to incentivize validators to adhere to fair ordering protocols, as well as creating methods to hold them accountable for any misbehavior.

3 Regulatory Framework

In this section, we describe the key U.S.[7] regulatory framework potentially applicable to MEV: the rules against market manipulation adopted by the two key agencies with jurisdiction over financial markets. We review Barczentewicz *et*

[5] Side-channel attacks are privacy attacks that probe secrets using side information outside the original security model, such as timing and resource consumption.

[6] This can be any predetermined ordering policy that promotes public good, such as first-come-first-served.

[7] As discussed in Sect. 1, we focus on U.S. law due to its exceptional global reach in financial regulation, and also for concreteness of discussions.

al.'s analysis of market-manipulation liability for MEV extractors, which argues that transaction publicness has significant regulatory implications.

3.1 Regulatory Agencies

The Securities and Exchange Commission (SEC) and the Commodity Futures Trading Commission (CFTC) serve as the two primary financial-markets regulatory bodies in the United States. They are charged with upholding the principles of market integrity, transparency, and investor protection. Though both agencies share the common goal of ensuring equitable and orderly markets, they focus on regulating different financial instruments. In particular, the SEC regulates *securities* (primarily stock and loans issued by companies to raise capital), while the CFTC regulates trading in *futures* (contracts for the future delivery of a commodity, like crude oil or aluminum) and similar contracts.[8]

The Mango Markets exploit [101] is an example case of crypto market manipulation. The attacker made purchases of MNGO on three digital exchanges that Mango Markets' price oracle depended on, in order to drive up the value of much larger MNGO-USDC swap contracts that he held. Both the SEC and the CFTC brought charges against the attacker, and he was criminally convicted. To be clear, the Mango Markets exploit is not an example of MEV extraction. It is closer in spirit to well-known illegal forms of manipulation in traditional financial markets, such as "banging the close" by submitting buy orders at the end of the trading day to drive up the reported closing price and increase the value of one's derivatives contracts based on that price. However, similar outcomes can be achieved with reduced risk by an MEV extractor who has significant control over transaction ordering [79]. They could employ the same strategy or even just freeride on the attacker's. The SEC and CFTC's action in the Mango Markets case send a signal that they believe that existing rules against market manipulation remain relevant for digital assets.

3.2 Price Manipulation

Both the SEC and CFTC have the power to address price manipulation. Section 9(a)(2) of the Securities Exchange Act (SEA) as amended (and codified at 15 U.S.C. §78i(a)(2)) prohibits price manipulation in securities, and Sect. 6(c)(3) of the Commodity Exchange Act (CEA) as amended (codified at 7 U.S.C. §9(3)) prohibits price manipulation in swaps and futures. Both these statutes have been held to require proof of two aspects: the existence of an artificial price and the specific intent of the defendant to manipulate the price.

CFTC Rule 180.2, 17 C.F.R. §180.2, which implements the CEA prohibition on price manipulation, makes it unlawful "for any person, directly or indirectly,

[8] The regulatory landscape for digital assets is still evolving. Both the SEC and CFTC have claimed jurisdiction over some digital assets, and both have taken numerous enforcement actions. The line between their respective jurisdictions is not firmly established, and several pending legislative efforts such as RFIA [4] may redraw it.

to manipulate or attempt to manipulate the price of any swap, or of any commodity in interstate commerce, or for future delivery on or subject to the rules of any registered entity." For the CFTC to apply Rule 180.2, it must satisfy the following four-part test [3]:

> (1) the defendant possessed the ability to influence market prices; (2) an artificial price existed; (3) the defendant caused the artificial price; and (4) the defendant intended to do so.

This test creates two significant hurdles to an enforcement action. First, there are conceptual difficulties in element (2) in distinguishing between a legitimate market price and an "artificial" price that "does not reflect basic forces of supply and demand." The concept of "legitimacy with respect to supply and demand" remains ambiguous in both law and economics, making the artificial-price test circular unless the sole consideration is whether the forces were triggered by an unlawful act.

Second, the intent required under element (4) is the specific intent to cause an artificial price. Specific intent is the highest standard of culpable mental state used in law. It is not enough to prove that the defendant was negligent, or reckless, or even knew that a result would happen; specific intent is established only when the accused has a conscious desire to achieve a particular outcome. Sometimes, as in the case of an individual trader who brags about price manipulation on a logged chat platform, it is easy to prove. But in other cases, anomalous prices could also plausibly have resulted without a specific intent, and could be due to a software bug or a trading strategy gone awry.

3.3 Fraud-Based Manipulation

Given the difficulties with the price-manipulation theory, an alternate and often more viable enforcement strategy is fraud-based manipulation, which does not require proof of an artificial price or of specific intent. The SEC has traditionally used this approach under SEA §10(b) and its implementing rule, SEC Rule 10b-5. The CFTC originally had limited authority to pursue price manipulation liability. But following the financial crash of 2008, Congress enacted the Dodd-Frank Wall Street Reform and Consumer Protection Act in 2010, which gave the CFTC power to make rules against fraud-based manipulation. It promptly adopted Rule 180.1, 7 C.F.R. §180.1, which mirrors SEC Rule 10b-5.

Both SEC Rule 10b-5 and CFTC Rule 180.1 make it unlawful (1) "to intentionally or recklessly" (2) employ any manipulative or deceptive devices to defraud. This test departs from price manipulation in two important ways. First, the violation is complete the moment the defendant acts fraudulently; there is no requirement that the fraud resulted in an artificial price. Second, it is sufficient to demonstrate recklessness, which is defined as the accused acting in a highly unreasonable manner, to the extent that it seems implausible they were unaware of their actions [2], irrespective of whether they foresaw or intended the result [6]. This standard is generally easier to establish than specific intent.

What the fraud-manipulation theory does require is that there must be evidence of deceit or false statements. This could be a misleading signal of market price (as above), but it could also be any other traditional form of fraud, such as false statements about a company's financial condition. Similarly, insider trading is generally regarded as fraudulent behavior and can lead to allegations of fraud-based manipulation. The existence of fraud typically presumes some level of trust that was established and subsequently violated.

The typical actions of an MEV extractor—inserting and reordering transactions—look superficially like legitimate trading activity. This is analogous to open-market manipulation in traditional financial markets: the transactions are real, and are entered into with real counterparties making wholly aboveboard trades. Courts have split over whether open-market manipulation qualifies as fraud-based manipulation. The answer usually depends on whether the defendant disseminated false information. A subset of open-market manipulation, known as covered open-market manipulation, involves trading activity that creates heightened expectations of trust and honest dealing [14], and is usually considered illegal. For instance, in the Mango Markets exploit, the MNGO price oracle depended on the prices on other exchanges, a dependency that suggests a trust relationship.

3.4 Sandwiching Public and Private Order Flow

MEV extraction via sandwiching—unlike other strategies such as oracle manipulation or arbitrage—is an inherent characteristic of decentralized blockchains on which transactions are committed in batches. It causes the sandwiched user to trade at a worse price and is generally viewed as toxic MEV [13]. Thus, our analysis will focus specifically on its regulatory implications.

Barczentewicz *et al.* divide MEV sandwiching attacks into two types, based on whether the sandwiched transaction originates from a public or private order flow [14]. Under their proposed test, a transaction should be deemed public when "an actor, who did not directly receive the transaction from the user who submitted it, can access it in an unencrypted state without undue delay and without any special arrangements with the node that initially received the transaction." This standard captures the differing life cycles of transactions that are sent directly to the public mempool versus those that utilize a private order flow service like Flashbots Protect. In particular, when a transaction is in the public mempool, any network participant can access it via a straightforward in-protocol command. In contrast, for transactions sent through Flashbots Protect, only Flashbots can access the unencrypted transaction.

As Barczentewicz *et al.* argue, the act of sandwiching public transactions unlikely constitutes market manipulation. These are naked open-market trades, and it will be very difficult for prosecutors to prove the specific specific intent to create an artificial price that leads to price manipulation liability.

Recklessness may be somewhat easier to show for fraud-based manipulation, but there is another problem. Transaction sandwiching does not inherently involve misleading statements. Furthermore, no deceptive devices are utilized to mislead ordinary traders. It is common knowledge that MEV extraction is prevalent on blockchains. DEX operators, for example, allow traders to specify a slippage limit [97], indicating the maximum price deviation they are willing to accept. Moreover, the broader DeFi market is unlikely to be misdirected by sandwiching transactions, as an ordinary user's transaction can either benefit from the sandwiching or become a sandwiched transaction itself.

Despite these complexities, Barczentewicz *et al.* propose two arguments to support a potential fraud-based manipulation liability for sandwiching that relies on dominance in transaction ordering. These involve proving price artificiality via market power dominance or conflict of interest. However, both arguments assume highly moralized courts and regulators, and given the cost of investigation and prosecution, the certainty of their application remains unclear.

Sandwiching private transactions, in contrast, is much more likely to result in liability under SEC Rule 10b-5 or CFTC Rule 180.1, provided it can be established that "(1) the MEV extractor had a 'pre-existing duty' with the user from whom they received [the private order flow], and (2) the MEV extractor breached this duty in the process of either extracting MEV from the user's POF transactions or by 'tipping' the user's transaction to another who then extracts MEV from the user's transaction." [14] For example, Flashbots has explicitly promised users that their transactions will remain private until they are included in a proposed block (although the risk of being orphaned still exists) and are thus resistant to MEV extraction if sent via Flashbots Protect. Users who send their transactions through private order flow services such as Flashbots Protect do so for the purpose of avoiding MEV extraction, typically in exchange for paying an additional fee or accepting a longer latency for transaction settlement. In this context, there is explicit trust from users towards these private order flow service providers. Therefore, if these providers sandwich users' transactions or sell them to a third party for sandwiching, all entities involved in such misconduct could potentially be held liable for fraud-based manipulation, or more specifically, insider trading.

While the classification of public and private transactions and the ensuing discussions about market manipulation liability as formulated by Barczentewicz *et al.* are well-articulated within their specific context of how Flashbots and Ethereum normally operate today, we contend that this framework may not be exhaustive due to the overwhelming focus on transaction publicness. This is particularly true when considering the evolving landscape of MEV mitigation strategies, and even in the context of the earlier discussed MEV-boost equivocation attack. A crucial aspect overlooked in the framework is that a breach of trust leading to fraud-based manipulation does not necessarily involve possession of non-public information. Specifically, when a transaction transitions from being private to public before being confirmed on chain, there may still be a breach of trust when sandwiching it. Thus, manipulation liability may be found without using the moralized reasoning as Barczentewicz *et al.* proposed. Additionally, the publicness condition is not comprehensive, which may lead to ambiguity or unintended outcomes in practical scenarios. We delve deeper into these complexities and propose a solution in the next section.

4 Implications of MEV Mitigation Designs

In this section, we suggest switching from Barczentewicz *et al.*'s focus on transaction publicness to respecting users' intent. We argue that this change better accommodates the evolving landscape of MEV extraction and mitigation. We also discuss some of the enforcement challenges, particularly given the decentralized nature of blockchains and issues inherent to the mitigation technologies.

4.1 Issues of Publicness Standard and Proposed Fix

There are two main concerns associated with a standard based on publicness: (1) the publicness condition does not comprehensively cover all practical scenarios, and (2) breach of trust is only considered in cases involving non-public information, despite the possibility of breaches in public transactions. Given these issues, applying the publicness standard for analyzing the legal liability of MEV extraction could lead to either ambiguous or undesired outcomes.

First, the publicness definition does not properly capture all plausible scenarios in today's MEV landscape. On the one hand, the definition is ambiguous as applied to transactions that are neither fully private nor fully public. Some transactions are partially visible in practice; mitigation strategies such as MEV-share and SUAVE allow users to choose only certain transaction information to disclose to MEV seekers. On the other hand, the definition may lead to undesired outcomes. In TEE-based blockchains such as Secret Network, private transactions may not be divulged from the original receiving node. Instead, encrypted transactions are propagated throughout the peer-to-peer network by protocol, and each registered TEE can access the decrypted transactions in their plaintext form. An actor may be able to access transaction details by interacting with their own TEE, either by exploiting design flaws or through side channel attacks targeting TEEs, but without having any special arrangement with the node that originally received the transaction.

Second, a trust relationship may exist and be breached at any stage of a transaction lifecycle, not just the period when the transaction is private. For instance, in the case of the MEV-boost equivocation attack, the attacker manipulated transaction ordering by first signing a block header to trick the relay into releasing transaction data, and then subsequently proposing an equivocating block to extract MEV. In this situation, the transactions initially deemed private became public before on-chain settlement, and the attacker sandwiched them after they became public. According to Barczentewicz *et al.*, the attacker targeted public transactions, so it would be hard to find market manipulation liability. However, we argue that even after users' transactions become public, it is still possible to identify fraud in such a case. In particular, if a relay provides blocks to validators only if they promise not to equivocate and extract MEV, that could suffice to establish a trust relationship between the relay and validators. Equivocation could be viewed as breach of that trust[9]. Similarly for fair ordering protocols, where transactions are publicly visible, validators are supposed to order transactions according to the pre-determined ordering policy (such as first-come-first-served) and not extract MEV. Here too it is possible to argue that there is beach of trust when validators sandwich public transactions.

Given these concerns, we propose a different rule for analyzing market manipulation liability of MEV extraction. Instead of focusing on the actual visibility of a transaction within the peer-to-peer network, we advocate for a perspective that respects *users' intent*. Specifically, if an actor profits from routing a user's transaction in a way contrary to the user's intentions or orders transactions using information not designated by the user for such purposes, this could lead to fraud-based manipulation liability. Since the prosecution bar is lower when pursuing fraud-based manipulation, the key questions are whether a trust relationship or duty exists between the sandwicher and the sandwichee, and whether that duty is breached. The focus of this approach is on the presence and potential breach of a trust relationship between MEV seekers and users. Our modification integrates Barczentewicz *et al.*'s concept of publicness by aligning it with the user's intended transaction routing, while also considering which aspects of transaction information the user consents to be used for ordering. The proposed rule not only upholds the liability analysis presented by Barczentewicz in specific cases they consider, but also extends its applicability to the more complex scenarios discussed in this paper. We delve into the challenges associated with proving fraud in the next two subsections.

4.2 Trust in the Decentralized Setting

As outlined by Barczentewicz *et al.*, holding MEV extractors liable under SEA §9(a)(2) or CFTC Rule 180.2 is a challenging proposition due to the difficulties in proving both the existence of an artificially manipulated price and the accused's specific intent. Fraud-based manipulation liability under SEC Rule 10b-5 and

[9] On May 15, 2024, as this article was going to press, the U.S. Department of Justice unsealed an indictment charging two defendants with wire fraud for the MEV-boost attack described in Sect. 2.1. Indictment, ECF No. 2, United States v. Peraire-Bueno, No. 1:24-cr-00293-UA (S.D.N.Y. filed May 8, 2024). The indictment alleges that the defendants made "material representations" by advertising lucrative "Lure Transactions" to attract transactions from MEV bots, and by transmitting a "False Signature" from a validator that they controlled (i.e., a signature that could not ultimately be validated for inclusion on the blockchain, but which would fool the relay into revealing private transaction data presented by the MEV bots to the relay).

The indictment's theories of falsity raise slightly different issues than we discuss. Lure transactions are best analyzed as a form of spoofing, which is already recognized as a form of manipulative conduct [91]. If the indictment is correct, the signature would be straightforwardly false, because it purports to be valid but is not. Our analysis of fraud-based manipulation applies to more general equivocation attacks. The indictment itself does not focus on the equivocation in the MEV-boost exploit, perhaps because these other theories of falsity were readily available.

CFTC Rule 180.1 emerges as a more feasible alternative, *if* a trust relationship between the users and the MEV extractor can be established. However, the concept of trust in a decentralized setting is complicated.

The trust relationship between users and a private order flow service provider such as Flashbots Protect is reasonably clear. Flashbots Protect makes specific claims about confidentiality, and users route transactions to it based on those representations. Violation of those claims, and misusing or disclosing users' transaction details other than as promised, could be a breach of trust.

However, it is empirically challenging to establish a trust relationship between blockchain users and a permissionless decentralized network of validators. Blockchains are fundamentally designed to inhibit misconduct via protocol enforcement; this is what makes them "trustless." For instance, most blockchains are constructed to be secure against a certain percentage of Byzantine participants who can deviate from the protocol at will. This design philosophy is followed by many new MEV mitigation strategies such as commit-reveal schemes and fair ordering protocols. Given the decentralized and Byzantine fault-tolerant nature of the protocol, anyone is theoretically capable of creating their own validator to participate. In the case of Ethereum, for example, there are multiple recommended implementations for validator software [45], yet only two contain explicit terms of use in their GitHub repositories [86,88], and none explicitly make any guarantees about avoiding MEV extraction.[10]

Instead of committing to running a specific version of software, an alternative may be to require validators, when joining the network or producing a block, to have a legally binding agreement on their alignment with the protocol specification. This requirement is not by itself in violation of decentralization, as validators can run any version of software, even one with their own modifications. However, given the current landscape and design philosophy of blockchains, establishing a legal commitment to MEV resistance for any validator may still be a difficult proposition, as shown in Flashbots' response to the MEV-boost equivocation attack. They initially added a blacklist of misbehaving validators, but subsequently removed it in pursuit of technical solutions [16]. This move could be attributed to concerns of potential censorship from the blockchain community, a point of contention that was earlier highlighted when Flashbots decided to comply with OFAC regulations [24].

Although the blockchain community has extensively debated whether to exclusively pursue technical solutions or to also embrace legal regulation, the absence of accountability in protocols for MEV resistance undeniably places extra strain on regulators. Even when fraud liability can be established, enforcement may prove challenging due to the decentralized and permissionless nature of validators. Hence, the incorporation of accountability within blockchain protocols is a pressing necessity for the evolution of the technology.

[10] There is a separate question of whether a blockchain user can rely on terms of use in a GitHub repository, but we leave that question for another day.

4.3 Failure Related to Trusted Execution Environment

Establishing fraud-based manipulation liability within TEE-based systems also presents challenges. In these systems, the trust relationship is not established between users and validators, but rather with the software development team and the underlying hardware it depends on. Thus whether a misbehaving validator can be held liable becomes questionable.

For instance, the Secret Network [93] promotes its confidential transaction execution and its resistance to MEV. However, a recent study [68] revealed substantial design flaws that entirely undermine the privacy assurances the network claims to provide. An actor operating as a validator, who has registered their own TEE within the Secret Network, can exploit these privacy vulnerabilities to access transaction details and subsequently extract MEV. To serve as a validator, an actor simply needs to download the source code and follow instructions to run it on suitable hardware with an embedded TEE. A TEE-based project typically has two types of code: trusted code and untrusted code. Trusted code is designed to run inside a TEE, providing guarantees of integrity and confidentiality. Additionally, the network can detect any modification to the trusted code, thanks to remote attestation. On the other hand, untrusted code operates outside the TEE and thus lacks such security guarantees. Its execution can be manipulated and its state can be probed. The Secret Network GitHub repository [92] clearly states that "the non-enclave code can be modified and ran on mainnet as long as there are no consensus-breaking changes." As the attack necessitates only modification of the untrusted code, the attacker does not breach any pre-existing trust or duties. In other words, it is a normal operation in the protocol and not a deceptive device. The standard fraud analysis applied to MEV extraction may not work in this scenario.

One may ask if bypassing the confidentiality of transactions inside a TEE using side-channel attacks could, alternatively, be seen as computer trespass under the Computer Fraud and Abuse Act (CFAA). Specifically, the use of a TEE establishes a code-based protective barrier, preventing external entities from accessing transaction details. On this theory, circumventing the protective function of a TEE might result in liability. However, given the Supreme Court's narrow interpretation of "unauthorized access" in *Van Buren v. United States* [1], courts are unlikely to find CFAA liability for side-channel attacks. These attacks rarely involve prohibited access as such; instead they are based on information inference from available resources.

5 Conclusion

In this paper, we have explored the current MEV landscape and existing mitigation strategies. We explored Barczentewicz et al.'s analysis, in which market manipulation liability for sandwiching transactions depends on those transactions' publicness, and identified limitations in that classification standard. We proposed a test for assessing manipulation liability that shifts the focus from network conditions to user intent. Our approach provides a more nuanced understanding of the dynamic lifecycle of transactions and accommodates the evolving landscape of MEV extraction strategies and mitigation measures, offering a more robust foundation for evaluating market manipulation liability.

We also described new enforcement challenges posed by the decentralized and permissionless nature of blockchain operations. Our research underscores the significant impact of the blockchain community's regulatory preferences and the crucial role of practical in-protocol accountability mechanisms in prosecuting MEV extraction. The recent equivocation attack incident highlights the inadequacy of Ethereum's incentive mechanisms, emphasizing the urgent need for accountable protocols and legal frameworks to reinforce the security and fairness of the ecosystem. By aligning technology with legal systems, we anticipate that MEV mitigation proposals can help illuminate the "dark forest" of the ecosystem.

Future research should continue to closely monitor the rapidly evolving MEV landscape. A comprehensive assessment of the harms and benefits of MEV, informed by well-defined metrics for user welfare, can significantly contribute to understanding its impact on the ecosystem.

A Technical Background

In this section, we will provide the essential technical background. First, we will introduce blockchain technology and its security features. Next, we will discuss smart contracts and demonstrate how they expand the range of potential applications on blockchains, with a focus on decentralized finance (DeFi). Finally, we will explain the concept of Miner/Maximal Extractable Value (MEV) and provide examples of how MEV can be exploited in practice.

A.1 Blockchain Technology

A blockchain is a public ledger composed of a linear chain of blocks,[11] each containing a series of sequentially ordered transactions submitted from users. These blocks are generated in a decentralized fashion by a group of validators, who are responsible for maintaining the integrity and security of the ledger. Blockchain technology offers security from several perspectives: (1) Immutability,[12] meaning that once a transaction is approved by validators, it will not be removed or

[11] There are a few exceptions, such as Avalanche and IOTA, that employ a DAG structure. The majority of blockchains in deployment today, however, are within the scope of discussion in this paper.

[12] There have been attempts in allowing mutability in history given validators' consensus on removal of certain illegal contents, but there is no such blockchain prevailing in the ecosystem yet.

altered, ensuring a temper-proof record; (2) Transparency,[13] allowing all participants in the network to verify if the execution result of all transactions included in the ledger is correct, promoting trust and accountability; (3) Decentralization, indicating that no single entity can take control of the system, enabling users who may not trust each other to transact reliably on the blockchain; and (4) Censorship-resistance, ensuring that even if a transaction is not favored by some entities, it will still be executed eventually as long as it is valid. Due to these features, blockchain technology offers a secure, transparent, and decentralized way to record and verify transactions, which fosters trust among users and enables various applications including asset management [28], supply chain tracking [10], digital identity management [66], and decentralized applications (dApps) [98].

A simplified depiction of the interactions between blockchain validators and users is presented in Fig. 2. In order to initiate a transaction, a user needs to send the transaction details to one or more validators. Validators, in turn, are responsible for disseminating valid transactions they receive to their peers, thereby creating a shared transaction mempool. A transaction is deemed valid if it satisfies specific criteria.[14] For example, the sender possesses an adequate amount of assets to be transferred to the recipient and the transaction is associated with a legitimate signature that confirms the asset owner's authenticity.

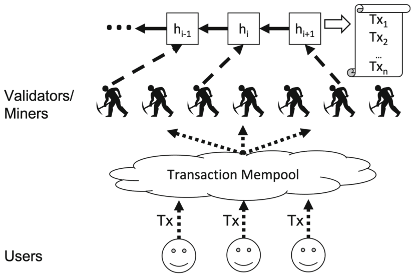

Fig. 2. Blockchain workflow.

The mempool serves as a temporary holding area for valid transactions awaiting execution and inclusion in the next block. In other words, transactions in the mempool do not have an immediate impact. Rather, they take effect-such as causing users' assets to be actually transferred-only after being incorporated into a block. Blocks are produced at a predetermined frequency, which may vary depending on the blockchain in question. For instance, the block interval of Bitcoin is roughly 10 min [18], while that of Ethereum is about 12 s [46].

[13] Transparency may vary in meaning for different blockchains, not necessarily requiring transactions to be publicly accessible. For example, there are rollups utilizing zero-knowledge proofs for public verification of the correct execution of transactions yet the transaction details are not needed.

[14] Another example: In account-based blockchains, users' sequential nonce also has to be incremental for preventing double executing the same transaction.

Valid transactions that enter the mempool during the corresponding interval have the potential to be included in the upcoming block. However, this inclusion is not guaranteed. Which transactions are selected for the next block can be impacted by several factors, such as network congestion, transaction fees, and the specific consensus mechanism at play. For instance, when the mempool holds more transactions than can be accommodated in a single block-due to each block having a block size limit, either measured by the size of transaction messages [17] or the computational resources for transaction execution as known as gas cost [56] - validators can only choose a subset of the pending transactions for the next block. The remaining transactions stay in the mempool until the network congestion subsides or they are eventually dropped. The block size limit arises from the fundamental trade-off between the scalability, security, and decentralization of blockchains [8], making it an unavoidable constraint. To encourage validators to prioritize the inclusion of their transactions, users can voluntarily offer transaction fees as an incentive [22]. By doing so, they increase the likelihood that their transactions will be processed more quickly, even during periods of high network congestion.

Validators rotate to take the responsibility of generating the next block and proposing it to the network for appendation to the blockchain. The selection of the block proposer depends on the underlying consensus protocol in use. In blockchains based on Proof-of-Work (PoW) [85], validators, commonly referred to as miners, consume substantial computational resources to compete in solving a cryptographically difficult puzzle. The winning miner who successfully solves the puzzle first earns the privilege to decide which transactions from the mempool to incorporate into the new block, and in return, is rewarded all the transaction fees associated with those transactions. On the other hand, Proof-of-Stake (PoS) blockchains [23,30] utilize a trustworthy source of randomness to appoint the validator responsible for proposing the next block [20]. The chosen validator is a member from a committee, each participant of which has previously committed a predetermined amount of assets as stake. In other words, a validator must lock up some native cryptocurrency as collateral and the chance of being selected to propose a new block is proportional to the validator's deposited stake. This approach contrasts with the resource-intensive nature of PoW, emphasizing the role of staked assets in securing the network, which is more energy efficient and environmentally friendly [67]. However, it is generally perceived as less secure than PoW [40]. To enhance security and accountability in practice, PoS blockchains are often facilitated with economic mechanisms to incentivize honest behavior among validators. For instance, one of the most commonly employed mechanisms is called slashing [42] - malicious validators are to be published by losing a portion of their staked assets, the proportion of which is determined by the type and severity of their misconduct.

As demonstrated above, although blockchains aim to promote decentralization in the long run, the generation of each block in most contemporary blockchains existing today is actually determined by a single validator. Consequently, decentralization does not have sufficient granularity at the individual block level, even though it is a fundamental aspect of the blockchain system as a whole. This key observation serves as the foundation for the discussion presented in this paper.

A.2 Smart Contracts and Decentralized Finance

While the most basic blockchain transactions are usually in the form of sending a specific amount of native cryptocurrency from one user to another,[15] many modern blockchains support more complex, Turing-complete[16] functionalities through the use of smart contracts. Smart contracts are essentially programmable scripts that reside on the blockchain and can be triggered by user transactions. They enable the automatic, atomic execution of functions with intricate logic, based on various conditions and corresponding actions. For example, consider the case of purchasing flight delay insurance [103]. Traditionally, claiming a refund from the insurance company could be time-consuming and require extensive paperwork. However, by utilizing a smart contract, the refund process can be automated, issuing the payout to the customer as soon as the flight delay is confirmed [105]. This eliminates the need for manual intervention, streamlining the entire process. Tornado Cash [99] is a more complex example of a smart contract application, which, although sanctioned by the OFAC, enables fully anonymous payments on the Ethereum blockchain.[17] Such a feature was not natively available on Ethereum without the use of smart contracts. By leveraging the power of smart contracts, Tornado Cash provides users with enhanced privacy in their transactions, while also raising concerns of being abused for illegal activities. In summary, smart contracts extend the capabilities of blockchains by incorporating programmable logic and automation, and enable various decentralized applications.

Decentralized Finance (DeFi) has been a rapidly growing sector in the industry largely due to the power of smart contracts. These trustless financial services circumvent traditional centralized intermediaries and foster transparency and accessibility, making DeFi increasingly appealing to traders.

Contrasting with traditional financial systems that depend on centralized operators such as Nasdaq or NYSE, DeFi features decentralized exchanges (DEXes) that operate using smart contracts. DEX users maintain full control of their assets at all times, and orders are settled directly and transparently on the blockchain without needing to trust an exchange operator or broker. In the early days, DEX users had to manually search for buy/sell orders and submit the bundled transaction on-chain for settlement [104]. However, modern DEXes, such as Uniswap v3 [96] - the largest DEX on Ethereum-now incorporate automated market makers (AMMs) [27] to streamline on-chain price discovery. This innovation enables traders to simply send their orders to the blockchain, which

[15] Bitcoin takes a slightly different semantic in the form of UTXO based.

[16] Although Turing-complete, the execution is guaranteed to halt due to the limit of gas.

[17] The anonymity comes from the mixer pool.

will be automatically settled by the exchange smart contract. Uniswap v3 boasts a 24-hour trading volume of around $290 million as of Sep 20, 2023 [34].

Another prominent application in DeFi is lending and borrowing platforms. These platforms allow users to lend their assets to earn interest or borrow assets by providing collateral. The process is managed by smart contracts, eliminating the need for users to place trust in a third party. Lending services contribute significantly to the DeFi ecosystem's efficiency by providing liquidity for all types of financial activities. The largest lending platforms, Aave [5] and JustLend [71], hold a combined total liquidity pool of over $9 billion in locked-up assets [39]. A unique innovation resulting from the atomic execution of smart contracts is the Flashloan [65], which enables borrowing without collateral. With Flashloans, a borrower can obtain a loan, utilize it, and return the assets to the lender all within a single atomic transaction. Since the smart contract guarantees full repayment (or the transaction reverts as if it never occurred), lenders face no risk in Flashloans.

Various types of assets can be represented and managed in DeFi with interoperability across multiple blockchains. For example, a user can trade Bitcoin for an equivalent amount of Wrapped Bitcoins (WBTCs) [95] on Ethereum using cross-chain bridging services, and then participate in diverse DeFi activities on Ethereum with WBTCs.

Gas tokens [57] serve as a unique example of a derivative contract on Ethereum, offering additional on-chain utility. Ethereum transaction fees are determined by computation and storage costs for validators, measured in units of gas. Each smart contract execution instruction incurs a predetermined gas cost, and the sender is required to pay a fee proportional to the gas usage by the transaction. However, gas prices can fluctuate due to network congestion. Storing data on a blockchain consumes gas, while deleting storage variables refunds gas. Users can mint gas tokens by storing data on-chain when the gas price is low (so is the transaction volume), then burning the storage and using the refunded gas to cover transaction fees when the price is high (usually due to network congestion or for the purpose of prioritizing an urgent transaction). Gas tokens can be viewed as futures contracts without expiration dates, and they played a significant role in the MEV game,[18] which we will discuss further in Sect. A.3.

In addition to digital assets and derivatives that are native to blockchains, smart contracts can tether off-chain assets with on-chain tokens for DeFi activities. Stablecoins [62], for instance, are backed by a reserve of assets, often in fiat currencies, at a 1:1 ratio to mitigate the volatility typically associated with cryptocurrencies. Another example is Non-fungible tokens (NFTs) which represent unique off-chain commodities that are not interchangeable, such as artwork [35], domain names [44], tickets and coupons [19]. Despite its non-fungible nature, an NFT can be converted to fungible assets by dividing the token into fractional shares and granting users partial ownership [58].

Overall, smart contracts facilitate a wide range of financial instruments and services, playing a crucial role in the expansion and development of the DeFi

[18] Gas token is now obsolete due to EIP-1559.

ecosystem. As the sector continues to evolve, it demonstrates the potential for a more accessible, transparent, and efficient financial landscape, underscoring the promising future of DeFi.

A.3 Miner/Maximal Extractable Value

Maximal Extractable Value (MEV), originally known as Miner Extractable Value, was introduced in a paper entitled "Flash Boys 2.0" by Daian *et al.* As previously mentioned, transactions are not settled in real-time but are first gathered in the mempool, which is publicly accessible. A validator then selects transactions and orders them in a block for execution. MEV exploitation refers to the practice of making profit through the strategic censorship, insertion and reordering of transactions within blocks. Validators used to be in the dominant position of determining what transactions are included in a block and how they are ordered, which is why the term was initially called Miner Extractable Value. This practice can have both positive and negative effects on the ecosystem, depending on how it is executed and the intentions of the actors involved.

DeFi is a complicated and interoperable ecosystem, so transactions from different users may have an impact on each other, and their execution results cannot be determined independently. Instead, they depend on their ordering. For example, an ordinary user wants to trade 40 WETHs for Example Dummy Tokens (EDTs) on Uniswap. Uniswap follows the rule $x \times y = k$ for a pair of exchange tokens [7], where x and y represent the amounts of WETH and EDT, respectively, and k is a predetermined constant. Let us assume $k = 1600$ and initially $x = y = 40$. After executing the user's transaction, we have $x = 80$ and $y = 20$, so the user receives 20 EDTs in return.

However, the execution result of this example transaction is not guaranteed. Once the transaction is sent to the public mempool, an MEV extractor can observe it and exploit the change in demand by creating a sandwiching transaction bundle for profit. In particular, the extractor first trades 120 WETHs for EDTs before the ordinary transaction. Based on the price invariant above, this frontrun transaction results in $x = 160$ and $y = 10$, with the extractor receiving 30 EDTs in return. Next, the sandwiched transaction is executed, leading to $x = 200$ and $y = 8$, with the ordinary user only getting 2 EDTs in return. At last, the extractor inserts a backrun transaction, trading the 30 EDTs for a profit of 38 WETHs, resulting in a loss of 18 EDTs for the ordinary user. As the saying goes in [12], "These two [inserted] transactions are the bread in the sandwich. [The ordinary user's] is the meat." It is worth noting that sandwiching attacks are not unique to Uniswap's price discovery mechanism, which was designed specifically for gas efficiency on blockchains. Sandwiching can also occur in traditional exchanges that employ limit order books [75]. In both cases, the sandwich attack takes advantage of the public nature of pending transactions to manipulate the order of trades and extract profits, potentially at the expense of other traders.

Because a transaction with potential MEV opportunity can only be sandwiched once, and anyone can create a sandwiching bundle upon an ordinary transaction in the public mempool, the MEV game is highly competitive. MEV extractors vie for the inclusion of their own sandwiching bundles by offering high transaction fees, incentivizing validators to prioritize their transactions. This leads to an implicit auction for transaction position within a block, which can be viewed as a scarce resource on blockchains. To win in the MEV auction, MEV extractors used to utilize gas tokens to save transaction fees while exploiting MEV, i.e., minting gas tokens when the gas price is low and then burning them to cover gas cost in transactions when the gas price is high. This was a common practice before Ethereum incorporated the EIP-1559 update [22], which introduced changes to the transaction fee mechanism. Nonetheless, the dynamics of MEV extraction has not changed significantly after a series of updates in Ethereum 2.0. While MEV exploitation has become more decentralized with a multi-stage, multi-entity MEV supply chain [49] introduced by Flashbots, the dominant position of validators in the game remains largely unchanged. Recent estimates suggest that over 90% of extracted MEV ultimately goes to validators, highlighting their continued influence in the ecosystem [12].

Sandwiching transactions is not the only way to extract MEV, and not all MEV exploitations are detrimental to the ecosystem. In fact, there exists "good MEV," which typically arises from arbitrage opportunities [47]. For instance, when a legitimate large trade is placed on one DEX, it may cause a significant price difference between that DEX and other DEXes for the same pair of tokens. An MEV extractor can take advantage of this price discrepancy by purchasing tokens at a lower price on one DEX and then selling them at a higher price on another DEX. In 2020, a savvy trader managed to generate a profit of $40K by exploiting the price discrepancy between two stable coins, USDC and USDT, leveraging the power of Flashloans for arbitrage opportunities [25]. This action does not harm any legitimate traders; instead, it helps to balance prices across DEXes, ensuring that they more accurately reflect the actual market price. MEV arbitrage is not considered price manipulation from a legal perspective because it does not create artificial prices. Instead, it plays a role in stabilizing markets and promoting price efficiency. Additionally, good MEV can contribute to the overall health of the ecosystem by reducing price discrepancies, fostering market equilibrium, and encouraging fair trading conditions for all participants.

Oracle manipulation presents yet another avenue for MEV extraction, offering potentially higher profits albeit with less frequent opportunities. DEXes and lending platforms often rely on price oracles to determine spot trading prices or collateral ratios [41]. Some on-chain oracles aggregate prices from multiple marketplaces and compute an average, which opens the door to arbitrage opportunities when prices on particular marketplaces are manipulated or do not accurately represent the actual market price. A notable instance of this strategy is the Mango Market exploit. Oracle manipulation can often yield higher profits than transaction sandwiching, as it does not require a counterparty and can be easily leveraged [11]. However, this form of MEV extraction can technically be more readily mitigated through improved oracle design [79] and risk monitoring measures implemented by exchanges - a lesson learned from the traditional financial market [26].

References

1. Van Buren v. United States, 141 S. Ct. 1648 (2021)
2. Drexel Burnham Lambert Inc. v. Commodity Futures Trading Commission, 850 F.2d 742 (D.C. Cir. 1988)
3. Commodity Futures Trading Commission v. Parnon Energy, 875 F. Supp. 2d (S.D.N.Y. 2012)
4. Lummis-Gillibrand Responsible Financial Innovation Act, S._, 118th Cong. (2023)
5. Aave. Aave liquidity protocol (2023). https://web.archive.org/web/20230920221835/https://aave.com/. Accessed 20 Sept 2023
6. Abrantes-Metz, R.M., Rauterberg, G., Verstein, A.: Revolution in manipulation law: the new cftc rules and the urgent need for economic and empirical analyses. U. Pa. J. Bus. L. **15**, 357 (2012)
7. Adams, H., Zinsmeister, N., Salem, M., Keefer, R., Robinson, D.: Uniswap v3 core. Technical report, Uniswap (2021)
8. Altarawneh, A., Herschberg, T., Medury, S., Kandah, F., Skjellum, A.: Buterin's scalability trilemma viewed through a state-change-based classification for common consensus algorithms. In: 2020 10th Annual Computing and Communication Workshop and Conference (CCWC), pp. 0727–0736. IEEE (2020)
9. Asayag, A., et al.: A fair consensus protocol for transaction ordering. In: 2018 IEEE 26th International Conference on Network Protocols (ICNP), pp. 55–65. IEEE (2018)
10. AWS. Blockchain for Supply Chain: Track and Trace (2023). https://web.archive.org/web/20230920195444/https://aws.amazon.com/blockchain/blockchain-for-supply-chain-track-and-trace/. Accessed 20 Sept 2023
11. Babel, K., Daian, P., Kelkar, M., Juels, A.: Clockwork finance: automated analysis of economic security in smart contracts. In: 2023 IEEE Symposium on Security and Privacy (SP), pp. 2499–2516. IEEE (2023)
12. Babel, K., Ji, Y., Juels, A., Kelkar, M.: PROF: fair transaction-ordering in a profit-seeking world (2023). https://web.archive.org/web/20230920225103/https://initc3org.medium.com/prof-fair-transaction-ordering-in-a-profit-seeking-world-b6dadd71f086. Accessed 20 Sept 2023
13. Barczentewicz, M.: Mev on ethereum: a policy analysis. ICLE White Paper, pp. 01–23 (2023)
14. Barczentewicz, M., Sarch, A., Vasan, N.: Blockchain transaction ordering as market manipulation. Ohio St. Tech. Law J. **20**, 1 (2023)
15. beaconcha.in. Relay Overview - Open Source Ethereum Blockchain Explorer (2023). https://web.archive.org/web/20230920235153/https://beaconcha.in/relays. Accessed 20 Sept 2023
16. Bert. Post mortem: April 3rd, 2023 mev-boost relay incident and related timing issue (2023). https://web.archive.org/web/20230921001800/https://collective.flashbots.net/t/post-mortem-april-3rd-2023-mev-boost-relay-incident-and-related-timing-issue/1540. Accessed 20 Sept 2023
17. Bitcoin Magazine. What is the bitcoin block size limit (2020). https://web.archive.org/web/20230920212208/https://bitcoinmagazine.com/guides/what-is-the-bitcoin-block-size-limit. Accessed 20 Sept 2023
18. Bitcoin Wiki. Block intervals (2014). https://web.archive.org/web/20230920204907/https://en.bitcoin.it/wiki/Block_intervals. Accessed 20 Sept 2023

19. BloXmove. NFTicket (2023). https://web.archive.org/web/20230920224650/https://bloxmove.com/technology/nfticket. Accessed 20 Sept 2023
20. Brown-Cohen, J., Narayanan, A., Psomas, A., Weinberg, S.M.: Formal barriers to longest-chain proof-of-stake protocols. In: Proceedings of the 2019 ACM Conference on Economics and Computation, pp. 459–473 (2019)
21. Buterin, V.: Two-slot proposer/builder separation (2021). https://web.archive.org/web/20230807101311/https://ethresear.ch/t/two-slot-proposer-builder-separation/10980?u=barnabe. Accessed 07 Aug 2023
22. Buterin, V., Conner, E., Dudley, R., Slipper, M., Norden, I., Bakhta, A.: Eip-1559: Fee market change for eth 1.0 chain. Published online (2019)
23. Buterin, V., Griffith, V.: Casper the friendly finality gadget. arXiv preprint arXiv:1710.09437 (2017)
24. Carreras, T.: 51% of Ethereum Blocks Can Now Be Censored. It's Time for Flashbots to Shut Down (2022). https://web.archive.org/web/20230921020027/https://cryptobriefing.com/51-of-ethereum-blocks-can-now-be-censored-its-time-for-flashbots-to-shut-down/. Accessed 20 Sept 2023
25. Cawrey, D., Sinclair, S.: First Mover: How a DeFi Trader Made an 89% Profit in Minutes Slinging Stablecoins (2021). https://web.archive.org/web/20230920231351/https://www.coindesk.com/markets/2020/08/12/first-mover-how-a-defi-trader-made-an-89-profit-in-minutes-slinging-stablecoins/. Accessed 20 Sept 2023
26. CFTC. CFTC Market Surveillance Program (2023). https://web.archive.org/web/20230920232458/https://www.cftc.gov/IndustryOversight/MarketSurveillance/CFTCMarketSurveillanceProgram/index.htm. Accessed 20 Sept 2023
27. Chainlink. What Are Automated Market Makers (AMMs) (2021). https://web.archive.org/web/20230920220828/https://chain.link/education-hub/what-is-an-automated-market-maker-amm. Accessed 20 Sept 2023
28. Chainlink. The Future of Asset Management Using Smart Contracts and Blockchain Oracles (2023). https://web.archive.org/web/20230920195133/https://blog.chain.link/the-future-of-asset-management-using-smart-contracts-and-blockchain-oracles/. Accessed 20 Sept 2023
29. Chakravarty, M.M.T., Chapman, J., MacKenzie, K., Melkonian, O., Peyton Jones, M., Wadler, P.: The extended UTXO model. In: Bernhard, M., Bernhard, M., et al. (eds.) FC 2020. LNCS, vol. 12063, pp. 525–539. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-54455-3_37
30. Chen, J., Micali, S.: Algorand. arXiv preprint arXiv:1607.01341 (2016)
31. Hager, C.: [@metachris]. flashbots/mev-boost-relay[commit: 84a9439] (2023). https://web.archive.org/web/20230921003137/https://github.com/flashbots/mev-boost-relay/commit/84a943925e62f20b812c60688b6e433fba8e0da7. Accessed 20 Sept 2023
32. Hager, C.: [@metachris]. flashbots/mev-boost-relay[pr: #338] (2023). https://web.archive.org/web/20230921003921/https://github.com/flashbots/mev-boost-relay/pull/338. Accessed 20 Sept 2023
33. Ciampi, M., Ishaq, M., Magdon-Ismail, M., Ostrovsky, R., Zikas, V.: Fairmm: a fast and frontrunning-resistant crypto market-maker. In: International Symposium on Cyber Security, Cryptology, and Machine Learning, pp. 428–446. Springer, Heidelberg (2022). https://doi.org/10.1007/978-3-031-07689-3_31
34. CoinGecko. Uniswap V3 (Ethereum) (2023). https://web.archive.org/web/20230920221522/https://www.coingecko.com/en/exchanges/uniswap-v3-ethereum. Accessed 20 Sept 2023

35. Coursera. What Is NFT Art? (How Does It Work) (2023). https://web.archive.org/web/20230920224409/https://www.coursera.org/articles/nft-art. Accessed 20 Sept 2023
36. U.S. Supreme Court. Morrison v. National Australia Bank Ltd. (2010)
37. Daian, P.: MEV... wat do (2021). https://web.archive.org/web/20221219142049/https://pdaian.com/blog/mev-wat-do/. Accessed 20 Sept 2023
38. Daian, P., et al.: Flash boys 2.0: frontrunning in decentralized exchanges, miner extractable value, and consensus instability. In 2020 IEEE Symposium on Security and Privacy (SP), pp. 910–927. IEEE (2020)
39. DefiLlama. All Protocols: Total Value Locked (2023). https://web.archive.org/web/20230920222250/https://defillama.com/. Accessed 20 Sept 2023
40. Deirmentzoglou, E., Papakyriakopoulos, G., Patsakis, C.: A survey on long-range attacks for proof of stake protocols. IEEE Access **7**, 28712–28725 (2019)
41. Dzyatkovskii, A.: The Strengths and Weaknesses of DeFi Price Oracles (2021). https://web.archive.org/web/20230920231609/https://hackernoon.com/the-strengths-and-weaknesses-of-defi-price-oracles-x7l35ui. Accessed 20 Sept 2023
42. Edgington, B.: Upgrading Ethereum: Slashing (2023). https://web.archive.org/web/20230920214103/https://eth2book.info/capella/part2/incentives/slashing/. Accessed 20 Sept 2023
43. Emmenegger, S., Zuber, F.: To infinity and beyond: U.S. dollar-based jurisdiction in the U.S. sanctions context. In: Swiss Review of Business and Financial Market Law, pp. 114–130 (2022)
44. ENS. Decentralised naming for wallets, websites, & more (2023). https://web.archive.org/web/20230920224534/https://ens.domains/. Accessed 20 Sept 2023
45. Ethereum. Staking Launchpad: Validator checklist (2023). https://web.archive.org/web/20230921020410/https://launchpad.ethereum.org/en/checklist. Accessed 20 Sept 2023
46. Etherscan. Ethereum Average Block Time Chart (2023). https://web.archive.org/web/20230920205201/https://etherscan.io/chart/blocktime. Accessed 20 Sept 2023
47. Finoa. The role of MEV in DEX arbitrage (2023). https://web.archive.org/web/20230920231201/https://www.finoa.io/blog/mev-arbitrage/. Accessed 20 Sept 2023
48. Flashbots. The Future of MEV is SUAVE (2022). https://web.archive.org/web/20230921005814/https://writings.flashbots.net/the-future-of-mev-is-suave/. Accessed 20 Sept 2023
49. Flashbots. The MEV Supply Chain: a peek into the future of this industry (2022). https://web.archive.org/web/20230920230907/https://flashbots.mirror.xyz/bqCakwfQZkMsq63b50vib-nibo5eKai0QuK7m-Dsxpo. Accessed 20 Sept 2023
50. Flashbots. Flashbots Doc (2023). https://web.archive.org/web/20230920194204/https://www.flashbots.net/. Accessed 20 Sept 2023
51. Flashbots. Flashbots Protect: Quick Start (2023). https://web.archive.org/web/20230920235910/https://docs.flashbots.net/flashbots-protect/quick-start. Accessed 20 Sept 2023
52. Flashbots. MEV-Boost Risks and Considerations (2023). https://web.archive.org/web/20230920234627/https://docs.flashbots.net/flashbots-mev-boost/architecture-overview/risks. Accessed 20 Sept 2023
53. Flashbots. MEV-Explore v1: Pre-merge Data (2023). https://web.archive.org/web/20230920184439/https://explore.flashbots.net/. Accessed 20 Sept 2023

54. Flashbots. What is MEV-Boost (2023). https://web.archive.org/web/20230920234424/https://docs.flashbots.net/flashbots-mev-boost/introduction. Accessed 20 Sept 2023
55. Flashbots. Flashbots Transparency Dashboard: REV activities since the Merge (2024). https://web.archive.org/web/20240510024731/https://transparency.flashbots.net/. Accessed 09 May 2024
56. Frankenfield, J.: Gas (Ethereum): How Gas Fees Work on the Ethereum Blockchain (2022). https://web.archive.org/web/20230920212352/https://www.investopedia.com/terms/g/gas-ethereum.asp. Accessed 20 Sept 2023
57. Breidenbach, L., Daian, P., Tramèr, F.: GasToken (2018). https://github.com/projectchicago/gastoken
58. Genç, E.: How Can You Share an NFT? Fractional NFTs Explained (2023). https://web.archive.org/web/20230920224744/https://www.coindesk.com/learn/how-can-you-share-an-nft-fractional-nfts-explained/. Accessed 20 Sept 2023
59. Goodin, D.: SGX, Intel's supposedly impregnable data fortress, has been breached yet again (2022). https://web.archive.org/web/20230921004825/https://arstechnica.com/information-technology/2022/08/architectural-bug-in-some-intel-cpus-is-more-bad-news-for-sgx-users/. Accessed 20 Sept 2023
60. Gosselin, S., Chiplunkar, A.: The Orderflow Auction Design Space (2023). https://web.archive.org/web/20230921001333/https://frontier.tech/the-orderflow-auction-design-space. Accessed 20 Sept 2023
61. Hager, C., Paape, F.: Block Building inside SGX (2023). https://web.archive.org/web/20230921004108/https://writings.flashbots.net/block-building-inside-sgx. Accessed 20 Sept 2023
62. Hayes, A.: Stablecoins: Definition, How They Work, and Types (2023). https://web.archive.org/web/20230920223845/https://www.investopedia.com/terms/s/stablecoin.asp. Accessed 20 Sept 2023
63. Heimbach, L., Wattenhofer, R.: Eliminating sandwich attacks with the help of game theory. In: Proceedings of the 2022 ACM on Asia Conference on Computer and Communications Security, pp. 153–167 (2022)
64. Lioba Heimbach and Roger Wattenhofer. Sok: Preventing transaction reordering manipulations in decentralized finance. arXiv preprint arXiv:2203.11520, 2022
65. Hertig, A.: What is a flash loan (2023). https://web.archive.org/web/20230920222954/https://www.coindesk.com/learn/what-is-a-flash-loan/. Accessed 20 Sept 2023
66. IBM. Blockchain for digital identity and credentials (2023). https://web.archive.org/web/20230920200735/https://www.ibm.com/blockchain-identity. Accessed 20 Sept 2023
67. Jain, A., Arora, S., Shukla, Y., Patil, T., Sawant-Patil, S.: Proof of stake with casper the friendly finality gadget protocol for fair validation consensus in ethereum. Int. J. Sci. Res. Comput. Sci. Eng. Inf. Technol. **3**(3), 291–298 (2018)
68. Jean-Louis, N., et al.: Sgxonerated: finding (and partially fixing) privacy flaws in tee-based smart contract platforms without breaking the tee. Cryptology ePrint Archive (2023)
69. Juels, A.: Fair Sequencing Services: Enabling a Provably Fair DeFi Ecosystem (2020). https://web.archive.org/web/20230921010521/https://blog.chain.link/chainlink-fair-sequencing-services-enabling-a-provably-fair-defi-ecosystem/. Accessed 20 Sept 2023

70. Juels, A., Eyal, I., Kelkar, M.: Miners, front-running-as-a-service is theft (2021). https://web.archive.org/web/20230920235654/https://www.coindesk.com/markets/2021/04/07/miners-front-running-as-a-service-is-theft/. Accessed 20 Sept 2023
71. JustLend. JustLend and Borrow in the Decentralized Platform (2023). https://web.archive.org/web/20230920222849/https://portal.justlend.org/. Accessed 20 Sept 2023
72. Kelkar, M., Deb, S., Kannan, S.: Order-fair consensus in the permissionless setting. In: Proceedings of the 9th ACM on ASIA Public-Key Cryptography Workshop, pp. 3–14 (2022)
73. Kelkar, M., Deb, S., Long, S., Juels, A., Kannan, S.: Themis: fast, strong order-fairness in byzantine consensus. In: Proceedings of the 2023 ACM SIGSAC Conference on Computer and Communications Security, pp. 475–489 (2023)
74. Kelkar, M., Zhang, F., Goldfeder, S., Juels, A.: Order-fairness for byzantine consensus. In: Micciancio, D., Ristenpart, T. (eds.) CRYPTO 2020. LNCS, vol. 12172, pp. 451–480. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-56877-1_16
75. Kenton, W.: What is a limit order book? definition and data (2022). https://web.archive.org/web/20230920225847/https://www.investopedia.com/terms/l/limitorderbook.asp. Accessed 20 Sept 2023
76. Knight, O.: Ethereum Bot Gets Attacked for $20M as Validator Strikes Back (2023). https://web.archive.org/web/20230921001518/https://www.coindesk.com/business/2023/04/03/ethereum-mev-bot-gets-attacked-for-20m-as-validator-strikes-back/. Accessed 20 Sept 2023
77. Kocher, P., et al.: Spectre attacks: exploiting speculative execution. Commun. ACM **63**(7), 93–101 (2020)
78. Kursawe, K.: Wendy, the good little fairness widget: achieving order fairness for blockchains. In Proceedings of the 2nd ACM Conference on Advances in Financial Technologies, pp. 25–36 (2020)
79. Mackinga, T., Nadahalli, T., Wattenhofer, R.: Twap oracle attacks: easier done than said. In: 2022 IEEE International Conference on Blockchain and Cryptocurrency (ICBC), pp. 1–8. IEEE (2022)
80. McKeen, F., etal.: Intel® software guard extensions (intel® sgx) support for dynamic memory management inside an enclave. In: Proceedings of the Hardware and Architectural Support for Security and Privacy 2016, pp. 1–9 (2016)
81. Neuder, M.M., Drake, J.: Why enshrine Proposer-Builder Separation? A viable path to ePBS (2023). https://ethresear.ch/t/why-enshrine-proposer-builder-separation-a-viable-path-to-epbs/15710
82. Miller, A., Xia, Y., Croman, K., Shi, E., Song, D.: The honey badger of bft protocols. In: Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, pp. 31–42 (2016)
83. Monnot, B.: Notes on Proposer-Builder Separation (PBS) (2022). https://web.archive.org/web/20230920234155/https://barnabe.substack.com/p/pbs. Accessed 20 Sept 2023
84. Gupta[@Mudit__Gupta], M.: That patch only patches a small bug but would've been irrelevant for the attack. The economic incentive is such that proposers will still manipulate the block. 25m profit for 1.8k slash. The vulnerability is in the design. X (2023). https://web.archive.org/web/20230921002806/https://twitter.com/Mudit__Gupta/status/1642873195475922946. Accessed 20 Sept 2023
85. Nakamoto, S.: Bitcoin whitepaper (2008). https://bitcoin.org/bitcoin.pdf. Accessed 17 July 2019

86. Nethermind[@NethermindEth]. Nethermind Ethereum client[version: aad88ee] (2023). https://web.archive.org/web/20230921021857/https://github.com/NethermindEth/nethermind. Accessed 20 Sept 2023
87. Pmcgoohan. MEV... do this (2021). https://web.archive.org/web/20230920235415/https://pmcgoohan.medium.com/mev-do-this-beb2754bca63. Accessed 20 Sept 2023
88. Prysm Ethereum Client[@prysmaticlabs]. Prysm: An Ethereum Consensus Implementation Written in Go[version: e76aedf] (2023). https://web.archive.org/web/20230921021611/https://github.com/prysmaticlabs/prysm. Accessed 20 Sept 2023
89. Reiter, M.K., Birman, K.P.: How to securely replicate services. ACM Trans. Program. Lang. Syst. (TOPLAS) **16**(3), 986–1009 (1994)
90. Sarkar, A.: Flashbots builds over 82% relay blocks, adding to Ethereum centralization (2022). https://web.archive.org/web/20230920234759/https://cointelegraph.com/news/flashbots-build-over-82-relay-blocks-adding-to-ethereum-centralization. Accessed 20 Sept 2023
91. Scopino, G.: Algo Bots and the Law: Technology, Automation, and the Regulation of Futures and Other Derivatives. Cambridge University Press, Cambridge (2020)
92. SCRT Labs[@scrtlabs]. Secret Network[version: f739659] (2023). https://web.archive.org/web/20230921021212/https://github.com/scrtlabs/SecretNetwork. Accessed 20 Sept 2023
93. Secret Network. Private Smart Contract on the Blockchain (2023). https://web.archive.org/web/20230921005955/https://scrt.network/about/about-secret-network. Accessed 20 Sept 2023
94. Espresso Systems. The espresso sequencer (2023). https://hackmd.io/@EspressoSystems/EspressoSequencer
95. Tran, K.C.: What is Wrapped Bitcoin (2022). https://web.archive.org/web/20230920223149/https://decrypt.co/resources/what-is-wbtc-explained-bitcoin-ethereum-defi. Accessed 20 Sept 2023
96. Uniswap. Introducing Uniswap v3 (2021). https://web.archive.org/web/20230920215730/https://blog.uniswap.org/uniswap-v3. Accessed 20 Sept 2023
97. Uniswap. Slippage Protection (2023). https://web.archive.org/web/20230921014544/https://uniswapv3book.com/docs/milestone_3/slippage-protection/. Accessed 20 Sept 2023
98. Upptic. Web3 Gaming (2023). https://web.archive.org/web/20230920201012/https://upptic.com/blog/web-3-gaming/. Accessed 20 Sept 2023
99. U.S. Department of the Treasury. U.S. Treasury Sanctions Notorious Virtual Currency Mixer Tornado Cash (2022). https://web.archive.org/web/20230920215017/https://home.treasury.gov/news/press-releases/jy0916. Accessed 20 Sept 2023
100. Van Bulck, J., et al.: Foreshadow: extracting the keys to the intel {SGX} kingdom with transient {Out-of-Order} execution. In: 27th USENIX Security Symposium (USENIX Security 18), pp. 991–1008 (2018)
101. Velasquez, F.: DeFi Exchange Mango's $114M Exploit Was 'Market Manipulation,' Not a Hack, Ex-FBI Special Agent Says (2022). https://web.archive.org/web/20230921011756/https://www.coindesk.com/tech/2022/10/20/defi-exchange-mangos-114m-exploit-was-market-manipulation-not-a-hack-ex-fbi-special-agent-says/. Accessed 20 Sept 2023
102. Weintraub, B., Torres, C.F., Nita-Rotaru, C., State, R.: A flash (bot) in the pan: measuring maximal extractable value in private pools. In: Proceedings of the 22nd ACM Internet Measurement Conference, pp. 458–471 (2022)

103. Whitmore, G.: Will You Purchase Blockchain Flight Delay Insurance (2022). https://web.archive.org/web/20230920214542/https://gum.criteo.com/syncframe?origin=publishertag&topUrl=www.forbes.com. Accessed 20 Sept 2023
104. Wikipedia. 0x (decentralized exchange infrastructure) (2023). https://web.archive.org/web/20230920215515/https://en.wikipedia.org/wiki/0x_%28decentralized_exchange_infrastructure%29. Accessed 20 Sept 2023
105. Zhang, F., Cecchetti, E., Croman, K., Juels, A., Shi, E.: Town crier: an authenticated data feed for smart contracts. In: Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, pp. 270–282 (2016)
106. Zhang, Y., Setty, S., Chen, Q., Zhou, L., Alvisi, L.: Byzantine ordered consensus without byzantine oligarchy. In: 14th USENIX Symposium on Operating Systems Design and Implementation (OSDI 20), pp. 633–649 (2020)
107. Zhou, L., Qin, K., Gervais, A.: A2mm: mitigating frontrunning, transaction reordering and consensus instability in decentralized exchanges. arXiv preprint arXiv:2106.07371 (2021)
108. Züst, P., Nadahalli, T., Wattenhofer, Y.W.R.: Analyzing and preventing sandwich attacks in ethereum. ETH Zürich (2021)