The Advent of a New Incentive - Maximal Extractable Value

Burak Öz[1], Felix Hoops[2]

Abstract

In public, permissionless blockchains, incentives play an essential role in the evolution of a network. Without sufficient incentives, people will not be interested in contributing to a network. Misalignment of incentives is dangerous as well. Given sufficient alternative incentives, network participants can attempt attacks that affect consensus stability. This could go as far as centralizing the entire network around a small set of vertically integrated actors. Until the explosion in decentralized finance protocols in the summer of 2020, the main incentives on Ethereum were the block reward and the transaction fees. That summer, it became clear that there is an extra value on top of these incentives, which protocol members like miners can permissionlessly extract by reordering, censoring, and including transactions in a block. In this chapter, we introduce the advent of this new incentive, known as Maximal Extractable Value (MEV), and discuss its impact on Ethereum.

Keywords:
Incentives, Consensus Stability, Vertical Integration, Maximal Extractable Value, Ethereum

JEL Classification:
Lorem Ipsum…

[1] Technical University of Munich
[2] Technical University of Munich

Table of Content

<a> Introduction

Information has always had value in finance. Knowing about an upcoming product launch might allow someone to buy into a company early. A politician, knowing about an upcoming regulatory change or an adjustment to financial policy before it is publicly announced, could benefit from buying or selling a corresponding company's stock. A brokerage employee might be aware of a trade in advance and use that to improve his own trading strategy. In all of these scenarios, an actor has information about an impending value shift in the market. And, critically, that information is not publicly available and thus constitutes insider information. Acting on such information, be it directly or indirectly, is insider trading. After much debate in the early 20th century, also on the fundamental question of whether insider trading is to be viewed negatively at all, the practice is now widely outlawed in traditional markets and regarded as a form of market manipulation.

Among the initially described forms of insider trading, some are based on circumstantial, i.e., external information, while others are based on market information. Those latter ones are of interest in the context of decentralized finance, as blockchains change the workings of the market, not the externalities. Frontrunning and backrunning are two simple instances of insider trading based on pure market information. The former means executing a trade before a targeted market order, and the latter afterwards. In both cases, the aim is to profit from the resulting shift in value of the underlying asset. To maximize profit, that market order ideally moves a large quantity of value. Consider knowing about an upcoming large scale purchase of a stock: knowing that it will drive up market value, one can buy the same stock before that purchase executes, netting an instant unrealized return on that position. This scenario is considered frontrunning. Selling any held position in the stock directly after the targeted market order executes to realize as much of the value shift as possible is considered backrunning.

Financial author Michael Lewis helped publicize the problem through his book “Flash Boys, A Wall Street Revolt” (2014) about high-frequency trading. The title comes from the term “flash order”[3] describing the practice of showing a market order to a closed circle of actors shortly before listing it publicly. While this is legal in the US, critics like Lewis view this as a form of frontrunning and, thus, market manipulation.

In contrast to traditional finance, public blockchains and their pending transactions are freely accessible by anyone. Thus, market information has been democratized, requiring a fresh look at insider trading strategies from traditional finance. After all, can it be considered malicious insider trading, if there are no insiders? This re-evaluation began back in 2019 when researchers published their findings on automated and problematic trading strategies on Ethereum (Daian et al., 2019). They paid tribute to Lewis's book by titling their paper “Flash Boys 2.0,” suggesting that, like traditional finance, decentralized finance might not be as free and democratized as it is made out to be. Among other things, they discovered that Ethereum miners had an even more important role in the system than initially intended. As block proposers, miners are in the

[3] https://www.sec.gov/news/press/2009/2009-201-factsheet.htm

unique position to have the final say about what transactions are executed as part of the block and in what order. Consequently, they have all the power to legally frontrun, backrun, or even replace any transaction. For a simple value transfer, such as a payment for a good or a service, that does not constitute a problem. However, with the adoption of smart contract-enabled blockchains, led by Ethereum, and the subsequent rise of decentralized exchanges (DEXes), the block proposer's power position became problematic. They were now in an unfair position to extract value from the blockchain in a way that was never intended. This value is bundled under the term “Miner Extractable Value'' or MEV for short. Arguably, MEV is the most important contribution of the “Flash Boys 2.0” paper. It introduces an incentive shift at the consensus level, which has wide-reaching consequences for the whole blockchain ecosystem.

Trivial MEV strategies include replacing transactions that leverage an arbitrage opportunity with a functionally identical transaction that secures the value of that opportunity for the block proposer. That scenario is possible because these strategies, such as arbitrage trading, have no dependency on whom they are executed by. Anyone seeing such a transaction is able to replicate it for themselves. This move would effectively steal this value from the original transaction creator.

To better understand this concept, let us consider a more sophisticated attack. Traders in decentralized finance usually express their price tolerance on a market order via a slippage tolerance, which is set as a percentage of the expected price. If the market order cannot be executed within the designated slippage, it is dropped. Thus, setting slippage too low can lead to missing out on trades that would have still been lucrative, but just outside the slippage tolerance. On the other hand, setting it too high opens up lucrative frontrunning and backrunning opportunities. Someone could frontrun the transaction in a carefully calculated way that ensures pushing the price of the asset in question to the maximum within the slippage tolerance, and then immediately backrun it to secure a profit seemingly out of thin air. This is called a sandwich attack, because it sandwiches the targeted transaction between the attacker's transactions. And critically, it is labeled as an attack because it actively harms the creator of that targeted transaction. As we will discuss in more detail in chapter <MEV Forms>, MEV is not necessarily always extracted at the expense of another market actor.

At the time of publishing of “Flashboys 2.0,” all the described phenomena or strategies were largely theoretical, with little realization in practice. But it would not take long for their theories to be proven right: during what is now known as “DeFi summer 2020,” bots working with strategies described in the paper, starting with simple arbitrage trades, were first confirmed in the real world. Confirmed as at least partially realistic in its raised concerns, “Flashboys 2.0” has since served as a starting point for many researchers to try and understand these phenomena better. One of the first findings was that miners seemingly avoided manipulating block contents for their own gain. That is most likely due to them realizing that it is vital that users know, or at least trust, that the miner will not use their transaction data against them. If, on any blockchain, users do not feel comfortable to send their transactions to a miner, that chain is doomed to be abandoned.

However, that did not deter others from analyzing the pending transactions in the Ethereum transaction pool, using bots to find unrealized value that they can claim for themselves. Why do long-winded, strategy-based analysis of the Ethereum block data if you can just wait for someone to find unrealized value and copy their transaction? Ultimately, that led to constant bidding wars, where large numbers of bots tried to get their transaction committed to the chain by offering higher transaction fees to the miners. These fees could get as high as comprising 90 percent of the value realized within a given transaction. In the end, miners still profited off these value opportunities without directly manipulating transaction order or inclusion.

Later on, it was discovered that blockchains not using Proof-of-Work (PoW), and consequently not having miners, also exhibited similar value opportunities. That led to the renaming MEV from Miner Extractable Value to Maximal Extractable Value. Alongside, the term's definition shifted. While there is no one true definition that we could cite here, the Ethereum community's understanding considers all value that can be realized during block creation by manipulating transaction order or inclusion apart from intended rewards, being the block reward and gas fees.

Understanding MEV is not only relevant to realizing value for oneself, but most importantly to understand and mitigate its effects on a blockchain ecosystem. It is realistic to assume that at one point in the not-so-far future, MEV will significantly outpace block rewards, making it the most important incentive for block proposers. This development would shift a proposer's primary priority from providing a stable service to the network to optimizing for their own gain, potentially at the cost of others. Their behavior might go as far as reorganizing blocks as part of a history rewrite.

We have come to understand that how competition for MEV opportunities is organized is essential. It could lead to centralization and the death of entire blockchains. For example, joining mining or validator pools might become even more advantageous, as these pools have more resources that they can use more effectively to improve their MEV extraction. As such, figuring out how to organize competition for MEV fairly, and effectively democratize MEV in the process, is a big priority of MEV focused research groups and organizations, such as Flashbots.

MEV remains a continued topic of debate in the whole decentralized finance community and beyond as it poses a centralization risk that every smart contract-enabled blockchain is susceptible to. In the following chapter, we will explore different forms of MEV in greater detail and shine some light on the future of MEV-aware blockchain development. To this end, we first look at different forms of MEV in section <MEV Forms>. Once we have defined what MEV is, we expand on its emergence in Ethereum, and how it was dealt with over time in section <MEV Timeline>. Then, we take a step back from Ethereum and discuss the existence of MEV in Bitcoin in section <MEV in Bitcoin>, before coming to a close with some concluding remarks and a look to the near future of MEV.

<a> MEV Forms

MEV emerges as a result of a state transition on a blockchain. Although not every state transition exposes MEV, like a simple Ether transfer between two addresses, transactions that interact with smart contracts, especially DeFi contracts, create leverageable opportunities. These opportunities are usually discovered and captured by users, called “searchers,” who have scripts monitoring the Ethereum memory pool (mempool).[4] The most common MEV forms captured by searchers are arbitrages, liquidations, and sandwiches. More secretive forms of MEV that yield more significant profits compared to the common forms also exist. Such MEV opportunities are known as MEV long tail.

<b> Arbitrages

The most common form of MEV is arbitrage, where one tries to exploit the listing price difference of an asset on different exchanges by the simultaneous purchase and sale of the asset. An arbitrage is a win-win situation for both the arbitrageur and the utilized exchanges, as the arbitrageur balances the prices across them while capturing profits. If arbitrage MEV is not captured, price discrepancies will occur among exchanges, and the DeFi market space will be disrupted. Hence, arbitrage MEV is contributing to the existence of DeFi. Moreover, as arbitrages do not pose any damage to another actor on the network, they are classified as *good/benign* MEV.

Compared to executing an arbitrage trade on a traditional marketplace, doing arbitrage on Ethereum is significantly less risky due to the *sequential* and *atomic* execution properties of the Ethereum Virtual Machine (EVM). In a traditional marketplace, when one tries to leverage an arbitrage opportunity, the market price may move against him during the execution of the trade, thus, making him earn less money than planned or, sometimes, lose money. In Ethereum, transactions are executed sequentially, based on their order in a block. Thus, the market cannot move against an arbitrageur while his arbitrage transaction is being executed.

Moreover, in Ethereum, a transaction either gets fully executed or not at all. Let us imagine a transaction that calls a smart contract, which then triggers a sequence of calls to other contracts. During this sequence of interactions, if one of the interacted contracts throws an error, maybe because a condition check failed, the whole execution will get reverted. For the execution to be successful, every interaction triggered by the initial call must be accomplished without errors.

The sequential and atomic execution properties of Ethereum transactions open the door for many profitable DeFi strategies, including *atomic arbitrages*. An atomic arbitrage executes every hop of the arbitrage in a *single* transaction, and it is only realized under user-favored conditions and reverted otherwise. In layman's terms, the arbitrageur can

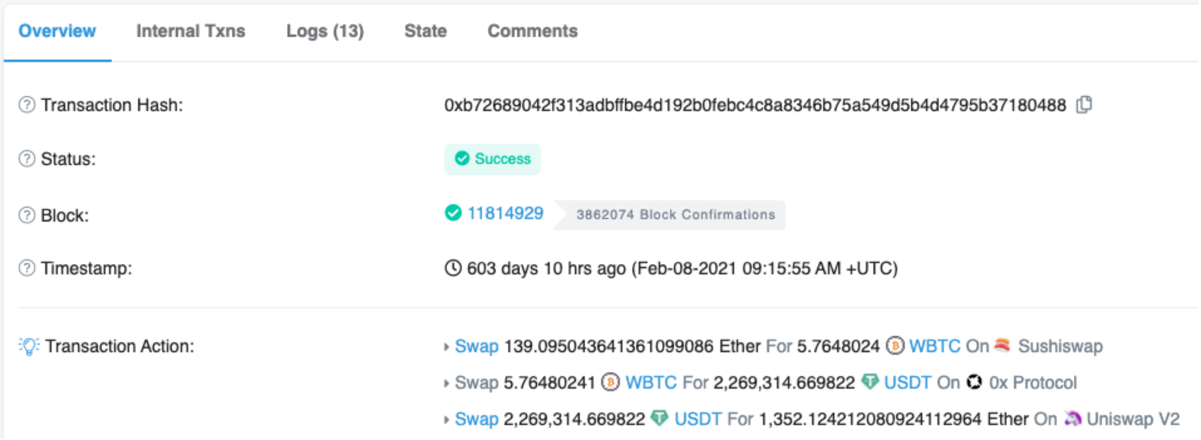
[4] Memory pool, or mempool for short, refers to the area where unconfirmed Ethereum transactions wait to be included in a block.

say that he only wants to execute this transaction if his end balance is greater than his starting balance. This way, he can guarantee that the arbitrage will only be performed if he makes a profit.

Figure XX shows the details of an atomic arbitrage transaction. The arbitrageur first swaps 139.09 ETH for 5.76 wrapped Bitcoin (WBTC) on Sushiswap. Then, he swaps 5.76 WBTC for 2,269,314 USDT stablecoins on 0x Protocol. Finally, he swaps the USDT for 1,352.12 ETH on Uniswap V2. The net profit of the arbitrageur from this triangular arbitrage is 2,124,457 USD.

- α: Starting Balance = 139.09 ETH
- β: Ending Balance = 1,352.12 ETH
- γ: Transaction Fee = 0.044 ETH

$$(\beta - \alpha - \gamma) \cdot 1,752\ (ETH/USD\ \ price\ when\ the\ transaction\ took\ place) = \$\ 2,124,457$$



Overview | Internal Txns | Logs (13) | State | Comments

Transaction Hash: 0xb72689042f313adbffbe4d192b0febc4c8a8346b75a549d5b4d4795b37180488

Status: Success

Block: 11814929 3862074 Block Confirmations

Timestamp: 603 days 10 hrs ago (Feb-08-2021 09:15:55 AM +UTC)

Transaction Action:
- Swap 139.095043641361099086 Ether For 5.7648024 WBTC On Sushiswap
- Swap 5.76480241 WBTC For 2,269,314.669822 USDT On 0x Protocol
- Swap 2,269,314.669822 USDT For 1,352.124212080924112964 Ether On Uniswap V2

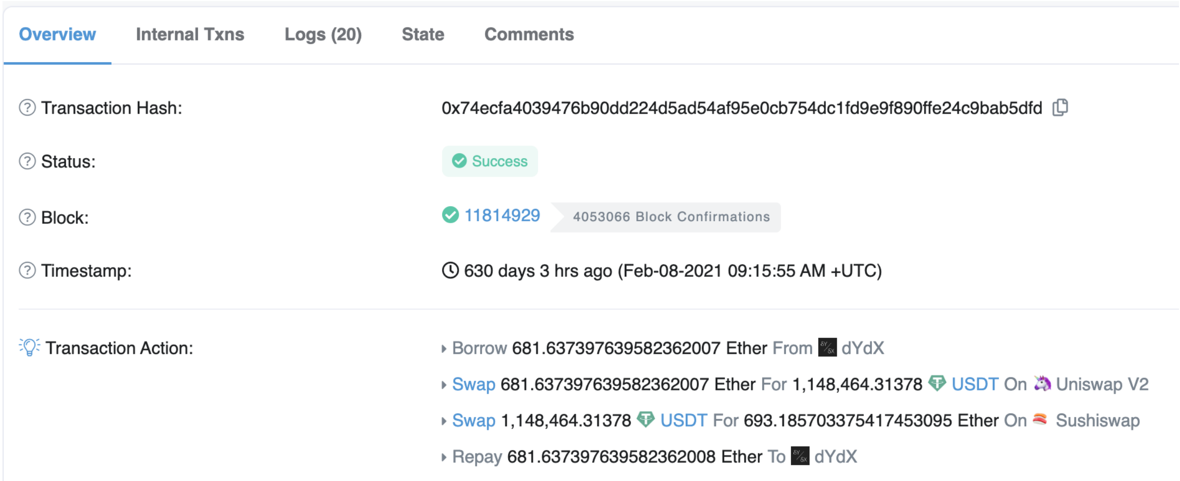
Details of an atomic arbitrage transaction. Screenshot taken from Etherscan.io.

Every arbitrage opportunity has an optimal input amount, and investing less than the optimal amount means leaving profits on the table. On the other hand, investing excessively will cause giving up on the profits. While the latter presents a problem that traders can solve by themselves, to solve the former, they need loans.

Flash loans, one of the most convenient forms of loans on DeFi, refers to a DeFi exclusive instrument in which the lending protocol lends money to a borrower without asking for any collateral, but under a single condition: the loan is paid back in the same transaction that it was borrowed. Lending protocols usually have a specific loan fee percentage (e.g., Aave 0.09%) that is applied to the borrowed amount to make this deal profitable for themselves. The fee must also be paid when the loan is paid back.

A smart contract of the lending protocol runs condition checks to ensure that the issued flash loan plus the fee is paid back. If any of the conditions fail, an error occurs in the contract, which causes the borrower's transaction to revert (remember that Ethereum transactions are atomic). Thus, the exchange goes back to the pre-transaction state where the lending protocol has not issued the loan yet. However, when the borrower pays back his loan, he can profit from an arbitrage opportunity to its full extent, as he can borrow the necessary amount of Ether. Currently, lending protocols issue flash loans worth millions of dollars.

In Figure XX, an atomic arbitrage that uses a flash loan to exploit an opportunity between Uniswap V2 and Sushiswap is shown. The arbitrageur initially takes a 681.63 ETH flash loan from the dYdX protocol. Then, he swaps the ETH for 1,148,464 USDT on Uniswap V2. Finally, he exchanges the USDT for 693.18 ETH on Sushiswap. After paying back his loan to dYdX, the arbitrageur is left with 11.55 ETH.



Transaction Hash: 0x74ecfa4039476b90dd224d5ad54af95e0cb754dc1fd9e9f890ffe24c9bab5dfd

Status: Success

Block: 11814929 4053066 Block Confirmations

Timestamp: 630 days 3 hrs ago (Feb-08-2021 09:15:55 AM +UTC)

Transaction Action:

- Borrow 681.637397639582362007 Ether From dYdX
- Swap 681.637397639582362007 Ether For 1,148,464.31378 USDT On Uniswap V2
- Swap 1,148,464.31378 USDT For 693.185703375417453095 Ether On Sushiswap
- Repay 681.637397639582362008 Ether To dYdX

Details of an atomic arbitrage transaction that uses a flash loan. Screenshot taken from Etherscan.io.

According to MEV-Explore,[5] a public dashboard that displays metrics about MEV extracted on Ethereum, more than 3 million arbitrage transactions have been issued through different DeFi protocols on Ethereum since January 2020. These transactions have extracted around 650M USD MEV in total. Although already a significant amount, it covers only some of the arbitrage MEV space. Currently, MEV-Explore only covers atomic arbitrages and not any CEX-DEX hybrid arbitrages, as CEX transactions are held off-chain and thus not publicly available.

<b> Liquidations

Like in traditional finance, loans also exist in DeFi. A user can take out a loan from a lending protocol by depositing an asset as collateral. However, in DeFi, lending protocols usually require the collateral to have a value higher than the borrowed amount, known as *over-collateralization*. This requirement is necessary for DeFi, as the borrower would otherwise have no incentive to pay back his loan.

[5] https://explore.flashbots.net/

As the prices fluctuate, the value of collateral often drops below a certain threshold determined by the protocol. When that happens, protocols often look out for users to liquidate the loan position in return for the collateralized asset plus a liquidation fee. Since liquidations enable lending protocols to keep their solvency, they are critical for DeFi, like arbitrages.

Qin et al. (2022) analyze the liquidations on four popular lending protocols (Aave, Compound, dYdX, and MakerDAO) until April 2021. They detected a total profit of 63.59M USD through 28,138 liquidations. Qin et al. identify 2,011 unique addresses capturing liquidation MEV, with each address making 31.62K USD profit on average. To capture the profits, liquidators take flash loans and track oracle price updates to foresee a liquidation position opening.

<b> Sandwiches

Ethereum nodes store validated but unconfirmed transactions into a public memory pool until they are included in a block. As transactions are transparent, anyone running a node can simulate them and observe their state changes. By simulating a transaction that executes a swap on a DEX such as Uniswap or Sushiswap, one can understand its price impact and profit from it. It is similar to insider trading, where one knows non-public information beforehand and takes action according to it. However, in this case, everyone has access to the same information. For example, suppose one knows that a particular swap transaction will increase the price of the XYZ token on a DEX. In that case, one can profit from it by buying XYZ tokens right before the swap gets executed (frontrunning) and then selling them immediately afterward (backrunning). This move is called sandwiching, as the initial swap transaction is sandwiched between two other transactions (i.e., the buy and sell orders).

At first glance, sandwich MEV seems harmful as the frontrunning buy order of the MEV searcher causes the sandwiched trade to take place with the worst price possible for the victim (depending on the set price slippage or maximum willingness to pay). However, from an economic perspective, sandwiches can also be considered beneficial as they increase the system efficiency by forcing the sandwiched users to pay their maximum willingness for that trade. If the MEV searchers do not execute sandwiches, there will be a consumer surplus, and the system will not be in equilibrium.

<b> MEV Long Tail

Although arbitrages, liquidations, and sandwiches are the most common forms of MEV, the real profit lies in the long tail, as there is less competition than in standard forms. Compared to detecting an arbitrage, looking out for an undiscovered MEV form is much more time-consuming and complex. Some strategies that long-tail MEV searchers adopt include exploring newly launched protocols and developing complex call sequences that interact with multiple protocols.

<a> MEV Timeline

<b> The Emergence of MEV in Ethereum (2019 - 2020)

The official Ethereum developer documents define MEV as the maximum value that can be extracted from block production by including, re-ordering, and censoring transactions in a block.[6] Although PoW-Ethereum miners always had these capabilities, MEV extraction only became prominent once DeFi protocols became popular in 2020. Until then, Ethereum blocks were mostly filled with simple money transfer transactions that did not expose any extractable value. With the growing DeFi ecosystem like DEXes, lending protocols, and automated market makers (AMMs), more ways for users to interact on Ethereum emerged.

<c> Flashboys 2.0 and The Dark Forest

In the seminal paper “Flashboys 2.0,” Daian et al. (2019) analyze the security threats exposed on Ethereum by DEX design flaws. Namely, they study the emerging bot community to capitalize on the arbitrage opportunities. They identify destructive actions such as frontrunning, initially explored by Eskandari et al. (2019), demonstrated by these bots to exploit the market. Furthermore, Daian et al. show that, for the miners, there is an extra value on top of the existing block reward and transaction fees that can be earned from the protocol by manipulating transactions in a block. They coined the term “Miner Extractable Value” (MEV) to represent this value.

When Daian et al. identified MEV, the threats it posed for Ethereum were mostly theoretical, since not much activity occurred on the application layer. In the summer of 2020, MEV became a *reality* when DEXes gained traction. During this period, arbitrage bots became highly active as many opportunities were exposed.

While arbitrage bots competed with each other to be the first to execute an arbitrage, as uncovered by Robinson & Konstantopoulos (2020) and Escaping the Dark Forest (2020), there was yet another type of bot monitoring the mempool and frontrunning *any profitable transaction.* A profitable transaction is any transaction that makes the ending balance of the issuer higher than the starting balance. These profit-seeking bots are called *generalized frontrunners*. Until private mempool solutions like Flashbots RPC were implemented, the bots were a real threat to any profit-leveraging transaction in the Ethereum public mempool. Since getting detected by a generalized frontrunner meant getting frontrun, thus losing profits, the Ethereum public mempool was compared to the dark forest from the famous science fiction author Liu Cixin's book “The Dark Forest,” in which getting detected in the forest meant instant death by the predators (Robinson & Konstantopoulos).

[6] https://ethereum.org/en/developers/docs/mev/

<c> Priority Gas Auctions

In PoW-Ethereum, a rational, utility-maximizing miner is expected to build his block based on the fee per gas transactions in the mempool offer.[7] Knowing this incentive, when MEV bots race agains teach other to be the first to capture a specific opportunity, they use the gas fee as a bidding mechanism to prioritize their transactions. As the mempool is public, each bot can monitor the bids (transactions) submitted by others and repetitively try to outbid everyone until the block is mined. These bidding interactions for block space are known as “Priority Gas Auctions" (PGA) (Daian et al., 2019).

PGAs have negative externalities for everyone except the winner, and in some cases, even for the winner. As PGAs resolve while the bidding is ongoing, losing parties usually do not get a window to submit a 0 gas fee transaction that would cancel their competitive transaction. Hence, they pay fees for profitless transactions, as the winner already captures MEV.[8] Sometimes, the gas fees are driven up so much that the revenue of the winner is eaten by the transaction cost, leading to negative-sum outcomes (Daian et al.). Furthermore, as PGAs congest the network traffic with spam transactions that fill up the block space, they lead to a bad user experience, such as longer waiting times and volatile gas prices for Ethereum users.

<c> The Impact on the Consensus Stability

MEV is an incentive that can, in value, exceed the block reward and transaction fees for the miners or, more generally, block proposers. However, block proposers are mainly hesitant to capture MEV directly due to reputation concerns. Nevertheless, they can issue any MEV-capturing transaction and censor the original one. For example, a miner can duplicate the transaction that wins a PGA while still claiming all the fees.

According to Daian et al. (2019), sufficient incentives can motivate the miners to attempt forking attacks, impacting the consensus stability of Ethereum. One example of such an attack is *undercutting attacks*, which occur when a miner ignores the latest block and forks the longest chain, capturing some of the MEV available while leaving a significant amount to incentivize the next miner to build on top of his block instead of the existing head of the chain which left less MEV to be captured.

Another MEV-incentivized attack on consensus stability is *time-bandit attacks*. These attacks exploit MEV available on past blocks (not only the latest block, like undercutting attacks do) by re-organizing the chain history. To execute such an attack, a miner needs to hold 51 percent of the hashing rate of the network, which can be remarkably expensive depending on the duration of the attack. For example, rewriting the last ten blocks would be cheaper than rewriting the last 1,000 blocks. Miners can subsidize these attacks by retroactively organizing the transactions in the past blocks to realize as much of MEV as possible.

[7] Gas is the unit to measure the computational work done to execute a transaction.

[8] PGAs are a mix of English and All-pay auctions in which the price ascends through repetitive bids, finally resolving to one winning bidder, while all participants are paying a cost (not only the winner).

<b> Enter Flashbots (2020 - September 2022)

In the summer of 2020, the MEV crisis reached its peak since the launch of Ethereum. Increasing numbers of MEV bots, frequently occurring priority gas auctions, complex smart contract interactions, and generalized frontrunning activities were all indicators that the MEV game was drifting Ethereum towards a dystopia where network congestion, limited block space, and unstable consensus were the reality. *Flashbots* emerged from the darkness during those days to “frontrun the MEV crisis.”[9] They are a research group consisting of blockchain researchers, white hat hackers, and developers to produce solutions to create a *transparent*, *democratized*, and *fairly distributed* MEV ecosystem. Although their work is mainly focused on Ethereum, as it is the most prominent stateful, smart contract blockchain, the solutions they develop can also be applied to other stateful chains.

<c> MEV Market Design and the Flashbots Auction

While MEV represents an existential threat for blockchains, how it is extracted from the determining factor regarding its impact. If block proposers decide to extract all the value and vertically integrate with trading firms, it leads to private transaction flows and centralization, undermining the core properties of a blockchain like Ethereum: *decentralization*, *permissionlessness*, and *transparency*. As in PGAs, if MEV extraction takes place publicly in a repetitive bidding format, it leads to network congestion and an inefficient block space allocation. Thus, there must be a market design that enables democratic access to MEV by all types of participants, not only the powerful ones, while not imposing any drawbacks to the user experience of the rest of the network.

Perhaps the first market design to come to mind consists of ordering transactions based on their arrival time, known as First Come, First Served (FCFS). Although FCFS will mitigate the adverse effects of PGAs, like volatile gas prices or network congestion through transaction spamming, it is not a silver bullet. FCFS can introduce similar detrimental impacts to blockchains as High-Frequency Trading (HFT) does on traditional finance marketplaces. Namely, latency wars. To have the most negligible latency, MEV searchers will be incentivized to invest in robust server racks, much like PoW miners investing in mining hardware. Moreover, searchers will be interested in co-locating with the most powerful block proposer and submitting their transactions directly to them. Thus, FCFS can easily lead to the compromise of decentralization.

Another way to realize MEV extraction can be by ordering transactions based on their hash values. While this can mitigate volatile prices, as paying higher gas fees will not impact a transaction's priority, it will also incentivize MEV searchers to spam the network until they land a transaction with the desired hash value. As block space will be filled up with spam transactions, the user experience will suffer from long waiting times.

[9] https://medium.com/flashbots/frontrunning-the-mev-crisis-40629a613752

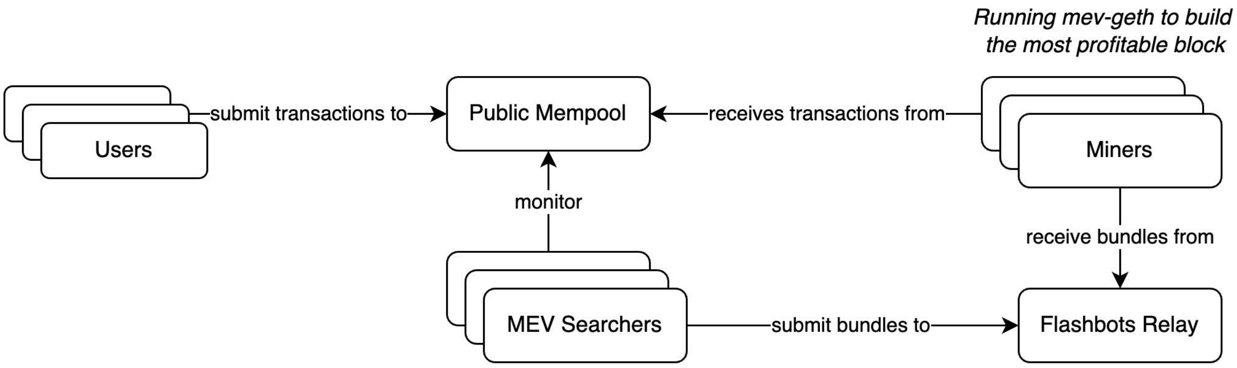
While PGAs signify one way to implement block space auctions for MEV extraction where searchers bid for order priority, better designs are also possible. Flashbots developed one such design for Ethereum called "The Flashbots Auction," a sealed-bid auction mechanism for block space realized through a private memory pool. The Flashbots Auction shifts the block space auction from on-chain, like PGAs do, to off-chain to avoid affecting everyday users with volatile gas prices and block space allocation by spam transactions. It enables MEV searchers to submit nondisclosed valuations for their preferred transaction order while preserving the privacy of the transactions against frontrunning attacks.

The Flashbots Auction introduces an option for MEV searchers to express their strategies in a more fine-grained manner compared to regular transactions, called *bundles*. A bundle is a set of transactions executed in the order in which they are placed in the bundle. Transactions in a bundle can be a mix of new searcher transactions and existing transactions from the public transaction pool. Bundles are useful for MEV searchers, as they can ensure their MEV extracting transactions are executed before or after their target transaction(s). For example, suppose a searcher aims to sandwich a swap transaction. In that case, he can include his frontrunning transaction (A), the swap transaction (B), and his backrunning transaction (C) in a bundle and guarantee that the transactions will be executed in the order *A->B->C*, given that he wins the auction.

A bundle submitted to the Flashbots Auction only gets revealed to the public once its bid wins the auction and lands in a block. Bundles submitted by losing bidders are never revealed with this mechanism, as they are not included in block building. As a result, no payment is required for them, unlike PGAs. Moreover, as bids are submitted to a private pool, congestion on the public network caused by MEV bots is partially overcome.

In the initial design of the Flashbots Auction, which was used in PoW-Ethereum, Flashbots operates a relay that controls the private transaction pool to which MEV searchers submit their bundles. Although each miner could have run its endpoint to receive bundles, this may have led to Denial-of-Service (DoS) attacks executed directly on the miner side. The Flashbots relay mitigates these problems by handling DoS attempts and checking the submitted bundles' validity as a first-line of defense against spam. It also prevents private relay centralization, as searchers can be incentivized to work with miners with more hashing power.

To receive bundles from the Flashbots relay, PoW-Ethereum miners run a modified version of the Go Ethereum (Geth) client called “mev-geth.” The modified client enables miners to effectively calculate a value for each submitted bundle and compute the most profitable block through the block space auction. Miners running mev-geth earn extra profits from the block space auction through direct payments by the searchers on top of the block reward and transaction fees.



An overview of the Flashbots Auction architecture.

The Flashbots Auction design inherently introduces trust assumptions to both the Flashbots relay and the miners, as they have access to the bundles' content and can potentially censor them. In a worst-case scenario, Flashbots can duplicate profitable bundles submitted to the relay and forward only their bundles to the miners while censoring the original searcher bundles. If miners act maliciously, they can also steal the MEV that Flashbots attempted to steal from the searchers.

However, defecting from honest strategies is not beneficial in the long run for any parties, perhaps even more for the miners than Flashbots. In the case of Flashbots, one can argue that the game is played on their premises, under surveillance. Thus they are already defecting from honest/decentralized strategies. However, more aggressive strategies like censoring or MEV stealing can lead to users losing trust in Flashbots products. For the miners, acting maliciously would cause them to lose access to the Flashbots relay, thus abandoning MEV profits.

Before Ethereum stopped running Proof-of-Work, around 90 percent of the miners utilized the mev-geth software. This adoption rate indicates how prominent of an incentive MEV is and why enabling democratic access to it is critical. Without a mechanism like the Flashbots Auction, MEV could have been an even more centralizing factor in Ethereum, where few dominant players extract all the value since they have access to private transaction flows, resulting in economies of scale, incentivizing history rewriting attacks.

<c> Flashbots Protect

The primary enabler of MEV strategies like frontrunning or sandwiching is that the target transaction is publicly visible. If the MEV-exposing transaction is submitted to a block proposer directly, assuming the user has an agreement with the block proposer, it will bypass the public memory pool. Thus it cannot be detected by MEV searchers anymore. However, this is only a feasible option for some users, as there is no existing market for users to make deals with block proposers.

“Flashbots Protect”[10] is a private transaction pool offered by Flashbots to protect Ethereum users against frontrunning and other similar MEV strategies. Users can configure their wallets to submit their transactions to the Flashbots Protect endpoint and avoid getting exposed. Compared to the Flashbots Auction, this is a more straightforward way for users to bypass the public transaction pool, as no bidding logic is required.

Flashbots Protect provides failed transaction protection by only including transactions that do not revert. An Ethereum transaction can revert for various reasons, including insufficient gas. In that case, as the block proposer has to execute the failing transaction until it runs out of gas, which consumes his energy, the transaction issuer is still required to pay the transaction fee (used gas amount * fee per unit of gas). Flashbots Protect saves costs for users by not making them pay for failing transactions.

The main reason a solution like Flashbots Protect works lies in the reputation of Flashbots, and the trust users have in them. Without such a trust assumption, it would be too risky for a user to submit his transaction to a private transaction pool, as there is no guarantee that the pool operator will not abuse the information the transaction exposes or censor it. Hence, the usefulness of such centrally controlled products directly depends on the operating party’s reputation, which, in a sense, contradicts the trust-freeness of a public, permissionless blockchain like Ethereum.

<c> MEV Detection and Quantification

Detection and quantification of MEV are critical for analyzing its impact. It can help us to understand the following:

- What kind of protocol design leads to more MEV exposure?
- Which MEV strategies are most prominent?
- What percent of the block proposer profits are provided through MEV extracting transactions?
- How profitable is it to run MEV bots?
- What are the costs incurred on targets of MEV strategies like sandwiching?

However, quantifying MEV is a complex task. Known MEV forms like arbitrages, sandwiches, and liquidations can be detected to an extent through scripts running pattern detection algorithms on transaction traces. Nonetheless, detecting every MEV extracting transaction is not a trivial task as there is a long tail of MEV strategies. Besides the extracted MEV, a theoretically extractable value that was not realized, such as manipulating transactions in a block by the block producer, also exists. Hence, any attempt at quantification can only constitute a *lower bound* to the actual maximal extractable value.

[10] https://docs.flashbots.net/flashbots-protect/overview

"MEV-Inspect"[11] is a script offered by Flashbots to detect and quantify arbitrages and liquidations that occurred on Ethereum. Although it has certain limitations, such as not handling multi-transaction MEV extraction opportunities like sandwiches or non-atomic arbitrages or not covering all protocols, it still achieves a lower bound for the total extracted value on Ethereum. This script is also the backend for the MEV metrics dashboard of Flashbots, *MEV-Explore*.

According to MEV-Explore, between *January 1, 2020*, and *September 15, 2022* (the last day of the PoW-Ethereum), a cumulative sum of 675.52M USD gross profit has been made by MEV searchers through atomic arbitrages and liquidations on nine prominent DeFi protocols.[12] During this period, 3,037,218 MEV transactions were successfully executed (99.11% arbitrages, 0.89% liquidations), which paid around 240M USD (36% of the searcher profits) to the miners in the form of transaction fees or direct payments through the Flashbots Auction. Miners also earned 1.42M USD from the gas fees paid by failing MEV transactions, such as any MEV transaction that reverted because it ran out of gas or deliberately stopped execution because the MEV opportunity vanished.

Although MEV-Explore only captures part of the picture of maximally extractable value available on Ethereum, it still succeeds in determining a lower bound for the extracted value on specific protocols. Using this information, everyday Ethereum users can understand which protocols lead to more MEV exposure. Furthermore, by looking at the searcher and miner profits, one can understand how profitable it is to run an MEV bot, how much of the extracted MEV stays with the searchers, and how prominent of an incentive MEV is compared to block reward and transaction fees.

<c> MEV Influenced Supply Chain

In Ethereum or any other blockchain, a chain of operations is undertaken when a user attempts to express his will. A simple example can be Alice wanting to swap her *WETH* tokens for *USDC* on Uniswap. To realize her intention, Alice must first interact with a wallet that would provide her with an interface to submit a transaction. Once the transaction is issued, it will be delivered to a public or private transaction pool, depending on the wallet, where it will await to be included in a block. During this period, the transaction will be spotted by an MEV searcher and placed in a bundle that extracts all the MEV it exposes. The searcher will submit this bundle to a block builder, who will then construct a block, including the bundle, and offer it to a validator, who will finally propose the block and reach a consensus over it with the rest of the validators.[13] This network of components which are included until Alice's transaction gets confirmed, or in other words, until her will gets accomplished, constitute a *supply chain* that operates under the influence of MEV.

[11] https://github.com/flashbots/mev-inspect-py
[12] Aave, Balancer, Bancor, Compound, Cream, Curve, Uniswap V2, Uniswap v3, and 0x
[13] In most blockchains, builders and validators are the same entity (e.g., miners in PoW-Ethereum). However, there are also designs where these roles are separated (e.g., sequencers in Layer-2 solutions, Proposer-Builder-Separation in Ethereum).

The MEV-influenced supply chain is a framework to better understand the different components included in transaction processing and how the value flows from users to validators. An overview of this network can give hints about the centralizing effects of MEV on a blockchain. In a dystopian world, components like wallets, searchers, builders, and validators are vertically integrated to capture most of the value. For example, a highly skilled searcher can make an off-chain deal with a wallet to get its transaction load in return for some percentage of the MEV those transactions expose. In another case, the searcher can integrate with certain builders and validators to guarantee preferred execution order in return for some profits again.

Vertical integration on the MEV supply chain leads to economies of scale in extracting/controlling MEV, which drives off the competition and creates centralized entities that dictate the user experience. Blockchains with such non-modular, centralized supply chains trade off transparency, censorship resistance, and being permissionless for opacity, exclusiveness, and non-democratic/permissioned access to MEV.

In a utopian world, an open market exists for each separate role in the supply chain, where individual entities compete. In such a world, fair competition drives off centralization and leads to a modular supply chain that is necessary for preserving a permissionless, uncensored, transparent blockchain. Furthermore, in a utopia, MEV does not accumulate only on the searchers or validators but also flows back to the users.

As MEV gets exposed after a particular state transition occurs, let us call this an “order;” the user that triggers this transition is the owner of the order flow. Without access to order flow, neither searchers nor builders can extract MEV. Thus, each order has a value depending on the MEV opportunity it exposes. Users can profit from the extractable value their orders expose by auctioning off the execution rights to searchers and builders. This way, MEV will benefit originators, i.e., users, as well as extractors.

<b> MEV Post Ethereum Merge (September 2022 - Today)

In September 2022, Ethereum stopped running Proof-of-Work (PoW) and longest-chain-based consensus (i.e., Nakamoto Consensus) and merged with the Beacon Chain,launched in December 2020, which runs a Proof-of-Stake (PoS) based consensus mechanism. The merge imposed a new architecture on Ethereum that separates the consensus layer (CL) and the execution layer (EL). While previously both CL and EL were baked into the same network, two separate networks exist for them after the merge.

Post-merge, the PoW-Ethereum network became the execution layer of Ethereum. The responsibilities of the EL include collecting and executing transactions, running the EVM, and preparing block payloads for the validators. Ethereum separated EL from CL in order to become a more modular protocol, as opposed to being a monolith. The modularity allows Ethereum to switch the current EL with another protocol, such as a Layer-2 rollup network if desired.

The Beacon Chain is Ethereum's new consensus engine, where block proposers, called "validators," are running PoS. While PoS determines who can be included in block production, validators are also running LMD Ghost (Latest Message Driven Greediest Heaviest Observed SubTree) and Casper FFG (Friendly Finality Gadget) to decide on the correct state of the chain.

- LMD GHOST: A weight/vote-based fork choice tool for finding the head of the chain.
- Casper FFG: A finality gadget which requires more than ⅔ of the total stake to vote on a block for it to be finalized eventually.

In PoW-Ethereum, the block time depended on how fast the miners could solve the mining puzzle with its current difficulty. Hence, it was a non-deterministic, random process with a constant average block production rate (i.e., a *Poisson* process). The new Ethereum protocol divides time into *epochs* and *slots,* where each epoch contains 32 slots, and every slot lasts 12 seconds. All validators have a role in a slot of an epoch, determined two epochs beforehand.[14] For each slot, a validator is selected to propose a block, and a set of validators, called a committee, is assigned to attest to the proposed block. Since there is a predetermined block proposer for each slot in an epoch, there is no need for a random process like solving a mining puzzle as in PoW-Ethereum. Therefore, PoS-Ethereum achieves a deterministic block time of 12 seconds.

In the new consensus mechanism, if an epoch can gather a total number of votes (attestations) cast by validators holding more than ⅔ of the available stake on Ethereum, it becomes *justified* with Casper FFG. If the attestations given during the next epoch also address the justified epoch as the parent epoch, and the epoch itself also collects more than ⅔ of the total stake's votes, then the justified epoch becomes *finalized* with every block in it. Hence, it takes two epochs (12.8 minutes) after the merge for a block to become final in Ethereum.[15]

Besides the changes in block production and finality, the transition to PoS also changed the security model of Ethereum. More specifically, it allowed the introduction of in-protocol penalties and slashing in case of misbehavior. While penalties only reduce the staked amount of a validator (e.g., for non-timely attestations), slashing leads to loss of the stake and ejection from the protocol. Slashable offenses include proposing multiple blocks in the same slot or submitting contradicting attestations. This mechanism also makes 51 percent attacks more expensive for the attacker, as his stake, which he used to conduct the attack, will be slashed and ejected. In PoW-Ethereum, there was no way to remove the attacker's hardware from the system without harming others.

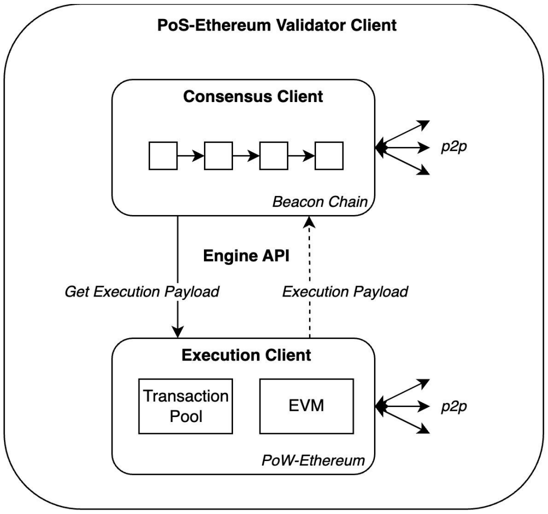
[14] Validators are allocated to a slot using a random number-generating mechanism called RANDAO.
[15] The genesis block is justified and finalized by default.

<c> Block Building and Proposer-Builder Separation

The separation of execution and consensus layers imposes several changes to the node implementations in Ethereum. Previously, miners were running a single node client to build a block and reach a consensus with the rest of the network. Although protocols like the Flashbots Auction took over some of the overhead of block building, miner clients were still responsible for proposing the prepared block and achieving an agreement over it with the rest of the network.

After the merge on Ethereum, validators run two separate nodes for consensus[16] and execution.[17] While the consensus node is in charge of proposing blocks and reaching a shared world state on the Beacon Chain, the execution node is responsible for running EVM and delivering an execution payload, including transaction content, to the consensus node. The communication between these two nodes, or more generally between execution and consensus layers, is realized through a new protocol developed for PoS-Ethereum called the *Engine API*.

When it is the turn for a validator to propose a block, his consensus node contacts his execution node via the Engine API and requests an execution payload. Upon receiving the request, the execution node prepares the most profitable block and delivers it back to the consensus node. The consensus node places the execution payload into the body of the block it is constructing alongside other mandatory fields, such as the proposer index, hash of the parent block, and hash of the current state, and proposes it to the rest of the network. The following Figure XX displays an overview of this structure.



An overview of a validator client on the PoS-Ethereum network.

When MEV extraction was not as prominent of an incentive as today, block building was a more straightforward task. It did not require block proposers to be exceptionally skilled at extracting all the value available in the mempool, as the block reward was already a

[16] Consensus Node Clients: Lighthouse, Lodestar, Nimbus, Prysm, Teku
[17] Execution Node Clients: Geth, Nethermind, Besu, Erigon, Akula

significant subsidy. In PoS-Ethereum, any validator not extracting MEV will be left behind in the game as the in-protocol, fixed block reward is remarkably reduced with the transition from PoW to PoS.[18] The PoS-Ethereum validators who can realize MEV will become centralized forces, as MEV promotes economies of scale.

To avoid incentivizing validators to become sophisticated in block building, *Proposer-Builder Separation* (PBS) is proposed on Ethereum. PBS divides the tasks of a block producer into two separate roles: *proposing* and *building*. In this model, a block proposer is only concerned about reaching a consensus, as he can outsource the block building to a specialized group of builders. When PBS is enshrined into the core protocol of PoS-Ethereum, a validator can just run a consensus node and get the execution payload from the builders market on the execution layer. This way, he can still profit from MEV available in the mempool while not directly being concerned about extracting it.

In PBS, block builders compete with each other to offer the most value to the validators, and their bidding power depends on the amount of MEV that they can extract from their order flow. Finding the most optimal set of transactions/bundles that provide the most value requires running complex algorithms, which can consume significant energy. The only incentive for the builders to bear this cost is to earn the right to be a monopoly for a single slot. In other words, through this competition, a builder buys the block space of an Ethereum block without doing any staking or mining in PoW-Ethereum.

To avoid validators stealing the MEV that the builder aims to capture for himself, PBS protocol makes an execution payload available only after the proposer accepts the bid and signs the payload header. PBS can also make validators stateless as it delegates the block-building task to builders. Thus validators would not have to store the entire Ethereum state anymore.

<c> Flashbots' PBS Implementation: MEV-Boost

There remains some time before PBS gets enshrined into the core Ethereum protocol due to complexities that need to be resolved, such as trust assumptions between proposers and builders. The Flashbots Auction was the first stage towards realizing PBS. It enabled more democratic access to MEV in PoW-Ethereum while helping miners outsource some block-building. However, it was not a silver bullet. mev-geth still required the MEV searchers to trust in the Flashbots relay and miners, as they can see the transaction content and potentially frontrun or censor it to keep the MEV for themselves.

Flashbots developed a new MEV market design to run on PoS-Ethereum, called “MEV-Boost,”[19] to mitigate some of the trust assumptions required by mev-geth.

[18] Ethereum has already started the process of becoming deflationary with EIP-1559 transactions, where the BASEFEE part of a transaction fee is burnt instead of given to the block producer. Visit ultrasound.money for statistics regarding the burnt amount and the supply change.

[19] https://boost.flashbots.net/

MEV-Boost can be described as a PBS stopgap implementation where validators can sell their block space to specialized block builders under the surveillance of relays. In MEV-Boost, the complete execution payload of a builder becomes available to the validator of the allocated slot only after he accepts the builder's bid and signs the payload header. Thus, a validator cannot attempt to steal the MEV before fully committing to the builder's payload. At that point, he can try to build a new block, but there is no guarantee that the consensus participants will accept this block as the canonical head over the initially signed block. Moreover, his stake will be slashed for proposing multiple blocks in a single slot.

Although MEV-Boost improves the trust model of mev-geth, it is not a complete solution for the trust assumptions between the validators and builders. The MEV-Boost architecture still requires the existence of mutually trusted relays sitting between the validators and builders. These relays are responsible for collecting execution payloads from the builders, simulating them, and finally submitting them to a relay-aggregator, called "mev-boost," which works as a middleware between the consensus and the execution clients. Moreover, relays are also needed to ensure data availability and DoS protection for the validators.

MEV-Boost aims to push the centralization effects of MEV to the builder level instead of the validator level to mitigate the risk of MEV becoming a threat to the consensus stability of Ethereum. However, this comes with the threat of censorship posed by the relays. A relay can:

- see the content of an execution payload submitted by a builder and attempt to steal the MEV in it,
- censor blocks of certain builders, and
- censor specific validators' access to MEV by not forwarding the builder blocks it received.

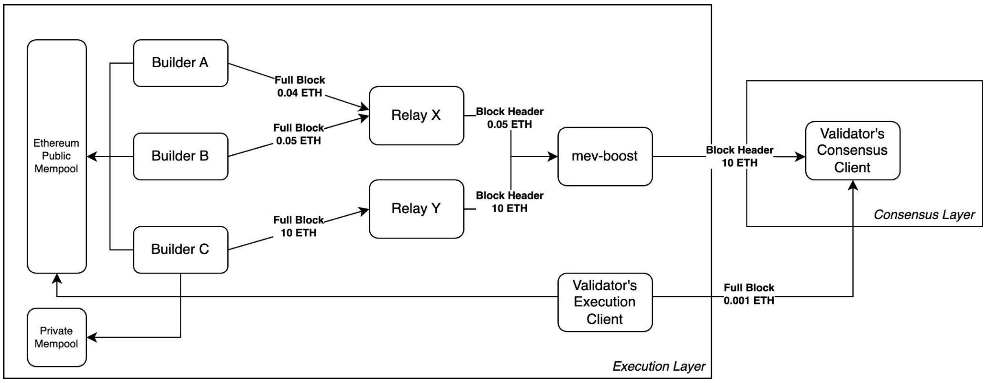
The mev-boost middleware allows validators to connect to as many relays as possible to mitigate the risk of censorship. Besides, a validator can always use his execution client for local block construction if there is no trusted relay. Moreover, builders are free to submit their execution payload to multiple relays. By creating a competition on the relay level, MEV-Boost aims to force relays to behave honestly, as relay activity is publicly auditable. As long as there are other options, malicious behavior will cause relays to lose their reputation alongside validators and builders.

<c> MEV-Boost Architecture

In the MEV-Boost architecture, a validator has to register with the relays he wants to work with. This process is required for validators to inform the relays about the address where they want to get paid, the gas limit of their blocks, and their public key. When the allocated slot of a validator arrives, he contacts his mev-boost middleware to get a payload header. The middleware collects the headers with associated bids from the relays that the validator has registered with and responds to the validator with the header that offers the most value. In a sense, the mev-boost middleware can be thought of as a multiplexer that gets block headers as inputs from various relays and outputs the most valuable one. The validator *blindly* signs (i.e., without seeing the content) this header and submits it to mev-boost, forwarding it to the respective relay. The relay finally verifies the signature of the validator on the signed header and responds back to the validator with the full execution payload.

On the block-building side of MEV-Boost, builders first check if the validator for the current slot is registered with a relay that they work with. In case it is, they start preparing their execution payload by collecting transactions from the public mempool and, if it exists, from their private order flow, such as a private mempool. After constructing the most valuable block possible, builders submit their full block, alongside a bid that they offer to the validator, to the relay(s) they work with. Relays check the validity of the submitted payloads, ensuring the prepared block pays the validator the promised amount and does not consume more gas than the limit set by the validator during registration. Then, they pick the most valuable, valid payload submitted by their connected builders and offer its header to the mev-boost middleware upon request. Finally, the middleware proposes the header which offers the most value to the validator.

Figure XX shows an example block production scenario for a validator that works with MEV-Boost. In this scenario, there are two relays, *Relay X* and *Relay Y*, which the validator works with. While *Builder A* and *Builder B* submit their blocks to Relay X, *Builder C* submits to Relay Y. Across all builders, only Builder C has access to an exclusive order flow through his private mempool. Hence, while the other two builders submit blocks with a value of less than 1 ETH each, Builder C can build a block that offers 10 ETH to the validator. As mev-boost middleware does profit switching, it forwards the block header with the most value to the validator, which is, in this case, from Relay Y. The validator's execution client also prepares an execution payload, but it can only accumulate 0.001 ETH value, even though it builds on the same order flow as Builders A and B.[20] Assuming the validator is *rational*, he accepts the payload by mev-boost.



An example block production scenario with MEV-Boost and a local execution client.

[20] This could be due to the fact that builders are more advanced in block building than a validator's local execution client.

<c> The Impact of MEV-Boost

The numbers shared in this section are based on the network activity between September 15, 2022 (Merge Day) and October 31, 2022 on Ethereum. Resources used: ChainsightAnalytics MEV-Boost dashboard,[21] mevboost.org,[22] and boost-relay.flashbots.net.[23]

MEV-Boost has been actively used since Ethereum transitioned to Proof-of-Stake. According to the numbers by ChainsightAnalytics, more than 160,000 blocks (45 percent of all PoS-Ethereum blocks) have been proposed by validators using MEV-Boost. These validators have received close to 34M USD payment from the block builders. The builder group that delivered the highest number of blocks (76,463) and made the most validator payment (17M USD) is Flashbots' builder group. After that, an anonymous builder known as "builder 0x69" has been the most successful one. So far, this builder has delivered 25,426 blocks and paid around 6M USD to the validators.

Currently, seven active relays are operating on MEV-Boost. Based on the data gathered from the "Data API" of each relay, an open API that provides data about delivered payloads and received builder blocks,[24] 136,200 blocks have been proposed by the Flashbots relay. Flashbots is followed by BloXroute Max Profit (11,396 blocks) and Eden (5,142). The least number of blocks have been relayed by Manifold (2,629).

The current status in the MEV-Boost supply chain points to a centralization around Flashbots, both on the builder and the relay side. On the builder side, this is not a surprise as the Flashbots builders have access to an exclusive order flow from MEV searchers via the Flashbots Auction, which the other builders do not have. Thus, they can usually accumulate more value from the transactions than other builders and, as a result, offer the highest payment to the validators.

On the relay side, the stats show that the Flashbots relay is not only used by the Flashbots' builders but also by other builders. We argue that this is due to both quantitative and qualitative reasons. The quantitative reason is the adoption of the Flashbots relay by the validators. Out of 457,727 active validators on Ethereum, 320,695 are registered with the Flashbots relay. Thus, there is a 70 percent probability that the block proposer of the next slot will have the Flashbots relay configured in his mev-boost middleware. Since block builders depend on the relays for reaching out to the validators, a relay included in the mev-boost configuration of most validators is highly preferable. Hence, the use of the Flashbots relay becomes justified.

As for the qualitative reason, builders prefer the Flashbots relay due to the reputation of Flashbots. The main risks relays pose to builders are censorship and MEV manipulation. Since Flashbots has been the leading research group working on solutions to democratize the MEV scene on Ethereum, any malicious behavior could

[21] https://dune.com/ChainsightAnalytics/mev-after-ethereum-merge

[22] https://www.mevboost.org/

[23] https://boost-relay.flashbots.net/

[24] https://flashbots.notion.site/Relay-API-Documentation-5fb0819366954962bc02e81cb33840f5

negatively impact their reputation and potentially the credibility of their products (Flashbots Auction, Flashbots Protect, MEV-Boost). Block builders are also aware of this and know that it is not an affordable risk for Flashbots to censor blocks or steal the MEV in them.[25] Therefore, they do not see a risk in submitting their blocks to the Flashbots relay.

The decentralization of MEV-Boost is only possible with more trustworthy relays and fairer access to order flow on the builder side. As the current numbers show, builders who do not have access to exclusive order flow can not compete with the ones who do. Although Flashbots currently dominates the MEV-Boost supply chain, they are not aiming to become a monopoly. On the contrary, Flashbots is working towards a more decentralized MEV scene by maximizing the competition. They are encouraging more relays to join MEV-Boost by open-sourcing their relay.[26] Moreover, they are currently working on a mechanism called “SUAVE” (Single Unifying Auction for Value Expression), which is a decentralized block builder that preserves the privacy of users and wallets and provides optimal, MEV-aware execution for them. SUAVE will be an open-source product that the competitors of Flashbots can also utilize.

[25] During this writing, Flashbots is censoring any transactions interacting with the smart contracts of the transaction mixer application *Tornado Cash*, to comply with the sanctions of the U.S. Department of the Treasury's Office of Foreign Assets Control (OFAC).

[26] https://github.com/flashbots/mev-boost-relay

<a> MEV in Bitcoin

Although most of the MEV discussion revolves around Ethereum, as it is the most popular and highly-used smart-contract blockchain, MEV also exists in other protocols. Theoretically, Bitcoin also includes MEV, which can be extracted, for example, by ordering transactions from the Lightning protocol or fee sniping by re-organizing the chain history. However, compared to Ethereum, the amount of extractable value exposed by Bitcoin transactions is significantly less due to the difference in the complexity of the application layer and the programmability of the two protocols. In Bitcoin, most of the transactions are simple money transfers, and there is close to no value exposed by state transitions, whereas, in Ethereum, there exist various ways for users to interact with the protocol via smart contracts that can generate extractable value.

On Ethereum, the MEV surface is immense due to the existence of smart contracts that enable countless ways for interaction. Since MEV occurs as a result of state transition, the protocol with more ways for users to interact naturally has more MEV in it. On Bitcoin, if the Lightning protocol starts to create harmful MEV opportunities, its capability can be limited by removing some operations from the Bitcoin Script. However, on Ethereum, it would only be possible to restrict all possible implementations of specific applications, like DeFi protocols, by constraining the general behavior of permissionless smart contracts. Thus, Ethereum's programmability, which enables an extensive application ecosystem, also acts as its "curse" against MEV. On the other hand, Bitcoin's more-restrictive use case space revolving around money transfers is its "remedy" for minimizing the exposed MEV.

<a> Conclusion

In this chapter, we first introduced how MEV emerged as an incentive and then we presented different forms of it. We made an extensive analysis of the evolution of MEV in Ethereum, presented in three subsections: *The Emergence of MEV in Ethereum (2019 - 2020)*, *Enter Flashbots (2020 - September 2022)*, and *MEV Post Ethereum Merge (September 2022 - Today).* In the first subsection, we looked at the initial MEV extracting activities on Ethereum, such as bot activities leading to Priority-Gas Auctions. We also presented the consensus-level threats MEV poses. In the following subsection, we introduced the MEV market design of Flashbots, The Flashbots Auction, and shared statistics regarding the impact that MEV had on the Ethereum protocol between January 2020 and September 2022. We also discussed the MEV-influenced supply chain framework and why it is critical to achieving modularity. In the final subsection, we explained the new architecture of the Ethereum protocol after the transition to Proof-of-Stake (PoS) and discussed why a solution like Proposer-Builder Separation (PBS) is needed. Here, we introduced Flashbots' solution for achieving partial PBS, MEV-Boost, until full PBS is enshrined into the Ethereum protocol. We also analyzed the use of MEV-Boost in PoS-Ethereum and pointed out the centralization around Flashbots' solutions. In the final section of this chapter, we discussed the existence of MEV in Bitcoin and argued why it is not as prominent a threat as it is for Ethereum.

The phenomena of MEV has gradually increased its impact surface since its formal recognition by Daian et al. in "Flashboys 2.0" (2019). In the early days, it only posed a theoretical level threat to the consensus stability and decentralization of Ethereum, as users were not heavily active in the DeFi space. In the last three years, MEV has shaped the incentive mechanism design of Ethereum and the decentralized applications built on it. Currently, MEV incentivizes private order flows and vertical integration with other actors of the value supply chain, like trading firms or wallets, to extract the most value out of the protocol and, potentially, become a monopoly.

Research groups like Flashbots are continuously working on new products to illuminate and democratize the MEV scene on Ethereum, as uncontrolled MEV can lead to centralization. To avoid Ethereum becoming a censored, opaque, permissioned network, it is critical to ensure that MEV not only accumulates on certain players but flows back to the protocol and the end-users.

References

Daian, P., Goldfeder, S., Kell, T., Li, Y., Zhao, X., Bentov, I., Breidenbach, L., & Juels, A. (2019). Flash Boys 2.0: Frontrunning, Transaction Reordering, and Consensus Instability in Decentralized Exchanges. ArXiv, abs/1904.05234.

Eskandari, S., Moosavi, S., & Clark, J. (2020). SoK: Transparent Dishonesty: Front-Running Attacks on Blockchain. Financial Cryptography and Data Security, 170–189. https://doi.org/10.1007/978-3-030-43725-1_13

Qin, K., Zhou, L., & Gervais, A. (2022). Quantifying Blockchain Extractable Value: How dark is the forest? *2022 IEEE Symposium on Security and Privacy (SP)*. https://doi.org/10.1109/sp46214.2022.9833734

Robinson, D., & Konstantopoulos, G. (2020, August 28). Ethereum is a Dark Forest. Paradigm. Retrieved October 16, 2022, from https://www.paradigm.xyz/2020/08/ethereum-is-a-dark-forest

Escaping the Dark Forest. (2020, October 7). Samczsun. Retrieved October 16, 2022, from https://samczsun.com/escaping-the-dark-forest/