

Beginner's Guide to mev-boost

Understanding mev-boost as an iteration on the original Flashbots auction, and how it paves the way for full in-protocol PBS

What is mev-boost?

mev-boost is the version of Flashbots that is adapted for and compatible with Proof of Stake Ethereum. It is related to the design for in-protocol proposer/builder separation, and can be conceptualised as an intermediate step to full in-protocol PBS. By understanding the current Flashbots auction, and by comparing this current state to the plans for PBS, we see how mev-boost provides block-building functionality that would otherwise be unavailable until PBS is completed.

Understanding the Flashbots Auction

Currently, Flashbots provides a private transaction pool (mev-relay) and a sealed bid blockspace auction mechanism (mev-geth). This allows miners to outsource the work of finding the optimal block construction.

![[current-fb-auction.png]](/img/current-fb-auction.png)

mev-geth introduced a new RPC endpoint, `eth_sendBundle` - the message sent to this endpoint is called a bundle. **Bundles** consist of one or many transactions to be executed in an atomic batch. They can be sent to the relay by a searcher, and to the miners by the relay.

Searchers are Ethereum users who prefer to use the FB private transaction pool over the regular p2p pool. They monitor the state of the chain and send bundles to the relayers. There is an option for searchers to express their bids for inclusion via the ether gas price, or by direct eth transfer to the miner's coinbase address.

The **relay** is a bundle propagation service that receives bundles from searchers and forwards them to miners. The relay is in charge of validating and routing FB bundles. Since searchers do not have to pay for failed bids (they can pay directly to the coinbase address, and make the payment conditional on their bundle succeeding), there is the potential that they could spam the network with invalid bundles. The relay therefore serves as a mitigation to this DOS threat. It will simulate each transaction and only forward valid bundles to the miner.

The **miner** (or more generally, block producer) is the party who ultimately collects all the bundles and produces a block. Miners traditionally run go-ethereum, and greedily order transactions by gas price. However, miners participating in Flashbots run a version of mev-geth.

These miners that are running mev-geth evaluate incoming bundles using the first-price sealed-bid auction and pick the most profitable bundles to place at the top of the block. The node then compares the Flashbots block to a normal block and begins mining on the most profitable.

The Flashbots bundle information allows the searcher to express blockspace preferences relative to the state of the chain as well as the state of the transaction pool. Before Flashbots, the only way for searchers to express their preference for transaction placement was via the gas price. For transactions that needed to be first in the block, this naturally led to competition in the gas price. However, for transactions that could be anywhere in the block, bidding on gas price might be ineffective. Flashbots Auction decoupled placement from gas price, thus enabling price discovery for discrete MEV opportunities instead of competition for priority. The miner can evaluate all bundles received and combine those which do not conflict in order to produce the most profitable block.

Miners have full access to bundle content and could arbitrarily reorder/steal/censor bundles sent to them by searchers and relayers. However, they are incentivised to comply, as Flashbots monitors for misbehaviour and removes miners who steal.

Proposer/Builder Separation

Proposer/block-builder separation (**PBS**) was proposed by Ethereum researchers as a response to the risk that MEV poses to decentralisation of consensus networks. They have suggested that uncontrolled MEV extraction promotes economies of scale that have a centralising effect, as well as causing complications to decentralised pooling.

PBS is a change to the core protocol that aims to mitigate this. Instead of the block proposer (currently the miner, after PoS the validator) also trying to produce a maximally profitable block by itself, it can outsource this to a block building *market*. Here, block builders would produce bundles consisting of a complete block and a fee for the proposer. The proposer just has to pick the block with the highest fee.

There are number of properties that are desired in the PBS design: - untrusted proposers and builders should be able to

participate successfully (there is little to no risk that a proposer would steal blocks from a builder, and vice versa) - proposers are not advantaged by having high resources or technical competence - proposers cannot extract transactions from bundles without paying the fee - that the new design works with the existing consensus layer.

PBS is still an open and active research area that is developing rapidly. The most recent design/specification for the in-protocol version is [two-slot PBS](#)

mev-boost

Mev-boost is an updated Flashbots architecture that is compatible with PoS Ethereum. It also implements the design goals behind PBS, although in a semi-trusted way. The design is as follows:

 (/img/mevboost-proposed-architecture.png)

We start with the existing Ethereum network and mempool in the above diagram. As discussed previously, in the current Flashbots model searchers take transactions from the public mempool, potentially adding their own, and arrange them into bundles.

The possibility of **private orderflow** is also anticipated at this point. Private orderflow (also known as exclusive orderflow) refers to transactions that can be included in blocks but are not visible in the public mempool. This could be because the transactions are sent to a certain entity and they may route these transactions to their own nodes, or keep them secret in order to build more profitable blocks for themselves.

Block builders are also introduced - these are services/providers that will aggregate various bundles and transactions into block templates. The builder orders the transactions in a block according to what is most profitable for them. The block templates are then forwarded to a relay.

The relay receives these block templates (also referred to as execution payloads), and will verify their validity. The **MEV-boost** component is a middleware which handles communication with the relays, the profit-switching logic, and a fallback mechanism in the case of some system failure.

What is the link between PBS and MEV-boost?

If Flashbots Auction was the first step of "block building as a service", then PBS could be thought of as a response to the success and mainstream adoption of Flashbots. Instead of the entire network utilising one block builder (and thereby creating a centralising effect at the block-building level), the protocol would split block proposing and block building into two separate components. Block proposing would be handled in-protocol by the validators, and it is then much easier for block building to be delegated to a market of competing parties.

However, the current designs of PBS require a change to beacon chain consensus (specifically the fork choice rule). For this reason (as well as the designs not being completely finalised yet), it will need to wait until after the merge is successfully completed.

Flashbots MEV-boost acts as a prototype for what a block-building market would look like, and provides the necessary proposer logic and middleware that is missing from PoS Ethereum.

Longer-term, PBS is important to the Ethereum protocol for a number of reasons. Splitting block proposal and block building into separate components actually has desired effects that align with the protocol's goals, such as removing the requirement for validators to hold the full state (stateless Ethereum initiative). PBS is also a necessary step for the current favoured sharding proposal (Danksharding - https://notes.ethereum.org/@vbuterin/proto_danksharding_faq).

Resources and Further Reading

These resources were directly referenced when creating this post - <https://docs.flashbots.net/flashbots-auction/overview> - <https://ethresear.ch/t/proposer-block-builder-separation-friendly-fee-market-designs/9725> - <https://ethresear.ch/t/mev-boost-merge-ready-flashbots-architecture/11177> - https://notes.ethereum.org/@vbuterin/pbs_censorship_resistance - <https://ethresear.ch/t/two-slot-proposer-builder-separation/10980>

Diagrams from <https://github.com/flashbots/mev-boost>

Further mev-boost and PBS resources and relevant talks have been curated by Flashbots here - <https://github.com/flashbots/mev-boost/wiki/Resources>