

# Searching Post-Merge

If you're a searcher on Proof of Work (PoW) Ethereum and you're wondering how your experience will change as Ethereum switches to Proof of Stake (PoS), then this post is for you.

In this document I'll explain new concepts like what proposer/builder separation (PBS), block building, and PoS validators are in simple terms. Moreover, I'll outline how these terms relate to searching and PoW Ethereum. Finally, I'll also address common questions related to these topics.

This post was written as a labor of love for the searcher community. We at Flashbots are working hard so that you can be merge-ready and start searching PoS Ethereum on day one!

---

- [Key Takeaways](#)
  - [Proposer/Builder Separation \(PBS\)](#)
  - [Bidding for Inclusion](#)
  - [How to Choose a Good Builder](#)
  - [What Stays the Same](#)
  - [What Changes for Searchers](#)
  - [Blockbuilding Algorithm](#)
  - [MEV-Blocks API](#)
  - [Better Support](#)
  - [Conclusion](#)
- 

## Key Takeaways

After the merge,

1. **mev-boost** separates what PoS validators (PoW miners) do into two roles: block building and block proposal. Searchers will send bundles to block builders, or they may be block builders themselves.
2. With mev-boost, searchers will need to reason about two auctions. First, searchers compete for bundle inclusion within blocks that builders are proposing. Second, builders compete for block inclusion. To be included on-chain a searcher's bundle must win the blockspace auction for the builder they submit to, *and* the builder they submit to must win the auction amongst different builders.
3. Choosing a builder that is performant and can be trusted with your transactions is critical.
4. Existing client libraries for Flashbots are already compatible with post-merge architecture —**no changes are needed from searchers to send bundles to Flashbots**.
5. The Flashbots builder will use a new algorithm to produce the most profitable block. We no longer guarantee top-of-block execution and ordering with respect to effective gas prices for bundles. Rather, the algorithm optimizes for overall profit. Bundles are still included in the block atomically, but non-bundle transactions may now be included between bundles.
6. We rolled out [searchersupport@flashbots.net](mailto:searchersupport@flashbots.net) and the [Searcher Self-Support Group](#) forum to improve searcher support.

## Proposer/Builder Separation (PBS)

PBS splits the responsibilities of PoW miners (PoS validators) into two roles: block building and block proposal. Block builders are specialists who accept transactions from users and searchers, and try to build the most profitable block possible from those transactions. mev-boost is a trusted form of PBS implemented post-merge where builders submit blocks to a trusted entity (or entities) called relays. Relays are responsible for aggregating blocks from builders and figuring out which block is the most profitable, which is ultimately what validators sign and what is included on-chain. You can read more about relays in [this forum post](#).

As a searcher on PoS Ethereum, your primary concern regarding PBS is sending your transactions/bundles to the right builder(s). Builders, and the relays they connect to, will be able to see your transactions before they're included (which means they can front-run, censor, etc), so it's important to choose builders that you trust.

While it doesn't directly impact searchers, it should also be noted that **mev-boost** is the component that enables validators to plug into the PBS builder network. In short, it chooses the most profitable block from multiple relays. If you want to learn more about this, check out the [official mev-boost website](#). It's also worth noting that this implementation of PBS is only a stopgap for in-protocol PBS. Eventually, PBS will be built into Ethereum natively, which will eliminate the need for trusted

relays.



## Bidding for Inclusion

In PoW Ethereum there is only one auction for inclusion: the blockspace auction. Searchers simply have to bid more than the other searchers vying for the same MEV opportunity, and as long as their transactions don't get censored by the miner, and are profitable enough to the miner, they will be included.

mev-boost adds a new auction to the mix: the **slot auction**. Validators using mev-boost fetch the most profitable block from the relays they connect to, which means that block builders effectively bid for inclusion with the value of the block they produce. When PBS is implemented in-protocol, there will no longer be relays, so validators will pick the most profitable block directly from builders. Check out [this forum post](#) for more information about relays.

In essence, the [slot](#) auction is a game of which builder can create the most profitable block for the validator. Builders may choose to rely only on transaction fees and MEV to constitute the value of their block, but they also have the option to send additional funds to the validator directly for a better chance of inclusion.

You may consider bidding for the slot auction *in addition* to the blockspace auction to increase the chance that your builder's block gets included on-chain. This can be done by sending ETH directly to the address of the validator responsible for filling the target slot.

Although sending bundles to a builder with less searcher competition will reduce your chance of being outbid in the blockspace auction, choosing a good builder that accepts transactions from other searchers will generally improve your chances of inclusion. The more a block is worth (transaction fees + MEV), the more likely it is to win the slot auction and be included on-chain.

You may also consider sending your bundles to multiple builders as part of your strategy. Even if you have the most profitable *bundle* in a block, that *block* could still lose the slot auction. If you have the most profitable bundle in several builders' blocks, your odds improve. That being said, different builders will have different sources of incoming transactions and bundles, and thus different potential conflicts with your bundle. Bid accordingly.

## How to Choose a Good Builder

There are a few criteria to look for in a block builder:

- Are they committed to fair and unbiased execution?
  - A good builder will not front-run, sandwich or censor bundles, or otherwise engage in activities that abuse privileged data access.
- Do they connect to a trusted relay (or relays)?
  - Keep in mind that the relay can also see raw transactions, which gives them the ability to front-run, censor, etc.
- Do their relays connect to enough validators?
  - The more validators a relay connects to, the more slots will generally be available for builders connected to that relay. When you're targeting a specific block/slot, it's imperative that you send your transactions to a builder which is connected to the validator responsible for proposing a block in that slot. More validators  $\Rightarrow$  better inclusion rates.
  - Note: Any validator can [use mev-boost to connect to the Flashbots relay and other relays](#)
  - It's also worth considering how much collective stake the validators connected to a relay have. Generally speaking, if more than one block is proposed to the network (unusual but possible), the block with the most collective stake attesting to it will be included. This scenario is explained in greater detail in the [Ethereum docs](#).

Also note that block builders have the freedom to specialize. You may find that one builder is more or less friendly to your strategy than others. Builders are competing with each other, so they are all incentivized to include your bundles in their blocks, but you may find that some builders will prioritize certain strategies over others regardless of potential profits. Builders might also censor certain bundles due to local regulations or corporate strategies and policies. There are a lot of variables in play here, so I recommend trying a few trustworthy builders and seeing how your mileage varies first-hand.

## What Stays the Same

Flashbots remains steadfast in our commitments to illuminate the dark forest, democratize MEV extraction, and distribute the benefits of MEV for social good, all while working in the open.

Flashbots will run a builder following the principles we've adhered to in PoW Ethereum: democratized access to MEV, fair & reliable execution, and privacy.

The Flashbots builder hosts all the same endpoints as the PoW mev-relay API. After the merge, the original URL for mev-relay ([relay.flashbots.net](https://relay.flashbots.net)) will point to the Flashbots builder, making the transition seamless — you won't have to change

anything. Clients used to send, simulate, and debug bundles are already compatible with our post-merge architecture.

## What Changes for Searchers

### Blockbuilding Algorithm

The Flashbots builder will use a new algorithm designed to produce the most profitable block possible. This design introduces some important changes for searchers to be aware of:

- Instead of ranking and including bundles based off of effective gas price the algorithm now optimizes for overall block profit.
- Top-of-block execution is no longer a guarantee.
- Bundle ordering by effective gas price is no longer a guarantee.
- Other transactions (e.g. from the mempool) may land between bundles (not between transactions in bundles, but between two different bundles).

*For example:*

If you have a bundle comprised of transactions [B1, B2]  
and someone else has a bundle comprised of transactions [C1, C2]  
and there are transactions in the mempool [t1, t2, ...],  
then the block may be built such that:  
BLOCK\_TXS = [..., B1, B2, t1, t2, C1, C2, ...].

The bundles still remain atomic, but the difference from the PoW architecture is that not all bundles will be ordered next to each other, and bundles are not necessarily placed at the top of the block.

### MEV-Blocks API

The data processed by the [Blocks API](#) will see some changes on the backend, but we will keep the existing API online through the merge by porting data from the Flashbots builder to the original API. This is done to maintain backwards compatibility with client libraries. We plan to expose a new /v2 API alongside the original, which will include information specific to PoS Ethereum — we'll release more details on this soon via [Twitter](#) and our [Discord server](#).

### Better Support

First, we've created a new forum for searchers: The [Searchers Self-Support Forum](#). This was created to facilitate long-form discussions about searching, and to help searchers find answers to common issues more quickly. We've also added new content in our [FAQ](#) related to searching post-merge. Lastly, we've created a new email account for searchers to report problems, ask questions, and suggest new ideas: [searchersupport@flashbots.net](mailto:searchersupport@flashbots.net)

## Conclusion

In this post, we covered the basics of proposer/builder separation in PoS Ethereum, and how that affects searchers. We discussed the blockspace auction, which PoW searchers are already familiar with, and the new slot auction in which builders compete. We also explain some considerations for choosing builders, and how Flashbots will make the transition to searching on PoS Ethereum effortless.

Answers to common questions related to searching after the merge can be found in our [FAQ](#) (latest questions at the end). We also invite you to join the discussion and share your questions and ideas in the **new** [Searchers Self-Support Forum](#).