Preface

The goal of this concept is to reduce the amount of MEV that is currently given to validators and give more of it back to users who are responsible for it.

The concept is simple (and has been discussed in many forms before) but this post wants to mention a very simple implementation strategy that can be done right now. The idea is that for each transaction a user sends there is an auction for the right to back-run the transaction. For public mem-pool transactions, those auctions are already happening (not only for back-running, also for sandwiching) but the bids go to validators.

Concept

A simple RPC endpoint for users that want to keep the MEV value of their transaction. For each transaction, the RPC provider offers the transaction (without signature) to all searchers (can be permissionless). Searchers can “bid” to be included first after the transaction. Part of the transaction needs to be a simple ETH transfer to the user (msg.origin) - the amount of that transaction is essentially the bid. The provider takes the highest bid and creates a “bundle” out of the user+backrun/bid transaction and can send it to all trusted builders.

If builds perform as expected this means the user transaction can only land on-chain together with the back-run transaction. Of course, if there are no bids (not all transactions create MEV) the transaction can be sent to a builder without a back-run tx (still protecting the user from being front-run).

Information leakage

While searchers can not just take the transaction and get them into the chain outside of this mechanism they still learn about the transaction. In theory, they could use that information and still try to front-run a user (for example a user wants to buy a huge amount of an exotic token which would move the price significantly). This can be combated by adding fake transactions to the transactions presented to the searcher. If e.g. User A spends 10 ETH to buy token A the RPC provider could simply present 3 more transactions to searchers in which the user would buy token B, C, or D. Essentially the higher the % of fake data the lower the information value that gets leaked. It should thus be possible to find the % that is high enough to prevent searchers from attempting to front-run a user as the searcher bears the cost of each failed front-run attempt.

Trust assumptions

This approach is a strict improvement over services that already today try to offer frontrunning protection as this approach has the same trust assumptions (trusting the RPC provider, builders and relays) but in addition gives value back to the user.