Hi Flashbots Collective! This is Doğan from Clave Team. We have designed a new AMM mechanism utilizes SUAVE for auction mechanisms. I want to bring the problem that we're solving and our solution to the forum for discussions.

AMMs are dumb

In AMMs, there are two main parties involved. On one hand, we have traders who aim to exchange their assets for another asset, and they pay fees to liquidity providers (LPs). On the other hand, we have LPs who seek to generate profits from their assets.

Let's imagine a scenario where the price suddenly spikes to 8 USDC, while the pool price remains at 2 USDC. This spike creates an opportunity for bots to make money. However, the price quickly drops back down to 2 USDC again. The IL of the LPs is zero, and traders haven't lost anything because volatility is nature of the trading. However, bots have managed to make some money. But where do these bots actually make money from?

Bots take advantage of the opportunity cost that LPs would have had to trade and make money during the spike. So they're stealing the LP's opportunity cost.

[

image

1920×832 125 KB

](https://collective.flashbots.net/uploads/default/original/2X/1/128229105238064ef8b03edd649c0fed4d22b6f6.jpeg)

Today, MEV bots engage in arbitrage by directly paying the builder. The builder receives a percentage fee from the profit, while the bots earn the remaining fees.

However, what would happen if we allowed MEV to be distributed to LPs?

[

image

1920×998 118 KB

](https://collective.flashbots.net/uploads/default/original/2X/f/fb518cc17b0bf80a8f82bab64882f9bb39b87921.jpeg)

SUAVE and UNI V4:

SUAVE is a credible off-chain computation platform designed to democratize block building. Developers can perform complex computations and execute smart contracts on it. Additionally, users have the ability to relay their arbitrary data to other blockchains. UNI V4 is Uniswap's newest upgrade for their protocol. It allows developers to build custom conditions, functions and novel applications on the top of Uniswap's Liquidity pools permissionlessly. UNIswAP v4 allows anyone to deploy new concentrated liquidity pools with custom functionality. For each pool, the creator can define a "hook contract" that implements logic executed at key points in a call's lifecycle. These hooks can also manage the swap fee of the pool, as well as withdrawal fees charged to liquidity providers. SUClave: A New AMM Design Using SUAVE and Uniswap V4

Arbitrage bots are doing MEV attacks through the Tap of Pool, aiming to be the first to access the liquidity pool. Being the first one that touches the pool gives an opportunity for MEV. While arbitrage is a common aspect of finance, the issue here is that Liquidity Providers (LPs) are missing out on potential profits. Essentially, these bots are earning profits using the funds provided by LPs, who are unable to capitalize on these opportunities themselves. For instance, consider a situation where the actual market price of ETH rises to $2000, but within the pool, it's still valued at $1000. In such cases, LPs could have profited by trading their ETH for USDC. However, it's the bots that are seizing these opportunities instead. Therefore, if there were a method to redistribute the profits made from arbitrage back to the LPs, it could effectively address this problem.

In SUAVE, various auction mechanisms allow bids to be placed freely, and developers are provided with an attestation signature by SUAVE. This signature serves as proof that a transaction originated from one of these auctions. Consequently, it becomes straightforward to implement an auction mechanism that can also be authenticated on the Ethereum network. To leverage this capability, we've integrated this auction approach with a custom UNI V4 Hook. This specific hook is designed to only accept transactions that are verified as coming from a SUAVE auction. How is it works?

6/ In our design, we create an auction on SUAVE where the bid directly goes to the LPs, thereby increasing their profitability.

The winner of each auction gains the advantage of being the first to interact with the pool and can profit from arbitrage opportunities.

Furthermore, we have designed a backrun mechanism for normal transactions that do not require being the first toucher to the pool. It is almost same to the MEVShare SUAPP where the profit goes to the reward contract of LP's and it also has an attestation

So, we have two types of attestations, both of which should be able to access and swap through our hook. To ensure their

validity, we have designed a hook that performs the necessary checks.

With this design, we can easily create an AMM that can generate profits with the MEV.

[

image

3200×2243 384 KB

](https://collective.flashbots.net/uploads/default/original/2X/1/14d92f8caffe442632e7252a5894046c6e3f4ee4.png)

In Clave wallet, we are using everyday devices' Trusted Execution Environments for key management to achieve best security that is available on everyday devices. But most of the TEE's (aka Secure Enclaves) are supporting only P256 curve. Thus, we implemented a custom precompile that verifies P256 curve and follows the EIP-7212 specs.

Consequently, we have an (MM that distributes the profits from MEV to the LPs, thereby reducing the negative impact of the LVR problem.

I believe that MEV will continue to exist in blockchains unless we achieve complete privacy. So we should find a way to make it harmless.

Last Thoughts:

We designed this mechanism during a 36-hour hackathon at ETHGlobal Istanbul. There might be some inaccuracies or missing points, so I'm open to feedback and suggestions for further exploration. And we've won Flashbots's LVR Track prize and also Code base is broken right now, I just wanted to share the mechanism design. I'm not sure if we will continue to this project or not, it was only for fun.