MEV

Front Running Attacks on Blockchain (https://arxiv.org/pdf/1902.05164.pdf)

“Front-running is a course of action where someone benefits from early access to market information about upcoming transactions and trades”

From https://hackmd.io/@flashbots/quantifying-REV (https://hackmd.io/@flashbots/quantifying-REV)
Maximal (formerly Miner) Extractable Value is the value that can be extracted from a blockchain by any agent without special permissions. Considering this permissionless nature, any agent with transaction ordering rights will be in a privileged position to perform the extraction.

Introductory Video

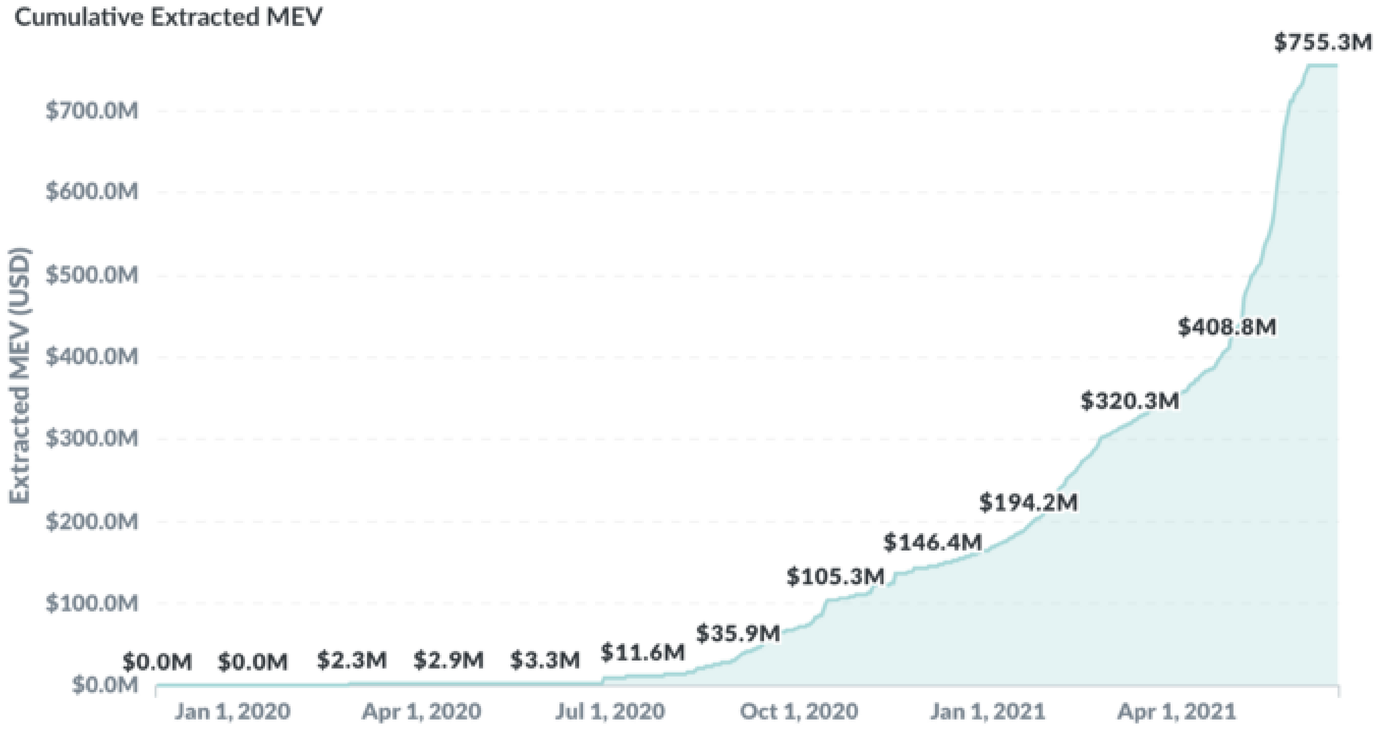
https://www.youtube.com/watch?v=UZ-NNd6yjFM (https://www.youtube.com/watch?v=UZ-NNd6yjFM)

Generic front running bots

The above example shows how bots can extract value to a transaction irrespective of the contract called.

Flashbots

Flashbots research : https://hackmd.io/@flashbots?tags=["research"]
(https://hackmd.io/@flashbots?tags=%5B%22research%22%5D)



A Taxonomy of Front-running Attacks

Attack types

- displacement
- insertion
- suppression

Displacement attack

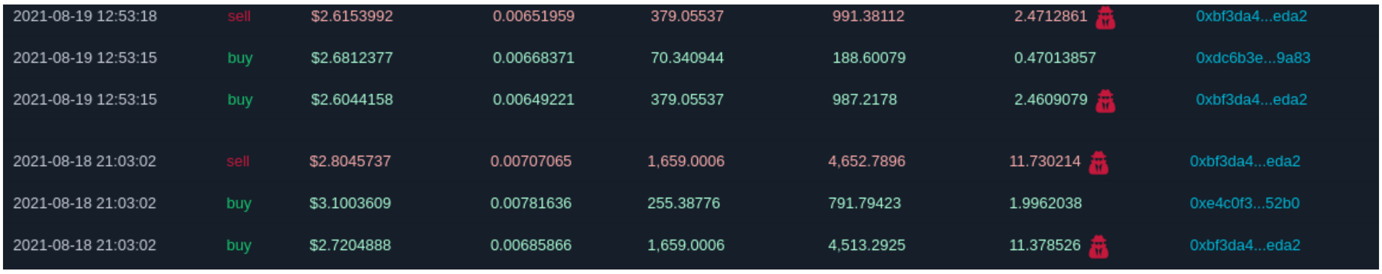
It is not important to the adversary for Alice’s function call to run after Mallory runs her function. Alice’s can be orphaned or run with no meaningful effect.
Examples of displacement include:

- Alice trying to register a domain name and Mallory registering it first
- Alice trying to submit a bug to receive a bounty and Mallory stealing it and submitting it first
- Alice trying to submit a bid in an auction and Mallory copying it.

Insertion attack

In an insertion attack (sandwich), after Mallory runs her function, the state of the contract is changed and she needs Alice’s original function to run on this modified state.
For example, if Alice places a purchase order on a blockchain asset at a higher price than the best offer, Mallory will insert two transactions: she will purchase at the best offer price and then offer the same asset for sale at Alice’s
slightly higher purchase price.
If Alice’s transaction is then run after, Mallory will profit on the price difference without having to hold the asset

Example



2021-08-19 12:53:18	sell	$2.6153992	0.00651959	379.05537	991.38112	2.4712861	0xbf3da4...eda2
2021-08-19 12:53:15	buy	$2.6812377	0.00668371	70.340944	188.60079	0.47013857	0xdc6b3e...9a83
2021-08-19 12:53:15	buy	$2.6044158	0.00649221	379.05537	987.2178	2.4609079	0xbf3da4...eda2
2021-08-18 21:03:02	sell	$2.8045737	0.00707065	1,659.0006	4,652.7896	11.730214	0xbf3da4...eda2
2021-08-18 21:03:02	buy	$3.1003609	0.00781636	255.38776	791.79423	1.9962038	0xe4c0f3...52b0
2021-08-18 21:03:02	buy	$2.7204888	0.00685866	1,659.0006	4,513.2925	11.378526	0xbf3da4...eda2

Suppression attack

The attacker wants to delay Alice running her function , but is then indifferent to what happens.
Used in an attack of a gambling DApp

The process of competing for priority in the block is known as a Priority Gas Auction.

Ethereum is a dark forest

https://www.paradigm.xyz/2020/08/ethereum-is-a-dark-forest/ (https://www.paradigm.xyz/2020/08/ethereum-is-a-dark-forest/)

“On Wednesday afternoon, someone asked whether it was possible to recover Uniswap liquidity tokens that had been accidentally sent to the pair contract itself.”

When anyone calls the burn function on a Uniswap core contract, the contract measures its own liquidity token balance and burns it, giving the withdrawn tokens to the address specified by the caller.
I found the contract. The liquidity tokens were still there—and were worth around $12,000.

“Because I’m a professional DeFi thought leader, I had never actually deployed a contract to Ethereum before.”

Deploy a Getter contract which, when called by its owner, would make the burn call ONLY if activated, and otherwise revert.
Deploy a Setter contract which, when called by its owner, would activate the Getter contract.
Submit the set transaction and the get transaction in the same block.

To our surprise, the get transaction would get rejected by Infura even when we manually overrode the gas estimator. After several failed attempts and resets, the time pressure got to us, and we got sloppy. We let the second transaction slip into a later block.

It was a fatal mistake.

Our get transaction did get included—but with a UniswapV2: INSUFFICIENT_LIQUIDITY_BURNED error, meaning the liquidity was gone. It turned out that, in the seconds after our get transaction entered the mempool, someone had executed the call and swept the funds.

Escaping the dark forest

https://samczsun.com/escaping-the-dark-forest/ (https://samczsun.com/escaping-the-dark-forest/)

On September 15, 2020, a small group of people worked through the night to rescue over 9.6MM USD from a vulnerable smart contract.

Lien Finance held 25K ETH ($9.6M) , but anyone could with a simple transaction mint, then burn tokens which would transfer the 25K ETH to themselves.
The Lien Finance team are anonymous, so how to prevent this / or reach out to the right team without divulging the exploit ?
Sam Sun eventually reached the team via some auditors they had worked with.

What was needed was an infrastructure solution, so they contacted Spark Pool who were developing their Taichi Network, their co founder Shaoping Zhang offered to help.

“I brought the whitehats’ request to our development team, and explained the urgency: our private transaction feature needed to be in production within a few hours. Our devs said they could try their best to finish in time, and we immediately got to work. We finished development of the private transaction feature in 2 hours, and then spent some time fixing bugs.”

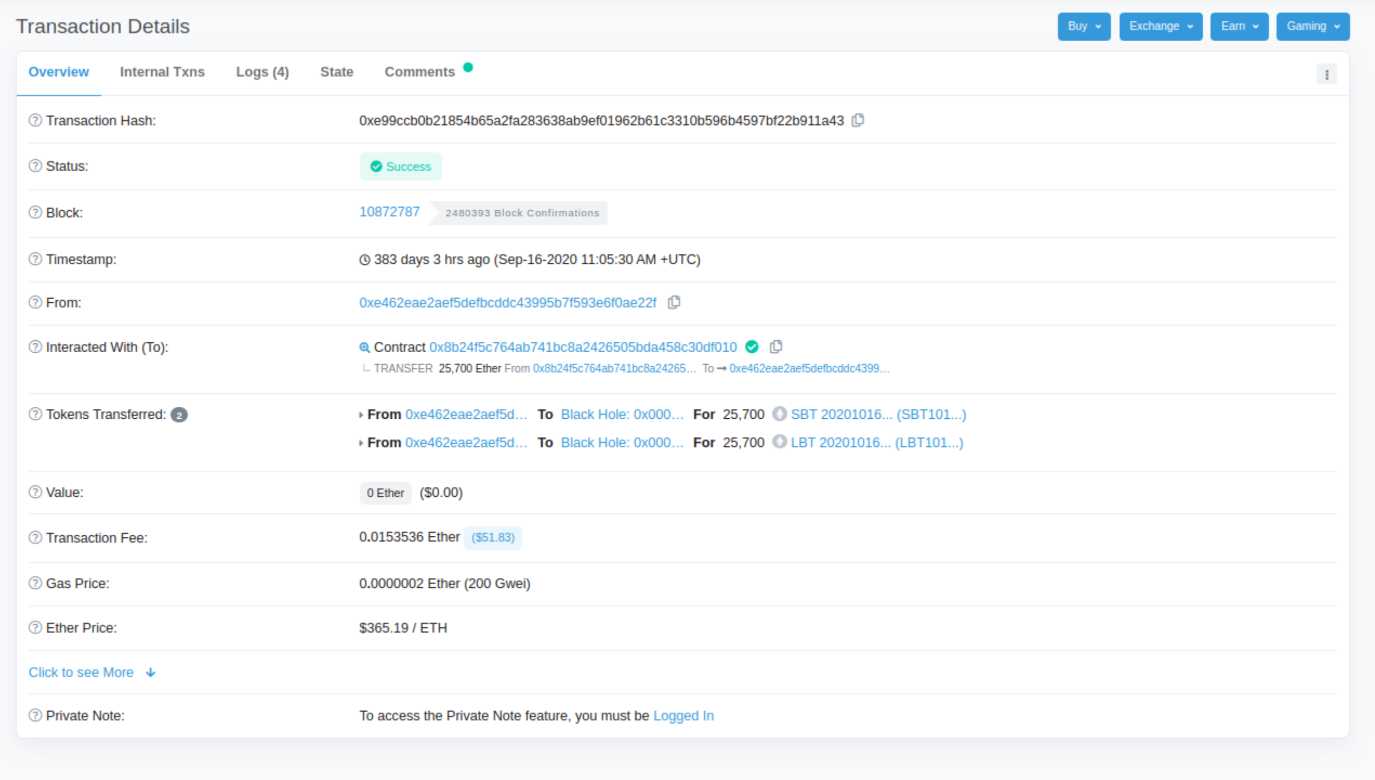
Scott Bigelow and Sam were developing the script to generate 4 sequential signed transactions to save the 25K ETH
This involved setting up some tokens that could eventually be burnt by the Lien team to release the 25K ETH.

They sent a test transaction to the Taichi network and watched the mempool, it didnt appear, then suddenly appeared in a block, the private network was working.

They then ran in the real transactions, and waited… only a portion of Spark Pools hashrate was being used for the private network.
Everyone watched etherscan nervously …

After about 15 blocks the transactions appeared in order, and slowly more blocks were built in top, reducing the possibility of a re org.

The Lien team were then able to send a transaction to withdraw the ETH at risk



Transaction Details

Transaction Hash:	0xe99ccb0b21854b65a2fa283638ab9ef01962b61c3310b596b4597bf22b911a43
Status:	Success
Block:	10872787 2480393 Block Confirmations
Timestamp:	383 days 3 hrs ago (Sep-16-2020 11:05:30 AM +UTC)
From:	0xe462eae2aef5defbcddc43995b7f593e6f0ae22f
Interacted With (To):	Contract 0x8b24f5c764ab741bc8a2426505bda458c30df010 └ TRANSFER 25,700 Ether From 0x8b24f5c764ab741bc8a24265... To → 0xe462eae2aef5defbcddc4399...
Tokens Transferred: 2	From 0xe462eae2aef5d... To Black Hole: 0x000... For 25,700 SBT 20201016... (SBT101...) From 0xe462eae2aef5d... To Black Hole: 0x000... For 25,700 LBT 20201016... (LBT101...)
Value:	0 Ether ($0.00)
Transaction Fee:	0.0153536 Ether ($51.83)
Gas Price:	0.0000002 Ether (200 Gwei)
Ether Price:	$365.19 / ETH
Click to see More ↓	
Private Note:	To access the Private Note feature, you must be Logged In

Illuminating the dark forest

https://extropy-io.medium.com/illuminating-the-dark-forest-748d915eeaa1 (https://extropy-io.medium.com/illuminating-the-dark-forest-748d915eeaa1)

MEV and EIP-1559

https://hackmd.io/@flashbots/MEV-1559 (https://hackmd.io/@flashbots/MEV-1559)

We found no critical way in which EIP-1559 interacts with MEV extraction. However, we identified several areas where new dynamics might take place, in particular around miners' incentives to extract more MEV or defeat the new fee mechanism by passively colluding around a potential nefarious Flashbots' software update.

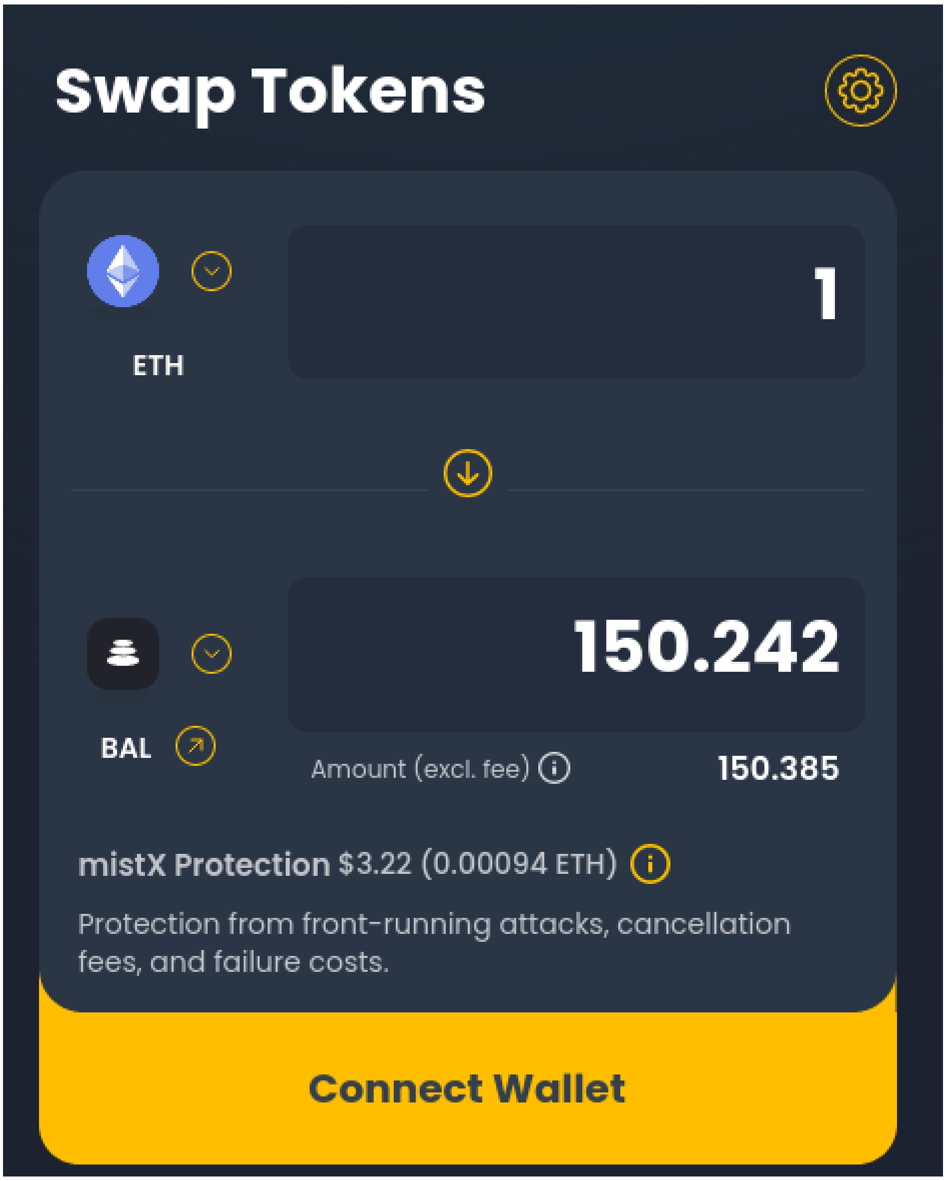
Mitigation

MEV-Geth

https://docs.flashbots.net/flashbots-auction/miners/mev-geth-spec/v03/
(https://docs.flashbots.net/flashbots-auction/miners/mev-geth-spec/v03/)
Part of Flashbots Alpha

A market-based alternative to unilateral trader-miner collusion for the extraction of MEV. Flashbots is a two-sided marketplace composed of miners and MEV searchers. Any miner or mining pool can run MEV-geth and receive additional income from MEV strictly more than what it would earn from running vanilla geth, without the need to enter into any bespoke deal with traders.

Mistx

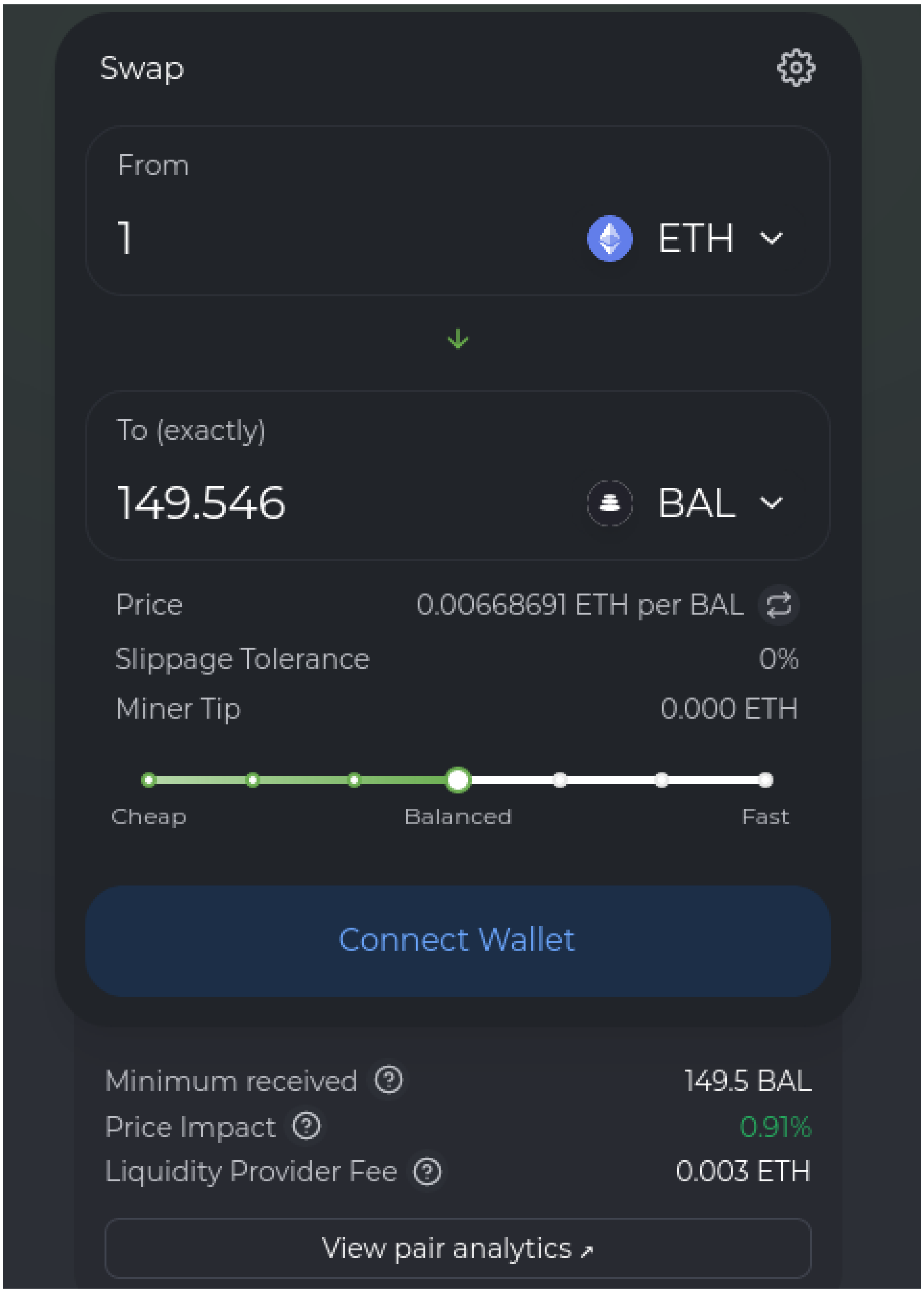


Amount (excl. fee) 150.385

mistX Protection $3.22 (0.00094 ETH)

Protection from front-running attacks, cancellation fees, and failure costs.

Archer Swap



Swap

From

1 ETH

To (exactly)

149.546 BAL

Price 0.00668691 ETH per BAL

Slippage Tolerance 0%

Miner Tip 0.000 ETH

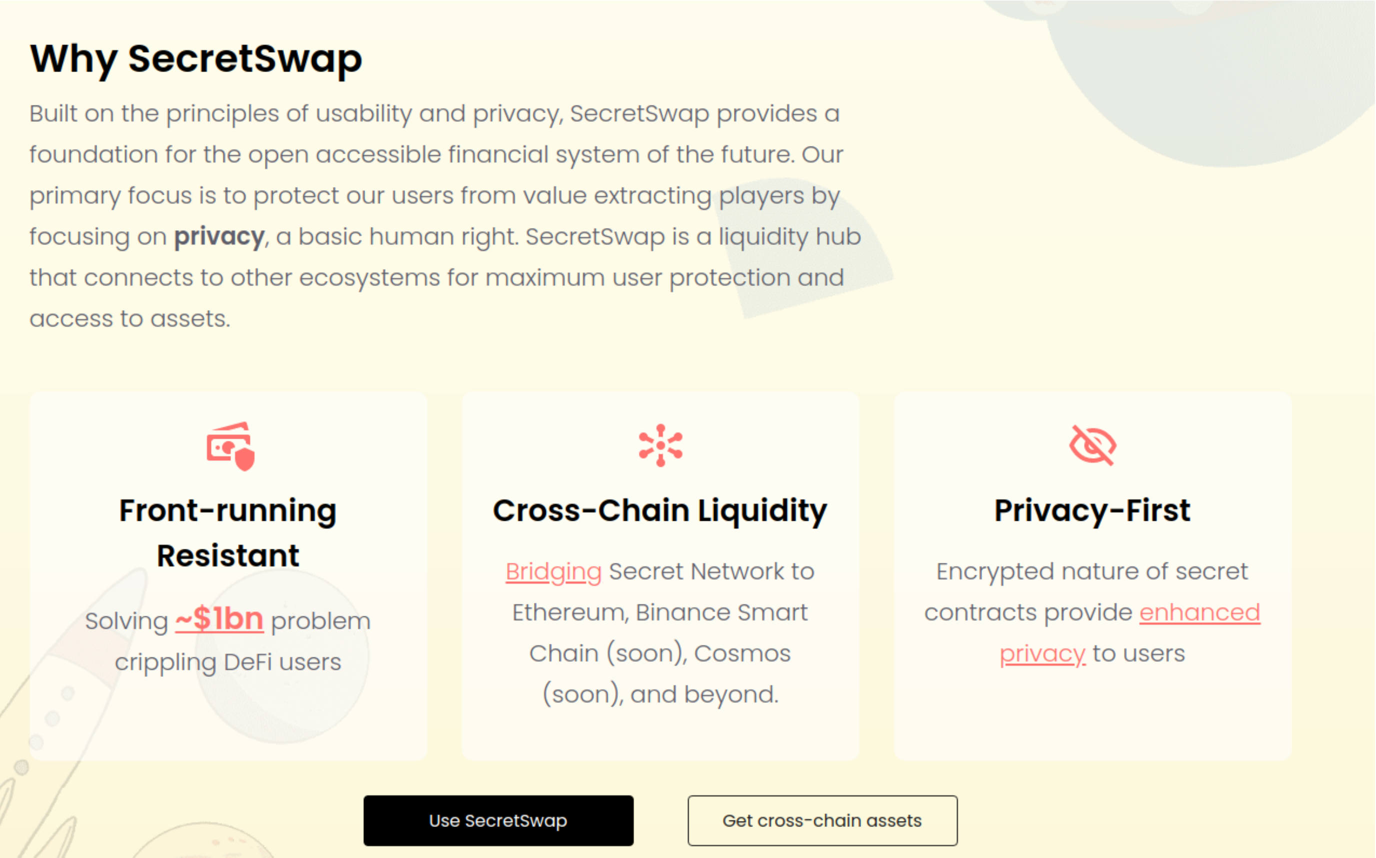
Minimum received 149.5 BAL

Price Impact 0.91%

Liquidity Provider Fee 0.003 ETH

Secret Swap

SecretSwap is a cross chain AMM built on Cosmos and Tendermint



Why SecretSwap

Built on the principles of usability and privacy, SecretSwap provides a foundation for the open accessible financial system of the future. Our primary focus is to protect our users from value extracting players by focusing on **privacy**, a basic human right. SecretSwap is a liquidity hub that connects to other ecosystems for maximum user protection and access to assets.

Front-running Resistant

Solving ~$1bn problem crippling DeFi users

Cross-Chain Liquidity

Bridging Secret Network to Ethereum, Binance Smart Chain (soon), Cosmos (soon), and beyond.

Privacy-First

Encrypted nature of secret contracts provide enhanced privacy to users

Cow Swap

CowSwap is a DEX and DEX aggregator hybrid backed by Gnosis Protocol V2 (GPv2) which is developed by Gnosis team in order to provide MEV protection. GPv2 optimizes for coincidence of wants (CoWs), which can be explained as “an economic phenomenon where two parties each hold an item the other wants, so they exchange these items directly.”, i.e. peer-to-peer transactions can be matched without having to go through a regular AMMs like Uniswap or Sushiswap. One of the benefits of this is that off-chain transactions will cost a lot less and be faster.

CowSwap brings in ‘Solver’ conception to realize this function. Solvers are encouraged to compete against each other to deliver the best order settlement for traders in exchange for the reward of each batch. CowSwap will use a united price to settle all orders in the same batch, which is called batch auction mechanism. This process is very similar to Ethereum meta-transactions proposed in 2018.

Who is doing this ?

From MEV and Me (https://research.paradigm.xyz/MEV) (Feb 2021)
The defining feature of Ethereum’s current era is that most miners are not attempting to exploit MEV themselves (yet). Nearly all of the current activity is driven by non-mining traders. However, some MEV can only be captured by miners, because they have the authority to arbitrarily order (or exclude) transactions. Non-mining traders can access a strictly smaller subset of “simple” MEV; “complex” preferences cannot be efficiently expressed through PGA’s.

According to a Flashbots report (Feb 2021)
At the time of writing, there are 5 mining pools running MEV-geth, collectively accounting for over 12% of total Ethereum hashrate, collecting 0.13ETH per block of additional MEV revenue from Flashbots transaction bundles. On the MEV searcher side, we are seeing a 3x increase in unique searchers who have successfully landed bundles on chain.