MEV is not inevitable

. It is an exploit

caused by a vulnerability

that we can fix

.

It is going to cost Ethereum users around 1.4 billion dollars this year alone. I have shown that this money will be taken by the rich from the poor who will be powerless to protect themselves.

Having been concerned about this issue since I first discovered it pre-genesis in 2014, I am so happy to offer a solution.

All of my work on this is open source. If you use any of my ideas I only ask for acknowledgement. I'm available for discussion, talks/presentations, brainstorming, specifications, modeling, tea drinking, etc to anyone sincerely wanting to fix this vulnerability, whether you are Flashbots, founders, Optimism, core devs, the EF, a private/public company etc or are just fans of Ethereum and concerned citizens like me.

Please pm me on this forum or discord:pmcgoohan#9435 or contribute to the docs on github. With love, pmcgoohan.

UPDATE: in this talk for EthGlobal I discuss more recent ideas for Plain, Dark and Fair variants of the Alex Content Layer protocols (slides) as well as the root causes of MEV with some real world examples given here.

Now, let's decentralize…

Targeting Zero MEV - A Content Layer Solution

Relevant proof for fairness assumptions concerning transaction ordering

Targeting Zero MEV - A Content Layer Solution

Introduction

No Satisfaction Guaranteed

A projected 1.4 billion dollars will be taken from Ethereum users in 2021 as Miner Extractable Value (MEV). For the first time this will surpass the amounts made in High Frequency Trading (HFT) in the traditional financial markets at around 1 billion dollars.

It seems odd that a decentralized blockchain like Ethereum could suffer worse exploits than it's traditional centralized competitors. Wasn't decentralization meant to fix this?

Well our instincts are correct. Decentralization will fix the problem. The reason these problems have not yet been fixed is that Ethereum has not yet fully decentralized.

Hidden Centralization

A network is only as decentralized as its weakest point.

Blockchain structure is fully decentralized. Blocks are proposed and validated by consensus across tens of thousands of nodes. But there is a dirty secret at the heart of each block. While the blockchain structure

is created collaboratively, the content

of each block is not.

This fact is not obvious because it happens in private in the few milliseconds it takes for a miner/validator to create a block and because it is couched in the elegantly distributed data structure that surrounds it.

But the fact is that the content

of each block is created by a centralized authority without recourse, the miner. As long as a proposed block is structurally

sound, the content

of the block is undisputed by the consensus.

This distinction between structure

and content

is profound because nothing about block structure

creates the problem of MEV. Frontrunning, backrunning, sandwiching and other attacks all come from the centralized way in which block content

is produced.

Block content is not trustless.

Content By Consensus

There's nothing wrong with the existing structural consensus layer in Ethereum, it works beautifully. But look at how block content creation sits uncomfortably within it, sneakily centralized in the miner.

Consider the famous double spending problem that blockchain technology was designed to solve: if one computer has complete control of a financial ledger, how can you stop it spending the same money twice? The answer is that you can't. Instead you build a structural consensus where no single computer is in complete control of currency transfers, and the problem is solved.

MEV is the equivalent of the double spending problem for executable blockchains. If one computer has complete control of transaction inclusion and ordering, how do you stop it from frontrunning, backrunning, sandwiching and generally exploiting everybody else? Again, you can't. Instead you build a content consensus layer where no single computer is in complete control of transaction inclusion and ordering, and the problem of MEV is solved.

So let's free content from miner control and give it a dedicated consensus layer. Now we have a content layer within a consensus protocol stack. No-one is in control, and everybody is. We have decentralized. Now that

feels good.

Advantages

We remove control over the content of a block from a single party and distribute it across the network.

Fairness

By stripping any one agent of their ability to manipulate content, applications become fair and equitable to all users by default. Fairness becomes an innate property of the network without the need for difficult and obstructive workarounds at the application level that are rarely implemented.

Our mechanisms for fair inclusion and ordering are provably close to optimal. They are certainly far more equitable than the current worst case of total miner control.

Integrity

MEV is all but eradicated because there is no centralized authority to bribe.

Auditable

As with the structural layer, the consensus layer is publicly auditable. Any observer is able to recreate the content of any given block using publicly available content consensus messages.

Impact

Block content protocols are a layer on top of existing block structure protocols. Tcp/Ip didn't need to be revised when p2p messenger apps came along. We don't need to revise the underlying block structure protocol to add the block content protocol beyond a few integration changes.

Interoperability

The protocol does not change whether we are creating content for an eth2 validator, a rollup sequencer, eth1 miner or any other Ethereum structural layer. A single content consensus implementation may be used across all of these networks and more. Solve it for one and we solve it for all.

Price Discovery

Inter-market mechanisms like simple arbitrage that are important for price discovery are still permitted. MEV as the

exploitation of a helpless victim by a privileged actor due to a network vulnerability is not.

Philosophy

There is currently a centralized aspect to the network and it is causing harm. We need to fix it if we are serious in our ambitions for full decentralization.

Alex - A Block Content Consensus Protocol

What follows is an overview of one possible block content consensus protocol called Alex.

Overview

Here is a simplified view of the protocol. Pickers choose transactions. Shufflers mix them up. The printer manages it all and prints the chunks to the blockchain (or rollup).

In Brief

- A scheduler allocates a set of roles at random from a pool of nodes to work on each chunk of content:
- Pickers each provide their unique view of the mempool by bundling pending transactions.
- These are combined to prevent transaction censorship

.

- Shufflers each provide entropy.
- These are combined to randomize each chunk of transactions and prevent transaction reordering

.

- Shufflers share their entropy with vaults who then reveal it if the shufflers don't to prevent withholding

.

- If the process halts because a participant has gone offline or is being obstructive, skippers act to jump the set and prevent denial of service

.

- eth2

: if a validator proposes a block that diverges from this consensus content, it fails attestation and is not included and the validator may be slashed

- centralized rollup sequencer

: if the sequencer fails to write the consensus content, they are slashed and possibly voted out

- distributed rollup sequencer

: as with eth2, their block is not be validated by the consensus and fails and/or they are slashed

Full text here…

Targeting Zero MEV - A Content Layer Solution

Relevant proof for fairness assumptions concerning transaction ordering