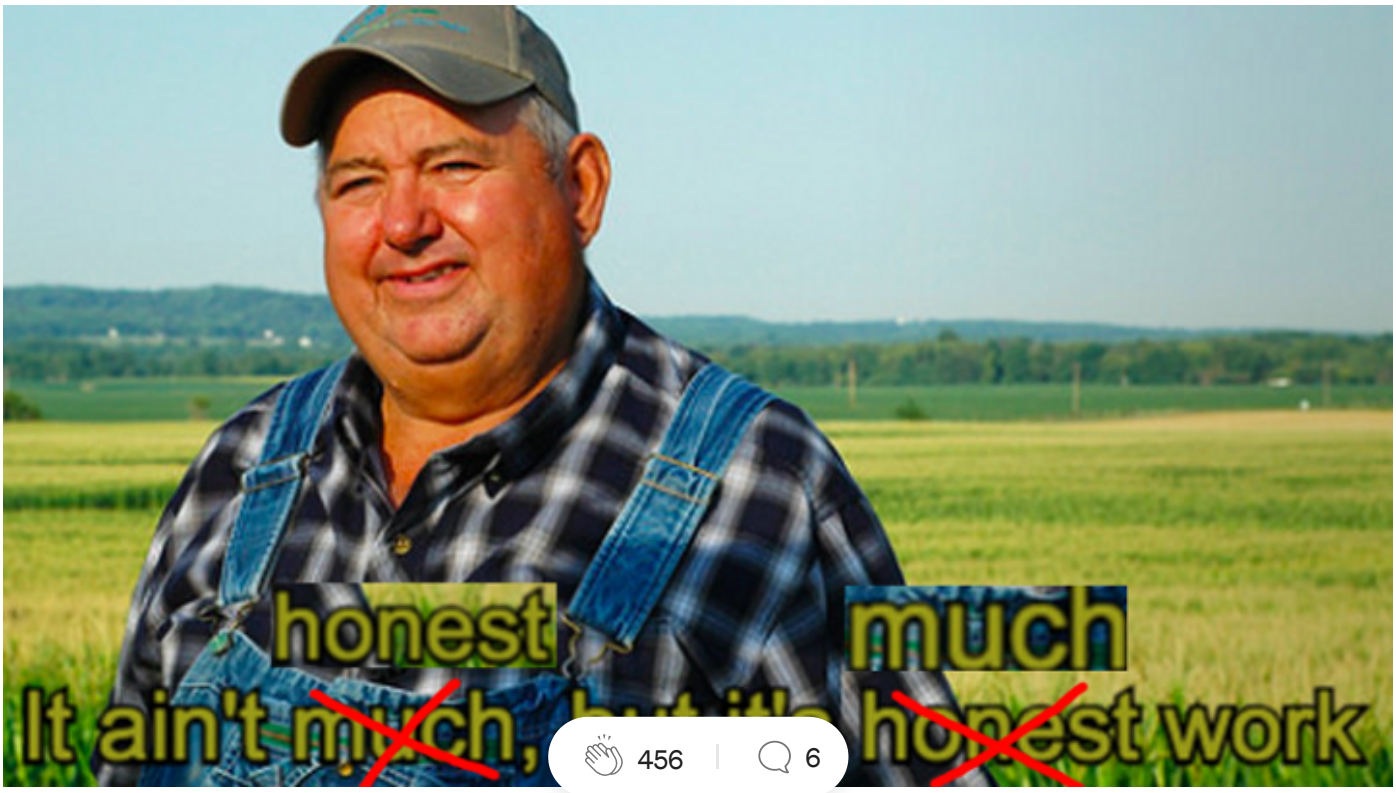
Aviv Yaish Follow
Aug 5 · 4 min read · Listen

Uncle Maker: (Time)Stamping Out The Competition in Ethereum

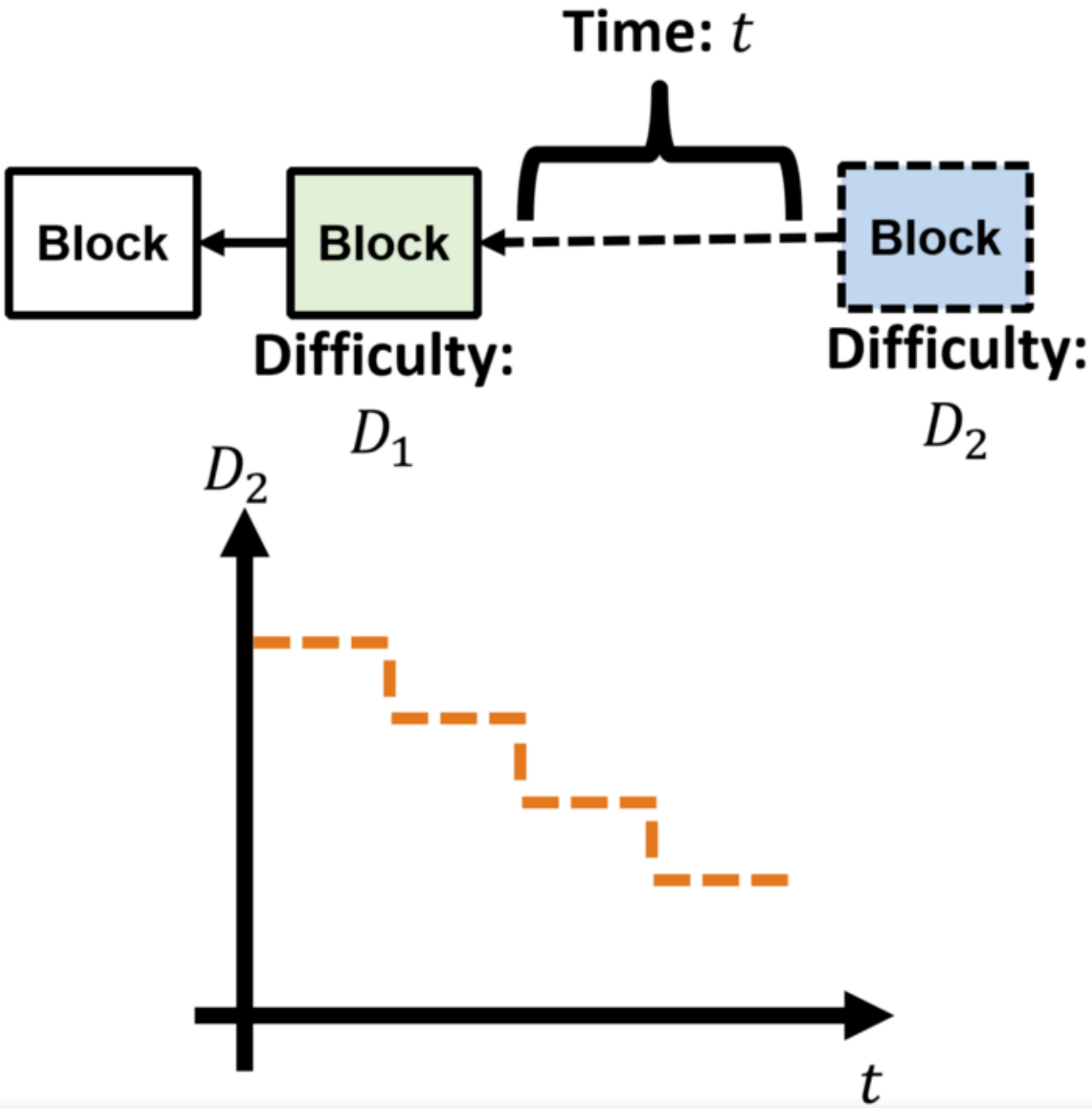
The First Evidence of An Attack on a Major Cryptocurrency

In a new paper, we (Aviv Yaish, Gilad Stern & Aviv Zohar) present and analyze a new attack vector on Proof-of-Work cryptocurrencies, such as Ethereum. We then go over recent Ethereum blocks to give the ***first*** evidence of a consensus-level attack executed in the wild on a major cryptocurrency!



Mining Difficulty in Ethereum

In Ethereum, the difficulty of mining the current block changes on-the-fly and decreases the longer the time that has passed without anyone mining a new valid block; this is done to ensure that inter-block times will not be too high in expectation. **What could go wrong?**

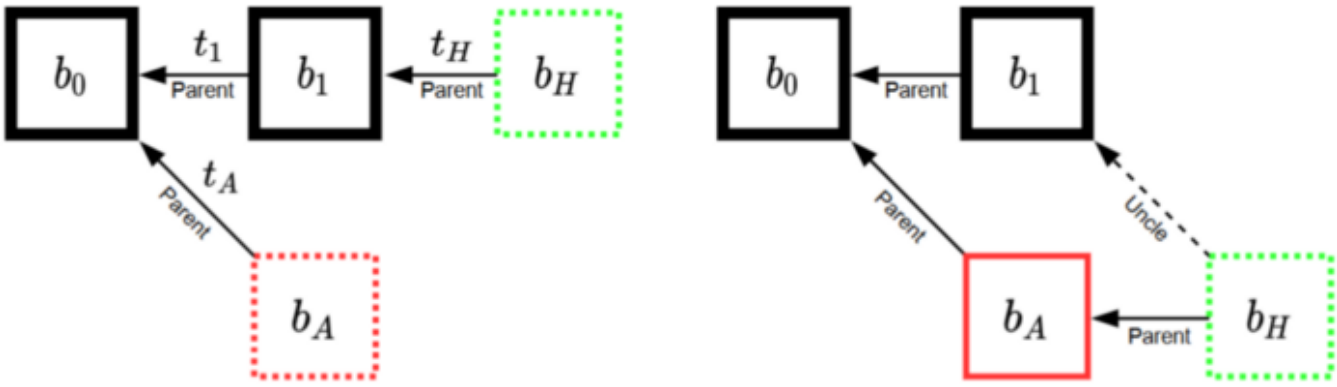


block timestamps. The problem with this is that miners have a certain degree of freedom when setting them, and can even set false timestamps. For example, a miner can start mining a block now, but set the block's timestamp to actually be 5 seconds in the past, or 10 seconds in the future. As long as this timestamp is within a certain reasonable bound, the block will still be considered valid, according to Ethereum's consensus laws.

These same consensus laws say that in case of ties between blocks of the same height, the block with a higher total mining difficulty should be picked to be the parent of the currently mined block, while the other one should be its uncle.

Thus, a miner who wishes to replace the last block on the blockchain, can do so by mining a new block of its own which has a timestamp which is low enough to increase the block's mining difficulty. This can be useful, for example, in cases where this last block has high paying transactions, or in order to double-spend a transaction contained within the block. Another possibility is for an attacker to preemptively mine blocks with such false timestamps, in order to make sure they win in case of ties with other blocks which might be mined concurrently, or which might've been mined in the recent past but haven't reached the attacker yet.

Note that this is in stark contrast to most of the existing literature, which attempts to replace blocks by requiring that miners secretly create a heavier chain, withhold it and then publish it.



No Risk, No Gain? No!

In our paper, we formally describe the aforementioned attack and rigorously analyze it. We show that by executing the attack in a specific manner, the attack does not entail any behavior which has a non-zero probability of earning less than mining honestly, meaning that *our attack dominates the honest mining strategy*.

In Search of Lost Time: Uncle Making in the Wild

By analyzing publicly available on-chain data, we can finally say that the answer to the long-standing question *do miners attack the consensus layer of major cryptocurrencies?* is a resounding **yes!**

Although most mining pools produce relatively inconspicuous-looking blocks, F2Pool blatantly disregards the rules and uses false timestamps for its blocks. Specifically, whenever a block should have a timestamp difference from its parent which is divisible by 9 (precisely the time at which mining difficulty decreases), F2Pool hangs at the preceding second a while longer, thereby increasing mining difficulty and profits. Thus, in the past two years, *F2Pool didn't have even a single block with a timestamp which is divisible by 9*.


F2Pool never mines blocks with timestamps which are divisible by 9, instead falsely setting their timestamps to be one second earlier, thereby increasing mining difficulty and profits!

When looking at blocks mined by other pools, one can observe that practically all others have an over-representation of uncle blocks with timestamps differences which *are* divisible by 9, meaning that F2Pool specifically attempts to replace these blocks.

Ethermine, Hiveon and 0×5a...4c have more uncles than we'd expect at the 9 seconds mark, because F2Pool is an uncle maker!

Take Home Message

We feel that it is very fitting to publish this paper on the cusp of The Merge, e.g. Ethereum's migration to Proof-of-Stake. The current version of Ethereum's consensus, which relies on Proof-of-Work, is known for adopting changes rapidly, without always carefully examining them and the effect they might have on the incentives of miners. Thus, changes which were designed to mitigate one vulnerability, open the door for new ones. Our paper shows that consensus mechanisms and changes to them should be rigorously analyzed, especially with regards to mining incentives.

This was responsibly disclosed to the Ethereum Foundation before the publication of

to via our websites: AvivY, Gilad, AvivZ.