VENOM: a Viable Ethereum Node Operation Mechanism

In this post we will go over the importance of validators on the Ethereum Proof-of-Stake consensus layer, the current state of their privacy and why it is important for these entities.

Additionally we will go deeper into describing the current threats and attack vectors, as well as the compromises made that enable these threats. The compromises are tightly correlated to other parameters, crucial for correct functioning of the network - such as network latency.

We will also describe these parameters and their role of being bottlenecks for improving validator anonymity. To overcome these issues, we analyse a potential solution which involves creating a forward contract market for block inclusion.

Note that we explicitly use the term, inclusion

. This is not per se a price solution, rather more so a censorship

solution. Ofc YMMV.

This is an introductionary post, we shall describe the implementation in detail in a follow up post. The current development details as well as it's overall robustness is ongoing.

Ethereum – Proof of Stak(ing)

Note: It is expected that the reader has a high level understanding of the Ethereum Proof of stake consensus mechanism.

Peering

Ethereum, like most permissionless blockchains, maintains a P2P network among its nodes. Each node is represented by its enode ID, which encodes the node's IP address and TCP and UDP ports [42]. Nodes in the network learn about each other via a node discovery protocol based on the Kademlia distributed hash table (DHT) protocol [30]. To bootstrap after a quiescent period or upon first joining the network, a node either queries its previous peers or hard-coded bootstrap peers about other nodes in the network. Specifically, it sends a FINDNODE

request using its own enode ID as the DHT query seed. The node's peers respond with the enode IDs and IP addresses of those nodes in their own peer tables that have IDs closest in distance to the query ID. Nodes use these responses to populate their local peer tables and identify new potential peers.

Consensus Layer

In Ethereum's Proof-of-Stake consensus layer, validators

are entities that are responsible for bringing the network to a consensus and to a finalized state. To become a validator, one must stake a fixed amount of Ether (32 Ether currently), which is necessary for the security of the network - but also necessary for disincentivising the validator for performing malicious actions, which would contribute for preventing the network to come to a consensus. In such cases, when the validators do not follow the consensus layer rules, they will be penalised appropriately, and in some cases they will be also excluded from the network. The penalties and rewards are described in more details here.

In contrast to that, the validators are rewarded by a small amount of Ether for appropriate behaviour according to the consensus layer rules - which serves as an incentive to continue participating in the network.

The Ethereum consensus layer is designed in a way that validators have different roles, which are assigned for each slot and might be different. The validator roles (or duties) are the following:

- Attestations
- Aggregations
- Sync committee aggregations
- Block proposals.

[At each slot, a single

validator (block proposer) proposes a block, while the other validators are attesting for the beacon block head (LMD GHOST vote) and the epoch checkpoint block (FFG vote).](https://kb.beaconcha.in/attestation)

The validators are rewarded according to the importance of the duty they are performing for the slot, as well as the current state of the network. The importance of each duty is different.

Block Proposals

Block proposals are the most important messages. Attestations are also very important, because they essentially vote what is valid and what is not. However because of the proportions of the obligation delegation (one block proposer vs many attesters), the emphasis on valid block proposals is set much higher. Additionally there is a buffer for the amount of the attestations necessary for justification and finalisation of a block, which means the network would continue forward even if not all validators have attested properly.

In order for blocks to be proposed, the validators with a block proposer duty should be active/online in the first place and then respond to their role appropriately.

There is an inherent risk of suboptimal functioning of the consensus layer, as well as network stall, from validator inactivity.

Decomposing Proposal from Inclusion

Network latencies are very important for the Proof-of-Stake consensus layer that Ethereum utilises, and it plays an important role in the slashing and reward distribution in the network. Validators are quite sensitive to the network latencies, and solutions which increase latencies by a large margin would not be feasible[1]

.

[

Validators rewards are dependent on the inclusion delay of their messages.

1600×679 6.64 KB

](https://global.discourse-cdn.com/standard10/uploads/manifold/original/1X/343df9219366d07c7f55449e0e63f6ad3c40276b.png)

Validators rewards are dependent on the inclusion delay of their messages.

Next

In the next article we will detail the unique solution set that addresses these issues outlined above. We will discuss how his new market works and how Manifold underwrites the risk to enable participation.

1. We are constrained by the block periodicity less potential time drift (as a bounded estimation value) when est. confirmation time ↩