

This is a first draft writeup of an idea I've had for a while.

The problem

Currently ethereum is highly centralized for anyone wishing to use dapps but not running their own node. While I do not have the numbers to prove this I'm sure 99% or more of users are simply using things like Metamask to "talk" to the ethereum network. While Metamask and others are fine solutions on the UX side of things, they all depend on centralized pre-determined lists of nodes that can send the required transactions for their user as well as get blockchain state information. If Ethereum is to stay decentralized it needs to solve this problem and provide "open api" nodes en masse.

The solution proposal

I think it should be possible to incentivize "Relay nodes" to allow arbitrary users to send transactions to the ethereum network. There are a few changes needed on the protocol level:

1. A new transaction type "relayed_tx" needs to be implemented supporting an additional "relayed_by" address field.
2. A new "build-in (ala casper)" smart contract to handle giving our relay rewards on demand (for a std. call price). Let's call this the "relay_collector" smart contract.
3. An additional reward per block for each "relayed_tx" summed and paid to the "relay_collector" contract.

The way it'd work is this:

1. User signs their own transaction as usual and sends it to a "relay_node" (a node with required APIs open)
2. "relay_node" adds it's own reward address to the signed TX and "Wraps" it as a relayed_tx and sends it out to the network as usual.
3. Miner calculates proper total relayed_tx sum of rewards pays them to the "relay_collector" contract (this is verifiable by other miners too, by checking the tx lists)
4. At some point "relay_node" owners can request to be paid out by sending a "collect(relay_node's address)" call to the "relay_collector" contract. This costs the usual fees to avoid being spammed.

This solve both the problem of incentivizing running of open "relay_nodes" as well as limiting the added work and data storage needed by the network.

By utilizing a single "relay_collector" smart contract with a hardcoded address the blocksize increase should be limited to "relayed_tx_count * address_size + payment_operation_size".

If we paid the amounts directly to each relayer address each time the blocksize would grow linear with each relayed_tx since a reward operation would be required for each sender as well.