Ethereum gives us several tools to make a great social media platform: identity/login via public key cryptography, name system via ENS, tipping, token based rewards, token based voting/ranking/curating, etc.

The thing that’s missing is the ability to publish content for free, scale to millions of users, and view this content without spam (spam/sybil resistance).

I think I might have a solution for this that could be used to recreate 99% of the functionalities of a website like reddit.

It would use a pubsub gossip protocol (IPFS already has an experimental implementation of this). A gossip protocol allows peers to send messages to all other peers. Peers filter messages they wan’t/don’t want using “topics”. “Topics” are arbitrary strings. In this case they would be an ENS or public key that represent the name of the subreddit.

A user would sign and publish a subreddit thread/comment/vote through an HTTP provider in his browser, the HTTP provider would broadcast the message to the gossip peers using the correct “topic” (subreddit).

The admin/moderators of the subreddit would run a gossip peer that listens to this “topic” (his subreddit name or public key). They can filter spam automatically by requesting the user includes a captcha challenge answer with his thread/comment/vote. The captcha challenge URL would be up to each subreddit admin, most would use Google or Cloudflare captchas, or whatever effective captcha of the time. They can also filter spam manually, like regular subreddit mods.

The admin/moderators would then publish approved posts to a “name system” gossip protocol (IPFS has this, called IPNS). The “name system” is different from pubsub, only the owner of the ENS or public key can publish to the “topic”, instead of everyone. The “topic” is the ENS name or public key instead of an arbitrary string.

A user would view the content of the subreddit in his browser through an HTTP provider, the HTTP provider would query the name system gossip protocol for the name of the subreddit, and send the content to the user. If the user is running a native app, he doesn’t need an HTTP provider, he can be a direct gossip peer. If not, he can choose any HTTP provider, like Ethereum with Infura.

Anyone sees any problem with this design? I am interested in writing a whitepaper for it, if anyone wants to help, let me know (I can fund the initial development). My telegram is estebanabaroa and discord estebanabaroa#2853