p2p network meeting note (Berlin Interop)

Intros

- Data-driven decision. We are gonna make decisions based on data.
- QUIC
- We will talk about ideal network stack

Some data

Talking about block propagation CDF from ethpandaops (from 9-12 sentry nodes)

- Blocks propagate quickly (~500ms). Maybe it's because they are mostly from mev-boost builders.
- But attestations propagate more slowly. Some clients attest as soon as they receive the blocks but some attest only at 4s. The CDF presented may not include the validation time.

We observed only 0.2Mbit/s per att subnet. So we probably have a problem that we give 4s for this job.

We observed that the min block size is 46KB, and the max block size is 168KB.

Ethpandaops: We did run a big block experiment to send >1MB blocks. In fact, we also have organic large blocks (>1MB).

Blob propagation

Blob sidecars can each take 1s to propagate. Bringing IDONTWANT is highly beneficial to help reduce the duplicates.

BW utilization throughout the slot

We have pulsatile/bursty bw utilization. We should restructure the slot. The last 4s probably use least bw. We should consider a way to send the atts directly to the aggregators because only the aggregators need them.

Design targets

- slot restructuring
- slot time compresion
- 100M gas, towards 300M gas
- towards 1000x blobs
- etc ...

The presenter reviewed through which libp2p implementation supports which feature.

We should make Gossipsub metadata context-aware, like having bitmaps in IHAVE rather than having the hashes.

Propagation layer planning

Gossipsub is designed for small messages (MTU sized or less), but we are sending big messages.

Solutions:

- Send fewer duplicates (today we focus here)
- Code messages differently
  - RLNC
  - Cell-layer messaging with encoding

Marco proposes Choke extensions

Receiver decides whether to receive eagerly or lazily. CHOKE means you should send me lazily. UNCHOKE means you should send me eagerly.

Mikel: there are differences between implementations. We probably should sync between teams to make sure they are consistent. From my experience, I connect to everyone and then suddenly I was disconnected from Teku.

Csaba: IWANT scheduling logic. This is a difference between implementations as well. Some clients limit the number of concurrent IWANTs.

Marco: We provide recommendations on Gossipsub for clients and each client can have its own strategy.

Only go-libp2p-pubsub doesn't have parameters per topic while others have.

s/o: We should change how we negotiate features. We had a problem on how to know

whether the node supports IDONTWANT or not.

Raul: we probably should have a dashboard of all features on gossipsub

Where do we go next?

Directions:

- Network coding (RLNC, Fountain codes)
- Streaming
- Structured topologies (top-down/deterministic (grid, tree), Emergent)
- Transport possibilities (unreliable datagrams, 0-RTT, large fanouts)

RLNC

- Gossipsub is too complex. RLNC makes the gossip layer super dump, by spraying chunks to the network.
- Next steps
  - Land QUIC unreliable datagrams on libp2p
  - Rebase Potuz prototype on top of libp2p

Transport layer

Deprecate mplex

We had a PR to deprecate it for a long time, but we haven't successfully enforce it yet.

QUIC intro

- 1-rtt vs 3-rtt in tcp+{tls,noise}

QUIC Datagram

We would like to go to QUIC Datagram. Think of UDP but better.

Benchmark & Simulation

We use Shadow as a simulator.

Benefits:

1. Implemenation-agnostic.
2. Discrete event simulator

Gossipsub Interop Tester

https://github.com/libp2p/test-plans/pull/648 (https://github.com/libp2p/test-plans/pull/648)

We can use it to benchmark libp2p implementations. We can see improvement when clients interact with each other.

We want to have feedback from telemetry/metrics to get more accurate simulations.

Telemetry and Analysis

We can get the metrics from telemetry and feed it to the simulations.

Ethpandaops has the infrastructure (xatu) to help do this.

Gossipsub traces exploration

Wonbin found that xatu nodes are connecting to one another. We need to break these links from the infrastructures.

Paranoid mode on

Validator anonymity

s/o: I think we shouldn't spend time providing validator anonymity. It has been broken and we had a study many years ago.

Others

s/o: We should kill EL's p2p network. (or combine both layers)

s/o: Currently we don't fully utilize the bw. We had an idea called "graph healing" where you keep soft connections to peers which they don't send anything but when they want to communicate they can use those soft connections.

First principles

We should rethink the entire network stack.

- QUIC is the first choice.
- Should we have other transports to support browser citizens?: WebTransport, WebRTC, WSS.
- Legacy stack: TCP + Yamux + Noise.
    - s/o: We don’t want to use Yamux.
- Drop modularity where none is needed
- MTU optimization
- EL/CL networking planes unifications
    - Francesco: I think blob txs are definitely worth unifications.