Reducing Beacon Chain Bandwidth
for Institutional and Home Stakers

Adrian Manning
Diva Martinez
Beacon Node Bandwidth
Beacon Nodes

There are a number of production beacon nodes:

- Lighthouse
- Teku
- Prysm
- Nimbus
- Lodestar

Although we’ll be looking at data from Lighthouse, the issues addressed are generic and apply to all clients.
The bandwidth associated with a beacon node is proportional to the number of validators attached to it.

The bandwidth plateaus once 64 validators are associated with a beacon node.
Three main groups of nodes:

- Nodes without validators
- Low validator counts (<10) - Indicative of home stakers
- High Validator Counts (64+) - Institutional stakers
Three main groups of nodes:

- Nodes without validators
- Low validator counts (<10) - Indicative of home stakers
- High Validator Counts (64+) - Institutional stakers
Hitting bandwidth limits leads to reduced validator performance

- Increased cost
- Lowers environment diversity -> decentralization
- Spikes in home internet activity -> degrade validator performance (missed/late attestations and blocks).
- Future Ethereum upgrades will require more bandwidth
Where is it coming from?
Beacon Node Network Internals

**Discovery**

- **Discv5**
  - **Transport:** UDP
  - **Encryption:** AES-GCM
  - Handles the discovery of new peers and their capabilities

**Not-Discovery**

- **Gossipsub**
- **Eth2 Req/Resp**
  - **Transport:** TCP
  - **Encryption:** Noise XX (Libp2p negotiated)
  - Rapid object dissemination through the network Direct p2p communication, including initial syncing.
Beacon Node Internals: Total Bandwidth
Discovery vs Libp2p Bandwidth
Discovery vs Libp2p Bandwidth

Discovery ~ 40kB/s
Libp2p ~ 200kB/s

Total ~ 250kB/s
Gossipsub is a publish-subscribe system:

- Subscribe to "topics" to receive messages
- Rapid dissemination of messages
- Large amounts of traffic is split into subnets
  - There are 64 attestation subnets
- A node must subscribe to 1 subnet for every attached validator
- Subscribing to a subnet means receiving and propagating traffic on that subnet
- Bandwidth use is proportional to subscribed subnets

High bandwidth is due to Gossipsub traffic.
Example: mesh degree: 3
Gossipsub Internals

Example: mesh degree: 3
Gossipsub Internals

Example: mesh degree: 3
Gossipsub Internals

Example: mesh degree: 3
Gossipsub Internals

Example: mesh degree: 3
Gossipsub Internals

Example: mesh degree: 3
Example: mesh degree: 3
Example: mesh degree: 3
Consensus Layer - Gossipsub

The current network specification dictates a mesh degree of 8 with a high value of 12.

Results of the specification:

- Incredibly robust network
- Low message latency (lower hops)
- Large number of duplicate messages
- Redundancy resulting in high bandwidth

Beacon nodes have a mesh degree of 8.
Gossipsub duplicates
Summary of current state

- Bandwidth is proportional to the number of validators a node has
- High amplification in gossipsub (duplicate messages)
Reducing Bandwidth

Minimising Topic Subscription
We need a solid peer set subscribed to each subnet in order to have stable subnets.

- Validators require peers on specific subnets at specific times in order to publish/aggregate attestations
- Finding peers is a slow process
- We need a stable set of peers on the network subscribed to subnets so validators can connect to them and publish/receive messages
- Current solution is: One “long-lived” subnet per attached validator to a beacon node:
  - Not enforceable - Beacon nodes could simply not do this (incentivised not to also)
  - Subnets are potentially over-subscribed
  - Excess/Redundant Bandwidth
- Why not every beacon node to subscribe to just one “long-lived” subnet?

https://github.com/ethereum/consensus-specs/issues/2749
We gain a bandwidth reduction for all beacon nodes with >1 validator attached.

- Single validator beacon nodes increase bandwidth by about 50-100kb/s
- Beacon nodes without validators now contribute to subnets
57% of scanned Beacon Nodes have no validators and contribute nothing to subnet stability

Current subnet distribution provides each subnet with ~9% of all nodes

The shift will force all beacon nodes to participate

Subnet distribution should become ~1.5% of all nodes (configurable)

Represents an 80% decrease from current subnet density

~10% of nodes (institutional) will see >90% reduction in bandwidth

Enforceable - Subscription tagged to node-id
Reducing Bandwidth

Episub
High mesh degrees cause high bandwidth.

Ways to reduce bandwidth:

- Lower our mesh degree
- Dynamically adjust mesh degree
- Lower clients peer count

Episub:

- Prevent peers from sending us duplicates on the mesh dynamically
- Dynamically make our mesh more efficient - reduce duplicates and latency

High mesh degree provides faster messages and greater resilience at the cost of high bandwidth.
Episub in a Nutshell

Message Stats
Episub records metrics of past messages and the peers that sent them.
In particular who has been sending the most duplicates and who sends us late messages.

Choking
Given the message statistics, the router “CHOKE”s peers which prevent them from sending messages on the mesh, only gossip.
This minimizes duplicate messages received.

Unchoking
If a “CHOKE”d peer gossips messages faster than we see them on the mesh, we “UNCHOKE” them, to improve mesh latency.
Episub - Preliminary Results

Vzyo (libp2p, protocol-labs) has completed some preliminary simulations

https://github.com/vyzo

Simulation (300 messages, 20 sources):
- 250 Nodes
- 500 Nodes
- 1000 Nodes

https://github.com/vyzo/gerbil-simsub
Vzyo (libp2p protocol-labs) has completed some preliminary simulations

https://github.com/vyzo

Simulation (300 messages, 20 sources):
- 250 Nodes
- 500 Nodes
- 1000 Nodes

https://github.com/vyzo/gerbil-simsub
What do I need to do?

Nothing! .... Just wait
Thank you!

Diva & Adrian