**** rust-**lib**p2p Open Maintainers Call

## Table of Contents

[Oct 8, 2024](#_orbwjecaoqda)

[Sep 24, 2024](#_v24e9mbmes25)

[Sep 10, 2024](#_3luq4f4fg1vv)

[Aug 27, 2024](#_ln73c2x255z)

[Aug 13, 2024](#_bkeu7m2i44t0)

[Jul 30, 2024](#_o0lj5zldvwdm)

[Jul 16, 2024](#_m932ebscgxb2)

[Jul 2, 2024](#_fcay4vwpa8cq)

[Jun 18, 2024](#_91k0s2jujpdj)

[Jun 4, 2024](#_t2sezu8m2sbr)

[May 21, 2024](#_wbejkf6lei22)

[May 7, 2024](#_ze5flbuvqc76)

[Apr 23, 2024](#_4e7buxnbv1ua)

[Apr 9, 2024](#_xjgonk92ss7w)

[Mar 26, 2024](#_ytoaytomf3tq)

[Mar 12, 2024](#_2kz8d3jp5mou)

[Feb 27, 2024](#_fc5yl7at4g4x)

[Feb 13, 2024](#_t4zppdwrzyzk)

# Oct 8, 2024

## Meeting Information

Meeting page: <https://lu.ma/2024-03-26-rust-libp2p>x

Video link: <https://meet.google.com/ftz-kjxw-vxe>

## Attendees

* @guillaumemichel
* @stormshield-frb
* @RawinSader
* @jxs
* @nazar-pc
* @jmininger
* @kalabukdima

## Agenda

* @stormshield-frb:
  + Any interest on a DialContext allowing to known who, where and why a Dial was triggered ?
  + Any interest to know when a Multiaddr got discovered in the peer (by different Behaviours, Kademlia, Identify, etc)

# 

# 

# Sep 24, 2024

## Meeting Information

Meeting page: <https://lu.ma/2024-03-26-rust-libp2p>x

Video link: <https://meet.google.com/ftz-kjxw-vxe>

## Attendees

* @guillaumemichel
* @p1r
* @kalabukdima

## Agenda

# Sep 10, 2024

## Meeting Information

Meeting page: <https://lu.ma/2024-03-26-rust-libp2p>

Video link: <https://meet.google.com/ftz-kjxw-vxe>

## Attendees

* ~~@dhuseby~~
* @jxs
* @nazar-pc
* @stormshield-frb
* @kalabukdima
* @p1r

## Agenda

* Old business
  + @p1r dev guild work update?
    - Are you stuck on anything?
  + libp2p Day @ Devcon update
    - We won’t have a standalone event due to budget constraints
    - I’m looking for a colocation
      * I have talked to Polygon and they had some options
      * It looks like we may be during the security summit just before Devcon: <https://www.eventbrite.com/e/defi-security-summit-2024-devcon-tickets-895458910707>
        + I think it makes more sense anyway considering our protocol work and security relevance
      * Anybody else know of groups holding events where we might be able to get colocation space?
* New business
  + Dev Guild cohort 1 is kicking off soon
    - I need mentorship volunteers
    - @p1r if you feel like you’re making progress and want to keep going, I’ll make a recommendation to get you into cohort 1 so you can continue

## Notes

David Perez

* Questions on testing, on the relay server example
  + <https://github.com/P1R/rust-libp2p/tree/identify_example>
  + <https://github.com/P1R/rust-libp2p/tree/relay-server-example>
* Talked about his work with the Dev Guild cohort1
  + Mentioned that he has a lot going on right now and isn’t sure when he will get to it. He said he will still try to join the calls and will try to learn more about it.

Francois R.

* Discussion about bugs on pr
  + Hasn’t forked yet and is planning to work on in the next month.
    - Doesn’t remember the bug, its an old commit, and it will take some time to rework it but is going to get to it.

Nazar

* Discussion about which version of libp2p

Dzmitry

* Discussion about reviewing Davids work

## Action Items

# Aug 27, 2024

## Meeting Information

Meeting page: <https://lu.ma/2024-03-26-rust-libp2p>

Video link: <https://meet.google.com/ftz-kjxw-vxe>

## Attendees

* @dhuseby
* @jxs
* @Wiezzel
* @p1r
* @stormshield-frb

## Agenda

* Old business
  + @Weizzel, gossip sub memory leak update?
  + @P1R how’s the devguild work going?
* New business

## Notes

* @weizzel gave up on trying to solve why gossip sub messages are in the network way longer
  + He ended up creating application specific message validation functions for preventing messages getting forwarded

## Action Items

# Aug 13, 2024

## Meeting Information

Meeting page: <https://lu.ma/2024-03-26-rust-libp2p>

Video link: <https://meet.google.com/ftz-kjxw-vxe>

## Attendees

* ~~@dhuseby~~
* @nazar-pc
* @p1r
* @guillaumemichel
* @stormshield-frb
* @Wiezzel
* @umgefahren

## Agenda

* Old business
  + I met with Joey Sarpong to discuss his company and his product that uses libp2p
  + It’s a good use case and will likely use js-libp2p
* New business
  + @Weizzel
    - Update on the gossip sub heartbeat messages sticking around way too long
    - This appears to be an eventual OOM condition due to not properly releasing old messages
  + @P1R
    - PL Dev Guild member
    - <https://github.com/libp2p/rust-libp2p/discussions/5533>
      * What do the specs say about the listen process?
      * Do we want to change the listen interface?
        + listen() returns a listen\_id
        + NewListenAddress events have a new listen\_id field
        + new ListenEnd event to signal that all side effects from the listen() call have been made
      * This is a breaking change to the API since nearly every peer using rust-libp2p handles NewListenAddress events. **We take breaking changes very seriously and they have to be very carefully handled.**
      * One possibility is to add a new listen\_with\_id() call and a NewListenAddresId event and ListenEndId event so that these changes are NOT breaking changes.
      * We added #[non\_exhaustive] to the event enumerations last year so we can add new events without breaking everybody’s code.
  + @umgefahren
    - AutoNATv2 Status
    - (optional) rust-libp2p on #![no\_std]
      * <https://github.com/umgefahren/rust-libp2p/tree/no-std>

## Notes

## Action Items

* Everybody email me a t-shirt color choice, size and mailing address so that I can ship you a libp2p day t-shirt.

# Jul 30, 2024

## Meeting Information

Meeting page: <https://lu.ma/2024-03-26-rust-libp2p>

Video link: <https://meet.google.com/ftz-kjxw-vxe>

## Attendees

* @dhuseby
* @guillaumemichel
* @stormshield-frb
* @p1r
* @threesonyaheadtop (Joey)
* @nazar-pc
* @Wiezzel
* Ingrid Gomes @wireshape\_admin
* Cristiano Policarpo - @policarpo\_0x

## Agenda

* Kademlia?
  + What’s the final answer to the proper way to resolve PeerID’s to multiaddr
    - Is it still adjusting bucket size?
    - fix(kad): correctly handle NoKnownPeers error when bootstrap [#5349](https://github.com/libp2p/rust-libp2p/pull/5349)
* @jxs mentioned idontwant in gossipsub is landed in the spec but not ready for upstreaming
* @Wiezzel: Is it possible to discard older messages in gossipsub (after 10 seconds) as no longer relevant?
  + No one knew the answer
* @dhuseby
  + What is the best way to have more than one custom protocol that depends on KAD?

## Notes

* @jxs IDONTWANT is…
  + Landed in rust
  + Under review for go
  + Issue out to Teku for jvm
* @Wiezzel
  + Using gossipsub to send ping messages
  + Is it possible to drop messages?
  + Set the history length to change how many messages are kept:
    - ​​<https://github.com/libp2p/rust-libp2p/blob/master/protocols/gossipsub/src/config.rs#L108>
  + MessageCache docs:
    - <https://github.com/libp2p/specs/blob/master/pubsub/gossipsub/gossipsub-v1.0.md#message-cache>
  + Maybe override the message validation?
    - If the validation function returns false does that prevent the forwarding of the message or does it just prevent it from being deliver
* @dhuseby
  + I followed up with Joey via email about his use case
* Question about the best way to have multiple custom behaviors that depend on KAD
  + The best practice is do the coordination at the swarm level and have a single KAD behavior
  + It is possible to compose their custom protocols with KAD but they will need to have different protocol identifiers and it will duplicate the state inside of the peer.

## Action Items

* Everybody email me a t-shirt color choice, size and mailing address so that I can ship you a libp2p day t-shirt.

# Jul 16, 2024

## Meeting Information

Meeting page: <https://lu.ma/2024-03-26-rust-libp2p>

Video link: <https://meet.google.com/ftz-kjxw-vxe>

## Attendees

* @nazar-pc
* @stormshield-frb
* @P1R
* @Wiezzel
* @jakubDoka
* @kamuik16

## Agenda

* @Wiezzel: Is it possible to discard older messages in gossipsub (after 10 seconds) as no longer relevant?
  + No one knew the answer
* What is the state of maintainers of rust-libp2p?
  + @jxs
  + @guillaumemichel specializing on Kademlia

## Notes

## Action Items

# Jul 2, 2024

## Meeting Information

Meeting page: <https://lu.ma/2024-03-26-rust-libp2p>

Video link: <https://meet.google.com/ftz-kjxw-vxe>

## Attendees

* @dhuseby
* @guillaumemichel
* @jxs
* @stormshield-frb

## Agenda

* @dhuseby
  + libp2p Day @ IPFS Camp is a GO!: <https://lu.ma/fowy8k15>
  + Looking for any maintainers to join me on the Maintainer’s Q&A panel
* Kademlia?
  + What’s the final answer to the proper way to resolve PeerID’s to multiaddr
    - Is it still adjusting bucket size?
    - fix(kad): correctly handle NoKnownPeers error when bootstrap [#5349](https://github.com/libp2p/rust-libp2p/pull/5349)
* @jxs release update?
* PR updates
  + @Wiezzel PR for proposed changes to the KAD behaviour?
  + @stormshield-frb update on feat(kad): improve automatic bootstrap [#5474](https://github.com/libp2p/rust-libp2p/pull/5474)
  + feat(swarm): Show task spawn paths in tokio-console [#5465](https://github.com/libp2p/rust-libp2p/pull/5465)
  + @guillaumemichel feat(kad): configurable bucket size [#5414](https://github.com/libp2p/rust-libp2p/pull/5414)
  + fix(swarm): rewrite NetworkBehaviour macro for more optimal code gen: [#5303](https://github.com/libp2p/rust-libp2p/pull/5303)
  + @stormshield-frb feat(kad): introduce AsyncBehaviour [#5294](https://github.com/libp2p/rust-libp2p/pull/5294)

## Notes

* @guillaumemichel update on the KAD PeerID resolution
  + Was fixed and the PR
* <https://github.com/ethereum/portal-network-specs/issues/232>
  + Tracking the potential use of libp2p to access the Portal Network
  + Potential opportunity for Discv5/Discv6 and devp2p interoperability
* Merged: chore(kad): remove unrequired generics [#5476](https://github.com/libp2p/rust-libp2p/pull/5476)
* @jxs mentioned idontwant in gossipsub is landed in the spec but not ready for upstreaming

## Action Items

# Jun 18, 2024

## Meeting Information

Meeting page: <https://lu.ma/2024-03-26-rust-libp2p>

Video link: <https://meet.google.com/ftz-kjxw-vxe>

## Attendees

* @dhuseby
* @Wiezzel
* @stormshield-frb
* @guillaumemichel
* @jxs
* @David Perez

## Agenda

* @gui talk about kademlia and bucket sizes
* <https://github.com/libp2p/rust-libp2p/pull/5349> should be good to merge!
  + <https://github.com/libp2p/rust-libp2p/issues/5341> what should we do about this one ?
* How to get discovered addresses from kademlia (when they’re not cached)?
* Proposed idea to make .listen() more useful
  + ListenCfg passed in to specify filtering criteria (e.g. loopback addresses, private addresses, public addresses) plus multiaddr and timeout
  + .listen() to return a “listen transaction ID”
  + A series of NewListenAddrs { addr, listen\_id }, FilteredListenAddr { addr, listen\_id }, ListenComplete { listen\_id }, ListenTimeout { listen\_id }

## Notes

* @Wiezzel asked @Gui about the proper way to resolve a PeerID into a multiaddr
  + If the bucket is full, the peer isn’t added
  + Proposed changed is to modify GetClosesPeersOk so that it contains a Vec of tuples of (PeerId, Multiaddr) so that the multiaddrs are available to the caller.
  + @jxs suggested that at some point we should probably consider adding a [PeerStore](https://github.com/libp2p/rust-libp2p/issues/4103) as a generic behaviour that responds to all of the events that report discovered peers from all other behviours
* @gui brought up [#5349](https://github.com/libp2p/rust-libp2p/pull/5349) is ready to merge and unblock the release
  + @jxs will carve a new breaking release for just kademlia crate with the kademlia fixes
    - New full libp2p release due when @umgafahren lands transport refactor
* @stormshield-frb asked about automatic bootstrapping where the default timeout might be too short
  + Related to [#5341](https://github.com/libp2p/rust-libp2p/issues/5341)
  + @stormshield-frb would like to make both the automatic and periodic bootstrap timeouts configurable as Option<Duration> in the Kad config.
  + @gui asked why the automatic bootstrap was necessary
  + The current behaviour is that the automatic bootstrap timeout is hard coded to 500ms and the periodic bootstrap timeout is configurable as Option<Duration>
  + This was the fix for bootstrap operations sometimes taking a long time due to failed bootstrap attempts
  + @gui suggests changing the automatic bootstrapping behavior to only happen when the number of peers in the routing table is less than the bucket size.
* @dhuseby confirmed new users of libp2p: <https://kindelia.org/>
  + Blockchain for interaction net distributed computation
  + Uses rust-libp2p at the p2p layer
* @dhuseby asked about the best practices around calling .listen() such that a peer only listens on the interfaces that it should listen on.
  + The typical behavior others are using is to just YOLO it by listening on 0.0.0.0 and going for it.
  + @dhuseby outlined a need to be able to discover interfaces and only listen on interfaces of a given class (e.g. loopback, non-routable, routable or any combination of them).
    - The right answer is to just collect some example code and existing best practices for common environments (e.g. Docker image with DHCP assigned address and NAT) and publish a blog post.

## Action Items

* @gui will start a thread on slack with @gallargh and @dhuseby to add @jxs and @gallargh to the multiformats github org.
* @Wiezzel to submit a PR for the proposed changes to the KAD behaviour
* @stormshield-frb to submit a PR to modify the automatic bootstrap behavior so that it only triggers when the number of peers in the routing table is less than the KAD bucket size

# Jun 4, 2024

## Meeting Information

Meeting page: <https://lu.ma/2024-03-26-rust-libp2p>

Video link: <https://meet.google.com/ftz-kjxw-vxe>

## Attendees

* @dhuseby
* @dariusc93
* @nazar-pc
* @Wiezzel
* @drHuangMHT
* @dgarus

## Agenda

* libp2p Day @ EthCC in Brussels is on.
  + Call for speakers!!
  + Submit them here: <https://airtable.com/appM094R1Ma5HG757/shrWn6XaRgUkYWPm3>
  + @dhuseby is the track lead so anything remotely libp2p related gets approved

## Notes

* @wiezzel
  + Microsoft and jemalloc allocators work significantly better than the default allocator in libp2p
    - <https://github.com/rust-lang/rust-analyzer/issues/1441>
* @jxs
  + Releasing a new version of request-response.
  + <https://github.com/libp2p/rust-libp2p/blob/master/protocols/request-response/CHANGELOG.md>
  + @umgafahren just pinged @jxs about final review on his work.
* @wiezzel
  + Question about the correct way to use kademlia to resolve a PeerID into a multiaddr
  + He’s doing a kad query and it comes back successful and then he tries to dial it but the dial is failing.
* @dgarus
  + Brought up his past PR on WebTransport <https://github.com/libp2p/rust-libp2p/pull/4874>
* @nazar
  + Kademlia bootstrap gets stuck forever in some cases: <https://github.com/libp2p/rust-libp2p/issues/5432>
  + kad: get\_providers() sometimes resolves with FinishedWithNoAdditionalRecord and no closest peers: <https://github.com/libp2p/rust-libp2p/issues/5418>
  + Asked about getting help resolving kademlia issues
    - @guillaumemichel is the primary reviewer now
    - Fixing the issues he’s seeing will require deep knowledge of how queries work

## Action Items

* @dhuseby follow up with @umgafahren about the autonatv2 and transport refactor work.
* @gui to attend next meeting to talk about kademlia and bucket sizes.

# May 21, 2024

## Meeting Information

Meeting page: <https://lu.ma/2024-03-26-rust-libp2p>

Video link: <https://meet.google.com/ftz-kjxw-vxe>

## Attendees

* @dhuseby
* @jxs
* @Wiezzel
* @stormshield-frb
* @p1r
* @psyferlab
* @guillaumemichel
* @gurinderu

## Agenda

* Old business:
  + @jxs Hannes?
    - I talked with him and he said he’s blocked on @sukun approval
  + @dhuseby
    - No movement on the address translation for QUIC
* New business:
  + @Wiezzel update
  + @jxs is planning a breaking changes release for the TLS transport crate which means the top level rust-libp2p crate needs a new release

## Notes

* Stormfront has started work with @umgafahren on the AutoNATv2 improvement
* @Wiezzel update on minalloc in their libp2p peer and it reduced the memory footprint significantly.
  + Begs the question of what is going on with the default allocator such that it is significantly worse than minalloc and other allocators
  + Rust discussion about different allocators: <https://github.com/rust-lang/rust-analyzer/issues/1441>
* @psyferlab asked about bootstrapping with rust-libp2p
  + Is there a way to persist the routing table between executions? Not at the moment. It’s only an in-memory routing table.
  + More info on bootstrapping: <https://blog.ipfs.tech/2023-rust-libp2p-based-ipfs-bootstrap-node/>
  + More info on peer store: <https://github.com/libp2p/rust-libp2p/issues/4103>
  + Ipfs-embed is still maintained
* @jxs
  + Asked @guillaumemichel for a minor release of the kademlia crate
  + @Wiezzel experimented with increasing the bucket size in the kademlia crate, @jxs wants to leave it out of the next release for now.
  + Primary goal is to just update the QUIC crate.
    - Depends on TLS transport create which can have a non-breaking release
* @jxs update on upnp.
  + @stormfront-frb fixed the panics in upnp…do they need to keep their fork?
  + @jxs thinks the upstream fixes the bugs that @stormfront-frb also fixed…can we close the PR?
    - <https://github.com/libp2p/rust-libp2p/pull/5272>
  + The panics are fixed but there

## Action Items

* @dhuseby to reach out to @dariusc93 and @drHuangMHT to see if they will attend the meeting
* @jxs working on gossip sub spec and will attend the next spec meeting on Jun 4, 2024
* @dhuseby to contact maintainers for protocols to get them into the spec meeting

# May 7, 2024

## Meeting Information

Meeting page: <https://lu.ma/2024-03-26-rust-libp2p>

Video link: <https://meet.google.com/ftz-kjxw-vxe>

## Attendees

* @dhuseby
* @jxs
* @Wiezzel
* @guillaumemichel
* @nazar

## Agenda

* Old business:
  + @jxs carved a new release of subcrates
    - <https://github.com/libp2p/rust-libp2p/discussions/5351>
    - CVE: <https://nvd.nist.gov/vuln/detail/CVE-2024-32984>
  + @dhuseby
    - Still looking into the [Transport redesign](https://github.com/libp2p/rust-libp2p/pull/4568) by @umgefahren
    - No answers yet…will need to ask more people
  + @dav1do, @nathanielc
    - Any update on the memory leak in the ceramic peer as mentioned in the last meeting.
* New business:
  + @Wiezzel forked rust-libp2p and wants to upstream changes
    - Address translation for QUIC
      * Should use the listen address for outgoing connections and translation shouldn’t be needed
      * It’s not working like that on routers
      * <https://github.com/libp2p/rust-libp2p/pull/4896>
    - There is address translation for TCP but not for UDP
      * With AutoNATv2 this will break
      * The spec does not have any guidance here

## Notes

* @jxs to submit a PR to [rust sec list](https://rustsec.org/contributing.html)
* @nazar announced the update to his network and are in the process of updating
* @wiezzel mentioned he’s had good experience with heaptrack
  + <https://github.com/KDE/heaptrack>
  + Works well with Rust
* @nazar suggested using <https://crates.io/crates/mimalloc> for debugging rust
* @Wiezzel uses QUIN for QUIC and it has an MTU discovery mechanism
  + Proposal to enable passing in the MTU config when configuring QUIC
  + Right now it isn’t configurable from the rust-libp2p side
  + @nazar you can disable path MTU discovery but you can’t configure it
* @nazar mentioned that local message passing is much more difficult than using NATS
  + <https://github.com/subspace/subspace/pull/2659>
  + We should take this seriously and see if we can address these challenges in the rust-libp2p
  + <https://github.com/subspace/subspace/blob/f8bf2809a565d3044623838a08e7417eade95168/crates/subspace-networking/src/node.rs#L309-L353>

## Action Items

* @jxs to reach out to Hannes @umgefahren about the transport work he’s been working on
  + <https://github.com/libp2p/rust-libp2p/pull/4568>
* @dhuseby to reach out to Ceramic about the memory leak they are seeing
* @dhuseby investigate the address translation for QUIC in AutoNATv1 and what the spec says
* @dhuseby to follow up with @nazar to do an interview and gather the pain points / challenges that made him consider using NATS instead.
  + Also will reach out to Bacalhau to get information from them to incorporate this.
  + <https://docs.nats.io/nats-concepts/subjects>
  + The goal is to gather feedback to form a set of recommendations on improving libp2p so that NATS isn’t such an obvious alternative
  + <https://github.com/subspace/subspace/blob/f8bf2809a565d3044623838a08e7417eade95168/crates/subspace-networking/src/node.rs#L309-L353>
  + <https://www.youtube.com/playlist?list=PLgqCaaYodvKY22TpvwlsalIArTmc56W9h>
  + <https://github.com/libp2p/rust-libp2p/pull/5294>
* @dhuseby to set up a rust-libp2p Project under libp2p to track work items
  + @Weizzel to submit PR for MTU discovery changes

# Apr 23, 2024

## Meeting Information

Meeting page: <https://lu.ma/2024-03-26-rust-libp2p>

Video link: <https://meet.google.com/ftz-kjxw-vxe>

## Attendees

* @dhuseby
* @guillaumemichel
* @stormshield-frb
* @Wiezzel
* @pidb
* @dav1do
* @nathanielc

## Agenda

* **NOTE:** back to all meetings being NA+EU friendly as @jxs and @AgeManning talk on the regular so there’s no need for an AUS friendly meeting ATM. The calendar feed has been updated.
* libp2p rPGF: <https://blog.libp2p.io/2024-04-08-libp2p-rpgf/>
* Release update from @jxs?
  + Permissions were fixed.
  + Anything you need help with?
* Old business:
  + uPNP panic [#5444](https://github.com/sigp/lighthouse/issues/5444) is fixed.
    - [#5272](https://github.com/libp2p/rust-libp2p/pull/5272) UPnP fixes from @stormshield-frb
  + FIND\_NODE [#5269](https://github.com/libp2p/rust-libp2p/issues/5269) has merged (thanks @Gui)
    - @achingbrain, js-libp2p behavior fixed: [#2499](https://github.com/libp2p/js-libp2p/pull/2499)
    - Proposed spec change PR: [#609](https://github.com/libp2p/specs/pull/609)
* Ask about AutoNAT v2 status and Transport redesign

### Notes

* Reached out to r/rust to invite them to join us: <https://www.reddit.com/r/rust/comments/1ca3ku5/comment/l0u0rms/>
* @guillaumemichel plans to present at libp2p day in July: <https://pl-strflt.notion.site/OmniDex-Composable-DHT-daf900796597425c8ca6ecc9563dae9c>

### Action Items

* @dhuseby to follow up @jxs about the rust release
* @guillaumemichel to reach out to [KademliaCon 2023](https://hackmd.io/@matt/kadcon2023) organizer to see if there will be a Kademliacon 2024
* @dhuseby to look into the status of the [Transport redesign](https://github.com/libp2p/rust-libp2p/pull/4568) by @umgefahren under grant work.
  + There’s a branch that was opened in September 2023 that is up for merging but it is getting old.
  + This is blocking the AutoNAT v2 implementation
* @stormshield-frb to pick up work on AutoNAT v2
  + They have cherry picked from @umgefahren’s branch to make their fork work
  + Comments in this PR: <https://github.com/umgefahren/rust-libp2p/pull/1>
* @dav1do and @nathanielc came in late to ask about when to register a waker or not
  + The code that’s getting starvations is here: <https://github.com/ceramicnetwork/rust-ceramic/blob/main/recon/src/protocol.rs>
  + They are also seeing memory usage increasing over time.
    - They using kademlia, identify, dcutr, and the custom protocol linked above
    - There’s a leak associated with peer count and general memory usage

# Apr 9, 2024

## Meeting Information

Meeting page: <https://lu.ma/2024-03-26-rust-libp2p>

Video link: <https://meet.google.com/ftz-kjxw-vxe>

## Attendees

* Dave
* @jxs
* @AgeManning

## Agenda

* libp2p rPGF: <https://blog.libp2p.io/2024-04-08-libp2p-rpgf/>
* Release Update

### Notes

* Release is on track
  + UPnP crate has a panic reported
* <https://github.com/libp2p/rust-libp2p/pull/5294>
* Parity Update

### Action Items

* Follow up with Parity Dmitry about getting him to attend a rust-libp2p meeting to discuss their recently filed Issues
* Engage on the r/rust subreddit and see if we can get a shout out about our rust-libp2p meetings

# Mar 26, 2024

## Meeting Information

Meeting page: <https://lu.ma/2024-03-26-rust-libp2p>

Video link: <https://meet.google.com/ftz-kjxw-vxe>

## Attendees

* Dave
* Joao (sigma prime)
* Nazar
* Gui (ipshipyard)
* Francois (stormshield)

## Agenda

* Project management using GH projects?
* PRs / Issues
  + <https://github.com/libp2p/rust-libp2p/pull/5255> #send-it fixed?
  + <https://github.com/libp2p/rust-libp2p/pull/5237> dep bumps not working?
    - Compile errors: <https://github.com/libp2p/rust-libp2p/actions/runs/8377404006/job/22939414539?pr=5237#step:6:507>
  + <https://github.com/libp2p/rust-libp2p/issues/5269> kad find node empty response
  + <https://github.com/sigp/lighthouse/issues/5444>

### Notes

* #send-it works again. Mergify bug is fixed.
* Dependabot deps bumps are working again.
* Joao wants to make a release
* Re: #5269
  + This is an issue about different behavior between go-libp2p and rust-libp2p. When sending a FIND\_NODE request for Peer A directly to Peer A, rust returns an empty response where go returns just Peer A.
  + The spec is ambiguous: <https://github.com/libp2p/specs/tree/master/kad-dht#peer-routing>
* UPnP
  + Panic in the upnp code on reconnection. @joao hasn’t been able to reproduce it
  + <https://github.com/sigp/lighthouse/issues/5444>

### Action Items

* Re: next rust-libp2p release
  + Dave to figure out the permissions for Joao to be able to cut a release of rust-libp2p
  + Get feedback from Parity on the changes since the kademlia changes are potentially breaking
    - Identify contacts in Parity we can talk to about the v0.52.2 upgrade in Substrate
      * Set up a dialog with them about their desire to replace libp2p with lightp2p
      * <https://github.com/dmitry-markin> (@dmitry-markin:parity.io on matrix)
      * <https://github.com/altonen>
      * <https://github.com/paritytech/polkadot-sdk/issues/3216#issuecomment-1930711630>
      * <https://github.com/paritytech/polkadot-sdk/pull/3375#issuecomment-1952054137>
      * <https://github.com/paritytech/polkadot-sdk/pull/1631>
      * <https://github.com/paritytech/polkadot-sdk/pull/2944>
      * <https://youtu.be/9jR9EWZ2bO8?feature=shared>
  + Joao will cut a release
* Re #5269
  + Tag @max on the issue to see if he can add information on why it changed last year
  + Create an issue in the spec repo to come to a conclusion on what the correct behavior is and get it into the specification.
  + Update the changes to go and rust to make them the same
  + Tag @achingbrain to check js-libp2p behavior
* Re: UPnP
  + @francois to try to upstream the libp2p-upnp changes to see if they fix the panic reported here: <https://github.com/sigp/lighthouse/issues/5444>

### Next Meeting Agenda



# Mar 12, 2024

## Meeting Information

Meeting page: <https://lu.ma/2024-03-12-rust-libp2p>

Video link: <https://meet.google.com/ftz-kjxw-vxe>

## Attendees

Dave Grantham (@dhuseby, [dave.huseby@libp2p.io](mailto:dave.huseby@libp2p.io))

William Blankenship(@retrohacker, william.blankenship@webai.com)

Marco (@marcopolo)

João Oliveira jxs

Adrian (@AgeManning)

## Agenda

### Follow-ups

* Old business
  + (@joao) Updates on the CI runners situation
  + (@dhuseb) Update on identify contacts in Parity we can talk to about the v0.52.2 upgrade in Substrate
    - Set up a dialog with them about their desire to replace libp2p with lightp2p
    - <https://github.com/dmitry-markin>
    - <https://github.com/altonen>
    - <https://github.com/paritytech/polkadot-sdk/issues/3216#issuecomment-1930711630>
    - <https://github.com/paritytech/polkadot-sdk/pull/3375#issuecomment-1952054137>
    - <https://github.com/paritytech/polkadot-sdk/pull/1631>
    - <https://github.com/paritytech/polkadot-sdk/pull/2944>
    - <https://youtu.be/9jR9EWZ2bO8?feature=shared>
* New Business
  + Next meeting: 2024-03-26, what time?
  + Decision: rotating each meeting between Americas+EU and the Americas+AUS timeslots

### Notes

* CI Runners
* Promoting @jxs (joao)?
  + https://github.com/libp2p/github-mgmt/pull/211

### Action Items

* Schedule the next meeting for the Americas + EU friendly time slot

### Next Meeting Agenda



# Feb 27, 2024

## Meeting Information

Meeting page: https://lu.ma/qpgpsvsk

Video link: https://meet.google.com/ftz-kjxw-vxe

## Attendees

Dave Grantham (@dhuseby, [dave.huseby@libp2p.io](mailto:dave.huseby@libp2p.io))

Nazar Mokrynskyi (@nazar-pc)

Joao

Francois

## Agenda

### Updates

* Schedule the next rust-libp2p Open Maintainer’s call to be EU/AUS friendly
  + March 6, ??
  + @joao with talk to Adrian and Thomas
* Libp2p Day @ EthDenver is on 2/29 at The Slate Hotel near the main event
  + Still looking for more speakers
* Cleaned up libp2p.io site coming: <https://small-math-1407.on.fleek.co/>
* EthDenver site coming together: <https://small-math-1407.on.fleek.co/ethdenver-2024/>
* EthCC event planning
  + Brussels July 8-11
  + Radisson in downtown Brussels
  + We get 100 people room with a stage+A/V and rounds all day, two days
  + Talks in the morning both days, with a “let’s hack together” in the afternoons both days.
  + Focus:
    - Web based libp2p projects/demos
* Discussed issues
  + <https://github.com/quinn-rs/quinn/issues/1750>
    - Confusing…there’s a trick here <https://docs.rs/libp2p/latest/libp2p/trait.Transport.html>
    - But it is confusing to new users
  + Need to update performance testing numbers for TCP vs QUIC
  + <https://github.com/libp2p/rust-yamux/pull/176>
    - Max’s updated yamux with dynamic window sizing
  + Parity’s fork (lightp2p)
    - Communication issues primarily between libp2p and Parity

### Notes

* Priorities:
  + Move to merge on GH
  + Retire the self-hosted runners
  + Update WebRTC library to str0m
  + Get new perf test numbers so we can use them to guide further engineering

### Action Items

* @joao and @dave will work with Thomas to get rust-libp2p moved to merge queue
* Ask Piotr and Laurent to join us in a slack group with @joao to work on a plan to move from self-hosted runners to GH runners and move to the merge queue to get our spend down to $0 for runners.
  + We didn’t use merge queue before because we couldn’t customize the merge message
  + Mergify is broken
  + <https://github.com/libp2p/rust-libp2p/discussions/4586>
* Identify contacts in Parity we can talk to about the v0.52.2 upgrade in Substrate
  + Set up a dialog with them about their desire to replace libp2p with lightp2p
  + <https://github.com/dmitry-markin>
  + <https://github.com/altonen>
  + <https://github.com/paritytech/polkadot-sdk/issues/3216#issuecomment-1930711630>
  + <https://github.com/paritytech/polkadot-sdk/pull/3375#issuecomment-1952054137>
  + <https://github.com/paritytech/polkadot-sdk/pull/1631>
  + <https://github.com/paritytech/polkadot-sdk/pull/2944>
  + <https://youtu.be/9jR9EWZ2bO8?feature=shared>

### Next Meeting Agenda



# Feb 13, 2024

## Meeting Information

Meeting page: <https://lu.ma/rust-libp2p>

Video link: <https://meet.google.com/ftz-kjxw-vxe>

Recording: [Meet Recordings](https://drive.google.com/drive/folders/1xZ_IdfbD8ow9SmMi38rqFXZ_4YpGPtHt?usp=sharing)

## Attendees

Dave Grantham (@dhuseby, [dave.huseby@libp2p.io](mailto:dave.huseby@libp2p.io))

Hannes Furmans (@umgefahren)

Saad Talaat (@SaadTalaat)

Francois Ribeau (@stormshield-frb)

João Oliveira (@jxs)

William Blankenship (@retrohacker)

Nazar Mokrynskyi (@nazar-pc)

## Agenda

### Updates

* My main priority is merging AutoNATv2 to ensure [@umgefahren](https://github.com/umgefahren) can finish his contract and we haven't done that massive piece of work for nothing.
* It would be cool to also land WebTransport server support in some form or another. See [feat: add WebTransport protocol #4874](https://github.com/libp2p/rust-libp2p/pull/4874). I am open to volunteering some time to this as well (cc [@dgarus](https://github.com/dgarus) )
* Outside of that, I am not planning on spending unpaid time on rust-libp2p in the future.

### New Business

* Maintainership?
  + Any volunteers
* [EngRes The Gathering](https://lu.ma/Engres-The-Gathering-02-15-24) is on 2/15
  + Rust-libp2p updates?
  + Anybody want to volunteer to present?
* Libp2p Day @ EthDenver is on 2/29 at The Slate Hotel near the main event

### Notes

* Stormshield using rust-libp2p for ZTNA applications
  + Interested in helping with AutoNAT
* WebAI build a distributed filesystem similar to IPFS that is not compatible for IPFS
* Hannes Furmans on a grant from Filecoin foundation <https://github.com/filecoin-project/devgrants/issues/1421>
  + Maintainership changes and school have gotten in the way of completion
* What is needed to keep rust-libp2p going?
  + Sigma Prime is blocked on a breaking change (<link>)
    - Concern that it shouldn’t land without Max or Thomas’ blessing
* On grants going forward we need to add stakeholders/advisors to the grant application so that companies needed the new feature are involved.
* BTLE: <https://github.com/libp2p/rust-libp2p/pull/971>

### Action Items

* Set up a slack channel for AutoNat v2 landing
* Sigma Prime is blocked [on a breaking change](https://github.com/libp2p/rust-libp2p/pull/4818)
* Dave to reach out to Parity about Substrate replacing rust-libp2p with a new [libp2p implementation](https://github.com/altonen/litep2p)
  + https://github.com/altonen/litep2p
* Dave to ask about libp2p Day travel grants and report back
* Dave to find official blog post on the future of libp2p/IPFS
* Joao to talk to Thomas about a EU/AUS friendly rust-libp2p meeting time
* Start an Issue for the next release of rust-libp2p (Joao will talk to Thomas)
* Review change to release procedure for when we have breaking changes: <https://github.com/libp2p/rust-libp2p/issues/3202>
* Reach out to Quinn about IPv6 warning messages <https://github.com/quinn-rs/quinn/issues/1750>
  + @joao will look into it

### Next Meeting Agenda

* Adjust release procedure for when we have breaking changes per decision made in: <https://github.com/libp2p/rust-libp2p/issues/3202>