Networking Challenges Ahead for Firefox

An Invitation to Collaborate

Max Inden

MIR^3 2025

What in the web isn’t complicated enough yet?

Proxying today

- Plenty of protocols thus far
  - SOCKS
  - HTTP CONNECT
  - …
- How to proxy HTTP/3? UDP on top of TCP?
- What if we could send encrypted, (congestion controlled,)but still unreliable Datagrams with a fallback?

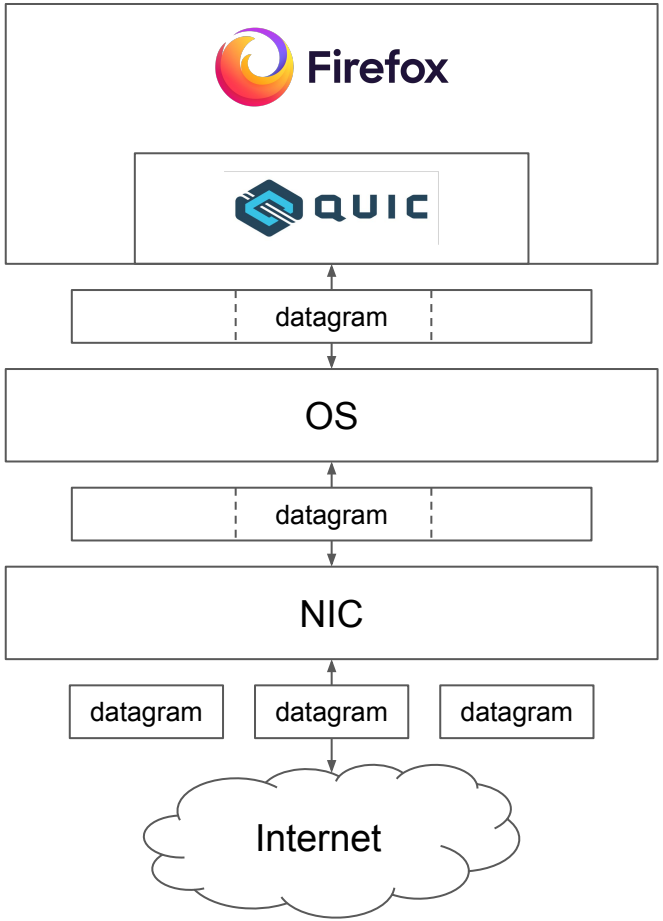
➡️ Inner connection ⬇️ Outer connection	HTTP1	HTTP2	HTTP3	WebRTC
HTTP1	**now**: HTTP CONNECT **future**: connect-tcp with fallback to HTTP CONNECT		**now**: unsupported **future**: connect-udp via HTTP Capsules	**now**: unsupported **future**: connect-udp via HTTP Capsules + connect-udp-listen
HTTP2			**now**: unsupported **future**: connect-udp via HTTP Capsules	**now**: unsupported **future**: connect-udp via HTTP Capsules + connect-udp-listen
HTTP3	**now**: unsupported **future**: connect-tcp with fallback to HTTP CONNECT	**now**: unsupported **future**: connect-tcp with fallback to HTTP CONNECT	**now**: unsupported **future**: connect-udp with fallback to HTTP Capsules	**now**: unsupported **future**: connect-udp either via HTTP Datagrams if QUIC Datagram extension is available, or HTTP Capsules if not + connect-udp-listen
WebRTC	never	never	never	never

MASQUE use-cases

- Single & two hop proxy (like iCloud Private Relay)
- Selective proxying
  - Web trackers
  - FavIcon fetch when restoring profile
  - …
- MASQUE vs OHTTP
  - MASQUE: replay defense, forward secrecy
  - OHTTP: request/response, unlinkeability
- Double congestion control a problem in masque
- Research questions:
  - Performance
  - Privacy

Fast UDP I/O

- Operating system support
  - E.g. no official multi-message on MacOS
  - Buggy segmentation offloading on Windows
- Driver support
  - Crashing drivers on segmentation offloading
- Research questions
  - Conflict with pacing
  - Tuning of constants (e.g. segment size)
  - Optimize packet building (e.g. memory management)



Happy Eyeball v3

- Why v3?
  - Firefox implements v1 today
  - Suboptimal QUIC prioritization
  - Aggressive fallback to HTTP/1 and HTTP/2
- What's New in v3
  - Adds support for HTTPS resource records
  - Helps discover alternative endpoints, protocol support (e.g. HTTP/3), and ECH configs
  - Goal: prefer IPv6 and QUIC
- Research questions
  - Performance impact on connection establishment latency
  - Constant tuning (e.g. *Connection Attempt Delay*)

DOH

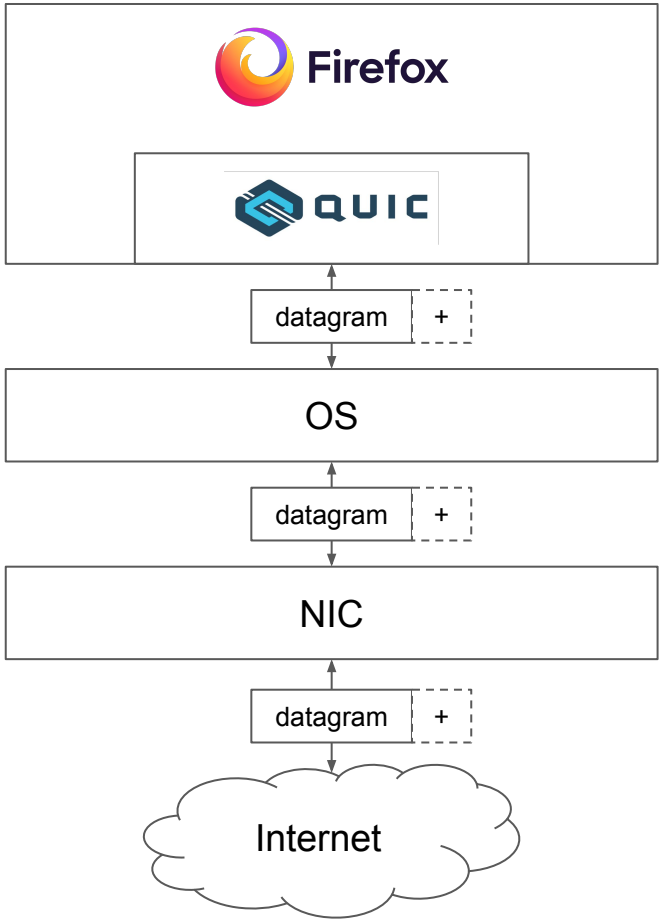
- Improve **T**rusted **R**ecursive **R**esolver program
- Fallback via *Discovery of Designated resolvers*
  - https://datatracker.ietf.org/doc/rfc9462/
- Rollout on Android
- Research questions
  - Performance impact of DOH
  - Global deployment
  - Optimizations (e.g. QUIC 0-RTT)

Replacing getaddrinfo with native DNS implementation

- Asynchronous resolution
  - esp. Relevant for Happy Eyeball v3
- HTTPS records on all platforms
  - Thus more QUIC
- Bypass various platform specific bugs
- E.g. using Rust library HickoryDNS
- Valentin's FOSDEM 2025 talk
- Research questions:
  - Operating specific optimizations

PLPMTUD

- Packetization Layer Path MTU Discovery for Datagram Transports (RFC 8899)
- probe maximum MTU on path
- 1280 bytes vs 1500 bytes makes a difference
- See also
  - Firefox QUIC implementation
  - Custura, A., Fairhurst, G., & Learmonth, I.: “**Exploring usable Path MTU in the Internet**”
- Research questions
  - Best starting value?
  - Search strategy? Binary search?



QUIC ACK frequency

- QUIC Acknowledgment Frequency IETF draft
- say a QUIC receiver ACKs every second packet on a 1 Gbit/s transfer
- 1 Gbit/s / 8 / 1500 / 2 ~= 40k ACK / s
- instead have the sender propose an ACK rate to the receiver
- Research questions
  - Relevancy given segmentation offloading and pacing
  - Performance impact on high throughput connections
  - Downsides on ACK clocking, RTT measurement, … in heterogeneous networks

WebTransport

- MOQ
- WebTransport on HTTP/2
- Firefox is at draft 4 (latest draft 13)
- Research questions
  - (real-time) media performance on WebTransport

QUIC Congestion Control

- HyStart++ or SEARCH
- ECN
  - Both marking and reporting rolling out as we speak
  - ~50% of outbound paths ECN capable
- L4S
- Research questions
  - Improvements to slow start and Cubic
  - TCP Prague implementation in Firefox’s QUIC stack
  - L4S impact for Firefox (down- and upload)

Privacy / Censorship resistance

- Encrypted Client Hello
  - Low adoption (i.e. basically Cloudflare only)
  - Easy to censor today
- Growing interest, e.g. IRTF armor
- Various tricks to make QUIC censorship harder
  - https://gfw.report/publications/usenixsecurity25/en/
- Research questions
  - Make (ECH / QUIC / …) censorship harder

Security

- Socket process
  - Firefox is already a multi-process application.
  - Network is not yet isolated in its own process.
  - Work-in-progress implementation exists
  - See about:config `network.http.network_access_on_socket_process.enabled`
- Oxidation

Firefox is open-

- source
  - https://firefox-source-docs.mozilla.org/setup/contributing_code.html
  - https://github.com/mozilla/nego
- data
  - https://dictionary.telemetry.mozilla.org/

Connection Migration

- Solution to the parking lot problem - moving from wifi to cellular
- Helpful on NAT rebinding

Steps

1. Client probes new path (PATH_CHALLENGE)
2. Client switches to new path

Get involved!

- contribute to Firefox
  - https://firefox-source-docs.mozilla.org/setup/contributing_code.html
- contribute to github.com/mozilla/neqo
  - Rust codebase
  - modern transport protocol
  - Look for good-first-issue.
- reach out:
  - post on GitHub
  - https://matrix.to/#/#necko:mozilla.org
  - https://matrix.to/#/#neqo:mozilla.org