

HTTP/3

Péter Jeszenszky

Faculty of Informatics, University of Debrecen

December 8, 2024

Terms of Use

This work is licensed under a [Creative Commons](#) “[Attribution 4.0 International](#)” license.



What is HTTP/3?

- The latest major version of HTTP that is based on the QUIC transport protocol.
- A mapping of HTTP semantics over the QUIC transport protocol.

QUIC

- QUIC is a secure general-purpose transport protocol that is built on the top of UDP.
- Pronunciation: quick

Development

- HTTP/3 and QUIC are both developed by the QUIC Working Group of IETF.
- Website: <https://quicwg.org/>

Specifications (1)

QUIC:

- Martin Thomson. *RFC 8999: Version-Independent Properties of QUIC*. May 2021. <https://www.rfc-editor.org/rfc/rfc8999>
- Jana Iyengar (ed.), Martin Thomson (ed.). *RFC 9000: QUIC: A UDP-Based Multiplexed and Secure Transport*. May 2021. <https://www.rfc-editor.org/rfc/rfc9000>
- Martin Thomson (ed.), Sean Turner (ed.). *RFC 9001: Using TLS to Secure QUIC*. May 2021. <https://www.rfc-editor.org/rfc/rfc9001>
- Jana Iyengar (ed.), Ian Swett (ed.). *RFC 9002: QUIC Loss Detection and Congestion Control*. May 2021. <https://www.rfc-editor.org/rfc/rfc9002>

Specifications (2)

HTTP/3:

- Mike Bishop (ed.). *RFC 9114: HTTP/3*. June 2022.
<https://www.rfc-editor.org/rfc/rfc9114.html>
- Charles 'Buck' Krasic, Mike Bishop, Alan Frindell (ed.). *RFC 9204: QPACK: Field Compression for HTTP/3*. June 2022.
<https://www.rfc-editor.org/rfc/rfc9204.html>

Preliminaries (1)

- QUIC was originally designed and implemented by Jim Roskind at Google in 2012, and was publicly announced in 2013.
- See: Jim Roskind. *Experimenting with QUIC*. June 27, 2013.
<https://blog.chromium.org/2013/06/experimenting-with-quit.html>

Preliminaries (2)

- Thereafter, Google has started to use HTTP/3 widely in its services (e.g., Gmail, YouTube) and in the Chrome browser.
- QUIC was submitted for standardization to IETF in June 2015.
- See:
 - Janardhan Iyengar, Ian Swett. *QUIC: A UDP-Based Secure and Reliable Transport for HTTP/2*. June 17, 2015.
<https://tools.ietf.org/html/draft-tsvwg-quic-protocol-00>
 - *QUIC, a multiplexed stream transport over UDP*
<https://www.chromium.org/quic/>

Adoption (1)

- HTTP/3-enabled websites:
 - Google <https://www.google.com/>
 - Facebook <https://www.facebook.com/>
 - Instagram <https://www.instagram.com/>
 - YouTube <https://www.youtube.com/>
 - ...
- A tool to check whether a server supports HTTP/3:
 - HTTP3Check <https://http3check.net/>

Adoption (2)

Adoption statistics:

- [Usage statistics of QUIC for websites \(W3Techs \)](#)
- [HTTP Archive: State of the Web—HTTP/3 Support](#)
- [The 2022 Web Almanac—HTTP/3](#)

Implementations (1)

A list of implementations:

<https://github.com/quicwg/base-drafts/wiki/Implementations>

Implementations (2)

Libraries:

- aioquic (written in: Python; license: New BSD License)
<https://github.com/aiortc/aioquic>
- LiteSpeed QUIC (written in: C; license: MIT License)
<https://github.com/litespeedtech/lsquic>
- MsQuic (written in: C; license: MIT License)
<https://github.com/microsoft/msquic>
- Proxygen (written in: C++; license: New BSD License)
<https://github.com/facebook/proxygen>
- quiche (written in: Rust; license: Simplified BSD License)
<https://docs.quic.tech/quiche/> <https://github.com/cloudflare/quiche>
- QUICHE (written in: C++; license: New BSD License)
<https://quiche.google.com/quiche>
<https://github.com/google/quiche>

Implementations (3)

Servers:

- Caddy (written in: Go; license: Apache License 2.0)
<https://caddyserver.com/> <https://github.com/caddyserver/caddy>
- OpenLiteSpeed (written in: C++; license: GPLv3)
<https://openlitespeed.org/>
<https://github.com/litespeedtech/openlitespeed>
- NGINX (written in: C; licenc: Simplified BSD License)
<https://nginx.org/>
 - See: *Introducing a Technology Preview of NGINX Support for QUIC and HTTP/3*. June 10, 2020.
<https://www.nginx.com/blog/introducing-technology-preview-nginx-support-for-quic-http-3/>
- Internet Information Services (written in: C++; license: non-free)
<https://www.iis.net/>

Implementations (4)

Browsers:

- Chromium, Google Chrome:
 - See:
 - `chrome://flags/#enable-quick`
 - *Chrome is deploying HTTP/3 and IETF QUIC*. October 7, 2020.
<https://blog.chromium.org/2020/10/chrome-is-deploying-http3-and-ietf-quick.html>
- Chromium-based Edge:
 - See: `edge://flags/#enable-quick`
- Opera:
 - See: `opera://flags/#enable-quick`
- Firefox:
 - See the `network.http.http3.enabled` option (`about:config`).

See: <https://caniuse.com/http3>

Implementations (5)

Other clients:

- curl (written in: C; license: X11 License) <https://curl.se/>
<https://github.com/curl/curl>

curl (1)

- Experimental HTTP/3 support was introduced in version 7.66.0 released in September 2019.
- See: Daniel Stenberg. *curl 7.66.0 – the parallel HTTP/3 future is here*. August 5, 2019. <https://daniel.haxx.se/blog/2019/09/11/curl-7-66-0-the-parallel-http-3-future-is-here/>

curl (2)

- To have HTTP/3 support curl must be built from source that requires a QUIC library.
 - A few QUIC libraries are supported from which users can choose, such as [ngtcp2](#) or [quiche](#).
- Building curl with HTTP3 support: *HTTP3 (and QUIC)*
<https://curl.se/docs/http3.html>

curl (2)

Example of use:

```
curl --http3 https://www.google.com/ -v
```

URIs

HTTP3 uses `https` URIs.

Comparison of HTTP/2 and HTTP/3 (1)

Similarities: both protocols offer the same set of features to clients.

- Streams
- Server push
- Field compression
- ...

Comparison of HTTP/2 and HTTP/3 (2)

Differences:

- In principle, HTTP/2 can be implemented without TLS, while HTTP/3 can not.
- HTTP/3 permits the use of a larger number of streams (i.e., $2^{62} - 1$) than HTTP/2 (i.e., $2^{31} - 1$).
- Several HTTP/2 frame types (i.e., PING, RST_STREAM, WINDOW_UPDATE) are not required in HTTP/3, because QUIC itself provides equivalent functionality.
 - Also, CONTINUATION and PRIORITY frames do not exist in HTTP/3.
- HTTP/3 uses a modified version of HPACK, called QPACK.

Comparison of HTTP/2 and HTTP/3 (2)

Performance:

- Sreeni Tellakula. *Comparing HTTP/3 vs. HTTP/2 Performance*. 14 April, 2020. <https://blog.cloudflare.com/http-3-vs-http-2/>

Further Recommended Reading

- Daniel Stenberg. *HTTP/3 Explained*.
<https://http3-explained.haxx.se/>
<https://github.com/bagder/http3-explained>
- Robin Marx. *HTTP/3 From A To Z: Core Concepts*. Aug 9, 2021.
<https://www.smashingmagazine.com/2021/08/http3-core-concepts-part1/>
- Robin Marx. *HTTP/3: Performance Improvements (Part 2)*. Aug 23, 2021. <https://www.smashingmagazine.com/2021/08/http3-performance-improvements-part2/>
- Robin Marx. *HTTP/3: Practical Deployment Options (Part 3)*. Sep 6, 2021. <https://www.smashingmagazine.com/2021/09/http3-practical-deployment-options-part3/>