RFC 9250: DNS-over-QUIC (DoQ)

Sara Dickinson
Christian Huitema
Allison Mankin

DNS-over-QUIC

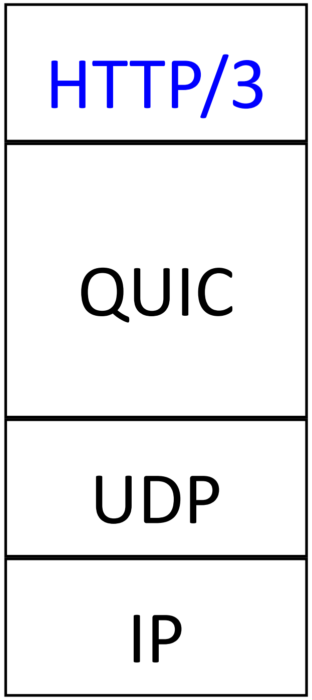
- Why did we standardize ANOTHER protocol for encrypted DNS?
- How is DoQ different to DoT/DoH?
- Where are we with implementation and deployment of DoQ?

QUIC - Background

- QUIC and HTTP/3 developed by Google as experiment in 2012
- Development moved to IETF in 2015, standardized in 2021(RFC 9000)
- Deployed by browsers and CDNs (7.6% websites)

QUIC - Background

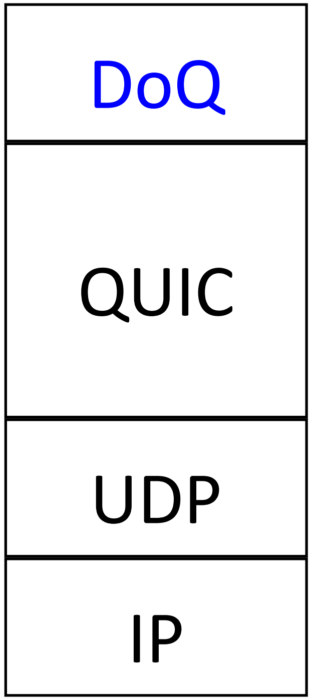
- QUIC and HTTP/3 developed by Google as experiment in 2012
- Development moved to IETF in 2015, standardized in 2021
- Deployed by browsers and CDNs (7.6% websites)
- **Key QUIC characteristics**
  - TLS 1.3 secured transport that runs over UDP
  - Reduced latency in handshake (0-RTT)
  - Stream based multiplexing - no head of line blocking
  - Improved error detection and loss recovery compared to TCP
  - Connection migration (IP address can change)
- HTTP/3 runs over QUIC



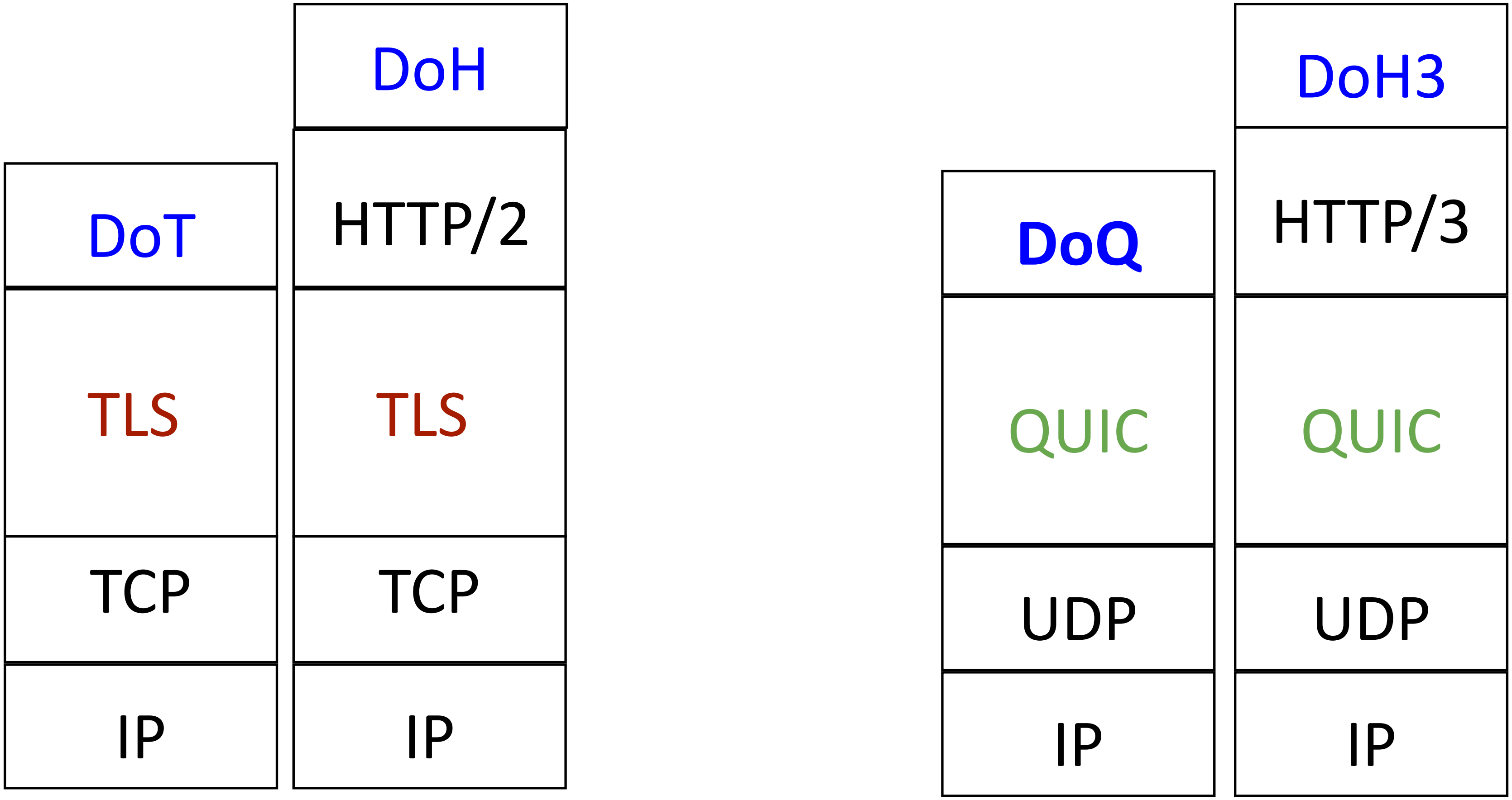
DoQ - Background

- Early realisation that DoQ would be a good fit for encrypted DNS
  - Low latency
  - UDP but with QUIC benefits and
    - Source address validation
    - Path MTU does not limit size of messages

- But… DoQ held up by QUIC standardization which took until last year



DoQ vs DoT vs DoH(3)?



DoT
TLS
TCP
IP

DoH
HTTP/2
TLS
TCP
IP

DoQ
QUIC
UDP
IP

DoH3
HTTP/3
QUIC
UDP
IP

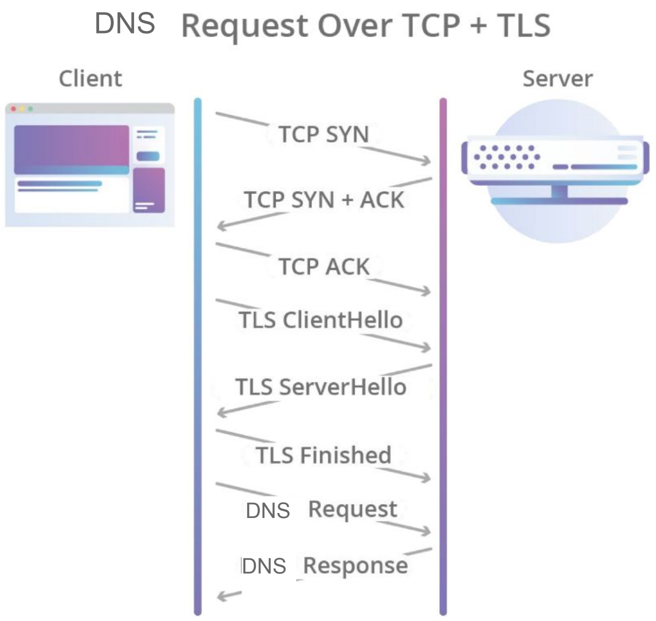
DoQ - Background

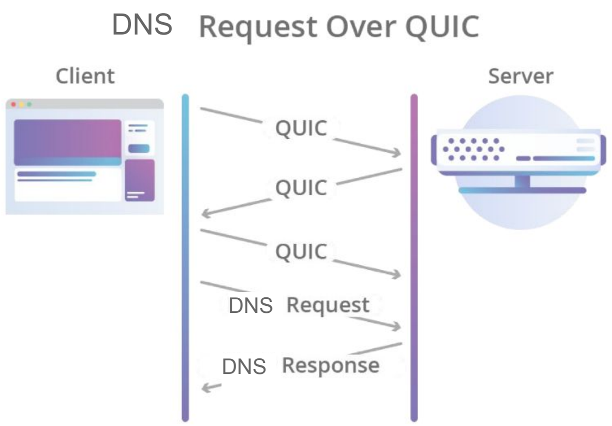
- **April 2017** - First Draft in QUIC WG
- **December 2018** - Adguard DoQ service launched

- **Apr 2020** - Draft adopted in DPRIVE WG (stub to rec ONLY)
- **Jul 2021** - **draft-ietf-dprive-dnsoquic-03 had big CHANGES**!
  - Re-scoped to include XFR and rec to auth
  - Mapping updated (will find pre and post change implementations)
  - Port 853 requested (more later)
- **Oct 2021-** Start of "Last Call" reviews
- **May 2022** - approved for publication

What does DoQ handshake look like?

- Set up a connection with a QUIC **handshake** (TLS 1.3)
- Uses ALPN ‘doq’



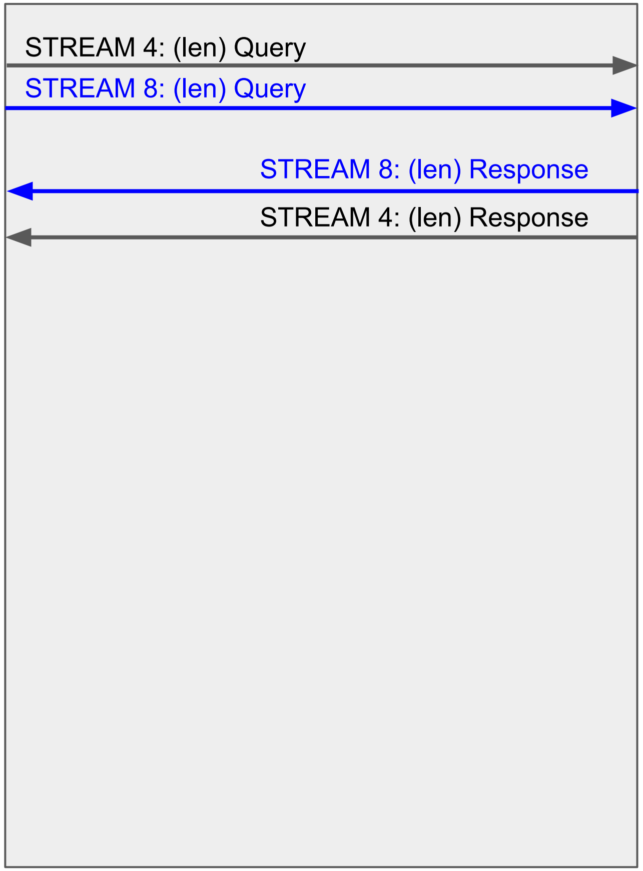


Images from https://blog.cloudflare.com/the-road-to-quic/

What does DoQ connection look like?

- Exchange of messages on streams (IDs are 4, 8,12)
- 1 stream is used for a single DNS query/response transaction (then closed)
- There are 2^{64} stream IDs - that's a lot of messages on one connection
    - MessageID is ALWAYS 0

Single QUIC connection

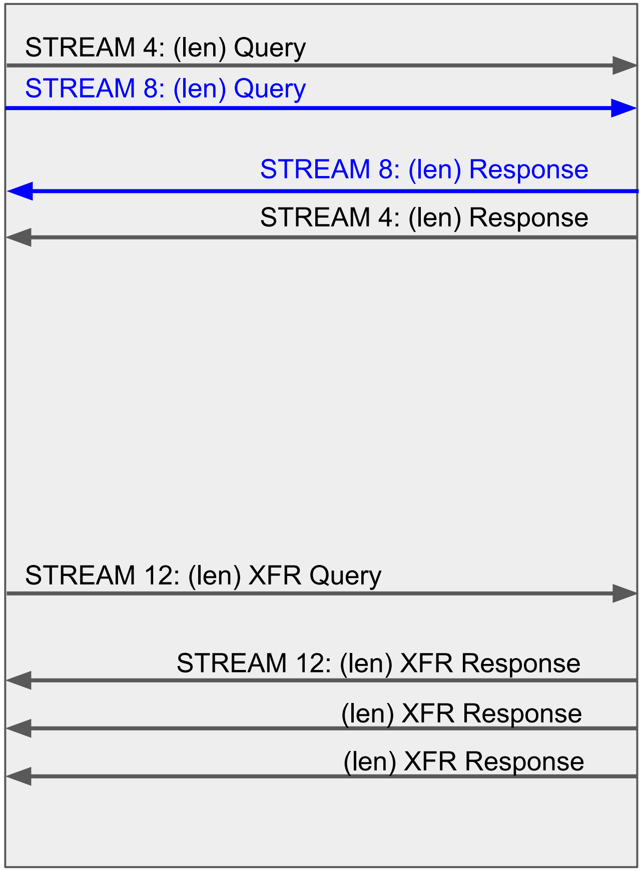


What does DoQ connection look like?

- Exchange of messages on streams (IDs are 4, 8,12)
- 1 stream is used for a single DNS query/response transaction (then closed)
- There are 2^{64} stream IDs - that's a lot of messages on one connection
  - MessageID is ALWAYS 0

- **Mapping (post -03)**
  - Prepend with length field (like TCP)
  - Server can send multiple responses

Single QUIC connection



DoQ is a general purpose protocol

- RFC 9250 describes 3 scenarios
  - **Stub-Recursive**: AdGuard claim **good performance** (used in mobile networks)
  - **Recursive-Auth**: More attractive than DoT/DoH/DoH3
  - **XFR:** RFC 9103 - XFR-over-TLS published in 2021

DoQ is a general purpose protocol

- RFC 9250 describes 3 scenarios
  - **Stub-Recursive**: AdGuard claim **good performance** (used in mobile networks)
  - **Recursive-Auth**: More attractive than DoT/DoH/DoH3
  - **XFR:** RFC 9103 - XFR-over-TLS published in 2021

- After some debate DoQ will use port 853 (assigned to DNS over DTLS in 2016).
  - TCP port 853: DNS over TLS
  - **UDP port 853: DNS over DTLS or QUIC**

    (QUIC v1 is designed to demux with DTLS)

DoQ Implementations (open source)

Implementation	Language	Notes
CoreDNS	Go	AdGuard use as DoQ server
AdGuard C++ DNS libs	C++	AdGuard use in mobile app
AdGuard DNS Proxy	Go	Simple proxy or server supporting DoQ (used in ADGuard Home)
dnslookup	Go	Command line utility wrapper for Adguard DNS proxy
Quicdoc	C	Simple DoQ impl based on Picoquic
aioquic	Python	QUIC implementation includes example DoQ client/server
Flamethrower	C++	DNS performance utility with experimental DoQ

- No implementations yet in the major OS recursive resolvers or authoritatives

DoQ Deployments

Recursive resolvers

Deployment	Notes
AdGuard	Running for 3+ years now in 10 countries
nextDNS.io	~200 globally distributed instances
Total of 1200 DoQ resolvers in Jan 2022	As measured in “One to Rule them All? A First Look at DNS over QUIC” https://arxiv.org/abs/2202.02987

Recursive to Authoritative

- Interest in recursive to auth experiments using ‘unilateral probing’ (draft-ietf-dprive-unilateral-probing)

DoQ ongoing work

- Padding
  - More work needed to develop current experimental models for message padding

- Some implementation issues observed in the wild - performance can be improved to reach 0-RTT

- Lacking formal performance measurements - particularly for recursive to authoritative traffic patterns

Summary

- DNS-over-QUIC is now an IETF standard (RFC 9250)
- Several stub-recursive DoQ deployments
- Likely candidate for recursive to auth experiments using probing

https://dnsprivacy.org