Client Authentication Recommendations for Encrypted DNS (CARED)

https://datatracker.ietf.org/doc/draft-tjjk-cared/

Tommy Jensen (Microsoft)
Jessica Krynitsky (Microsoft)
Jeff Damick (Amazon)
Matt Engskow (Amazon)

Context

- Enterprises are increasing encrypted DNS deployments
- Applying client policy often relies on the client's IP address
- Enterprises want to only allow their own clients to connect
- Addressing both when clients can be work-from-home requires client authentication (or per-client tunnel gateways…)

Why a draft?

- To maximize interop between DNS implementations by recommending best practices for authentication mechanisms

- To codify when client auth with encrypted DNS is appropriate so implementations can avoid regressing user privacy in use cases that do not justify client authentication

Draft in a nutshell

- Using client auth with encrypted DNS is restricted to when…
  - The server gates access/behavior to a specific set of clients AND
  - The clients are pre-configured by their admin to auth to this server
  - [optionally] The server needs to resolve names differently per client

- All other DNS use cases should not use client auth
  - Different general server behavior can be exposed as different endpoints
    - Ex: ad-blocking, malware filtering, adult content filtering
  - Servers cannot expect clients they have no out-of-band relationship with to present auth when challenged

Draft in a nutshell

- Requirements considered when evaluating client auth mechanisms:
  - SHOULD be per-connection, not per-query
  - SHOULD use existing open standards
  - SHOULD be reusable across encrypted DNS protocols
  - SHOULD NOT require human interaction to complete

- Rationale: avoid vendor lock-in, optimize for long-running connections, solve the problem once for DoT, DoH, and DoQ, avoid the “click through” effect

Draft in a nutshell

- The draft recommends mTLS as a best practice for encrypted DNS stub and recursive resolvers to implement for interop

- The draft enumerates why and compares against other client auth mechanisms:
  - JWTs: per request, DoH specific
  - HTTP auth: ~~just no~~ DoH specific,
  - FIDO: human interaction required, heavier lift for servers
  - New solution: ~~just no~~ slower adoption, no need when solutions exist

Next steps

- Seeking guidance on appropriate WG for adoption
- If DNSOP, then asking for adoption-blocking feedback
- Feedback welcome: https://github.com/mstojens/draft-tjjk-cared