Network Working Group
Internet-Draft
Intended status: Informational
Expires: 16 October 2025

T. Jensen
J. Krynitsky
Microsoft
J. Damick
M. Engskow
Amazon
J. Abley
Cloudflare

14 April 2025

# Client Authentication Recommendations for Encrypted DNS
## draft-jaked-cared-01

## Abstract

Some encrypted DNS clients require anonymity from their encrypted DNS servers to prevent third parties from correlating client DNS queries with other data for surveillance or data mining purposes. However, there are cases where the client and server have a pre-existing relationship and each wants to prove its identity to the other. For example, an encrypted DNS server may only wish to accept queries from encrypted DNS clients that are managed by the same enterprise, and an encrypted DNS client may need to confirm the identity of the encrypted DNS server it is communicating with. This requires mutual authentication.

This document discusses the circumstances under which client authentication is appropriate to use with encrypted DNS, the benefits and limitations of doing so, and recommends authentication mechanisms to be used when communicating with TLS-based encrypted DNS protocols.

Table of Contents

## 1. Introduction

There are times when a client needs to authenticate itself before it can be authorised to use a DNS server. One example is when an encrypted DNS server only accepts connections from pre-approved clients, such as an encrypted DNS service provided by an enterprise for its remote employees. Encrypted DNS clients trying to connect to such a server might experience refused connections if they did not provide authentication that allows the enterprise's server to authorise its use.

This is different from general use of encrypted DNS by anonymous clients to public DNS resolvers, where it is bad practice for the client to provide any kind of identifying information to the server. For example, Section 8.2 of [RFC8484] discourages use of HTTP cookies with DNS-over-HTTPS (DoH). This ensures that clients provide a minimal amount of identifiable or correlatable information to servers that do not need to know anything about the client in order to provide name resolution.

Because of the significant difference in these scenarios, it is important to define the situations in which interoperable encrypted DNS clients can use client authentication without compromising the privacy provided by encrypted DNS in the first place. Even then, it is important to recognize what value client authentication provides to encrypted DNS clients versus encrypted DNS servers in the context of both connection management and DNS resolution utility. This document discusses these topics and recommends best practice for authentication.

## 2. Conventions and Definitions

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

## 3. Terminology

Encrypted DNS server: a recursive DNS resolver that implements one or more encrypted DNS protocols, such as those defined in [RFC9499] (DNS-over-TLS, DNS-over-HTTPS, DNS-over-QUIC).

Protective DNS server: a policy-implementing, recursive DNS resolver service that filters DNS queries to prevent resolution for known malicious domains and/or IP addresses. Protective DNS servers support DNS technologies including encrypted DNS protocol support (DoH/DoT) and IPv6 resolution.

Other terminology related to the DNS is used in this document as defined in [RFC9499].

## 4. Benefits of client authentication with encrypted DNS

Strong identification of encrypted DNS clients by the encrypted DNS server allows the DNS server to apply client-specific resolution policies in a securely-verifiable way. Today, a common practice is to establish client-specific resolution policies that are applied on behalf of particular clients based on their observed IP address. This is not only an insecure method that cannot account for clients sending requests through middleboxes, but it can only be used when expected IP addresses are known in advance. This is not practical for enterprises with remote employees without introducing a dependency on tunneling DNS traffic to a managed gateway or proxy whose IP address is known. This in turn forces enterprises to choose between running proxy or gateway infrastructure per client (or at least per-client IP address mappings) or losing client identification.

Strong identification of encrypted DNS clients by the encrypted DNS server also brings identification up to the application layer, which insulates the mechanism used for identity management from network topology changes and allows the sense of client identity in the server to persist despite client IP address changes.

## 5. Drawbacks of client authentication with encrypted DNS

While there are benefits in using client authentication with encrypted DNS in limited circumstances, authentication has drawbacks that make it inappropriate in many other cases. For example, client authentication is generally considered to be bad practice in uses of encrypted DNS that are motivated by client anonyminity, since it allows a full history of resolution requests to be associated confidently with the identity of an individual client. This is unacceptable practice.

Public DNS servers generally implement allowlists and blocklists at the IP layer as part of a layered defence against abuse; IP-layer access control allows unwanted traffic to be discarded early and helps mitigate unwanted server load. Public encrypted DNS servers with the additional requirement to permit authenticated clients makes

it impossible to drop unwanted traffic early based on source IP address, which increases the cost of mitigation and adds complexity that may introduce additional attack vectors.

## 6. When to use client authentication with encrypted DNS

Encrypted DNS servers that provide resolution using transport protocols that incorporate TLS or dTLS such as DoH [RFC8484], DoT [RFC7858], and DoQ [RFC9250] SHOULD NOT provide client authentication except in the limited situations described in this section.

Encrypted DNS servers that use other transport protocols are not in scope for this document.

### 6.1. When servers should require client authentication

Encrypted DNS servers MUST NOT challenge clients for authentication unless they need to restrict connections to a set of clients they have a pre-existing relationship with as defined in {restrict-clients}, regardless of whether or not the requirements in {per-client} apply. Encrypted DNS servers also MUST NOT challenge clients for authentication when it knows its identity cannot be securely verified. Examples of this include using self-signed certificates or making itself discoverable using DDR’s Discovery Using Resolver IP Addresses mechanism described in Section 4 of [RFC9462].

Encrypted DNS servers that meet the requirements in {restrict-clients} MAY challenge clients to authenticate to avoid achieving the same goal of identifying clients through other, less secure means (such as IP address or data in the DNS query payload).

### 6.2. Restricting connections to allowed clients

An encrypted DNS server that provides resolution to a specific set of clients and refuses service to all other clients MAY require clients to authenticate.

For example, an encrypted DNS server owned by an enterprise that only allows connections from devices managed by that same enterprise might require clients to authenticate.

Note that this does not apply to scenarios where the owner of the encrypted DNS server and the device using the encrypted DNS client, such as an ISP and its customers. While this is a scenario where the ISP may wish to restrict access to its encrypted DNS resolver to only its customers, client authentication of DNS traffic is not necessary to achieve that.

### 6.3. Resolving names differently per client

An encrypted DNS server that provides client-specific resolution behaviour MAY require clients to authenticate in circumstances where the appropriate policy for particular clients is specified by the operator of the encrypted DNS server.

For example, an encrypted DNS server that is configured to allow some clients to resolve certain names while other clients are not allowed needs to identify the client so that the resolution behaviour for those names can be implemented accurately.

Encrypted DNS servers SHOULD NOT attempt to authenticate clients to identify the appropriate resolution policy to use when the difference in resolution behavior between them is not imposed by the operator of the server but instead is chosen by the client.

For example, some public encrypted DNS services provide clients with the option of blocking resolution requests for particular categories of domain names, such as names associated with malware distribution, adult content or advertisers, and makes servers with particular resolution policies available on different IP addresses. A client who wishes to opt-in to a server with a particular resolution policy can do so by sending queries to the corresponding IP address, and no client identification is required.

### 6.4. When clients should attempt to authenticate

An encrypted DNS client MUST NOT offer authentication to any encrypted DNS server unless it was specifically configured to expect that server to require authentication, independent of the mechanism by which the client chose or discovered the encrypted DNS server to use. In other words, encrypted DNS client authentication MUST require administrator action typical of enterprise managed devices rather than being out of the box default behavior for an application or operating system.

An encrypted DNS client MUST NOT offer authentication when the server connection was established using Section 4 of [RFC9462], even if the IP address of the original DNS server was specifically configured for the encrypted DNS client as one that might require authentication. This is because in that circumstance there is not a pre-existing relationship with the encrypted DNS server (or else DDR bootstrapping into encrypted DNS would not have been necessary).

An encrypted DNS client MAY choose to present authentication to a server that requests it, but is not required to do so; for example, a client MAY choose instead not to use the server. If a client does

present an identity to a server, the identity SHOULD be unique to that server, or unique to servers provided by the same provider when the client has that information, to reduce the risk of a client's resolution history on multiple colluding providers being correlated.

## 7. Recommendations for authentication mechanisms

This section enumerates recommended considerations for selecting a client authentication mechanism and recommends solutions to guide implementors who wish to maximize chances of interoperability wherever other implementations hold the same considerations.

The following requirements were considered when formulating the recommended authentication mechanism for encrypted DNS clients. it is RECOMMENDED that authentication mechanisms:

1. be per-connection, not per-query, e.g. to avoid unnecessary payload overheads
2. use open, standard mechanisms where possible, e.g. to avoid vendor lock-in and specialized cryptography
3. be reusable across multiple encrypted DNS protocols, e.g. to avoid protocol preference
4. not require human user interaction to complete authentication
5. be resistant to replay attacks

This document concludes that the best current mechanism available to recommend for enabling interoperable encrypted DNS client authentication is mTLS [RFC8705] for the following reasons:

1. mTLS identifies and authenticates clients, not users, per-connection
2. mTLS is an existing standard and is often readily available for TLS clients
3. X.509 certificates used for TLS client authentication allow a server to include other attributes in an authentication decision, such as the client's organization via PKI hieracrchy
4. mTLS is reusable across multiple encrypted DNS protocols
5. mTLS allows session resumption [RFC8446]

6. mTLS does not require user interaction or apaplication layer input for authentication

Encrypted DNS clients and servers that support offering or requesting client authentication SHOULD support at least the use of mTLS with TLS 1.3 in addition to whatever other mechanism they wish to support. Client authentication using PKI certificates is RECOMMENDED, but pre-shared keys MAY also be used to meet the requirements listed above for recommendable practices, provided (EC)DHE key exchange is used to maintain perfect forward secrecy [RFC8446]. Versions of TLS lower than 1.3 lack some security features that new protocols (as of this writing) are taking for granted or making recommended behavior. When TLS 1.3 is discouraged in favor of future versions of TLS or its future replacement, that guidance supercedes this paragraph.

Encrypted DNS clients and servers SHOULD prefer long-lived connections when using client authentication to minimize the cost over time of doing repeated TLS handshakes and identity verification.

## 7.1. Why these requirements were chosen

### 7.1.1. Per-connection scope

Any data added to each DNS message will have greater bandwidth and processing costs than presenting authentication once per connection. This is especially true in this case because the drawback of having long-running encrypted DNS connections is the decreased privacy through increased volume of directly correlatable queries. However, this privacy threat does not apply to this situation because the client’s queries can already be correlated by the identity it presents. Therefore, per-connection bandwidth and data processing overhead is expected to be much lower than per-query because there is no incentive for clients and servers to not have long-running encrypted DNS connections.

This is not expected to create excessive cost for server operators because supporting encrypted DNS without client authentication already requires per-connection state management.

### 7.1.2. Reusable open standards

Reusing open standards ensures wide interoperability between vendors that choose to implement client authentication in their encrypted DNS stacks.

#### 7.1.3. Reusable across protocols

If a client authentication method for encrypted DNS were defined or recommended that would only be usable by some TLS-encrypted DNS protocols, it would encourage the development of a second or even third solution later.

#### 7.1.4. Does not require human interaction

Humans using devices that use encrypted DNS, when given any kind of prompt or login relating to establishing encrypted DNS connectivity, are unlikely to understand what is happening and why. This will inevitably lead to click-through behavior. Because the scope of scenarios where client authentication for encrypted DNS is limited to pre-existing relationships between the client and server, there should be no need for at-run-time intervention by a human user.

#### 7.1.5. Resistance to replay attacks

Today, there are some attempts to identify clients that involve use of client- specific DoH templates or DoT hostnames, addition of magic strings to requests, and other mechanisms to enable proprietary experiences. It is important that recommendations for client authentication are restricted to mechanisms that protect their secrets or keys from replay attacks or compromise by unprivileged processes.

### 7.2. Why alternatives are not recommended

#### 7.2.1. Web tokens

OAuth or JSON web tokens alone require HTTP to validate, so would not be a solution for encrypted DNS protocols other than DoH. Web access tokens can be used as certificate-bound access tokens in combination with mTLS if they are needed to prove identity with another authorization server, as described in [RFC8705].

#### 7.2.2. HTTP authentication

HTTP authentication as defined in [RFC9110] provides a basic authentication scheme for the HTTP protocol. Unless it is used with TLS, i.e. over HTTPS, the credentials are encoded but not encrypted which is insecure. As TLS is already used by the encrypted DNS protocols in this document’s scope, it is simpler to handle client authentication and authorization at the TLS layer. Additionally, mTLS is more broadly-adopted than HTTP authentication. HTTP authentication would only be a viable option for DoH, and not extensible to other encrypted DNS solutions.

#### 7.2.3. FIDO

Web Authentication (WebAuthN) and the FIDO2 Client to Authenticator Protocol (CTAP) use CBOR Object Signing and Encryption (COSE), described in [RFC8812]. FIDO and WebAuthN are passkey solutions designed to replace passwords for user authentication for online services, and they are not typically used for general client authentication. Passkeys are unique for each online service and require user input for registration, and would require DNS servers to support the WebAuthN protocol. Additionally, each sign-in requires user input for local verification, using biometric, local PIN, or a FIDO security key.

#### 7.2.4. Designing a novel solution

Designing a novel solution is never recommended when there is an existing standard that meets the requirements. Doing so would make the encrypted DNS solution more difficult and time-consuming to adopt, and most likely would introduce vendor lock-in.

## 8. Operational Considerations

### 8.1. Avoiding connectivity deadlocks

When deploying encrypted DNS Clients, careful consideration should be made how certificate rollover and revocation will happen. If an encrypted DNS server only allows connections from clients with valid certificates, and the client is configured to only use the encrypted DNS server, then there will be a deadlock when the certificate expires or is revoked such that the client device will not have the connectivity needed to renew or replace its certificate.

Encrypted DNS servers that challenge clients for authentication SHOULD have a separate resolution policy for clients that do not have valid credentials that allows them to resolve the subset of names needed to connect to the infrastructure needed to acquire certificates.

Alternatively, encrypted DNS clients that are configured to use encrypted DNS servers that will require authentication MAY consider configuring knowledge of certificate issuing infrastructure in advance so that the DNS deadlock can be avoided without introducing less secure DNS servers to their configuration (i.e. hard coding IP addresses and host names for certificate checking).

## 9. Security Considerations

This document describes when and how encrypted DNS clients can authenticate themselves to an encrypted DNS server. It does not introduce any new security considerations beyond those of TLS and mTLS. This document does not define recommendations for when and how to use encrypted DNS client authentication for encrypted DNS protocols that are not based on TLS or dTLS.

### 9.1. Custom Deployments

This document was written to provide guidance to implementors interoperating with third party implementations to maximize the chances of out-of-the-box compatibility. There are certainly many ways of accomplishing client authentication with encrypted DNS not listed as recommended practice here. Implementors are encouraged to use the reasoning in this document explaining its choice in recommendations, but not following this document’s recommendations does not imply any violation of protocol compliance for encrypted DNS protocols or whatever authentication mechanism the implementor selects.

### 9.2. Use of DNS for Indicators of Compromise

DNS queries can sometimes act as a source of Indicators of Compromise [RFC9424], further placing value on strong client authentication when it is appropriate as discussed in earlier sections.

Deployers should weigh the recommendations and reasoning in this document against their threat models to ensure their Protective DNS deployments provide useful Indicators of Compromise in addition to the need for interoperability.

## 10. IANA Considerations

This document has no IANA actions.

## 11. References

### 11.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/rfc/rfc2119>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/rfc/rfc8174>.

[RFC9424] Paine, K., Whitehouse, O., Sellwood, J., and A. Shaw, "Indicators of Compromise (IoCs) and Their Role in Attack Defence", RFC 9424, DOI 10.17487/RFC9424, August 2023, <https://www.rfc-editor.org/rfc/rfc9424>.

[RFC9499] Hoffman, P. and K. Fujiwara, "DNS Terminology", BCP 219, RFC 9499, DOI 10.17487/RFC9499, March 2024, <https://www.rfc-editor.org/rfc/rfc9499>.

### 11.2. Informative References

[RFC7858] Hu, Z., Zhu, L., Heidemann, J., Mankin, A., Wessels, D., and P. Hoffman, "Specification for DNS over Transport Layer Security (TLS)", RFC 7858, DOI 10.17487/RFC7858, May 2016, <https://www.rfc-editor.org/rfc/rfc7858>.

[RFC8446] Rescorla, E., "The Transport Layer Security (TLS) Protocol Version 1.3", RFC 8446, DOI 10.17487/RFC8446, August 2018, <https://www.rfc-editor.org/rfc/rfc8446>.

[RFC8484] Hoffman, P. and P. McManus, "DNS Queries over HTTPS (DoH)", RFC 8484, DOI 10.17487/RFC8484, October 2018, <https://www.rfc-editor.org/rfc/rfc8484>.

[RFC8705] Campbell, B., Bradley, J., Sakimura, N., and T. Lodderstedt, "OAuth 2.0 Mutual-TLS Client Authentication and Certificate-Bound Access Tokens", RFC 8705, DOI 10.17487/RFC8705, February 2020, <https://www.rfc-editor.org/rfc/rfc8705>.

[RFC8812] Jones, M., "CBOR Object Signing and Encryption (COSE) and JSON Object Signing and Encryption (JOSE) Registrations for Web Authentication (WebAuthn) Algorithms", RFC 8812, DOI 10.17487/RFC8812, August 2020, <https://www.rfc-editor.org/rfc/rfc8812>.

[RFC9110] Fielding, R., Ed., Nottingham, M., Ed., and J. Reschke, Ed., "HTTP Semantics", STD 97, RFC 9110, DOI 10.17487/RFC9110, June 2022, <https://www.rfc-editor.org/rfc/rfc9110>.

[RFC9250] Huitema, C., Dickinson, S., and A. Mankin, "DNS over Dedicated QUIC Connections", RFC 9250, DOI 10.17487/RFC9250, May 2022, <https://www.rfc-editor.org/rfc/rfc9250>.

[RFC9462] Pauly, T., Kinnear, E., Wood, C. A., McManus, P., and T. Jensen, "Discovery of Designated Resolvers", RFC 9462, DOI 10.17487/RFC9462, November 2023, <https://www.rfc-editor.org/rfc/rfc9462>.

## Acknowledgments

The authors are grateful to the following individuals for their feedback on or contributions to the text, with no implication of endorsement: Andrew S2, Ben Schwartz, Erik Nygren, Jim Reid, Q Misell, Tim Wicinski, Tobias Fiebig, Warren Kumari, Wes Hardaker.

## Authors' Addresses

Tommy Jensen
Microsoft
Email: tojens.ietf@gmail.com

Jessica Krynitsky
Microsoft
Email: jess.krynitsky@microsoft.com

Jeffrey Damick
Amazon
Email: jdamick@amazon.com

Matt Engskow
Amazon
Email: mengskow@amazon.com

Joe Abley
Cloudflare
Email: jabley@cloudflare.com