

Network Working Group
Internet-Draft
Intended status: Informational
Expires: January 3, 2019

M. Nottingham
July 02, 2018

DOH Digests
draft-nottingham-doh-digests-00

Abstract

The lack of flexible configuration and selection mechanisms for DOH servers is identified as suboptimal for privacy and performance in some applications.

This document makes a straw-man proposal for an improvement.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <https://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on January 3, 2019.

Copyright Notice

Copyright (c) 2018 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	2
1.1. DOH's Additional Benefits for Associated Services	2
1.2. Achieving DOH's Privacy Goals through Diversity	3
2. Conventions and Definitions	4
3. DOH Digests	4
3.1. Using DOH Digests	4
3.2. The DOH Digest Format	5
3.3. Hostname Normalisation	5
4. Security Considerations	5
5. IANA Considerations	5
6. References	5
6.1. Normative References	5
6.2. Informative References	6
Author's Address	6

1. Introduction

One of the core motivations for DOH [I-D.ietf-doh-dns-over-https] is to improve end-user privacy by obfuscating the stream of DNS requests that the DOH client makes. It does this by mixing DOH requests into a stream of "normal" HTTP requests to a configured Web server; for example, a large Web site or a Content Delivery Network.

However, DOH intentionally avoids defining a mechanism for configuring a particular DOH server for a given application or host. So far, the most common way to do so is to select one from a pre-configured list of services in an application, such as a Web browser.

Typically, the list of available DOH services is vetted by the application's vendor to assure that they will honour the application's requirements for handling of sensitive data (i.e., the client's DNS request stream) and similar concerns.

This document proposes a means of selecting a DOH server that encourages the deployment of DOH servers by sharing some of its additional benefits with servers that are good candidates for serving DOH traffic.

1.1. DOH's Additional Benefits for Associated Services

When a DOH server is colocated with (or closely coordinated with) other network services - especially HTTP services - those associated services enjoy a few additional benefits beyond those seen by adopting DOH in the first place.

- o Associated services have an additional privacy benefit; there is one less party involved in the interaction, whereas "normal" DNS and DOH to an unassociated HTTP server require a third party to resolve names.
- o Removing a third party also removes a separate point of potential failure, improving control over service quality and availability. See [fragile] for further discussion.
- o Finally, the DOH server can use DNS to optimise the provision of associated services. For example, DNS results can be optimised based on the client's request stream with a higher degree of certainty.

In the future, a DOH server might use Secondary Certificates [I-D.ietf-httpbis-http2-secondary-certs] to further optimise performance of associated services, by using the information in the DNS request stream to aggregate all of its traffic into a small number of connections (possibly only one), thereby allowing greater coordination of congestion control and avoiding connection setup costs.

1.2. Achieving DOH's Privacy Goals through Diversity

Overall, a major goal for deployment of DOH is to assure that DNS connectivity is robust and private. Arguably, this is best served by having a diverse set of available DOH servers that are colocated with popular HTTP content, so that it's more difficult to discriminate DOH from "regular" HTTP, and so the it's more difficult to block DOH services, due to the high impact of blocking a popular site.

One way to encourage the development of such a set is to offer the additional benefits above to parties that are good candidates for serving such traffic. When clients can direct their DOH queries to the HTTP server which will eventually serve their traffic, it provides both better privacy properties and better performance and availability to a broader set of servers.

This is a marked improvement over the static configuration mechanism commonly in place now; accruing such privacy, availability, and performance benefits to whatever DOH server the application or user selects means that only parties who have a relationship with that service will realise these benefits.

This document proposes one way to achieve this.

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. DOH Digests

A DOH Digest is a Bloom filter indicating the set of hosts a given DOH server should be used for.

3.1. Using DOH Digests

When an application has a valid DOH digest for a given DOH server, it tests the digest for each DNS request it makes by hostname; if the hostname (after normalisation) is found in the digest, all DNS requests regarding that hostname SHOULD be sent to the corresponding DOH server. If multiple DOH digests match a given hostname, any matching DOH server MAY be used; the client SHOULD select one of the candidates randomly.

If the DOH service is unavailable, produces errors (HTTP or DNS), or the application otherwise fails to obtain an answer from it, the application MAY (but is not required to) fall back to using another configured DOH server, or to using "normal" DNS.

Likewise, hosts that do not match any configured bloom filter SHOULD be sent to a randomly selected DOH server that is available.

The means of discovering a DOH digest for a given DOH server is out of scope for this document, but generally it will be pre-arranged between the application and the DOH server.

The nature of this arrangement is highly dependent upon the application and its desired properties. That said, a number of requirements are placed upon this arrangement.

- o The digest MUST be conveyed in a manner that is secure and authenticated; e.g., TLS with appropriate certificate checks. Clients MUST enforce this.
- o The application MUST consider the DOH service as meeting whatever criteria it deems fit for configuring a "catch-all" DOH service (e.g., in terms of privacy, service availability, etc.), since false positives might be sent to the service, and hosts not matched by any configured bloom filter might be sent to it.

- o The digest **MUST** be updated on a periodic basis; e.g., once a day. Clients **SHOULD NOT** use stale digests.

3.2. The DOH Digest Format

TBD - likely just a bloom filter.

3.3. Hostname Normalisation

TBD

4. Security Considerations

Because a DOH digest allows a DOH server to claim traffic from an arbitrary hostname, applications need to take extreme care in selecting the DOH servers they will be accepted from, as well as assuring that their integrity and authentication have not been compromised.

Applications might mitigate this by monitoring DOH servers for such abuse and terminating their ability to use DOH digests when it is found.

TBD - more advanced mitigations

A hostname is effectively captured by a DOH server until the digest that reflects any change in its status is updated in the application. This delay should not result in any loss of functionality, since the "old" configuration will still direct requests to a functional DOH server.

5. IANA Considerations

This document currently has no IANA actions, but may grow some as the document progresses.

6. References

6.1. 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/info/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/info/rfc8174>>.

6.2. Informative References

- [fragile] Kashaf, A., Zarate, C., Wang, H., Agarwal, Y., and V. Sekar, "Oh, What a Fragile Web We Weave: Third-party Service Dependencies In Modern Webservices and Implications", June 2018, <<https://arxiv.org/pdf/1806.08420.pdf>>.
- [I-D.ietf-doh-dns-over-https] Hoffman, P. and P. McManus, "DNS Queries over HTTPS (DoH)", draft-ietf-doh-dns-over-https-12 (work in progress), June 2018.
- [I-D.ietf-httpbis-http2-secondary-certs] Bishop, M., Sullivan, N., and M. Thomson, "Secondary Certificate Authentication in HTTP/2", draft-ietf-httpbis-http2-secondary-certs-02 (work in progress), June 2018.

Author's Address

Mark Nottingham

Email: mnot@mnot.net

URI: <https://www.mnot.net/>

Internet Engineering Task Force
Internet-Draft
Intended status: Informational
Expires: January 3, 2019

T. Pusateri
Unaffiliated
W. Toorop
NLnet Labs
July 2, 2018

DHCPv6 Options for private DNS Discovery
draft-pusateri-dhc-dns-driu-00

Abstract

This draft provides a series of DHCPv6 options for a DHCPv6 client to request from a DHCPv6 server to aid in configuring DNS servers that support private requests/responses.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <https://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on January 3, 2019.

Copyright Notice

Copyright (c) 2018 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	2
1.1. Requirements Language	2
2. Trust	3
3. DHCPv6 Encapsulating Options	3
4. DHCPv6 DNS over TLS Encapsulated Options	5
4.1. IPv6 Address Option	5
4.2. ADN Option	6
4.3. Port Option	6
5. DHCPv6 DNS over DTLS Encapsulated Options	7
6. DHCPv6 DNS over HTTPS (DoH) Encapsulated Options	7
6.1. URI Option	7
7. Security Considerations	8
8. IANA Considerations	9
9. Acknowledgements	9
10. References	9
10.1. Normative References	9
10.2. Informative References	10
Appendix A. ISC DHCPv6 Configuration Example	11
A.1. ISC DHCPv6 Server Configuration	11
A.2. ISC DHCPv6 Client Configuration	11
Authors' Addresses	11

1. Introduction

There are three standardized forms for providing privacy to DNS including DNS over TLS (as defined in [RFC7858]), DNS over DTLS (as defined in [RFC8094]), and DNS over HTTPS (DoH) as defined in [I-D.ietf-doh-dns-over-https]. In order to use these encrypted forms of DNS securely, more information is needed by the client than the DNS Server list defined in Section 3 of [RFC3646]. This document defines three new DHCPv6 encapsulating options containing additional DHCPv6 options for clients to configure secure DNS in one of these forms. Each top level option specifies ONE server. Multiple servers are specified by including multiple instances of the same top level option with different encapsulated options.

1.1. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

2. Trust

Encrypting the DNS transport provides privacy of the information contained in the DNS requests/responses across the connection. It does not provide privacy at the endpoints of the connection. The private DNS configuration parameters obtained by a client via DHCPv6 are not automatically trusted. Trust is established in many ways or not at all. The environment a client finds itself in will determine how trustworthy the DHCPv6 reply may or may not be. There should be no false sense of privacy derived from the presence of these options in a DHCPv6 reply.

The following points may assist the DHCPv6 client:

1. clients can choose whether or not to use the DNS server configuration parameters they receive via DHCPv6 and may simply override these parameters with their own configuration.
2. DHCPv6 servers already provide unencrypted DNS server parameters to clients that are regularly used because the client has decided to trust the server reply in that environment.
3. client implementations (or operating system vendors) could establish whitelists (or blacklists) of known good (bad) servers.
4. the community could establish a registry of trusted DNS privacy servers.

3. DHCPv6 Encapsulating Options

Encrypted DNS DHCPv6 configuration parameters will be encapsulated in one or more of the following top level encapsulating options. These options can be repeated as many times as necessary to configure a list of secure DNS servers with one secure server per encapsulation. This is permitted by Section 22 of [RFC3315]. There is no order implied by the order of options sent or received. It is up to the receiving client to determine which order to use the DNS server configurations.

The format for the DNS over TLS [RFC7858] encapsulating option code is:

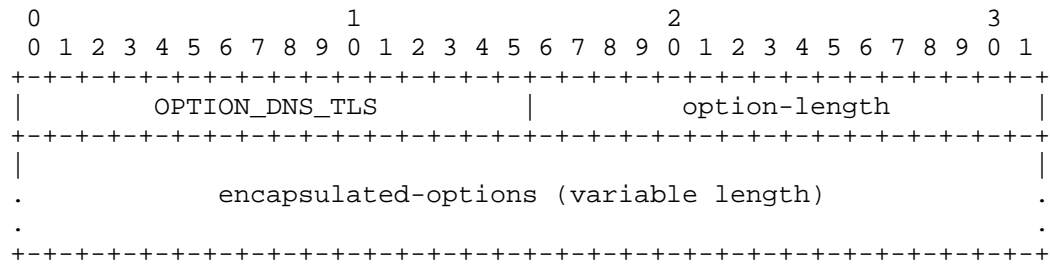


Figure 1: DNS over TLS option

option-code: OPTION_DNS_TLS (TBD)

option-len: Length of the sum of the lengths of the encapsulated options.

The format for the DNS over DTLS [RFC8094] encapsulating option code is:

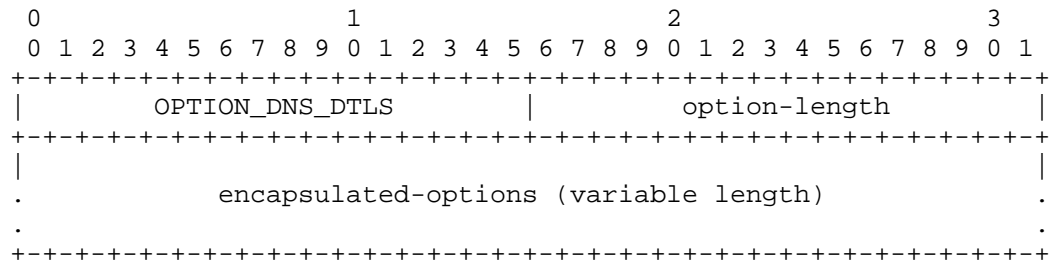


Figure 2: DNS over DTLS option

option-code: OPTION_DNS_DTLS (TBD)

option-len: Length of the sum of the lengths of the encapsulated options.

The format for the DNS over HTTPS [I-D.ietf-doh-dns-over-https] encapsulating option code is:

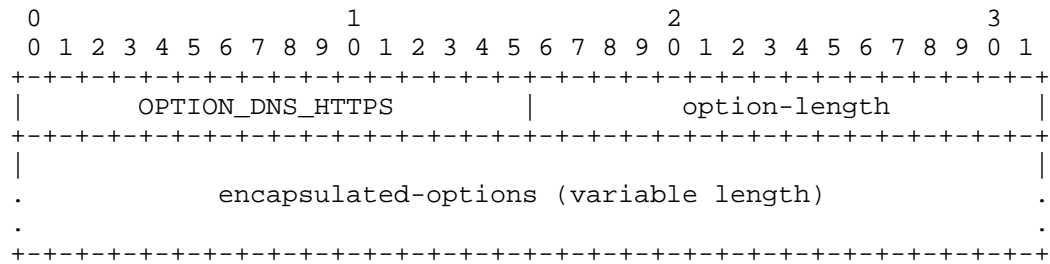


Figure 3: DNS over HTTPS option

option-code: OPTION_DNS_HTTPS (TBD)

option-len: Length of the sum of the lengths of the encapsulated options.

4. DHCPv6 DNS over TLS Encapsulated Options

There are four possible DHCPv6 encapsulated options contained in a top level OPTION_DNS_TLS option. Each sub-option MUST NOT appear more than once within a top level option.

4.1. IPv6 Address Option

The first is an OPTION_IPV6 which is REQUIRED. This is a fixed length and contains the IPv6 address of the server.

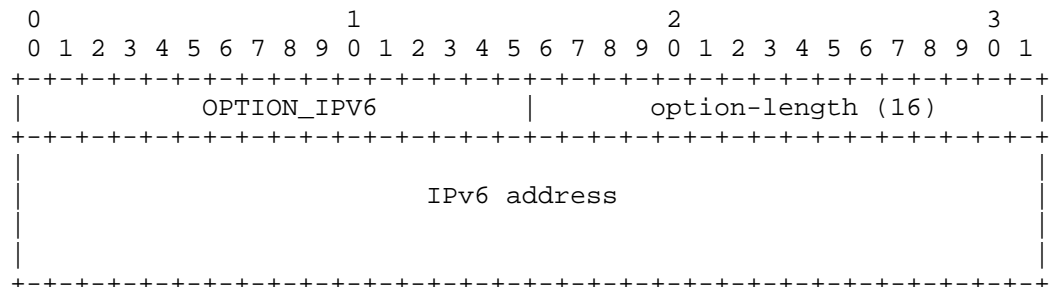


Figure 4: IPv6 option

option-code: OPTION_IPV6 (TBD)

option-len: 16 bytes

4.2. ADN Option

The second is `OPTION_ADN`. This is a variable length string containing the Authentication Domain Name as specified in [RFC8310]. This name **MUST** be verified in accordance with [RFC6125] or subsequent updates to this document. The client **SHOULD** send the Authenticated Domain Name when establishing the TLS connection to the DNS server using the TLS Server Name Indication (SNI) extension as defined in Section 3 of [RFC6066]. The use of `OPTION_ADN` by the server is **OPTIONAL** but strongly encouraged. The string is a DNS FQDN encoded according to Section 3.1 of [RFC1035].

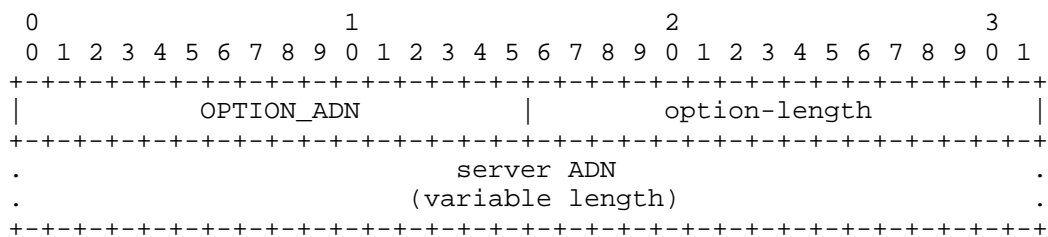


Figure 5: Server Name Indication option

option-code: `OPTION_ADN` (TBD)

option-len: Length of the encoded string

4.3. Port Option

The third option is `OPTION_PORT`. This is a fixed length option containing the port number of the listening server. It defaults to port 853 as defined in Section 3.1 of [RFC7858]. This DHCPv6 option is **OPTIONAL** and there is no need to specify it when the server is listening on port 853.

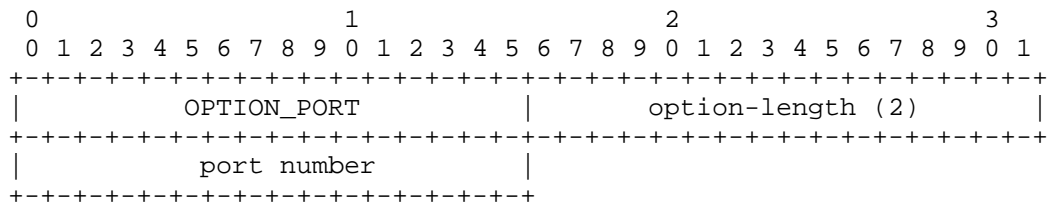


Figure 6: Port option

option-code: OPTION_PORT (TBD)

option-len: 2 bytes

5. DHCPv6 DNS over DTLS Encapsulated Options

DNS over DTLS has the exact same set of possible DHCPv6 options as DNS over TLS. The OPTION_IPV6 defined in Section 4.1 is REQUIRED. OPTION_ADN defined in Section 4.2 is OPTIONAL. OPTION_PORT defined in Section 4.3 is OPTIONAL. Please refer the these sections for their definitions and descriptions.

6. DHCPv6 DNS over HTTPS (DoH) Encapsulated Options

The DNS over HTTPS top level OPTION_DNS_HTTPS encapsulation has only one option defined at this time which is OPTION_URI. This option is REQUIRED.

[[Q1: Should we allow OPTION_IPV6? --TJP]]

6.1. URI Option

OPTION_URI includes a server URI string that provides the necessary components to connect to a DNS over HTTPS server as defined in [I-D.ietf-doh-dns-over-https]. Note that the DNS over HTTPS specification requires a server to respond to both GET and POST methods. The URI MUST NOT include the query component beginning with a "?" and including the "dns" variable that is used in a GET method request. It is up to the client to decide whether to issue a GET or POST method in the HTTP request. Therefore, the client is responsible for appending the "?" and "dns" variable along with its base64 encoded value to the URI for GET method HTTP requests.

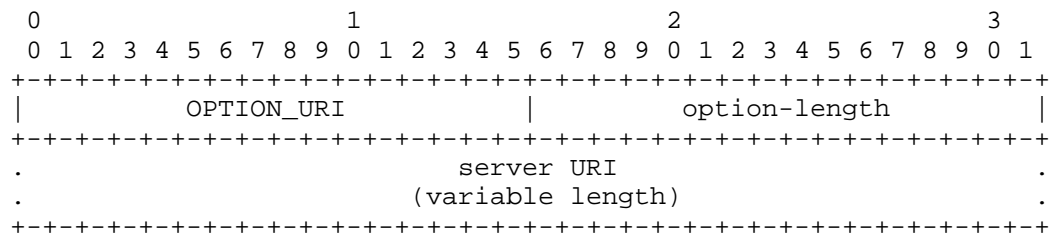


Figure 7: URI option

option-code: OPTION_URI (TBD)

option-len: Length of server URI string

7. Security Considerations

Compromising domain name resolution may provide a way to direct client network traffic to non-authentic service providers. Sending a DNS client to a non-authentic DNS server could return DNS responses with IPv6 addresses that do not represent the actual authoritative AAAA records for the names queried but pretend to do so. Then applications on the client computer would attempt to connect to the server carrying out man-in-the-middle or trojan attacks. Before this specification existed, DHCPv6 domain name servers could have directed DHCPv6 clients to compromised DNS servers. Adding encrypted DNS configuration parameters does not change this fact.

There are ways to verify the integrity of unencrypted DNS responses using DNSSEC if a client begins with the root trust anchor. This ensures the entire DNS root hasn't been replaced with a forgery.

In the same way, the integrity of the responses must still be verified when the responses are received over an encrypted DNS connection.

There are additional verification checks that can be done given the additional parameters provided with these private DNS DHCPv6 options to increase the likelihood a client is connecting to an authentic DNS recursive resolver that are not possible if only the IPv6 address of the DNS server is known:

1. When the ADN option is present, the client can use DNSSEC to validate the address records for the DNS server and the matching authentication domain name, followed by verifying the certificate of the encrypted DNS Server through verification of the corresponding TLSA records as described in DANE [RFC6698] and updated in [RFC7671].
2. There is intentionally no option for the SPKI pin as defined in [RFC7469] and usage as related to DNS as described in Section 4.2 of [RFC7858]. This is because there is no way for a client to check the integrity of the pin when received from the network operator via DHCPv6. The SPKI pin can still be used to validate a private DNS server certificate but the SPKI pin must be obtained out of band through a trusted method to be useful for verification.

8. IANA Considerations

This document identifies several new DHCPv6 Option Codes that require an assigned number.

Name	Option Code	Definition
OPTION_DNS_TLS	TBD	Section 3
OPTION_DNS_DTLS	TBD	Section 3
OPTION_DNS_HTTPS	TBD	Section 3
OPTION_IPV6	TBD	Section 4.1
OPTION_ADN	TBD	Section 4.2
OPTION_PORT	TBD	Section 4.3
OPTION_URI	TBD	Section 6.1

Table 1

9. Acknowledgements

This document was motivated in part by Section 7.3.1 of [RFC8310]. Thanks to the authors Sara Dickinson, Daniel Kahn Gillmor, and Tirumaleswar Reddy for documenting the issue. Thanks also to Ted Lemon for appropriate warnings about this work.

10. References

10.1. Normative References

- [I-D.ietf-doh-dns-over-https]
Hoffman, P. and P. McManus, "DNS Queries over HTTPS (DoH)", draft-ietf-doh-dns-over-https-12 (work in progress), June 2018.
- [RFC1035] Mockapetris, P., "Domain names - implementation and specification", STD 13, RFC 1035, DOI 10.17487/RFC1035, November 1987, <<https://www.rfc-editor.org/info/rfc1035>>.
- [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/info/rfc2119>>.
- [RFC3315] Droms, R., Ed., Bound, J., Volz, B., Lemon, T., Perkins, C., and M. Carney, "Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", RFC 3315, DOI 10.17487/RFC3315, July 2003, <<https://www.rfc-editor.org/info/rfc3315>>.

- [RFC6066] Eastlake 3rd, D., "Transport Layer Security (TLS) Extensions: Extension Definitions", RFC 6066, DOI 10.17487/RFC6066, January 2011, <<https://www.rfc-editor.org/info/rfc6066>>.
- [RFC6125] Saint-Andre, P. and J. Hodges, "Representation and Verification of Domain-Based Application Service Identity within Internet Public Key Infrastructure Using X.509 (PKIX) Certificates in the Context of Transport Layer Security (TLS)", RFC 6125, DOI 10.17487/RFC6125, March 2011, <<https://www.rfc-editor.org/info/rfc6125>>.
- [RFC6698] Hoffman, P. and J. Schlyter, "The DNS-Based Authentication of Named Entities (DANE) Transport Layer Security (TLS) Protocol: TLSA", RFC 6698, DOI 10.17487/RFC6698, August 2012, <<https://www.rfc-editor.org/info/rfc6698>>.
- [RFC7469] Evans, C., Palmer, C., and R. Sleevi, "Public Key Pinning Extension for HTTP", RFC 7469, DOI 10.17487/RFC7469, April 2015, <<https://www.rfc-editor.org/info/rfc7469>>.
- [RFC7671] Dukhovni, V. and W. Hardaker, "The DNS-Based Authentication of Named Entities (DANE) Protocol: Updates and Operational Guidance", RFC 7671, DOI 10.17487/RFC7671, October 2015, <<https://www.rfc-editor.org/info/rfc7671>>.
- [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/info/rfc7858>>.
- [RFC8094] Reddy, T., Wing, D., and P. Patil, "DNS over Datagram Transport Layer Security (DTLS)", RFC 8094, DOI 10.17487/RFC8094, February 2017, <<https://www.rfc-editor.org/info/rfc8094>>.
- [RFC8310] Dickinson, S., Gillmor, D., and T. Reddy, "Usage Profiles for DNS over TLS and DNS over DTLS", RFC 8310, DOI 10.17487/RFC8310, March 2018, <<https://www.rfc-editor.org/info/rfc8310>>.

10.2. Informative References

- [RFC3646] Droms, R., Ed., "DNS Configuration options for Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", RFC 3646, DOI 10.17487/RFC3646, December 2003, <<https://www.rfc-editor.org/info/rfc3646>>.

Appendix A. ISC DHCPv6 Configuration Example

The DHCPv6 options defined in this specification were tested with the ISC DHCPv6 server `_dhcpd_` and client `_dhclient_` version 4.4.1. Using this version, it was possible to send a single DNS over TLS encapsulated option containing an IPv6 address, authentication domain name, and port number. Multiple servers using multiple DNS over TLS encapsulated options were not available via the client hooks script.

A.1. ISC DHCPv6 Server Configuration

```
option space tls;
option tls.ipv6 code 226 = ip6-address;
option tls.port code 227 = unsigned integer 16;
option tls.adn code 228 = domain-list;
option dhcp6.tls-encapsulation code 225 = encapsulate tls;

subnet6 2001:DB8:01::/64 {
    option tls.ipv6 2a04:b900:0:100::37;
    option tls.adn "getdnsapi.net";

    option tls.ipv6 2620:fe::fe;
    option tls.adn "dns.quad9.net";
}
```

A.2. ISC DHCPv6 Client Configuration

```
option space tls;
option tls.ipv6 code 226 = ip6-address;
option tls.port code 227 = unsigned integer 16;
option tls.adn code 228 = domain-list;
option dhcp6.tls-encapsulation code 225 = encapsulate tls;

request dhcp6.tls-encapsulation;
```

Authors' Addresses

Tom Pusateri
Unaffiliated
Raleigh, NC 27608
USA

Phone: +1 919 867 1330
Email: pusateri@bangj.com

Willem Toorop
NLnet Labs
Science Park 400
Amsterdam 1098 XH
Netherlands

Email: willem@nlnetlabs.nl