

Domain Name System Operations
Internet-Draft
Intended status: Best Current Practice
Expires: May 17, 2018

J. Kristoff
DePaul University
D. Wessels
Verisign
November 13, 2017

DNS Transport over TCP - Operational Requirements
draft-ietf-dnsop-dns-tcp-requirements-01

Abstract

This document encourages the practice of permitting DNS messages to be carried over TCP on the Internet. It also describes some of the consequences of this behavior and the potential operational issues that can arise when this best common practice is not upheld.

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 May 17, 2018.

Copyright Notice

Copyright (c) 2017 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.

Internet-Draft

DNS Transport over TCP

November 2017

Table of Contents

1.	Introduction	2
1.1.	Requirements Language	3
2.	Background	3
2.1.	Uneven Transport Usage and Preference	3
2.2.	Waiting for Large Messages and Reliability	4
2.3.	EDNS0	4
2.4.	Fragmentation and Truncation	5
2.5.	"Only Zone Transfers Use TCP"	6
3.	DNS over TCP Requirements	6
4.	Network and System Considerations	7
5.	DNS over TCP Filtering Risks	7
5.1.	DNS Wedgie	8
5.2.	DNS Root Zone KSK Rollover	8
5.3.	DNS-over-TLS	8
6.	Logging and Monitoring	8
7.	Acknowledgments	8
8.	IANA Considerations	9
9.	Security Considerations	9
10.	Privacy Considerations	9
11.	References	9
11.1.	Normative References	9
11.2.	Informative References	9
Appendix A.	Standards Related to DNS Transport over TCP	13
A.1.	TODO - additional, relevant RFCs	13
A.2.	IETF RFC 7477 - Child-to-Parent Synchronization in DNS	13
A.3.	IETF RFC 7720 - DNS Root Name Service Protocol and Deployment Requirements	13
A.4.	IETF RFC 7766 - DNS Transport over TCP - Implementation Requirements	13
A.5.	IETF RFC 7828 - The edns-tcp-keepalive EDNS0 Option	14
A.6.	IETF RFC 7858 - Specification for DNS over Transport Layer Security (TLS)	14
A.7.	IETF RFC 7873 - Domain Name System (DNS) Cookies	14
A.8.	IETF RFC 7901 - CHAIN Query Requests in DNS	14
A.9.	IETF RFC 8027 - DNSSEC Roadblock Avoidance	14
A.10.	IETF RFC 8094 - DNS over Datagram Transport Layer Security (DTLS)	15
A.11.	IETF RFC 8162 - Using Secure DNS to Associate Certificates with Domain Names for S/MIME	15
	Authors' Addresses	15

1. Introduction

DNS messages may be delivered using UDP or TCP communications. While most DNS transactions are carried over UDP, some operators have been led to believe that any DNS over TCP traffic is unwanted or

unnecessary for general DNS operation. As usage and features have evolved, TCP transport has become increasingly important for correct and safe operation of the Internet DNS. Reflecting modern usage, the DNS standards were recently updated to declare support for TCP is now a required part of the DNS implementation specifications in [\[RFC7766\]](#). This document is the formal requirements equivalent for the operational community, encouraging operators to ensure DNS over TCP communications support is on par with DNS over UDP communications.

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 [RFC 2119](#) [\[RFC2119\]](#).

2. Background

2.1. Uneven Transport Usage and Preference

In the original suite of DNS specifications, [\[RFC1034\]](#) and [\[RFC1035\]](#) clearly specified that DNS messages could be carried in either UDP or TCP, but they also made clear a preference for UDP as the transport for queries in the general case. As stated in [\[RFC1035\]](#):

"While virtual circuits can be used for any DNS activity, datagrams are preferred for queries due to their lower overhead and better performance."

Another early, important, and influential document, [\[RFC1123\]](#), detailed the preference for UDP more explicitly:

"DNS resolvers and recursive servers MUST support UDP, and SHOULD support TCP, for sending (non-zone-transfer) queries."

and further stipulated:

A name server MAY limit the resources it devotes to TCP queries, but it SHOULD NOT refuse to service a TCP query just because it would have succeeded with UDP.

Culminating in [[RFC1536](#)], DNS over TCP came to be associated primarily with the zone transfer mechanism, while most DNS queries and responses were seen as the dominion of UDP.

[2.2.](#) Waiting for Large Messages and Reliability

As stipulated in the original specifications, DNS messages over UDP were restricted to a 512-byte message size. However, even while [[RFC1123](#)] made a clear preference for UDP, it foresaw DNS over TCP becoming more popular in the future:

"[...] it is also clear that some new DNS record types defined in the future will contain information exceeding the 512 byte limit that applies to UDP, and hence will require TCP.

At least two new, widely anticipated developments were set to elevate the need for DNS over TCP transactions. The first was dynamic updates defined in [[RFC2136](#)] and the second was the set of extensions collectively known as DNSSEC originally specified in [[RFC2541](#)]. The former suggested "requestors who require an accurate response code must use TCP", while the latter warned "[...] larger keys increase the size of KEY and SIG RRs. This increases the chance of DNS UDP packet overflow and the possible necessity for using higher overhead TCP in responses."

Yet defying some expectations, DNS over TCP remained little used in real traffic across the Internet. Dynamic updates saw little deployment between autonomous networks. Around the time DNSSEC was first defined, another new feature affecting DNS over UDP helped solidify its dominance for message transactions.

[2.3.](#) EDNS0

In 1999 the IETF published the Extension Mechanisms for DNS (EDNS0) in [[RFC2671](#)]. This document standardized a way for communicating DNS nodes to perform rudimentary capabilities negotiation. One such capability written into the base specification and present in every EDNS0 compatible message is the value of the maximum UDP payload size the sender can support. This unsigned 16-bit field specifies in bytes the maximum (possibly fragmented) DNS message size a node is capable of receiving. In practice, typical values are a subset of the 512 to 4096 byte range. EDNS0 became widely deployed over the next several years and numerous surveys have shown many systems currently support larger UDP MTUs [[CASTRO2010](#)], [[NETALYZR](#)] with EDNS0.

The natural effect of EDNS0 deployment meant DNS messages larger than 512 bytes would be less reliant on TCP than they might otherwise have been. While a non-negligible population of DNS systems lack EDNS0 or may still fall back to TCP for some transactions, DNS over TCP transactions remain a very small fraction of overall DNS traffic [[VERISIGN](#)].

[2.4.](#) Fragmentation and Truncation

Although EDNS0 provides a way for endpoints to signal support for DNS messages exceeding 512 bytes, the realities of today's Internet mean large messages do not always get through. Any IP packet whose size exceeds the MTU of a link it transits will be fragmented and then reassembled by the receiving host. Unfortunately, it is not uncommon for middleboxes and firewalls to block IP fragments. If one or more fragments do not arrive, the application does not receive the message and the request times out.

For IPv4-connected hosts, the de-facto MTU is the Ethernet payload size of 1500 bytes. This means that the largest unfragmented UDP DNS message that can be sent over IPv4 is 1472 bytes. For IPv6 the situation is a little more complicated. First, IPv6 headers are 40 bytes (versus 20 in IPv4). Second, it seems as though some people have mis-interpreted IPv6's required minimum MTU of 1280 as a required maximum. Third, fragmentation in IPv6 can only be done by the host originating the packet. The need to fragment is conveyed in an ICMPv6 "packet too big" message. The originating host indicates a fragmented packet with IPv6 extension headers. Unfortunately, it is quite common for both ICMPv6 and IPv6 extension headers to be blocked

by middleboxes. According to [\[HUSTON\]](#) some 35% of IPv6-capable recursive resolvers are unable to receive a fragmented IPv6 packet.

The practical consequence of all this is that DNS requestors must be prepared to retry queries with different EDNS0 maximum message size values. Administrators of BIND are likely to be familiar with seeing "success resolving ... after reducing the advertised EDNS0 UDP packet size to 512 octets" in their system logs.

Often, reducing the EDNS0 UDP packet size leads to a successful response. That is, the necessary data fits within the smaller packet size. However, when the data does not fit, the server sets the truncated flag in its response, indicating the client should retry over TCP to receive the whole response. This is undesirable from the client's point of view because it adds more latency, and potentially undesirable from the server's point of view due to the increased resource requirements of TCP.

The issues around fragmentation, truncation, and TCP are driving certain implementation and policy decisions in the DNS. Notably, Cloudflare implemented what it calls "DNSSEC black lies" [\[CLOUDFLARE\]](#) and uses ECDSA algorithms, such that their signed responses fit easily in 512 bytes. The KSK Rollover design team [\[DESIGNTEAM\]](#) spent a lot of time thinking and worrying about response sizes. There is growing sentiment in the DNSSEC community that RSA key sizes beyond 2048-bits are impractical and that critical infrastructure zones

should transition to elliptic curve algorithms to keep response sizes manageable.

[2.5.](#) "Only Zone Transfers Use TCP"

There are many in the DNS community who configure DNS over TCP services and expect DNS over TCP transactions to occur without interference. However there has also been a long held belief by some operators, particularly for security-related reasons, that DNS over TCP services should be purposely limited or not provided at all [\[CHES94\]](#), [\[DJBDNS\]](#). A popular meme has also held the imagination of some that DNS over TCP is only ever used for zone transfers and is generally unnecessary otherwise, with filtering all DNS over TCP traffic even described as a best practice.

The position on restricting DNS over TCP had some justification given that historic implementations of DNS nameservers provided very little in the way of TCP connection management (for example see [Section 6.1.2 of \[RFC7766\]](#) for more details). However modern standards and implementations are moving to align with the more sophisticated TCP management techniques employed by, for example, HTTP(S) servers and load balancers.

3. DNS over TCP Requirements

An average increase in DNS message size, the continued development of new DNS features and a denial of service mitigation technique (see [Section 9](#)) have suggested that DNS over TCP transactions are as important to the correct and safe operation of the Internet DNS as ever, if not more so. Furthermore, there has been serious research that has suggested connection-oriented DNS transactions may provide security and privacy advantages over UDP transport [[TDNS](#)]. In fact, [[RFC7858](#)], a Standards Track document is just this sort of specification. Therefore, it might be desirable for network operators to avoid artificially inhibiting the potential utility and advances in the DNS such as these.

[Section 6.1.3.2 in \[RFC1123\]](#) is updated: All general-purpose DNS servers MUST be able to service both UDP and TCP queries.

- o Authoritative servers MUST service TCP queries so that they do not limit the size of responses to what fits in a single UDP packet.
- o Recursive servers (or forwarders) MUST service TCP queries so that they do not prevent large responses from a TCP-capable server from reaching its TCP-capable clients.

Regarding the choice of limiting the resources a server devotes to queries, [Section 6.1.3.2 in \[RFC1123\]](#) also says:

A name server MAY limit the resources it devotes to TCP queries, but it SHOULD NOT refuse to service a TCP query just because it would have succeeded with UDP.

This requirement is hereby updated: A name server MAY limit the the

resources it devotes to queries, but it MUST NOT refuse to service a query just because it would have succeeded with another transport protocol.

Filtering of DNS over TCP is considered harmful in the general case. DNS resolver and server operators MUST provide DNS service over both UDP and TCP transports. Likewise, network operators MUST allow DNS service over both UDP and TCP transports. It must be acknowledged that DNS over TCP service can pose operational challenges that are not present when running DNS over UDP alone, and vice-versa. However, it is the aim of this document to argue that the potential damage incurred by prohibiting DNS over TCP service is more detrimental to the continued utility and success of the DNS than when its usage is allowed.

[4.](#) Network and System Considerations

TODO: refer to IETF [RFC 7766](#) connection handling discussion, various TCP hardening documents, network operator protocol and traffic best practices, etc.

[5.](#) DNS over TCP Filtering Risks

Networks that filter DNS over TCP risk losing access to significant or important pieces of the DNS name space. For a variety of reasons a DNS answer may require a DNS over TCP query. This may include large message sizes, lack of EDNS0 support, DDoS mitigation techniques, or perhaps some future capability that is as yet unforeseen will also demand TCP transport.

For example, both [[RFC7901](#)] and [[draft-extra-answers](#)] describe latency-avoiding techniques that send extra data in DNS responses. This makes the responses larger and potentially damaging in DDoS reflection attacks. The former mandates the use of TCP or DNS Cookies ([\[RFC7873\]](#)) and the latter offers it as a possibility in Security Considerations.

Even if any or all particular answers have consistently been returned successfully with UDP in the past, this continued behavior cannot be guaranteed when DNS messages are exchanged between autonomous

systems. Therefore, filtering of DNS over TCP is considered harmful

and contrary to the safe and successful operation of the Internet. This section enumerates some of the known risks we know about at the time of this writing when networks filter DNS over TCP.

[5.1.](#) DNS Wedgie

Networks that filter DNS over TCP may inadvertently cause problems for third party resolvers as experienced by [\[TOYAMA\]](#). If for instance a resolver receives a truncated answer from a server, but when the resolver resends the query using TCP and the TCP response never arrives, not only will full answer be unavailable, but the resolver will incur the full extent of TCP retransmissions and time outs. This situation might place extreme strain on resolver resources. If the number and frequency of these truncated answers are sufficiently high, we refer to the steady-state of lost resources as a result a "DNS" wedgie". A DNS wedgie is often not easily or completely mitigated by the affected DNS resolver operator.

[5.2.](#) DNS Root Zone KSK Rollover

Recent plans for a new root zone DNSSEC KSK have highlighted a potential problem in retrieving the keys.[\[LEWIS\]](#) Some packets in the KSK rollover process will be larger than 1280 bytes, the IPv6 minimum MTU for links carrying IPv6 traffic.[\[RFC2460\]](#) While studies have shown that problems due to fragment filtering or an inability to generate and receive these larger messages are negligible, any DNS server that is unable to receive large DNS over UDP messages or perform DNS over TCP may experience severe disruption of DNS service if performing DNSSEC validation.

[5.3.](#) DNS-over-TLS

TODO: DNS-over-TLS

[6.](#) Logging and Monitoring

TODO: Developers of applications that log or monitor DNS are advised to not ignore TCP because it is hard. Also be prepared for connection reuse, pipelining, and out-of-order responses. If using packet-based (e.g. libpcap) collection techniques, the application must be prepared to implement/perform TCP segment reassembly.

[7.](#) Acknowledgments

This document was initially motivated by feedback from students who pointed out that they were hearing contradictory information about filtering DNS over TCP messages. Thanks in particular to a teaching

colleague, JPL, who perhaps unknowingly encouraged the initial research into the differences of what the community has historically said and did. Thanks to all the NANOG 63 attendees who provided feedback to an early talk on this subject.

The following individuals provided an array of feedback to help improve this document: Sara Dickinson, Bob Harold, Tatuya Jinmei, and Paul Hoffman. The authors are indebted to their contributions. Any remaining errors or imperfections are the sole responsibility of the document authors.

8. IANA Considerations

This memo includes no request to IANA.

9. Security Considerations

Ironically, returning truncated DNS over UDP answers in order to induce a client query to switch to DNS over TCP has become a common response to source address spoofed, DNS denial-of-service attacks [[RRL](#)]. Historically, operators have been wary of TCP-based attacks, but in recent years, UDP-based flooding attacks have proven to be the most common protocol attack on the DNS. Nevertheless, a high rate of short-lived DNS transactions over TCP may pose challenges. While many operators have provided DNS over TCP service for many years without duress, past experience is no guarantee of future success.

DNS over TCP is not unlike many other Internet TCP services. TCP threats and many mitigation strategies have been well documented in a series of documents such as [[RFC4953](#)], [[RFC4987](#)], [[RFC5927](#)], and [[RFC5961](#)].

10. Privacy Considerations

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/info/rfc2119>>.

11.2. Informative References

Internet-Draft

DNS Transport over TCP

November 2017

[CASTRO2010]

Castro, S., Zhang, M., John, W., Wessels, D., and k. claffy, "Understanding and preparing for DNS evolution", 2010.

[CHES94]

Cheswick, W. and S. Bellovin, "Firewalls and Internet Security: Repelling the Wily Hacker", 1994.

[CLOUDFLARE]

Grant, D., "Economical With The Truth: Making DNSSEC Answers Cheap", June 2016, <<https://blog.cloudflare.com/black-lies/>>.

[DESIGNTEAM]

Design Team Report, "Root Zone KSK Rollover Plan", December 2015, <<https://www.iana.org/reports/2016/root-ksk-rollover-design-20160307.pdf>>.

[DJBDNS]

D.J. Bernstein, "When are TCP queries sent?", 2002, <<https://cr.yp.to/djbdns/tcp.html#why>>.

[[draft-extra-answers](#)]

Kumari, W., Yan, Z., Hardaker, W., and D. Lawrence, "Returning extra answers in DNS responses.", [draft-wkumari-dnsop-multiple-responses-05](#) (work in progress), July 2017.

[HUSTON]

Huston, G., "Dealing with IPv6 fragmentation in the DNS", August 2017, <<https://blog.apnic.net/2017/08/22/dealing-ipv6-fragmentation-dns/>>.

[LEWIS]

Lewis, E., "2017 DNSSEC KSK Rollover", RIPE 74 Budapest, Hungary, May 2017, <<https://ripe74.ripe.net/presentations/25-RIPE74-lewis-submission.pdf>>.

[NETALYZR]

Kreibich, C., Weaver, N., Nechaev, B., and V. Paxson, "Netalyzr: Illuminating The Edge Network", 2010.

- [RFC1034] Mockapetris, P., "Domain names - concepts and facilities", STD 13, [RFC 1034](https://www.rfc-editor.org/info/rfc1034), DOI 10.17487/RFC1034, November 1987, <<https://www.rfc-editor.org/info/rfc1034>>.
- [RFC1035] Mockapetris, P., "Domain names - implementation and specification", STD 13, [RFC 1035](https://www.rfc-editor.org/info/rfc1035), DOI 10.17487/RFC1035, November 1987, <<https://www.rfc-editor.org/info/rfc1035>>.

Kristoff & Wessels

Expires May 17, 2018

[Page 10]

Internet-Draft

DNS Transport over TCP

November 2017

- [RFC1123] Braden, R., Ed., "Requirements for Internet Hosts - Application and Support", STD 3, [RFC 1123](https://www.rfc-editor.org/info/rfc1123), DOI 10.17487/RFC1123, October 1989, <<https://www.rfc-editor.org/info/rfc1123>>.
- [RFC1536] Kumar, A., Postel, J., Neuman, C., Danzig, P., and S. Miller, "Common DNS Implementation Errors and Suggested Fixes", [RFC 1536](https://www.rfc-editor.org/info/rfc1536), DOI 10.17487/RFC1536, October 1993, <<https://www.rfc-editor.org/info/rfc1536>>.
- [RFC2136] Vixie, P., Ed., Thomson, S., Rekhter, Y., and J. Bound, "Dynamic Updates in the Domain Name System (DNS UPDATE)", [RFC 2136](https://www.rfc-editor.org/info/rfc2136), DOI 10.17487/RFC2136, April 1997, <<https://www.rfc-editor.org/info/rfc2136>>.
- [RFC2460] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", [RFC 2460](https://www.rfc-editor.org/info/rfc2460), DOI 10.17487/RFC2460, December 1998, <<https://www.rfc-editor.org/info/rfc2460>>.
- [RFC2541] Eastlake 3rd, D., "DNS Security Operational Considerations", [RFC 2541](https://www.rfc-editor.org/info/rfc2541), DOI 10.17487/RFC2541, March 1999, <<https://www.rfc-editor.org/info/rfc2541>>.
- [RFC2671] Vixie, P., "Extension Mechanisms for DNS (EDNS0)", [RFC 2671](https://www.rfc-editor.org/info/rfc2671), DOI 10.17487/RFC2671, August 1999, <<https://www.rfc-editor.org/info/rfc2671>>.
- [RFC4953] Touch, J., "Defending TCP Against Spoofing Attacks", [RFC 4953](https://www.rfc-editor.org/info/rfc4953), DOI 10.17487/RFC4953, July 2007, <<https://www.rfc-editor.org/info/rfc4953>>.
- [RFC4987] Eddy, W., "TCP SYN Flooding Attacks and Common

Mitigations", [RFC 4987](#), DOI 10.17487/RFC4987, August 2007, <<https://www.rfc-editor.org/info/rfc4987>>.

[RFC5927] Gont, F., "ICMP Attacks against TCP", [RFC 5927](#), DOI 10.17487/RFC5927, July 2010, <<https://www.rfc-editor.org/info/rfc5927>>.

[RFC5961] Ramaiah, A., Stewart, R., and M. Dalal, "Improving TCP's Robustness to Blind In-Window Attacks", [RFC 5961](#), DOI 10.17487/RFC5961, August 2010, <<https://www.rfc-editor.org/info/rfc5961>>.

[RFC7477] Hardaker, W., "Child-to-Parent Synchronization in DNS", [RFC 7477](#), DOI 10.17487/RFC7477, March 2015, <<https://www.rfc-editor.org/info/rfc7477>>.

Kristoff & Wessels

Expires May 17, 2018

[Page 11]

Internet-Draft

DNS Transport over TCP

November 2017

[RFC7720] Blanchet, M. and L-J. Liman, "DNS Root Name Service Protocol and Deployment Requirements", [BCP 40](#), [RFC 7720](#), DOI 10.17487/RFC7720, December 2015, <<https://www.rfc-editor.org/info/rfc7720>>.

[RFC7766] Dickinson, J., Dickinson, S., Bellis, R., Mankin, A., and D. Wessels, "DNS Transport over TCP - Implementation Requirements", [RFC 7766](#), DOI 10.17487/RFC7766, March 2016, <<https://www.rfc-editor.org/info/rfc7766>>.

[RFC7828] Wouters, P., Abley, J., Dickinson, S., and R. Bellis, "The edns-tcp-keepalive EDNS0 Option", [RFC 7828](#), DOI 10.17487/RFC7828, April 2016, <<https://www.rfc-editor.org/info/rfc7828>>.

[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>>.

[RFC7873] Eastlake 3rd, D. and M. Andrews, "Domain Name System (DNS) Cookies", [RFC 7873](#), DOI 10.17487/RFC7873, May 2016, <<https://www.rfc-editor.org/info/rfc7873>>.

[RFC7901] Wouters, P., "CHAIN Query Requests in DNS", [RFC 7901](#),

DOI 10.17487/RFC7901, June 2016,
<<https://www.rfc-editor.org/info/rfc7901>>.

[RFC8027] Hardaker, W., Gudmundsson, O., and S. Krishnaswamy, "DNSSEC Roadblock Avoidance", [BCP 207](#), [RFC 8027](#), DOI 10.17487/RFC8027, November 2016, <<https://www.rfc-editor.org/info/rfc8027>>.

[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>>.

[RFC8162] Hoffman, P. and J. Schlyter, "Using Secure DNS to Associate Certificates with Domain Names for S/MIME", [RFC 8162](#), DOI 10.17487/RFC8162, May 2017, <<https://www.rfc-editor.org/info/rfc8162>>.

[RRL] Vixie, P. and V. Schryver, "DNS Response Rate Limiting (DNS RRL)", ISC-TN 2012-1 Draft1, April 2012.

[TDNS] Zhu, L., Heidemann, J., Wessels, D., Mankin, A., and N. Somaiya, "Connection-oriented DNS to Improve Privacy and Security", 2015.

[TOYAMA] Toyama, K., Ishibashi, K., Ishino, M., Yoshimura, C., and K. Fujiwara, "DNS Anomalies and Their Impacts on DNS Cache Servers", NANOG 32 Reston, VA USA, 2004.

[VERISIGN] Thomas, M. and D. Wessels, "An Analysis of TCP Traffic in Root Server DITL Data", DNS-OARC 2014 Fall Workshop Los Angeles, 2014.

[Appendix A](#). Standards Related to DNS Transport over TCP

This section enumerates all known IETF RFC documents that are currently of status standard, informational, best common practice or experimental and either implicitly or explicitly make assumptions or

statements about the use of TCP as a transport for the DNS germane to this document.

[A.1.](#) TODO - additional, relevant RFCs

[A.2.](#) IETF [RFC 7477](#) - Child-to-Parent Synchronization in DNS

This standards track document [[RFC7477](#)] specifies a RRType and protocol to signal and synchronize NS, A, and AAAA resource record changes from a child to parent zone. Since this protocol may require multiple requests and responses, it recommends utilizing DNS over TCP to ensure the conversation takes place between a consistent pair of end nodes.

[A.3.](#) IETF [RFC 7720](#) - DNS Root Name Service Protocol and Deployment Requirements

This best current practice[RFC7720] declares root name service "MUST support UDP [[RFC768](#)] and TCP [[RFC793](#)] transport of DNS queries and responses."

[A.4.](#) IETF [RFC 7766](#) - DNS Transport over TCP - Implementation Requirements

The standards track document [[RFC7766](#)] might be considered the direct ancestor of this operational requirements document. The implementation requirements document codifies mandatory support for DNS over TCP in compliant DNS software.

[A.5.](#) IETF [RFC 7828](#) - The edns-tcp-keepalive EDNS0 Option

This standards track document [[RFC7828](#)] defines an EDNS0 option to negotiate an idle timeout value for long-lived DNS over TCP connections. Consequently, this document is only applicable and relevant to DNS over TCP sessions and between implementations that support this option.

[A.6.](#) IETF [RFC 7858](#) - Specification for DNS over Transport Layer Security (TLS)

This standards track document [[RFC7858](#)] defines a method for putting DNS messages into a TCP-based encrypted channel using TLS. This specification is noteworthy for explicitly targetting the stub-to-recursive traffic, but does not preclude its application from recursive-to-authoritative traffic.

[A.7.](#) IETF [RFC 7873](#) - Domain Name System (DNS) Cookies

This standards track document [[RFC7873](#)] describes an EDNS0 option to provide additional protection against query and answer forgery. This specification mentions DNS over TCP as a reasonable fallback mechanism when DNS Cookies are not available. The specification does make mention of DNS over TCP processing in two specific situations. In one, when a server receives only a client cookie in a request, the server should consider whether the request arrived over TCP and if so, it should consider accepting TCP as sufficient to authenticate the request and respond accordingly. In another, when a client receives a BADCOOKIE reply using a fresh server cookie, the client should retry using TCP as the transport.

[A.8.](#) IETF [RFC 7901](#) - CHAIN Query Requests in DNS

This experimental specification [[RFC7901](#)] describes an EDNS0 option that can be used by a security-aware validating resolver to request and obtain a complete DNSSEC validation path for any single query. This document requires the use of DNS over TCP or a source IP address verified transport mechanism such as EDNS-COOKIE.[\[RFC7873\]](#)

[A.9.](#) IETF [RFC 8027](#) - DNSSEC Roadblock Avoidance

This document [[RFC8027](#)] details observed problems with DNSSEC deployment and mitigation techniques. Network traffic blocking and restrictions, including DNS over TCP messages, are highlighted as one reason for DNSSEC deployment issues. While this document suggests these sorts of problems are due to "non-compliant infrastructure" and is of type BCP, the scope of the document is limited to detection and mitigation techniques to avoid so-called DNSSEC roadblocks.

[A.10.](#) IETF [RFC 8094](#) - DNS over Datagram Transport Layer Security (DTLS)

This experimental specification [[RFC8094](#)] details a protocol that uses a datagram transport (UDP), but stipulates that "DNS clients and

servers that implement DNS over DTLS MUST also implement DNS over TLS in order to provide privacy for clients that desire Strict Privacy [...]. This requirement implies DNS over TCP must be supported in case the message size is larger than the path MTU.

A.11. IETF [RFC 8162](#) - Using Secure DNS to Associate Certificates with Domain Names for S/MIME

This experimental specification [[RFC8162](#)] describes a technique to authenticate user X.509 certificates in an S/MIME system via the DNS. The document points out that the new experimental resource record types are expected to carry large payloads, resulting in the suggestion that "applications SHOULD use TCP -- not UDP -- to perform queries for the SMIMEA resource record."

Authors' Addresses

John Kristoff
DePaul University
Chicago, IL 60604
US

Phone: +1 312 493 0305
Email: jtk@depaul.edu
URI: <https://aharp.iorc.depaul.edu>

Duane Wessels
Verisign
12061 Bluemont Way
Reston, VA 20190
US

Phone: +1 703 948 3200
Email: dwessels@verisign.com
URI: <http://verisigninc.com>