dprive
Internet-Draft
Intended status: Standards Track
Expires: September 22, 2016

S. Dickinson
Sinodun
D. Gillmor
ACLU
T. Reddy
Cisco

March 21, 2016

Authentication and (D)TLS Profile for DNS-over-TLS and DNS-over-DTLS
draft-ietf-dprive-dtls-and-tls-profiles-01

Abstract

This document describes how a DNS client can use a domain name to authenticate a DNS server that uses Transport Layer Security (TLS) and Datagram TLS (DTLS). Additionally, it defines (D)TLS profiles for DNS clients and servers implementing DNS-over-TLS and DNS-over-DTLS.

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 http://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 September 22, 2016.

Table of Contents

1. Introduction

The DPRIVE working group has two active documents that provide DNS privacy between DNS clients and DNS servers (to address the concerns in [RFC7626]):

- DNS-over-TLS [I-D.ietf-dprive-dns-over-tls]

- DNS-over-DTLS [I-D.ietf-dprive-dnsodtls]

This document defines usage profiles and authentication mechanisms for DTLS [RFC6347] and TLS [RFC5246] that specify how a DNS client should authenticate a DNS server based on a domain name. In particular, it describes:

- How a DNS client can obtain a domain name for a DNS server to use for (D)TLS authentication.

- What are the acceptable credentials a DNS server can present to prove its identity for (D)TLS authentication based on a given domain name.

- How a DNS client can verify that any given credential matches the domain name obtained for a DNS server.

This document also defines a (D)TLS protocol profile for use with DNS. This profile defines the configuration options and protocol extensions required of both parties to optimize connection establishment and session resumption for transporting DNS, and to support the authentication profiles defined here.

2. Terminology

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].

Several terms are used specifically in the context of this draft:

- DNS client: a DNS stub resolver or forwarder/proxy. In the case of a forwarder, the term "DNS client" is used to discuss the side that sends queries.

- DNS server: a DNS recursive resolver or forwarder/proxy. In the case of a forwarder, the term "DNS server" is used to discuss the side that responds to queries.

- Privacy-enabling DNS server: A DNS server that:

  * MUST implement DNS-over-TLS [I-D.ietf-dprive-dns-over-tls] and MAY implement DNS-over-DTLS [I-D.ietf-dprive-dnsodtls].

  * Can offer at least one of the credentials described in Section 9.

    * Implements the (D)TLS profile described in Section 11.

- (D)TLS: For brevity this term is used for statements that apply to both Transport Layer Security [RFC5246] and Datagram Transport Layer Security [RFC6347]. Specific terms will be used for any statement that applies to either protocol alone.

- DNS-over-(D)TLS: For brevity this term is used for statements that apply to both DNS-over-TLS [I-D.ietf-dprive-dns-over-tls] and DNS-over-DTLS [I-D.ietf-dprive-dnsodtls]. Specific terms will be used for any statement that applies to either protocol alone.

- Credential: Information available for a DNS server which proves its identity for authentication purposes. Credentials discussed here include:

    * X.509 certificate

    * DNSSEC validated chain to a TLSA record

    but may also include SPKI pinsets.

- SPKI Pinsets: [I-D.ietf-dprive-dns-over-tls] describes the use of cryptographic digests to "pin" public key information in a manner similar to HPKP [RFC7469]. An SPKI pinset is a collection of these pins that constrains a DNS server.

- Reference Identifier: a Reference Identifier as described in [RFC6125], constructed by the DNS client when performing TLS authentication of a DNS server.

3. Scope

This document is limited to domain-name-based authentication of DNS servers by DNS clients (as defined in the terminology section), and the (D)TLS profiles needed to support this. As such, the following things are out of scope:

- Authentication of authoritative servers by recursive resolvers.

- Authentication of DNS clients by DNS servers.

- SPKI-pinset-based authentication. This is defined in [I-D.ietf-dprive-dns-over-tls]. However, Section 10 does describe how to combine that approach with the domain name based mechanism described here.

- Any server identifier other than domain names, including IP address, organizational name, country of origin, etc.

4. Discussion

4.1. Background

To protect against passive attacks DNS privacy requires encrypting the query (and response). Such encryption typically provides integrity protection as a side-effect, which means on-path attackers cannot simply inject bogus DNS responses. For DNS privacy to also provide protection against active attackers pretending to be the server, the client must authenticate the server.

4.2. Usage Profiles

A DNS client has a choice of privacy usage profiles available. This choice is briefly discussed in both [I-D.ietf-dprive-dns-over-tls] and [I-D.ietf-dprive-dnsodtls]. In summary, the usage profiles are:

- Strict Privacy: the DNS client requires both an encrypted and authenticated connection to a privacy-enabling DNS Server. A hard failure occurs if this is not available. This requires the client to securely obtain information it can use to authenticate the server. This profile can include some initial meta queries (performed using Opportunistic Privacy) to securely obtain the IP address and authentication information for the privacy-enabling DNS server to which the DNS client will subsequently connect. The rationale for this is that requiring Strict Privacy for such meta queries would introduce significant deployment obstacles. This profile provides strong privacy guarantees to the client. This is discussed in detail in Section 6.

- Opportunistic Privacy: the DNS client uses Opportunistic Security as described in [RFC7435]

  > "... the use of cleartext as the baseline communication security policy, with encryption and authentication negotiated and applied to the communication when available."

  In the best case scenario (authenticated and encrypted connection) this is equivalent to Strict Privacy, in the worst case (clear text connection) this is equivalent to No Privacy. Clients will try for the best case but are willing to fallback to intermediate cases and eventually the worst case scenario in order to obtain a response. This provides an undetermined privacy guarantee to the user depending on what kind of connection is actually used. This is discussed in Section 5.

o No Privacy: the DNS client does not require or attempt to use either encryption or authentication. Queries are always sent in clear text. This provides no privacy guarantees to the client.

Usage Profile	Passive Attacker	Active Attacker
No Privacy	N	N
Opportunistic Privacy	N (D)	N (D)
Strict Privacy	P	P

P == protection; N == no protection; D == detection is possible

Table 1: DNS Privacy Protection by Usage Profile and type of attacker

Since Strict Privacy provides the strongest privacy guarantees it is preferable to Opportunistic Privacy which is preferable to No Privacy. However since the different profiles require varying levels of configuration (or a trusted relationship with a provider) DNS clients will need to carefully select which profile to use based on their communication privacy needs. For the case where a client has a trusted relationship with a provider it is expected that the provider will provide either a domain name or SPKI pinset via a secure out-of-band mechanism and therefore Strict Privacy should be used.

A DNS client SHOULD select a particular usage profile when resolving a query. A DNS client MUST NOT fallback from Strict Privacy to Opportunistic Privacy during the resolution process as this could invalidate the protection offered against active attackers.

4.3. Authentication

This document describes authentication mechanisms that can be used in either Strict or Opportunistic Privacy for DNS-over-(D)TLS.

4.3.1. DNS-over-(D)TLS Bootstrapping Problems

Many (D)TLS clients use PKIX authentication [RFC6125] based on a domain name for the server they are contacting. These clients typically first look up the server's network address in the DNS before making this connection. A DNS client therefore has a bootstrap problem. DNS clients typically know only the IP address of a DNS server.

As such, before connecting to a DNS server, a DNS client needs to learn the domain name it should associate with the IP address of a

DNS server for authentication purposes. Sources of domains names are discussed in Section 7 and Section 8.

One advantage of this domain name based approach is that it encourages association of stable, human recognisable identifiers with secure DNS service providers.

4.3.2. Credential Verification

The use of SPKI pinset verification is discussed in [I-D.ietf-dprive-dns-over-tls].

In terms of domain name based verification, once a domain name is known for a DNS server a choice of mechanisms can be used for authentication. Section 9 discusses these mechanisms in detail, namely X.509 certificate based authentication and DANE.

Note that the use of DANE adds requirements on the ability of the client to get validated DNSSEC results. This is discussed in more detail in Section 9.2.

4.3.3. Implementation guidance

Section 11 describes the (D)TLS profile for DNS-over(D)TLS. Additional considerations relating to general implementation guidelines are discussed in both Section 13 and in Appendix A.

5. Authentication in Opportunistic DNS-over(D)TLS Privacy

An Opportunistic Security [RFC7435] profile is described in [I-D.ietf-dprive-dns-over-tls] which MAY be used for DNS-over-(D)TLS.

DNS clients issuing queries under an opportunistic profile which know of a domain name or SPKI pinset for a given privacy-enabling DNS server MAY choose to try to authenticate the server using the mechanisms described here. This is useful for detecting (but not preventing) active attack, since the fact that authentication information is available indicates that the server in question is a privacy-enabling DNS server to which it should be possible to establish an authenticated, encrypted connection. In this case, whilst a client cannot know the reason for an authentication failure, from a privacy standpoint the client should consider an active attack in progress and proceed under that assumption. Attempting authentication is also useful for debugging or diagnostic purposes if there are means to report the result. This information can provide a basis for a DNS client to switch to (preferred) Strict Privacy where it is viable.

6. Authentication in Strict DNS-over(D)TLS Privacy

To authenticate a privacy-enabling DNS server, a DNS client needs to know the domain name for each server it is willing to contact. This is necessary to protect against active attacks on DNS privacy.

A DNS client requiring Strict Privacy MUST either use one of the sources listed in Section 8 to obtain a domain name for the server it contacts, or use an SPKI pinset as described in [I-D.ietf-dprive-dns-over-tls].

A DNS client requiring Strict Privacy MUST only attempt to connect to DNS servers for which either a domain name or a SPKI pinset is known (or both). The client MUST use the available verification mechanisms described in Section 9 to authenticate the server, and MUST abort connections to a server when no verification mechanism succeeds.

With Strict Privacy, the DNS client MUST NOT commence sending DNS queries until at least one of the privacy-enabling DNS servers becomes available.

A privacy-enabling DNS server may be temporarily unavailable when configuring a network. For example, for clients on networks that require registration through web-based login (a.k.a. "captive portals"), such registration may rely on DNS interception and spoofing. Techniques such as those used by DNSSEC-trigger [dnssec-trigger] MAY be used during network configuration, with the intent to transition to the designated privacy-enabling DNS servers after captive portal registration. The system MUST alert by some means that the DNS is not private during such bootstrap.

7. In Band Source of Domain Name: SRV Service Label

This specification adds a SRV service label "domain-s" for privacy-enabling DNS servers.

Example service records (for TLS and DTLS respectively):

```
_domain-s._tcp.dns.example.com.  SRV 0 1 853 dns1.example.com.
_domain-s._tcp.dns.example.com.  SRV 0 1 853 dns2.example.com.

_domain-s._udp.dns.example.com.  SRV 0 1 853 dns3.example.com.
```

8. Out of Band Sources of Domain Name

8.1. Full direct configuration

DNS clients may be directly and securely provisioned with the domain name of each privacy-enabling DNS server. For example, using a client specific configuration file or API.

In this case, direct configuration for a DNS client would consist of both an IP address and a domain name for each DNS server.

8.2. Direct configuration of name only

A DNS client may be configured directly and securely with only the domain name of its privacy-enabling DNS server. For example, using a client specific configuration file or API.

A DNS client might learn of a default recursive DNS resolver from an untrusted source (such as DHCP's DNS server option [RFC3646]). It can then use opportunistic DNS connections to untrusted recursive DNS resolver to establish the IP address of the intended privacy-enabling DNS server by doing a lookup of SRV records. Such records MUST be validated using DNSSEC. Private DNS resolution can now be done by the DNS client against the configured privacy-enabling DNS server.

Example:

- A DNSSEC validating DNS client is configured with the domain name dns.example.net for a privacy-enabling DNS server
- Using Opportunistic Privacy to a default DNS resolver (acquired, for example, using DHCP) the client performs look ups for
  - SRV record for _domain-s._tcp.dns.example.net to obtain the server host name
  - A and/or AAAA lookups to obtain IP address for the server host name
- Client validates all the records obtained in the previous step using DNSSEC.
- If the records successfully validate the client proceeds to connect to the privacy-enabling DNS server using Strict Privacy.

A DNS client so configured that successfully connects to a privacy-enabling DNS server MAY choose to locally cache the looked up addresses in order to not have to repeat the opportunistic lookup.

8.3. DHCP

Some clients may have an established trust relationship with a known DHCP [RFC2131] server for discovering their network configuration. In the typical case, such a DHCP server provides a list of IP addresses for DNS servers (see section 3.8 of [RFC2132]), but does not provide a domain name for the DNS server itself.

A DHCP server might use a DHCP extension to provide a list of domain names for the offered DNS servers, which correspond to IP addresses listed.

Note that this requires the client to trust the DHCP server, and to have a secured/authenticated connection to it. Therefore this mechanism may be limited to only certain environments. This document does not attempt to describe secured and trusted relationships to DHCP servers.

[NOTE: It is noted (at the time of writing) that whilst some implementation work is in progress to secure IPv6 connections for DHCP, IPv4 connections have received little to no implementation attention in this area.]

[QUESTION: The authors would like to solicit feedback on the use of DHCP to determine whether to pursue a new DHCP option in a later version of this draft, or defer it.]

9. Credential Verification

9.1. X.509 Certificate Based Authentication

When a DNS client configured with a domain name connects to its configured DNS server over (D)TLS, the server may present it with an X.509 certificate. In order to ensure proper authentication, DNS clients MUST verify the entire certification path per [RFC5280]. The DNS client additionally uses [RFC6125] validation techniques to compare the domain name to the certificate provided.

A DNS client constructs two Reference Identifiers for the server based on the domain name: A DNS-ID and an SRV-ID [RFC4985]. The DNS-ID is simply the domain name itself. The SRV-ID uses a "_domain-s." prefix. So if the configured domain name is "dns.example.com", then the two Reference Identifiers are:

DNS-ID: dns.example.com

SRV-ID: _domain-s.dns.example.com

If either of the Reference Identifiers are found in the X.509 certificate's subjectAltName extension as described in section 6 of [RFC6125], the DNS client should accept the certificate for the server.

A compliant DNS client MUST only inspect the certificate's subjectAltName extension for these Reference Identifiers. In particular, it MUST NOT inspect the Subject field itself.

9.2. DANE

DANE [RFC6698] provides mechanisms to root certificate and raw public keys trust with DNSSEC. However this requires a domain name which must be obtained via a trusted source.

It is noted that [RFC6698] says

> "Clients that validate the DNSSEC signatures themselves MUST use standard DNSSEC validation procedures. Clients that rely on another entity to perform the DNSSEC signature validation MUST use a secure mechanism between themselves and the validator."

The specific DANE record would take the form:

_853._tcp.[server-domain-name] for TLS

_853._udp.[server-domain-name] for DTLS

9.2.1. Direct DNS Lookup

The DNS client MAY choose to perform the DNS lookups to retrieve the required DANE records itself. The DNS queries for such DANE records MAY use opportunistic encryption or be in the clear to avoid trust recursion. The records MUST be validated using DNSSEC as described above in [RFC6698].

9.2.2. TLS DNSSEC Chain extension

The DNS client MAY offer the TLS extension described in [I-D.shore-tls-dnssec-chain-extension]. If the DNS server supports this extension, it can provide the full chain to the client in the handshake.

If the DNS client offers the TLS DNSSEC Chain extension, it MUST be capable of validating the full DNSSEC authentication chain down to the leaf. If the supplied DNSSEC chain does not validate, the client MUST ignore the DNSSEC chain and validate only via other supplied credentials.

[TODO: specify guidance for DANE parameters to be used here. For example, a suggestion to use Certificate Usage of 3 (EE-DANE) (section 2.1.1 of [RFC6698]) and a Selector of 1 (SPKI) (section 2.1.2) would completely remove all X.509 and certificate authorities from the verification path and allows for private certification]

[TODO: discuss combination of DNSSEC Chain Extension with cert validation. Note that the combination depends on the Certificate Usage value of the TLSA response.]

10. Combined Credentials with SPKI Pinsets

The SPKI pinset profile described in [I-D.ietf-dprive-dns-over-tls] MAY be used with DNS-over-(D)TLS.

This draft does not make explicit recommendations about how a SPKI pinset based authentication mechanism should be combined with a domain based mechanism from an operator perspective. However it can be envisaged that a DNS server operator may wish to make both an SPKI pinset and a domain name available to allow clients to choose which mechanism to use. Therefore, the following is guidance on how clients ought to behave if they choose to configure both, as is possible in HPKP [RFC7469].

A DNS client that is configured with both a domain name and a SPKI pinset for a DNS sever SHOULD match on both a valid credential for the domain name and a valid SPKI pinset when connecting to that DNS server.

11. (D)TLS Protocol Profile

This section defines the (D)TLS protocol profile of DNS-over-(D)TLS.

There are known attacks on (D)TLS, such as machine-in-the-middle and protocol downgrade. These are general attacks on (D)TLS and not specific to DNS-over-TLS; please refer to the (D)TLS RFCs for discussion of these security issues. Clients and servers MUST adhere to the (D)TLS implementation recommendations and security considerations of [RFC7525] except with respect to (D)TLS version. Since encryption of DNS using (D)TLS is virtually a green-field deployment DNS clients and server MUST implement only (D)TLS 1.2 or later.

Implementations MUST NOT offer or provide TLS compression, since compression can leak significant amounts of information, especially to a network observer capable of forcing the user to do an arbitrary DNS lookup in the style of the CRIME attacks [CRIME].

Implementations compliant with this profile MUST implement all of the following items:

- TLS session resumption without server-side state [RFC5077] which eliminates the need for the server to retain cryptographic state for longer than necessary.

- Raw public keys [RFC7250] which reduce the size of the ServerHello, and can be used by servers that cannot obtain certificates (e.g., DNS servers on private networks).

Implementations compliant with this profile SHOULD implement all of the following items:

- TLS False Start [I-D.ietf-tls-falsestart] which reduces round-trips by allowing the TLS second flight of messages (ChangeCipherSpec) to also contain the (encrypted) DNS query

- Cached Information Extension [I-D.ietf-tls-cached-info] which avoids transmitting the server's certificate and certificate chain if the client has cached that information from a previous TLS handshake

[NOTE: The references to (works in progress) should be upgraded to MUST's if those references become RFC's prior to publication of this document.]

Guidance specific to TLS or DTLS is provided in either [I-D.ietf-dprive-dnsodtls] or [I-D.ietf-dprive-dns-over-tls].

12. IANA Considerations

This memo includes no request to IANA.

13. Security Considerations

Security considerations discussed in [RFC7525], [I-D.ietf-dprive-dnsodtls] and [I-D.ietf-dprive-dns-over-tls] apply to this document.

13.1. Counter-measures to DNS Traffic Analysis

This section makes suggestions for measures that can reduce the ability of attackers to infer information pertaining to encrypted client queries by other means (e.g. via an analysis of encrypted traffic size, or via monitoring of resolver to authoritative traffic).

DNS-over-(D)TLS clients and servers SHOULD consider implementing the following relevant DNS extensions

- EDNS(0) padding [I-D.ietf-dprive-edns0-padding], which allows encrypted queries and responses to hide their size.

DNS-over-(D)TLS clients SHOULD consider implementing the following relevant DNS extensions

- Privacy Election using Client Subnet in DNS Queries [I-D.ietf-dnsop-edns-client-subnet]. If a DNS client does not include an EDNS0 Client Subnet Option with a SOURCE PREFIX-LENGTH set to 0 in a query, the DNS server may potentially leak client address information to the upstream authoritative DNS servers. A DNS client ought to be able to inform the DNS Resolver that it does not want any address information leaked, and the DNS Resolver should honor that request.

14. Acknowledgements

Thanks to the authors of both [I-D.ietf-dprive-dnsodtls] and [I-D.ietf-dprive-dns-over-tls] for laying the ground work that this draft builds on and for reviewing the contents. The authors would also like to thank John Dickinson, Shumon Huque, Melinda Shore, Gowri Visweswaran, Ray Bellis, Stephane Bortzmeyer and Jinmei Tatuya for review and discussion of the ideas presented here.

15. References

15.1. 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, <http://www.rfc-editor.org/info/rfc2119>.

[RFC4985] Santesson, S., "Internet X.509 Public Key Infrastructure Subject Alternative Name for Expression of Service Name", RFC 4985, DOI 10.17487/RFC4985, August 2007, <http://www.rfc-editor.org/info/rfc4985>.

[RFC5077] Salowey, J., Zhou, H., Eronen, P., and H. Tschofenig, "Transport Layer Security (TLS) Session Resumption without Server-Side State", RFC 5077, DOI 10.17487/RFC5077, January 2008, <http://www.rfc-editor.org/info/rfc5077>.

[RFC5246] Dierks, T. and E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.2", RFC 5246, DOI 10.17487/RFC5246, August 2008, <http://www.rfc-editor.org/info/rfc5246>.

[RFC5280] Cooper, D., Santesson, S., Farrell, S., Boeyen, S., Housley, R., and W. Polk, "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile", RFC 5280, DOI 10.17487/RFC5280, May 2008, <http://www.rfc-editor.org/info/rfc5280>.

[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, <http://www.rfc-editor.org/info/rfc6125>.

[RFC6347] Rescorla, E. and N. Modadugu, "Datagram Transport Layer Security Version 1.2", RFC 6347, DOI 10.17487/RFC6347, January 2012, <http://www.rfc-editor.org/info/rfc6347>.

[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, <http://www.rfc-editor.org/info/rfc6698>.

[RFC7250] Wouters, P., Ed., Tschofenig, H., Ed., Gilmore, J., Weiler, S., and T. Kivinen, "Using Raw Public Keys in Transport Layer Security (TLS) and Datagram Transport Layer Security (DTLS)", RFC 7250, DOI 10.17487/RFC7250, June 2014, <http://www.rfc-editor.org/info/rfc7250>.

[RFC7525] Sheffer, Y., Holz, R., and P. Saint-Andre, "Recommendations for Secure Use of Transport Layer Security (TLS) and Datagram Transport Layer Security (DTLS)", BCP 195, RFC 7525, DOI 10.17487/RFC7525, May 2015, <http://www.rfc-editor.org/info/rfc7525>.

15.2. Informative References

[CRIME] Rizzo, J. and T. Duong, "The CRIME Attack", 2012.

[dnssec-trigger] NLnetLabs, "Dnssec-Trigger", May 2014, <https://www.nlnetlabs.nl/projects/dnssec-trigger/>.

[I-D.ietf-dnsop-edns-client-subnet]
Contavalli, C., Gaast, W., tale, t., and W. Kumari, "Client Subnet in DNS Queries", draft-ietf-dnsop-edns-client-subnet-06 (work in progress), December 2015.

[I-D.ietf-dprive-dns-over-tls]
Zi, Z., Zhu, L., Heidemann, J., Mankin, A., Wessels, D., and P. Hoffman, "Specification for DNS over TLS", draft-ietf-dprive-dns-over-tls-09 (work in progress), March 2016.

[I-D.ietf-dprive-dnsodtls]
Reddy, T., Wing, D., and P. Patil, "DNS over DTLS (DNSoD)", draft-ietf-dprive-dnsodtls-05 (work in progress), March 2016.

[I-D.ietf-dprive-edns0-padding]
Mayrhofer, A., "The EDNS(0) Padding Option", draft-ietf-dprive-edns0-padding-03 (work in progress), March 2016.

[I-D.ietf-tls-cached-info]
Santesson, S. and H. Tschofenig, "Transport Layer Security (TLS) Cached Information Extension", draft-ietf-tls-cached-info-22 (work in progress), January 2016.

[I-D.ietf-tls-falsestart]
Langley, A., Modadugu, N., and B. Moeller, "Transport Layer Security (TLS) False Start", draft-ietf-tls-falsestart-01 (work in progress), November 2015.

[I-D.shore-tls-dnssec-chain-extension]
Shore, M., Barnes, R., Huque, S., and W. Toorop, "A DANE Record and DNSSEC Authentication Chain Extension for TLS", draft-shore-tls-dnssec-chain-extension-02 (work in progress), October 2015.

[RFC2131] Droms, R., "Dynamic Host Configuration Protocol", RFC 2131, DOI 10.17487/RFC2131, March 1997, <http://www.rfc-editor.org/info/rfc2131>.

[RFC2132] Alexander, S. and R. Droms, "DHCP Options and BOOTP Vendor Extensions", RFC 2132, DOI 10.17487/RFC2132, March 1997, <http://www.rfc-editor.org/info/rfc2132>.

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

[RFC7435] Dukhovni, V., "Opportunistic Security: Some Protection Most of the Time", RFC 7435, DOI 10.17487/RFC7435, December 2014, <http://www.rfc-editor.org/info/rfc7435>.

[RFC7469] Evans, C., Palmer, C., and R. Sleevi, "Public Key Pinning Extension for HTTP", RFC 7469, DOI 10.17487/RFC7469, April 2015, <http://www.rfc-editor.org/info/rfc7469>.

[RFC7626] Bortzmeyer, S., "DNS Privacy Considerations", RFC 7626, DOI 10.17487/RFC7626, August 2015, <http://www.rfc-editor.org/info/rfc7626>.

Appendix A. Server capability probing and caching by DNS clients

This section presents a non-normative discussion of how DNS clients might probe for and cache privacy capabilities of DNS servers.

Deployment of both DNS-over-TLS and DNS-over-DTLS will be gradual. Not all servers will support one or both of these protocols and the well-known port might be blocked by some middleboxes. Clients will be expected to keep track of servers that support DNS-over-TLS and/or DNS-over-DTLS, and those that have been previously authenticated.

If no server capability information is available then (unless otherwise specified by the configuration of the DNS client) DNS clients that implement both TLS and DTLS should try to authenticate using both protocols before failing or falling back to a lower security. DNS clients using opportunistic security should try all available servers (possibly in parallel) in order to obtain an authenticated encrypted connection before falling back to a lower security. (RATIONALE: This approach can increase latency while discovering server capabilities but maximizes the chance of sending the query over an authenticated encrypted connection.)

Appendix B. Changes between revisions

[Note to RFC Editor: please remove this section prior to publication.]

B.1. -01 version

Section 4.2: Make clear that the Strict Privacy Profile can include meta queries performed using Opportunistic Privacy.

Section 4.2, Table 1: Update to clarify that Opportunistic Privacy does not guarantee protection against passive attack.

Section 4.2: Add sentence discussing client/provider trusted relationships.

Section 5: Add more discussion of detection of active attacks when using Opportunistic Privacy.

Section 8.2: Clarify description and example.

B.2. draft-ietf-dprive-dtls-and-tls-profiles-00

Re-submission of draft-dgr-dprive-dtls-and-tls-profiles with name change to draft-ietf-dprive-dtls-and-tls-profiles. Also minor nits fixed.

Authors' Addresses

Sara Dickinson
Sinodun Internet Technologies
Magdalen Centre
Oxford Science Park
Oxford OX4 4GA
UK

Email: sara@sinodun.com
URI: http://sinodun.com

Daniel Kahn Gillmor
ACLU
125 Broad Street, 18th Floor
New York NY 10004
USA

Email: dkg@fifthhorseman.net

Tirumaleswar Reddy
Cisco Systems, Inc.
Cessna Business Park, Varthur Hobli
Sarjapur Marathalli Outer Ring Road
Bangalore, Karnataka 560103
India

Email: tireddy@cisco.com