

dprive
Internet-Draft
Updates: [7858](#) (if approved)
Intended status: Standards Track
Expires: October 12, 2017

S. Dickinson
Sinodun
D. Gillmor
ACLU
T. Reddy
Cisco
April 10, 2017

**Authentication and (D)TLS Profile for DNS-over-(D)TLS
draft-ietf-dprive-dtls-and-tls-profiles-09**

Abstract

This document discusses Usage Profiles, based on one or more authentication mechanisms, which can be used for DNS over Transport Layer Security (TLS) or Datagram TLS (DTLS). This document also specifies new authentication mechanisms - it describes several ways a DNS client can use an authentication domain name to authenticate a DNS server. Additionally, it defines (D)TLS profiles for DNS clients and servers implementing DNS-over-(D)TLS.

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 October 12, 2017.

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 (<http://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	3
2.	Terminology	4
3.	Scope	6
4.	Discussion	6
5.	Usage Profiles	7
5.1.	DNS Resolution	9
6.	Authentication in DNS-over(D)TLS	9
6.1.	DNS-over-(D)TLS Startup Configuration Problems	9
6.2.	Credential Verification	10
6.3.	Summary of Authentication Mechanisms	10
6.4.	Combining Authentication Mechanisms	13
6.5.	Authentication in Opportunistic Privacy	13
6.6.	Authentication in Strict Privacy	13
6.7.	Implementation guidance	14
7.	Sources of Authentication Domain Names	14
7.1.	Full direct configuration	14
7.2.	Direct configuration of name only	14
7.3.	DHCP	15
8.	Authentication Domain Name based Credential Verification	15
8.1.	PKIX Certificate Based Authentication	15
8.2.	DANE	16
8.2.1.	Direct DNS Lookup	17
8.2.2.	TLS DNSSEC Chain extension	17
9.	(D)TLS Protocol Profile	17
10.	IANA Considerations	18
11.	Security Considerations	18
11.1.	Counter-measures to DNS Traffic Analysis	18
12.	Acknowledgements	19
13.	References	19
13.1.	Normative References	19
13.2.	Informative References	21
Appendix A.	Server capability probing and caching by DNS clients	22
Appendix B.	Changes between revisions	22
B.1.	-09 version	22
B.2.	-08 version	23
B.3.	-07 version	23
B.4.	-06 version	23
B.5.	-05 version	24
B.6.	-04 version	24
B.7.	-03 version	24

B.8.	-02 version	24
B.9.	-01 version	24
B.10.	draft-ietf-dprive-dtls-and-tls-profiles-00	25
Authors' Addresses	25

1. Introduction

DNS Privacy issues are discussed in [[RFC7626](#)]. Two documents that provide DNS privacy between DNS clients and DNS servers are:

- o Specification for DNS over Transport Layer Security (TLS) [[RFC7858](#)], referred to here as simply 'DNS-over-TLS'
- o DNS-over-DTLS (DNSoD) [[I-D.ietf-dprive-dnsodtls](#)], referred to here simply as 'DNS-over-DTLS'. Note that this document has the Intended status of Experimental.

Both documents are limited in scope to encrypting DNS messages between stub clients and recursive resolvers and the same scope is applied to this document (see [Section 2](#) and [Section 3](#)). The proposals here might be adapted or extended in future to be used for recursive clients and authoritative servers, but this application was out of scope for the Working Group charter at the time this document was finished.

This document specifies two Usage Profiles (Strict and Opportunistic) for DTLS [[RFC6347](#)] and TLS [[RFC5246](#)] which define the security properties a user should expect when using that profile to connect to the available DNS servers. [Section 5](#) presents a generalised discussion of Usage Profiles by separating the Usage Profile, which is based purely on the security properties it offers the user, from the specific mechanism(s) that are used for authentication. The Profiles described are:

- o A Strict Profile that requires an encrypted connection and successful authentication of the DNS server which provides strong privacy guarantees (at the expense of providing no DNS service if this is not available).
- o An Opportunistic Profile that will attempt, but does not require, encryption and successful authentication; it therefore provides no privacy guarantees but offers maximum chance of DNS service.

The above Usage Profiles attempt authentication of the server using at least one authentication mechanism. [Section 6.4](#) discusses how to combine authentication mechanisms to determine the overall authentication result. Depending on that overall authentication

result (and whether encryption is available) the Usage Profile will determine if the connection should proceed, fallback or fail.

One authentication mechanism is already described in [[RFC7858](#)]. That document specifies an SPKI based authentication mechanism for DNS-over-TLS in the context of a specific case of a Strict Usage Profile using that single authentication mechanism. Therefore the "Out-of-band Key-pinned Privacy Profile" described in [[RFC7858](#)] would qualify as a "Strict Usage Profile" that used SPKI pinning for authentication.

This document extends the use of SPKI pinset based authentication so that it is considered a general authentication mechanism that can be used with either DNS-over-(D)TLS Usage Profile. That is, the SPKI pinset mechanism described in [[RFC7858](#)] MAY be used with DNS-over-(D)TLS.

This document also describes a number of additional authentication mechanisms all of which specify how a DNS client should authenticate a DNS server based on an 'authentication domain name'. In particular, the following is described:

- o How a DNS client can obtain the combination of an authentication domain name and IP address for a DNS server. See [Section 7](#).
- o What are the acceptable credentials a DNS server can present to prove its identity for (D)TLS authentication based on a given authentication domain name. See [Section 8](#).
- o How a DNS client can verify that any given credential matches the authentication domain name obtained for a DNS server. See [Section 8](#).

In [Section 9](#) this document 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 all currently specified authentication mechanisms.

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:

- o DNS client: a DNS stub resolver or forwarder. In the case of a forwarder, the term "DNS client" is used to discuss the side that sends queries.
- o DNS server: a DNS recursive resolver or forwarder. In the case of a forwarder, the term "DNS server" is used to discuss the side that responds to queries. For emphasis, in this document the term does not apply to authoritative servers.
- o Privacy-enabling DNS server: A DNS server that:
 - * MUST implement DNS-over-TLS [[RFC7858](#)] and MAY implement DNS-over-DTLS [[I-D.ietf-dprive-dnsodtls](#)].
 - * Can offer at least one of the credentials described in [Section 8](#).
 - * Implements the (D)TLS profile described in [Section 9](#).
- o (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.
- o DNS-over-(D)TLS: For brevity this term is used for statements that apply to both DNS-over-TLS [[RFC7858](#)] and DNS-over-DTLS [[I-D.ietf-dprive-dnsodtls](#)]. Specific terms will be used for any statement that applies to either protocol alone.
- o Authentication domain name: A domain name that can be used to authenticate a DNS Privacy enabling server. Sources of authentication domain names are discussed in [Section 7](#).
- o SPKI Pinsets: [[RFC7858](#)] 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.
- o Authentication information: Information a DNS client may use as the basis of an authentication mechanism. In this context that can be either a:
 - * a SPKI pinset or
 - * an authentication domain name

- o Reference Identifier: a Reference Identifier as described in [[RFC6125](#)], constructed by the DNS client when performing TLS authentication of a DNS server.
 - o Credential: Information available for a DNS server which proves its identity for authentication purposes. Credentials discussed here include:
 - * PKIX certificate
 - * DNSSEC validated chain to a TLSA record
- but may also include SPKI pinsets.

3. Scope

This document is limited to describing

- o Usage Profiles based on general authentication mechanisms
- o The details of domain-name-based authentication of DNS servers by DNS clients (as defined in the terminology section)
- o The (D)TLS profiles needed to support authentication in DNS-over-(D)TLS.

As such, the following things are out of scope:

- o Authentication of authoritative servers by recursive resolvers.
- o Authentication of DNS clients by DNS servers.
- o The details of how to perform SPKI-pinset-based authentication. This is defined in [[RFC7858](#)].
- o Any server identifier other than domain names, including IP address, organizational name, country of origin, etc.

4. Discussion

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.

This draft discusses Usage Profiles, which provide differing levels of privacy guarantees to DNS clients, based on the requirements for authentication and encryption, regardless of the context (for example, which network the client is connected to). A Usage Profile is a distinct concept to a usage policy or usage model, which might dictate which Profile should be used in a particular context (enterprise vs coffee shop), with a particular set of DNS Servers or with reference to other external factors. A description of the variety of usage policies is out of scope of this document, but may be the subject of future work.

5. Usage Profiles

A DNS client has a choice of privacy Usage Profiles available. This choice is briefly discussed in both [\[RFC7858\]](#) and [\[I-D.ietf-dprive-dnsodtls\]](#). These Usage Profiles are:

- o 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 authentication 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 Profile is discussed in detail in [Section 6.6](#).
- o 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."

The use of Opportunistic Privacy is intended to support incremental deployment of security capabilities with a view to widespread adoption of Strict Privacy. It should be employed when the DNS client might otherwise settle for cleartext; it provides the maximum protection available. As described in [\[RFC7435\]](#) it might result in

- * an encrypted and authenticated connection
- * an encrypted connection

- * a clear text connection

depending on the fallback logic of the client, the available authentication information and the capabilities of the DNS Server. In all these cases the DNS client is willing to continue with a connection to the DNS Server and perform resolution of queries.

To compare the two Usage profiles the table below shows successful Strict Privacy along side the 3 possible outcomes of Opportunistic Privacy. In the best case scenario for Opportunistic Privacy (an authenticated and encrypted connection) it is equivalent to Strict Privacy. In the worst case scenario it is equivalent to clear text. Clients using Opportunistic Privacy SHOULD try for the best case but MAY fallback to the intermediate case and eventually the worst case scenario in order to obtain a response. It therefore provides no privacy guarantee to the user and varying protection depending on what kind of connection is actually used.

Note that there is no requirement in Opportunistic Security to notify the user what type of connection is actually used, the 'detection' described below is only possible if such connection information is available. However, if it is available and the user is informed that an unencrypted connection was used to connect to a server then the user should assume (detect) that the connection is subject to both active and passive attack since the DNS queries are sent in clear text. This might be particularly useful if a new connection to a certain server is unencrypted when all previous connections were encrypted. Similarly if the user is informed that an encrypted but unauthenticated connection was used then the user can detect that the connection may be subject to active attack. This is discussed in [Section 6.5](#).

Usage Profile	Connection	Passive Attacker	Active Attacker
Strict	A, E	P	P
Opportunistic	A, E	P	P
Opportunistic	E	P	N, D
Opportunistic		N, D	N, D

P == protection; N == no protection; D == detection is possible; A == Authenticated Connection; E == Encrypted Connection

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

Strict Privacy provides the strongest privacy guarantees and therefore SHOULD always be implemented in DNS clients along with Opportunistic Privacy.

A DNS client that implements DNS-over-(D)TLS SHOULD NOT default to the use of clear text (no privacy).

The choice between the two profiles depends on a number of factors including which is more important to the particular client:

- o DNS service at the cost of no privacy guarantee (Opportunistic) or
- o guaranteed privacy at the potential cost of no DNS service (Strict).

Additionally the two profiles require varying levels of configuration (or a trusted relationship with a provider) and DNS server capabilities, therefore DNS clients will need to carefully select which profile to use based on their communication privacy needs.

A DNS server that implements DNS-over-(D)TLS SHOULD provide at least one credential so that those DNS clients that wish to do so are able to use Strict Privacy (see [Section 2](#)).

[5.1.](#) DNS Resolution

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.

[6.](#) Authentication in DNS-over(D)TLS

This section describes authentication mechanisms and how they can be used in either Strict or Opportunistic Privacy for DNS-over-(D)TLS.

[6.1.](#) DNS-over-(D)TLS Startup Configuration Problems

Many (D)TLS clients use PKIX authentication [[RFC6125](#)] based on an authentication 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. Such a DNS client therefore has a bootstrap problem, as it will typically only know the IP address of its DNS server.

In this case, before connecting to a DNS server, a DNS client needs to learn the authentication domain name it should associate with the

IP address of a DNS server for authentication purposes. Sources of authentication domain names are discussed in [Section 7](#).

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

[6.2.](#) Credential Verification

The use of SPKI pinset verification is discussed in [[RFC7858](#)].

In terms of domain name based verification, once an authentication domain name is known for a DNS server a choice of authentication mechanisms can be used for credential verification. [Section 8](#) discusses these mechanisms in detail, namely PKIX 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 8.2](#).

[6.3.](#) Summary of Authentication Mechanisms

This section provides an overview of the various authentication mechanisms. The table below indicates how the DNS client obtains information to use for authentication for each option; either statically via direct configuration or dynamically. Of course, the Opportunistic Usage Profile does not require authentication and so a client using that profile may choose to connect to a Privacy Enabling DNS server on the basis of just an IP address.

#	Static Config	Dynamically Obtained	Short name: Description
1	SPKI + IP		SPKI: SPKI pinset(s) and IP address obtained out of band [RFC7858]
2	ADN + IP		ADN: ADN and IP address obtained out of band (see Section 7.1)
3	ADN	IP	ADN only: Opportunistic lookups to a NP DNS server for A/AAAA (see Section 7.2)
4		ADN + IP	DHCP: DHCP configuration only (see Section 7.3)
5	[ADN + IP]	[ADN + IP] TLSA record	DANE: DNSSEC chain obtained via Opportunistic lookups to NP DNS server (see Section 8.2.1)
6	[ADN + IP]	[ADN + IP] TLSA record	TLS extension: DNSSEC chain provided by PE DNS server in TLS DNSSEC chain extension (see Section 8.2.2)

SPKI == SPKI pinset(s), IP == IP Address, ADN == Authentication Domain Name, NP == Network provided, PE == Privacy-enabling, [] == Data may be obtained either statically or dynamically

Table 2: Overview of Authentication Mechanisms

The following summary attempts to present some key attributes of each of the mechanisms (using the 'Short name' from Table 2), indicating attractive attributes with a '+' and undesirable attributes with a '-'.

1. SPKI

+ Minimal leakage (Note that the ADN is always leaked in the SNI field in the Client Hello in TLS when communicating with a privacy enabling DNS server)

- Overhead of on-going key management required

2. ADN

- + Minimal leakage
- + One-off direct configuration only

3. ADN only

- + Minimal one-off direct configuration, only a human recognisable domain name needed
- A/AAAA meta queries leaked to network provided DNS server

4. DHCP

- + No static config
- Requires a non-standard or future DHCP option in order to provide the ADN
- Requires secure and trustworthy connection to DHCP server

5. DANE

The ADN and/or IP may be obtained statically or dynamically and the relevant attributes of that method apply

- + DANE options (e.g. matching on entire certificate)
- Requires a DNSSEC validating stub implementation (deployment of which is limited at the time of writing)
- DNSSEC chain meta queries leaked to network provided DNS server

6. TLS extension

The ADN and/or IP may be obtained statically or dynamically and the relevant attributes of that method apply

- + Reduced latency compared with 'DANE'
- + No network provided DNS server required if ADN and IP statically configured
- + DANE options (e.g. matching on entire certificate)
- Requires a DNSSEC validating stub implementation

6.4. Combining Authentication Mechanisms

This draft does not make explicit recommendations about how an 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 an authentication 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 an authentication domain name and a SPKI pinset for a DNS server SHOULD match on both a valid credential for the authentication domain name and a valid SPKI pinset (if both are available) when connecting to that DNS server. The overall authentication result should only be considered successful if both authentication mechanisms are successful.

6.5. Authentication in Opportunistic Privacy

An Opportunistic Security [[RFC7435](#)] profile is described in [[RFC7858](#)] which MAY be used for DNS-over-(D)TLS.

DNS clients issuing queries under an opportunistic profile which know authentication information 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.

6.6. Authentication in Strict Privacy

To authenticate a privacy-enabling DNS server, a DNS client needs to know authentication information 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 7](#) to obtain an authentication domain name

for the server it contacts, or use an SPKI pinset as described in [\[RFC7858\]](#).

A DNS client requiring Strict Privacy MUST only attempt to connect to DNS servers for which at least one piece of authentication information is known. The client MUST use the available verification mechanisms described in [Section 8](#) 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.

[6.7.](#) Implementation guidance

[Section 9](#) describes the (D)TLS profile for DNS-over(D)TLS. Additional considerations relating to general implementation guidelines are discussed in both [Section 11](#) and in [Appendix A](#).

[7.](#) Sources of Authentication Domain Names

[7.1.](#) Full direct configuration

DNS clients may be directly and securely provisioned with the authentication 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.

[7.2.](#) Direct configuration of name only

A DNS client may be configured directly and securely with only the authentication 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 an untrusted recursive DNS resolver to establish the IP address of the intended privacy-enabling DNS resolver by doing a lookup of A/AAAA records. Private DNS resolution can now be done by the DNS client against the pre-configured privacy-enabling DNS resolver, using the IP address gathered from the untrusted DNS resolver.

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

[7.3.](#) 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.

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

Use of such a mechanism with any DHCP server when using an Opportunistic profile is reasonable, given the security expectation of that profile. However when using a Strict profile the DHCP servers used as sources of authentication domain names MUST be considered secure and trustworthy. This document does not attempt to describe secured and trusted relationships to DHCP servers.

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.

[8.](#) Authentication Domain Name based Credential Verification

[8.1.](#) PKIX Certificate Based Authentication

When a DNS client configured with an authentication domain name connects to its configured DNS server over (D)TLS, the server may present it with a PKIX 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 one Reference Identifier for the server based on the authentication domain name: A DNS-ID which is simply the authentication domain name itself.

If the Reference Identifier is found in the PKIX 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 the Reference Identifier. In particular, it MUST NOT inspect the Subject field itself.

8.2. DANE

DANE [[RFC6698](#)] provides mechanisms to anchor certificate and raw public key trust with DNSSEC. However this requires the DNS client to have an authentication domain name for the DNS Privacy Server which must be obtained via a trusted source.

This section assumes a solid understanding of both DANE [[RFC6698](#)] and DANE Operations [[RFC7671](#)]. A few pertinent issues covered in these documents are outlined here as useful pointers, but familiarity with both these documents in their entirety is expected.

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."

It is noted that [[RFC7671](#)] covers the following topics:

- o [Section 4.1](#): Opportunistic Security and PKIX Usages and [Section 14](#): Security Considerations, which both discuss the use of PKIX-TA(0) and PKIX-EE(1) for Opportunistic Security.
- o [Section 5](#): Certificate-Usage-Specific DANE Updates and Guidelines. Specifically [Section 5.1](#) which outlines the combination of Certificate Usage DANE-EE(3) and Selector Usage SPKI(1) with Raw Public Keys [[RFC7250](#)]. [Section 5.1](#) also discusses the security implications of this mode, for example, it discusses key lifetimes and specifies that validity period enforcement is based solely on the TLSA RRset properties for this case.
- o [Section 13](#): Operational Considerations, which discusses TLSA TTLS and signature validity periods.

The specific DANE record for a DNS Privacy Server would take the form:

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

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

8.2.1. 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](#)].

8.2.2. TLS DNSSEC Chain extension

The DNS client MAY offer the TLS extension described in [[I-D.ietf-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.

9. (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 a green-field deployment DNS clients and servers 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:

- o TLS session resumption without server-side state [[RFC5077](#)] which eliminates the need for the server to retain cryptographic state for longer than necessary (This statement updates [[RFC7858](#)]).
- o 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:

- o TLS False Start [[RFC7918](#)] which reduces round-trips by allowing the TLS second flight of messages (ChangeCipherSpec) to also contain the (encrypted) DNS query
- o Cached Information Extension [[RFC7924](#)] which avoids transmitting the server's certificate and certificate chain if the client has cached that information from a previous TLS handshake

Guidance specific to TLS is provided in [[RFC7858](#)] and that specific to DTLS it is provided in[I-D.ietf-dprive-dnsodtls].

10. IANA Considerations

This memo includes no request to IANA.

11. Security Considerations

Security considerations discussed in [[RFC7525](#)], [[I-D.ietf-dprive-dnsodtls](#)] and [[RFC7858](#)] apply to this document.

DNS Clients SHOULD consider implementing support for the mechanisms described in [Section 8.2](#) and offering a configuration option which limits authentication to using only those mechanisms (i.e. with no fallback to pure PKIX based authentication) such that authenticating solely via the PKIX infrastructure can be avoided.

11.1. 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

- o EDNS(0) padding [[RFC7830](#)], which allows encrypted queries and responses to hide their size.

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

- o Privacy Election using Client Subnet in DNS Queries [[RFC7871](#)]. 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.

12. Acknowledgements

Thanks to the authors of both [[I-D.ietf-dprive-dnsodtls](#)] and [[RFC7858](#)] 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, Jinmei Tatuya, Paul Hoffman, Christian Huitema and John Levine for review and discussion of the ideas presented here.

13. References

13.1. 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>>.
- [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>>.
- [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, <<http://www.rfc-editor.org/info/rfc7671>>.
- [RFC7830] Mayrhofer, A., "The EDNS(0) Padding Option", [RFC 7830](#), DOI 10.17487/RFC7830, May 2016, <<http://www.rfc-editor.org/info/rfc7830>>.
- [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, <<http://www.rfc-editor.org/info/rfc7858>>.

13.2. 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-dprive-dnsodtls]
Reddy, T., Wing, D., and P. Patil, "Specification for DNS over Datagram Transport Layer Security (DTLS)", [draft-ietf-dprive-dnsodtls-15](#) (work in progress), December 2016.
- [I-D.ietf-tls-dnssec-chain-extension]
Shore, M., Barnes, R., Huque, S., and W. Toorop, "A DANE Record and DNSSEC Authentication Chain Extension for TLS", [draft-ietf-tls-dnssec-chain-extension-03](#) (work in progress), March 2017.
- [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>>.
- [RFC7871] Contavalli, C., van der Gaast, W., Lawrence, D., and W. Kumari, "Client Subnet in DNS Queries", [RFC 7871](#), DOI 10.17487/RFC7871, May 2016,
<<http://www.rfc-editor.org/info/rfc7871>>.

- [RFC7918] Langley, A., Modadugu, N., and B. Moeller, "Transport Layer Security (TLS) False Start", [RFC 7918](#), DOI 10.17487/RFC7918, August 2016, <<http://www.rfc-editor.org/info/rfc7918>>.
- [RFC7924] Santesson, S. and H. Tschofenig, "Transport Layer Security (TLS) Cached Information Extension", [RFC 7924](#), DOI 10.17487/RFC7924, July 2016, <<http://www.rfc-editor.org/info/rfc7924>>.

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. -09 version

Remove the SRV record to simplify the draft.

Add suggestion that clients offer option to avoid using only PKIX authentication.

Clarify that the MUST on implementing TLS session resumption updates [RFC7858](#).

Update page header to be '(D)TLS Authentication for TLS'.

B.2. -08 version

Removed hard failure as an option for Opportunistic Usage profile.

Added a new section comparing the Authentication Mechanisms

B.3. -07 version

Re-work of the Abstract and Introduction to better describe the contents in this version.

Terminology: New definition of 'authentication information'.

Scope: Changes to the Scope section.

Moved discussion of combining authentication mechanism earlier.

Changes to the section headings and groupings to make the presentation more logical.

B.4. -06 version

Introduction: Re-word discussion of Working group charter.

Introduction: Re-word first and third bullet point about 'obtaining' a domain name and IP address.

Introduction: Update reference to DNS-over-TLS draft.

Terminology: Change forwarder/proxy to just forwarder

Terminology: Add definition of 'Authentication domain name' and use this throughout

[Section 4.2](#): Remove parenthesis in the table.

[Section 4.2](#): Change the text after the table as agreed with Paul Hoffman.

[Section 4.3.1](#): Change title and remove brackets around last statement.

[Section 11](#): Split second paragraph.

B.5. -05 version

Add more details on detecting passive attacks to [section 4.2](#)

Changed X.509 to PKIX throughout

Change comment about future I-D on usage policies.

B.6. -04 version

Introduction: Add comment that DNS-over-DTLS draft is Experiments

Update 2 I-D references to RFCs.

B.7. -03 version

[Section 9](#): Update DANE section with better references to [RFC7671](#) and [RFC7250](#)

B.8. -02 version

Introduction: Added paragraph on the background and scope of the document.

Introduction and Discussion: Added more information on what a Usage profiles is (and is not) the the two presented here.

Introduction: Added paragraph to make a comparison with the Strict profile in [RFC7858](#) clearer.

[Section 4.2](#): Re-worked the description of Opportunistic and the table.

[Section 8.3](#): Clarified statement about use of DHCP in Opportunistic profile

Title abbreviated.

B.9. -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.10. 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: tiredy@cisco.com

