Workgroup: Internet Engineering Task Force
Internet-Draft:
draft-huque-tls-dane-clientid-02
Published: 8 October 2020
Intended Status: Standards Track
Expires: 11 April 2021
Authors: S. Huque Salesforce
V. Dukhovni Two Sigma

TLS Extension for DANE Client Identity

Abstract

This document specifies a TLS and DTLS extension to convey a DNS-Based Authentication of Named Entities (DANE) Client Identity to a TLS or DTLS server. This is useful for applications that perform TLS client authentication via DANE TLSA records.

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 11 April 2021.

Copyright Notice

Table of Contents

1. Introduction

This document specifies a Transport Layer Security (TLS) extension [RFC6066] to convey a DANE [RFC6698] Client Identity to the TLS server. This is useful for applications that perform TLS client authentication via DANE TLSA records, as described in [DANECLIENT]. The extension could be empty to indicate to the server that the client has a DANE record and that the server can perform DANE authentication of the client with the identity extracted from the client certificate. Or the extension can contain the full client identity, in the form of the DNS domain name that is expected to have a DANE TLSA record published for it.

This extension supports both TLS [RFC5246] [RFC8446] and DTLS [RFC6347], and the term TLS in this document is used generically to describe both protocols.

2. Overview

When TLS clients use X.509 client certificates or raw public keys that are authenticated via DANE TLSA records, it is useful for them to convey their intent to be authenticated via DANE, or even to convey their complete DANE identity to the server. The TLS extension defined in this document is used to accomplish this.

In the case of X.509 client certificates, a TLS server can learn the client's identity by examining subject alternative names included in the certificate itself. However, without a mechanism such as the one defined in this extension, the TLS server cannot know apriori that the client has a published TLSA record, and thus may unnecessarily issue DNS queries for DANE TLSA records in-band with the TLS handshake even in cases where the client has no TLSA record associated with it. When multiple identities are present in the certificate, a client can use this extension to specify exactly which one the server should use. An additional situation in which this extension helps is where some TLS servers may need to selectively prompt for client certificate credentials only for clients that are equipped to provide certificates.

When TLS raw public keys [RFC7250] are being used to authenticate the client, the client uses this extension to explicitly indicate to the server what its domain name identity is (since there is no X.509 certificate from which the identity can be extracted).

Detailed protocol behavior of TLS clients and servers is described in [DANECLIENT].

3. DANE Client Identity Extension

The DANE Client Identity Extension type, "dane_clientid", will have a value assigned and registered in the IANA TLS Extensions registry. Its format is similar to the TLS Server Name Indication extension.

A TLS client implementing this specification SHOULD send an extension of type "dane_clientid" in the ClientHello handshake message to TLS servers it intends to perform DANE client authentication with.

If the client only needs to indicate that it has a DANE record and that the client's domain name identity can be obtained from its certificate, then the extension sent can be empty.

If the client additionally needs to send its domain name identity, then the "extension_data" field of the extension MUST contain a "ClientNameList" data structure defined in the following format:

```
struct {
    NameType name_type;
    select (name_type) {
        case host_name: HostName;
    } name;
} ClientName;

enum {
    host_name(0), (255)
} NameType;

opaque HostName<1..2^16-1>;

struct {
    ClientName client_name_list<1..2^16-1>
} ClientNameList;
```

The opaque HostName field contains the domain name of the client in textual presentation format, as described in RFC 1035 [RFC1035]. The ClientNameList MUST NOT contain more than one name of the same name_type. Currently only one "host_name" type is defined.

A TLS server implementing this specification MAY send back an empty extension of type "dane_clientid" in its SeverHello handshake message to indicate that it understands the extension and intends to perform DANE client authentication. (Is there a compelling reason to do this?)

4. Security Considerations

This extension is sent in the first flight of the TLS client's network data (Client Hello), which is in clear text.

To prevent unnecessary privacy leakage of the client's name in cleartext, a TLS client implementing this specification should be configured to only send this extension to TLS servers it intends to perform client authentication with.

Ideally, this extension should be used with the proposed TLS Encrypted Client Hello extension [ECH], which encrypts the entire Client Hello message. This will prevent leakage of the hostname, if included in the extension, in clear text.

5. IANA Considerations

This extension requires the registration of a new value in the TLS ExtensionsType registry.

6. Normative References

[DANECLIENT] Huque, S. and V. Dukhovni, "TLS Client Authentication via DANE TLSA Records", <https://tools.ietf.org/html/draft-huque-dane-client-cert>.

[ECH] Rescorla, E., Oku, K., Sullivan, N., and C.A. Wood, "TLS Encrypted Client Hello", <https://tools.ietf.org/html/draft-ietf-tls-esni>.

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

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

[RFC6066] Eastlake 3rd, D., "Transport Layer Security (TLS) Extensions: Extension Definitions", RFC 6066, DOI 10.17487/RFC6066, January 2011, <https://www.rfc-editor.org/info/rfc6066>.

[RFC6347] Rescorla, E. and N. Modadugu, "Datagram Transport Layer Security Version 1.2", RFC 6347, DOI 10.17487/RFC6347, January 2012, <https://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, <https://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, <https://www.rfc-editor.org/info/rfc7250>.

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

Authors' Addresses

Shumon Huque
Salesforce

Email: shuque@gmail.com

Viktor Dukhovni
Two Sigma

Email: ietf-dane@dukhovni.org