

Internet Engineering Task Force
Internet-Draft
Updates: [6698](#), 7671 (if approved)
Intended status: Standards Track
Expires: July 08, 2016

S. Huque
Verisign Labs
D. James
Verisign, Inc.
V. Dukhovni
Two Sigma
January 05, 2016

TLS Client Authentication via DANE TLSA records
draft-huque-dane-client-cert-02

Abstract

The DANE TLSA protocol [[RFC6698](#)] [[RFC7671](#)] describes how to publish Transport Layer Security (TLS) server certificates or public keys in the DNS. This document updates [RFC 6698](#) and [RFC 7671](#). It describes how to additionally use the TLSA record to publish client certificates or public keys, and also the rules and considerations for using them with 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 July 08, 2016.

Copyright Notice

Copyright (c) 2016 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 and Motivation	2
2.	Associating Client Identities in TLSA Records	2
3.	Authentication Model	3
4.	Client Identifiers in X.509 certificates	3
5.	Signaling the Client's DANE Identity in TLS	4
6.	Example TLSA records for clients	4
7.	Changes to Client and Server behavior	5
8.	Raw Public Keys	7
9.	Acknowledgements	7
10.	IANA Considerations	7
11.	Security Considerations	7
12.	References	7
12.1.	Normative References	7
12.2.	Informative References	8
	Authors' Addresses	9

[1.](#) Introduction and Motivation

The Transport Layer Security (TLS) protocol [[RFC5246](#)] optionally supports the authentication of clients using X.509 certificates [[RFC5280](#)] or raw public keys [[RFC7250](#)]. TLS applications that perform DANE authentication of servers using TLSA records may also desire to authenticate clients using the same mechanism, especially if the client identity is in the form of or can be represented by a DNS domain name. Some design patterns from the Internet of Things (IoT) make use of this form of authentication, where large networks of physical objects identified by DNS names may authenticate themselves using TLS to centralized device management and control platforms.

In this document, the term TLS is used generically to describe both the TLS and DTLS (Datagram Transport Layer Security) [[RFC6347](#)] protocols.

[2.](#) Associating Client Identities in TLSA Records

When specifying client identities (i.e. client domain names) in TLSA records, the owner name of the TLSA record has the following format:

`_service.[client-domain-name]`

The first label identifies the application service name. The remaining labels are composed of the client domain name.

Encoding the application service name into the owner name allows the same client domain name to have different authentication credentials for different application services. There is no need to encode the transport label - the same name form is usable with both TLS and DTLS.

The `_service` label could be a custom string for an application, but more commonly is expected to be a service name registered in the IANA Service Name Registry [[SRVREG](#)].

The RDATA or data field portion of the TLSA record is formed exactly as specified in [RFC 6698](#) and [RFC 7671](#), and carries the same meaning.

3. Authentication Model

The authentication model assumed in this document is the following:

The client is assigned an identity corresponding to a DNS domain name. This domain name doesn't necessarily have any relation to its network layer addresses. Clients often have dynamic or unpredictable addresses, and may move around the network, so tying their identity to network addresses is not feasible or wise in the general case.

The client generates (or has generated for it) a private and public key pair. Where client certificates are being used, the client also has a certificate binding the name to its public key. The certificate or public key has a corresponding TLSA record published in the DNS, which allows it to be authenticated directly via the DNS (using the DANE-TA or DANE-EE certificate usage modes) or via a PKIX public CA system constraint (using the PKIX-TA or PKIX-EE certificate usage modes).

4. Client Identifiers in X.509 certificates

If the TLS DANE Client Identity extension is not being used, the client certificate MUST have the client's DNS name specified in the Subject Alternative Name extension's `dNSName` type. Or, if an application specific identity is preferred or needed, the SRV-ID (PKIX `OtherName SRVName`) MUST be used to specify the application service and the client's name, e.g. `"_smtp-client.device1.example.com"`. See [[RFC6125](#)] and [[RFC4985](#)] for a discussion of application specific identifiers in X.509 certificates.

If the TLS DANE Client Identity extension is in use, then with DANE-EE(3), the subject name need not be present in the certificate.

5. Signaling the Client's DANE Identity in TLS

In general, the client SHOULD explicitly signal its DANE identity via the TLS protocol.

The most important reason is that the server may want an explicit indication from the client that it has a DANE record, so as to avoid unnecessary DNS queries in-band with the TLS handshake for clients that don't support this. In principle, this indication could come in the form of a new X.509 certificate extension but there are a number of additional scenarios where this would not work.

Where client certificate authentication is optional, in response to the server's Certificate Request message, the client can respond with a Client Certificate message with no certificate, and the server may at its discretion continue the handshake without client authentication. However, in practice, problems may arise. There are deployed client software implementations that do not react gracefully when encountering a certificate request message from the TLS server that they did not expect.

DANE client authentication using raw public keys needs a separate mechanism to convey the domain name identity to the TLS server.

Hence, to address this issue generally, a new client identity signaling solution is needed, whereby the client indicates its DANE identity (i.e. its domain name identity and the fact that this identity has an associated TLSA record) to the server. A new TLS extension to convey such an identity [[TLSCLIENTID](#)] has been developed for this purpose. Client implementations of this specification SHOULD use this extension. This extension SHOULD also elicit a "Certificate Request" from servers that implement this protocol, and don't require client certificates otherwise.

6. Example TLSA records for clients

The following examples are provided in the textual presentation format of the TLSA record.

An example TLSA record for the client "device1.example.com." and the application "smtp-client". This record specifies the SHA-256 hash of a PKIX CA certificate to authenticate the client's certificate.

```
_smtp-client.device1.example.com. IN TLSA (
```



```
0 0 1 d2abde240d7cd3ee6b4b28c54df034b9
      7983a1d16e8a410e4561cb106618e971 )
```

An example TLSA record for the client "client2.example.com." and the application "localsvc". This record specifies the SHA-512 hash of the subject public key component of the client's certificate. The certificate usage for this record is 3 (DANE-EE) and thus is validated in accordance with [section 5.1 of RFC 7671](#).

```
_localsvc.client2.example.com. IN TLSA (
  3 1 2 0f8b48ff5fd94117f21b6550aaee89c8
      d8adbc3f433c8e587a85a14e54667b25
      f4dcd8c4ae6162121ea9166984831b57
      b408534451fd1b9702f8de0532ecd03c )
```

7. Changes to Client and Server behavior

A TLS Client conforming to this specification MUST have a signed DNS TLSA record published corresponding to its DNS name and X.509 certificate or public key. The client presents this certificate or public key in the TLS handshake with the server. The client should not offer ciphersuites that are incompatible with its certificate or public key. If the client's certificate has a DANE record with a certificate usage other than DANE-EE, then the presented client certificate MUST have the client's DNS name specified either in the Subject Alternative Name extension's dNSName type, or the SRVName type.

Additionally the client SHOULD use the TLS DANE Client Identity extension [[TLSCIENTID](#)] to explicitly indicate its DNS name.

A TLS Server implementing this specification performs the following steps:

- o Request a client certificate in the TLS handshake (the "Client Certificate Request" message). This could be done unconditionally, or only when it receives the TLS DANE Client Identity extension from the client.
- o If the client has sent the DANE Client Identity extension, then extract the client's domain name from the extension. Otherwise, extract the client identity from the Subject Alternative Name extension's dNSName or SRVName type in the client certificate.

- o Construct the DNS query name for the corresponding TLSA record. If the TLS DANE client identity extension was present, then this name should be used. Otherwise, identities from the client certificate are used. For `dnsName`, the underscored application service label is prepended to the domain name, corresponding to the application in use. For `srvName`, the DNS query name is identical to the content of the `srvName` identifier. See [Section 2](#) for the proposed owner name format.
- o Look up the TLSA record set in the DNS. The response MUST be cryptographically validated using DNSSEC. The server could perform the DNSSEC validation itself. It could also be configured to trust responses obtained via a validating resolver to which it has a secure connection.
- o Extract the RDATA of the TLSA record and match it to the presented client certificate according to the rules specified in the DANE TLS protocol [[RFC6698](#)] [[RFC7671](#)]. If successfully matched, the client is authenticated and the TLS session proceeds. If unsuccessful, the server MUST treat the client as unauthenticated (e.g. it could terminate the session, or proceed with the session giving the client access to resources as a generic unauthenticated user).
- o If there are multiple records in the TLSA record set, then the client is authenticated as long as at least one of the TLSA records matches, subject to [RFC7671](#) digest agility, which SHOULD be implemented.

If the DANE Client Identity extension is not present, and the presented client certificate has multiple distinct reference identifier types (e.g. a `dnsName`, and an `rfc822Name`) then TLS servers configured to perform DANE authentication according to this specification should only examine and authenticate the `dnsName` or `srvName` identity. If the certificate contains both `dnsName` and `srvName` identities, `srvName` should be preferred. See [[RFC6125](#)] for a description of reference identifiers and matching rules.

If the presented client certificate has multiple `dnsName` or `srvName` identities, then the client MUST use the TLS DANE client identity extension to unambiguously indicate its intended name to the server.

Specific applications may be designed to require additional validation steps. For example, a server might want to verify the client's IP address is associated with the certificate in some manner, e.g. by confirming that a secure reverse DNS lookup of that address ties it back to the same domain name, or by requiring an `ipAddress` component to be included in the certificate. Such details

are outside the scope of this document, and should be outlined in other documents specific to the applications that require this behavior.

Servers may have their own whitelisting and authorization rules for which certificates they accept. For example a TLS server may be configured to only allow TLS sessions from clients with certificate identities within a specific domain or set of domains.

8. Raw Public Keys

When using raw public keys in TLS [[RFC7250](#)], this specification requires the use of the TLS DANE Client Identity extension. The associated DANE TLSA records employ only certificate usage 3 (DANE-EE) and a selector value of 1 (SPKI), as described in [[RFC7671](#)].

9. Acknowledgements

This document benefited from discussions with the following people: Duane Wessels, Allison Mankin, Casey Deccio, and Warren Kumari.

10. IANA Considerations

This document includes no request to IANA.

11. Security Considerations

This document makes a narrow update to [RFC 6698](#) by defining the use of the TLSA record for client TLS certificates. There are no security considerations for this document beyond those described in [RFC 6698](#) and [RFC 7671](#) and in the specifications for TLS and DTLS [[RFC5246](#)], [[RFC6347](#)].

12. References

12.1. Normative References

- [RFC4035] Arends, R., Austein, R., Larson, M., Massey, D., and S. Rose, "Protocol Modifications for the DNS Security Extensions", [RFC 4035](#), March 2005.
- [RFC4985] Santesson, S., "Internet X.509 Public Key Infrastructure Subject Alternative Name for Expression of Service Name", [RFC 4985](#), August 2007.
- [RFC5246] Dierks, T. and E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.2", [RFC 5246](#), August 2008.

- [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](#), May 2008.
- [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](#), March 2011.
- [RFC6347] Rescorla, E. and N. Modadugu, "Datagram Transport Layer Security Version 1.2", [RFC 6347](#), January 2012.
- [RFC6698] Hoffman, P. and J. Schlyter, "The DNS-Based Authentication of Named Entities (DANE) Transport Layer Security (TLS) Protocol: TLSA", [RFC 6698](#), August 2012.
- [RFC7218] Gudmundsson, O., "Adding Acronyms to Simplify Conversations about DNS-Based Authentication of Named Entities (DANE)", [RFC 7218](#), April 2014.
- [RFC7250] Wouters, P., Tschofenig, H., Gilmore, J., Weiler, S., and T. Kivinen, "Using Raw Public Keys in Transport Layer Security (TLS) and Datagram Transport Layer Security (DTLS)", [RFC 7250](#), June 2014.
- [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>>.
- [TLSCLIENTID] Huque, S. and V. Dukhovni, "TLS Extension for DANE Client Identity", , <<https://tools.ietf.org/html/draft-huque-tls-dane-clientid>>.

12.2. Informative References

- [RFC3552] Rescorla, E. and B. Korver, "Guidelines for Writing RFC Text on Security Considerations", [BCP 72](#), [RFC 3552](#), July 2003.
- [SRVREG] IANA, ., "Service Name and Transport Protocol Port Number Registry", , <<https://www.iana.org/assignments/service-names-port-numbers/service-names-port-numbers.txt>>.

Authors' Addresses

Shumon Huque
Verisign Labs

Email: shuque@verisign.com

Dan James
Verisign, Inc.

Email: djames@verisign.com

Viktor Dukhovni
Two Sigma

Email: ietf-dane@dukhovni.org

