Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: September 12, 2019

M. Bretelle
Facebook

March 11, 2019

Encoding DNS-over-TLS (DoT) Subject Public Key Info (SPKI) in Name Server name
draft-bretelle-dprive-dot-spki-in-ns-name-00

Abstract

This document describes a mechanism to exchange the Subject Public Key Info (SPKI) ([RFC5280] Section 4.1.2.7) fingerprint associated with a DNS-over-TLS (DoT [RFC7858]) authoritative server by encoding it as part of its name. The fingerprint can thereafter be used to validate the certificate received from the DoT server as well as being able to discover support for DoT on the server.

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 September 12, 2019.

Copyright Notice

Table of Contents

1. Introduction

This document describes a mechanism to exchange the Subject Public Key Info (SPKI) ([RFC5280] Section 4.1.2.7) fingerprint associated with a DNS-over-TLS (DoT [RFC7858]) authoritative server by encoding it as part of its name. The fingerprint can thereafter be used to validate the certificate received from the DoT server as well as being able to discover support for DoT on the server.

2. Terminology

A server that supports DNS-over-TLS is called a "DoT server" to differentiate it from a "DNS Server" (one that provides DNS service over any other protocol), likewise, a client that supports this protocol is called a "DoT client"

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

3. Validating a remote DoT server

While DoT provides protection against eavesdropping and on-path tampering of the DNS queries exchanged with an authoritative server, a recursive server that is talking to a remote DoT server needs a mechanism to authenticate that the name server it is communicating with is indeed the one that the authority of the zone manages or has delegated responsibility to.

A common mechanism is to have TLS certificates issued by "Certification Authorities" (CAs), those CA public keys are used as trust anchors, and through a chain of trust, a leaf TLS certificate can be validated. Any CA is able to issue a certificate for any domain, which can have its drawbacks ([RFC6698] Section 1.1).

Another method is to leverage DANE/TLSA ([RFC6698]), in which case a recursive resolver would be provided the certificate or SPKI hash over DNS and validate it using DNSSEC ([RFC4033], [RFC4034], and [RFC4035]).

This document describes a mechanism to signal to a recursive resolver that DoT is supported by the authoritative name server as well as providing a fingerprint of the SPKI to expect from the name server, this is done by formatting a special first label for the name servers. Recursive servers that understand the naming convention detailed in this document will be able to upgrade their connection to the authoritative server to TLS, while the ones that don't will transparently use the name servers as a standard UDP/53 and TCP/53 servers. This format is heavily inspired from [dnscurve].

4. Encoding data in a domain name label

A label is limited to a maximum of 63 octets ([RFC1035] Section 2.3.4) and has a limited set of characters that can be used ([RFC1035] Section 2.3.1), limiting both the amount of data that can be embedded in a label as well as the encoding format.

The set of character used by Base32 encoding ([RFC4648] Section 6), without padding character, is suitable to be used in a label. Base32 encodes a 5-bit group into 1 byte which allows to encode up to 39 bytes within the 63 bytes space of a label.

floor(63 * 5 / 8)

While this limits what can be encoded in a label, there is enough space to store the hash produced by sha256 which requires 32 bytes, leaving 7 bytes to spare.

4.1. Formatting DoT SPKI in name server name.

The formatting of a name server is defined as follow:

```
<label> ::= <dot-header> <b32-spki-fingerprint>
<dot-header> ::= "dot-"
<b32-spki-fingerprint> ::= base32encode(<spki-fingerprint>)
<spki-fingerprint> ::= sha256(<spki>)
<spki> ::= der-encoded binary structure of SubjectPublicKeyInfo
```

4.1.1. Example

For the zone example.com, having 2 name servers, one at IPv4 192.0.2.1 and one at IPv6 2001:DB8::1, both of them providing DoT support and using certificate cert.pem, the "<b32-spki-fingerprint>" can be generated using the following command line:

```
openssl x509 -in /path/to/cert.pem  -pubkey -noout | \
openssl pkey -pubin -outform der | \
openssl dgst -sha256 -binary | \
base32 | tr -d '=' | tr '[:upper:]' '[:lower:]'
tpwxmgqdaurcqxqsckxvdq5sty3opxlgcbjj43kumdq62kpqr72a
```

To generate the full label, "dot-" get prefixed to the base32 encoded fingerprint.

```
...
...
;; QUESTION SECTION:
```

```
;example.com.      IN      NS

;; AUTHORITY SECTION:
example.com. 3600  IN      NS      dot-tpwxmgqdaurcqxqsckxvdq5sty3opxlgcbjj43ku
example.com. 3600  IN      NS      dot-tpwxmgqdaurcqxqsckxvdq5sty3opxlgcbjj43ku

;; ADDITIONAL SECTION:
dot-tpwxmgqdaurcqxqsckxvdq5sty3opxlgcbjj43kumdq62kpqr72a.a.example.com. 3600 IN
dot-tpwxmgqdaurcqxqsckxvdq5sty3opxlgcbjj43kumdq62kpqr72a.b.example.com. 3600 IN
...
...
```

4.2. Handling by the recursive servers

4.2.1. Servers supporting this specification

When a recursive server gets the list of authoritative servers serving a specific zone, it gets a list of name of hosts.

If:

- the first label is 56 bytes long
- AND the first 4 bytes matches "dot-"
- AND the remaining 52 bytes can be base32-decoded

the recursive server will attempt to connect to the name server using TLS over port 853 and validate that the SHA256 hash of the SPKI in the certificate provided by the name server matches what was previously decoded.

If the TLS session fail to establish, either unavailability of the service on port 853, TLS authentication failure, the behaviour of the recursive server depends on whether it is operating in strict or opportunistic mode ([I-D.ietf-dprive-dtls-and-tls-profiles]).

In strict mode, the resolver MUST stop using this authoritative name server, and MUST try other servers of the DNS zone. In opportunistic mode, the resolver MUST use the authoritative name server despite the failure. It MAY try other name servers of the zone before, in the

hope they will accept TLS and be authenticated.

4.2.2. Servers not supporting this specification

A server not supporting this specification will be unaware of anything special with this name server and consider it like any other name servers.

5. Security Considerations

TODO Security

6. IANA Considerations

TODO: This document requires IANA actions (new RR type).

7. References

7.1. Normative References

[I-D.ietf-dprive-dtls-and-tls-profiles]
Dickinson, S., Gillmor, D., and T. Reddy, "Usage and (D)TLS Profiles for DNS-over-(D)TLS", draft-ietf-dprive-dtls-and-tls-profiles-11 (work in progress), September 2017.

[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>.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC4033] Arends, R., Austein, R., Larson, M., Massey, D., and S. Rose, "DNS Security Introduction and Requirements", RFC 4033, DOI 10.17487/RFC4033, March 2005, <https://www.rfc-editor.org/info/rfc4033>.

[RFC4034] Arends, R., Austein, R., Larson, M., Massey, D., and S.

Rose, "Resource Records for the DNS Security Extensions", RFC 4034, DOI 10.17487/RFC4034, March 2005, <https://www.rfc-editor.org/info/rfc4034>.

[RFC4035] Arends, R., Austein, R., Larson, M., Massey, D., and S. Rose, "Protocol Modifications for the DNS Security Extensions", RFC 4035, DOI 10.17487/RFC4035, March 2005, <https://www.rfc-editor.org/info/rfc4035>.

[RFC4648] Josefsson, S., "The Base16, Base32, and Base64 Data Encodings", RFC 4648, DOI 10.17487/RFC4648, October 2006, <https://www.rfc-editor.org/info/rfc4648>.

[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, <https://www.rfc-editor.org/info/rfc5280>.

[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>.

[RFC7858] Hu, Z., Zhu, L., Heidemann, J., Mankin, A., Wessels, D., and P. Hoffman, "Specification for DNS over Transport Layer Security (TLS)", RFC 7858, DOI 10.17487/RFC7858, May 2016, <https://www.rfc-editor.org/info/rfc7858>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

7.2. Informative References

[dnscurve]
"DNSCurve", n.d., <https://dnscurve.org/>.

Acknowledgments

TODO acknowledge.

Author's Address

Emmanuel Bretelle
Facebook

Email: chantra@fb.com