Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: October 12, 2014

P. Hoffman
VPN Consortium
April 10, 2014

Optimistic Encryption using TLS Signaling in the DNS
draft-hoffman-trytls-02

Abstract

Many Internet servers offer content in two transports: unencrypted, and encrypted with TLS. A user who accesses some content with a URL that indicates unencrypted (such as "http:") might prefer to get the content encrypted but doesn't bother to, or can't, change the URL to indicate this. This proposal allows Internet clients, particularly web clients and mail user agents, to do a DNS lookup to see whether they might expect content for a particular host to also be available under TLS. Using the DNS for this is much faster than attempting a TLS session that might time out or take many round trips in order to discover that the content is not available.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

Starting a TLS [RFC5246] session takes time and resources, so applications tend not to do it unless specifically asked, such as when a user enters a "https:" or "imaps:" URL. The downside of this is that some Internet traffic that might be encrypted goes unencrypted even when a user might want encryption.

A classic example of this problem is a web user who cares about encrypting as much content as possible and is willing to type URLs with "https:", but goes to a web page whose URLs are all "http". Some of those pages might be served under either "http:" or "https:", but you can't specify both in an HTML page.

Although most people think of this as a problem for HTTP [RFC2817], it also affects mail user agents that use either POP [RFC1939] or IMAP [RFC3501]. Although it is uncommon to see "pop:" or "imap:" URLs, many applications use them internally. Allowing servers that allow both the unencrypted and encrypted versions of these protocols would also go a long way towards encrypting more traffic on the Internet.

A potential solution to this problem is to allow a site operator to tell applications that content that is available unencrypted is likely to also be available encrypted with TLS. If the application can do a quick check for TLS availability, the application might be more willing to risk the setup time for TLS. This document proposed to do that with a new DNS RRtype, TRYTLS, that is a non-binding indicator from the site owner that clients that can use TLS coming to this domain name are likely to find a TLS server for a particular protocol.

An orthogonal solution that applies only to HTTP is "HTTP Alternative Services", [AltSvc]. That proposal allows the server in an existing cleartext HTTP connection to indicate to the client that an alternate service (in this case, TLS) exists, as well as to give its location. The proposal in this document is not meant to be a replacement for HTTP Alternate Services; instead, it allows clients to find out about a pontential TLS server before even sending any cleartext.

2. The TRYTLS Resource Record

The TRYTLS resource record type, whose value is TBD1, lists the port on which a particular TLS-based service might be found for a given application protocol.

The presentation format is:

```
_appname.hostname IN TRYTLS sec-port
```

The application name ("_appname") being queried is taken from a new IANA registry. The initial values for the names in the registry are "_http", "_pop", and "_imap".

The secure port number (called "sec-port") is a two-octet positive integer.

3. Semantics of the TRYTLS Record

The lack of a TRYTLS record in a zone implies absolutely nothing.

The presence of a TRYTLS record for a particular application type indicates that there is likely to be a server for that protocol, running under TLS, at the port number given. There is absolutely no guarantee that such a server exists, or that the TLS server's certificate will be trusted by any particular client. If the record exists, the port number in the response is the port number a client should use to access the server over TLS.

The presence of a TRYTLS record for HTTP (such as "_http.www.example.com") indicates that some HTTP origins which have the given hostname will also be available over TLS. The presence of such a record does not indicate that all origins, or all specific URLs that include those origins, will be served under TLS.

The existence or absence of a TRYTLS record does not have any effect on other ways of discovering whether there is a TLS service for a particular application.

4. Comparison to Other Proposals

Some people interpret the DANE TLSA RRtype [RFC6698] as indicating that TLS is available for HTTP at a particular hostname, even though this interpretation is not part of the specification. Such an indication is being discussed in the DANE WG. The TRYTLS differs from TLSA in that TRYTLS does not need to be protected by DNSSEC. Thus, doing a TRYTLS lookup is available to all clients, not just those with their own validating DNS resolvers or secure connections to such resolvers. However, doing a successful TLSA lookup will lead to the client also having a much stronger trust of the eventual TLS session because the client will also have the TLS trust anchor or end entity certificate validated through the DNSSEC trust chain.

An earlier Internet-Draft, draft-hoffman-server-has-tls, tried to combine the semantics of the TRYTLS record with the idea of a server-provided policy for fallback. That draft has been abandoned because the IETF community could not come to any agreement on whether such a fallback policy was a good or terrible idea.

5. IANA Considerations

** Insert DNS RRtype template here for TRYTLS that assigns TBD1. **

** Create a new registry for _appname **

6. Security Considerations

There is a general positive security effect on the Internet when more traffic is encrypted. There are probably some exceptions to this statement, and probably some people who would say that the effect is much more positive than "general".

There is no reason to require TRYTLS to be protected by DNSSEC. An attacker who adds a TRYTLS record when TLS is not available will cause a slight denial-of-service attack, but one that is not much worse than the case today where a client might try a TLS connection anyway.

7. Informative References

[AltSvc] Nottingham, M., McManus, P., and J. Reschke, "HTTP Alternative Services", draft-ietf-httpbis-alt-svc (work in progress), 2014.

[RFC1939] Myers, J. and M. Rose, "Post Office Protocol - Version 3", STD 53, RFC 1939, May 1996.

[RFC2817] Khare, R. and S. Lawrence, "Upgrading to TLS Within HTTP/1.1", RFC 2817, May 2000.

[RFC3501] Crispin, M., "INTERNET MESSAGE ACCESS PROTOCOL - VERSION 4rev1", RFC 3501, March 2003.

[RFC5246] Dierks, T. and E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.2", RFC 5246, August 2008.

[RFC6698] Hoffman, P. and J. Schlyter, "The DNS-Based Authentication of Named Entities (DANE) Transport Layer Security (TLS) Protocol: TLSA", RFC 6698, August 2012.

Author's Address

Paul Hoffman
VPN Consortium

Email: paul.hoffman@vpnc.org