HTTP Working Group
Internet-Draft
Intended status: Experimental
Expires: June 19, 2016

M. Nottingham

M. Thomson
Mozilla

December 17, 2015

Opportunistic Security for HTTP
draft-ietf-httpbis-http2-encryption-03

Abstract

This document describes how "http" URIs can be accessed using Transport Layer Security (TLS) to mitigate pervasive monitoring attacks.

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 June 19, 2016.

Copyright Notice

Copyright (c) 2015 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 2
   1.1. Goals and Non-Goals 3
   1.2. Notational Conventions 3
2. Using HTTP URIs over TLS 3
3. Server Authentication 4
4. Interaction with "https" URIs 4
5. Requiring Use of TLS 5
   5.1. The HTTP-TLS Header Field 5
   5.2. Operational Considerations 7
6. Security Considerations 7
   6.1. Security Indicators 7
   6.2. Downgrade Attacks 7
   6.3. Privacy Considerations 8
   6.4. Confusion Regarding Request Scheme 8
7. References 8
   7.1. Normative References 8
   7.2. Informative References 9
Appendix A. Acknowledgements 9
Authors' Addresses 9

1. Introduction

This document describes a use of HTTP Alternative Services [I-D.ietf-httpbis-alt-svc] to decouple the URI scheme from the use and configuration of underlying encryption, allowing a "http" URI to be accessed using TLS [RFC5246] opportunistically.

Serving "https" URIs require acquiring and configuring a valid certificate, which means that some deployments find supporting TLS difficult. This document describes a usage model whereby sites can serve "http" URIs over TLS without being required to support strong server authentication.

Opportunistic Security [RFC7435] does not provide the same guarantees as using TLS with "https" URIs; it is vulnerable to active attacks, and does not change the security context of the connection. Normally, users will not be able to tell that it is in use (i.e., there will be no "lock icon").

By its nature, this technique is vulnerable to active attacks. A mechanism for partially mitigating them is described in Section 5.

1.1. Goals and Non-Goals

The immediate goal is to make the use of HTTP more robust in the face of pervasive passive monitoring [RFC7258].

A secondary goal is to limit the potential for active attacks. It is not intended to offer the same level of protection as afforded to "https" URIs, but instead to increase the likelihood that an active attack can be detected.

A final (but significant) goal is to provide for ease of implementation, deployment and operation. This mechanism is expected to have a minimal impact upon performance, and require a trivial administrative effort to configure.

1.2. Notational Conventions

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].

2. Using HTTP URIs over TLS

An origin server that supports the resolution of "http" URIs can indicate support for this specification by providing an alternative service advertisement [I-D.ietf-httpbis-alt-svc] for a protocol identifier that uses TLS, such as "h2" [RFC7540].

A client that receives such an advertisement MAY make future requests intended for the associated origin ([RFC6454]) to the identified service (as specified by [I-D.ietf-httpbis-alt-svc]).

A client that places the importance of protection against passive attacks over performance might choose to withhold requests until an encrypted connection is available. However, if such a connection cannot be successfully established, the client can resume its use of the cleartext connection.

A client can also explicitly probe for an alternative service advertisement by sending a request that bears little or no sensitive information, such as one with the OPTIONS method. Likewise, clients with existing alternative services information could make such a request before they expire, in order minimize the delays that might be incurred.

3. Server Authentication

By their nature, "http" URIs do not require cryptographically strong server authentication; that is only implied by "https" URIs. Furthermore, doing so (as per [RFC2818]) creates a number of operational challenges. For these reasons, server authentication is not mandatory for "http" URIs when using the mechanism described in this specification.

When connecting to an alternative service for an "http" URI, clients are not required to perform the server authentication procedure described in Section 3.1 of [RFC2818]. The server certificate, if one is proffered by the alternative service, is not necessarily checked for validity, expiration, issuance by a trusted certificate authority or matched against the name in the URI. Therefore, the alternative service can provide any certificate, or even select TLS cipher suites that do not include authentication.

A client MAY perform additional checks on the offered certificate if the server does not select an unauthenticated TLS cipher suite. This document doesn't define any such checks, though clients could be configured with a policy that defines what is acceptable.

As stipulated by [I-D.ietf-httpbis-alt-svc], clients MUST NOT use alternative services with a host other than the origin's, unless the alternative service itself is strongly authenticated (as the origin's host); for example, using TLS with a certificate that validates as per [RFC2818].

4. Interaction with "https" URIs

When using alternative services, requests for resources identified by both "http" and "https" URIs might use the same connection, because HTTP/2 permits requests for multiple origins on the same connection.

Since "https" URIs rely on server authentication, a connection that is initially created for "http" URIs without authenticating the server cannot be used for "https" URIs until the server certificate is successfully authenticated. Section 3.1 of [RFC2818] describes the basic mechanism, though the authentication considerations in [I-D.ietf-httpbis-alt-svc] also apply.

Connections that are established without any means of server authentication (for instance, the purely anonymous TLS cipher suites), cannot be used for "https" URIs.

5. Requiring Use of TLS

Editors' Note: this is a very rough take on an approach that would provide a limited form of protection against downgrade attack. It's unclear at this point whether the additional effort (and modest operational cost) is worthwhile.

The mechanism described in this specification is trivial to mount an active attack against, for two reasons:

- A client that doesn't perform authentication is an easy victim of server impersonation, through man-in-the-middle attacks.
- A client that is willing to use HTTP over cleartext to resolve the resource will do so if access to any TLS-enabled alternative services is blocked at the network layer.

Given that the primary goal of this specification is to prevent passive attacks, these are not critical failings (especially considering the alternative - HTTP over cleartext). However, a modest form of protection against active attacks can be provided for clients on subsequent connections.

When an alternative service is able to commit to providing service for a particular origin over TLS for a bounded period of time, clients can choose to rely upon its availability, failing when it cannot be contacted. Effectively, this makes the choice to use a secured protocol "sticky" in the client.

5.1. The HTTP-TLS Header Field

A alternative service can make this commitment by sending a "HTTP-TLS" header field, described here using the '#' ABNF extension defined in Section 7 of [RFC7230]:

```
HTTP-TLS     = 1#parameter
```

When it appears in a HTTP response from a strongly authenticated alternative service, this header field indicates that the availability of the origin through TLS-protected alternative services is "sticky", and that the client MUST NOT fall back to cleartext protocols while this information is considered fresh.

For example:

```
GET /index.html HTTP/1.1
Host: example.com


HTTP/1.1 200 OK
Content-Type: text/html
Cache-Control: max-age=600
Age: 30
Date: Thu, 1 May 2014 16:20:09 GMT
HTTP-TLS: ma=3600
```

This header field creates a commitment from the origin [RFC6454] of the associated resource (in the example, "http://example.com"). For the duration of the commitment, clients SHOULD strongly authenticate the server for all subsequent requests made to that origin, though this creates some risks for clients (see Section 5.2).

Authentication for HTTP over TLS is described in Section 3.1 of [RFC2818], noting the additional requirements in Section 2.1 of [I-D.ietf-httpbis-alt-svc]. The header field MUST be ignored if strong authentication fails; otherwise, an attacker could create a persistent denial of service by falsifying a commitment.

The commitment to use authenticated TLS persists for a period determined by the value of the "ma" parameter. See Section 4.2.3 of [RFC7234] for details of determining response age.

```
ma-parameter     = delta-seconds
```

The commitment made by the "HTTP-TLS" header field applies only to the origin of the resource that generates the "HTTP-TLS" header field.

Requests for an origin that has a persisted, unexpired value for "HTTP-TLS" MUST fail if they cannot be made over an authenticated TLS connection.

Note that the commitment is not bound to a particular alternative service. Clients SHOULD use alternative services that they become aware of. However, clients MUST NOT use an unauthenticated alternative service for an origin with this commitment. Where there is an active commitment, clients MAY instead ignore advertisements for unsecured alternatives services.

5.2. Operational Considerations

To avoid situations where a persisted value of "HTTP-TLS" causes a client to be unable to contact a site, clients SHOULD limit the time that a value is persisted for a given origin. A lower limit might be appropriate for initial observations of "HTTP-TLS"; the certainty that a site has set a correct value - and the corresponding limit on persistence - can increase as the value is seen more over time.

Once a server has indicated that it will support authenticated TLS, a client MAY use key pinning [RFC7469] or any other mechanism that would otherwise be restricted to use with "https" URIs, provided that the mechanism can be restricted to a single HTTP origin.

6. Security Considerations

6.1. Security Indicators

User Agents MUST NOT provide any special security indicia when an "http" resource is acquired using TLS. In particular, indicators that might suggest the same level of security as "https" MUST NOT be used (e.g., using a "lock device").

6.2. Downgrade Attacks

A downgrade attack against the negotiation for TLS is possible. With the "HTTP-TLS" header field, this is limited to occasions where clients have no prior information (see Section 6.3), or when persisted commitments have expired.

For example, because the "Alt-Svc" header field [I-D.ietf-httpbis-alt-svc] likely appears in an unauthenticated and unencrypted channel, it is subject to downgrade by network attackers. In its simplest form, an attacker that wants the connection to remain in the clear need only strip the "Alt-Svc" header field from responses.

Downgrade attacks can be partially mitigated using the "HTTP-TLS" header field, because when it is used, a client can avoid using cleartext to contact a supporting server. However, this only works when a previous connection has been established without an active attacker present; a continuously present active attacker can either prevent the client from ever using TLS, or offer its own certificate.

6.3. Privacy Considerations

Cached alternative services can be used to track clients over time; e.g., using a user-specific hostname. Clearing the cache reduces the ability of servers to track clients; therefore clients MUST clear cached alternative service information when clearing other origin-based state (i.e., cookies).

6.4. Confusion Regarding Request Scheme

Many existing HTTP/1.1 implementations use the presence or absence of TLS in the stack to determine whether requests are for "http" or "https" resources. This is necessary in many cases because the most common form of an HTTP/1.1 request does not carry an explicit indication of the URI scheme.

HTTP/1.1 MUST NOT be used for opportunistically secured requests.

Some HTTP/1.1 implementations use ambient signals to determine if a request is for an "https" resource. For example, implementations might look for TLS on the stack or a port number of 443. An implementation that supports opportunistically secured requests SHOULD suppress these signals if there is any potential for confusion.

7. References

7.1. Normative References

[I-D.ietf-httpbis-alt-svc]
mnot, m., McManus, P., and J. Reschke, "HTTP Alternative Services", draft-ietf-httpbis-alt-svc-09 (work in progress), November 2015.

[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>.

[RFC2818] Rescorla, E., "HTTP Over TLS", RFC 2818, DOI 10.17487/RFC2818, May 2000, <http://www.rfc-editor.org/info/rfc2818>.

[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>.

[RFC6454] Barth, A., "The Web Origin Concept", RFC 6454, DOI 10.17487/RFC6454, December 2011, <http://www.rfc-editor.org/info/rfc6454>.

[RFC7230] Fielding, R., Ed. and J. Reschke, Ed., "Hypertext Transfer Protocol (HTTP/1.1): Message Syntax and Routing", RFC 7230, DOI 10.17487/RFC7230, June 2014, <http://www.rfc-editor.org/info/rfc7230>.

[RFC7234] Fielding, R., Ed., Nottingham, M., Ed., and J. Reschke, Ed., "Hypertext Transfer Protocol (HTTP/1.1): Caching", RFC 7234, DOI 10.17487/RFC7234, June 2014, <http://www.rfc-editor.org/info/rfc7234>.

[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>.

[RFC7540] Belshe, M., Peon, R., and M. Thomson, Ed., "Hypertext Transfer Protocol Version 2 (HTTP/2)", RFC 7540, DOI 10.17487/RFC7540, May 2015, <http://www.rfc-editor.org/info/rfc7540>.

7.2. Informative References

[RFC7258] Farrell, S. and H. Tschofenig, "Pervasive Monitoring Is an Attack", BCP 188, RFC 7258, DOI 10.17487/RFC7258, May 2014, <http://www.rfc-editor.org/info/rfc7258>.

[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>.

Appendix A. Acknowledgements

Thanks to Patrick McManus, Eliot Lear, Stephen Farrell, Guy Podjarny, Stephen Ludin, Erik Nygren, Paul Hoffman, Adam Langley, Eric Rescorla and Richard Barnes for their feedback and suggestions.

Authors' Addresses

Mark Nottingham

Email: mnot@mnot.net
URI: http://www.mnot.net/

Martin Thomson
Mozilla

Email: martin.thomson@gmail.com