

HTTP Working Group
Internet-Draft
Intended status: Standards Track
Expires: September 25, 2015

A. Hutton
Unify
J. Uberti
Google
M. Thomson
Mozilla
March 24, 2015

The Tunnel-Protocol HTTP Header Field
draft-ietf-httpbis-tunnel-protocol-02

Abstract

This specification allows HTTP CONNECT requests to indicate what protocol will be used within the tunnel once established, using the Tunnel-Protocol header field.

Editorial Note (To be removed by RFC Editor)

Discussion of this draft takes place on the HTTPBIS working group mailing list (ietf-http-wg@w3.org), which is archived at [\[1\]](#).

Working Group information can be found at [\[2\]](#) and [\[3\]](#); source code and issues list for this draft can be found at [\[4\]](#).

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 September 25, 2015.

Copyright Notice

Copyright (c) 2015 IETF Trust and the persons identified as the document authors. All rights reserved.

Internet-Draft

Tunnel-Protocol

March 2015

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.	Requirements Language	3
2.	The Tunnel-Protocol HTTP Header Field	3
2.1.	Header Field Values	3
2.2.	Syntax	3
3.	IANA Considerations	4
4.	Security Considerations	4
5.	References	5
5.1.	Normative References	5
5.2.	Informative References	5
5.3.	URIs	6

[1.](#) Introduction

The HTTP CONNECT method ([Section 4.3.6 of \[RFC7231\]](#)) requests that the recipient establish a tunnel to the identified origin server and thereafter forward packets, in both directions, until the tunnel is closed. Such tunnels are commonly used to create end-to-end virtual connections, through one or more proxies.

The HTTP Tunnel-Protocol header field identifies the protocol that will be spoken within the tunnel, using the Application Layer Protocol Negotiation identifier (ALPN, [\[RFC7301\]](#)).

When the CONNECT method is used to establish a tunnel, the Tunnel-Protocol header field can be used to identify the protocol that the client intends to use with that tunnel. For a tunnel that is then secured using TLS [\[RFC5246\]](#), the header field carries the same application protocol label as will be carried within the TLS handshake. If there are multiple possible application protocols, all of those application protocols are indicated.

The Tunnel-Protocol header field carries an indication of client intent only. In TLS, the final choice of application protocol is made by the server from the set of choices presented by the client. Other protocols could negotiate protocols differently.

Proxies do not implement the tunneled protocol, though they might choose to make policy decisions based on the value of the header field. For example, a proxy could use the application protocol to select appropriate traffic prioritization.

[1.1.](#) Requirements Language

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 [RFC 2119](#) [[RFC2119](#)].

[2.](#) The Tunnel-Protocol HTTP Header Field

Clients include the Tunnel-Protocol header field in an HTTP CONNECT request to indicate the application layer protocol that will be used within the tunnel, or the set of protocols that might be used within the tunnel.

[2.1.](#) Header Field Values

Valid values for the protocol field are taken from the "Application-Layer Protocol Negotiation (ALPN) Protocol ID" registry ([5]) established by [[RFC7301](#)].

[2.2.](#) Syntax

The ABNF (Augmented Backus-Naur Form) syntax for the Tunnel-Protocol header field is given below. It is based on the Generic Grammar defined in [Section 2 of \[RFC7230\]](#).

```
Tunnel-Protocol = "Tunnel-Protocol": " 1#protocol-id
protocol-id      = token ; percent-encoded ALPN protocol identifier
```

ALPN protocol names are octet sequences with no additional constraints on format. Octets not allowed in tokens ([\[RFC7230\]](#),

[Section 3.2.6](#)) MUST be percent-encoded as per [Section 2.1 of \[RFC3986\]](#). Consequently, the octet representing the percent character "%" (hex 25) MUST be percent-encoded as well.

In order to have precisely one way to represent any ALPN protocol name, the following additional constraints apply:

- o Octets in the ALPN protocol MUST NOT be percent-encoded if they are valid token characters except "%", and
- o When using percent-encoding, uppercase hex digits MUST be used.

With these constraints, recipients can apply simple string comparison to match protocol identifiers.

For example:

```
CONNECT www.example.com HTTP/1.1
Host: www.example.com
Tunnel-Protocol: h2, http%2F1.1
```

[3.](#) IANA Considerations

HTTP header fields are registered within the "Message Headers" registry maintained at [6]. This document defines and registers the Tunnel-Protocol header field, according to [\[RFC3864\]](#) as follows:

Header Field Name: Tunnel-Protocol

Protocol: http

Status: Standard

Reference: [Section 2](#)

Change Controller: IETF (iesg@ietf.org) - Internet Engineering Task Force

[4.](#) Security Considerations

In case of using HTTP CONNECT to a TURN server ("Traversal Using Relays around NAT", [RFC5766]) the security considerations of [Section 4.3.6 of \[RFC7231\]](#) apply. It states that there "are significant risks in establishing a tunnel to arbitrary servers, particularly when the destination is a well-known or reserved TCP port that is not intended for Web traffic. Proxies that support CONNECT SHOULD restrict its use to a limited set of known ports or a configurable whitelist of safe request targets."

The Tunnel-Protocol header field described in this document is an OPTIONAL header field. Clients and HTTP proxies could choose to not support the header and therefore fail to provide it, or ignore it when present. If the header is not available or ignored, a proxy cannot identify the purpose of the tunnel and use this as input to any authorization decision regarding the tunnel. This is indistinguishable from the case where either client or proxy does not support the Tunnel-Protocol header field.

The value of the Tunnel-Protocol header field could be falsified by a client. If the data being sent through the tunnel is encrypted (for example, with TLS [RFC5246]), then the proxy might not be able to directly inspect the data to verify that the claimed protocol is the one which is actually being used, though a proxy might be able to perform traffic analysis [TRAFFIC]. A proxy therefore cannot rely on the value of the Tunnel-Protocol header field as a policy input in all cases.

[5.](#) References

[5.1.](#) Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), March 1997, <<http://www.rfc-editor.org/info/rfc2119>>.
- [RFC3864] Klyne, G., Nottingham, M., and J. Mogul, "Registration Procedures for Message Header Fields", [BCP 90](#), [RFC 3864](#), September 2004, <<http://www.rfc-editor.org/info/rfc3864>>.
- [RFC3986] Berners-Lee, T., Fielding, R., and L. Masinter, "Uniform

Resource Identifier (URI): Generic Syntax", STD 66, [RFC 3986](#), January 2005, <<http://www.rfc-editor.org/info/rfc3986>>.

- [RFC7230] Fielding, R. and J. Reschke, "Hypertext Transfer Protocol (HTTP/1.1): Message Syntax and Routing", [RFC 7230](#), June 2014, <<http://www.rfc-editor.org/info/rfc7230>>.
- [RFC7231] Fielding, R. and J. Reschke, "Hypertext Transfer Protocol (HTTP/1.1): Semantics and Content", [RFC 7231](#), June 2014, <<http://www.rfc-editor.org/info/rfc7231>>.
- [RFC7301] Friedl, S., Popov, A., Langley, A., and E. Stephan, "Transport Layer Security (TLS) Application-Layer Protocol Negotiation Extension", [RFC 7301](#), July 2014, <<http://www.rfc-editor.org/info/rfc7301>>.

[5.2.](#) Informative References

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

Hutton, et al.

Expires September 25, 2015

[Page 5]

Internet-Draft

Tunnel-Protocol

March 2015

- [RFC5766] Mahy, R., Matthews, P., and J. Rosenberg, "Traversal Using Relays around NAT (TURN): Relay Extensions to Session Traversal Utilities for NAT (STUN)", [RFC 5766](#), April 2010, <<http://www.rfc-editor.org/info/rfc5766>>.
- [TRAFFIC] Pironti, A., Strub, P-Y., and K. Bhargavan, "Website Users by TLS Traffic Analysis: New Attacks and Effective Countermeasures, Revision 1", 2012, <<https://alfredo.pironti.eu/research/publications/full/identifying-website-users-tls-traffic-analysis-new-attacks-and-effective-counterme>>.

[5.3.](#) URIs

- [1] <http://www.iana.org/assignments/tls-extensiontype-values/#alpn->

[protocol-ids](#)

[2] <https://www.iana.org/assignments/message-headers>

Authors' Addresses

Andrew Hutton
Unify
Technology Drive
Nottingham NG9 1LA
UK

EMail: andrew.hutton@unify.com

Justin Uberti
Google
747 6th Ave S
Kirkland, WA 98033
US

EMail: justin@uberti.name

Martin Thomson
Mozilla
331 E Evelyn Street
Mountain View, CA 94041
US

EMail: martin.thomson@gmail.com