

HTTPBis Working Group
Internet-Draft
Intended status: Informational
Expires: April 24, 2014

S. Loreto
R. Skog
H. Spaak
Ericsson
D. Druta
M. Hafeez
AT&T
October 21, 2013

2.0 Proxy in HTTP/2.0
draft-loreto-httpbis-proxy20-01

Abstract

This document defines and clarifies the role of proxies (aka intermediaries) in HTTP 2.0. It aims to assure that HTTP 2.0 contains the same proxy features present in HTTP 1.1. It also defines HTTP 2.0 proxies and advocates the importance and the benefits that they can provide for HTTP 2.0. This document aims to start the discussion within the HTTPBis wg.

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 April 24, 2014.

Copyright Notice

Copyright (c) 2013 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	3
2. Terminology	4
3. Current Proxy usages	4
4. HTTP 2.0 Proxy	5
4.1. HTTP 2.0 Proxy Features	5
4.2. 2.0 Proxy: discovery mechanisms	6
5. Acknowledgments	6
6. References	6
6.1. Normative References	6
6.2. Informative References	7
Authors' Addresses	7

1. Introduction

Proxies (aka intermediaries) are an important part of existing HTTP deployments; they both significantly help to support and improve the scalability needs of the Web. Large enterprise deployments are leveraging proxies in their architecture and home networking solutions depend on intermediaries in order to connect to the internet. These are just two examples of the use cases relying on intermediaries and the current deployment scale will demand increased compatibility with existing and future services. However till now little effort has been spent to define and clarify the role intermediaries play in HTTP 2.0.

This draft aims to assure that HTTP/2.0 contains the same proxy features present in HTTP/1.1 [I-D.ietf-httpbis-pl-messaging], [RFC2616] and [RFC2817].

This draft also advocates the importance and the benefits that proxies can provide for HTTP/2.0 and aims to start a discussion on this topic within the HTTPBis wg. Caching is not addressed in this initial version of the document.

A Proxy is defined in HTTP/1.1 is a

"a message forwarding agent that is selected by the client, usually via local configuration rules, to receive requests for some type(s) of absolute URI and attempt to satisfy those requests via translation through the HTTP interface."

A proxy acts as a server and a client for the purpose of making requests on behalf of the client. The requests can be serviced internally (i.e. if the Proxy also implement a cache) or by passing them on to the origin server.

Proxies are often used to group an organization's HTTP requests through a common intermediary for the sake of security, annotation services, or shared caching.

Moreover Proxies can improve the Quality of Experience (QoE) in particular scenarios, such as in a mobile network.

There are important Proxy uses cases currently used in HTTP 1.1 and most likely they will be also important for 2.0 (see Section 3). There are also Proxy use cases that will be 2.0 specific (see Section 4).

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. Terminology

HTTP1.1 [I-D.ietf-httpbis-pl-messaging] defines three form of intermediaries:

Transforming proxy: Is designed and configured to modify request or response messages in a semantically meaningful way. Such transformation is presumed to be desired by the client that selected the proxy.

Gateway (a.k.a., reverse proxy): is an intermediary that acts as an origin server for the outbound connection, but translates received requests and forwards them inbound to another server or servers.

Gateways also include proxies that transform requests from one transport protocol to another, e.g. as in WAP (Wireless Application Protocol) gateways which transformed WAP1 transport to HTTP 1.1. For HTTP 2.0, which includes many of the same features as WAP1 (e.g. header compression, body compression, tunneling of multiple requests over a single connection/session, push, etc), proxies may also need to provide such a transformation, e.g. to enable a single client transport protocol connection (HTTP 2.0), but transform that to an HTTP 1.1 connection for servers that do not support HTTP 2.0.

Tunnel: acts as a blind relay between two connections without changing the messages. Once active, a tunnel is not considered a party to the HTTP communication, though the tunnel might have been initiated by an HTTP request. A tunnel ceases to exist when both ends of the relayed connection are closed. Tunnels are used to extend a virtual connection through an intermediary.

HTTP2.0 defines the following intermediaries:

2.0 Proxy: an interposed entity the user-agent is informed about its existence (by explicit configuration or other TBD mechanisms), and that can be easily bypassed if the user-agent decide to do it.

3. Current Proxy usages

Among all the possible existing Proxy usages, there are some that really improve the user QoE and also help the users while accessing the Web.

Network access control

Protocol Enhancement Proxy

Perform DNS requests on behalf of the user

The list above only enumerates some of the reliable Proxy usage that would provide value also in 2.0.

4. HTTP 2.0 Proxy

In HTTP 2.0 an interposed proxy should always be discoverable by the user-agent so that the user can consent it to stay or easily bypass it.

The actual discovery mechanism is not discussed in this draft.

4.1. HTTP 2.0 Proxy Features

"2.0 Proxy" will provide in additions to the usages listed in Section 4 also other specific HTTP 2.0 usages. Some of them are listed below.

Protocol version translation such as HTTP2.0 vs HTTP1.0/1.1 The HTTP 2.0 benefits will be valuable when it is used in mobile networks; however not all sites on Internet will support 2.0 from day one. The "2.0 Proxy" receives the HTTP 2.0 request by the user-agent and sends a new request to the Origin Server. If the Origin Server does not support HTTP 2.0 then the "2.0 Proxy" will translate the HTTP 2.0 request to HTTP 1.1 request and will then translate the HTTP 1.1 response to the HTTP 2.0 response.

Improve the HTTP/2.0 Flow Control management If both the user-agent and the origin server support HTTP 2.0, the 2.0 proxy can improve the Flow Control management as it is a hop-by-hop mechanism defined to protect endpoints that are operating under resource constraints.

Stream priority management A "2.0 Proxy" can improve the transmission order for streams based on knowledge of the network bandwidth-delay.

Push streams management A "2.0 Proxy" can adhere to network condition and apply push streams management policy. As an example a mobile user roaming can have policy saying not allowed push.

4.2. 2.0 Proxy: discovery mechanisms

The end user, using HTTP 2.0, should become always aware of the existence of any "2.0 Proxy" present in the network. The end user may also be entitled to explicitly consent to use it or to bypass it, although such entitlement may be limited for some applications or service environments.

The actual discovery mechanism is not discussed in this draft. The human factor implications of "proxy awareness" by the user are also not discussed.

5. Acknowledgments

The authors wish to thank Thorsten Herber, Preeyaa Rawlani, Bryan Sullivan for their invaluable comments.

6. References

6.1. Normative References

- [I-D.ietf-httpbis-http2]
Belshe, M., Peon, R., Thomson, M., and A. Melnikov,
"Hypertext Transfer Protocol version 2.0",
draft-ietf-httpbis-http2-06 (work in progress),
August 2013.
- [I-D.ietf-httpbis-pl-messaging]
Fielding, R. and J. Reschke, "Hypertext Transfer Protocol
(HTTP/1.1): Message Syntax and Routing",
draft-ietf-httpbis-pl-messaging-24 (work in progress),
September 2013.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate
Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC2616] Fielding, R., Gettys, J., Mogul, J., Frystyk, H.,
Masinter, L., Leach, P., and T. Berners-Lee, "Hypertext
Transfer Protocol -- HTTP/1.1", RFC 2616, June 1999.
- [RFC2817] Khare, R. and S. Lawrence, "Upgrading to TLS Within
HTTP/1.1", RFC 2817, May 2000.
- [RFC3986] Berners-Lee, T., Fielding, R., and L. Masinter, "Uniform
Resource Identifier (URI): Generic Syntax", STD 66,
RFC 3986, January 2005.

6.2. Informative References

- [RFC3040] Cooper, I., Melve, I., and G. Tomlinson, "Internet Web Replication and Caching Taxonomy", RFC 3040, January 2001.
- [RFC4732] Handley, M., Rescorla, E., and IAB, "Internet Denial-of-Service Considerations", RFC 4732, December 2006.

Authors' Addresses

Salvatore Loreto
Ericsson
Hirsalantie 11
Jorvas 02420
Finland

Email: salvatore.loreto@ericsson.com

Robert Skog
Ericsson
Sweeden

Email: robert.skog@ericsson.com

Hans Spaak
Ericsson
Sweeden

Email: hans.spaak@ericsson.com

Dan Druta
AT&T

Phone:
Email: dd5826@att.com

Mohammad Hafeez
AT&T

Phone:
Email: mh2897@att.com

