

An HTTP Status Code for Indicating Hints
draft-ietf-httpbis-early-hints-02

Abstract

This memo introduces an informational HTTP status code that can be used to convey hints that help a client make preparations for processing the final response.

Note to Readers

Discussion of this draft takes place on the HTTP working group mailing list (ietf-http-wg@w3.org), which is archived at <https://lists.w3.org/Archives/Public/ietf-http-wg/> .

Working Group information can be found at <https://httpwg.github.io/> ; source code and issues list for this draft can be found at <https://github.com/httpwg/http-extensions/labels/early-hints> .

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 November 17, 2017.

Copyright Notice

Copyright (c) 2017 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.	Notational Conventions	3
2.	103 Early Hints	3
3.	Security Considerations	4
4.	IANA Considerations	4
5.	Acknowledgements	4
6.	Changes	4
6.1.	Since draft-ietf-httpbis-early-hints-01	4
6.2.	Since draft-ietf-httpbis-early-hints-00	4
7.	References	5
7.1.	Normative References	5
7.2.	Informative References	5
	Author's Address	5

[1.](#) Introduction

It is common for HTTP responses to contain links to external resources that need to be fetched prior to their use; for example, rendering HTML by a Web browser. Having such links available to the client as early as possible helps to minimize perceived latency.

The "preload" ([\[Preload\]](#)) link relation can be used to convey such links in the Link header field of an HTTP response. However, it is not always possible for an origin server to generate a response header block immediately after receiving a request. For example, the origin server might need to query a database before generating a response, or it might delegate a request to an upstream HTTP server running at a distant location.

The dilemma here is that even though it is preferable for an origin server to send some headers as soon as it receives a request, it cannot do so until the status code and the full headers of the final HTTP response are determined.

HTTP/2 ([\[RFC7540\]](#)) server push can be used as a solution to this issue, but has its own limitations. The responses that can be pushed

Oku

Expires November 17, 2017

[Page 2]

using HTTP/2 are limited to those belonging to the same origin. Also, it is impossible to send only the links using server push. Finally, sending HTTP responses for every resource is an inefficient way of using bandwidth, especially when a caching server exists as an intermediary.

This memo defines a status code for sending an informational response ([\[RFC7231\]](#), [section 6.2](#)) that contains headers that are likely to be included in the final response. A server can send the informational response containing some of the headers to help the client start making preparations for processing the final response, and then run time-consuming operations to generate the final response. The informational response can also be used by an origin server to trigger HTTP/2 server push at a caching intermediary.

1.1. 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. 103 Early Hints

The 103 (Early Hints) informational status code indicates the client that the server is likely to send a final response with the headers included in the informational response.

A server **MUST NOT** include Content-Length, Transfer-Encoding, or any hop-by-hop header fields ([\[RFC7230\]](#), [section 6.1](#)) in a 103 (Early Hints) response.

A client **MAY** speculatively evaluate the headers included in a 103 (Early Hints) response while waiting for the final response. For example, a client might recognize a Link header field value containing the relation type "preload" and start fetching the target resource.

However, this **MUST NOT** affect how the final response is processed; when handling it, the client **MUST** behave as if it had not seen the informational response. In particular, a client **MUST NOT** process the headers included in the final response as if they belonged to the informational response, or vice versa.

An intermediary **MAY** drop the informational response. It **MAY** send HTTP/2 ([\[RFC7540\]](#)) server pushes using the information found in the 103 (Early Hints) response.

Oku

Expires November 17, 2017

[Page 3]

3. Security Considerations

Some clients may have issues handling 103 (Early Hints), since informational responses are rarely used in reply to requests not including an Expect header ([\[RFC7231\]](#), [section 5.1.1](#)).

In particular, an HTTP/1.1 client that mishandles an informational response as a final response is likely to consider all responses to the succeeding requests sent over the same connection to be part of the final response. Such behavior may constitute a cross-origin information disclosure vulnerability in case the client multiplexes requests to different origins onto a single persistent connection.

Therefore, a server might refrain from sending Early Hints over HTTP/1.1 unless when the client is known to handle informational responses correctly.

HTTP/2 clients are less likely to suffer from incorrect framing since handling of the response headers does not affect how the end of the response body is determined.

4. IANA Considerations

The HTTP Status Codes Registry will be updated with the following entry:

- o Code: 103
- o Description: Early Hints
- o Specification: [this document]

5. Acknowledgements

Thanks to Tatsuhiro Tsujikawa for coming up with the idea of sending the link headers using an informational response.

6. Changes

6.1. Since [draft-ietf-httpbis-early-hints-01](#)

- o Editorial changes.

6.2. Since [draft-ietf-httpbis-early-hints-00](#)

- o Forbid processing the headers of a 103 response as part of the informational response.

7. References

7.1. Normative References

- [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>>.
- [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>>.
- [RFC7231] Fielding, R., Ed. and J. Reschke, Ed., "Hypertext Transfer Protocol (HTTP/1.1): Semantics and Content", [RFC 7231](#), DOI 10.17487/RFC7231, June 2014, <<http://www.rfc-editor.org/info/rfc7231>>.
- [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

- [Preload] Grigorik, I., "Preload", September 2016, <<https://w3c.github.io/preload/>>.

Author's Address

Kazuho Oku
DeNA Co., Ltd.

Email: kazuhooku@gmail.com

