Internet Engineering Task Force (IETF)
Request for Comments: 7538
Obsoletes: 7238
Category: Standards Track
ISSN: 2070-1721

J. Reschke
greenbytes
April 2015

The Hypertext Transfer Protocol Status Code 308 (Permanent Redirect)

Abstract

This document specifies the additional Hypertext Transfer Protocol (HTTP) status code 308 (Permanent Redirect).

Status of This Memo

This is an Internet Standards Track document.

This document is a product of the Internet Engineering Task Force (IETF). It represents the consensus of the IETF community. It has received public review and has been approved for publication by the Internet Engineering Steering Group (IESG). Further information on Internet Standards is available in Section 2 of RFC 5741.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at http://www.rfc-editor.org/info/rfc7538[1].

Copyright Notice

Copyright © 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[2]) 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.

[1] http://www.rfc-editor.org/info/rfc7538
[2] http://trustee.ietf.org/license-info

Table of Contents

1 Introduction **3**

2 Notational Conventions **4**

3 308 Permanent Redirect **5**

4 Deployment Considerations **6**

5 Security Considerations **7**

6 IANA Considerations **8**

7 References **9**

7.1 Normative References 9

7.2 Informative References 9

Author's Address **11**

1. Introduction

HTTP defines a set of status codes for the purpose of redirecting a request to a different URI ([RFC3986]). The history of these status codes is summarized in Section 6.4 of [RFC7231], which also classifies the existing status codes into four categories.

The first of these categories contains the status codes 301 (Moved Permanently), 302 (Found), and 307 (Temporary Redirect), which can be classified as below:

	Permanent	**Temporary**
Allows changing the request method from POST to GET	301	302
Does not allow changing the request method from POST to GET	-	307

Section 6.4.7 of [RFC7231] states that it does not define a permanent variant of status code 307; this specification adds the status code 308, defining this missing variant (Section 3).

This specification contains no technical changes from the Experimental RFC 7238, which it obsoletes.

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].

3. 308 Permanent Redirect

The *308 (Permanent Redirect)* status code indicates that the target resource has been assigned a new permanent URI and any future references to this resource ought to use one of the enclosed URIs. Clients with link editing capabilities ought to automatically re-link references to the effective request URI (Section 5.5 of [RFC7230]) to one or more of the new references sent by the server, where possible.

The server SHOULD generate a Location header field ([RFC7231], Section 7.1.2) in the response containing a preferred URI reference for the new permanent URI. The user agent MAY use the Location field value for automatic redirection. The server's response payload usually contains a short hypertext note with a hyperlink to the new URI(s).

A 308 response is cacheable by default; i.e., unless otherwise indicated by the method definition or explicit cache controls (see [RFC7234], Section 4.2.2).

> **Note:** This status code is similar to 301 (Moved Permanently) ([RFC7231], Section 6.4.2), except that it does not allow changing the request method from POST to GET.

4. Deployment Considerations

Section 6 of [RFC7231] requires recipients to treat unknown 3xx status codes the same way as status code 300 (Multiple Choices) ([RFC7231], Section 6.4.1). Thus, servers will not be able to rely on automatic redirection happening similar to status codes 301, 302, or 307.

Therefore, the use of status code 308 is restricted to cases where the server has sufficient confidence in the client's understanding the new code or when a fallback to the semantics of status code 300 is not problematic. Server implementers are advised not to vary the status code based on characteristics of the request, such as the User-Agent header field ("User-Agent Sniffing") — doing so usually results in code that is both hard to maintain and hard to debug and would also require special attention to caching (i.e., setting a "Vary" response header field, as defined in Section 7.1.4 of [RFC7231]).

Note that many existing HTML-based user agents will emulate a refresh when encountering an HTML <meta> refresh directive ([HTML], Section 4.2.5.3). This can be used as another fallback. For example:

Client request:

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

Server response:

```
HTTP/1.1 308 Permanent Redirect
Content-Type: text/html; charset=UTF-8
Location: http://example.com/new
Content-Length: 356

<!DOCTYPE HTML>
<html>
   <head>
      <title>Permanent Redirect</title>
      <meta http-equiv="refresh"
            content="0; url=http://example.com/new">
   </head>
   <body>
      <p>
         The document has been moved to
         <a href="http://example.com/new"
         >http://example.com/new</a>.
      </p>
   </body>
</html>
```

5. Security Considerations

All security considerations that apply to HTTP redirects apply to the 308 status code as well (see Section 9 of [RFC7231]).

Unsecured communication over the Internet is subject to man-in-the-middle modification of messages, including changing status codes or redirect targets. Use of Transport Layer Security (TLS) is one way to mitigate those attacks. See Section 9 of [RFC7230] for related attacks on authority and message integrity.

6. IANA Considerations

The "Hypertext Transfer Protocol (HTTP) Status Code Registry" (defined in Section 8.2 of [RFC7231] and located at <http://www.iana.org/assignments/http-status-codes>) has been updated to reference this specification.

Value	Description	Reference
308	Permanent Redirect	Section 3 of this specification

7. References

7.1. 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>.

[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., Ed. and J. Reschke, Ed., "Hypertext Transfer Protocol (HTTP/1.1): Message Syntax and Routing", RFC 7230, 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, June 2014, <http://www.rfc-editor.org/info/rfc7231>.

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

7.2. Informative References

[HTML] Hickson, I., Berjon, R., Faulkner, S., Leithead, T., Doyle Navara, E., O'Connor, E., and S. Pfeiffer, "HTML5", W3C Recommendation REC-html5-20141028, October 2014, <http://www.w3.org/TR/2014/REC-html5-20141028/>.
Latest version available at <http://www.w3.org/TR/html5/>.

Acknowledgements

The definition for the new status code 308 reuses text from the HTTP/1.1 definitions of status codes 301 and 307.

Furthermore, thanks to Ben Campbell, Cyrus Daboo, Adrian Farrell, Eran Hammer-Lahav, Bjoern Hoehrmann, Barry Leiba, Subramanian Moonesamy, Kathleen Moriarty, Peter Saint-Andre, Robert Sparks, and Roy Fielding for feedback on this document.

Author's Address

Julian F. Reschke
greenbytes GmbH
Hafenweg 16
Muenster, NW 48155
Germany
EMail: julian.reschke@greenbytes.de
URI: http://greenbytes.de/tech/webdav/