Network Working Group
Internet-Draft
Obsoletes: 2616 (if approved)
Updates: 2617 (if approved)
Intended status: Standards Track
Expires: May 20, 2009

R. Fielding, Ed.
Day Software
J. Gettys
One Laptop per Child
J. Mogul
HP
H. Frystyk
Microsoft
L. Masinter
Adobe Systems
P. Leach
Microsoft
T. Berners-Lee
W3C/MIT
Y. Lafon, Ed.
W3C
J. Reschke, Ed.
greenbytes

November 16, 2008

HTTP/1.1, part 7: Authentication
draft-ietf-httpbis-p7-auth-05

Status of this Memo

Abstract

The Hypertext Transfer Protocol (HTTP) is an application-level protocol for distributed, collaborative, hypermedia information systems. HTTP has been in use by the World Wide Web global information initiative since 1990. This document is Part 7 of the seven-part specification that defines the protocol referred to as "HTTP/1.1" and, taken together, obsoletes RFC 2616. Part 7 defines HTTP Authentication.

Editorial Note (To be removed by RFC Editor)

Discussion of this draft should take place on the HTTPBIS working group mailing list (ietf-http-wg@w3.org). The current issues list is at <http://tools.ietf.org/wg/httpbis/trac/report/11> and related documents (including fancy diffs) can be found at <http://tools.ietf.org/wg/httpbis/>.

The changes in this draft are summarized in Appendix B.6.

Table of Contents

1. Introduction

This document defines HTTP/1.1 access control and authentication. Right now it includes the extracted relevant sections of RFC 2616 with only minor changes. The intention is to move the general framework for HTTP authentication here, as currently specified in [RFC2617], and allow the individual authentication mechanisms to be defined elsewhere. This introduction will be rewritten when that occurs.

HTTP provides several OPTIONAL challenge-response authentication mechanisms which can be used by a server to challenge a client request and by a client to provide authentication information. The general framework for access authentication, and the specification of "basic" and "digest" authentication, are specified in "HTTP Authentication: Basic and Digest Access Authentication" [RFC2617]. This specification adopts the definitions of "challenge" and "credentials" from that specification.

1.1. Requirements

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].

An implementation is not compliant if it fails to satisfy one or more of the MUST or REQUIRED level requirements for the protocols it implements. An implementation that satisfies all the MUST or REQUIRED level and all the SHOULD level requirements for its

protocols is said to be "unconditionally compliant"; one that satisfies all the MUST level requirements but not all the SHOULD level requirements for its protocols is said to be "conditionally compliant."

2. Notational Conventions and Generic Grammar

This specification uses the ABNF syntax defined in Section 2.1 of [Part1].

The ABNF rules below are defined in other specifications:

```
  OWS           = <OWS, defined in [Part1], Section 2.2>


  challenge   = <challenge, defined in [RFC2617], Section 1.2>
  credentials = <credentials, defined in [RFC2617], Section 1.2>
```


3. Status Code Definitions

3.1. 401 Unauthorized

The request requires user authentication. The response MUST include a WWW-Authenticate header field (Section 4.4) containing a challenge applicable to the requested resource. The client MAY repeat the request with a suitable Authorization header field (Section 4.1). If the request already included Authorization credentials, then the 401 response indicates that authorization has been refused for those credentials. If the 401 response contains the same challenge as the prior response, and the user agent has already attempted authentication at least once, then the user SHOULD be presented the entity that was given in the response, since that entity might include relevant diagnostic information. HTTP access authentication is explained in "HTTP Authentication: Basic and Digest Access Authentication" [RFC2617].

3.2. 407 Proxy Authentication Required

This code is similar to 401 (Unauthorized), but indicates that the client must first authenticate itself with the proxy. The proxy MUST

return a Proxy-Authenticate header field (Section 4.2) containing a challenge applicable to the proxy for the requested resource. The client MAY repeat the request with a suitable Proxy-Authorization header field (Section 4.3). HTTP access authentication is explained in "HTTP Authentication: Basic and Digest Access Authentication" [RFC2617].

4. Header Field Definitions

This section defines the syntax and semantics of HTTP/1.1 header fields related to authentication.

4.1. Authorization

A user agent that wishes to authenticate itself with a server--usually, but not necessarily, after receiving a 401 response--does so by including an Authorization request-header field with the request. The field "Authorization" consists of credentials containing the authentication information of the user agent for the realm of the resource being requested.

```
Authorization   = "Authorization" ":" OWS Authorization-v
Authorization-v = credentials
```

HTTP access authentication is described in "HTTP Authentication:

Basic and Digest Access Authentication" [RFC2617]. If a request is authenticated and a realm specified, the same credentials SHOULD be valid for all other requests within this realm (assuming that the authentication scheme itself does not require otherwise, such as credentials that vary according to a challenge value or using synchronized clocks).

When a shared cache (see Section 9 of [Part6]) receives a request containing an Authorization field, it MUST NOT return the corresponding response as a reply to any other request, unless one of the following specific exceptions holds:

1. If the response includes the "s-maxage" cache-control directive, the cache MAY use that response in replying to a subsequent request. But (if the specified maximum age has passed) a proxy

cache MUST first revalidate it with the origin server, using the request-headers from the new request to allow the origin server to authenticate the new request. (This is the defined behavior for s-maxage.) If the response includes "s-maxage=0", the proxy MUST always revalidate it before re-using it.

2. If the response includes the "must-revalidate" cache-control directive, the cache MAY use that response in replying to a subsequent request. But if the response is stale, all caches MUST first revalidate it with the origin server, using the request-headers from the new request to allow the origin server to authenticate the new request.

3. If the response includes the "public" cache-control directive, it MAY be returned in reply to any subsequent request.

4.2. Proxy-Authenticate

The response-header field "Proxy-Authenticate" MUST be included as part of a 407 (Proxy Authentication Required) response. The field value consists of a challenge that indicates the authentication scheme and parameters applicable to the proxy for this Request-URI.

```
  Proxy-Authenticate   = "Proxy-Authenticate" ":" OWS
                         Proxy-Authenticate-v
  Proxy-Authenticate-v = 1#challenge
```

The HTTP access authentication process is described in "HTTP Authentication: Basic and Digest Access Authentication" [RFC2617]. Unlike WWW-Authenticate, the Proxy-Authenticate header field applies only to the current connection and SHOULD NOT be passed on to downstream clients. However, an intermediate proxy might need to obtain its own credentials by requesting them from the downstream

client, which in some circumstances will appear as if the proxy is forwarding the Proxy-Authenticate header field.

4.3. Proxy-Authorization

The request-header field "Proxy-Authorization" allows the client to identify itself (or its user) to a proxy which requires authentication. The Proxy-Authorization field value consists of

credentials containing the authentication information of the user agent for the proxy and/or realm of the resource being requested.

```
  Proxy-Authorization   = "Proxy-Authorization" ":" OWS
                          Proxy-Authorization-v
  Proxy-Authorization-v = credentials
```

The HTTP access authentication process is described in "HTTP Authentication: Basic and Digest Access Authentication" [RFC2617]. Unlike Authorization, the Proxy-Authorization header field applies only to the next outbound proxy that demanded authentication using the Proxy-Authenticate field. When multiple proxies are used in a chain, the Proxy-Authorization header field is consumed by the first outbound proxy that was expecting to receive credentials. A proxy MAY relay the credentials from the client request to the next proxy if that is the mechanism by which the proxies cooperatively authenticate a given request.

4.4. WWW-Authenticate

The WWW-Authenticate response-header field MUST be included in 401 (Unauthorized) response messages. The field value consists of at least one challenge that indicates the authentication scheme(s) and parameters applicable to the Request-URI.

```
  WWW-Authenticate   = "WWW-Authenticate" ":" OWS WWW-Authenticate-v
  WWW-Authenticate-v = 1#challenge
```

The HTTP access authentication process is described in "HTTP Authentication: Basic and Digest Access Authentication" [RFC2617]. User agents are advised to take special care in parsing the WWW-Authenticate field value as it might contain more than one challenge, or if more than one WWW-Authenticate header field is provided, the contents of a challenge itself can contain a comma-separated list of authentication parameters.

5. IANA Considerations

5.1. Message Header Registration

The Message Header Registry located at <http://www.iana.org/assignments/message-headers/message-header-index.html> should be updated with the permanent registrations below (see [RFC3864]):

Header Field Name	Protocol	Status	Reference
Authorization	http	standard	Section 4.1
Proxy-Authenticate	http	standard	Section 4.2
Proxy-Authorization	http	standard	Section 4.3
WWW-Authenticate	http	standard	Section 4.4

The change controller is: "IETF (iesg@ietf.org) - Internet Engineering Task Force".

6. Security Considerations

This section is meant to inform application developers, information providers, and users of the security limitations in HTTP/1.1 as described by this document. The discussion does not include definitive solutions to the problems revealed, though it does make some suggestions for reducing security risks.

6.1. Authentication Credentials and Idle Clients

Existing HTTP clients and user agents typically retain authentication information indefinitely. HTTP/1.1 does not provide a method for a server to direct clients to discard these cached credentials. This is a significant defect that requires further extensions to HTTP. Circumstances under which credential caching can interfere with the application's security model include but are not limited to:

- Clients which have been idle for an extended period following which the server might wish to cause the client to reprompt the user for credentials.
- Applications which include a session termination indication (such as a `logout' or `commit' button on a page) after which the server side of the application `knows' that there is no further reason for the client to retain the credentials.

This is currently under separate study. There are a number of work-arounds to parts of this problem, and we encourage the use of password protection in screen savers, idle time-outs, and other

methods which mitigate the security problems inherent in this problem. In particular, user agents which cache credentials are encouraged to provide a readily accessible mechanism for discarding cached credentials under user control.

7. Acknowledgments

[[anchor2: TBD.]]

8. References

8.1. Normative References

[Part1] Fielding, R., Ed., Gettys, J., Mogul, J., Frystyk, H., Masinter, L., Leach, P., Berners-Lee, T., Lafon, Y., Ed., and J. Reschke, Ed., "HTTP/1.1, part 1: URIs, Connections, and Message Parsing", draft-ietf-httpbis-p1-messaging-05 (work in progress), November 2008.

[Part6] Fielding, R., Ed., Gettys, J., Mogul, J., Frystyk, H., Masinter, L., Leach, P., Berners-Lee, T., Lafon, Y., Ed., and J. Reschke, Ed., "HTTP/1.1, part 6: Caching", draft-ietf-httpbis-p6-cache-05 (work in progress), November 2008.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC2617] Franks, J., Hallam-Baker, P., Hostetler, J., Lawrence, S., Leach, P., Luotonen, A., and L. Stewart, "HTTP Authentication: Basic and Digest Access Authentication", RFC 2617, June 1999.

8.2. Informative References

[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.

[RFC3864] Klyne, G., Nottingham, M., and J. Mogul, "Registration Procedures for Message Header Fields", BCP 90, RFC 3864, September 2004.

Appendix A. Compatibility with Previous Versions

A.1. Changes from RFC 2616

Appendix B. Change Log (to be removed by RFC Editor before publication)

B.1. Since RFC2616

Extracted relevant partitions from [RFC2616].

B.2. Since draft-ietf-httpbis-p7-auth-00

Closed issues:

- <http://tools.ietf.org/wg/httpbis/trac/ticket/35>: "Normative and Informative references"

B.3. Since draft-ietf-httpbis-p7-auth-01

Ongoing work on ABNF conversion (<http://tools.ietf.org/wg/httpbis/trac/ticket/36>):

- Explicitly import BNF rules for "challenge" and "credentials" from RFC2617.

- Add explicit references to BNF syntax and rules imported from other parts of the specification.

B.4. Since draft-ietf-httpbis-p7-auth-02

Ongoing work on IANA Message Header Registration (<http://tools.ietf.org/wg/httpbis/trac/ticket/40>):

- Reference RFC 3984, and update header registrations for headers defined in this document.

B.5. Since draft-ietf-httpbis-p7-auth-03

B.6. Since draft-ietf-httpbis-p7-auth-04

Ongoing work on ABNF conversion (<http://tools.ietf.org/wg/httpbis/trac/ticket/36>):

- Use "/" instead of "|" for alternatives.
- Introduce new ABNF rules for "bad" whitespace ("BWS"), optional whitespace ("OWS") and required whitespace ("RWS").

- Rewrite ABNFs to spell out whitespace rules, factor out header value format definitions.

Index

Authors' Addresses

Roy T. Fielding (editor)
Day Software
23 Corporate Plaza DR, Suite 280
Newport Beach, CA 92660
USA

Phone: +1-949-706-5300
Fax: +1-949-706-5305
Email: fielding@gbiv.com
URI: http://roy.gbiv.com/

Jim Gettys
One Laptop per Child
21 Oak Knoll Road
Carlisle, MA 01741
USA

Email: jg@laptop.org
URI: http://www.laptop.org/

Jeffrey C. Mogul

Hewlett-Packard Company
HP Labs, Large Scale Systems Group
1501 Page Mill Road, MS 1177
Palo Alto, CA 94304
USA

Email: JeffMogul@acm.org

Henrik Frystyk Nielsen
Microsoft Corporation
1 Microsoft Way
Redmond, WA 98052
USA

Email: henrikn@microsoft.com

Larry Masinter
Adobe Systems, Incorporated
345 Park Ave
San Jose, CA 95110
USA

Email: LMM@acm.org
URI: http://larry.masinter.net/

Paul J. Leach
Microsoft Corporation
1 Microsoft Way
Redmond, WA 98052

Email: paulle@microsoft.com

Tim Berners-Lee
World Wide Web Consortium
MIT Computer Science and Artificial Intelligence Laboratory
The Stata Center, Building 32
32 Vassar Street
Cambridge, MA 02139
USA

Email: timbl@w3.org
URI: http://www.w3.org/People/Berners-Lee/

Yves Lafon (editor)
World Wide Web Consortium
W3C / ERCIM
2004, rte des Lucioles
Sophia-Antipolis, AM 06902
France

Email: ylafon@w3.org
URI: http://www.raubacapeu.net/people/yves/

Julian F. Reschke (editor)
greenbytes GmbH
Hafenweg 16
Muenster, NW 48155
Germany

Phone: +49 251 2807760
Fax: +49 251 2807761
Email: julian.reschke@greenbytes.de
URI: http://greenbytes.de/tech/webdav/

Full Copyright Statement

Intellectual Property