HTTPAuth Working Group
Internet-Draft
Updates: 2617 (if approved)
Intended status: Experimental
Expires: August 8, 2014

J. Reschke
greenbytes

February 4, 2014

An Encoding Parameter for HTTP Basic Authentication
draft-ietf-httpauth-basicauth-enc-02

Abstract

The "Basic" authentication scheme defined in RFC 2617 does not properly define how to treat non-ASCII characters. This has led to a situation where user agent implementations disagree, and servers make different assumptions based on the locales they are running in. There is little interoperability for the non-ASCII characters in the ISO-8859-1 character repertoire, and even less interoperability for any characters beyond that.

This document defines a backwards-compatible extension to "Basic", specifying the server's character encoding scheme expectation, using a new authentication scheme parameter.

Editorial Note (To be removed by RFC Editor before publication)

Discussion of this draft takes place on the HTTPAuth working group mailing list (http-auth@ietf.org), which is archived at <http://www.ietf.org/mail-archive/web/http-auth/current/maillist.html>.

XML versions, latest edits and the issues list for this document are available from <http://greenbytes.de/tech/webdav/#draft-ietf-httpauth-basicauth-enc>.

The changes in this draft are summarized in Appendix C.11.

The contents of this document will likely be included into the new specification of the "Basic" scheme, see <http://tools.ietf.org/html/draft-ietf-httpauth-basicauth-update>.

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 August 8, 2014.

Copyright Notice

Table of Contents

1. Introduction

The "Basic" authentication scheme defined in Section 2 of [RFC2617] does not properly define how to treat non-ASCII characters ([USASCII]): it uses the Base64 ([RFC4648], Section 4) encoding of the concatenation of username, separator character, and password without stating which character encoding scheme to use.

This has led to a situation where user agent implementations disagree, and servers make different assumptions based on the locales they are running in. There is little interoperability for the non-ASCII characters in the ISO-8859-1 character repertoire ([USASCII], [ISO-8859-1]), and even less interoperability for any characters beyond that.

This document defines a backwards-compatible extension to "Basic", specifying the server's character encoding scheme expectation, using a new auth-param for use in the Proxy-Authenticate and WWW-Authenticate header fields, as defined in [draft-ietf-httpbis-p7-auth].

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].

The terms (character) repertoire and character encoding scheme are defined in Section 2 of [RFC6365].

3. The 'charset' auth-param

In challenges, servers MAY use the "charset" authentication parameter (case-insensitive) to indicate the character encoding scheme they expect the user agent to use when generating "user-pass" (a sequence of octets) from "userid" and "password" ([RFC2617], Section 2).

The only allowed value is "UTF-8", to be matched case-insensitively (see [RFC2978], Section 2.3), indicating that the server expects the UTF-8 character encoding scheme to be used ([RFC3629]).

Other values are reserved for future use.

> Note: The 'charset' parameter cannot be included when sending credentials (e.g. in the Authorization or Proxy-Authorization header fields), as the "Basic" scheme uses a single token for credentials ('token68' syntax), not a parameter list ('#auth-param' syntax); see Section 2.1 of [draft-ietf-httpbis-p7-auth].

> Note: The name 'charset' has been chosen for consistency with Section 2.1.1 of [RFC2831]. A better name would have been 'accept-charset', as it is not about the message it appears in, but the server's expectation.

4. Example

In the example below, the server prompts for authentication in the "foo" realm, using Basic authentication, with a preference for the UTF-8 character encoding scheme:

```
WWW-Authenticate: Basic realm="foo", charset="UTF-8"
```

Note that the parameter value can be either a token or a quoted string; in this case the server chose to use the quoted-string notation.

The user's name is "test", and his password is the string "123" followed by the Unicode character U+00A3 (POUND SIGN). Following Section 1.2 of [RFC2617], but using the character encoding scheme UTF-8, the user-pass, converted to a sequence of octets, is:

```
 't' 'e' 's' 't' ':' '1' '2' '3' pound
 74  65  73  74  3A  31  32  33  C2  A3
```

Encoding this octet sequence in Base64 ([RFC4648], Section 4) yields:

```
  dGVzdDoxMjPCow==
```

Thus the Authorization header field would be:

```
  Authorization: Basic dGVzdDoxMjPCow==
```

Or, for proxy authentication:

```
  Proxy-Authorization: Basic dGVzdDoxMjPCow==
```

5. Security Considerations

This document does not introduce any new security considerations beyond those defined for the "Basic" authentication scheme ([RFC2617], Section 4), and those applicable to the handling of UTF-8 ([RFC3629], Section 10).

6. IANA Considerations

There are no IANA Considerations related to this specification.

7. Acknowledgements

The internationalisation problem has been reported as a Mozilla bug back in the year 2000 (see <https://bugzilla.mozilla.org/show_bug.cgi?id=41489> and also the more recent <https://bugzilla.mozilla.org/show_bug.cgi?id=656213>). It was Andrew Clover's idea to address it using a new auth-param.

Thanks to Stephen Farrell, Bjoern Hoehrmann, Amos Jeffries, James Manger, Yaron Sheffer, and Martin Thomson for providing feedback on this document.

8. References

8.1. Normative References

[ISO-8859-1] International Organization for Standardization, "Information technology -- 8-bit single-byte coded graphic character sets -- Part 1: Latin alphabet No. 1", ISO/IEC 8859-1:1998, 1998.

[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.

[RFC2978] Freed, N. and J. Postel, "IANA Charset Registration Procedures", BCP 19, RFC 2978, October 2000.

[RFC3629] Yergeau, F., "UTF-8, a transformation format of ISO 10646", STD 63, RFC 3629, November 2003.

[RFC6365] Hoffman, P. and J. Klensin, "Terminology Used in Internationalization in the IETF", BCP 166, RFC 6365, September 2011.

[USASCII] American National Standards Institute, "Coded Character Set -- 7-bit American

Standard Code for Information Interchange", ANSI X3.4, 1986.

[draft-ietf-httpbis-p7-auth] Fielding, R., Ed. and J. Reschke, Ed., "Hypertext Transfer Protocol (HTTP/1.1): Authentication", draft-ietf-httpbis-p7-auth-25 (work in progress), November 2013.

8.2. Informative References

[RFC2831] Leach, P. and C. Newman, "Using Digest Authentication as a SASL Mechanism", RFC 2831, May 2000.

[RFC4648] Josefsson, S., "The Base16, Base32, and Base64 Data Encodings", RFC 4648, October 2006.

[XHR] van Kesteren, A., Steen, H., Aubourg, J., and J. Song, "XMLHttpRequest Level 1", W3C Working Draft WD-XMLHttpRequest-20140130, January 2014, <http://www.w3.org/TR/2014/WD-XMLHttpRequest-20140130/>.

Latest version available at <http://www.w3.org/TR/XMLHttpRequest/>.

Appendix A. Deployment Considerations

A.1. User Agents

User agents not implementing this specification should continue to work as before, ignoring the new parameter.

User agents which already default to the UTF-8 encoding implement this specification by definition. Note that some user agents also have different defaults depending on whether the request originates from page navigation as opposed to a script-driven request using XMLHttpRequest [XHR].

Other user agents can keep their default behavior, and switch to UTF-8 when seeing the new parameter.

A.1.1. Alternative approach

On the other hand, the strategy below may already improve the user-visible behavior today:

- In the first authentication request, choose the character encoding scheme based on the user's credentials: if they do not need any characters outside the ISO-8859-1 character repertoire, default to ISO-8859-1, otherwise use UTF-8.

- If the first attempt failed and the encoding used was ISO-8859-1, retry once with UTF-8 encoding instead.

Note that there's a risk if the site blocks an account after multiple login failures (for instance, when it doesn't reset the counter after a successful login).

A.2. Origin Servers

Origin servers that do not support non-ASCII characters in credentials do not require any changes.

Origin servers that need to support non-ASCII characters, but can't use the UTF-8 encoding will not be affected; they will continue to function as well or as badly as before.

Finally, origin servers that need to support non-ASCII characters and can use the UTF-8 encoding can opt in as described above. In the worst case, they'll continue to see either broken credentials or no credentials at all (depending on how legacy clients handle characters they can not encode).

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

B.1. Why not simply switch the default encoding to UTF-8?

There are sites in use today that default to a locale encoding, such as ISO-8859-1, and expect user agents to use that encoding. These sites will break if the user agent uses a different encoding, such as UTF-8.

B.2. What about Digest?

The Digest scheme has similar issues with respect to internationalization. The HTTPAuth Working Group is chartered to address this problem as well, and the solution might be very similar.

B.3. Will existing UAs ignore the parameter?

It appears they will. See <http://greenbytes.de/tech/tc/httpauth/#simplebasicnewparam1> and <http://greenbytes.de/tech/tc/httpauth/#simplebasicnewparam2>.

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

C.1. Since draft-reschke-basicauth-enc-00

Add and close issues "credparam" and "paramcase". Rewrite the deployment considerations.

C.2. Since draft-reschke-basicauth-enc-01

Note more recent Mozilla bugzilla entry; add behavior of existing UAs to FAQ (with pointer to test cases).

C.3. Since draft-reschke-basicauth-enc-02

Add and resolve issue "xhrutf8".

C.4. Since draft-reschke-basicauth-enc-03

Add and resolve issue "proxy".

C.5. Since draft-reschke-basicauth-enc-04

Add and resolve issues "paramname" and "sentparam". Add issues "terminology" and "unorm". Update HTTPbis reference.

C.6. Since draft-reschke-basicauth-enc-05

Update HTTPbis reference.

C.7. Since draft-reschke-basicauth-enc-06

Update HTTPbis and XHR references.

C.8. Since draft-reschke-basicauth-enc-07

"b64token" -> "token68" (ABNF term changed in HTTPbis P7). Change contact information from HTTPbis WG to HTTPAUTH WG. Add issue parmname2831. Changed intended status to "experimental".

C.9. Since draft-reschke-basicauth-enc-08

Made it a draft of the IETF HTTPauth Working Group.

C.10. Since draft-ietf-httpauth-basicauth-enc-00

Clarify what encoding step the charset selection applies to.

Use RFC 6365 terminology.

Rename the parameter to "charset" for consistency with RFC 2831.

C.11. Since draft-ietf-httpauth-basicauth-enc-01

Update httpbis and XHR references. Add a note about draft-ietf-httpauth-basicauth-update.

Appendix D. Open issues (to be removed by RFC Editor prior to publication)

D.1. edit

Type: edit

julian.reschke@greenbytes.de (2010-08-11): Umbrella issue for editorial fixes/enhancements.

D.2. unorm

Type: edit

julian.reschke@greenbytes.de (2012-02-02): We need a statement about unicode normalization forms.

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/