The Printer Working Group

August 3, 2017
White Paper

IPP Authentication Methods
(IPPAUTH)

Status: Initial

Abstract: This document is a whitepaper that describes the interaction between IPP and various authentication mechanisms used byIPP's HTTP and HTTPS transports, and how they might affect the authentication user experience on systems running an IPP Client.

This document is a White Paper. For a definition of a "White Paper", see: http://ftp.pwg.org/pub/pwg/general/pwg-process30.pdf

This document is available electronically at:

http://ftp.pwg.org/pub/pwg/ipp/whitepaper/tb-ippauth-20170802.odt
http://ftp.pwg.org/pub/pwg/ipp/whitepaper/tb-ippauth-20170802.pdf

Title: IPP Authentication Methods *(IPPAUTH)*

The material contained herein is not a license, either expressed or implied, to any IPR owned or controlled by any of the authors or developers of this material or the Printer Working Group. The material contained herein is provided on an “AS IS” basis and to the maximum extent permitted by applicable law, this material is provided AS IS AND WITH ALL FAULTS, and the authors and developers of this material and the Printer Working Group and its members hereby disclaim all warranties and conditions, either expressed, implied or statutory, including, but not limited to, any (if any) implied warranties that the use of the information herein will not infringe any rights or any implied warranties of merchantability or fitness for a particular purpose.

Table of Contents

List of Figures

List of Tables

1 Introduction

The Internet Printing Protocol (hereafter, IPP) uses HTTP as its underlying transport [RFC8010]. When an IPP Printer is configured to limit access to its services to only those Clients operated by an authorized User, IPP employs various different HTTP authentication methods. But since an IPP Client isn't usually a typical HTTP User Agent (e.g. it isn't a commonly used Web browser), some limits, constraints and conventions ought to be considered when implementing support for one of these different HTTP authentication methods.

2 Terminology

2.1 Protocol Roles Terminology

This document defines the following protocol roles in order to specify unambiguous conformance requirements:

Client: Initiator of outgoing IPP session requests and sender of outgoing IPP operation requests (Hypertext Transfer Protocol -- HTTP/1.1 [RFC7230] User Agent).

Printer: Listener for incoming IPP session requests and receiver of incoming IPP operation requests (Hypertext Transfer Protocol -- HTTP/1.1 [RFC7230] Server) that represents one or more Physical Devices or a Logical Device.

2.2 Other Terms Used in This Document

User: A person or automata using a Client to communicate with a Printer.

2.3 Acronyms and Organizations

IANA: Internet Assigned Numbers Authority, http://www.iana.org/

IETF: Internet Engineering Task Force, http://www.ietf.org/

ISO: International Organization for Standardization, http://www.iso.org/

PWG: Printer Working Group, http://www.pwg.org/

3 Rationale for IPP Authentication Methods

This white paper describes how various HTTP based authentication systems integrate into IPP communications between a Client and a Printer. Although the authentication protocols themselves do not need to change to be integrated into IPP communications, the IPP Client is not a Web browser, so some considerations must be made by IPP Client implementors. The “uri-authentication-supported” attribute [RFC8011] Printer Description attribute indicates the authentication systems supported by the Printer.

3.1 Client Authentication Methods

The “uri-authentication-supported” attribute [RFC8011] indicates the authentication method used for a corresponding URI in “printer-uri-supported”. A Printer uses the identity to authorize access to capabilities such as operations, resources, and attributes. As in most other contexts, authentication is the process of establishing that an entity claiming to have a particular identity is who they say they are.

Each of the authentication method keywords currently registered for “uri-authentication-supported” is described below, with an accompanying sequence diagram for illustration purposes.

3.1.1 The 'none' IPP Authentication Method

The 'none' IPP Authentication Method [RFC8011] very simply indicates that the receiving Printer is provided no method whatsoever to determine the identity of the User who is operating the Client that is making IPP operation requests. The user name for the operation is assumed to be 'anonymous'.

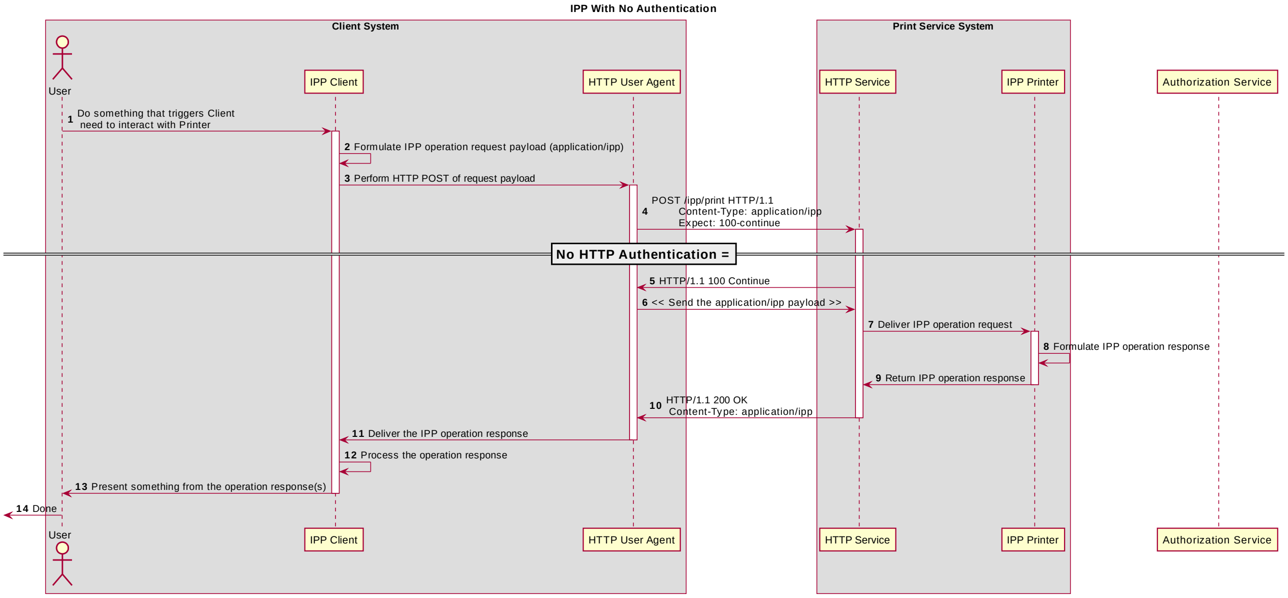

IPP With No Authentication
Client System
Print Service System
User
IPP Client
HTTP User Agent
HTTP Service
IPP Printer
Authorization Service
1 Do something that triggers Client need to interact with Printer
2 Formulate IPP operation request payload (application/ipp)
3 Perform HTTP POST of request payload
4 POST /ipp/print HTTP/1.1 Content-Type: application/ipp Expect: 100-continue
No HTTP Authentication =
5 HTTP/1.1 100 Continue
6 << Send the application/ipp payload >>
7 Deliver IPP operation request
8 Formulate IPP operation response
9 Return IPP operation response
10 HTTP/1.1 200 OK Content-Type: application/ipp
11 Deliver the IPP operation response
12 Process the operation response
13 Present something from the operation response(s)
14 Done
User
IPP Client
HTTP User Agent
HTTP Service
IPP Printer
Authorization Service

Figure 3.1: Sequence diagram for the 'none' IPP Authentication Method

This method is not recommended unless the Printer's operator has the objective of providing an anonymous print service. In most cases, the Client SHOULD provide the “requesting-user-name” operation attribute, as described in section 3.1.2.

3.1.2 The 'requesting-user-name' IPP Authentication Method

In the 'requesting-user-name' IPP Authentication Method [RFC8011], the Client MUST provides the “requesting-user-name” operation attribute [RFC8011] in its IPP operation request. The Printer uses this unauthenticated name as the identity of the actor operating the Client.

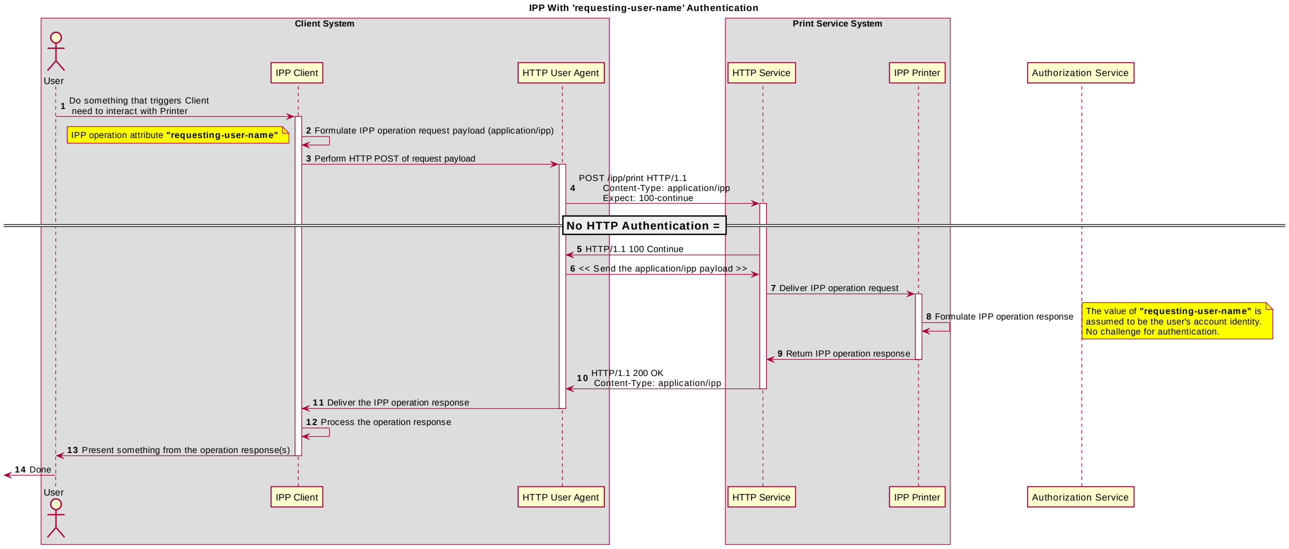
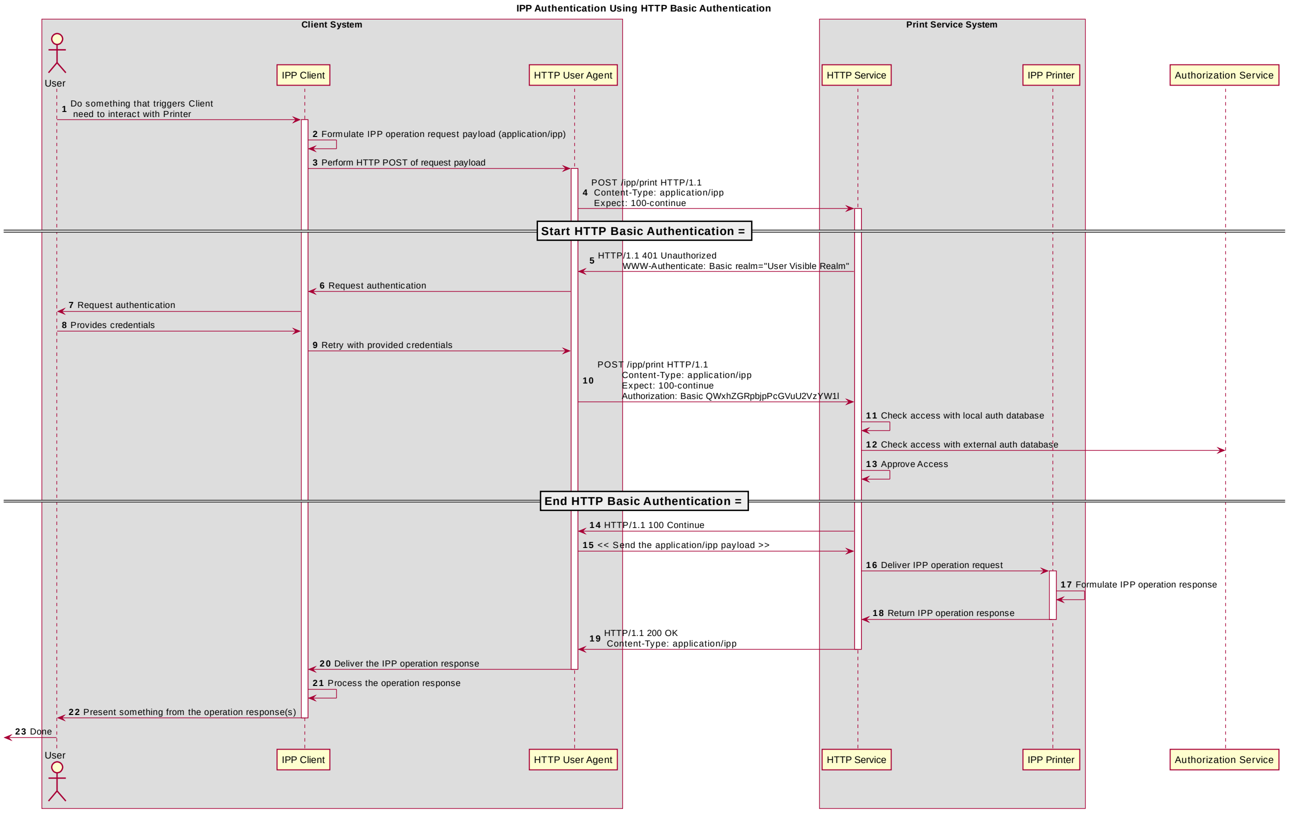

IPP With 'requesting-user-name' Authentication
Client System
Print Service System
User
IPP Client
HTTP User Agent
HTTP Service
IPP Printer
Authorization Service
1 Do something that triggers Client need to interact with Printer
IPP operation attribute "requesting-user-name"
2 Formulate IPP operation request payload (application/ipp)
3 Perform HTTP POST of request payload
4 POST /ipp/print HTTP/1.1 Content-Type: application/ipp Expect: 100-continue
No HTTP Authentication =
5 HTTP/1.1 100 Continue
6 << Send the application/ipp payload >>
7 Deliver IPP operation request
8 Formulate IPP operation response
The value of "requesting-user-name" is assumed to be the user's account identity. No challenge for authentication.
9 Return IPP operation response
10 HTTP/1.1 200 OK Content-Type: application/ipp
11 Deliver the IPP operation response
12 Process the operation response
13 Present something from the operation response(s)
14 Done
User
IPP Client
HTTP User Agent
HTTP Service
IPP Printer
Authorization Service

Figure 3.2: Sequence diagram for the 'requesting-user-name' IPP Authentication Method

This method is not recommended since there is no actual authentication performed as there is no credential provided to prove the identity claimed in the “requesting-user-name”.

3.1.3 The 'basic' IPP Authentication Method

The 'basic' IPP Authentication Method uses HTTP “basic” authentication scheme [RFC7617]. It is employed in IPP in much the same way that it is employed in conventional HTTP workflows using a Web browser; when the IPP Client encounters an HTTP 401 Unauthorized response, it evaluates whether it supports the authentication method identified by the value of the “WWW-Authenticated” header in the response. In this case, if it supports 'basic', it will present UI asking the User to provide username and password credentials that may be used to authenticate with the HTTP Server providing access to the IPP Printer. If the HTTP Server successfully authenticates that set of credentials, then the IPP operation request is passed on to the IPP Printer, which responds as usual.



IPP Authentication Using HTTP Basic Authentication
Client System
Print Service System
User
IPP Client
HTTP User Agent
HTTP Service
IPP Printer
Authorization Service
1 Do something that triggers Client need to interact with Printer
2 Formulate IPP operation request payload (application/ipp)
3 Perform HTTP POST of request payload
4 POST /ipp/print HTTP/1.1 Content-Type: application/ipp Expect: 100-continue
Start HTTP Basic Authentication =
5 HTTP/1.1 401 Unauthorized WWW-Authenticate: Basic realm="User Visible Realm"
6 Request authentication
7 Request authentication
8 Provides credentials
9 Retry with provided credentials
10 POST /ipp/print HTTP/1.1 Content-Type: application/ipp Expect: 100-continue Authorization: Basic QWxhZGRpbjpPcGVuU2VzYW1l
11 Check access with local auth database
12 Check access with external auth database
13 Approve Access
End HTTP Basic Authentication =
14 HTTP/1.1 100 Continue
15 << Send the application/ipp payload >>
16 Deliver IPP operation request
17 Formulate IPP operation response
18 Return IPP operation response
19 HTTP/1.1 200 OK Content-Type: application/ipp
20 Deliver the IPP operation response
21 Process the operation response
22 Present something from the operation response(s)
23 Done
User
IPP Client
HTTP User Agent
HTTP Service
IPP Printer
Authorization Service

Figure 3.3 : Sequence diagram for the 'basic' IPP Authentication Method

3.1.4 The 'digest' IPP Authentication Method

The 'digest' IPP Authentication method uses the HTTP “digest” authentication scheme [RFC7616]. It is employed in IPP in much the same way that it is employed in conventional HTTP workflows using a Web browser; when the IPP Client encounters an HTTP 401 Unauthorized response, it evaluates whether it supports the authentication method identified by the value of the “WWW-Authenticated” header in the response. In this case, if it supports 'digest', it will present UI asking the User to provide username and password credentials that may be used to authenticate with the HTTP Server providing access to the IPP Printer. If the HTTP Server successfully authenticates that set of credentials, then the IPP operation request is passed on to the IPP Printer, which responds as usual.

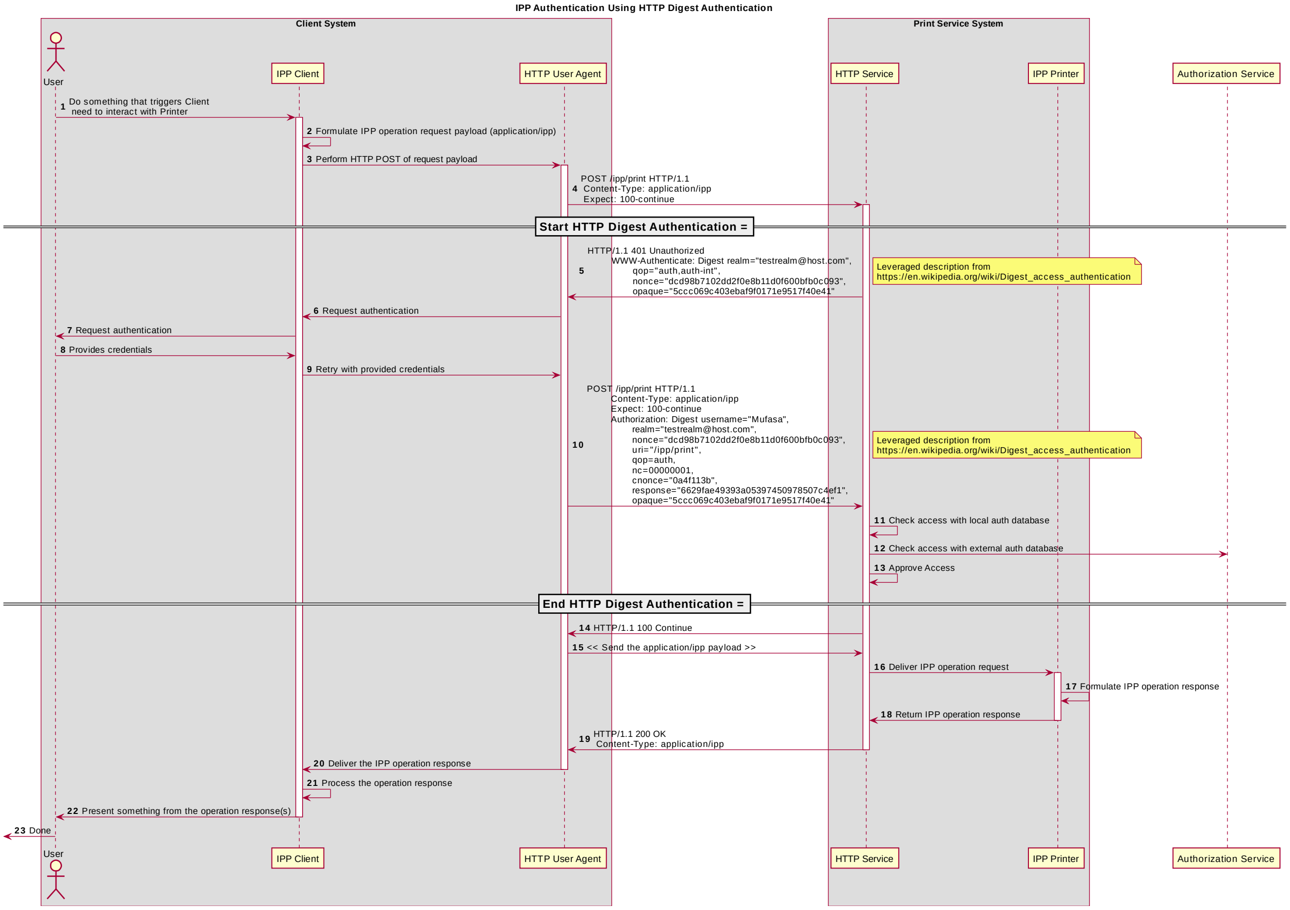

IPP Authentication Using HTTP Digest Authentication
Client System
Print Service System
User
IPP Client
HTTP User Agent
HTTP Service
IPP Printer
Authorization Service
1 Do something that triggers Client need to interact with Printer
2 Formulate IPP operation request payload (application/ipp)
3 Perform HTTP POST of request payload
4 POST /ipp/print HTTP/1.1
Content-Type: application/ipp
Expect: 100-continue
Start HTTP Digest Authentication =
5 HTTP/1.1 401 Unauthorized
WWW-Authenticate: Digest realm="testrealm@host.com",
qop="auth,auth-int",
nonce="dcd98b7102dd2f0e8b11d0f600bfb0c093",
opaque="5ccc069c403ebaf9f0171e9517f40e41"
Leveraged description from https://en.wikipedia.org/wiki/Digest_access_authentication
6 Request authentication
7 Request authentication
8 Provides credentials
9 Retry with provided credentials
10 POST /ipp/print HTTP/1.1
Content-Type: application/ipp
Expect: 100-continue
Authorization: Digest username="Mufasa",
realm="testrealm@host.com",
nonce="dcd98b7102dd2f0e8b11d0f600bfb0c093",
uri="/ipp/print",
qop=auth,
nc=00000001,
cnonce="0a4f113b",
response="6629fae49393a05397450978507c4ef1",
opaque="5ccc069c403ebaf9f0171e9517f40e41"
Leveraged description from https://en.wikipedia.org/wiki/Digest_access_authentication
11 Check access with local auth database
12 Check access with external auth database
13 Approve Access
End HTTP Digest Authentication =
14 HTTP/1.1 100 Continue
15 << Send the application/ipp payload >>
16 Deliver IPP operation request
17 Formulate IPP operation response
18 Return IPP operation response
19 HTTP/1.1 200 OK
Content-Type: application/ipp
20 Deliver the IPP operation response
21 Process the operation response
22 Present something from the operation response(s)
23 Done
User
IPP Client
HTTP User Agent
HTTP Service
IPP Printer
Authorization Service

Figure 3.4 : Sequence diagram for the 'digest' IPP Authentication Method

124 3.1.5 The 'negotiate' IPP Authentication Method

125 The 'negotiate' IPP Authentication method uses the HTTP “negotiate” authentication
126 scheme [RFC4559].

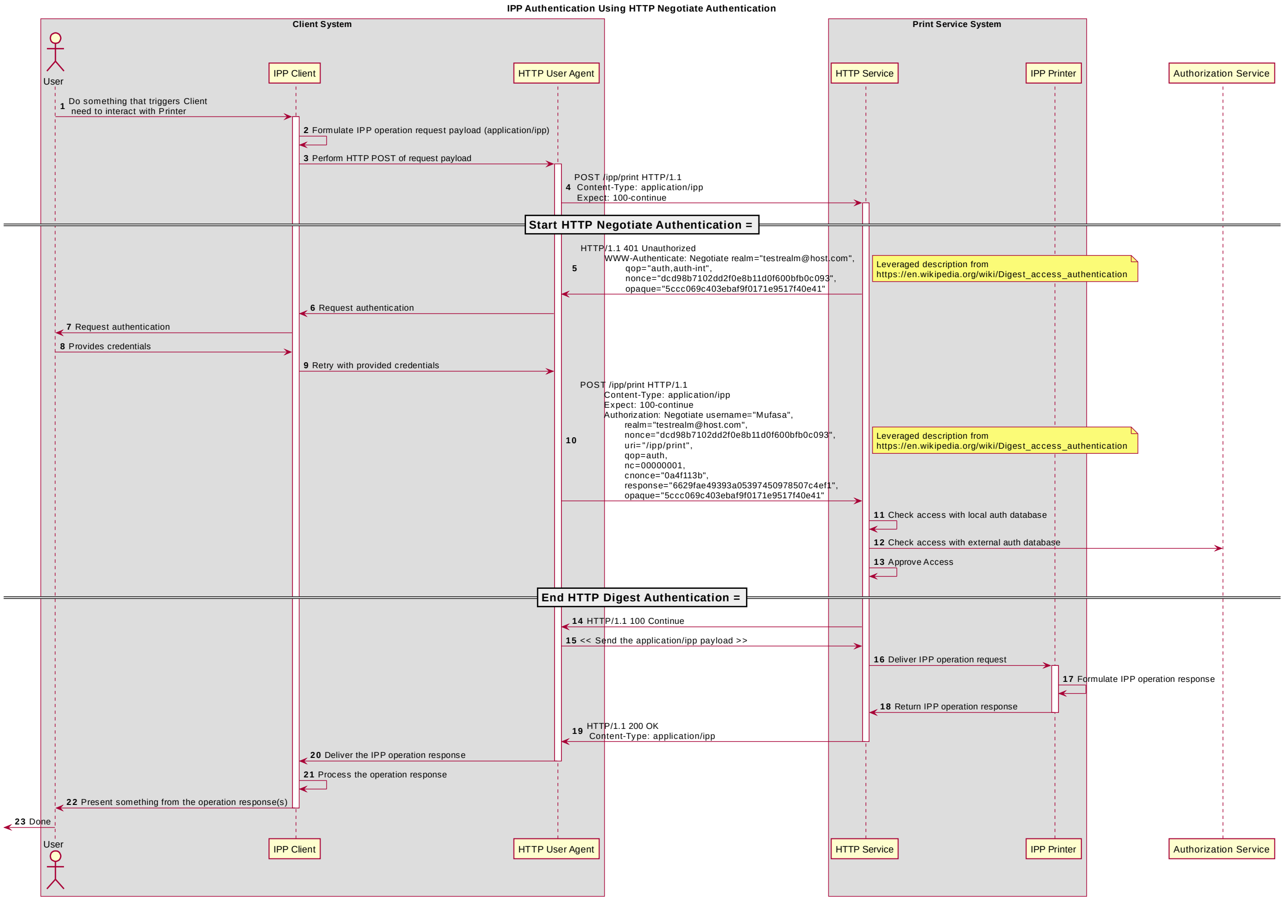

IPP Authentication Using HTTP Negotiate Authentication
Client System
Print Service System
User
IPP Client
HTTP User Agent
HTTP Service
IPP Printer
Authorization Service
1 Do something that triggers Client need to interact with Printer
2 Formulate IPP operation request payload (application/ipp)
3 Perform HTTP POST of request payload
4 POST /ipp/print HTTP/1.1 Content-Type: application/ipp Expect: 100-continue
Start HTTP Negotiate Authentication =
5 HTTP/1.1 401 Unauthorized WWW-Authenticate: Negotiate realm="testrealm@host.com", qop="auth,auth-int", nonce="dcd98b7102dd2f0e8b11d0f600bfb0c093", opaque="5ccc069c403ebaf9f0171e9517f40e41"
Leveraged description from https://en.wikipedia.org/wiki/Digest_access_authentication
6 Request authentication
7 Request authentication
8 Provides credentials
9 Retry with provided credentials
10 POST /ipp/print HTTP/1.1 Content-Type: application/ipp Expect: 100-continue Authorization: Negotiate username="Mufasa", realm="testrealm@host.com", nonce="dcd98b7102dd2f0e8b11d0f600bfb0c093", uri="/ipp/print", qop=auth, nc=00000001, cnonce="0a4f113b", response="6629fae49393a05397450978507c4ef1", opaque="5ccc069c403ebaf9f0171e9517f40e41"
Leveraged description from https://en.wikipedia.org/wiki/Digest_access_authentication
11 Check access with local auth database
12 Check access with external auth database
13 Approve Access
End HTTP Digest Authentication =
14 HTTP/1.1 100 Continue
15 << Send the application/ipp payload >>
16 Deliver IPP operation request
17 Formulate IPP operation response
18 Return IPP operation response
19 HTTP/1.1 200 OK Content-Type: application/ipp
20 Deliver the IPP operation response
21 Process the operation response
22 Present something from the operation response(s)
23 Done
User
IPP Client
HTTP User Agent
HTTP Service
IPP Printer
Authorization Service

Figure 3.5 : Sequence diagram for the 'negotiate' IPP Authentication Method

3.1.6 The 'oauth' IPP Authentication Method

The 'oauth' IPP Authentication method uses the HTTP “oauth” authentication scheme [RFC5849].

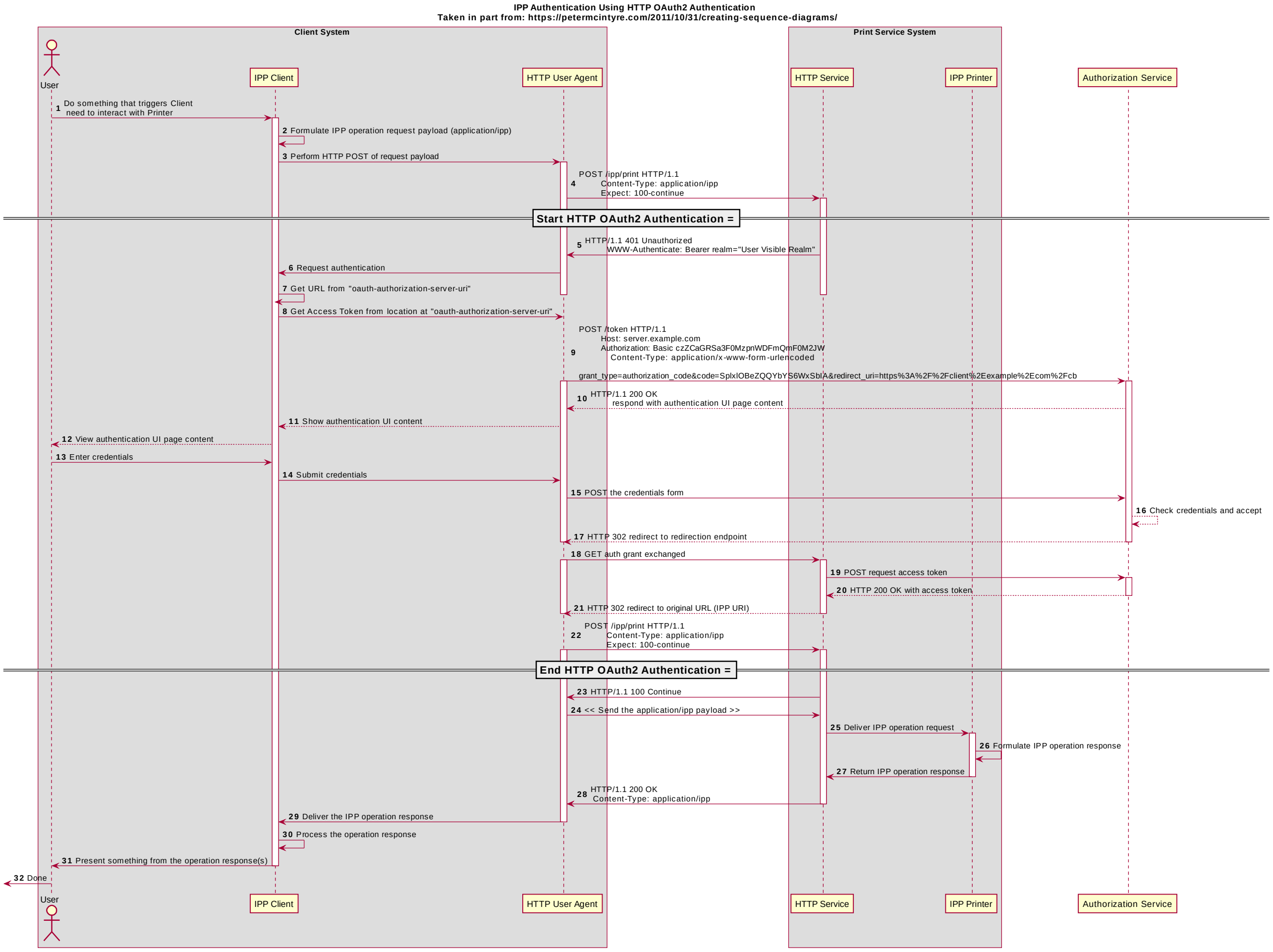

IPP Authentication Using HTTP OAuth2 Authentication
Taken in part from: https://petermcintyre.com/2011/10/31/creating-sequence-diagrams/
Client System
Print Service System
User
IPP Client
HTTP User Agent
HTTP Service
IPP Printer
Authorization Service
1 Do something that triggers Client need to interact with Printer
2 Formulate IPP operation request payload (application/ipp)
3 Perform HTTP POST of request payload
4 POST /ipp/print HTTP/1.1 Content-Type: application/ipp Expect: 100-continue
Start HTTP OAuth2 Authentication =
5 HTTP/1.1 401 Unauthorized WWW-Authenticate: Bearer realm="User Visible Realm"
6 Request authentication
7 Get URL from "oauth-authorization-server-uri"
8 Get Access Token from location at "oauth-authorization-server-uri"
9 POST /token HTTP/1.1 Host: server.example.com Authorization: Basic czZCaGRSa3F0MzpnWDFmQmF0M2JW Content-Type: application/x-www-form-urlencoded grant_type=authorization_code&code=SplxlOBeZQQYbYS6WxSbIA&redirect_uri=https%3A%2F%2Fclient%2Eexample%2Ecom%2Fcb
10 HTTP/1.1 200 OK respond with authentication UI page content
11 Show authentication UI content
12 View authentication UI page content
13 Enter credentials
14 Submit credentials
15 POST the credentials form
16 Check credentials and accept
17 HTTP 302 redirect to redirection endpoint
18 GET auth grant exchanged
19 POST request access token
20 HTTP 200 OK with access token
21 HTTP 302 redirect to original URL (IPP URI)
22 POST /ipp/print HTTP/1.1 Content-Type: application/ipp Expect: 100-continue
End HTTP OAuth2 Authentication =
23 HTTP/1.1 100 Continue
24 << Send the application/ipp payload >>
25 Deliver IPP operation request
26 Formulate IPP operation response
27 Return IPP operation response
28 HTTP/1.1 200 OK Content-Type: application/ipp
29 Deliver the IPP operation response
30 Process the operation response
31 Present something from the operation response(s)
32 Done
User
IPP Client
HTTP User Agent
HTTP Service
IPP Printer
Authorization Service

Figure 3.6 : Sequence diagram for the 'oauth' IPP Authentication Method

4 Implementation Recommendations

TBD?

5 Internationalization Considerations

For interoperability and basic support for multiple languages, conforming implementations MUST support the Universal Character Set (UCS) Transformation Format -- 8 bit (UTF-8) [RFC3629] encoding of Unicode [UNICODE] [ISO10646] and the Unicode Format for Network Interchange [RFC5198].

Implementations of this specification SHOULD conform to the following standards on processing of human-readable Unicode text strings, see:

- Unicode Bidirectional Algorithm [UAX9] – left-to-right, right-to-left, and vertical
- Unicode Line Breaking Algorithm [UAX14] – character classes and wrapping
- Unicode Normalization Forms [UAX15] – especially NFC for [RFC5198]
- Unicode Text Segmentation [UAX29] – grapheme clusters, words, sentences
- Unicode Identifier and Pattern Syntax [UAX31] – identifier use and normalization
- Unicode Collation Algorithm [UTS10] – sorting
- Unicode Locale Data Markup Language [UTS35] – locale databases

Implementations of this specification are advised to also review the following informational documents on processing of human-readable Unicode text strings:

- Unicode Character Encoding Model [UTR17] – multi-layer character model
- Unicode in XML and other Markup Languages [UTR20] – XML usage
- Unicode Character Property Model [UTR23] – character properties
- Unicode Conformance Model [UTR33] – Unicode conformance basis

6 Security Considerations

Provide security considerations for this document.

6.1 Human-readable Strings

Implementations of this specification SHOULD conform to the following standard on processing of human-readable Unicode text strings, see:

- Unicode Security Mechanisms [UTS39] – detecting and avoiding security attacks

Implementations of this specification are advised to also review the following informational document on processing of human-readable Unicode text strings:

- Unicode Security FAQ [UNISECFAQ] – common Unicode security issues

7 References

7.1 Normative References

[IANA-HTTP-AUTH] Hypertext Transfer Protocol (HTTP) Authentication Scheme Registry, Internet Assigned Numbers Authority, https://www.iana.org/assignments/http-authschemes/http-authschemes.xml

[ISO10646] "Information technology -- Universal Coded Character Set (UCS)", ISO/IEC 10646:2011

[PWG5100.12] R. Bergman, H. Lewis, I. McDonald, M. Sweet, "IPP Version 2.0, 2.1, and 2.2", PWG 5100.12-2015, October 2015, http://ftp.pwg.org/pub/pwg/standards/std-ipp20-20151030-5100.12.pdf

[PWG5100.13] M. Sweet, I. McDonald, P. Zehler, "IPP: Job and Printer Extensions - Set 3 (JPS3)", PWG 5100.13-2012, July 2012, http://ftp.pwg.org/pub/pwg/candidates/cs-ippjobprinterext3v10-20120727-5100.13.pdf

[PWG5100.14] M. Sweet, I. McDonald, A. Mitchell, J. Hutchings, "IPP Everywhere", 5100.14-2013, January 2013, http://ftp.pwg.org/pub/pwg/candidates/cs-ippeve10-20130128-5100.14.pdf

[PWG5100.19] S. Kennedy, "IPP Implementor's Guide v2.0", PWG 5100.19-2015, August 2015, http://ftp.pwg.org/pub/pwg/candidates/cs-ippig20-20150821-5100.19.pdf

[PWG5100.SYSTEM] I. McDonald, “IPP System Service v1.0”, PWG 5100.SYSTEM, TBD, http://ftp.pwg.org/pub/pwg/ipp/wd/wd-ippsystem10-20170719.pdf

[RFC2817] R. Khare, S. Lawrence, “Upgrading to TLS Within HTTP/1.1”, RFC 2817, May 2000, https://www.ietf.org/rfc/rfc2817.txt

[RFC3629] F. Yergeau, “UTF-8, a transformation format of ISO 10646”, RFC 3629, November 2003, https://www.ietf.org/rfc/rfc3629.txt

[RFC5198] J. Klensin, M. Padlipsky, “Unicode Format for Network Interchange”, RFC 5198, March 2008, https://www.ietf.org/rfc/rfc5198.txt

[RFC7230] R. Fielding, J. Reschke, "Hypertext Transfer Protocol (HTTP/1.1): Message Syntax and Routing", RFC 7230, June 2014, https://www.ietf.org/rfc/rfc7230.txt

[RFC7616] R. Shekh-Yusef, D. Ahrens, S. Bremer, “HTTP Digest Access Authentication”, RFC 7616, September 2015, https://www.ietf.org/rfc/rfc7616.txt

[RFC7617] J. Reschke, “The 'Basic' HTTP Authentication Scheme”, RFC 7617, September 2015, https://www.ietf.org/rfc/rfc7617.txt

[RFC8010] M. Sweet, I. McDonald, “Internet Printing Protocol/1.1: Encoding and Transport”, RFC 8010, January 2017, https://www.ietf.org/rfc/rfc8010.txt

[RFC8011] M. Sweet, I. McDonald, “Internet Printing Protocol/1.1: Model and Semantics”, RFC 8011, January 2017, https://www.ietf.org/rfc/rfc8011.txt

[UAX9] Unicode Consortium, “Unicode Bidirectional Algorithm”, UAX#9, May 2016, http://www.unicode.org/reports/tr9

[UAX14] Unicode Consortium, “Unicode Line Breaking Algorithm”, UAX#14, June 2016, http://www.unicode.org/reports/tr14

[UAX15] Unicode Consortium, “Normalization Forms”, UAX#15, February 2016, http://www.unicode.org/reports/tr15

[UAX29] Unicode Consortium, “Unicode Text Segmentation”, UAX#29, June 2016, http://www.unicode.org/reports/tr29

[UAX31] Unicode Consortium, “Unicode Identifier and Pattern Syntax”, UAX#31, May 2016, http://www.unicode.org/reports/tr31

[UNICODE] The Unicode Consortium, “Unicode® 10.0.0”, June 2017, http://unicode.org/versions/Unicode10.0.0/

[UTS10] Unicode Consortium, “Unicode Collation Algorithm”, UTS#10, May 2016, http://www.unicode.org/reports/tr10

[UTS35] Unicode Consortium, “Unicode Locale Data Markup Language”, UTS#35, October 2016, http://www.unicode.org/reports/tr35

[UTS39] Unicode Consortium, “Unicode Security Mechanisms”, UTS#39, June 2016, http://www.unicode.org/reports/tr39

7.2 Informative References

[UNISECFAQ] Unicode Consortium “Unicode Security FAQ”, November2016, http://www.unicode.org/faq/security.html

[UTR17] Unicode Consortium “Unicode Character Encoding Model”, UTR#17, November 2008, http://www.unicode.org/reports/tr17

[UTR20] Unicode Consortium “Unicode in XML and other Markup Languages”, UTR#20, January 2013, http://www.unicode.org/reports/tr20

[UTR23] Unicode Consortium “Unicode Character Property Model”, UTR#23, May 2015, http://www.unicode.org/reports/tr23

[UTR33] Unicode Consortium “Unicode Conformance Model”, UTR#33, November 2008, http://www.unicode.org/reports/tr33

8 Authors' Addresses

Primary authors (using Address style):

Smith Kennedy
11311 Chinden Blvd.
Boise ID 83714
smith.kennedy@hp.com

The authors would also like to thank the following individuals for their contributions to this whitepaper:

Mike Sweet – Apple Inc.
Zapp Brannigan - Democratic Order of Planets

9 Change History

9.1 August 3, 2017

Initial revision.