December 5, 2017
White Paper

The Printer Working Group

IPP Authentication Methods (IPPAUTH)

Status: InterimInitial

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-20171205.odt
~~http://ftp.pwg.org/pub/pwg/ipp/whitepaper/tb-ippauth-20170802.odt~~
http://ftp.pwg.org/pub/pwg/ipp/whitepaper/tb-ippauth-20171205.pdf
~~http://ftp.pwg.org/pub/pwg/ipp/whitepaper/tb-ippauth-20170802.pdf~~

Title: IPP Authentication Methods *(IPPAUTH)*

Table of Contents

List of Figures

~~List of Tables~~

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.

1 Terminology

1.1 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.

1.2 Other Terms Used in This Document

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

1.3 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/

2 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.

2.1 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.

2.1.1 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'.

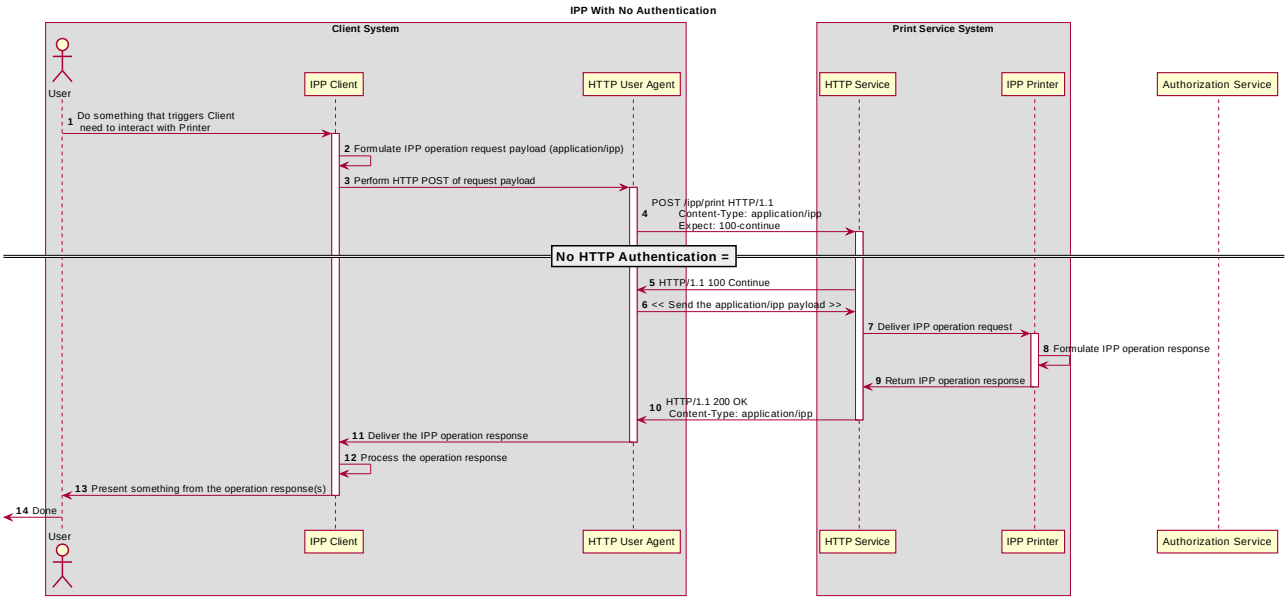


Figure 2.1: Sequence diagram for the 'none' IPP Authentication Method

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 2.1.2.

2.1.2 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.

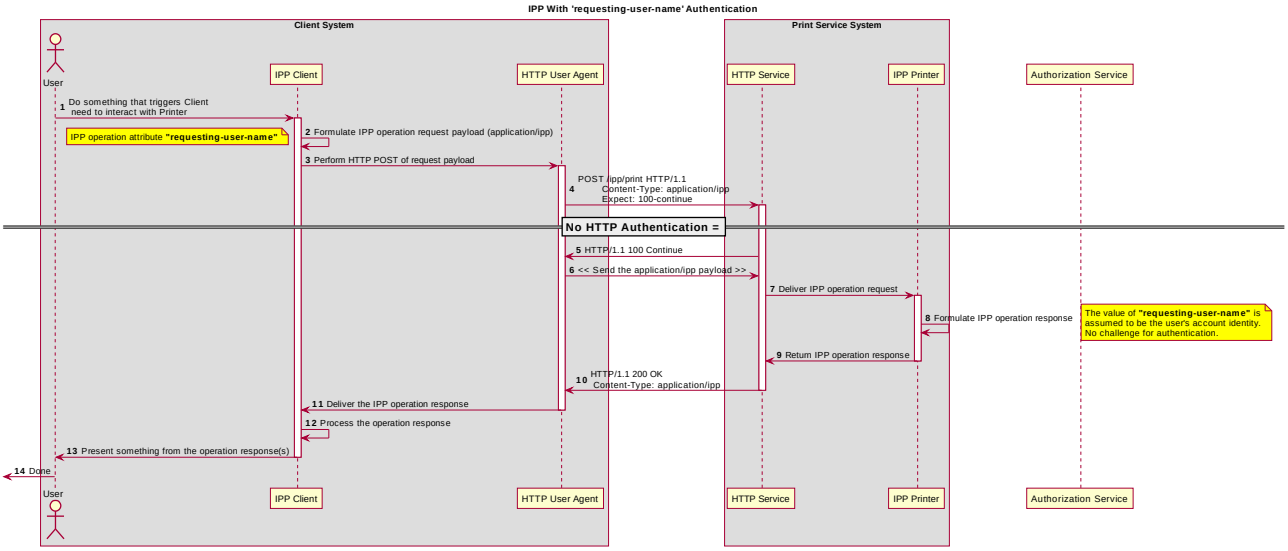


Figure 2.2: Sequence diagram for the 'requesting-user-name' IPP Authentication Method

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”.

2.1.3 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.

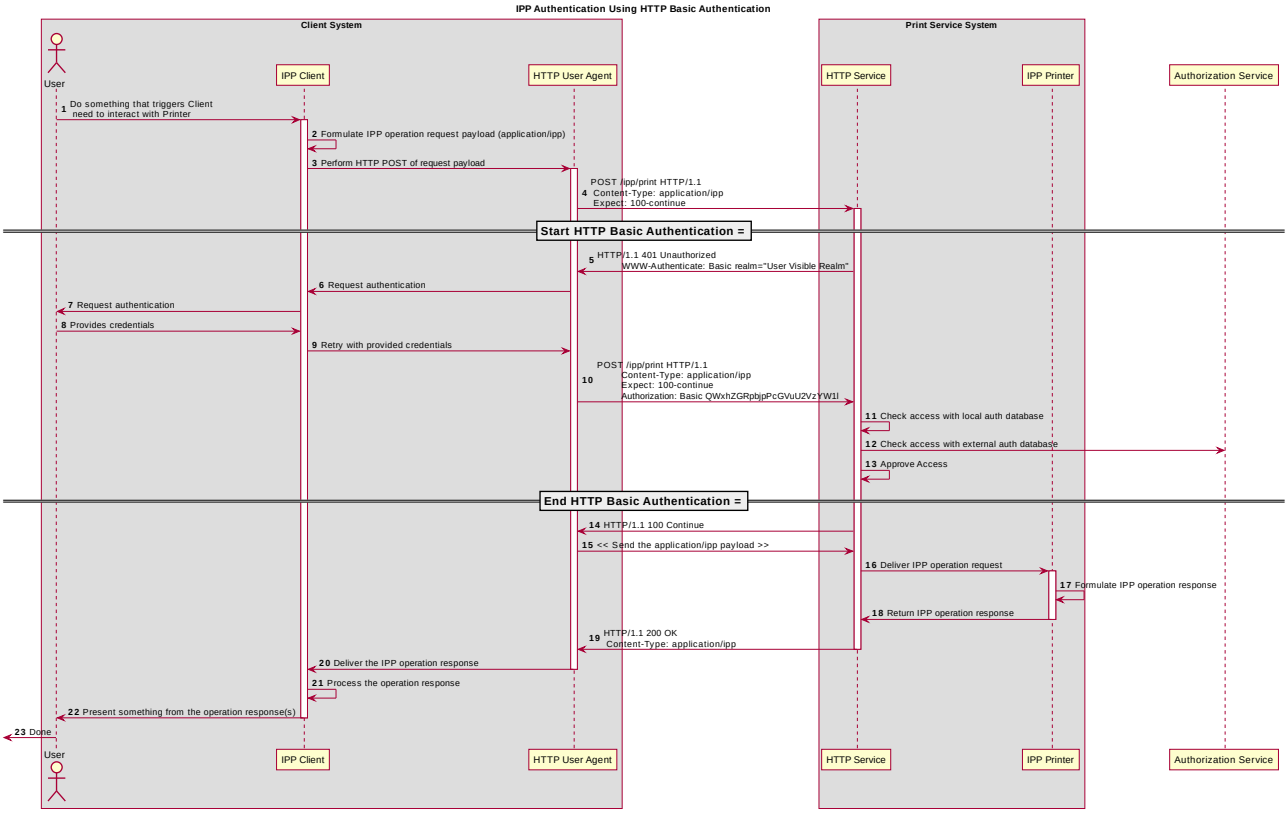


Figure 2.3 : Sequence diagram for the 'basic' IPP Authentication Method

2.1.4 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.

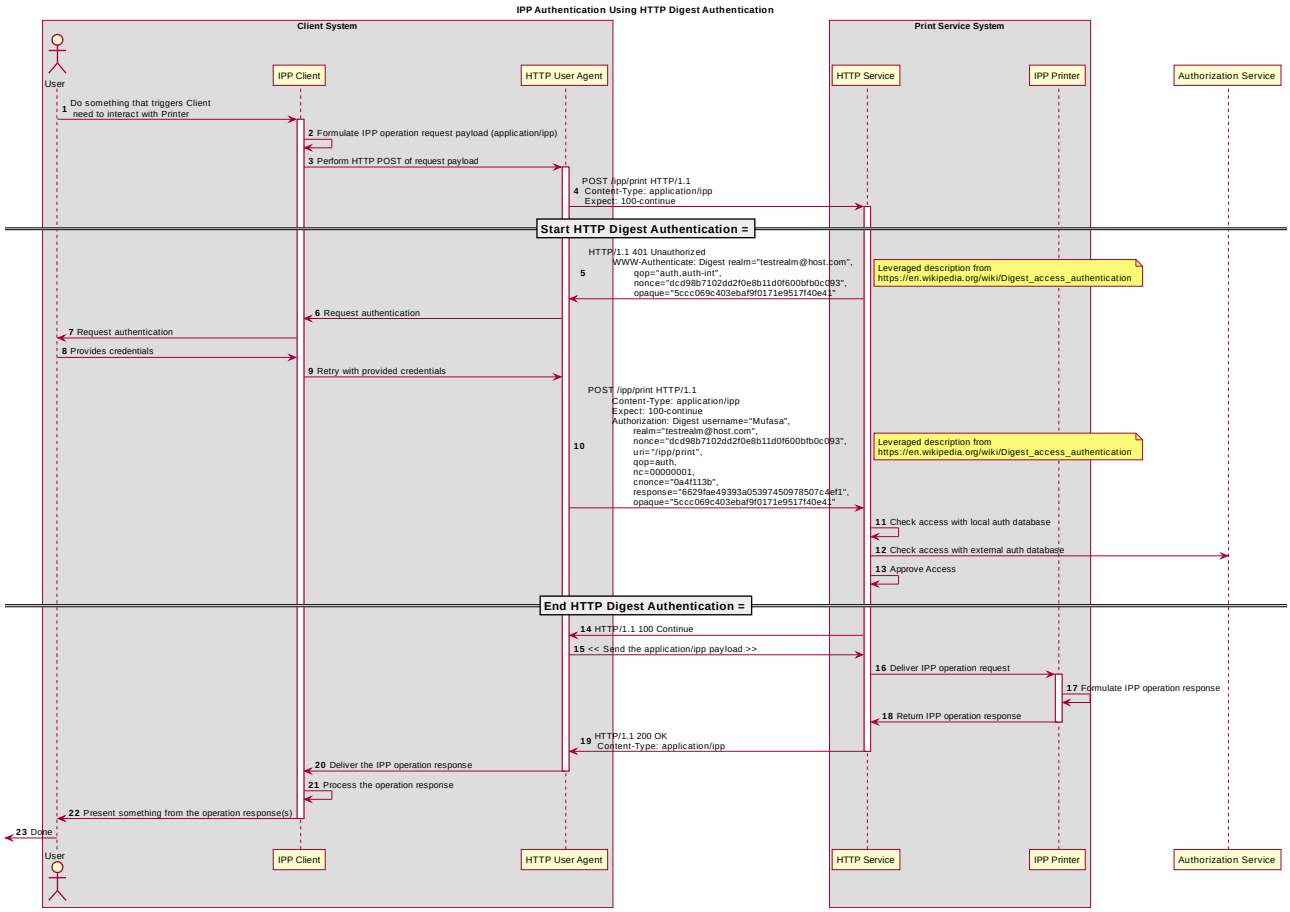


Figure 2.4 : Sequence diagram for the 'digest' IPP Authentication Method

2.1.5 The 'negotiate' IPP Authentication Method

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

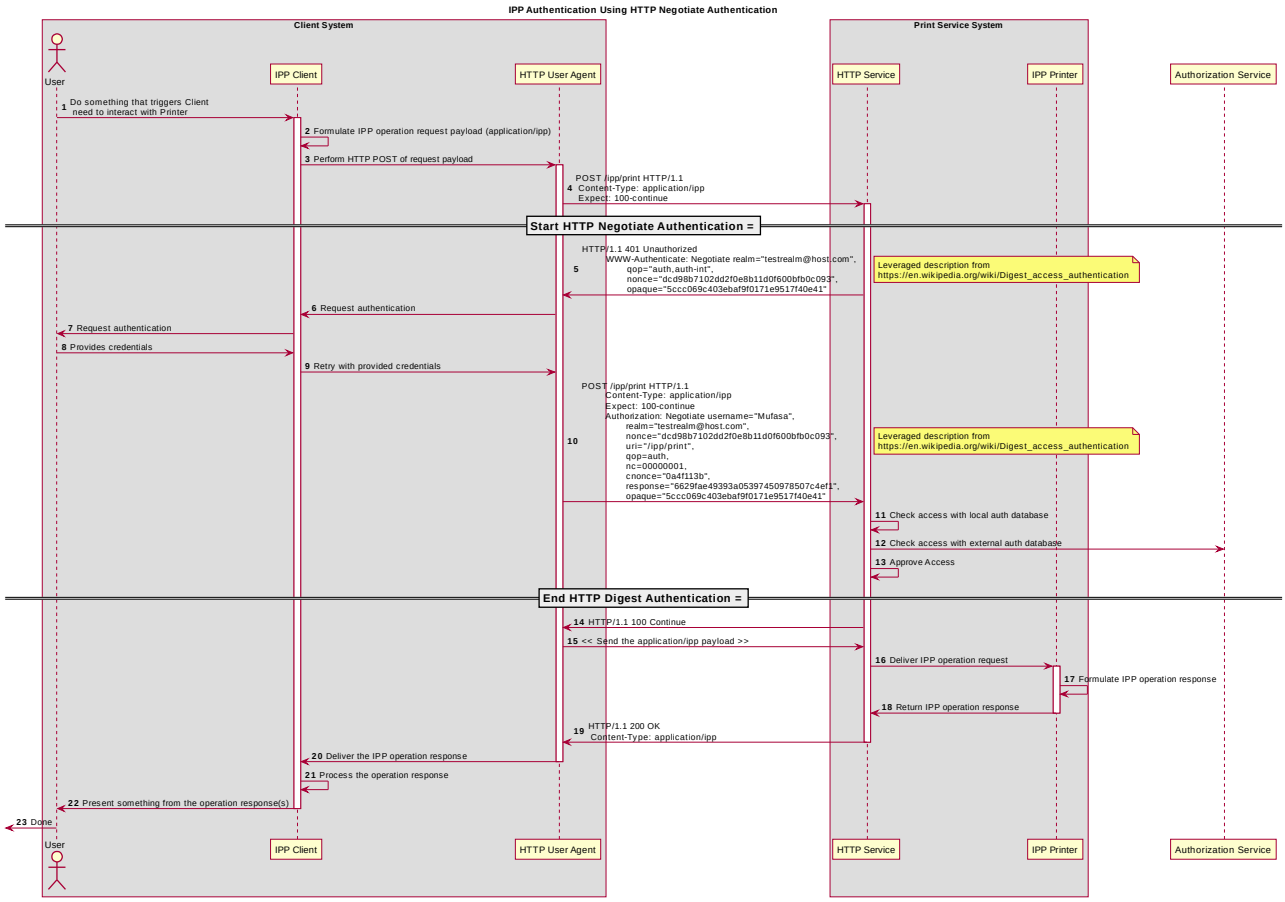


Figure 2.5 : Sequence diagram for the 'negotiate' IPP Authentication Method

2.1.6 The 'oauth' IPP Authentication Method

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

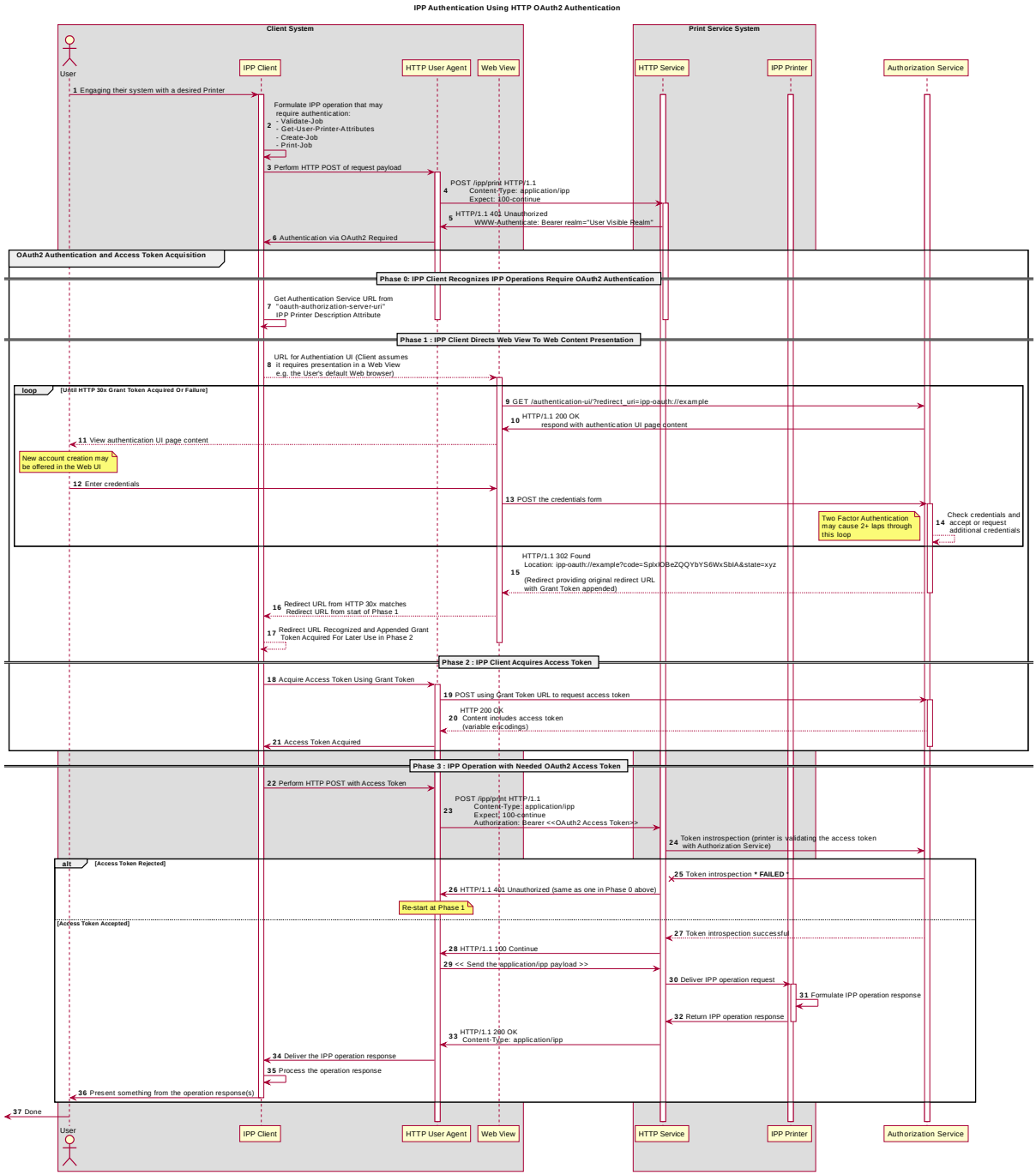
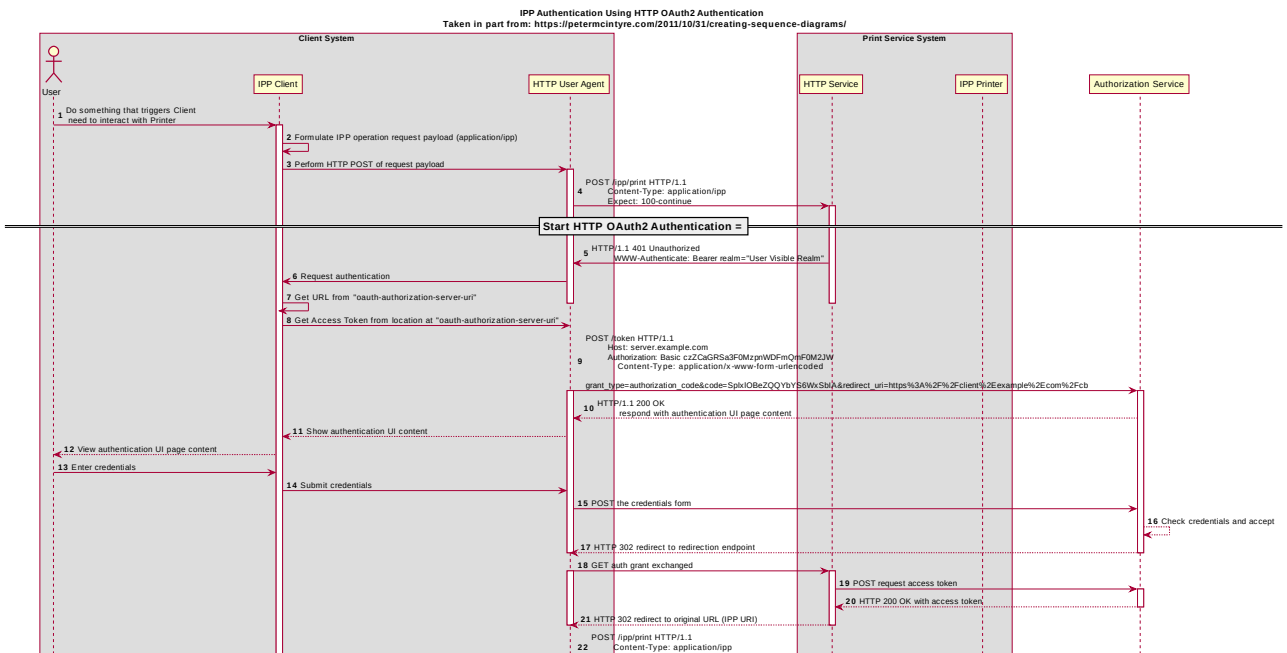


Figure 2.6 : Sequence diagram for the 'oauth' IPP Authentication Method



Implementation Recommendations

2.2 Client Implementation Recommendations

2.2.1 General Recommendations

A Client SHOULD as a general principle limit the number of additional windows presented to the user during the course of an authentication workflow, to avoid causing a fragmented, disruptive user experience.

2.2.2 OAuth2 Recommendations

A Client that supports OAuth2 authentication

- User experience considerations
- Information Disclosure
  - If the native app uses an embedded web view, then the native app might have access to the web view (directly or indirectly). That means the native app might have access to the controls and the information in that web view. That may or may not be desirable...
  - RFC 7636 (PKCE) and RFC 8252 (native apps OAuth2 recommendations) should be examined for further recommendations to be leveraged here and calling out specific sections of those that pertain to the use cases that are relevant to PWG / IPP (e.g. printer discovery UI, print dialog UI)

2.3 Printer Implementation Recommendations

TBD

~~TBD?~~

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

3 Security Considerations

Provide security considerations for this document.

3.1 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

3.2 Client Security Considerations

An IPP Client SHOULD follow the recommendations below

1. A Client SHOULD securely store at rest any personally identifiable information (PII) and authentication credentials such as passwords.
2. A Client SHOULD only respond to an authentication challenge over a secure connection (TLS) [RFC8010][RFC8011] unless TLS is not supported over that transport (e.g. IPP USB).
3. A Client SHOULD provide a means to allow the User to examine a Printer's provided identity.
4. A Client SHOULD provide one or more means of notification when it is engaging with a previously encountered Printer whose identity has changed.
5. Validating the Printer identity (am I talking to whom I think I'm talking to?) → look in 8010 / 8011 for guidance or references to guidance

3.3 Printer Security Considerations

An IPP Printer SHOULD follow the recommendations below.

1. A Printer SHOULD securely store at rest any personally identifiable information (PII) and authentication credentials such as passwords that are local to the Printer.
2. A Printer SHOULD only challenge a Client for authentication over a secure connection (TLS) [RFC8010][RFC8011] unless TLS is not supported over that transport (e.g. IPP USB).
3. Certificates
    1. What is an acceptable certificate?
    2. How long is a self-signed certificate expected to last?
    3. How long should a CA issued certificate last? (e.g. recent work on short lives CA certificates...)
    4. Let's Encrypt and IPP (and OAuth2 or in general?)
4. Point to best practice documents

4 References

4.1 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

4.2 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

5 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

6 Change History

6.1 December 5, 2017

Updated as per feedback from the November 2017 PWG vF2F and subsequent work with IPP WG members on specific details

- Corrected OAuth2 sequence diagram to more correctly describe the sequence of operations and actors involved in an OAuth2 authenticated IPP Printer scenario.
- Added Implementation Recommendations that were revealed during the course of correcting the OAuth2 sequence diagram.

6.2 August 3, 2017

Initial revision.

