Abstract: This Best Practice document provides implementation guidance on how to best integrate various authentication mechanisms used over IPP's HTTP and HTTPS transports into IPP protocol exchanges when printer access or print feature policy require authorization.

This is a PWG Best Practice document. For the definition of "PWG Best Practices", see:


This document is available electronically at:
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.
List of Tables

Table 4.1: IPP 'certificate' Authentication Method Error Condition Status Codes..................20
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, it challenges the Client for authentication credentials using one of the HTTP or TLS authentication methods. User experience problems can occur if the Printer or associated authentication and authorization infrastructure assumes that all User Agents are web browsers, since IPP Clients are HTTP User Agents but do not implement many content technologies used in contemporary web browsers, and their use of HTTP is constrained.

This document surveys the HTTP authentication methods employed today that support and are supported by IPP, and outlines limits, constraints and conventions that ought to be considered by Client developers, Printer developers, and Infrastructure Administrators when implementing support for one of these HTTP authentication methods in IPP communications, to ensure a high quality printing user experience.

2. Terminology

2.1. Conformance Terminology

Capitalized terms, such as MUST, MUST NOT, RECOMMENDED, REQUIRED, SHOULD, SHOULD NOT, MAY, and OPTIONAL, have special meaning relating to conformance as defined in Key words for use in RFCs to Indicate Requirement Levels [BCP14]. The term CONDITIONALLY REQUIRED is additionally defined for a conformance requirement that applies when a specified condition is true.

2.2. 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.

2.3. Other Terms Used in This Document

**Authentication**: The corroboration that a peer entity in an association is the one claimed. ([ITUX.800] definition for “peer entity authentication”)
Authorization: The granting of rights, which includes the granting of access based on access rights. ([ITUX.800])

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

2.4. Acronyms and Organizations

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

3. Requirements

3.1. Rationale

Given the following existing specifications:

1. Internet Printing Protocol/1.1: Encoding and Transport [RFC8010] and Internet Printing Protocol/1.1: Model and Semantics [RFC8011] define the core Internet Printing Protocol/1.1 IETF STD 92
2. RFC 7617 defines the 'Basic' HTTP Authentication Scheme
3. RFC 7616 defines HTTP Digest Access Authentication
4. RFC 4559 defines SPNEGO-based Kerberos and NTLM HTTP Authentication
5. RFC 6749 defines the OAuth 2.0 Authorization Framework
6. RFC 8252 describes best practices for OAuth 2.0 for Native Apps

And given the need for Clients and Printers to provide and support a positive user experience while supporting these HTTP authentication methods and in many cases not supporting the full functionality of a Web browser, this IPP Authentication Methods Best Practices document should:

• Describe each HTTP authentication system;
• Highlight details and consider pitfalls that can impact the IPP Client user experience
3.2. Use Cases

3.2.1. Authentication Required for Authorized Printer Access

Andy is at work and wants to print from his laptop. He finds and selects a printer on his network. The IPP Client in his laptop checks to see if using the Printer will require authentication, so that the User's expectations can be appropriately managed. The Printer responds with an authentication challenge, and the Client presents a user interface appropriate for the HTTP authentication type in the challenge. Andy provides his credential information to the Client, and the Client submits that to the Printer. The Printer authenticates Andy's credentials and confirms Andy's account is authorized to print without restrictions, and specifies the features he is authorized to use. The laptop provides the usual print dialog user interface, allowing Andy to select among those authorized print options.

3.3. Exceptions

3.3.1. Authentication Failure Prevents Access To Printer

Lisa is visiting Andy's office and wants to print from her tablet. She uses her tablet to discover available printers, and selects one listed. The printer is configured to limit access to only authorized users.

The printer challenges the tablet for authentication, and the tablet presents an authentication dialog to Lisa. Lisa doesn't have an account, but enters her email address and guesses at a password anyway. The printer rejects these credentials, and sends another challenge. Her tablet shows the authentication dialog again. Lisa clicks “Cancel” and looks for a different printer.

3.3.2. Authorization Policy Limits Access To Print Features

Harry is an intern who works at Andy's office, and he wants to print some photos from his laptop. He uses his laptop to discover available printers, and selects one listed. The printer is configured to allow on limit access to color printing to only authorized users to print in color, and interns are not authorized to use this feature. His laptop has a modern IPP Client that supports the IPP Get-User-Printer-Attributes operation, so features that he isn't allowed to use will not be listed in the print dialog.

When he selects the printer, the laptop sends the Get-User-Printer-Attributes IPP operation to request the list of authorized features available to Harry's account. The printer responds to the laptop with an authentication challenge. The laptop has stored single sign-on credentials, so it uses those to avoid bothering its user with a distraction. The printer accepts these credentials, and provides the list of features his account is authorized to use. The laptop shows this set of features. Harry is disappointed that he cannot print in color, so he abandons trying to print the photos because he doesn't want black-and-white prints.
3.4. Out of Scope

The following are considered out of scope for this document:

1. Definition of new HTTP authentication methods
2. Definition of how specific authorization mechanisms are used by an IPP Printer.

4. Client Authentication Methods

Authentication is the process of establishing some level of trust that an entity is who or what they are claiming to be. A Printer uses the “authenticated identity” or the “most authenticated user” [RFC8011] to determine whether to authorize the requesting Client to access requested capabilities such as operations, resources, and attributes. The Internet Printing Protocol/1.1 [RFC8011] defines authorization roles for end users, operators, and administrators, but does not define how a Printer or an authorization mechanism maps those roles to authenticated users.

A Printer specifies its supported authentication methods via several IPP attributes. The “uri-authentication-supported” attribute [RFC8011] indicates the authentication method used for a corresponding URI in “printer-uri-supported” [RFC8011]. The “xri-authentication” member attribute of “printer-xri-supported” [RFC3380] specifies the same corresponding values, if the Printer implements the “printer-xri-supported” attribute. Each of the authentication method keywords currently registered for “uri-authentication-supported” is described in its own subsection below. Some authentication methods may have additional IPP attributes associated with them.

One authentication & authorization system system not described in this document is SAML (Security Assertion Markup Language) [SAMLCORE]. As of this writing, none of the standard SAML bindings to HTTP directly support IPP. OAuth 2.0 can indirectly support SAML via a SAML / OAuth 2.0 gateway. The gateway typically uses the SAML 2.0 assertion as an OAuth 2.0 Bearer token. Specific instructions for how to configure this depends on the SAML and OAuth 2.0 system implementations, and as with other infrastructure topics is beyond the scope of this document.
4.1. The 'none' IPP Authentication Method

The 'none' IPP Authentication Method [RFC8011] indicates that the receiving Printer provides no method to accept an asserted identity for the User operating the Client. The user name for the operation is assumed to be 'anonymous'. This authentication method is not recommended unless the Printer's operator intends to provide an anonymous print service.

Figure 4.1 illustrates how the 'none' authentication method integrates into an IPP operation request / response exchange.
4.2. The 'requesting-user-name' IPP Authentication Method

The 'requesting-user-name' IPP Authentication Method [RFC8011] indicates that the Client will provide the "requesting-user-name" operation attribute [RFC8011] in its IPP operation request. The Printer uses this unauthenticated name as the identity of the User operating the Client. This method is not recommended if job accounting or access authorization is important, since the Printer does not challenge the Client to prove the identity claimed in the "requesting-user-name". Also, some Clients always send a fixed identity name (e.g. "mobile") as a privacy defense when sending requests. As such, from a Printer's perspective, this method is increasingly undependable.

Figure 4.2 illustrates how the 'requesting-user-name' authentication method integrates into an IPP operation request / response exchange.
4.3. The 'basic' IPP Authentication Method

The 'basic' IPP Authentication Method uses the HTTP Basic authentication scheme [RFC7617]. It is employed in IPP in much the same way as in conventional HTTP workflows using a Web browser. When the IPP Client receives an HTTP 401 Unauthorized response status and the “WWW-Authenticated” header in that response specifies 'Basic', a supporting Client will present UI asking the User to provide a user name and password. The Client will re-submit the IPP operation request to the HTTP Server providing access to the IPP Printer, including the “Authorization” HTTP header field with the provided credentials encoded in the format defined for the 'Basic' authentication method [RFC7617]. If the HTTP Server accepts that set of credentials, the IPP Printer authorizes access to the requested IPP operation and attributes for that account, and will respond accordingly.

Figure 4.3 illustrates how the 'basic' authentication method integrates into an IPP operation request / response exchange.
Best Practice – IPP Authentication Methods (IPPAUTH)  March 28, 2019

Figure 4.3: Sequence diagram for the 'basic' IPP Authentication Method
4.4. 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 as in conventional HTTP workflows using a Web browser. When the IPP Client receives an HTTP 401 Unauthorized response status and the “WWW-Authenticated” header in that response specifies 'Digest', a supporting Client will present UI asking the User to provide a user name and password. The Client will re-submit the IPP operation request to the HTTP Server providing access to the IPP Printer, including the “Authorization” HTTP header field with the provided credentials encoded in the format defined for the 'Digest' authentication method [RFC7616]. If the HTTP Server accepts that set of credentials, the IPP Printer authorizes access to the requested IPP operation and attributes for that account, and will respond accordingly.

Figure 4.4 illustrates how the 'digest' authentication method integrates into an IPP operation request / response exchange.
**Best Practice – IPP Authentication Methods (IPPAUTH)**  
March 28, 2019

---

**Figure 4.4: Sequence diagram for the ‘digest’ IPP Authentication Method**

---

Do something that triggers client need to interact with Printer

1. Formulate IPP operation request

2. HTTP POST with request payload

3. POST /ipp/print HTTP/1.1
   Content-Type: application/ipp
   Expect: 100-continue

4. HTTP/1.1 100 Continue

5. << Send the application/ipp payload >>

6. Deliver IPP operation request

7. Request policy approval for access to specified IPP operations & attributes

8. Authorization FAILED

9. Authentication required

10. Start HTTP Digest Authentication

11. HTTP/1.1 401 Unauthorized
   WWW-Authenticate: Digest realm="testrealm@host.com",
   qop="auth,auth-int",
   nonce="dcd98b7102dd2f0e8b11d0f600bfb0c093",
   opaque="5ccc069c403ebaf9f0171e9517f40e41"

12. Request authentication

13. Request authentication

14. Provides credentials

15. Retry with provided credentials

16. POST /ipp/print HTTP/1.1
   Content-Type: application/ipp
   Authorization: Digest username="Mufasa",
   realm="testrealm@host.com",
   nonce="dcd98b7102dd2f0e8b11d0f600bfb0c093",
   uri= "/ipp/print",
   qop= "auth",
   nc=00000001,
   cnonce="0a4f113b",
   response="6629fae49393a05397450978507c4ef1",
   opaque="5ccc069c403ebaf9f0171e9517f40e41"

17. HTTP/1.1 100 Continue

18. << Send the application/ipp payload >>

19. Deliver IPP operation request & identity credentials

20. Request policy approval for access to specified IPP operations & attributes with given credentials

21. Authorization Success

22. End HTTP Digest Authentication

23. Formulate IPP Response

24. HTTP/1.1 200 OK
   Content-Type: application/ipp

25. Return IPP operation response

26. Process the operation response

27. Present something from the operation response(s)

---
4.5. The 'negotiate' IPP Authentication Method

The 'negotiate' IPP Authentication method uses the HTTP Negotiate authentication scheme [RFC4559], which is used to support Kerberos and NTLM authentication methods with HTTP. It is employed in IPP in much the same way as in conventional HTTP workflows using a Web browser. When the IPP Client receives an HTTP 401 Unauthorized response status and the “WWW-Authenticated” header in that response specifies ‘Negotiate’, a supporting Client will present UI asking the User to provide a user name and password. The Client will re-submit the IPP operation request to the HTTP Server providing access to the IPP Printer, including the “Authorization” HTTP header field with the provided credentials encoded in the format defined for the 'Negotiate' authentication method [RFC4559]. If the HTTP Server accepts that set of credentials, the IPP Printer authorizes access to the requested IPP operation and attributes for that account, and will respond accordingly.

Figure 4.5 illustrates how the 'negotiate' authentication method integrates into an IPP operation request / response exchange.
Figure 4.5: Sequence diagram for the 'negotiate' IPP Authentication Method
4.6. The 'oauth' IPP Authentication Method

The 'oauth' IPP Authentication method pertains to OAuth 2.0, which uses:

- the OAuth 2.0 authentication scheme [RFC6749], which defines the OAuth 2.0 system, authentication protocol framework, and OAuth 2.0 access tokens, which represents the scope, duration, and other attributes of an authorization grant;

- The OAuth 2.0 Bearer Token [RFC6750] which specifies the ways that an OAuth 2.0 access token can be encoded into general purpose HTTP requests and responses as an HTTP Bearer Token;

- The OAuth 2.0 Authentication Server Metadata [RFC8414] which provides the necessary metadata for interoperability.

When the IPP Client receives an HTTP 401 Unauthorized response status, and the "WWW-Authenticated" header in that response specifies 'Bearer', a supporting Client will initiate the OAuth 2.0 flow by presenting a web view UI directed at the URL specified by the Printer's "oauth-authorization-server-uri" Printer Description attribute [PWG5100.18]. Once the Client has acquired an OAuth 2.0 Access Token, it will encode that in the Bearer Token format and re-submit the IPP operation to the IPP Printer, including the "Authorization" HTTP header field with the provided credentials encoded in the OAuth 2.0 Bearer Token format [RFC6750]. If the HTTP Server accepts that set of credentials, the IPP Printer authorizes access to the requested IPP operation and attributes for that account, and will respond accordingly.

OAuth 2.0 is an authorization service framework that uses one or more authentication services, such as SAML 2.0 [SAMLCORE]. Figure 4.6 illustrates how the 'oauth' authentication method integrates into an IPP operation request / response exchange.
Best Practice – IPP Authentication Methods (IPPAUTH)  
March 28, 2019

Figure 4.6: Sequence diagram for the 'oauth' IPP Authentication Method
4.7. The 'certificate' IPP Authentication Method

The 'certificate' IPP Authentication method uses X.509 certificate authentication via TLS [RFC5246]. This authentication method is initiated by the Printer when it sends a TLS Certificate Request message during the Transport Layer Security (TLS) handshake. The Client responds by sending a TLS Certificate message with the X.509 certificate identifying the User and/or Client. The Client then sends a TLS Certificate Verify message to prove to the Printer that the Client has the corresponding private key. If the Client has no X.509 certificate to provide to the Printer, it sends an empty TLS Certificate message.

The Printer SHOULD allow both empty and valid X.509 certificates. The Printer SHOULD return the IPP status code listed in Table 4.1 when the corresponding authentication exception occurs. The Client SHOULD respond to the reported status code with the corresponding response listed in Table 4.1.

<table>
<thead>
<tr>
<th>Operation Status Code</th>
<th>Authentication Exception</th>
<th>Recommended Client Response</th>
</tr>
</thead>
<tbody>
<tr>
<td>'client-error-not-authenticated'</td>
<td>Authentication required but no X.509 certificate supplied</td>
<td>Close the connection; select a certificate (with possible user interaction); retry connection with selected certificate</td>
</tr>
<tr>
<td>'client-error-not-authorized'</td>
<td>Access denied for the identity specified by the provided X.509 certificate; try again</td>
<td>Close the connection; select a different certificate (with possible user interaction); retry connection with selected certificate</td>
</tr>
<tr>
<td>'client-error-forbidden'</td>
<td>Access denied for the identity specified by the provided X.509 certificate; don't try again</td>
<td>Close the connection and present User with error dialog (&quot;Access denied&quot;)</td>
</tr>
</tbody>
</table>

Table 4.1: IPP 'certificate' Authentication Method Error Condition Status Codes

Figure 4.7 illustrates how the TLS authentication method integrates into an IPP operation request / response exchange.
Figure 4.7: Sequence diagram for the "certificate" IPP Authentication Method
5. Implementation Recommendations

5.1. Provide possible technical solutions/approaches in this section. Include pros and cons for each technical solution or approach. Include references to specific protocols and/or data models when appropriate. Include mapping and gateway considerations when appropriate.

5.2. Client Implementation Recommendations

5.2.1. General Recommendations

A Client SHOULD 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.

Since some tasks require multiple IPP operations, a Client SHOULD store non-persistent authentication credentials for reuse in later IPP operations for the duration of that task.

Client security considerations (section 7.2) should also be followed.

5.2.2. Handling Authentication Failure

A Client that encounters an authentication failure SHOULD offer the User another opportunity to provide valid authentication credentials and SHOULD abandon new attempts when the User rejects the offer for different credentials (e.g. by clicking on a “Cancel” button in an authentication dialog window). For HTTP authentication, the Client will receive an HTTP 401 Unauthorized response. For TLS authentication, the Client will receive an HTTP 200 OK with an IPP message body with status code 'client-error-not-authorized' [RFC8011].

5.2.3. Handling Authorization Failure

A Client that encounters an authorization failure SHOULD abandon communications with the target Printer because, while the credentials are recognized and authenticated, the identity corresponding to those valid credentials is not authorized to proceed. For HTTP authentication, the Client will receive an HTTP 403 Forbidden response. For TLS authentication, the Client will receive an HTTP 200 OK with an IPP message body with status code 'client-error-forbidden' [RFC8011].
5.2.4. OAuth 2.0 Recommendations

The Client that supports Resource Owner Grants (username and password) SHOULD otherwise follow the guidelines laid out in current OAuth 2.0 best practices including “Proof Key for Code Exchange by OAuth Public Clients” [RFC7636], “OAuth 2.0 for Native Apps” [RFC7636] and “OAuth 2.0 Security Best Current Practice” [OAUTH2SECBP].

5.3. Printer Implementation Recommendations

5.3.1. General Recommendations

The Printer or the Job might also need to store a token or identifier (UUID, JWT, etc.) that represents the User's authenticated identity or authentication session, in cases where the Printer depends on an external authorization service for print policy evaluation. This token is considered by IPP to be an internal implementation detail, and the Printer never MUST provide Clients access to these tokens via IPP, as discussed in [RFC8011] section 5.3.6.

When handing an IPP Job Creation request, the Printer will also need to populate the Job's "job-originating-user-name" Job Status attribute. In cases where the Printer relies upon an external authentication service, it will need to acquire a meaningfully printable value from the authentication service.

Client security considerations (section 4) should also be followed.

5.3.2. Handling Authentication Failure

If a Printer receives an IPP operation request, challenges the Client for authentication using one of the methods described in this document, and the credentials are invalid, how the Printer reports the authentication failure depends on the authentication method. For HTTP authentication, the Printer returns an HTTP 401 Unauthorized response. For TLS authentication, the Printer returns an HTTP 200 OK with an IPP message body specifying a 'client-error-not-authorized' status code [RFC8011].

5.3.3. Handling Authorization Failure

If a Printer receives an IPP operation request, and the Client credentials have been authenticated, but the identity corresponding to the credentials is not authorized to use the Printer or the operations or attributes specified in the request, how the Printer reports the authorization failure depends on the authentication method. For HTTP authentication, the Printer returns an HTTP 403 Forbidden response. For TLS authentication, the Printer returns an HTTP 200 OK with an IPP message body specifying a 'client-error-forbidden' status code [RFC8011].
5.3.4. HTTP Digest Recommendations

A Printer SHOULD NOT invalidate any HTTP Digest parameters (nonce, etc.) in the middle of an IPP operation request. Especially in the case of operations that are streaming document data (Print-Job, Send-Document), the data stream might not be cacheable by the Client, and this can cause a significant burden to the Client, degrade the user experience, or cause the operation to fail. Once a Printer has received a Job Creation operation request or a Validate-Job operation request, it SHOULD NOT change the nonce used for HTTP Digest authentication until the Job Submission operations for that Job have concluded.

5.3.5. OAuth 2.0 Recommendations

A Printer deployed in an OAuth 2.0 environment SHOULD follow current OAuth 2.0 best practices including “Proof Key for Code Exchange by OAuth Public Clients” [RFC7636], “OAuth 2.0 for Native Apps” [RFC7636] and “OAuth 2.0 Security Best Current Practice” [OAUTH2SECBP].

6. 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
7. Security Considerations

7.1. Human-readable Strings

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


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

7.2. Client Security Considerations

The following are the security recommendations for an IPP Client.

1. A Client SHOULD use the most secure authentication method supported by the Printer.

2. A Client SHOULD securely store at rest any personally identifiable information (PII) and authentication credentials such as passwords or session tokens.

3. A Client SHOULD only respond to an authentication challenge if it is conveyed over a secure connection (TLS) [RFC8010][RFC8011] unless a secure connectionTLS is not supported over that transport (e.g. IPP USB [IPPUSB]).

4. A Client SHOULD validate the identity of the Printer by whatever means are available for that connection type. If the connection is secured via TLS [RFC8010], the Client SHOULD validate the server's TLS certificate, match it to the originating host, cross-check it to match the host name or IP address in the IPP URI for the target Printer, and otherwise follow industry best practices for validating the Printer's identity using X.509 certificates over TLS [RFC6125]. If the connection is not secured via TLS, other means could be necessary to validate the Printer's identity.

5. A Client SHOULD provide a means to allow the User to examine a Printer's provided identity.
6. A Client SHOULD provide one or more means of notification when it is engaging with a previously encountered Printer whose identity has changed.

7. A Client supporting OAuth 2.0 SHOULD conform to the recommendations in “Proof Key for Code Exchange by OAuth Public Clients” [RFC7636] and “OAuth 2 for Native Apps” [RFC8252] if the print system provides its own user interface presentation and controls for handling the OAuth 2.0 authentication steps, to mitigate the risks described therein.

8. A Client SHOULD use the most secure authentication method available for a given Printer. In some cases, a Printer could support more than one authentication method for a particular URI. It can specify this by listing the same URI multiple times in its “printer-uri-supported” attribute, and specifying the different authentication methods in each of the corresponding values specified by its “uri-authentication-supported” attribute.

3. In most cases, the Printer SHOULD support and the Client SHOULD provide the “requesting-user-name” operation attribute, as described in section 4.2, if no more sophisticated method is supported for asserting a User's identity.

4. Printer Security Considerations

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. A Printer SHOULD A Printer MUST support Administrator/User-provisioned X.509 server certificates that persist across power cycles, and these certificates SHOULDs. These certificates MUST NOT be automatically renewed or replaced.

4. A Printer SHOULD support self-generated self-signed X.509 certificates that persist across power cycles. The certificate SHOULD have a minimum default expiration of 5 years from the date of issuance / generation, SHOULD be automatically renewed (regenerated), using a new private key if the previous certificate has expired, SHOULD be generated using the mDNS, DHCP and/or manually-configured DNS hostname(s) and regenerated whenever these change, and SHOULD comply with the recommendations from the CA/Browser Forum [CABCORE] relating to, among other things, the set of cryptographic primitives, algorithms and key lengths to use to produce the certificate.
5. In cases where the Printer supports more than one authentication method for a particular URI, the Printer SHOULD specify the alternative authentication schemes by listing the same URI multiple times in its “printer-uri-supported” attribute and specifying a different authentication method for each corresponding value in its “uri-authentication-supported” attribute.

6. A Printer supporting OAuth 2.0 SHOULD conform to the recommendations in “Proof Key for Code Exchange by OAuth Public Clients” [RFC7636] and “OAuth 2 for Native Apps” [RFC8252] to mitigate the risks described therein.

8. References

8.1. Normative References


[PWG5100.SYSTEM] I. McDonald, M. Sweet, “IPP System Service v1.0”, PWG

[RFC2817] R. Khare, S. Lawrence, “Upgrading to TLS Within HTTP/1.1”, RFC

(IPP): Job and Printer Set Operations”, RFC 3380, September 2002,

[RFC3629] F. Yergeau, “UTF-8, a transformation format of ISO 10646”, RFC

NTLM HTTP Authentication in Microsoft Windows”, RFC 4559, June


Token Usage”, RFC 6750, October 2012,

Message Syntax and Routing”, RFC 7230, June 2014,


[RFC7617] J. Reschke, “The 'Basic' HTTP Authentication Scheme”, RFC 7617,

Exchange by OAuth Public Clients", RFC 7636, September 2015,

8.3. Informative References


9. Authors' Addresses

Primary authors:

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

Michael Sweet
Apple Inc.
One Apple Park Way
MS 111-HOMC
Cupertino, CA 95014
msweet@apple.com

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

Ira McDonald – High North, Inc.
William Wagner – TIC Inc.

10. Change History

10.1. March 28, 2019

Updated as per review from 2019-03-28 IPP WG teleconference, reviewing the changes from 2019-03-04 below. Additional changes:

- Section 3.2.1: add "without restrictions" after "confirms Andy's account is authorized to print"
Section 3.3.2: “The printer is configured to limit access to color printing to only authorized users” --> replace some of the instances of “to” or rewrite the sentence

Section 7.3: Rewrote bullet 3

Section 4.7: Add “TLS” before each TLS message label (e.g. “Certificate message” → “TLS Certificate message”)

GLOBAL: Change all “MUST” to “SHOULD” or alternately change normative to declarative statement (e.g. “MUST provide” → “will provide”) as appropriate

Removed the text at the root of section 5

Move [ITUX.800] reference from informative to normative

10.2. March 4, 2019

Updated with changes to address all comments from first PWG Last Call. Some changes were technical rather than editorial, so another PWG Last Call is needed.

Respondents (10, needed 7 for quorum):

• Rick Yardumian, Canon (RY)
• Smith Kennedy, HP (SK)
• Mike Sweet, Apple (MS)
• Ira McDonald, High North (IM)
• Jeremy Leber, Lexmark (JL)
• Brian Smithson, Ricoh (BS)
• Alan Sukert, Xerox (AS)
• William Wagner, TIC (WW)
• Paul Tykodi (PT)
• Cihan Colakoglu, Kyocera Document Solutions (CC)

Comments (18 TOTAL, 17 RESOLVED, 1 REJECTED):

RY1 - Page 14, Lines 159-174, Section 3.3: Sections 3.3.1 and 3.3.2 are exactly the same except one is for user Lisa and the other is for user Harry. One section is about Authentication Failure and the other is Authorization Failure. This is a bit confusing since
the paragraphs are exactly the same except for the use case user name and the section
titles.

RESOLVED: Updated 3.3.2 to describe an Authorization failure case more
accurately.

RY2 - Page 30, Section 7.3: Section 7.3 is a security recommendation description, where
SHOULD is used for all list items except for item 3 which states “A Printer MUST support
User-provisioned X.509.”. Should this be SHOULD as well?

RESOLVED: (Needs further discussion in IPP WG)

AS1 - Page 23, section 4.7: Minor comment (grammatically sentence did not read
correctly; suggested addition is in red type) that can be ignored if needed to approve -
Lines 272-274: The 'certificate' IPP Authentication method uses X.509 certificate
authentication via TLS. X.509 certificate authentication via TLS and is initiated by the
Printer by sending a Certificate Request message during the Transport Layer Security
(TLS) [RFC5246] handshake.

Also feedback from Cihan Colakoglu that the sentences in an interim draft discussed on
the reflector were not grammatically correct.

RESOLVED: Rewrote first paragraph of section 4.7.

WW1 - All UML Diagrams (Figures 4.1-4.7): The diagrams contain a lot of information but
are unreadable without magnification. The alternative would be to break each transaction
into multiple figures, which would also be cumbersome (and a lot more work).

RESOLVED: Reformatted the diagrams to hopefully make the text larger and more
readable (Since OAuth 2.0 is so complicated, Figures 4.6 and 4.7 will always be
difficult to read, unfortunately...)

WW2 - Line 155, page 14, section 3.2.1: “Andy enters his credential to prove access…”
Presumably, Andy enters his credentials to support he is who he says he is, which may or
may not provide access. Perhaps just “Andy enters his credential.”

RESOLVED: Rewrote the use case to be more clear

WW3 - Lines 159 - 174: Canon commented “Sections 3.3.1 and 3.3.2 are exactly the same
except one is for user Lisa and the other is for user Harry. One section is about
Authentication Failure and the other is Authorization Failure. This is a bit confusing since
the paragraphs are exactly the same except for the use case user name and the section
titles.” I agree. Presumably one can have an account and a valid password but still nor be
authorized to use the printer for some other reason. (para 5.1.3 and para 5.2.3 discuss
this). The use cases should include a clear case of an authentication failure (unless it is
out of scope for this document, in which case it should be under para 3.4.)
RESOLVED: Resolution for RY1 and PT1.

WW4 - Although I may be missing it, the diagrams do not make clear what is an authentication failure vs an authorization failure. (indeed, the distinction between the terms in the diagrams is unclear in many cases, with the Authorization Service clearly doing authentication in many cases). Aside from the Use Cases and the failure handling in section 5, the text does not appear to help in the distinction either.

I recognize that (I think) the common use is that the user is authorized on the basis of authentication credentials, thus:

a. HTTP Status Code 401 Unauthorized: The request has not been applied because it lacks valid authentication credentials.

b. The comment that the use of the 'oauth' authentication method ... depends on the Printer supporting the “oauth-authorization-server-uri” Printer Description attribute).

But some help in distinguishing an Authentication failure from an Authorization failure might be useful.

RESOLVED: All sequence diagrams have been updated. Several points:

1. The authentication failure and authorization failure cases were added to the sequence diagrams in the 20181109 draft; during review at the November 2018 F2F, it was decided that these additions negatively impacted readability and so these changes were backed out.

2. Resolution of RY1 should make more clear the exception case difference between authentication failure and authorization failure.

3. For IPP authentication and authorization success cases, the diagrams do not clearly illustrate the separate authentication vs. authorization steps.

PT1 - Technical Comment – I think that overall the current version of the document lacks clarity because the terms Authentication and Authorization have not been provided definitions, for the purpose of their usage in the document, at the beginning of the document. I believe that definitions for these two terms should be added.


CC1 - Line 14: This is a PWG Best Practice. For the definition of a "PWG Best Practice", see:

Suggestion: This is a PWG Best Practice document. For the definition of “PWG Best Practice”, see:
RESOLVED: Accepted but called it "PWG Best Practices" since that is what the subsection of PWG Process 3.0 section 4.9 is entitled.

CC2 - Lines 158-174: 3.3.1. Authentication Failure Prevents Access / 3.3.2. Authorization Failure Prevents Access

Suggestion: Same as Canon and TIC: We need to differentiate user story of Authentication vs Authorization failure.

RESOLVED: Accepted and corrected as for RY1 and PT1

CC3 - Line 195: these cases, the Printer could still need to acquire the User's identity in order to

Suggestion: these cases, the Printer could still acquire the User's identity in order to

REJECTED: The "need" word is necessary, but "acquire" isn't. In light of this comment and others that suggest more clarity about "authentication" and "authorization" and their functional purposes in IPP, and other LCRC edits, I decided to rewrite the entire paragraph.

CC4 - Lines 221-222: In the 'requesting-user-name' IPP Authentication Method [RFC8011], the Client MUST provides …

Suggestion: In the 'requesting-user-name' IPP Authentication Method [RFC8011], the Client MUST provide …

RESOLVED: Accepted

CC5 - Lines 235-236: It is employed in IPP in much the same way that it is employed in conventional HTTP workflows

Suggestion: It is employed in IPP in much the same way as in conventional HTTP workflows...

RESOLVED: Accepted

CC6 - Lines 248-249: It is employed in IPP in much the same way that it is employed in conventional HTTP workflows

Suggestion: It is employed in IPP in much the same way as in conventional HTTP workflows...

RESOLVED: Accepted

CC7 - Line 268: the OAuth2 authentication scheme [RFC6749], which provides...
768 Question: Is this sentence a placeholder (incomplete); meant to be completed later?
769  RESOLVED: Added missing text
770 CC8 - Line 269: The OAuth2 Bearer Token [RFC6750] which provides...
771 Question: Is this sentence a placeholder (incomplete); meant to be completed later?
772  RESOLVED: Added missing text
773 CC9 - Lines 302-304: Provide possible technical solutions/approaches in this section.
774 Include pros and cons …
775 Question: Is this paragraph a placeholder (incomplete); meant to be completed later?
776  RESOLVED: Added missing text
777 SK1 - All diagrams: The UML sequence diagrams need to illustrate the authentication and
778 authorization request steps in the process.
779  RESOLVED: Updated UML sequence diagrams to better illustrate these steps.

10.3. January 17, 2019
781 Updated with live edits and feedback from the January 17 IPP WG meeting.
782 • Status changed to Stable in preparation for Changed all “might” to “could”
783 • Fixed all IETF RFC URLs to use the “https://tools.ietf.org/html/rfcXXX” format
784 • Changed the OAuth2 recommendations in sections 5.1.4 and 5.2.5 to simply point
    to best practice RFCs elsewhere.
785 • A few other minor editorial changes

10.4. January 16, 2019
788 Changed status to Prototype draft.

10.5. January 9, 2019
790 Added mention of “oauth-authorization-server-uri” and reference to 5100.18 in section 4.6
791 since it is mentioned in the sequence diagram.
10.6. January 7, 2019

- Minor editorial fixes to section 4.
- Editorial fixes to section 3.3.2

10.7. December 22, 2018

Updated with changes and feedback from review in November 2018 PWG F2F:

- Updated exception cases in section 3.3 to delineate authorization and authentication failure exception cases
- Restored all UML diagrams to their previous state, removing the authentication and authorization failure cases
- Rewrote recommendations in section 5.

10.8. November 9, 2018

Updated as per IPP WG review feedback from 2018-10-25:

- Added discussion of SAML 2.0 in appropriate locations in section 4 and 4.7, and added an informative reference to the OASIS SAML 2.0 specification.
- Added authorization and authentication failure and success cases to the sequence diagrams
- Fixed sub-section numbering for section 4
- Resolved all other issues from that review's meeting minutes

10.9. October 19, 2018

Added Printer guidance for how to specify support for multiple authentication methods for a particular URI, and how a Client might discover this and process it.

10.10. September 13, 2018

Updated with additional recommendations for Client and Printer on when (and when not) to rotate HTTP Digest parameters, to prevent operation failure.

10.11. September 5, 2018

Updated as per feedback from PWG August 2018 F2F:
Updated file name and structure to make it a “best practices” document

Moved all the authentication methods to a new section 4

### 10.12. June 29, 2018

Updated as per feedback from PWG May 2018 F2F:

- Added line numbers
- Resolved typos in diagrams in figures 3.5, 3.6, and the “new” 3.7 (TLS)
- Removed the second OAuth2 diagram
- Rewrote the TLS client authentication scheme description (contributed by Mike Sweet) and re-titled the section for its corresponding “uri-authentication-supported” keyword (‘certificate’)

### 10.13. May 10, 2018

Updated figures 6 and 7 (relating to OAuth2) to add a note indicating where the Printer might be able to acquire a user identifier suitable for making policy choices. Also made a few minor editorial updates.

### 10.14. April 30, 2018

Changed to Apache OpenOffice template. Added Mike Sweet as a co-author since he has contributed a great deal of content to the document. Resolved all “to-do” highlighted areas and resolved issues identified in the February 2018 vF2F minutes ([https://ftp.pwg.org/pub/pwg/ipp/minutes/ippv2-f2f-minutes-20180207.pdf](https://ftp.pwg.org/pub/pwg/ipp/minutes/ippv2-f2f-minutes-20180207.pdf)):

- Added sequence diagram for X.509 client authentication
- Added sequence diagram for hybrid 'oauth' / 'digest' authentication
- Many other changes

### 10.15. January 23, 2018

Updated as per email feedback and discussion:

- Fixed some editorial issues with naming HTTP Basic, HTTP Digest, and HTTP Negotiate, and some names of sections.
- Added mention of “printer-xri-supported”.

---

Copyright © 2017-2019 The Printer Working Group. All rights reserved.
• Added additional references.

• Added additional sub-sections to capture Client and Printer recommendations for appropriate behavior when authentication is unsuccessful since the negative cases can vary widely.

10.16. 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.

10.17. August 3, 2017

Initial revision.