Abstract: This best practice document provides implementation guidance on how to best integrate the various authentication mechanisms used over IPP’s HTTP and HTTPS transports into IPP protocol exchanges and the design of authentication user experiences on IPP Client systems.

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


This document is available electronically at:

https://ftp.pwg.org/pub/pwg/ipp/ippauth-20181109.odt

Copyright © 2017-2018 The Printer Working Group. All rights reserved.

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

1. Introduction ................................................................................................................................. 5
2. Terminology ................................................................................................................................. 5
   2.1. Conformance Terminology ................................................................................................. 5
   2.2. Protocol Roles Terminology .............................................................................................. 5
   2.3. Other Terms Used in This Document .................................................................................. 5
3. Requirements ............................................................................................................................... 7
   3.1. Rationale .............................................................................................................................. 7
   3.2. Use Cases ............................................................................................................................ 7
      3.2.1. Authenticated Printing .............................................................................................. 7
   3.3. Exceptions ........................................................................................................................... 8
      3.3.1. Authenticated Printing Blocks Access ........................................................................ 8
4. Client Authentication Methods ................................................................................................... 8
   4.1. The 'none' IPP Authentication Method ............................................................................... 9
   4.2. The 'requesting-user-name' IPP Authentication Method ................................................... 10
   4.3. The 'basic' IPP Authentication Method ............................................................................. 11
   4.4. The 'digest' IPP Authentication Method ............................................................................. 13
   4.5. The 'negotiate' IPP Authentication Method ....................................................................... 15
   4.6. The 'oauth' IPP Authentication Method ............................................................................. 16
   4.7. The 'certificate' IPP Authentication Method ....................................................................... 18
5. Implementation Recommendations .............................................................................................. 20
   5.1. Client Implementation Recommendations ....................................................................... 20
      5.1.1. General Recommendations ....................................................................................... 20
      5.1.2. Handling Authentication Failure ................................................................................. 20
      5.1.3. Handling Authorization Failure .................................................................................. 20
      5.1.4. OAuth2 Recommendations ......................................................................................... 20
   5.2. Printer Implementation Recommendations ........................................................................... 21
      5.2.1. Handling Authentication Failure ................................................................................ 21
      5.2.2. Handling Authorization Failure .................................................................................. 21
      5.2.3. HTTP Digest Recommendations ................................................................................ 21
      5.2.4. OAuth2 Recommendations ......................................................................................... 21
6. Internationalization Considerations ............................................................................................ 21
7. Security Considerations ............................................................................................................. 22
   7.1. Human-readable Strings ........................................................................................................ 22
   7.2. Client Security Considerations ............................................................................................ 22
   7.3. Printer Security Considerations ........................................................................................... 23
8. References.................................................................................................................... 24
  8.1. Normative References.......................................................................................... 24
  8.2. Informative References....................................................................................... 27
9. Authors' Addresses..................................................................................................... 27
10. Change History......................................................................................................... 28
    10.1. November 9, 2018....................................................................................... 28
    10.2. October 19, 2018...................................................................................... 28
    10.3. September 13, 2018.................................................................................. 28
    10.4. September 5, 2018..................................................................................... 29
    10.5. June 29, 2018............................................................................................. 29
    10.6. May 10, 2018............................................................................................. 29
    10.7. April 30, 2018............................................................................................. 29
    10.8. January 23, 2018....................................................................................... 29
    10.9. December 5, 2017...................................................................................... 30
    10.10. August 3, 2017......................................................................................... 30

List of Figures
Figure 4.1 : Sequence diagram for the 'none' IPP Authentication Method......................... 9
Figure 4.2 : Sequence diagram for the 'requesting-user-name' IPP Authentication Method ........................................................................................................................................ 10
Figure 4.3 : Sequence diagram for the 'basic' IPP Authentication Method.......................... 12
Figure 4.4 : Sequence diagram for the 'digest' IPP Authentication Method........................ 14
Figure 4.5 : Sequence diagram for the 'negotiate' IPP Authentication Method.................... 15
Figure 4.6 : Sequence diagram for the 'oauth' IPP Authentication Method.......................... 17
Figure 4.7 : Sequence diagram for X.509 Certificate Authentication Via TLS..................... 19

List of Tables
Table 4.1 : IPP 'certificate' Authentication Method Error Condition Status Codes............. 18
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 authentication methods or TLS certificate authentication. But an IPP Client is not a typical HTTP User Agent (e.g. a Web browser). User experience problems can occur if the Printer or associated authentication infrastructure assumes that all User Agents are web browsers.

This document surveys the common 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 implementers, Printer implementers, and Infrastructure Administrators when implementing support for one of these different HTTP authentication methods in IPP communications, to ensure a high quality printing user experience.

Terminology

1.1. 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.

1.2. 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).
124 *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.

1.3. Other Terms Used in This Document

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

1.4. Acronyms and Organizations

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


133 *PWG*: Printer Working Group, http://www.pwg.org/
2. Requirements

2.1. Rationale

Given the following existing specifications:


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 user experience provided by an IPP Client

2.2. Use Cases

2.2.1. Authenticated Printing

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 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 user interface elements corresponding to the HTTP authentication type. Andy enters his credential to prove access, and the Printer approves access. The laptop then provides the usual print user interface allowing Andy to select print options.
2.3. Exceptions

2.3.1. Authenticated Printing Blocks Access

Lisa is visiting Andy's office and wants to print from her tablet. She finds and selects a
printer on the network. The printer challenges the tablet for authentication, and the tablet
presents an authentication dialog to Lisa. She doesn't have an account, and so she picks a
different printer.

2.4. Out of Scope

The following are considered out of scope for this document:

1. Definition of new HTTP authentication methods

3. 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 allow the requesting Client access
to capabilities such as operations, resources, and attributes. 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.

In some cases, the Printer is not directly involved in the authentication process, such as when OAuth2 is used, or when the Printer depends on an external authentication system. The Printer may not be directly aware of the User's identity following authentication. In these cases, the Printer might still need to acquire the User's identity in order to accurately document the User's identity in the Job Object's Job Status attributes.
or to support IPP operations such as Get-User-Printer-Attributes [IPPGUPA] that depend on the User's identity to provide meaningfully filtered operation responses.

One authentication system not described below is SAML (Security Assertion Markup Language). As of this writing, none of the standard SAML bindings to HTTP directly support IPP. SAML can be used indirectly, by implementing a SAML / OAuth2 gateway.
Each of the authentication method keywords currently registered for "uri-authentication-supported" is described below, with an accompanying sequence diagram for illustration purposes, as well as a discussion of each method's advantages and shortcomings.

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'. 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.

Figure 3.2 illustrates how the 'none' authentication method integrates into an IPP operation request / response exchange. Other authentication methods will expand on this baseline request / response exchange.

Figure 3.1: Sequence diagram for the 'none' IPP Authentication Method
Figure 3.2: Sequence diagram for the ‘none’ IPP Authentication Method

Figure 3.3: Sequence diagram for the ‘none’ IPP Authentication Method
3.1. The 'requesting-user-name' IPP Authentication Method

In the 'requesting-user-name' IPP Authentication Method [RFC8011], the Client MUST provide 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. 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".

Figure 3.5 illustrates how the 'requesting-user-name' authentication method integrates into an IPP operation request / response exchange. This is basically identical to the 'none' method from a protocol perspective.

Figure 3.4: Sequence diagram for the 'requesting-user-name' IPP Authentication Method
IPP With 'requesting-user-name' Authentication

Client System

1. Do something that requires the user to interact with the printer

2. Formulate IPP operation request payload (application/ipp)

3. Perform HTTP POST of request payload

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

4. No HTTP Authentication

   HTTP/1.1 100 Continue

5. Send the application/ipp payload

6. Deliver IPP operation request

7. Formulate IPP operation response

   The value of "requesting-user-name" is assumed to be the user's account identity.

   No challenge for authentication.

8. Return IPP operation response

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

9. Deliver the IPP operation response

10. Process the operation response

11. Present something from the operation response(s)

12. Done

Print Service System

Figure 3.5: Sequence diagram for the 'requesting-user-name' IPP Authentication Method

Figure 3.6: Sequence diagram for the 'requesting-user-name' IPP Authentication Method
3.2. 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 3.7 illustrates how the 'basic' authentication method integrates into an IPP operation request / response exchange.
**IPP Authentication Using HTTP Basic Authentication**

1. Do something that triggers the Client need to interact with Printer
2. Formulate IPP operation request payload (application/ipp)
3. Perform HTTP POST of request payload

   
   ```
   POST /ipp/print HTTP/1.1
   Content-Type: application/ipp
   ```

4. POST /ipp/print HTTP/1.1
   
   ```
   Content-Type: application/ipp
   ```

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
    ```

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 response

17. Formulate IPP operation response

18. Return IPP operation response

19. HTTP/1.1 200 OK

20. Deliver the IPP operation response

21. Process the operation response

22. Present something from the operation response(s)

23. Done
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 3.10 illustrates how the 'digest' authentication method integrates into an IPP operation request / response exchange.
IPP Authentication Using HTTP Digest Authentication

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

5. Start HTTP Digest Authentication

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

6. Leveraged description from
https://en.wikipedia.org/wiki/Digest_access_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"

11. Leveraged description from
https://en.wikipedia.org/wiki/Digest_access_authentication

12. Check access with local auth database

13. Check access with external auth database

14. Approve Access

15. End HTTP Digest Authentication

16. HTTP/1.1 100 Continue

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

18. Deliver IPP operation request

19. Process the operation response

20. Present something from the operation response(s)
Figure 3.12: Sequence diagram for the 'digest' IPP Authentication Method

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.

Figure 3.13 illustrates how the 'negotiate' authentication method integrates into an IPP operation request / response exchange.
Figure 3.13: Sequence diagram for the 'negotiate' IPP Authentication Method
3.3. The 'oauth' IPP Authentication Method

The 'oauth' IPP Authentication method uses the OAuth2 authentication scheme [RFC6749] and the OAuth2 Bearer Token [RFC6750]. OAuth is an authorization framework that uses one or more authentication frameworks, such as SAML 2.0 [SAMLCORE]. Figure 3.15 illustrates how the 'oauth' authentication method integrates into an IPP operation request / response exchange.
Figure 3.15: Sequence diagram for the 'oauth' IPP Authentication Method
3.4. The 'certificate' IPP Authentication Method

Figure 3.17: Sequence diagram for the 'oauth' IPP Authentication Method
The 'certificate' IPP Authentication Method

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

The Printer SHOULD allow both empty and valid X.509 certificates. The Printer SHOULD return the IPP status code listed in Table 3.1 when the corresponding authentication exception occurs. The Client SHOULD respond to the reported status code with the corresponding response listed in Table 3.1.

<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 3.1: IPP 'certificate' Authentication Method Error Condition Status Codes

Figure 3.18 illustrates how the TLS authentication method integrates into an IPP operation request / response exchange.
TLS Client certificate authentication

1. Present dialog with available X.509 certificates.
2. Select and approve X.509 certificate for authentication.
3. Perform HTTP POST of request payload...
4. Deliver the IPP operation request
5. Generate the Master Secret and session keys from Pre-Master Secret
6. Perform the operation response
7. Deliver the IPP operation response
8. Generate the Master Secret and session keys from Pre-Master Secret
9. Perform the operation response
10. Deliver IPP operation request
11. POST (without HTTP/1.1) Expect: 100-continue
12. Deliver IPP operation request
13. End HTTP TLS Authentication with Client Certificate

HTTP POST with request payload
1. Add hash value (if required)
2. HTTP POST with request payload

Authorization Success

Authorization Failure
Figure 3.20: Sequence diagram for X.509 Certificate Authentication Via TLS

Implementation Recommendations

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.

3.5. Client Implementation Recommendations

3.5.1. 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.

3.5.2. Handling Authentication Failure

A Client that encounters an authentication failure SHOULD offer the User another opportunity to provide valid authentication credentials, and should only 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).

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

Handling Authorization Failure

If a Printer rejects authentication credentials provided by a Client in response to an authentication challenge following an IPP operation request, the Printer MAY return an IPP operation response with the 'client-error-not-authenticated' status code, or the error may be reported at the protocol layer managing the authentication process. Either way, if the Printer leaves the connection open after reporting an authentication failure, the Client SHOULD treat the connection the same way it handles a stalled connection, and close it after a short period of time.

3.5.3. OAuth2 Recommendations

The OAuth2 authorization service may have a complicated user presentation. If possible, select a presentation alternative that is the least complicated or the most similar to the user experience provided for older authentication methods (HTTP Basic or HTTP Digest) that may be more familiar to the user.

3.6. Printer Implementation Recommendations

3.6.1. 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, the authentication failure will be reported at the layer managing the authentication process (HTTP or TLS layer for the methods outlined in this document).

3.6.2. Handling Authorization Failure

If a Printer receives an IPP operation request, challenges the Client for authentication, and the credentials are valid, then the Printer will receive and process the IPP operation request. But if the identity bound to those credentials does not have access rights to the requested IPP resources (IPP attributes or operations), then the Printer SHOULD indicate
the authorization failure by returning the 'client-error-not-authorized' IPP operation status code.

If a Printer receives an IPP operation request, challenges the Client for authentication, and the authentication process fails, the Printer SHOULD send an appropriate IPP operation response indicating the cause of the failure.

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 may not be cacheable by the Client, and this can cause a significant burden to the Client, degrade, and/or cause 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.

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. This will help to avoid additional HTTP message thrash that provides little real security value and wastes bandwidth.

OAuth2 Recommendations

An To align with existing Client authentication user experience for HTTP Basic or HTTP Digest authentication, the OAuth2 Authorization Server SHOULD use HTTP Basic or HTTP Digest authentication rather than presenting an authentication dialog page using its own web content. If that isn’t practical, an OAuth2 Authorization Service used in an IPP printing workflow SHOULD direct a Client to an authentication page that facilitates an appropriate presentation on even limited Client systems such as smart phones. In deployments where SAML [SAMLCORE] is used for network authentication and IPP authentication is needed, an OAuth2 / SAML gateway SHOULD be implemented to provide compatibility with IPP.

4. 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
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

5. Security Considerations

5.1. 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

5.2. Client Security Considerations

The following are the security recommendations for an IPP Client. A Client SHOULD follow these recommendations:

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.

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 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 may be necessary to validate the Printer's identity.

4. A Client SHOULD provide a means to allow the User to examine a Printer's provided identity.

5. A Client SHOULD provide one or more means of notification when it is engaging with a previously encountered Printer whose identity has changed.

3. OAuth2 Considerations

4. A Client supporting OAuth2 SHOULD conform to the recommendations in “Proof Key for Code Exchange by OAuth Public Clients” [RFC7636] and “OAuth2 SHOULD be for Native Apps” [RFC8252] if followed, since the print system provided its own user interface presentation and controls for handling the OAuth2 authentication steps, to mitigate the risks described therein.

6. A Client SHOULD use the most secure authentication method available for a given Printer. In some cases, a Printer might 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.

1. The recommendations in “OAuth 2 for Native Apps” [RFC8252] should be followed if the print system provides its own user interface presentation and controls for handling the OAuth2 authentication steps, to mitigate the risks described therein.

5. In some cases, a Printer might 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. The Client SHOULD prefer more robust authentication methods over less robust or less secure authentication methods.

1. Printer Security Considerations

   The following are the security recommendations for an IPP Printer.

   An IPP Printer:

   A Printer SHOULD securely store at rest any personally identifiable information (PII) and authentication credentials such as passwords that are local to the Printer.

   1. 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).

   2. A Printer MUST support User-provisioned X.509 certificates that persist across power cycles. These certificates MUST NOT be automatically renewed or replaced.

   3. 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.

   4. In cases where the Printer supports more than one authentication method for a particular URI, the Printer MUST 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.

   5. A Printer supporting OAuth2 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.

   6. 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).

   SHOULD support User-provisioned X.509 certificates:
1. The certificate MUST persist across power cycles

2. The certificate MUST NOT be automatically renewed or replaced

3. The certificate SHOULD have a maximum expiration of 3 years from the date of issuance

4. The certificate SHOULD NOT use MD5 or SHA-1 hashes

5. SHOULD support self-generated self-signed X.509 certificates:
   a. The certificate persists across power cycles
   b. The certificate has a minimum default expiration of 5 years from the date of issuance/generation
   c. The certificate is automatically renewed (regenerated), using a new private key if the previous certificate has expired
   d. The certificate is generated using the mDNS, DHCP and/or manually-configured DNS hostname(s) and regenerated whenever these change
   e. The Printer MUST be able to generate RSA certificates with a key length of 2048 bits using SHA-256 hash
   f. The Printer SHOULD be able to generate ECDSA certificates using the secp256r1 (P-256), secp384r1 (P-384), or secp521r1 (P-521) curves and a SHA-256 hash.
   g. The Printer MUST NOT generate self-signed certificates using MD5 or SHA-1 hashes

8. In cases where the Printer supports more than one authentication method for a particular URI, the Printer can convey these alternatives by including the same URI multiple times in “printer-uri-supported”, and specifying a different authentication method for each corresponding values specified by the “uri-authentication-supported” attribute.

2. References

5.3. Normative References

Best Practice – IPP Authentication Methods (IPPAUTH)


(STD92]


5.4. Informative References


6. 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.

7. Change History

7.1. 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
7.2. 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.

7.3. 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.

7.4. 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

7.5. 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”)

7.6. 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.

7.7. 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):
7.8. 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”.
- 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.

7.9. 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.

7.10. August 3, 2017

Initial revision.