



Liberty ID-WSF Profiles for **Liberty-Enabled** ~~Liberty enabled~~ User Agents and Devices

Version:

2.0-errata-v1.0

Editors:

Robert Aarts, Hewlett-Packard
Jukka Kainulainen, Nokia Corporation
John Kemp, Nokia Corporation

Contributors:

Darryl

Champagne
, IEEE-ISTO

Rob

Lockhart
, IEEE-ISTO

Abstract:

User agents or devices, *i.e.*, personal computers, mobile terminals, etc., participate in ID-WSF transactions in various ways. This document specifies profiles for some cases where user agents or devices act as an ID-WSF entity, *i.e.*, execute software that implements at least parts of the ID-WSF specifications.

Filename: liberty-idwsf-client-profiles-2.0-diff-v1.0.pdf

1

Notice

2 This document has been prepared by Sponsors of the Liberty Alliance. Permission is hereby granted to use the document
3 solely for the purpose of implementing the Specification. No rights are granted to prepare derivative works of this
4 Specification. Entities seeking permission to reproduce portions of this document for other uses must contact the Liberty
5 Alliance to determine whether an appropriate license for such use is available.

6 Implementation of certain elements of this document may require licenses under third party intellectual property rights,
7 including without limitation, patent rights. The Sponsors of and any other contributors to the Specification are not and
8 shall not be held responsible in any manner for identifying or failing to identify any or all such third party intellectual
9 property rights. **This Specification is provided "AS IS", and no participant in the Liberty Alliance makes any**
10 **warranty of any kind, express or implied, including any implied warranties of merchantability, non-infringe-**
11 **ment of third party intellectual property rights, and fitness for a particular purpose.** Implementers of this
12 Specification are advised to review the Liberty Alliance Project's website (<http://www.projectliberty.org/>) for infor-
13 mation concerning any Necessary Claims Disclosure Notices that have been received by the Liberty Alliance
14 Management Board.

15 Copyright © 2007 2FA Technology; Adobe Systems; Agencia Catalana De Certificacio; America Online, Inc.; Amer-
16 ican Express Company; Amsoft Systems Pvt Ltd.; Avatier Corporation; BIPAC; BMC Software, Inc.; Axalto; Bank of
17 America Corporation; Beta Systems Software AG; BIPAC; British Telecommunications plc; Computer Associates
18 International, Inc.; Credentica; DataPower Technology, Inc.; Deutsche Telekom AG, T-Com; Diamelle Technologies,
19 Inc.; Diversinet Corp.; Drummond Group Inc.; Enosis Group LLC; Entrust, Inc.; Epok, Inc.; Ericsson; Falkin Systems
20 LLC; Fidelity Investments; Forum Systems, Inc.; France Télécom; French Government Agence pour le développement
21 de l'administration électronique (ADAE); Fugen Solutions, Inc; Fulvens Ltd.; GSA Office of Governmentwide Policy;
22 Gamefederation; Gemalto; General Motors; GeoFederation; Giesecke & Devrient GmbH; Hewlett-Packard GSA Office
23 Company; Hochhauser & Co., Policy; Hewlett-Packard LLC; IBM Corporation; Intel Corporation; Intuit Inc.; Kant-
24 ega; Kayak Interactive; Livo Technologies; Luminance Consulting Services; MasterCard International; MedCommons
25 Inc.; Mobile Telephone Networks (Pty) Ltd; NEC Corporation; NTT DoCoMo, Inc.; Netegrity, Inc.; Neustar, Inc.;
26 New Zealand Government State Services Commission; Nippon Telegraph and Telephone Corporation; Nokia Corpo-
27 ration; Novell, Inc.; NTT DoCoMo, Inc.; OpenNetwork; Oracle Corporation; Ping Identity Corporation; RSA Security
28 Inc.; Reactivity Inc.; Royal Mail Group plc; RSA Security Inc.; SAP AG; Senforce; Sharp Laboratories of America;
29 Sigaba; SmartTrust; Sony Corporation; Sun Microsystems, Inc.; Supremacy Financial Corporation; Symlabs, Inc.;
30 Telecom Italia S.p.A.; Telefónica Móviles, S.A.; Telenor R&D; Thales e-Security; Trusted Network Technologies;
31 UNINETT AS; UTI; VeriSign, Inc.; Vodafone Group Plc.; Wave Systems Corp. All rights reserved.

32 Liberty Alliance Project
33 Licensing Administrator
34 c/o IEEE-ISTO
35 445 Hoes Lane
36 Piscataway, NJ 08855-1331, USA
37 info@projectliberty.org

38 Contents

39	1. Notation and Conventions	5
40	2. Overview	6
41	3. LUAD-WSC Profile.....	8
42	3.1. Rules for WSPs that offer service to LUADs	8
43	3.2. Examples.....	9
44	4. LUAD acting as WSP.....	10
45	4.1. LUAD-WSP profile.....	10
46	5. LUAD implementations of a Discovery Service.....	11
47	5.1. LUAD-DS Profile	11
48	References.....	12

49 1. Notation and Conventions

50 This specification uses schema documents conforming to W3C XML Schema (see [Schema1-2]) and normative text
51 to describe the syntax and semantics of XML-encoded messages.

52 The key words "MUST," "MUST NOT," "REQUIRED," "SHALL," "SHALL NOT," "SHOULD," "SHOULD NOT,"
53 "RECOMMENDED," "MAY," and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

54 These keywords are thus capitalized when used to unambiguously specify requirements over protocol and application
55 features and behavior that affect the interoperability and security of implementations. When these words are not cap-
56 italized, they are meant in their natural-language sense.

57 Namespaces

- 58 • The prefix disco: represents the namespace defined in [LibertyDisco].
- 59 • The prefix sa: represents the namespace defined in [LibertyAuthn].
- 60 • The prefix sec: represents the namespace defined in [LibertySecMech].
- 61 • S: represents the namespace defined in [SOAPv1.1]

62 2. Overview

63 The ID-WSF specifications define a number of protocols that enable any party to act as a *Web Service Consumer*, a
64 *Web Service Provider*, or both. When user agents or devices wish to act in these roles, some particular issues need to
65 be addressed and hence additional specifications are useful to guarantee interoperability. The Liberty Alliance specifies
66 the [ID-WSF Authentication Service](#) by which a WSC on a user agent or device may authenticate to an identity provider,
67 and [LibertyPAOS](#) to enable a user agent or device to act as a WSP. Also, whenever a WSC or WSP acts as a user agent
68 it typically represents only a very small number of users, hence there are some particular considerations regarding
69 privacy.

70 User agents and devices that send or consume protocol messages specified in the ID-WSF (, ID-FF or SAML) speci-
71 fications are called *Liberty enabled User Agents and Devices*, abbreviated as *LUAD*. The defining characteristic of a
72 LUAD is that it is closely associated with one user (or a few users, such as a family); the LUAD represents that user.
73 This is very different from a web-site that acts as WSC or WSP and may represent thousands of users. In addition, a
74 LUAD is often, but certainly not always, *not* a highly-available HTTP server, unlike web-site based WSCs and WSPs.

75 To illustrate some of the issues we briefly sketch out a scenario where a LUAD acts as a WSC in a typical ID-WSF
76 setting. The following, as well as the remainder of this document, assumes familiarity with the Liberty ID-WSF spec-
77 ifications, especially the [Discovery Service](#) and [Security Mechanisms](#).

78 Any WSC that wishes to contact an ID-WSF WSP requires ~~an `Service-Instance-Endpoint` Reference~~ and often
79 some security tokens. A WSC typically obtains these from a Liberty ID-WSF Discovery Service (discovery service).
80 However, the discovery service is a WSP too, so for the WSC to make a request to the discovery service, it needs ~~an~~
81 `<wsa:EndpointReference>` and security tokens for the discovery service.

82 A WSC can get such discovery service specific information when it acts as an SP during a single-sign-on transaction
83 using SAML (or ID-FF); the identity provider can insert in a response an `<AttributeStatement>` containing the
84 necessary information to contact the discovery service. This process is informally known as "bootstrapping ID-
85 WSF" (see [[LibertyDisco](#)]).

86 But a LUAD-WSC is not a web-site that acts as SP. So when the LUAD-WSC needs to contact the discovery service
87 it needs somehow to contact a party that can issue the `<wsa:EndpointReference>` and tokens needed. **Here**, we rec-
88 ommend that the LUAD-WSC obtains this information through the Liberty ID-WSF Authentication Service
89 ([\[LibertyAuthn\]](#)) offered by an identity provider.

90 The identity provider will need to authenticate the LUAD -- this is similar to the identity provider authenticating
91 Principals that use a browser. As the LUAD-WSC is not a full-blown browser, however, it may not be able to present
92 a login form.

93 The identity provider and LUAD should use a protocol for authentication. The use of [[LibertyAuthn](#)] is recommended
94 for this purpose.

95 Once the LUAD-WSC can make requests to the discovery service it can ask the discovery service for descriptions and
96 tokens for a particular identity service type (a WSP). If the WSP that is referred to in the discovery service response
97 requires security tokens, the discovery service will create such tokens. Normally such tokens include a `ProviderID`
98 for the WSC and require that the WSC can authenticate as that provider to the WSP, perhaps by signing the request
99 with a particular key. A **LUAD-WSC, however**, does not have a `ProviderID` as a `ProviderID` could compromise the
100 privacy of the LUAD user: the LUAD-WSC would show the same `ProviderID` to various WSPs allowing the WSPs
101 to collude about the LUAD-WSC and hence about the user. Thus the content of security tokens should be profiled for
102 various situations.

103 In summary, this document then specifies how *LUAD* implementations should utilize the various Liberty Alliance
104 specifications in order to enable particular scenarios while ensuring a high degree of interoperability, security and

105 privacy. The following sections specify and discuss profiles for particular uses of a LUAD. Note that in each section,
106 profiles are defined for both the LUAD as well as for the providers that (wish to) interact with the LUAD.

107 3. LUAD-WSC Profile

108 A LUAD-WSC will often need to authenticate to a provider; for example when that LUAD-WSC wants to make a
109 request to a discovery service. The discovery service may have been set up to require a security token; web-site based
110 WSCs typically obtain such a token during a authentication transaction with an identity provider associated to that
111 discovery service. But with a LUAD-WSC there may not be an associated browsing session, hence no interaction with
112 an identity provider has occurred and the WSC cannot have a valid security token for the discovery service. In another
113 typical scenario the WSP is not an ID-WSF WSP, *i.e.*, not an "identity providing" service but an "identity consuming"
114 service (here we abbreviate those non-ID-WSF Web Service Providers as *wSP* to indicate that these are a subclass of
115 SPs). A LUAD-WSC that requests service from such *wSP*s may need to obtain SAML (or ID-FF) authentication
116 assertions that will be presented as security tokens to the *wSP*.

117 As the LUAD represents at most a few users, the LUAD should not use a single authentication identity towards different
118 providers. To achieve the required level of security and privacy the LUAD and provider must carefully choose the
119 authentication mechanism and nature of credentials.

120 A LUAD-WSC implementation must adhere to the following rules:

- 121 1. The LUAD-WSC SHOULD avoid being traceable across providers. Hence, the LUAD SHOULD NOT authen-
122 ticate to different providers using a single credential.

123 **Note**

124 This implies that if a LUAD-WSC employs [message level confidentiality protection](#), different signing
125 keys should be used in communication with each individual provider.

- 126 2. If a LUAD-WSC is required to authenticate to a provider directly, because it does not have or cannot obtain
127 security tokens, the LUAD-WSC SHOULD authenticate using [LibertyAuthn](#).

128 **Note**

129 This applies to situations where the LUAD *itself* needs to assert its identity to a provider -- typically only
130 when a LUAD authenticates to an identity provider. In most cases a LUAD-WSC can obtain (bearer)
131 security tokens from a Liberty ID-WSF Discovery Service and would include these tokens in the message
132 to the WSP.

- 133 3. A LUAD-WSC SHOULD use the ID-WSF Authentication Service specified in [LibertyAuthn](#) to obtain security
134 tokens from an identity provider; these tokens can then be used when submitting a `<disco:Query>` to a Discovery
135 Service.

- 136 4. A LUAD-WSC that wishes to interact with a WSP SHOULD support at least the

137 `urn:liberty:security:2005-02:TLS:Bearer` security mechanism as specified in [LibertySecMech](#).

138 **Note**

139 Note that these rules *do* allow the LUAD to authenticate to a provider using a client certificate. However, that
140 same certificate should not be used to authenticate to another provider. For example a LUAD-WSC could use
141 its certificate to authenticate to a discovery service or an identity provider (to both if both interfaces are offered
142 by one provider) but not then to another WSP.

143 3.1. Rules for WSPs that offer service to LUADs

144 ID-WSF compliant WSPs that register with a discovery service SHOULD support at least the `urn:liberty:`
145 `security:2005-02:TLS:Bearer` security mechanism as specified in [LibertySecMech](#).

146 **3.2. Examples**

147 See [LibertyAuthn] for examples of interactions of a LUAD-WSC.

148 4. LUAD acting as WSP

149 A WSP that is deployed on a LUAD is again not very different from a network WSP. One issue for a client-WSP is
150 reachability: a LUAD is typically not acting as a HTTP/SOAP server, may be behind a firewall, and does not have a
151 fixed IP address.

152 A second issue is that a LUAD-WSP, by definition, offers service for only one, or a few, Principals. Hence, the LUAD-
153 WSP cannot have a *service provider* identity. Normally a WSP needs to offer a **ProviderID** and metadata that WSCs
154 use to construct requests. A LUAD-WSP should not have a **ProviderID** and, hence, can not ~~cannot~~ publish metadata.
155 Metadata and signing keys make the client traceable to different WSCs, compromising the privacy of the LUAD user.

156 Note

157 A [Liberty ID-WSF Discovery Service](#) hosted on a LUAD has to satisfy additional rules (see next section).

158 4.1. LUAD-WSP profile

159 A LUAD-WSP must adhere to the following rules:

- 160 1. It is RECOMMENDED that LUADs that are not normally reachable expose ID-WSP web services over [LibertyPAOS](#)
161

162 Note

163 Note that future versions of the ID-WSF specifications may include SOAP bindings for alternative ap-
164 proaches, such as SIP.

- 165 2. The LUAD-WSP SHOULD avoid being traceable. If the WSP uses [message level confidentiality protection](#),
166 different signing keys for communications with different WSCs SHOULD be used.

- 167 3. As the LUAD-WSP is not an entity different from the Principal it represents, it should not have a **ProviderID**. A
168 discovery service ~~can not~~ cannot issue an ~~EndpointReferenceService Instance Endpoint Reference~~ for en-
169 tities that do not have a **ProviderID**. Hence, A LUAD-WSP SHOULD NOT register with a Discovery Service.

170 An ID-WSF WSC that requests services from a LUAD-WSP must adhere to the following rules:

- 171 1. If authentication of the WSP is needed it is RECOMMENDED that SP/WSCs authenticate the LUAD-WSP using
172 SAML (or ID-FF), presumably before making an ID-WSF request to the PAOS-exposed WSP.

173 See [\[LibertyPAOS\]](#) for an example of interaction with a LUAD-WSP. Another example is given in [Section 5](#).

174 5. LUAD implementations of a Discovery Service

175 A LUAD implementation of a [discovery service](#), *i.e.*, a LUAD-DS, can be useful as a discovery service can inform
176 parties in its immediate proximity about identity services for the user of the LUAD. For example a LUAD-DS could
177 inform a mall entrance about a personal profile service, or inform a parking exit about a payment service. As with any
178 LUAD-WSP implementations there are some issues around traceability of the client, but in a discovery service these
179 problems are more important as a discovery service very likely must issue signed security tokens to parties that sub-
180 sequently will submit those tokens to a WSP.

181 5.1. LUAD-DS Profile

182 An ID-WSF discovery service that executes at a LUAD must adhere to the following rules:

- 183 1. The LUAD-DS implementation SHOULD adhere to the rules defined for [LUAD-WSP implementations](#).
- 184 2. The key that the LUAD-DS uses to sign security tokens SHOULD be unique for each WSP that registers with the
185 LUAD-DS. The LUAD-DS SHOULD inform the WSP about the key when the WSP registers with the LUAD-
186 DS, *i.e.*, the LUAD should include the key in the `disco:ModifyResponse` as specified in [\[LibertyDisco\]](#).

187 When the LUAD-DS sends key material it MUST ensure [Transport Layer Channel Protection](#), and in the presence
188 of intermediaries MUST also ensure [Message Confidentiality Protection](#), using one of the mechanisms specified
189 in [\[LibertySecMech\]](#).

References

Normative

- 192 [LibertyDisco] Cahill, Conor, Hodges, Jeff, eds. "Liberty ID-WSF Discovery Service Specification," Version 2.0-
193 errata-v1.0, Liberty Alliance Project (29 November, 2006). <http://www.projectliberty.org/specs>
- 194 [LibertyInteract] Aarts, Robert, Madsen, Paul, eds. "Liberty ID-WSF Interaction Service Specification," Version 2.0-
195 errata-v1.0, Liberty Alliance Project (21 April, 2007). <http://www.projectliberty.org/specs>
- 196 [LibertyPAOS] Aarts, Robert, Kemp, John, eds. "Liberty Reverse HTTP Binding for SOAP Specification," Version
197 2.0, Liberty Alliance Project (30 July, 2006). <http://www.projectliberty.org/specs>
- 198 [LibertySecMech] Hirsch, Frederick, eds. "Liberty ID-WSF Security Mechanisms Core," Version 2.0-errata-v1.0,
199 Liberty Alliance Project (21 April, 2007). <http://www.projectliberty.org/specs>
- 200 [LibertyAuthn] Hodges, Jeff, Aarts, Robert, Madsen, Paul, Cantor, Scott, eds. "Liberty ID-WSF Authentication, Single
201 Sign-On, and Identity Mapping Services Specification," Version 2.0-errata-v1.0, Liberty Alliance Project (28
202 November, 2006). <http://www.projectliberty.org/specs>
- 203 [LibertySOAPBinding] Hodges, Jeff, Kemp, John, Aarts, Robert, Whitehead, Greg, Madsen, Paul, eds. "Liberty ID-
204 WSF SOAP Binding Specification," Version 2.0-errata-v1.0, Liberty Alliance Project (21 April, 2007). [http://](http://www.projectliberty.org/specs)
205 www.projectliberty.org/specs
- 206 [LibertyIDWSFv20Errata] Champagne, Darryl, Lockhart, Rob, Tiffany, Eric, eds. "Liberty ID-WSF 2.0 Errata," Ver-
207 sion 1.0, Liberty Alliance Project (13 April, 2007). <http://www.projectliberty.org/specs>
- 208 [RFC2119] S. Bradner "Key words for use in RFCs to Indicate Requirement Levels," RFC 2119, The Internet Engi-
209 neering Task Force (March 1997). <http://www.ietf.org/rfc/rfc2119.txt>
- 210 [RFC2616] Fielding, R., Gettys, J., Mogul, J., Frystyk, H., Masinter, L., Leach, P., Berners-Lee, T., eds. (June 1999).
211 "Hypertext Transfer Protocol -- HTTP/1.1," RFC 2616, The Internet Engineering Task Force [http://](http://www.ietf.org/rfc/rfc2616.txt)
212 www.ietf.org/rfc/rfc2616.txt
- 213 [RFC3066] Alvestrand, H., eds. (January 2001). "Tags for the Identification of Languages," RFC 3066., Internet En-
214 gineering Task Force <http://www.ietf.org/rfc/rfc3066.txt>
- 215 [Schema1-2] Thompson, Henry S., Beech, David, Maloney, Murray, Mendelsohn, Noah, eds. (28 October 2004).
216 "XML Schema Part 1: Structures Second Edition," Recommendation, World Wide Web Consortium [http://](http://www.w3.org/TR/xmlschema-1/)
217 www.w3.org/TR/xmlschema-1/
- 218 [SOAPv1.1] "Simple Object Access Protocol (SOAP) 1.1," Box, Don, Ehnebuske, David, Kakivaya, Gopal, Layman,
219 Andrew, Mendelsohn, Noah, Nielsen, Henrik Frystyk, Winer, Dave, eds. World Wide Web Consortium W3C
220 Note (08 May 2000). <http://www.w3.org/TR/2000/NOTE-SOAP-20000508/>

Informative

- 222 [LibertyIDPP] Kellomäki, Sampo, Lockhart, Rob, eds. "Liberty ID-SIS Personal Profile Service Specification," Ver-
223 sion 1.1, Liberty Alliance Project (29 September, 2005). <http://www.projectliberty.org/specs>