Liberty Architecture Implementation Guidelines

Draft Version 1.1-06

5 November 2002

Document identifier: draft-liberty-architecture-implementation-guidelines-06

Notice

Editors

63

64   Lena Kannappan, France Telecom, lena.kannappan@rd.francetelecom.com

65   Matthieu Lachance, OpenWave, matthieu.lachance@openwave.com

66   John Kemp, IEEE-ISTO

Contributors

67

ActivCard	Netegrity
American Express Travel Related Services	NeuStar
America Online, Inc.	Nextel Communications
Bank of America	Nippon Telegraph and Telephone Company
Bell Canada	Nokia Corporation
Cingular Wireless	Novell, Inc.
Cisco Systems, Inc.	NTT DoCoMo, Inc.
Citigroup	OneName Corporation
Communicator, Inc.	Openwave Systems Inc.
Consignia	PricewaterhouseCoopers LLP
Deloitte & Touche LLP	Register.com
EarthLink, Inc.	RSA Security Inc
Electronic Data Systems, Inc.	Sabre Holdings Corporation
Entrust, Inc.	SAP AG
Ericsson	SchlumbergerSema
Fidelity Investments	SK Telecom
France Telecom	Sony Corporation
Gemplus	Sun Microsystems, Inc.
General Motors	United Airlines
Hewlett-Packard Company	VeriSign, Inc.
i2 Technologies, Inc.	Visa International
Intuit Inc.	Vodafone Group Plc
MasterCard International	Wave Systems
NEC Corporation	

68

69

Document History

70

Rev	Date	By Whom	Description
00	02-Apr-02	Lena Kannappan	Initial outline
01	12-Apr-02	Lena Kannappan	Introduced new Columns and Tables to separate ZIC-UA and LECP requirements
02	26-Apr-02	Matthieu Lachance Lena Kannappan	• Renamed the document from Static Conformance Requirements to Liberty Architecture Implementation Guidelines document as it no longer consists of Mandatory or Optional requirements for identity provider or SP or LECP. • Restructured the document to reflect that the document is service or deployment focused.

			• Recommended good identity provider and SP profiles were added. • Checklists of requirements for identity provider, SP and LECP were created.
03	09-May-02	Lena Kannappan	• Changes based on Change Requests #1051, and #1061 • Various minor editorial changes to indicate the recent section number changes to Authentication Context specifications.
04	16-May-02	Lena Kannappan	• Corrected CR# to 1061 in the previous Document History (May 09th). • Editorial changes to sections 3.4, 3.5 and 3.6 to reflect changes to Authentication Context specifications ver 06.
05	29-Oct-02	Lena Kannappan	• Inclusion of Reference to Liberty Architecture Overview document for Policy/Security and Technical notes associated with implementations. • Changes based on CR#1126 : two references to the "DoFederate" element of the AuthNRequest in the sections for IDP and SP guidelines to "Federate" element. • Sections 2.1, 2.2, 3.1 and 3.2 modified to address CR#1162: Register Name Identification extended to include IDP and SP initiation of changes, • Various editorial changes.
06	05-Nov-2002	John Kemp	• Small editorial changes for format • Fixed several of the references • Tightened language in 2.1, 2.2 for CR #1162

71
72

72

Table of Contents

87

88

1   Introduction

This document defines the recommended implementation guidelines and checklists for the Liberty architecture focused on deployments for the service-providing entities: service providers, identity providers, and Liberty-enabled clients or proxies (LECPs). It is intended to provide recommended implementation guidelines to Liberty component developers to help them decide what they need to implement to meet their business needs. Because Liberty Phase 1 does not provide formal compliance, this document does not contain any conformance requirements — only recommendations. A recommended profile tailored according to the high-level Liberty features is provided for different Liberty service-providing entities. Implementers facing specific needs can decide to implement what they need and claim support for each specific feature separately.

The document also provides a checklist of requirements based on the following Liberty architecture specification categories that implementers can use to advertise their supported feature set:

- Functionality in the Liberty protocols and schemas described

- Bindings and profiles defined for each Liberty protocol type (specific interactions between identity providers, service providers, and LECPs)

- The authentication request and reply context-specific information

Definitions for Liberty-specific terms can be found in [LibertyGloss]. Note: Phrases and numbers in brackets [ ] refer to other documents; details of these references can be found in Section 4 (at the end of this document).

Policy/Security and Technical notes related to implementations are covered by Liberty Architecture Overview document associated with this Implementation Guidelines document specified by [LibertyArchOverview].

2   Recommended Liberty Architecture Implementation Guidelines

The recommended implementation guidelines for identity providers, service providers, and LECPs are listed in the tables in 2.1 through 2.3. The guidelines refer to front-channel-based and back-channel-based mechanisms. *Front channel* is described as a communication channel where HTTP redirect-, GET-, and POST-based request and response protocol messages between the identity provider and the service provider flow through the Web browser. *Back channel* is a SOAP/HTTP-based direct communication channel between the identity provider and the service provider. A service provider with SOAP client support is considered to be a "back-channel-capable SP" whereas a "basic SP" is not back-channel-capable.

2.1   Identity Provider Implementation Guidelines

Liberty Feature	Recommendations
Single Sign-On	It is strongly recommended that identity providers support the LECP single sign-on profile to ensure forward compatibility. The LECP profile is intended for future clients of all kinds (thin and thick) as well as existing wireless thin clients (WML, HDML, etc) when used with a LEP.
	Identity providers that want to support existing HTML client environments should implement the browser artifact and the browser POST single sign-on profiles.

Liberty Feature	Recommendations
	To support existing WML client in environments that do not contain any LEP, identity providers should support the WML single sign-on profile.
Identity Federation	Identity providers that want to support permanent identity linking between service providers and identity providers (beyond the stateless single sign-on association) should support the `<Federate>` element of the `<AuthnRequest>` for all the supported single sign-on profiles.
Federation Termination Notification	Identity providers that support identity federation should also support the Federation Termination Notification Protocol. When supported, both service-provider-initiated and identity-provider-initiated federation termination notification should be supported. Liberty offers two federation termination notification mechanisms: • Front channel, or HTTP-redirect-based • Back channel, or SOAP-based As a minimum, identity providers should support the front-channel-based mechanism. Identity providers that want to support back-channel-capable SPs should implement both mechanisms.
Name Registration	The Name Registration Protocol allows the service provider to use its own opaque handle to identify the Principal when communicating with the identity provider (rather than using the identity provider's opaque handle generated during federation). This protocol also allows the identity provider to register a new name identifier with the service provider at any time after federation. When supported, both service-provider-initiated and identity-provider-initiated Name Registration should be supported. Liberty offers two Name Registration mechanisms: • Front channel, or HTTP-redirect-based • Back channel, or SOAP-based At a minimum, identity providers should support the front-channel-based mechanism. Identity providers that want to support back-channel-capable SPs should implement both mechanisms.
Single Logout	The Single Logout Protocol allows logging out a Principal from all its active sessions to service providers, linked to an identity provider. Identity providers keeping trace of the Principal's service provider sessions should implement this feature. When supported, both service-provider-initiated and identity-provider-initiated single logout should be supported. Liberty offers two single logout mechanisms: • Front channel, or HTTP-redirect-based • Back channel, or SOAP-based As a minimum, identity providers supporting this feature should support the front-channel-based mechanism. Identity providers that want to support back-channel-capable SPs should implement both mechanisms.
Identity Provider Introduction	Identity providers that want to support more than a single circle of trust simultaneously should support the Identity Provider Introduction Protocol.

121

122 ## 2.2   Service Provider Implementation Guidelines

123 In general service providers are divided in two categories: the back-channel-capable SPs and the
124 basic SPs (that are not back-channel-capable).

125

125

Liberty Feature	Recommendations
Single Sign-On	It is strongly recommended that service providers support the LECP single sign-on profile to ensure forward compatibility. The LECP profile is intended for future clients of all kinds (thin and thick) as well as existing wireless thin clients (WML, HDML, etc) when used with a LEP. Service providers that want to support existing HTML client environments should implement the browser artifact and the browser POST single sign-on profiles. To support existing WML client in environments that do not contain any LEP, service providers should support the WML single sign-on profile.
Identity Federation	Service providers that want to support permanent identity linking between service providers and identity providers (beyond the stateless single sign-on association) should support the `<Federate>` element of the `<AuthnRequest>` for all the supported single sign-on profiles.
Federation Termination Notification	Service providers that support identity federation should also support the Federation Termination Notification Protocol. When supported, both service-provider-initiated and identity-provider-initiated federation termination notification should be supported. Service providers should support either the front-channel or back-channel federation termination notification mechanisms depending on their respective capabilities although nothing prevents them from supporting both mechanisms if desired.
Name Registration	The Name Registration Protocol allows the service provider to use its own opaque handle to identify the Principal when communicating with the identity provider (rather than using the identity provider's opaque handle generated during federation). This protocol also allows the service provider to register a new name identifier with the identity provider at any time after federation. Service providers should support either the front-channel or back-channel Name Registration mechanisms depending on their respective capabilities although nothing prevents them from supporting both mechanisms if desired.
Single Logout	The Single Logout Protocol allows logging out a Principal from all its active sessions to service providers, linked to an identity provider. When supported, both service-provider-initiated and identity-provider-initiated single logout should be supported. Service providers should support either the front-channel or back-channel single logout mechanisms depending on their respective capabilities although nothing prevents them from supporting both mechanisms if desired.
Identity Provider Introduction	Service providers that want to support networks with more than a single circle of trust simultaneously should support the Identity Provider Introduction Protocol.

126

2.3   LECP Implementation Guidelines

128

Liberty Feature	Recommendations
Single Sign-On	Support for LECP single sign-on profile.

129

3   Liberty Architecture Specifications Checklist

3.1   Liberty Profiles and Bindings Requirements — Identity Provider

132

Req ID#	Description	Ref	Y/N
IDP-FED-1	Identity Federation	Section 3.2.1 [LibertyBindProf]	
IDP-SSO-1	Single Sign-On using Browser Artifact	Section 3.2.2 [LibertyBindProf]	
IDP-SSO-2	Single Sign-On using Browser POST	Section 3.2.3 [LibertyBindProf]	
IDP-SSO-3	Single Sign-On using WML POST	Section 3.2.4 [LibertyBindProf]	
IDP-SSO-4	Single Sign-On using LECP	Section 3.2.5 [LibertyBindProf]	
IDP-REG-1	Register Name Identifier — Front Channel	Section 3.3 [LibertyBindProf]	
IDP-REG-2	Register Name Identifier — Back Channel	Section 3.3 [LibertyBindProf]	
IDP-REG-3	Register Name Identifier (Identity Provider initiated) — Front Channel	Section 3.3.1.1 [LibertyBindProf]	
IDP-REG-4	Register Name Identifier (Identity Provider initiated) — Back Channel	Section 3.3.1.2 [LibertyBindProf]	
IDP-REG-5	Register Name Identifier (Service Provider initiated) — Front Channel	Section 3.3.2.1 [LibertyBindProf]	
IDP-REG-6	Register Name Identifier (Service Provider initiated) — Back Channel	Section 3.3.2.2 [LibertyBindProf]	
IDP-FED-2	Identity Federation Termination — Front Channel	Section 3.4 [LibertyBindProf]	
IDP-FED-3	Identity Federation Termination — Back Channel	Section 3.4 [LibertyBindProf]	
IDP-FED-4	Federation Termination Notification (Identity Provider Initiated) — Front Channel	Section 3.4.1.1 [LibertyBindProf]	
IDP-FED-5	Federation Termination Notification (Identity Provider Initiated) — Back Channel	Section 3.4.1.2 [LibertyBindProf]	
IDP-FED-6	Federation Termination Notification (Service Provider Initiated) — Front Channel	Section 3.4.2.1 [LibertyBindProf]	
IDP-FED-7	Federation Termination Notification (Service Provider Initiated) — Back Channel	Section 3.4.2.2 [LibertyBindProf]	
IDP-SLO-1	Single Logout	Section 3.5 [LibertyBindProf]	
IDP-SLO-2	Single Logout Initiated by Identity Provider: Redirect	Section 3.5.1.1 [LibertyBindProf]	

Req ID#	Description	Ref	Y/N
IDP-SLO-3	Single Logout Initiated by Identity Provider: SOAP	Section 3.5.1.2 [LibertyBindProf]	
IDP-SLO-4	Single Logout Initiated by Service Provider: Redirect	Section 3.5.2.1 [LibertyBindProf]	
IDP-SLO-5	Single Logout Initiated by Service Provider: SOAP	Section 3.5.2.2 [LibertyBindProf]	
IDP-INT-1	Identity Provider Introduction	Section 3.6 [LibertyBindProf]	
IDP-COM-1	HTTP Connection over SSL3.0 or TLS1.0 [RFC2246], WTLS	[SSLv3], [RFC2246], [WTLS]	
IDP-COM-2	Support for Minimum URL length of 256 bytes	[RFC2965]	
IDP-COM-3	Support for Session Cookies	[RFC2965]	

133

3.2   Liberty Profiles and Bindings Requirements — Service Provider

135

Req ID#	Description	Ref	Y/N
SP-FED-1	Identity Federation	Section 3.2.1 [LibertyBindProf]	
SP-SSO-1	Single Sign-On using Browser Artifact	Section 3.2.2 [LibertyBindProf]	
SP-SSO-2	Single Sign-On using Browser POST	Section 3.2.3 [LibertyBindProf]	
SP-SSO-3	Single Sign-On using WML	Section 3.2.4 [LibertyBindProf]	
SP-SSO-4	Single Sign-On using LECP	Section 3.2.5 [LibertyBindProf]	
SP-REG-1	Register Name Identifier — Front Channel	Section 3.3 [LibertyBindProf]	
SP-REG-2	Register Name Identifier — Back Channel	Section 3.3 [LibertyBindProf]	
SP-REG-3	Register Name Identifier (Identity Provider initiated) — Front Channel	Section 3.3.1.1 [LibertyBindProf]	
SP-REG-4	Register Name Identifier (Identity Provider initiated) — Back Channel	Section 3.3.1.2 [LibertyBindProf]	
SP-REG-5	Register Name Identifier (Service Provider initiated) — Front Channel	Section 3.3.2.1 [LibertyBindProf]	
SP-REG-6	Register Name Identifier (Service Provider initiated) — Back Channel	Section 3.3.2.2 [LibertyBindProf]	
SP-FED-2	Identity Federation Termination — Front Channel	Section 3.4 [LibertyBindProf]	
SP-FED-3	Identity Federation Termination — Back Channel	Section 3.4 [LibertyBindProf]	
SP-FED-4	Federation Termination Notification (Identity Provider Initiated) — Front Channel	Section 3.4.1.1 [LibertyBindProf]	
SP-FED-5	Federation Termination Notification (Identity Provider Initiated) — Back Channel	Section 3.4.1.2 [LibertyBindProf]	

Req ID#	Description	Ref	Y/N
SP-FED-6	Federation Termination Notification (Service Provider Initiated) — Front Channel	Section 3.4.2.1 [LibertyBindProf]	
SP-FED-7	Federation Termination Notification (Service Provider Initiated) — Back Channel	Section 3.4.2.2 [LibertyBindProf]	
SP-SLO-1	Single Logout	Section 3.5 [LibertyBindProf]	
SP-SLO-2	Single Logout Initiated by Identity Provider: Redirect	Section 3.5.1.1 [LibertyBindProf]	
SP-SLO-3	Single Logout Initiated by Identity Provider: SOAP	Section 3.5.1.2 [LibertyBindProf]	
SP-SLO-4	Single Logout Initiated by Service Provider: Redirect	Section 3.5.2.1 [LibertyBindProf]	
SP-SLO-5	Single Logout Initiated by Service Provider: SOAP	Section 3.5.2.2 [LibertyBindProf]	
SP-INT-1	Identity Provider Introduction	Section 3.6 [LibertyBindProf]	
SP-COM-1	HTTP Connection over SSL3.0 or TLS1.0 [RFC2246], WTLS	[SSLv3], [RFC2246], [WTLS]	
SP-COM-2	Support for Minimum URL Length of 256 bytes	[RFC2965]	
SP-COM-3	Support for Session Cookies	[RFC2965]	

136

3.3   Liberty Profiles and Bindings Requirements — LECP

138

Req ID#	Description	Ref	Y/N
LECP-SSO-1	Single Sign-On using LECP	Section 3.2.5 [LibertyBindProf]	
LECP-COM-1	Support for Minimum URL Length of 256 bytes	[RFC2965]	
LECP-COM-2	Support for Session Cookies	[RFC2965]	

139

3.4   Authentication Context Requirements — Identity Provider

141

Req ID#	Description	Ref	Y/N
IDP-AUTHN-01	MobileContract	Section 5.1.1 [LibertyAuthnContext]	
IDP-AUTHN-02	MobileDigitalID	Section 5.1.2 [LibertyAuthnContext]	
IDP-AUTHN-03	MobileUnregistered	Section 5.1.3 [LibertyAuthnContext]	
IDP-AUTHN-04	Password	Section 5.1.4 [LibertyAuthnContext]	
IDP-AUTHN-05	Password-ProtectedTransport	Section 5.1.5 [LibertyAuthnContext]	
IDP-AUTHN-06	Previous-Session	Section 5.1.6	

Req ID#	Description	Ref	Y/N
		[LibertyAuthnContext]	
IDP-AUTHN-07	Smartcard	Section 5.1.7 [LibertyAuthnContext]	
IDP-AUTHN-08	Smartcard-PKI	Section 5.1.8 [LibertyAuthnContext]	
IDP-AUTHN-09	Software-PKI	Section 5.1.9 [LibertyAuthnContext]	
IDP-AUTHN-10	Time-Sync-Token	Section 5.1.10 [LibertyAuthnContext]	

142

3.5 Authentication Context Requirements — Service Provider

144

Req ID#	Description	Ref	Y/N
SP-AUTHN-01	MobileContract	Section 5.1.1 [LibertyAuthnContext]	
SP-AUTHN-02	MobileDigitalID	Section 5.1.2 [LibertyAuthnContext]	
SP-AUTHN-03	MobileUnregistered	Section 5.1.3 [LibertyAuthnContext]	
SP-AUTHN-04	Password	Section 5.1.4 [LibertyAuthnContext]	
SP-AUTHN-05	Password-ProtectedTransport	Section 5.1.5 [LibertyAuthnContext]	
SP-AUTHN-06	Previous-Session	Section 5.1.6 [LibertyAuthnContext]	
SP-AUTHN-07	Smartcard	Section 5.1.7 [LibertyAuthnContext]	
SP-AUTHN-08	Smartcard-PKI	Section 5.1.8 [LibertyAuthnContext]	
SP-AUTHN-09	Software-PKI	Section 5.1.9 [LibertyAuthnContext]	
SP-AUTHN-10	Time-Sync-Token	Section 5.1.10 [LibertyAuthnContext]	

145

3.6 Authentication Context Requirements — LECP

147

Req ID#	Description	Ref	Y/N
LECP-AUTHN-01	MobileContract	Section 5.1.1 [LibertyAuthnContext]	
LECP-AUTHN-02	MobileDigitalID	Section 5.1.2 [LibertyAuthnContext]	
LECP-AUTHN-03	MobileUnregistered	Section 5.1.3 [LibertyAuthnContext]	
LECP-AUTHN-04	Password	Section 5.1.4	

Req ID#	Description	Ref	Y/N
		[LibertyAuthnContext]	
LECP-AUTHN-05	Password-ProtectedTransport	Section 5.1.5 [LibertyAuthnContext]	
LECP-AUTHN-06	Previous-Session	Section 5.1.6 [LibertyAuthnContext]	
LECP-AUTHN-07	Smartcard	Section 5.1.7 [LibertyAuthnContext]	
LECP-AUTHN-08	Smartcard-PKI	Section 5.1.8 [LibertyAuthnContext]	
LECP-AUTHN-09	Software-PKI	Section 5.1.9 [LibertyAuthnContext]	
LECP-AUTHN-10	Time-Sync-Token	Section 5.1.10 [LibertyAuthnContext]	

148

4 References

150 [LibertyArchOverview]   J. Hodges, "Liberty Architecture Overview.",
151   http://projectliberty.org/specs/liberty-architecture-overview-v1.0.pdf,
152   October 2002

153 [LibertyAuthnContext]   P. Madsen, "Liberty Authentication Context Specification.",
154   http://projectliberty.org/specs/liberty-architecture-authentication-
155   context-v1.0.pdf, October 2002

156 [LibertyBindProf]   J. Rouault, "Liberty Bindings and Profiles Specification.",
157   http://projectliberty.org/specs/liberty-architecture-bindings-profiles-
158   v1.0.pdf, October 2002

159 [LibertyGloss]   H. Mauldin, "Liberty Glossary", http://projectliberty.org/specs/liberty-
160   tech-glossary-v1.0.pdf, October 2002.

161 [LibertyProtSchema]   J. Beatty, "Liberty Protocols and Schemas Specification,"
162   http://projectliberty.org/specs/liberty-architecture-protocols-schemas-
163   v1.0.pdf, October 2002.

164 [RFC2246]   T. Dierks, C. Allen, "The TLS Protocol Version 1.0,"
165   **http://www.ietf.org/rfc/rfc2246.txt**, January 1999.

166 [RFC2965]   D. Cristol, L. Montulli, "HTTP State Management Mechanism,"
167   **http://www.ietf.org/rfc/rfc2965.txt**, October 2000.

168 [SAMLBind]   Mishra, Prateek, et al., "Bindings and Profiles for the OASIS Security
169   Assertion Markup Language (SAML)," **http://www.oasis-**
170   **open.org/committees/security/docs/draft-sstc-bindings-model-**
171   **10.doc**, OASIS, 10 January 2002.

172 [SAMLCore]   Hallam-Baker, P., et al., "Assertions and Protocol for the OASIS
173   Security Assertion Markup Language (SAML)," **http://www.oasis-**
174   **open.org/committees/security/docs/draft-sstc-core-21.pdf**, OASIS,
175   December 2001.

176 [SOAP1.1]   D. Box et al., "Simple Object Access Protocol (SOAP) 1.1,"
177   **http://www.w3.org/TR/SOAP**, World Wide Web Consortium Note,
178   May 2000.

179   [SSLv3]   A. Frier et al, "The SSL Protocol Version 3.0,"
180   **http://www.mozilla.org/projects/security/pki/nss/ssl/draft302.txt**,
181   18 November 1996.

182   [WML1.3 **]**   **"**Wireless Application Protocol Wireless Markup Language
183   Specification Version 1.3," Wireless Application Protocol Forum, Ltd.,
184   **http://www.wapforum.org/,** 19 February 2000.

185   [WTLS]   "Wireless Transport Layer Security,"
186   **http://www1.wapforum.org/tech/documents/WAP-261_102-**
187   **WTLS-20011027-a.pdf.**

188   [XMLSig]   D. Eastlake et al., "XML-Signature Syntax and Processing,"
189   **http://www.w3.org/TR/xmldsig-core/**, World Wide Web Consortium.