



Identity Provider Discovery Service Protocol and Profile

Committee Specification 01 27 March 2008

Specification URIs:

This Version:

<http://docs.oasis-open.org/security/saml/Post2.0/sstc-saml-idp-discovery-cs-01.html>

<http://docs.oasis-open.org/security/saml/Post2.0/sstc-saml-idp-discovery-cs-01.odt>

<http://docs.oasis-open.org/security/saml/Post2.0/sstc-saml-idp-discovery-cs-01.pdf>

Previous Version:

<http://docs.oasis-open.org/security/saml/Post2.0/sstc-saml-idp-discovery-cd-02.html>

<http://docs.oasis-open.org/security/saml/Post2.0/sstc-saml-idp-discovery-cd-02.odt>

<http://docs.oasis-open.org/security/saml/Post2.0/sstc-saml-idp-discovery-cd-02.pdf>

Latest Version:

<http://docs.oasis-open.org/security/saml/Post2.0/sstc-saml-idp-discovery.html>

<http://docs.oasis-open.org/security/saml/Post2.0/sstc-saml-idp-discovery.odt>

<http://docs.oasis-open.org/security/saml/Post2.0/sstc-saml-idp-discovery.pdf>

Technical Committee:

[OASIS Security Services TC](#)

Chair(s):

Hal Lockhart, BEA Systems, Inc.

Brian Campbell, Ping Identity Corporation

Editor(s):

Rod Widdowson, Edinburgh University

Scott Cantor, Internet2

Related Work:

This specification is an alternative to the SAML V2.0 Identity Provider Discovery profile in the SAML V2.0 Profiles specification [SAML2Prof].

Declared XML Namespace(s):

`urn:oasis:names:tc:SAML:profiles:SSO:idp-discovery-protocol`

31 **Abstract:**
32 Defines a generic browser-based protocol by which a centralized discovery service implemented
33 independently of a given service provider can provide a requesting service provider with the
34 unique identifier of an identity provider that can authenticate a principal.

35 **Status:**
36 This document was last revised or approved by the SSTC on the above date. The level of
37 approval is also listed above. Check the current location noted above for possible later revisions
38 of this document. This document is updated periodically on no particular schedule.

39 TC members should send comments on this specification to the TC's email list.
40 Others should send comments to the TC by using the "Send A Comment" button on
41 the TC's web page at <http://www.oasis-open.org/committees/security>.

42 For information on whether any patents have been disclosed that may be essential to
43 implementing this specification, and any offers of patent licensing terms, please refer to the IPR
44 section of the TC web page (<http://www.oasis-open.org/committees/security/ipr.php>).

45 The non-normative errata page for this specification is located at [http://www.oasis-](http://www.oasis-open.org/committees/security)
46 [open.org/committees/security](http://www.oasis-open.org/committees/security).

Notices

47

48 Copyright © OASIS Open 2007. All Rights Reserved.

49 All capitalized terms in the following text have the meanings assigned to them in the OASIS Intellectual
50 Property Rights Policy (the "OASIS IPR Policy"). The full Policy may be found at the OASIS website.

51 This document and translations of it may be copied and furnished to others, and derivative works that
52 comment on or otherwise explain it or assist in its implementation may be prepared, copied, published,
53 and distributed, in whole or in part, without restriction of any kind, provided that the above copyright
54 notice and this section are included on all such copies and derivative works. However, this document
55 itself may not be modified in any way, including by removing the copyright notice or references to OASIS,
56 except as needed for the purpose of developing any document or deliverable produced by an OASIS
57 Technical Committee (in which case the rules applicable to copyrights, as set forth in the OASIS IPR
58 Policy, must be followed) or as required to translate it into languages other than English.

59 The limited permissions granted above are perpetual and will not be revoked by OASIS or its successors
60 or assigns.

61 This document and the information contained herein is provided on an "AS IS" basis and OASIS
62 DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY
63 WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY
64 OWNERSHIP RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A
65 PARTICULAR PURPOSE.

66 OASIS requests that any OASIS Party or any other party that believes it has patent claims that would
67 necessarily be infringed by implementations of this OASIS Committee Specification or OASIS Standard,
68 to notify OASIS TC Administrator and provide an indication of its willingness to grant patent licenses to
69 such patent claims in a manner consistent with the IPR Mode of the OASIS Technical Committee that
70 produced this specification.

71 OASIS invites any party to contact the OASIS TC Administrator if it is aware of a claim of ownership of
72 any patent claims that would necessarily be infringed by implementations of this specification by a patent
73 holder that is not willing to provide a license to such patent claims in a manner consistent with the IPR
74 Mode of the OASIS Technical Committee that produced this specification. OASIS may include such
75 claims on its website, but disclaims any obligation to do so.

76 OASIS takes no position regarding the validity or scope of any intellectual property or other rights that
77 might be claimed to pertain to the implementation or use of the technology described in this document or
78 the extent to which any license under such rights might or might not be available; neither does it
79 represent that it has made any effort to identify any such rights. Information on OASIS' procedures with
80 respect to rights in any document or deliverable produced by an OASIS Technical Committee can be
81 found on the OASIS website. Copies of claims of rights made available for publication and any
82 assurances of licenses to be made available, or the result of an attempt made to obtain a general license
83 or permission for the use of such proprietary rights by implementers or users of this OASIS Committee
84 Specification or OASIS Standard, can be obtained from the OASIS TC Administrator. OASIS makes no
85 representation that any information or list of intellectual property rights will at any time be complete, or
86 that any claims in such list are, in fact, Essential Claims.

87 The name "OASIS" is a trademark of [OASIS](http://www.oasis-open.org), the owner and developer of this specification, and should
88 be used only to refer to the organization and its official outputs. OASIS welcomes reference to, and
89 implementation and use of, specifications, while reserving the right to enforce its marks against
90 misleading uses. Please see <http://www.oasis-open.org/who/trademark.php> for above guidance.

91 **Table of Contents**

92 1 Introduction.....5
93 1.1 Terminology.....5
94 1.2 Normative References.....5
95 1.3 Non-Normative References.....6
96 1.4 Conformance.....6
97 1.4.1 Identity Provider Discovery Protocol Profile.....6
98 2 Identity Provider Discovery Protocol and Profile.....7
99 2.1 Required Information.....7
100 2.2 Background.....7
101 2.3 Discovery Policy.....8
102 2.4 Protocol Description.....8
103 2.4.1 HTTP Request to Discovery Service.....9
104 2.4.2 Discovery Service determines appropriate Identity Provider.....9
105 2.4.3 HTTP Redirect to Service Provider.....10
106 2.5 Use of Metadata.....10
107 Appendix A.Acknowledgments.....12
108 Appendix B.Revision History.....13

1 Introduction

This specification defines a browser-based protocol by which a centralized discovery service can provide a requesting service provider with the unique identifier of an identity provider that can authenticate a principal. Thus, the protocol provides an alternative means of addressing section 4.1.3.2 of [SAML2Prof]. The profile for discovery defined in section 4.3 of [SAML2Prof] is similar, but has different deployment properties, such as the requirement for a shared domain.

Instead, this profile relies on a normative, redirect-based wire protocol that allows for independent implementation and deployment of the service provider and discovery service components, a model that has proven useful in some large-scale deployments in which managing common domain membership may be impractical.

Note that most Web SSO protocols and profiles, including the multiple versions of SAML, share similar properties and requirements for identity provider discovery (although terminology often differs). This protocol, while suited to SAML V2.0 SSO requirements, is not specific to them.

1.1 Terminology

The keywords "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this specification are to be interpreted as described in [RFC 2119].

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

Listings of XML schemas appear like this.

Example code listings appear like this.

Conventional XML namespace prefixes are used throughout the listings in this specification to stand for their respective namespaces as follows, whether or not a namespace declaration is present in the example:

Prefix	XML Namespace	Comments
md:	urn:oasis:names:tc:SAML:2.0:metadata	This is the SAML V2.0 metadata namespace defined in the SAML V2.0 metadata specification Error: Reference source not found.
idpdisc:	urn:oasis:names:tc:SAML:profiles:SSO:idp-discovery-protocol	This is the SAML V2.0 metadata extension namespace defined by this document and its accompanying schema [IDPDisco-XSD].
xsd:	http://www.w3.org/2001/XMLSchema	This namespace is defined in the W3C XML Schema specification [Schema1]. In schema listings, this is the default namespace and no prefix is shown.

This specification uses the following typographical conventions in text: <SAMLElement>, <ns:ForeignElement>, Attribute, **Datatype**, OtherKeyword.

1.2 Normative References

- [RFC 2119] S. Bradner. *Key words for use in RFCs to Indicate Requirement Levels*. IETF RFC 2119, March 1997. <http://www.ietf.org/rfc/rfc2119.txt>.

139	[IDPDisco-XSD]	S. Cantor et al. Metadata Extension Schema for Identity Provider Discovery Service Protocol, OASIS SSTC January 2007. Document ID sstc-saml-idp-discovery.xsd. See http://www.oasis-open.org/committees/security/ .
140		
141		
142	[SAML2Core]	S. Cantor et al. <i>Assertions and Protocols for the OASIS Security Assertion Markup Language (SAML) V2.0</i> . OASIS Standard, March 2005. Document ID saml-core-2.0-os. See http://docs.oasis-open.org/security/saml/v2.0/saml-core-2.0-os.pdf .
143		
144		
145		
146	[SAML2Meta]	S. Cantor et al. <i>Metadata for the OASIS Security Assertion Markup Language (SAML) V2.0</i> . OASIS Standard, March 2005. Document ID saml-metadata-2.0-os. See http://docs.oasis-open.org/security/saml/v2.0/saml-metadata-2.0-os.pdf .
147		
148		
149	[SAML2Meta-xsd]	S. Cantor et al. SAML V2.0 metadata schema. OASIS Standard, March 2005. Document ID saml-schema-metadata-2.0. See http://docs.oasis-open.org/security/saml/v2.0/saml-schema-metadata-2.0.xsd .
150		
151		
152	[SAML2Prof]	S. Cantor et al. <i>Profiles for the OASIS Security Assertion Markup Language (SAML) V2.0</i> . OASIS Standard, March 2005. Document ID saml-profiles-2.0-os. See http://docs.oasis-open.org/security/saml/v2.0/saml-profiles-2.0-os.pdf .
153		
154		
155	[Schema1]	H. S. Thompson et al. <i>XML Schema Part 1: Structures</i> . World Wide Web Consortium Recommendation, May 2001. See http://www.w3.org/TR/2001/REC-xmlschema-1-20010502/ .
156		
157		

158 1.3 Non-Normative References

159	[ShibProt]	S. Cantor et al. <i>Shibboleth Architecture: Protocols and Profiles</i> . Internet2-MACE, September 2005. Document ID internet2-mace-shibboleth-arch-protocols. http://shibboleth.internet2.edu/shibboleth-documents.html .
160		
161		

162 2.4 Conformance

163 2.4.1 Identity Provider Discovery Protocol Profile

164 An implementation of this profile shall be a conforming Service Provider or a conforming Discovery
165 Service (or both):

- 166 1. A conforming Service Provider MUST conform to the normative statements in section 2 that
167 pertain to Service Provider behavior.
- 168 2. A conforming Discovery Service MUST conform to the normative statements in section 2 that
169 pertain to Discovery Service behavior.

170 All conforming Implementations MUST support, at minimim, a discovery service policy value of
171 "urn:oasis:names:tc:SAML:profiles:SSO:idp-discovery-protocol:single", which is the default value.

2 Identity Provider Discovery Protocol and Profile

2.1 Required Information

Identification: urn:oasis:names:tc:SAML:profiles:SSO:idp-discovery-protocol

Contact information: security-services-comment@lists.oasis-open.org

Description: Given below.

Updates: Provides an alternative to the cookie-based discovery profile in section 4.3 of [SAML2Prof].

2.2 Background

Approaches to web single sign-on (SSO) typically fall into the broad categories of "active" and "passive" profiles, based on the capabilities of the client. So-called passive profiles rely on the features common to web browsers without extensions or plugins that would require additional downloads or operating system support, while active profiles require more advanced (and today at least, generally undeployed) capabilities. The SAML standard includes both kinds of profiles.

One problem that distinguishes federated deployment of passive SSO profiles is identity provider discovery. Passive profiles rely on the service provider, often termed a relying party, to relay (using GET or POST) the user agent to the identity provider. Leaving aside some of the security considerations that this introduces, a fundamental problem exists: how does the service provider know where to send the user agent?

There are a wide range of "solutions" to this problem, but they all share the trait of functioning well only in the presence of certain assumptions about the nature of the deployment and the expectations of users. For example, the most straightforward approach is for each service provider to simply ask the user (and possibly cache the result locally). This allows for maximum control over the experience by the service provider, as well as supplying an unambiguous result. It also leads to increased interference with the SSO experience due to per-site prompts, as well as extra work for relying parties and the need for users to understand how to make an unambiguous selection.

At the other extreme, there has been a model promulgated around the idea of discovery as a function of large federations of identity providers, as in the older Shibboleth "WAYF" model [ShibProt], in which the discovery service acts as a proxy for the service provider and relays a request to the selected identity provider. In theory, this seems attractive since every service provider can share a single point of discovery, and the user experience can be seamless across many/all services. In practice, this model falls apart quickly because of course there is no single point of discovery that accomodates the entire world of federation.

The cookie-based discovery profile in section 4.3 of [SAML2Prof] falls somewhere between these extremes by focusing on deployments in which all the parties can share a presence in a common domain. DNS itself does not constrain the size of such deployments but practical limitations on domain management tends to inhibit truly large-scale use. It also requires a great deal of static pre-configuration, which limits run-time flexibility.

The protocol and profile outlined here is intended as a hybrid of earlier approaches that brings additional benefits, including:

- Independent implementation of the service provider, identity provider, and discovery service components.
- Flexibility with regard to how the discovery process integrates with services.

- 213 ● Meta-behavior that enables the protocol to "emulate" other, similar proposals observed in
- 214 existing SAML and non-SAML software.
- 215 ● Better handling of multi-protocol deployments than the older Shibboleth WAYF model.
- 216 ● Accomodation for passive SSO (in the SAML 2.0 *IsPassive* sense).
- 217 ● Less static pre-configuration than a shared domain solution.

218 All that said, it is not intended as a panacea, but simply an alternative to fill another deployment niche.

219 2.3 Discovery Policy

220 To provide for future extensibility, multiple "flavors" of this discovery profile can be defined and selected
 221 using a URI-valued *policy* query string parameter.

222 Currently, only a policy of "urn:oasis:names:tc:SAML:profiles:SSO:idp-discovery-protocol:single" is
 223 defined and acts as the default value. Additional policies MAY specify alternate or additional behavior to
 224 that defined in section 2.4 as long as an alternate policy value is supplied.

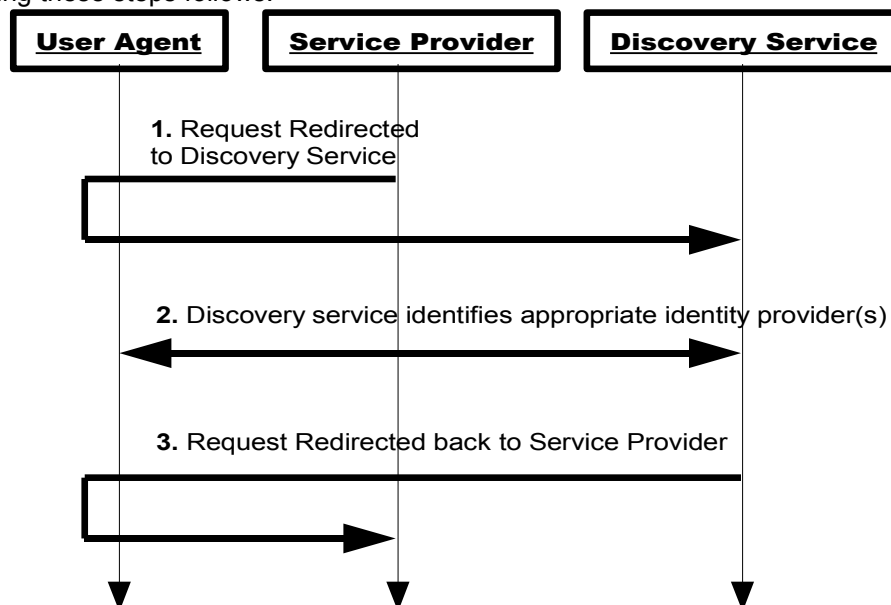
225 2.4 Protocol Description

226 This protocol can be used during web-based SSO when a service provider needs to establish an identity
 227 provider associated with a principal. It is assumed that the user wields a standard HTTP user agent.

228 The discovery protocol encompasses three steps, including two normative message exchanges:

- 229 1. The service provider redirects the user agent to the discovery service with a set of parameters
- 230 that make up the request.
- 231 2. The discovery service interacts with the principal via the user agent to establish one or more
- 232 suitable identity providers.
- 233 3. The discovery service redirects the user agent back to the service provider with the selected
- 234 identity provider(s) or an empty response.

235 A diagram showing these steps follows:



236 2.4.1 HTTP Request to Discovery Service

237 In the first step, a requesting service provider redirects the user agent to the discovery service with an
238 HTTP GET request.

239 The following parameter MUST be present on the query string (and URL-encoded):

240 `entityID`

241 The unique identifier of the service provider the end user is (or will be) interacting with, following
242 successful authentication by an identity provider.

243 The following parameters MAY be present:

244 `return`

245 A URL, which MAY itself include a query string. However, such a query string MUST NOT
246 contain a parameter with the same name as the value of the `returnIDParam` parameter in the
247 request (see below) or the name "entityID" if no `returnIDParam` parameter is supplied. (This
248 guards against the possibility of a multiply-valued query string parameter in the response.)

249 The discovery service MUST redirect the user agent to this location in response to this request
250 (see section 2.4.3). If metadata is used (as in section 2.5), then this parameter MAY be omitted;
251 the return location MUST then be based on the default `<idpdisc:DiscoveryResponse>`
252 element. Otherwise, if metadata is not used, then this parameter becomes mandatory and MUST
253 be present.

254 `policy`

255 A parameter name used to indicate the desired behavior controlling the processing of the
256 discovery service. If omitted, it defaults to a value of "urn:oasis:names:tc:SAML:profiles:SSO:idp-
257 discovery-protocol:single".

258 `returnIDParam`

259 A parameter name used to return the unique identifier of the selected identity provider to the
260 original requester. If this parameter is omitted, it defaults to a value of "entityID". This parameter
261 can be used to customize the response to the service provider so that software relying on
262 alternate approaches to discovery can be utilized in conjunction with this protocol.

263 `isPassive`

264 A boolean value of "true" or "false" that controls whether the discovery service is allowed to
265 visibly interact with the user agent in the second step below. If a value is not provided, the
266 default is "false".

267 2.4.2 Discovery Service determines appropriate Identity Provider

268 In this step, the discovery service and user agent interact via unspecified means in order to establish the
269 user's choice of identity provider. This may involve user selection, hints obtained through various means,
270 and filtering based on the service provider (identified by the `entityID` parameter in the first step
271 above), preferred SSO protocols or profiles, etc.

272 If the `isPassive` parameter is set to "true", the discovery service MUST NOT visibly take control of the
273 user interface from the requesting service provider and interact with the user agent in a noticeable
274 fashion. Additional redirection is permitted, however, provided the passive guarantee can be met.

275 The discovery service MAY rely on saved state, such as HTTP cookies, to determine the appropriate
276 identity provider. If a single cookie is used, it SHOULD conform to the name and format specified by the
277 Identity Provider Discovery Profile in section 4.3 of [SAML2Prof].

278 2.4.3 HTTP Redirect to Service Provider

279 If the `policy` parameter is omitted or set to "urn:oasis:names:tc:SAML:profiles:SSO:idp-discovery-
280 protocol:single", then the single selection policy in effect designates that the discovery service is to
281 respond to the service provider and either return a single selected identity provider or none at all using
282 the processing rules defined in this section. Other `policy` values may define alternate behavior to that
283 defined here.

284 If an identity provider was determined and other requirements (such as metadata) are satisfied, the
285 discovery service MUST respond by redirecting the user agent back to the requesting service provider
286 with an HTTP GET request, at the location supplied in the `return` parameter in the original request (or
287 to the default location identified in metadata if no such parameter was supplied). The unique identifier of
288 the selected identity provider MUST be included as the value of the query string parameter whose name
289 was specified as the value of the `returnIDParam` parameter in the original request (or `entityID` if no
290 parameter was supplied).

291 If instead an identity provider was not determined, or the discovery service cannot or will not answer,
292 then the discovery service MAY halt processing by displaying an error to the user agent or MAY redirect
293 the user agent back to the requesting service provider. If the service provider included the `isPassive`
294 parameter in its original request, then the discovery service has no option and MUST redirect the user
295 agent back to the service provider. If it responds, then it MUST NOT include the query string parameter
296 whose name was specified as the value of the `returnIDParam` parameter in the original request (or
297 `entityID` if no parameter was supplied). The absence of this parameter is the indication of failure to
298 return a selection.

299 Note that the discovery service MUST take care to preserve any query string that may already be
300 present within the `return` URL.

301 2.5 Use of Metadata

302 All redirection-based SSO protocols share a common property in that the service provider is permitted to
303 (and in most cases must) redirect the user agent to the identity provider. This creates opportunities for
304 phishing attacks against the user's authentication credentials when weak (but extremely common) forms
305 of authentication such as passwords are used.

306 This protocol has the potential for creating additional opportunities for phishing if arbitrary web sites are
307 permitted to utilize the protocol and obtain the user's identity provider, the key piece of knowledge
308 required to fake the expected authentication experience. To mitigate this threat, metadata can be used to
309 limit the sites authorized to use a discovery service, without introducing more complex (though stronger)
310 approaches such as message authentication.

311 A discovery service SHOULD require that the service providers making use of it supply metadata (out of
312 band or using techniques such as those described in the SAML V2.0 Metadata specification
313 [SAML2Meta]).

314 An extension element, `<idpdisc:DiscoveryResponse>`, of type `md:IndexedEndpointType`, is used
315 to define the acceptable locations to which the discovery service should respond with the user's identity
316 provider. The `Binding` attribute of the extension element MUST be set to:

317 `urn:oasis:names:tc:SAML:profiles:SSO:idp-discovery-protocol`

318 Upon receiving a request, the discovery service SHOULD ensure that it recognizes the requesting
319 service provider, as identified by the `entityID` parameter in the request. The location supplied in the
320 `return` parameter (if any) SHOULD then be compared to the `Location` attribute of any
321 `<idpdisc:DiscoveryResponse>` elements found in the `<md:Extensions>` element of the service
322 provider's `<md:SPSSODescriptor>` element. (Note that the `ResponseLocation` endpoint attribute is

323 unused in this profile.) When metadata is used, the requesting service provider MAY also omit the
324 return parameter in its request in favor of the default endpoint supplied in its metadata.

325 In the case that the return parameter includes a query string, the discovery service MUST ignore it for
326 the purposes of this comparison.

327 The schema for the <idpdisc:DiscoveryResponse> element is as follows:

```
328 <schema
329     targetNamespace="urn:oasis:names:tc:SAML:profiles:SSO:idp-discovery-
330 protocol"
331     xmlns:idpdisc="urn:oasis:names:tc:SAML:profiles:SSO:idp-discovery-
332 protocol"
333     xmlns:md="urn:oasis:names:tc:SAML:2.0:metadata"
334     xmlns="http://www.w3.org/2001/XMLSchema"
335     elementFormDefault="unqualified"
336     attributeFormDefault="unqualified"
337     blockDefault="substitution"
338     version="1.0">
339     <annotation>
340         <documentation>
341             Document identifier: sstc-saml-idp-discovery
342             Location: http://www.oasis-open.org/committees/documents.php?
343 wg_abbrev=security
344             Revision history:
345             V1.0 (January 2007):
346             Initial version.
347         </documentation>
348     </annotation>
349     <import namespace="urn:oasis:names:tc:SAML:2.0:metadata"
350         schemaLocation="saml-schema-metadata-2.0.xsd"/>
351     <element name="DiscoveryResponse" type="md:IndexedEndpointType"/>
352 </schema>
```

Appendix A. Acknowledgments

354 The editors would like to acknowledge the contributions of the OASIS Security Services Technical
355 Committee, whose voting members at the time of publication were:

- 356 • George Fletcher, AOL
- 357 • Hal Lockhart, BEA Systems, Inc.
- 358 • Steve Anderson, BMC Software
- 359 • Jeff Bohren, BMC Software
- 360 • Rob Philpott, EMC Corporation
- 361 • Carolina Canales-Valenzuela, Ericsson
- 362 • Lakshmi Thiyagarajan, Hewlett-Packard
- 363 • Anthony Nadalin, IBM
- 364 • Scott Cantor, Internet2
- 365 • Bob Morgan, Internet2
- 366 • Eric Tiffany, Liberty Alliance Project
- 367 • Tom Scavo, National Center for Supercomputing Applications (NCSA)
- 368 • Peter Davis, Neustar, Inc.
- 369 • Jeff Hodges, Neustar, Inc.
- 370 • Frederick Hirsch, Nokia Corporation
- 371 • Abbie Barbir, Nortel Networks Limited
- 372 • Paul Madsen, NTT Corporation
- 373 • Prateek Mishra, Oracle Corporation
- 374 • Brian Campbell, Ping Identity Corporation
- 375 • Anil Saldhana, Red Hat
- 376 • Eve Maler, Sun Microsystems
- 377 • Emily Xu, Sun Microsystems
- 378 • Kent Spaulding, Tripod Technology Group, Inc.
- 379 • David Staggs, Veterans Health Administration

380

Appendix B. Revision History

381

- Draft 01, initial draft based on document prepared by Rod for Shibboleth project

382

- Draft 02, default various parameters, add policy extension parameter, clarify some processing rules.

383

384

- Draft 03, add background material, clarify DS error handling and query string constraints, add mini-outline of protocol and diagram, switch to IndexedEndpointType for metadata element for easier defaulting.

385

386

387

- Committee Draft 01, boilerplate edits for CD status.

388

- Draft 04, add conformance section, clarify a metadata issue.

389

- Committee Draft 02, boilerplate edits for CD status