



Service Provider Request Initiation Protocol and Profile Version 1.0

Committee Specification 01 5 November 2010

Specification URIs:

This Version:

<http://docs.oasis-open.org/security/saml/Post2.0/ssstc-request-initiation-cs-01.html>

<http://docs.oasis-open.org/security/saml/Post2.0/ssstc-request-initiation-cs-01.odt> (Authoritative)

<http://docs.oasis-open.org/security/saml/Post2.0/ssstc-request-initiation-cs-01.pdf>

Previous Version:

<http://docs.oasis-open.org/security/saml/Post2.0/ssstc-request-initiation-cd-01.html>

<http://docs.oasis-open.org/security/saml/Post2.0/ssstc-request-initiation-cd-01.odt> (Authoritative)

<http://docs.oasis-open.org/security/saml/Post2.0/ssstc-request-initiation-cd-01.pdf>

Latest Version:

<http://docs.oasis-open.org/security/saml/Post2.0/ssstc-request-initiation.html>

<http://docs.oasis-open.org/security/saml/Post2.0/ssstc-request-initiation.odt> (Authoritative)

<http://docs.oasis-open.org/security/saml/Post2.0/ssstc-request-initiation.pdf>

Technical Committee:

[OASIS Security Services TC](#)

Chair:

Thomas Hardjono, M.I.T.

Nate Kingenstein, Internet2

Editor:

Scott Cantor, Internet2

Related Work:

This specification composes with the Identity Provider Discovery Service Protocol and Profile [IdPDisco], and with multiple standards for browser-based Single Sign-On, such as SAML 2.0 and WS-Federation [WS-Fed].

Declared XML Namespace(s):

`urn:oasis:names:tc:SAML:profiles:SSO:request-init`

Abstract:

Defines a generic browser-based protocol by which a request can be made to a service provider to initiate a protocol-specific request for authentication, and to ask that particular options be used when making such a request.

36 **Status**

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

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

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

46

47 Notices

48 Copyright © OASIS Open 2010. 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 notice
54 and this section are included on all such copies and derivative works. However, this document itself may
55 not be modified in any way, including by removing the copyright notice or references to OASIS, except as
56 needed for the purpose of developing any document or deliverable produced by an OASIS Technical
57 Committee (in which case the rules applicable to copyrights, as set forth in the OASIS IPR Policy, must be
58 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](#), the owner and developer of this specification, and should be
88 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 Notation..... 5
94 1.2 Normative References..... 6
95 1.3 Non-Normative References..... 6
96 2 Service Provider Request Initiation Protocol and Profile..... 7
97 2.1 Required Information..... 7
98 2.2 Protocol Description..... 7
99 2.3 HTTP Request Format..... 7
100 2.3.1 Defined Parameters..... 7
101 2.3.2 Extensions..... 8
102 2.4 Use of Metadata..... 8
103 2.5 Security Considerations..... 9
104 3 Conformance..... 10
105 3.1 Service Provider Request Initiation Profile Version 1.0..... 10
106 Appendix A. Acknowledgements..... 11
107 Appendix B. Revision History..... 12
108

109 1 Introduction

110 Modern standards for browser-based Single Sign-On (SSO) typically include the ability to initiate the
111 authentication process from either the identity provider (IdP) or service provider (SP) participating in the
112 exchange. However, the standards typically lack a defined mechanism for asking either end to actually
113 initiate the process, relying on proprietary interfaces, or on the user agent accessing a protected resource
114 at the service provider.

115 IdP-initiated SSO assumes a variety of information is known at the time of a request, including the identity
116 provider itself and its location, protocol features and binding/profile details to apply, how to express the
117 desired resource to access, etc. In general, it suffers by leaving the service provider "out of the loop" in
118 formulating the request and applying its own decision-making in doing so.

119 On the other hand, SP-initiated SSO suffers from a lack of standardization, particularly when support for
120 "deep-linking", or unauthenticated access to resources within a protected system, is lacking. Many
121 complex deployments are unable to fully support direct access in that fashion, and require special
122 conventions or work-arounds that are often propagated to links constructed outside of the affected site,
123 creating brittle links and maintenance challenges.

124 A standard protocol for invoking the SSO functionality available at a service provider in an abstracted,
125 protocol-neutral fashion solves both problems.

126 1.1 Notation

127 This specification uses normative text.

128 The keywords "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD
129 NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this specification are to be interpreted as
130 described in [RFC2119]:

131 ...they MUST only be used where it is actually required for interoperation or to limit behavior
132 which has potential for causing harm (e.g., limiting retransmissions)...

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

136 Listings of XML schemas appear like this.

137 Example code listings appear like this.

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

Prefix	XML Namespace	Comments
saml:	urn:oasis:names:tc:SAML:2.0:assertion	This is the SAML V2.0 assertion namespace [SAML2Core].
samlp:	urn:oasis:names:tc:SAML:2.0:protocol	This is the SAML V2.0 protocol namespace [SAML2Core].
md:	urn:oasis:names:tc:SAML:2.0:metadata	This is the SAML V2.0 metadata namespace .
init:	urn:oasis:names:tc:SAML:profiles:SSO:request-init	This is the SAML V2.0 metadata extension namespace defined by this document and its accompanying schema [ReqInit-XSD].

142 This specification uses the following typographical conventions in text: <SAML*Element*>,
143 <ns:ForeignElement>, Attribute, **Datatype**, OtherCode.

144 1.2 Normative References

- 145 **[ReqInit-XSD]** OASIS Committee Specification 01, *Metadata Extension Schema for Service*
146 *Provider Request Initiation Protocol and Profile Version 1.0*, November 2010.
147 <http://docs.oasis-open.org/security/saml/Post2.0/sstc-request-initiation.xsd>
- 148 **[RFC2119]** S. Bradner. *Key words for use in RFCs to Indicate Requirement Levels*. IETF
149 RFC 2119, March 1997. <http://www.ietf.org/rfc/rfc2119.txt>
- 150 **[RFC2616]** R. Fielding, et. al. *Hypertext Transfer Protocol 1.1*. IETF RFC 2616, June 1999.
151 <http://www.ietf.org/rfc/rfc2616.txt>
- 152 **[SAML2Bind]** OASIS Standard, *Bindings for the OASIS Security Assertion Markup Language*
153 *(SAML) V2.0*, March 2005. [http://docs.oasis-open.org/security/saml/v2.0/saml-](http://docs.oasis-open.org/security/saml/v2.0/saml-bindings-2.0-os.pdf)
154 [bindings-2.0-os.pdf](http://docs.oasis-open.org/security/saml/v2.0/saml-bindings-2.0-os.pdf)
- 155 **[SAML2Core]** OASIS Standard, *Assertions and Protocols for the OASIS Security Assertion*
156 *Markup Language (SAML) V2.0*, March 2005. [http://docs.oasis-](http://docs.oasis-open.org/security/saml/v2.0/saml-core-2.0-os.pdf)
157 [open.org/security/saml/v2.0/saml-core-2.0-os.pdf](http://docs.oasis-open.org/security/saml/v2.0/saml-core-2.0-os.pdf)
- 158 **[SAML2Errata]** OASIS Approved Errata, *SAML V2.0 Errata*, October 2009. [http://docs.oasis-](http://docs.oasis-open.org/security/saml/v2.0/sstc-saml-approved-errata-2.0.pdf)
159 [open.org/security/saml/v2.0/sstc-saml-approved-errata-2.0.pdf](http://docs.oasis-open.org/security/saml/v2.0/sstc-saml-approved-errata-2.0.pdf)
- 160 **[SAML2Meta]** OASIS Standard, *Metadata for the OASIS Security Assertion Markup Language*
161 *(SAML) V2.0*, March 2005. [http://docs.oasis-open.org/security/saml/v2.0/saml-](http://docs.oasis-open.org/security/saml/v2.0/saml-metadata-2.0-os.pdf)
162 [metadata-2.0-os.pdf](http://docs.oasis-open.org/security/saml/v2.0/saml-metadata-2.0-os.pdf)
- 163 **[SAML2Prof]** OASIS Standard, *Profiles for the OASIS Security Assertion Markup Language*
164 *(SAML) V2.0*, March 2005. [http://docs.oasis-open.org/security/saml/v2.0/saml-](http://docs.oasis-open.org/security/saml/v2.0/saml-profiles-2.0-os.pdf)
165 [profiles-2.0-os.pdf](http://docs.oasis-open.org/security/saml/v2.0/saml-profiles-2.0-os.pdf)

166 1.3 Non-Normative References

- 167 **[IdPDisco]** OASIS Committee Specification, *Identity Provider Discovery Service Protocol*
168 *and Profile*, March 2008. [http://docs.oasis-open.org/security/saml/Post2.0/sstc-](http://docs.oasis-open.org/security/saml/Post2.0/sstc-saml-idp-discovery.pdf)
169 [saml-idp-discovery.pdf](http://docs.oasis-open.org/security/saml/Post2.0/sstc-saml-idp-discovery.pdf)
- 170 **[WS-Fed]** OASIS Standard, *Web Services Federation Language V1.2*, May 2009.
171 [http://docs.oasis-open.org/wsfed/federation/v1.2/os/ws-federation-1.2-spec-](http://docs.oasis-open.org/wsfed/federation/v1.2/os/ws-federation-1.2-spec-os.pdf)
172 [os.pdf](http://docs.oasis-open.org/wsfed/federation/v1.2/os/ws-federation-1.2-spec-os.pdf)

173 2 Service Provider Request Initiation Protocol and 174 Profile

175 2.1 Required Information

176 **Identification:** urn:oasis:names:tc:SAML:profiles:SSO:request-init

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

178 **Description:** Given below.

179 **Updates:** None.

180 2.2 Protocol Description

181 This protocol is used to ask that a service provider supporting a federated authentication protocol produce
182 a request for authentication using particular options or assumptions. It is assumed that the user wields a
183 standard HTTP user agent. The protocol is specified between the user agent and the service provider.
184 Any technical means may be used to cause the user agent to submit a request using this protocol,
185 including static or dynamic links on any web site, client-side scripting, manual entry by a user, etc.

186 The request initiation protocol consists of a single HTTP [RFC2616] request/response, a normative
187 request followed by an arbitrary response from the service provider. This response MAY be a request for
188 authentication using a selected protocol (the format of which is left to the definition of that protocol), or it
189 MAY be a refusal to perform the requested action or any other response the service provider deems
190 appropriate.

191 In the event of failure, the response SHOULD inform the user as to the nature of the problem or prompt
192 for additional information as required. For example, in the event that the request does not identify the
193 identity provider to use, the response could be in the form of a request to a discovery service, per
194 [IdPDisco].

195 2.3 HTTP Request Format

196 The request to the service provider MUST use the GET method, and MAY contain one or more URL-
197 encoded query string parameters, as defined below. Parameter names are case-sensitive.

198 Implementations that do not recognize a parameter defined other than within this specification (i.e., an
199 extension defined privately or separately) MUST ignore that parameter.

200 2.3.1 Defined Parameters

201 `entityID`

202 The unique identifier of an identity provider the service provider is instructed to use. If it cannot or
203 will not do so, the service provider MUST NOT return a request for authentication to a different
204 identity provider (i.e., it MUST NOT ignore the choice). If this parameter is omitted, the service
205 provider is free to respond in any fashion it wishes, including but not limited to the use of any
206 supported discovery mechanism to determine the identity provider itself.

207 `target`

208 The location of a resource to which the user agent should be returned, when possible, following
209 successful authentication. If this parameter is omitted, the service provider MUST use a default
210 value (which it unilaterally determines).

211 isPassive
212 A boolean value of "true" or "false" that indicates whether the request generated by the service
213 provider should include an option to prevent visible user interaction with the identity provider. This
214 corresponds to the SAML 2.0 `IsPassive` attribute in a `<samlp:AuthnRequest>` message.

215 If this parameter is present and "true", and the authentication protocol supported by the service
216 provider and identity provider in common does not support this feature, then the service provider
217 MUST redirect the user agent to the value of the `target` parameter.

218 forceAuthn
219 A boolean value of "true" or "false" that indicates whether the request generated by the service
220 provider should include an option to bypass an existing security context and require explicit user
221 interaction during authentication to the identity provider. This corresponds to the SAML 2.0
222 `ForceAuthn` attribute in a `<samlp:AuthnRequest>` message.

223 If this parameter is present and "true", and the authentication protocol supported by the service
224 provider and identity provider in common does not support this feature, then the service provider
225 MUST NOT return a request for authentication.

226 2.3.2 Extensions

227 Parameters whose name begins with the case-sensitive string "ext_" are reserved for future use by this
228 or related specifications from this Technical Committee and MUST NOT be used for third-party extensions
229 of this protocol.

230 Parameters other than those specified above, or with the "ext_" prefix, MAY be present, but their
231 meaning is undefined by this specification.

232 The conventions for naming extensions are somewhat counter-intuitive but are necessary for compatibility
233 with existing implementations.

234 2.4 Use of Metadata

235 This protocol exists outside the purview of actual authentication protocols, but for documentation
236 purposes, or as an aid in the dynamic construction of links in support of this protocol, service providers
237 that are described using the SAML V2.0 Metadata specification MAY document endpoints supporting this
238 protocol using an extension element, `<init:RequestInitiator>`, of type **md:EndpointType**. The
239 `Binding` attribute of the extension element MUST be set to:

240 urn:oasis:names:tc:SAML:profiles:SSO:request-init

241 The schema for the `<init:RequestInitiator>` element is as follows:

```
242   <schema  
243       targetNamespace="urn:oasis:names:tc:SAML:profiles:SSO:request-init"  
244       xmlns:init="urn:oasis:names:tc:SAML:profiles:SSO:request-init"  
245       xmlns:md="urn:oasis:names:tc:SAML:2.0:metadata"  
246       xmlns="http://www.w3.org/2001/XMLSchema"  
247       elementFormDefault="unqualified"  
248       attributeFormDefault="unqualified"  
249       blockDefault="substitution"  
250       version="1.0">  
251       <annotation>  
252            <documentation>  
253                Document identifier: sstc-request-initiation
```

```
254         Location: http://www.oasis-open.org/committees/documents.php?
255 wg_abbrev=security
256         Revision history:
257         V1.0 (March 2010):
258         Initial version.
259     </documentation>
260 </annotation>
261 <import namespace="urn:oasis:names:tc:SAML:2.0:metadata"
262         schemaLocation="saml-schema-metadata-2.0.xsd"/>
263 <element name="RequestInitiator" type="md:EndpointType"/>
264 </schema>
```

265 **2.5 Security Considerations**

266 Some authentication protocols may involve the use of digital signatures or other cryptography, and thus
267 the generation of requests by a service provider may be computationally intensive. In such cases, support
268 for this protocol could provide a Denial of Service opportunity for an attacker, but not typically a new or
269 distinct one.

270 The ability to externally specify an identity provider could give an attacker the ability to derive information
271 about the sources of authentication trusted by a service provider based on its willingness or lack thereof
272 to respond with an authentication request or an error.

273 Exposing control over portions of the authentication request process to an outside agency could introduce
274 vulnerabilities if a service provider implementation is not careful in interpreting authentication responses
275 on their own merits rather than making assumptions about its requests. This is not dissimilar from the
276 requirements associated with handling IdP-initiated responses and should not generally create new
277 complications.

278 Finally, values of the `target` parameter should always be sanitized where used in the generation of
279 responses to user agents, to protect against cross-site scripting attacks and related problems.

280 **3 Conformance**

281 **3.1 Service Provider Request Initiation Profile Version 1.0**

282 A conforming Service Provider MUST conform to the normative statements in section 2 that pertain to
283 Service Provider behavior, and MUST properly interpret all the parameters defined in section 2.3.1 in the
284 manner prescribed in that section.

285 **Appendix A. Acknowledgements**

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

- 288 • Rob Philpott, EMC Corporation
- 289 • John Bradley, Individual
- 290 • Scott Cantor, Internet2
- 291 • Nate Klingenstein, Internet2
- 292 • Bob Morgan, Internet2
- 293 • Thomas Hardjono, M.I.T.
- 294 • Anthony Nadalin, Microsoft Corporation
- 295 • Tom Scavo, National Center for Supercomputing Applications (NCSA)
- 296 • Frederick Hirsch, Nokia Corporation
- 297 • Thinh Nguyenphu, Nokia Siemens Networks Gmb
- 298 • Paul Madsen, NTT Corporation
- 299 • Ari Kermaier, Oracle Corporation
- 300 • Hal Lockhart, Oracle Corporation
- 301 • Emily Xu, Oracle Corporation
- 302 • Anil Saldhana, Red Hat
- 303 • David Staggs, Veterans Health Administration

304 **Appendix B. Revision History**

- 305 ● Draft 01, first working draft based on Shibboleth implementation of the protocol.
- 306 ● Draft 02, clarify handling of unrecognized parameters.
- 307 ● Committee Draft 01, CD edits.