

---

# Kantara Initiative eGovernment Implementation Profile of SAML V2.0

## Version 2.0

**Working Draft 034**  
**MarchApril 198, 2010**

### Document identifier:

draft-kantara-egov-saml2-profile-2.0

### Location:

TBD

### Editors:

Scott Cantor, Internet2

### Contributors:

Kantara eGovernment WG

Andreas Åkre Solberg, UNINETT

### Abstract:

This document contains an implementation profile for eGovernment use of SAML V2.0, suitable for the purposes of testing conformance of implementations of SAML V2.0. It is not a deployment profile, and does not provide for or reflect specific behavior expected of implementations when used within a particular deployment context.

### Notice:

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

Implementation or use of certain elements of this document may require licenses under third party intellectual property rights, including without limitation, patent rights. The Participants of and any other contributors to the Specification are not and shall not be held responsible in any manner for identifying or failing to identify any or all such third party intellectual property rights. This Specification is provided "AS IS," and no Participant in Kantara Initiative makes any warranty of any kind, expressed or implied, including any implied warranties of merchantability, non-infringement of third party intellectual property rights, and fitness for a particular purpose. Implementers of this Specification are advised to review Kantara Initiative's website (<http://www.kantarainitiative.org/>) for information concerning any Necessary Claims Disclosure Notices that have been received by the Kantara Initiative Board of Trustees.

Copyright: The content of this document is copyright of Kantara Initiative. © 2010 Kantara Initiative.

# Table of Contents

41	1 Introduction.....	4
42	1.1 Notation.....	4
43	1.2 Normative References.....	5
44	2 SAML V2.0 Implementation Profile.....	7
45	2.1 Required Information.....	7
46	2.2 Metadata and Trust Management.....	7
47	2.2.1 Conformance Criteria.....	8
48	2.3 Name Identifiers.....	8
49	2.3.1 Conformance Criteria.....	8
50	2.4 Attributes.....	8
51	2.4.1 Conformance Criteria.....	8
52	2.5 Browser Single Sign-On.....	8
53	2.5.1 Identity Provider Discovery.....	9
54	2.5.1.1 Conformance Criteria.....	9
55	2.5.2 Authentication Requests.....	9
56	2.5.2.1 Binding and Security Requirements.....	9
57	2.5.2.1.1 Conformance Criteria.....	9
58	2.5.2.2 Message Content.....	9
59	2.5.2.2.1 Conformance Criteria.....	9
60	2.5.3 Responses.....	10
61	2.5.3.1 Binding and Security Requirements.....	10
62	2.5.3.1.1 Conformance Criteria.....	10
63	2.5.3.2 Message Content.....	10
64	2.5.3.2.1 Conformance Criteria.....	10
65	2.5.4 Artifact Resolution.....	11
66	2.5.4.1 Artifact Resolution Requests.....	11
67	2.5.4.1.1 Conformance Criteria.....	11
68	2.5.4.2 Artifact Resolution Responses.....	11
69	2.5.4.2.1 Conformance Criteria.....	11
70	2.6 Browser Holder of Key Single Sign-On.....	11
71	2.6.1 Conformance Criteria.....	11
72	2.7 SAML 2.0 Proxying.....	11
73	2.7.1 Authentication Requests.....	12
74	2.7.1.1 Conformance Criteria.....	12
75	2.7.2 Responses.....	12
76	2.7.2.1 Conformance Criteria.....	12
77	2.8 Single Logout.....	12
78	2.8.1 Logout Requests.....	12
79	2.8.1.1 Binding and Security Requirements.....	12
80	2.8.1.1.1 Conformance Criteria.....	12
81	2.8.1.2 User Interface Behavior.....	13
82	2.8.1.2.1 Conformance Criteria.....	13
83	2.8.2 Logout Responses.....	13
84	2.8.2.1 Binding and Security Requirements.....	13
85	2.8.2.1.1 Conformance Criteria.....	13
86	3 Conformance Classes.....	14
87	3.1 Standard.....	14
88	3.2 Standard with Logout.....	14
89	3.3 Full.....	14
90	Appendix A. Open Issues.....	15

91 Appendix B. Change Log..... 16  
92

---

# 1 Introduction

SAML V2.0 is a rich and extensible standard that must be profiled to be used interoperably, and the profiles that typically emerge from the broader standardization process usually remain fairly broad and include a number of options and features that increase the burden for implementers and make deployment-time decisions more difficult.

The Kantara Initiative eGovernment Implementation Profile provides a SAML V2.0 conformance specification for Identity Provider and Service Provider implementations operating in eGovernment federations and deployments. The profile is based on the SAML V2.0 specifications created by the Security Services Technical Committee (SSTC) of OASIS, and related specifications approved by that body. It constrains and supplements the base SAML V2.0 features, elements, and attributes required for eGovernment federations and deployments.

Implementation profiles define the features that software implementations must support such that deployers can be assured of the ability to meet their own (possibly varied) deployment requirements. Deployment profiles define specific options and constraints to which deployments are required to conform; they guide product configuration and federation operations, and provide criteria against which actual deployments may be tested. This document does not include a deployment profile, but reflects the features deemed necessary or desirable from software implementations in support of a variety of deployment profiles planned and in use. This includes requirements deemed useful to further the eventual goal of interfederation between deployments.

## 1.1 Notation

This specification uses normative text to describe the use of SAML capabilities.

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 [RFC2119]:

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

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:

- The prefix `saml2:` stands for the SAML 2.0 assertion namespace, `urn:oasis:names:tc:SAML:2.0:assertion`
- The prefix `saml2p:` stands for the SAML 2.0 protocol namespace, `urn:oasis:names:tc:SAML:2.0:protocol`
- The prefix `md:` stands for the SAML 2.0 metadata namespace, `urn:oasis:names:tc:SAML:2.0:metadata`
- The prefix `idpdisc:` stands for the Identity Provider Discovery Service Protocol and Profile [IdPDisco] namespace, `urn:oasis:names:tc:SAML:profiles:SSO:idp-discovery-protocol`

- 137 • The prefix `mdattr:` stands for the Metadata Extension for Entity Attributes Version 1.0 [MetaAttr]  
138 namespace, `urn:oasis:names:tc:SAML:metadata:attribute`

139 This specification uses the following typographical conventions in text: `<ns:Element>`, `Attribute`,  
140 **Datatype**, `OtherCode`.

## 141 1.2 Normative References

- 142 **[RFC2119]** IETF RFC 2119, *Key words for use in RFCs to Indicate Requirement Levels*,  
143 March 1997. <http://www.ietf.org/rfc/rfc2119.txt>
- 144 **[RFC2616]** IETF RFC 2616, *Hypertext Transfer Protocol – HTTP/1.1*, June 1999.  
145 <http://www.ietf.org/rfc/rfc2616.txt>
- 146 **[RFC2818]** IETF RFC 2818, *HTTP Over TLS*, May 2000. <http://www.ietf.org/rfc/rfc2818.txt>
- 147 **[RFC4051]** IETF RFC 4051, *Additional XML Security Uniform Resource Identifiers*, April  
148 2005. <http://www.ietf.org/rfc/rfc4051.txt>
- 149 **[HoKSSO]** OASIS Committee Specification, *SAML V2.0 Holder-of-Key Web Browser SSO*  
150 *Profile Version 1.0*, July 2009. [http://docs.oasis-](http://docs.oasis-open.org/security/saml/Post2.0/sstc-saml-holder-of-key-browser-sso-cs-01.pdf)  
151 [open.org/security/saml/Post2.0/sstc-saml-holder-of-key-browser-sso-cs-01.pdf](http://docs.oasis-open.org/security/saml/Post2.0/sstc-saml-holder-of-key-browser-sso-cs-01.pdf)
- 152 **[IdPDisco]** OASIS Committee Specification, *Identity Provider Discovery Service Protocol*  
153 *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)  
154 [saml-idp-discovery.pdf](http://docs.oasis-open.org/security/saml/Post2.0/sstc-saml-idp-discovery.pdf)
- 155 **[MetaAttr]** OASIS Committee Specification, *SAML V2.0 Metadata Extension for Entity*  
156 *Attributes Version 1.0*, August 2009. [http://docs.oasis-](http://docs.oasis-open.org/security/saml/Post2.0/sstc-metadata-attr.pdf)  
157 [open.org/security/saml/Post2.0/sstc-metadata-attr.pdf](http://docs.oasis-open.org/security/saml/Post2.0/sstc-metadata-attr.pdf)
- 158 **[MetalOP]** OASIS Committee Specification, *SAML V2.0 Metadata Interoperability Profile*  
159 *Version 1.0*, August 2009. [http://docs.oasis-open.org/security/saml/Post2.0/sstc-](http://docs.oasis-open.org/security/saml/Post2.0/sstc-metadata-iop.pdf)  
160 [metadata-iop.pdf](http://docs.oasis-open.org/security/saml/Post2.0/sstc-metadata-iop.pdf)
- 161 **[SAML2Core]** OASIS Standard, *Assertions and Protocols for the OASIS Security Assertion*  
162 *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)  
163 [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)
- 164 **[SAML2Meta]** OASIS Standard, *Metadata for the OASIS Security Assertion Markup Language*  
165 *(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)  
166 [metadata-2.0-os.pdf](http://docs.oasis-open.org/security/saml/v2.0/saml-metadata-2.0-os.pdf)
- 167 **[SAML2Bind]** OASIS Standard, *Bindings for the OASIS Security Assertion Markup Language*  
168 *(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)  
169 [bindings-2.0-os.pdf](http://docs.oasis-open.org/security/saml/v2.0/saml-bindings-2.0-os.pdf)
- 170 **[SAML2Prof]** OASIS Standard, *Profiles for the OASIS Security Assertion Markup Language*  
171 *(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)  
172 [profiles-2.0-os.pdf](http://docs.oasis-open.org/security/saml/v2.0/saml-profiles-2.0-os.pdf)
- 173 **[SAML2Err]** OASIS Approved Errata, *SAML V2.0 Errata*, Dec 2009. [http://docs.oasis-](http://docs.oasis-open.org/security/saml/v2.0/sstc-saml-approved-errata-2.0.pdf)  
174 [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)[http://www.oasis-](http://www.oasis-open.org/committees/download.php/37166/sstc-saml-approved-errata-2.0-02.pdf)  
175 [open.org/committees/download.php/37166/sstc-saml-approved-errata-2.0-02.pdf](http://www.oasis-open.org/committees/download.php/37166/sstc-saml-approved-errata-2.0-02.pdf)
- 176 **[SAML-X500]** OASIS Committee Specification, *SAML V2.0 X.500/LDAP Attribute Profile*, March  
177 2008. [http://docs.oasis-open.org/security/saml/Post2.0/sstc-saml-attribute-](http://docs.oasis-open.org/security/saml/Post2.0/sstc-saml-attribute-x500.pdf)  
178 [x500.pdf](http://docs.oasis-open.org/security/saml/Post2.0/sstc-saml-attribute-x500.pdf)
- 179 **[XMLEnc]** D. Eastlake et al. *XML Encryption Syntax and Processing*. World Wide  
180 Web Consortium Recommendation. <http://www.w3.org/TR/2002/REC-xmlenc-core-20021210/>
- 181 **[XMLSig]** D. Eastlake et al. *XML-Signature Syntax and Processing*. Second Edition. World  
182 Wide Web Consortium Recommendation, June 2008.  
183 <http://www.w3.org/TR/xmlsig-core/>

184 **Non-Normative References**

185	<b>[eGov15]</b>	Kyle Meadors, <i>Liberty Alliance eGov Profile for SAML 2.0 Version 1.5</i> .
186		<a href="http://www.projectliberty.org/liberty/content/download/4711/32210/file/Liberty_Alliance_eGov_Profile_1.5_Final.pdf">http://www.projectliberty.org/liberty/content/download/4711/32210/file/Liberty_Alliance_eGov_Profile_1.5_Final.pdf</a>
187		

---

## 2 SAML V2.0 Implementation Profile

188

189 This profile specifies behavior and options that implementations of a selected set of SAML V2.0 profiles  
190 [SAML2Prof] are required to support. The requirements specified are *in addition to* all normative  
191 requirements of the original profiles, as modified by the Approved Errata [SAML2Err], and readers should  
192 be familiar with all relevant reference documents. Any such requirements are not repeated here except  
193 where deemed necessary to highlight a point of discussion or draw attention to an issue addressed in  
194 errata, but remain implied.

195 SAML leaves substantial latitude to implementations with regard to how software is architected and  
196 combined with authentication and application infrastructure. Where the terms "Identity Provider" and  
197 "Service Provider" are used, they should be understood to include the total software footprint intended to  
198 provided the desired functionality; no specific assumptions are made as to how the required features are  
199 exposed to deployers, only that there is some method for doing so.

### 2.1 Required Information

200

201 | **Identification:** TBD <http://kantarainitiative.org/eGov/profiles/SAML2.0/v2.0>

202 | **Contact information:** TBD <http://kantarainitiative.org/confluence/display/eGov/Home>

203 | **Description:** Given below

204 | **Updates:** Liberty Alliance eGov Profile for SAML 2.0 [eGov15]

### 2.2 Metadata and Trust Management

205

206 Identity Provider, Service Provider, and Discovery Service implementations MUST support the use of  
207 SAML V2.0 Metadata [SAML2Meta] in conjunction with their support of the SAML V2.0 profiles referenced  
208 by subsequent sections. Additional expectations around the use of particular metadata elements related to  
209 profile behavior may be encountered in those sections.

210 Implementations MUST support the SAML V2.0 Metadata Interoperability Profile Version 1.0 [MetalOP].

211 Implementations MUST support the **TBD: insert profile for PKI here**

212 It is OPTIONAL for implementations to support the generation or exportation of metadata, but  
213 implementations MUST support the publication of metadata using the Well-Known-Location method  
214 defined in section 4.1 of [SAML2Meta] (under the assumption that entityID values used are suitable for  
215 such support).

216 Implementations MUST support the following mechanisms for the importation of metadata:

- 217 • local file
- 218 • remote resource at fixed location accessible via HTTP 1.1 [RFC2616] or HTTP 1.1 over TLS/SSL  
219 [RFC2818]

220 In the case of HTTP resolution, implementations MUST support use of the "ETag" header for cache  
221 management; other cache control support is OPTIONAL. Implementations SHOULD support the use of  
222 more than one fixed location for the importation of metadata, but MAY leave their behavior unspecified if a  
223 single entity's metadata is present in more than one source.

224 In accordance with [MetalOP], importation of multiple entities' metadata contained within an  
225 <md:EntitiesDescriptor> element MUST be supported.

226 Verification of metadata, if supported, MUST include XML signature verification at least at the root  
227 element level, and SHOULD support the following mechanisms for signature key trust establishment:

- 228 • direct comparison against known keys
- 229 • some form of path-based certificate validation against one or more trusted root certificates and
- 230 certificate revocation lists

231 The latter mechanism does not impose a particular profile for certificate validation, as no such profile has  
232 wide enough adoption across tools and libraries to warrant such a requirement, but should be understood  
233 as being consistent with the "usual" practices encountered in the implementation of certificate validation.  
234 Where possible, implementations SHOULD document known limitations of the mechanisms they employ.

235 Implementations SHOULD support the SAML V2.0 Metadata Extension for Entity Attributes Version 1.0  
236 [MetaAttr] and provide policy controls on the basis of SAML attributes supplied via this extension  
237 mechanism.

238 Finally, implementations SHOULD allow for the automated updating/reimportation of metadata without  
239 significant disruption of services.

## 240 **2.2.1 Conformance Criteria**

241 TBD

## 242 **2.3 Name Identifiers**

243 In conjunction with their support of the SAML V2.0 profiles referenced by subsequent sections, Identity  
244 Provider and Service Provider implementations MUST support the following SAML V2.0 name identifier  
245 formats, in accordance with the normative obligations associated with them by [SAML2Core]:

- 246 • `urn:oasis:names:tc:SAML:2.0:nameid-format:persistent`
- 247 • `urn:oasis:names:tc:SAML:2.0:nameid-format:transient`

248 Support for other formats is OPTIONAL.

## 249 **2.3.1 Conformance Criteria**

250 TBD

## 251 **2.4 Attributes**

252 In conjunction with their support of the SAML V2.0 profiles referenced by subsequent sections, Identity  
253 Provider and Service Provider implementations MUST support the generation and consumption of  
254 `<saml2:Attribute>` elements that conform to the SAML V2.0 X.500/LDAP Attribute Profile [SAML-  
255 X500].

256 The ability to support `<saml2:AttributeValue>` elements whose values are not simple strings (e.g.,  
257 `<saml2:NameID>`, or other XML values) is OPTIONAL. Such content could be base64-encoded as an  
258 alternative.

## 259 **2.4.1 Conformance Criteria**

260 TBD

## 261 **2.5 Browser Single Sign-On**

262 This section defines an implementation profile of the SAML V2.0 Web Browser SSO Profile [SAML2Prof].



263 **2.5.1 Identity Provider Discovery**

264 Service Provider and Discovery Service implementations MUST support the Identity Provider Discovery  
265 Service Protocol Profile in conformance with section 2.4.1 of [IdPDisco].

266 **2.5.1.1 Conformance Criteria**

267 TBD

268 **2.5.2 Authentication Requests**

269 **2.5.2.1 Binding and Security Requirements**

270 Identity Provider and Service Provider implementations MUST support the use of the HTTP-Redirect  
271 binding [SAML2Bind] for the transmission of `<saml2p:AuthnRequest>` messages, including the  
272 generation or verification of signatures in conjunction with this binding.

273 Support for other bindings is OPTIONAL.

274 **2.5.2.1.1 Conformance Criteria**

275 TBD

276 **2.5.2.2 Message Content**

277 In addition to standard core- and profile-driven requirements, Service Provider implementations MUST  
278 support the inclusion of at least the following `<saml2p:AuthnRequest>` child elements and attributes  
279 (when appropriate):

- 280 • `AssertionConsumerServiceURL`
- 281 • `ProtocolBinding`
- 282 • `ForceAuthn`
- 283 • `IsPassive`
- 284 • `AttributeConsumingServiceIndex`
- 285 • `<saml2p:RequestedAuthnContext>`
- 286 • `<saml2p:NameIDPolicy>`

287 Identity Provider implementations MUST support all `<saml2p:AuthnRequest>` child elements and  
288 attributes defined by [SAML2Core], but MAY provide that support in the form of returning appropriate  
289 errors when confronted by particular request options. However, implementations MUST fully support the  
290 options enumerated above.

291 Implementations MAY limit their support of the `<saml2p:RequestedAuthnContext>` element to the  
292 value "exact" for the `Comparison` attribute.

293 **2.5.2.2.1 Conformance Criteria**

294 TBD

## 295 **2.5.3 Responses**

### 296 **2.5.3.1 Binding and Security Requirements**

297 Identity Provider and Service Provider implementations MUST support the use of the HTTP-POST and  
298 HTTP-Artifact bindings [SAML2Bind] for the transmission of `<saml2p:Response>` messages.

299 Support for other bindings, and for artifact types other than  
300 `urn:oasis:names:tc:SAML:2.0:artifact-04`, is OPTIONAL.

301 Identity Provider and Service Provider implementations MUST support the generation and consumption of  
302 unsolicited `<saml2p:Response>` messages (i.e., responses that are not the result of a  
303 `<saml2p:AuthnRequest>` message).

304 Identity Provider implementations MUST support the issuance of `<saml2p:Response>` messages (with  
305 appropriate status codes) in the course of encountering error conditions, provided that the user agent  
306 remains available and the location to deliver the response is knowable. Note that this is a stronger  
307 requirement than the comparable language in [SAML2Prof].

308 Identity Provider and Service Provider implementations MUST support the signing of  
309 `<saml2:Assertion>` elements in responses; support for signing of the `<saml2p:Response>` element  
310 is OPTIONAL.

311 Identity Provider and Service Provider implementations MUST support the use of XML Encryption via the  
312 `<saml2:EncryptedAssertion>` element when using the HTTP-POST binding; support for the  
313 `<saml2:EncryptedID>` and `<saml2:EncryptedAttribute>` elements is OPTIONAL.

#### 314 **2.5.3.1.1 Conformance Criteria**

315 TBD

### 316 **2.5.3.2 Message Content**

317 The Web Browser SSO Profile allows responses to contain any number of assertions and statements.  
318 Identity Provider implementations MUST allow the number of `<saml2:Assertion>`,  
319 `<saml2:AuthnStatement>`, and `<saml2:AttributeStatement>` elements in the  
320 `<saml2p:Response>` message to be limited to one.

321 In turn, Service Provider implementations MAY limit support to a single instance of those elements when  
322 processing `<saml2p:Response>` messages.

323 Identity Provider implementations MUST support the inclusion of a `Consent` attribute in  
324 `<saml2p:Response>` messages, and a `SessionIndex` attribute in `<saml2:AuthnStatement>`  
325 elements.

326 Service Provider implementations that provide some form of session semantics MUST support the  
327 `<saml2:AuthnStatement>` element's `SessionNotOnOrAfter` attribute.

#### 328 **2.5.3.2.1 Conformance Criteria**

329 TBD

## 330 **2.5.4 Artifact Resolution**

331 Pursuant to the requirement in section 2.5.3.1 for support of the HTTP-Artifact binding [SAML2Bind] for  
332 the transmission of <saml2p:Response> messages, implementations MUST support the SAML V2.0  
333 Artifact Resolution profile [SAML2Prof] as constrained by the following subsections.

### 334 **2.5.4.1 Artifact Resolution Requests**

335 Identity Provider and Service Provider implementations MUST support the use of the SAML SOAP (using  
336 HTTP as a transport) binding [SAML2Bind] for the transmission of <saml2p:ArtifactResolve>  
337 messages.

338 Implementations MUST support the use of SAML message signatures and TLS server authentication to  
339 authenticate requests; support for TLS client authentication, or other forms of authentication in conjunction  
340 with the SAML SOAP binding, is OPTIONAL.

#### 341 **2.5.4.1.1 Conformance Criteria**

342 TBD

### 343 **2.5.4.2 Artifact Resolution Responses**

344 Identity Provider and Service Provider implementations MUST support the use of the SAML SOAP (using  
345 HTTP as a transport) binding [SAML2Bind] for the transmission of <saml2p:ArtifactResponse>  
346 messages.

347 Implementations MUST support the use of SAML message signatures and TLS server authentication to  
348 authenticate responses; support for TLS client authentication, or other forms of authentication in  
349 conjunction with the SAML SOAP binding, is OPTIONAL.

#### 350 **2.5.4.2.1 Conformance Criteria**

351 | TBD

352 |

## 353 | **2.6 Browser Holder of Key Single Sign-On**

354 This section defines an implementation profile of the SAML V2.0 Holder-of-Key Web Browser SSO Profile  
355 Version 1.0 [HoKSSO].

356 The implementation requirements defined in section 2.5 for the non-holder-of-key profile apply to  
357 implementations of this profile.

### 358 **2.6.1 Conformance Criteria**

359 TBD

## 360 | **2.7 SAML 2.0 Proxying**

361 | Section 3.4.1.5 of [SAML2Core] defines a formalized approach to proxying the SAML 2.0 Authentication  
362 | Request protocol between multiple Identity Providers. This section defines an implementation profile for  
363 | this behavior suitable for composition with the Single Sign-On profiles defined in sections 2.5 and 2.6. The  
364 | requirements of the profile are imposed on Identity Provider implementations acting as a proxy.

## 365 | **2.7.1 Authentication Requests**

366 | Proxying Identity Provider implementations MUST support the mapping of incoming to outgoing  
367 | <saml2p:RequestedAuthnContext> and <saml2p:NameIDPolicy> elements, such that deployers  
368 | may choose to pass through values or map between different vocabularies as required.

### 369 | **2.7.1.1 Conformance Criteria**

370 | TBD

## 371 | **2.7.2 Responses**

372 | Proxying Identity Provider implementations MUST support the mapping of incoming to outgoing  
373 | <saml2:AuthnContext> elements, such that deployers may choose to pass through values or map  
374 | between different vocabularies as required.

### 375 | **2.7.2.1 Conformance Criteria**

376 | TBD

377 |

## 378 | **2.8 Single Logout**

379 | This section defines an implementation profile of the SAML V2.0 Single Logout Profile [SAML2Prof].

380 | For clarification, the technical requirements for each message type below reflect the intent to normatively  
381 | require initiation of logout by a Service Provider using either the front- or back-channel, and  
382 | initiation/propagation of logout by an Identity Provider using the back-channel.

### 383 | **2.8.1 Logout Requests**

#### 384 | **2.8.1.1 Binding and Security Requirements**

385 | Identity Provider implementations MUST support the SAML SOAP (using HTTP as a transport) binding  
386 | [SAML2Bind] for the issuance of <saml2p:LogoutRequest> messages, and MUST support the SAML  
387 | SOAP (using HTTP as a transport) and HTTP-Redirect bindings [SAML2Bind] for the reception of  
388 | <saml2p:LogoutRequest> messages.

389 | Service Provider implementations MUST support the SAML SOAP (using HTTP as a transport) binding  
390 | [SAML2Bind] for both issuance and reception of <saml2p:LogoutRequest> messages.

391 | Support for other bindings is OPTIONAL.

392 | Implementations MUST support the use of SAML message signatures and TLS server authentication to  
393 | authenticate requests; support for TLS client authentication, or other forms of authentication in conjunction  
394 | with the SAML SOAP binding, is OPTIONAL.

395 | Identity Provider and Service Provider implementations MUST support the use of XML Encryption via the  
396 | <saml2:EncryptedID> element when using the HTTP-Redirect binding.

##### 397 | **2.8.1.1.1 Conformance Criteria**

398 | TBD

## 399 **2.8.1.2 User Interface Behavior**

400 Identity Provider implementations MUST support both user-initiated termination of the local session only  
401 and user-initiated Single Logout. Upon receipt of a <saml2p:LogoutRequest> message via a front-  
402 channel binding. Identity Provider implementations MUST support user intervention governing the choice  
403 of propagating logout to other Service Providers, or limiting the operation to the Identity Provider. Of  
404 course, implementations MUST return status information (e.g. partial logout indication) as appropriate.

405 ~~option. Identity Provider and Service Provider implementations MUST support "local" logout as well as~~  
406 ~~initiation of Single Logout, subject to deployer and user Service Provider implementations MUST support~~  
407 ~~both user-initiated termination of the local session only and user-initiated Single Logout.~~

408 TBD: Requirements on administrative logout (i.e., not the user)?

### 409 **2.8.1.2.1 Conformance Criteria**

410 TBD

## 411 **2.8.2 Logout Responses**

### 412 **2.8.2.1 Binding and Security Requirements**

413 Identity Provider implementations MUST support the SAML SOAP (using HTTP as a transport) and  
414 HTTP-Redirect bindings [SAML2Bind] for the issuance of <saml2p:LogoutResponse> messages, and  
415 MUST support the SAML SOAP (using HTTP as a transport) binding [SAML2Bind] for the reception of  
416 <saml2p:LogoutResponse> messages.

417 Service Provider implementations MUST support the SAML SOAP (using HTTP as a transport) binding  
418 [SAML2Bind] for both issuance and reception of <saml2p:LogoutResponse> messages.

419 Support for other bindings is OPTIONAL.

420 Implementations MUST support the use of SAML message signatures and TLS server authentication to  
421 authenticate responses; support for TLS client authentication, or other forms of authentication in  
422 conjunction with the SAML SOAP binding, is OPTIONAL.

### 423 **2.8.2.1.1 Conformance Criteria**

424 TBD

425

## 3 Conformance Classes

426

### 3.1 Standard

427

Conforming Identity Provider and/or Service Provider implementations MUST support the normative requirements in sections 2.2, 2.3, 2.4, and 2.5.

428

429

Implementations MUST support the signature and digest algorithms identified by the following URIs in conjunction with the creation and verification of XML Signatures [XMLSig]:

430

431

- <http://www.w3.org/2001/04/xmldsig-more#rsa-sha256> (defined in [RFC4051])

432

- <http://www.w3.org/2001/04/xmlenc#sha256> (defined in [XMLEnc])

433

### 3.2 Standard with Logout

434

Conforming Identity Provider and/or Service Provider implementations MUST meet the conformance requirements in section 3.1, and MUST in addition support the normative requirements in sections 2.2, 2.3, 2.4, 2.5, and 2.8.

435

436

437

### 3.3 Full

438

Conforming Identity Provider and/or Service Provider implementations MUST meet the conformance requirements in section 3.1, and MUST in addition support the normative requirements in sections 2.6, 2.7, and 2.8. ~~Conforming Identity Provider and/or Service Provider implementations MUST support all normative requirements in section 2.~~

439

440

441

442

## Appendix A. Open Issues

443  
444  
445

- Need an alternative to IOP, or agreement to drop PKI outside of metadata exchange. Alternative needs to specify PKI to some degree AND address the exact content and semantics of metadata as relates to runtime certificate evaluation and/or identity of SAML peer.

446

- Do implementations need to be able to prevent non-use of TLS on front-channel?

447

- Need for more than exact AuthnContext matching?

448

- Need for specific MTI behavior on ACS checking?

449  
450

- ~~Need some clarification of some of the original single logout language around user consent. Single logout language around UI and consent needs review, and need text on administrative logout.~~

451

- ~~Updated crypto algorithm conformance rules for implementers and deployers?~~

452

- Populate with conformance criteria.

453

- Is feature discussion of AuthnContext and metadata tagging enough to cover LOA issues?

454

- ~~IdP proxying~~

455

- Need to bump HoK reference to new profile version ~~once~~if it reaches CS-02

456

## Appendix B. Change Log

457

- Draft 01: first working draft based on similar document created by InCommon Federation

458

459

460

461

- Draft 02: first round of feedback incorporated, deployment section dropped, new section on Artifact Resolution added, artifact added for SSO responses, SOAP added for logout, discovery moved under SSO, language on non-string attributes added, changed SHOULD to MUST for IdP support of selected AuthnRequest features

462

463

464

- Draft 03: moved Artifact Resolution into a SSO profile subsection, new language on SOAP security and SLO bindings, added metadata publication via WKL, added language on IdP error handling, added Holder of Key SSO profile, added Conformance Classes

465

466

- [Draft 04: added UI language around SLO, layered conformance language and added MTI algorithms, added section for Proxying](#)