



Basic Metadata Aggregation Profile

Version 0.2

[Stage]

August 9th, 2010

Specification URIs:

This Version:

[http://docs.oasis-open.org/ssstc/\[additional path/filename\].html](http://docs.oasis-open.org/ssstc/[additional path/filename].html)
[http://docs.oasis-open.org/ssstc/\[additional path/filename\].doc](http://docs.oasis-open.org/ssstc/[additional path/filename].doc)
[http://docs.oasis-open.org/ssstc/\[additional path/filename\].pdf](http://docs.oasis-open.org/ssstc/[additional path/filename].pdf)

Previous Version:

[http://docs.oasis-open.org/ssstc/\[additional path/filename\].html](http://docs.oasis-open.org/ssstc/[additional path/filename].html)
[http://docs.oasis-open.org/ssstc/\[additional path/filename\].doc](http://docs.oasis-open.org/ssstc/[additional path/filename].doc)
[http://docs.oasis-open.org/ssstc/\[additional path/filename\].pdf](http://docs.oasis-open.org/ssstc/[additional path/filename].pdf)

Latest Version:

[http://docs.oasis-open.org/ssstc/\[additional path/filename\].html](http://docs.oasis-open.org/ssstc/[additional path/filename].html)
[http://docs.oasis-open.org/ssstc/\[additional path/filename\].doc](http://docs.oasis-open.org/ssstc/[additional path/filename].doc)
[http://docs.oasis-open.org/ssstc/\[additional path/filename\].pdf](http://docs.oasis-open.org/ssstc/[additional path/filename].pdf)

Technical Committee:

OASIS [official name of technical committee] TC

Chair(s):

[Chair name]
[Chair name]

Editor(s):

Andreas Åkre Solberg, UNINETT AS, andreas.solberg@uninett.no

Related work:

This specification is related to:

- Metadata for the OASIS Security Assertion Markup Language (SAML)

Abstract:

This profile defines a basic set of processing rules when aggregating SAML Metadata from multiple sources. The profile aims at being extensible.

Status:

This document was last revised or approved by the [TC name | membership of OASIS] on the above date. The level of approval is also listed above. Check the “Latest Version” or “Latest Approved Version” location noted above for possible later revisions of this document.

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

For information on whether any patents have been disclosed that may be essential to implementing this specification, and any offers of patent licensing terms, please refer to the Intellectual Property Rights section of the Technical Committee web page (<http://www.oasis-open.org/committees/sstc/ipr.php>).

The non-normative errata page for this specification is located at <http://www.oasis-open.org/committees/sstc/>.

Notices

Copyright © OASIS® 2008. All Rights Reserved.

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

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

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

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

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

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

OASIS takes no position regarding the validity or scope of any intellectual property or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; neither does it represent that it has made any effort to identify any such rights. Information on OASIS' procedures with respect to rights in any document or deliverable produced by an OASIS Technical Committee can be found on the OASIS website. Copies of claims of rights made available for publication and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary

rights by implementers or users of this OASIS Committee Specification or OASIS Standard, can be obtained from the OASIS TC Administrator. OASIS makes no representation that any information or list of intellectual property rights will at any time be complete, or that any claims in such list are, in fact, Essential Claims.

The names "OASIS", [insert specific trademarked names and abbreviations here] are trademarks of OASIS, the owner and developer of this specification, and should be used only to refer to the organization and its official outputs. OASIS welcomes reference to, and implementation and use of, specifications, while reserving the right to enforce its marks against misleading uses. Please see <http://www.oasis-open.org/who/trademark.php> for above guidance.

Table of Contents

1	Introduction	7
1.1	Terminology	7
1.2	Normative References	7
1.3	Non-Normative References	7
2	Aggregator Configuration Statement	8
3	Registration of Input Sources and Trust Management	9
4	Processing	10
4.1	Fetching metadata from input sources	10
4.2	Processing each entity	11
4.2.1	Entity Acceptance Processing	11
4.2.2	Removing information	11
4.2.3	Calculating entity expiration time	11
4.3	Processing Extensions	12
4.4	Processing the RoleDescriptors of an Entity	12
4.5	Cache Entity	12
4.6	Preparing the output document	12
5	Extending this profile	14
#	Conformance	15
A.	Acknowledgements	16
B.	XSLT describing behavior of Basic Aggregator	17
C.	Example Aggregator Configuration Statement	18
D.	Aggregator Extension for Documentation and Registration Information	19
D.1	Processing Metadata Document	19
D.2	Entity Acceptance Processing	19
D.3	Processing Extensions	19
D.4	Processing RoleDescriptors	19
E.	Aggregator Extension for Entity Attributes	20
E.1	Processing Metadata Document	20
E.2	Entity Acceptance Processing	20
E.3	Processing Extensions	20
E.4	Processing RoleDescriptors	20
F.	eduGAIN Aggregator Profile	21
F.1	Processing Metadata Document	21
F.2	Entity Acceptance Processing	21
F.3	Processing Extensions	21
F.4	Processing RoleDescriptors	21
G.	Revision History	22

1 Introduction

This profile describes the processing rules of a Metadata Aggregator; a component that fetches *SAML V.2.0 Metadata* [SAML2Meta] from multiple input sources over HTTP, merges the metadata and publishes the result set of metadata.

The processing rules of aggregating metadata, is split into these sections:

1. Fetching metadata documents
2. Processing metadata documents
3. Processing each entity
4. Preparing output document

1.1 Terminology

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

1.2 Normative References

- [RFC2119] S. Bradner, *Key words for use in RFCs to Indicate Requirement Levels*, <http://www.ietf.org/rfc/rfc2119.txt>, IETF RFC 2119, March 1997.
- [SAML2Meta] OASIS, *Metadata for the OASIS Security Assertion Markup Language (SAML) V2.0*
<http://docs.oasis-open.org/security/saml/v2.0/saml-metadata-2.0-os.pdf>
- [MDAttrib] OASIS, *SAML V2.0 Metadata Extension for Entity Attributes Version 1.0*
<http://docs.oasis-open.org/security/saml/Post2.0/sstc-metadata-attr.html>
- [MetaReg] OASIS, *SAML Metadata Documentation and Registration Information Extensions, version 0.2*
https://spaces.internet2.edu/download/attachments/9731/saml_md_dri_02.odt
- [MDX] *IETF Internet-Draft, Metadata Query Protocol*
<http://tools.ietf.org/pdf/draft-lajoie-md-query-00.pdf>

1.3 Non-Normative References

- [Reference] [Full reference citation]

34

35 2 Aggregator Configuration Statement

36 A deployment of an aggregator, MUST in order to satisfy this profile make available an
37 *aggregator configuration statement* document to all consumers of the output metadata.

38 The configuration statement MUST include at least the following information about this
39 deployment:

- 40 • Supported aggregation profiles: this profile, as well as any number of
41 complimentary profiles
- 42 • List of supported metadata access protocols, and corresponding HTTP
43 endpoints.
- 44 • Public key or certificate of the aggregator or a corresponding trust root (CA),
45 used for signing the output document(s)
- 46 • Aggregator fetch interval
- 47 • Maximum validity period of input entity

48 Complimentary profiles MAY define additional information that is required to be stated in
49 the aggregator configuration statement.

50 **3 Registration of Input Sources and Trust**
51 **Management**

52 For each input source, from where the aggregator fetches metadata, the aggregator
53 MUST register at least the following information associated with each input source:

- 54 • PublisherID (adopting terminology from [MetaReg])
- 55 • A list of one or more equivalent HTTP endpoints, from where a full metadata
56 document can be retrieved. Multiple endpoints MAY be provided for improved
57 reliability. The list is not prioritized, and all endpoints are equivalent.
- 58 • For each endpoint as mentioned above, one or more public key / certificate
59 MUST be provided. This public key / certificate represents the signature on the
60 top-level element of the published metadata document. Multiple public keys /
61 certificates MAY be provided in order for a smooth rollover.

62 4 Processing

63 4.1 Fetching metadata from input sources

64 The aggregator fetches metadata from each input source at a configured interval. The
65 retrieved metadata document will be processed for some basic verification, described in
66 this section. If any of these verification steps fails, further processing **MUST** be stopped,
67 and aggregator will move on to processing the next input source. The aggregator
68 **SHOULD** implement re-tries depending on the nature of the failure; in example the
69 aggregator may wait and retry downloading the document if a network error occurred.

70 In the case of a network or access problem, the aggregator **SHOULD** iterate through the
71 list of endpoints, if multiple are configured, as described in section 3.

72 The aggregator **MUST** perform the following verification steps:

- 73 1. The provided document is well formed XML.
- 74 2. The root element is an `md:EntitiesDescriptor` or `md:EntityDescriptor`.
- 75 3. The document **MUST** be signed at the root element. The signing key / certificate
76 **MUST** correspond to the configured key / certificate / CA for this particular input
77 source, as described in section 3.
- 78 4. The metadata document **MUST** be schema-valid, validating the following XML
79 namespaces:
 - 80 • `urn:oasis:names:tc:SAML:2.0:metadata`
 - 81 • `urn:oasis:names:tc:SAML:2.0:assertion`
 - 82 • `http://www.w3.org/2000/09/xmlsig#`
 - 83 • `http://www.w3.org/2001/04/xmlenc#`
- 84 5. Complimentary profiles referred to in the *Aggregator Configuration Statement*
85 may add further verification steps under the header "Processing Metadata
86 Document", including in example additional XML namespaces to validate.

87 **NOTE:** `RoleDescriptors` with a namespace unknown to the aggregator,
88 not mentioned in this profile or any supported complimentary profiles,
89 **SHOULD** be treated as if the element was not there in the validation step.
90 Without access to the schemas, the aggregator will not know if the
91 element is inheriting from the abstract `RoleDescriptor` element or not, and
92 **SHOULD** consider it as it was valid.

93 Failure at this point **MUST NOT** invalidate already cached entities from the same input
94 source.

95 If verification succeeded, the aggregator **MUST** remove cached entities, originating from
96 the same input source, but is not present in the new metadata document.

97 The metadata document **MAY** include any level of nested `md:EntitiesDescriptor`
98 elements. No semantic meaning should be associated with the hierarchy in how
99 `md:EntityDescriptor` elements are located, beyond what is defined in [SAMLMeta].

100 Further processing will be performed for each `md:EntitiesDescriptor`.

101 **4.2 Processing each entity**

102 The aggregator will process each entity individually as specified in the following
103 sections.

104 Important: content that is not associated with any explicit process instruction in this
105 section, the aggregator MUST pass through unmodified.

106 **4.2.1 Entity Acceptance Processing**

107 An entity rejected at this point MUST NOT invalidate an already cached version of the
108 same entity.

109 An aggregator MUST provide a mechanism to protect Entities from being hijacked by
110 another input source. This section describes the default processing rules protecting
111 against Entity hijacking. A complimentary profile MAY provide an alternative measures
112 of protection, such as a controlled prefixed entityID namespace.

113 The aggregator SHOULD store an association table mapping an EntityID with the
114 PublisherID of the input source where it was first discovered. This association SHOULD
115 be stored for at least the validity period of the contained metadata.

116 If the aggregator receives an `md:EntityDescriptor` with an EntityID registered at
117 another Input source, the aggregator MUST reject this `md:EntityDescriptor`.

118 A complimentary profile MAY provide additional entity acceptance processing rules
119 under a section named "Entity Acceptance Processing".

120 **4.2.2 Removing information**

121 If the `md:EntityDescription` element is signed, this signature MUST be removed. The
122 reason is that it will not be valid, due to the modifications necessary in the following
123 processing.

124 The aggregator MUST ensure that there will be no collisions in the XML ID attributes of
125 all of its entities. The RECOMMENDED approach is to remove the XML ID attributes at
126 the `md:EntityDescription`.

127 **4.2.3 Calculating entity expiration time**

128 The expiration time of an entity is calculated to be the first of these times:

- 129 • The `validUntil` attribute of the `md:EntityDescriptor` element
- 130 • The `validUntil` attribute of each of the surrounding `md:EntitiesDescriptor`
131 elements
- 132 • The maximum validity period configured for the aggregator
- 133 • The `cacheDuration` attribute of the `md:EntityDescriptor` element + time the
134 metadata was fetched
- 135 • The `cacheDuration` attribute of each of the surrounding `md:EntitiesDescriptor`
136 elements + time the metadata was fetched

137 The `validUntil` attribute of the `md:EntityDescriptor` MUST be set to the new
138 calculated expiration time (replace the current attribute value if present).

139 The aggregator MAY remove the `cacheDuration` attribute of the `md:EntityDescriptor` or
140 leave it unmodified.

141 Note: expiration information on `RoleDescriptor` elements or other elements inside an
142 `md:EntityDescriptor` SHOULD not be considered by the aggregator but passed
143 through.

144 4.3 Processing Extensions

145 The Basic Aggregator Profile does not define any special processing of any particular
146 Extension namespace, but any complimentary profile MAY define a set of Extension
147 namespaces associated with processing rules under the section header “Processing
148 Extensions”.

149 The aggregator MUST express a default extension namespace processing rule in the
150 aggregator configuration statement. It may select one of the following default policies:

151 **PASSTHRU**

152 Extension is passed through without further processing.

153 **REMOVE**

154 Extension is removed from metadata.

155 4.4 Processing the RoleDescriptors of an Entity

156 The Basic Aggregator Profile does not define any special processing of any particular
157 RoleDescriptor, but any complimentary profile MAY define a set of namespaced
158 RoleDescriptor elements associated with processing rules under the section header
159 “Processing the RoleDescriptor of an Entity”.

160 The aggregator SHOULD enforce the PASSTHRU policy for all the RoleDescriptor,
161 unless explicitly overridden in a complimentary profile.

162 The aggregator MUST express a default RoleDescriptor processing policy in its
163 *Aggregator Configuration Statement*. It may select one of the following default policies:

164 **PASSTHRU**

165 RoleDescriptor is passed through without further processing.

166 **REMOVE**

167 RoleDescriptor is removed from metadata.

168 4.5 Cache Entity

169 After Entity is processed and accepted, the processed Entity MUST be cached by the
170 aggregator for the validity period of the entity.

171 Each cached Entity instance MUST be independent of other Entities, even those
172 originating from the same input source.

173 4.6 Preparing the output document

174 The aggregator MUST examine the expiration time of each entity, and if an entity is
175 expired, the aggregator MUST not include the entity in the output.

176 It is RECOMMENDED that the aggregator adds non-destructive line-breaks and
177 indentation of easier human inspection – before signing.

- 178 The aggregator SHOULD perform non-destructive XML namespace declaration
179 optimization.
- 180 The aggregator SHOULD prepare an output metadata document including all entities in
181 a flat structure (only one level of `md:EntitiesDescriptor`, the root note). The document
182 MUST be provided over HTTP and the document MUST be signed at the root note with
183 the public key / certificate indicated in the configuration statement.
- 184 This profile does not dictate any particular access protocol for the output metadata, but
185 RECOMMENDS taking the [MDX] protocol into consideration.

186 **5 Extending this profile**

187 A profile that extends the basic aggregator profile **MUST** have a name, and an
188 introduction that explains the purpose of the profile. Next it **MAY** include any of these
189 sections:

- 190 6. Aggregator configuration statement
- 191 7. Processing Metadata Document
- 192 8. Entity Acceptance Processing
- 193 9. Processing Extensions
- 194 10. Processing the RoleDescriptor of an Entity

195 Description of processing rules in these sections are handled are described in the
196 appropriate location in this profile.

197 **# Conformance**

198 The last numbered section in the specification must be the Conformance section.

199 Conformance Statements/Clauses go here.

200 **A. Acknowledgements**

201 The following individuals have participated in the creation of this specification and are
202 gratefully acknowledged:

203 **Participants:**

204 Leif Johansson, NorduNet
205 Lukas Hämmerle, SWITCH
206 Thomas Legenhagger, SWITCH
207 Miroslav Milinovic, SRCE
208 Dubravko Penezic, SRCE
209 Dubravko Voncina, SRCE
210 Mehdi Hached, RENATER

211 **B. XSLT describing behavior of Basic Aggregator**

212 Most of the behavior described in this document MAY be represented in the format of
213 XSLT.

214 The XSLT is yet to be written.

215 C. Example Aggregator Configuration Statement

216 Supported Profiles

- 217 • Basic Aggregation Profile
- 218 • eduGAIN Aggregator Profile
- 219 • Aggregator Extension for Documentation and Registration Information
- 220 • Aggregator Extension for Entity Attributes

221 Metadata Access Protocols

- 222 • Basic HTTP access:
223 <http://aggregator.edugain.edu/all.xml>
- 224 • MDX:
225 <http://aggregator.edugain.edu/mdx/>

226 Signing certificate

```
227 -----BEGIN CERTIFICATE-----  
228 MIICgTCCAeoCCQCbOlrWdX7FTANBgkqhkiG9w0BAQUFADCBhDELMAkGA1UEBhMC  
229 Tk8xGDAWBgNVBAgTD0FuZHJlYXNAdW5pbmV0dC5ubzAeFw0wNzA2MTUxMjAxMzVaFw0wNzA4  
230 VQKKEwdVtk1ORVRUMRgwFgYDVQDEw9mZWlkZS5lcxhbmccubm8xITAfBgkqhkiG  
231 9w0BCQEWEmFuZHJlYXNAdW5pbmV0dC5ubzAeFw0wNzA2MTUxMjAxMzVaFw0wNzA4  
232 MTQxMjAxMzVaMIGEMQswCQYDVQGEwJOTzEYMBYGA1UECBMPQW5kcmVhcyBTb2xi  
233 ZXJnMQwwCgYDVQHEwNGB28xEDAObGNVBAoTB1VOSU5FVFBQxGDAWBgNVBAMTD2Zl  
234 aWRlLmVybGFuZy5ubzEhMB8GCSqGSIb3DQEJARYSYW5kcmVhcyBTb2xiZW5v  
235 MIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQDibvhr7P516x/S3BqKxupQe0LO  
236 NolIupiBOesCO3SHbDr13+q9TbfnfmE04rNuMcPsIxBl61TddpIesLCn7c8aPHIS  
237 K0tPlAeTZSnb8QAU7arJzq3+PbrP5uW3TcfCGPtKTythOge/0lJbo078dVhXQ14d  
238 1EDwXJW1rRXuUt4C8QIDAQABMA0GCSqGSIb3DQEBBQUAA4GBACDVfp86HObqY+e8  
239 BUoWQ9+VMQx1ASDohBjwOsg2WykUqRXF+dLfcUH9dWR63CtZIKFDbStNomPnQz7n  
240 bK+onygwBspVEbnHuUihZq3ZUdmumQqCw4Uvs/1Uvq3orOo/WJVhTyvLgFVK2Qar  
241 Q4/67OzfHd7R+POBXhophSMv1ZOo  
242 -----END CERTIFICATE-----
```

243 Aggregator fetch interval

244 24 hours

245 Maximum metadata validity period

246 5 days

247 Default Extension Processing Policy

248 REMOVE

249 Default RoleDescriptor Processing Policy

250 REMOVE

251 **D. Aggregator Extension for Documentation and**
252 **Registration Information**

253 This profile extends the Basic Aggregator Profile. It assumes that both the input sources
254 and the aggregator support the SAML Metadata Documentation and Registration
255 Information Extensions [MDReg].

256 **D.1 Processing Metadata Document**

257 When validating the document, the aggregator MUST in addition consider the following
258 additional namespace:

- 259
 - `urn:oasis:names:tc:SAML:2.0:metadata:dri`

260 All `md:EntitiesDescriptor` elements containing an `md:Extension` with a
261 `dri:DocumentInfo` or a `dri:RegistrationInfo` MUST apply the `dri:DocumentInfo` and
262 `dri:RegistrationInfo` elements to all containing `md:EntityDescriptor`, to be
263 processed in the “Entity Acceptance Processing” step.

264 **D.2 Entity Acceptance Processing**

265 If multiple `dri:DocumentInfo` or `dri:RegistrationInfo` elements are associated with the
266 entity, values from the inner elements MUST take precedence.

267 If a cached version of the Entity contains a `dri:DocumentInfo/SerialNumber` the
268 aggregator MUST not accept a new version of the Entity unless the new Entity contains
269 a `dri:DocumentInfo/SerialNumber` being the same or a higher integer value.

270 If a cached version of the Entity contains a `dri:DocumentInfo/CreationInstant` the
271 aggregator MUST not accept a new version of the Entity unless the new Entity contains
272 a `dri:DocumentInfo/CreationInstant` being the same or a more recent timestamp.

273 **D.3 Processing Extensions**

274 If multiple `dri:DocumentInfo` or `dri:RegistrationInfo` elements are associated with
275 the entity, values from the inner elements MUST take precedence. The aggregator
276 MUST merge multiple `dri:DocumentInfo` and `dri:RegistrationInfo` elements into one
277 of each. Each value (sub element with content) MUST be considered independent from
278 the others.

279 **D.4 Processing RoleDescriptors**

280 NA

281 E. Aggregator Extension for Entity Attributes

282 This profile extends the Basic Aggregator Profile adding support and processing rules
283 for SAML V2.0 Metadata Extension for Entity Attributes [MDAttrib].

284 E.1 Processing Metadata Document

285 When validation the document, the aggregator MUST in addition consider the following
286 additional namespace:

- 287 • `urn:oasis:names:tc:SAML:metadata:attribute`

288 Extension elements in this namespace, associated with any `md:EntitiesDescriptor`
289 MUST be ignored.

290 E.2 Entity Acceptance Processing

291 NA

292 E.3 Processing Extensions

293 If an Entity contains an `mdata:EntityAttribute` this element SHOULD be sent through
294 the aggregator unmodified.

295 No further processing of the `mdata:EntityAttribute` content is defined in this profile
296 extension.

297 E.4 Processing RoleDescriptors

298 NA

299 **F. eduGAIN Aggregator Profile**

300 This profile extends the Basic Aggregator Profile.

301 NOTE: This appendix is only temporarily part of this document. Will
302 eventually be made into a separate document. It is convenient to have this
303 inside, until the basic profile is somewhat more mature.

304 **F.1 Processing Metadata Document**

305 The metadata document SHOULD be schema valid against the XML namespaces in the
306 Basic Aggregator Profile, but in addition the following XML namespaces:

307 ...

308 **F.2 Entity Acceptance Processing**

309 The entity MUST include the required information as specified in eduGAIN Metadata
310 Profile.

311 **F.3 Processing Extensions**

312 ...

313 **F.4 Processing RoleDescriptors**

314 The aggregator MUST REJECT all RoleDescriptors except IdPSSODescriptor and
315 SPSSODescriptor.

316

G. Revision History

317

Revision	Date	Editor	Changes Made
0.2	August 9 th 2010	Andreas Åkre Solberg	Incorporating comments from Thomas Lenggenhager and Lukas Hämmerle
0.1	July 17 th 2010	Andreas Åkre Solberg	Initial draft

318

319