



SAML V2.0 Change Notify Protocol Version 1.0

Committee Specification 01

22 September 2011

Specification URIs

This version:

<http://docs.oasis-open.org/security/saml/Post2.0/sstc-saml2-notify-protocol/v1.0/cs01/sstc-saml2-notify-protocol-v1.0-cs01.odt> (Authoritative)
<http://docs.oasis-open.org/security/saml/Post2.0/sstc-saml2-notify-protocol/v1.0/cs01/sstc-saml2-notify-protocol-v1.0-cs01.html>
<http://docs.oasis-open.org/security/saml/Post2.0/sstc-saml2-notify-protocol/v1.0/cs01/sstc-saml2-notify-protocol-v1.0-cs01.pdf>

Previous Version:

<http://docs.oasis-open.org/security/saml/Post2.0/sstc-saml2-notify-protocol/v1.0/csprd01/sstc-saml2-notify-protocol-v1.0-csprd01.odt> (Authoritative)
<http://docs.oasis-open.org/security/saml/Post2.0/sstc-saml2-notify-protocol/v1.0/csprd01/sstc-saml2-notify-protocol-v1.0-csprd01.html>
<http://docs.oasis-open.org/security/saml/Post2.0/sstc-saml2-notify-protocol/v1.0/csprd01/sstc-saml2-notify-protocol-v1.0-csprd01.pdf>

Latest Version:

<http://docs.oasis-open.org/security/saml/Post2.0/sstc-saml2-notify-protocol/v1.0/sstc-saml2-notify-protocol-v1.0.odt> (Authoritative)
<http://docs.oasis-open.org/security/saml/Post2.0/sstc-saml2-notify-protocol/v1.0/sstc-saml2-notify-protocol-v1.0.html>
<http://docs.oasis-open.org/security/saml/Post2.0/sstc-saml2-notify-protocol/v1.0/sstc-saml2-notify-protocol-v1.0.pdf>

Technical Committee:

OASIS Security Services TC

Chairs:

Thomas Hardjono (hardjono@mit.edu), M.I.T.
Nathan Klingenstein (ndk@internet2.edu), Internet2

Editors:

Phil Hunt (phil.hunt@oracle.com), Oracle, Inc.
Thinh Nguyenphu (thinh.nguyenphu@nsn.com), Nokia Siemens Networks

Additional artifacts:

- XML Schema and Examples:
<http://docs.oasis-open.org/security/saml/Post2.0/sstc-saml2-notify-protocol/v1.0/cs01/xml/>

Related work:

- This specification is related to:
- *Security Assertion Markup Language (SAML) v2.0*. OASIS Standard.
<http://docs.oasis-open.org/security/saml/v2.0/saml-2.0-os.zip>

Declared XML namespaces:

- urn:oasis:names:tc:SAML:2.0:notify

Abstract:

The SAML V2.0 Change Notify Protocol describes request and response messages for informing SAML endpoints about available changes to subjects and attributes associated with subjects.

Status:

This document was last revised or approved by the OASIS Security Services (SAML) TC on the above date. The level of approval is also listed above. Check the "Latest version" location noted above for possible later revisions of this document.

Technical Committee members should send comments on this Work Product 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/security/>.

For information on whether any patents have been disclosed that may be essential to implementing this Work Product, 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/security/ipr.php>).

Citation format:

When referencing this Work Product the following citation format should be used:

[SAML2CNP-V1.0]

SAML V2.0 Change Notify Protocol Version 1.0. 22 September 2011. OASIS Committee Specification 01. <http://docs.oasis-open.org/security/saml/Post2.0/sstc-saml2-notify-protocol/v1.0/cs01/sstc-saml2-notify-protocol-v1.0-cs01.html>

Notices

Copyright © OASIS Open 2011. 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 name "OASIS" is a trademark 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.....	6
1.1	Notation.....	6
1.2	Terminology.....	6
1.3	Normative References.....	7
1.4	Non-normative References.....	7
2	SAML V2.0 Change Notify Protocol.....	8
2.1	Required Information.....	8
2.2	Description.....	8
2.3	Assumptions.....	9
2.4	Status URIs.....	9
2.5	Protocol URIs.....	9
2.6	Element <ChangeNotifyRequest>.....	10
2.7	Notification Elements.....	11
2.7.1	Notification Element <NewSubject>.....	11
2.7.2	Notification Element <ModifySubject>.....	11
2.7.3	Notification Element <RetireSubject>.....	12
2.8	Element <ChangeNotifyResponse>.....	12
2.9	Processing Rules.....	13
3	Bindings.....	15
4	Profile.....	16
4.1	Required Information.....	16
4.2	Profile Overview.....	16
4.3	Front-Channel Examples.....	17
4.3.1	SP Initiated Change Using Web Browser SSO.....	17
4.3.1.1	Mixed Front and Back Channel Variation.....	19
4.3.2	IDP Initiated Change Using Web Browser SSO.....	19
4.4	Back-Channel Change Notification to a SAML Subject.....	21
4.5	Profile Description.....	22
4.5.1	Change Event Triggers Notifications.....	22
4.5.2	<ChangeNotifyRequest> issued to Notify Target.....	22
4.5.2.1	Notify Target Determines Action.....	22
4.5.2.2	Notify Target Responds With <ChangeNotifyResponse>.....	23
4.5.2.3	Protocol Action.....	23
5	Conformance.....	24
Appendix A.	Use Cases.....	25
A.1.	Offline/Backchannel Mode*.....	25

A.2. Browser/Synchronous Profile	26
Appendix B. Acknowledgments	27
Appendix C. Revision History	28

1 Introduction

The Change Notify Protocol is a message exchange protocol by which a service provider (e.g. web service provider, identity provider) notifies a federated service provider of changes to principals and related attributes in a federated system. After notification, the receiver of the notification is then able to take an appropriate action to effect appropriate changes to affected principals.

This message exchange protocol uses the SAML Protocols V2.0 [SAML2Core] and bindings [SAML2-Bind].

1.1 Notation

This specification uses normative text. 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:

Prefix	XML Namespace	Comments
saml:	urn:oasis:names:tc:SAML:2.0:assertion	This is the SAML V2.0 assertion namespace defined in the SAML V2.0 core specification [SAML2Core].
samlp:	urn:oasis:names:tc:SAML:2.0:protocol	This is the SAML V2.0 core protocol namespace defined in the SAML V2.0 core specification [SAML2Core].
samln:	urn:oasis:names:tc:SAML:2.0:notify	This is the new Change Notify protocol namespace defined in this document.
xs:	http://www.w3.org/2001/XMLSchema	This is the XML Schema namespace [Schema1].
xsi:	http://www.w3.org/2001/XMLSchema-instance	This is the XML Schema namespace for schema-related markup that appears in XML instances [Schema1].

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

1.2 Terminology

Notify Issuer The issuer of a change notification request is a SAML Requester. The issuer MAY be any SAML entity, including but not limited to a relying party or an identity provider.

Notify Target The target of a change notification is a SAML Responder. The responder MAY be any SAML entity, including but not limited to a relying party or an identity provider.

31 **Subject** Any principle or entity that can be referenced by a SAML Name Identifier. A
32 subject is the object about which change notifications are made.

33 **1.3 Normative References**

- 34 **[RFC2119]** S. Bradner. *Key words for use in RFCs to Indicate Requirement Levels*. IETF
35 RFC 2119, March 1997. <http://www.ietf.org/rfc/rfc2119.txt>
- 36 **[RFC2246]** T. Dierks. *The TLS Protocol Version 1.0*. IETF RFC 2246, January 1999, See
37 <http://www.ietf.org/rfc/rfc2246.txt>
- 38 **[SAML2Bind]** OASIS Standard, *Bindings for the OASIS Security Association Markup Language*
39 (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)
40 [bindings-2.0-os.pdf](http://docs.oasis-open.org/security/saml/v2.0/saml-bindings-2.0-os.pdf)
- 41 **[SAML2Core]** OASIS Standard, *Assertions and Protocols for the OASIS Security Assertion*
42 *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)
43 [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)
- 44 **[SAML2Meta]** S. Cantor et al. *Metadata for the OASIS Security Assertion Markup Language*
45 (SAML) V2.0. OASIS SSTC, March 2005. [http://docs.oasis-](http://docs.oasis-open.org/security/saml/v2.0/saml-metadata-2.0-os.pdf)
46 [open.org/security/saml/v2.0/saml-metadata-2.0-os.pdf](http://docs.oasis-open.org/security/saml/v2.0/saml-metadata-2.0-os.pdf)
- 47 **[SAML2Prof]** OASIS Standard, *Profiles for the OASIS Security Assertion Markup Language*
48 (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)
49 [profiles-2.0-os.pdf](http://docs.oasis-open.org/security/saml/v2.0/saml-profiles-2.0-os.pdf)
- 50 **[Schema1]** H. S. Thompson et al. *XML Schema Part 1: Structures*. World Wide Web
51 Consortium Recommendation, May 2001. See [http://www.w3.org/TR/2001/REC-](http://www.w3.org/TR/2001/REC-xmlschema-1-20010502/)
52 [xmlschema-1-20010502/](http://www.w3.org/TR/2001/REC-xmlschema-1-20010502/)
- 53 **[SSL3]** A. Frier et al. *The SSL 3.0 Protocol*. Netscape Communications Corp, November
54 1996.

55 **1.4 Non-normative References**

- 56 **[OpenID]** OpenID Community, OpenID Authentication 2.0, December 5, 2007.
57 http://openid.net/specs/openid-authentication-2_0.html
- 58 **[Portable]** Joseph Smarr, Plaxo, 5 August 2008. <http://portablecontacts.net/draft-spec.html>
- 59 **[RFC2251]** M. Wahl, T. Howes, S. Kille, Lightweight Directory Access Protocol (v3), IETF
60 RFC 2251, December 1997. <http://www.ietf.org/rfc/rfc2251.txt>
- 61 **[SPMLv2]** G. Cole et al. OASIS Service Provisioning Language (SPML) Version 2, 1 April
62 2006. [http://www.oasis-open.org/committees/download.php/17708/pstc-spml-2.0-](http://www.oasis-open.org/committees/download.php/17708/pstc-spml-2.0-os.zip)
63 [os.zip](http://www.oasis-open.org/committees/download.php/17708/pstc-spml-2.0-os.zip)
- 64 **[WS-Trust]** Anthony Nadalin, Marc Goodner, et. al., OASIS WS-Trust 1.3 Specification,
65 March 2007. <http://docs.oasis-open.org/ws-sx/ws-trust/200512>

66 2 SAML V2.0 Change Notify Protocol

67 2.1 Required Information

68 This section describes all of the required information of a profile as defined in section 2.1 of the Profile
69 the Profiles for the OASIS Security Assertion Markup Language (SAML) V2.0 [SAML2Prof].

70 **Identification:** urn:oasis:names:tc:SAML:2.0:notify

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

72 **Description:** Given below.

73 **Updates:** None.

74 2.2 Description

75 The SAML Change Notify Protocol is a two-step message exchange protocol by which a Notify Issuer
76 (SAML Requester) notifies a Notify Target server (SAML Responder) of changes to Subjects and related
77 attributes. The Notify Issuer and Notify Target server each MAY be a Service Provider and/or Identity Pro-
78 vider. After a change notification has been received, the Issuer and Target servers are able to negotiate
79 secondary actions to propagate changes, if appropriate, in a protocol agnostic fashion. This message ex-
80 change protocol uses the SAML Protocols V2.0 [SAML2Core] and SAML Profile specifications [SAML2-
81 Prof].

82 In typical SAML scenarios, user information is propagated through the use of the Browser SSO Profile
83 [SAML2Prof] and similar profile variants. However, except for just-in-time SSO provisioning, and for the
84 SAML Name Identifier Management Protocol [SAML2Core], there is no clear common method by which
85 federated SAML entities can inform each other of changes to user principals and attributes that occur
86 over time. Change Notify Protocol allows service providers to coordinate subject changes while maintain-
87 ing separate state and administrative control. Instead of initiating specific data change commands,
88 Change Notify Protocol simply informs service providers about changes that may be of interest.
89 Further, Change Notify Protocol allows service providers to infer more meaning information than that
90 available from existing SAML protocol features. For example, while the <Terminate> option of <Mana-
91 geNameIDRequest> is used for de-federation, Change Notify Protocol adds functionality to distinguish
92 between de-federation and a de-provisioning event. Some examples include:

- 93 • An enterprise provisioning and de-provisioning accounts to cloud service providers
- 94 • An enterprise updating employee roles and attributes persisted in the cloud
- 95 • An IDP informing RPs that retained information (e.g. from a past SAML Attribute Query) requires
96 updating.

97 There are many instances where service providers that generate identity related attributes wish to inform
98 IDPs of available changes. Some examples include:

- 99 • A service provider migrating legacy database/directory users to a federated provider
- 100 • A service provider transferring a user from one IDP to another
- 101 • A service provider generating or updating attribute data for which it is deemed authoritative

102 As part of the Change Notify request, the Notify Issuer specifies one or more protocol URIs that it wants
103 to use to facilitate transfer or management of data. Examples include:

- 104 • SAML AttributeQuery (for back-channel mode)
- 105 • SAML Web SSO (for front-channel mode)
- 106 • SPMLv2 [SPMLv2]
- 107 • PortableContacts [Portable]
- 108 • Other

109 The request also includes information on the nature of the change, the affected subjects, and affected at-
110 tributes.

111 The Notify Target responds with a Change Notify Protocol response message that indicates acknowledg-
112 ment and the chosen data transfer protocol.

113 2.3 Assumptions

114 It is assumed that the Notify Issuer and Notify Target have agreements with each other that permits the
115 exchange of attributes and extended status information between parties.

116 Such agreements might include:

- 117 • Definitions of how Change Notify Protocol operations are to be issued and interpreted by parties.
118 For example, what happens when a Notify Target receives a RetireSubject notification. Does it
119 delete the subject, disable the subject, or suspend the subject?
- 120 • Definitions of what notifications will be issued for which entities between servers.
- 121 • Definitions of how many transactions may be included in a single request-response exchange,
122 and how frequently they may occur.
- 123 • Definitions of how updates between parties impacts and supports overall subject provisioning and
124 management.
- 125 • Definitions of which protocols are to be used within specific circumstances. For example, after re-
126 ceiving notification of a large number of NewSubjects, the responder MAY wish to make a dy-
127 namic decision to use SPML instead of SAML AttributeQuery to process the subjects at a later
128 time.

129 Exact terms of such an agreement are out of scope of this specification, However, the exact interpretation
130 of the Change Notify request and response messages, processing, and profile are defined in this specific-
131 ation.

132 2.4 Status URIs

133 In addition to the Status URIs defined in [SAML2Core], the following top-level `<samlp:StatusCode>` is
134 defined related to Change Notify protocol:

135 `urn:oasis:names:tc:SAML:2.0:status:notify:protocol`

136 The request could not be performed as the protocol was unavailable at the time of the request for
137 the subjects, and/or notification elements requested.

138 2.5 Protocol URIs

139 In the protocol, the issuer and target MAY negotiate a protocol to implement changes indicated in change
140 notify requests. The protocols supported MAY include but are not limited to the following URIs:

141 `urn:oasis:names:tc:SAML:2.0:notify:protocol:SAML:BackChannel`

142 In back-channel (synchronous) mode, this URI indicates that Notify Target will query the Notify
143 Issuer for the affected SAML Identifier using SAML `AttributeQuery`. When initiated in front-
144 channel (asynchronous/mixed) mode, indicates that information will be exchanged via a back-
145 channel by using SAML `AttributeQuery`. For `<RetireSubject>` elements, indicates that SAML
146 `<ManageNameIDRequest>` will be used.

147 `urn:oasis:names:tc:SAML:2.0:notify:protocol:SAML:FrontChannel`

148 In front-channel mode (asynchronous) mode, this URI indicates that information will be
149 exchanged via the `<AuthnRequest>/<Response>` SAML protocol using the any supported
150 profile (e.g. web SSO) of the Authentication Request protocol. If target initiated, the request will
151 begin with an `<AuthnRequest>`. If initiated by the Issuer, the Issuer will simply use an
152 unsolicited `<Response>` message to transfer the user. For `<RetireSubject>` elements, no
153 further action will be taken.

154 `urn:oasis:names:tc:SAML:2.0:notify:protocol:STS`

155 Indicates that change information will be exchanged via WS-Trust protocol [WS-Trust]. Typically
156 the Target initiates WS-Trust transactions to the endpoint defined by the issuer.

157 `urn:oasis:names:tc:SAML:2.0:notify:protocol:OpenID`

158 Indicates that change information will be exchanged via OpenID protocol [OpenID]. Typically the
159 Target initiates OpenID transactions to the OpenID endpoint defined by the issuer.

160 urn:oasis:names:tc:SAML:2.0:notify:protocol:SPMLv2
 161 Indicates that change information will be exchanged via SPMLv2 protocol [SPMLv2]. Typically the
 162 issuer initiates SPML transactions to the endpoint defined by the Target.

163 urn:oasis:names:tc:SAML:2.0:notify:protocol:LDAPv3
 164 Indicates that change information will be exchanged via LDAPv3 protocol. If the Notify Issuer is
 165 declared the initiator, then the Notify Issuer will follow with one or more LDAP Add, Modify, and/or
 166 Delete operations, as defined in [RFC2251]. If the Notify Target is declared the initiator, the target
 167 will initiate action with one or more LDAP Search operations.

168 urn:oasis:names:tc:SAML:2.0:notify:protocol:PortableContact
 169 Indicates that the <saml:Subject>s will be transferred by the Notify Target using the
 170 PortableContacts specification [Portable] using the endpoint specified by the issuer.

171 urn:oasis:names:tc:SAML:2.0:notify:protocol:Other
 172 Indicates that change information will be exchanged via a protocol negotiated via end-point URIs.

173 urn:oasis:names:tc:SAML:2.0:notify:protocol:None
 174 Indicates that no transactional action will take place.

175 2.6 Element <ChangeNotifyRequest>

176 Used by a Notify Issuer to send a <ChangeNotifyRequest> message that SHALL contain one or more
 177 of the following Notification Elements: <NewSubject>, <ModifySubject>, or <RetireSubject>.

178 This <ChangeNotifyRequest> message is a complex type based on ChangeNotifyRequestType,
 179 which extends RequestAbstractType.

180 The <ChangeNotifyRequest> element allows for one or more notification elements to allow multiple
 181 change notifications to be passed in a single request message. It includes the following attributes:

182 expires [optional]

183 The time at which the notified changes expire. Default is never.

184 protocol [required]

185 The URI of a protocol that MAY be used to act or implement a change as defined in section 2.5,
 186 or any other URIs pre-negotiated between service providers.

187 endpoint [optional]

188 The URI of the Notifiers service endpoint associated with the protocol. When omitted, the
 189 endpoint is assumed to be the current endpoint of the request message issuer.

190 issuerInitiated [default=true]

191 A flag indicating whether the issuer is to initiate the action operation.

192 redirect_uri [optional]

193 An optional URI that can be used to redirect the browser to a new site following the completion of
 194 the action protocol step. For example, this option MAY be used in the front-channel to redirect the
 195 browser back to the Notifier after completion of a an operation at a Target service provider.

196 The following schema fragment defines the <ChangeNotifyRequest> protocol message:

```

197 <element name="ChangeNotifyRequest" type="saml:ChangeNotifyRequestType" />
198 <complexType name="ChangeNotifyRequestType">
199   <complexContent>
200     <extension base="saml:RequestAbstractType">
201       <sequence>
202         <choice>
203           <element name="NewSubject" type="saml:NewSubjectType" minOccurs="0"
204             maxOccurs="unbounded" />
205           <element name="ModifySubject" type="saml:ModifySubjectType"
206             minOccurs="0" maxOccurs="unbounded" />
207           <element name="RetireSubject" type="saml:ChangeSubjectType"

```

```

208         minOccurs="0" maxOccurs="unbounded" />
209             </choice>
210         </sequence>
211         <attribute name="expires" type="dateTime" use="optional"/>
212         <attribute name="protocol" type="anyURI" use="required"/>
213         <attribute name="endpoint" type="anyURI" use="optional"/>
214         <attribute name="issuerInitiated" type="boolean"
215         default="true"/>
216         <attribute name="redirect_uri" type="anyURI"
217         use="optional"/>
218     </extension>
219 </complexContent>
220 </complexType>

```

221 2.7 Notification Elements

222 Notification elements are an extension of <ChangeSubjectType> which defines a common type for de-
 223 fining changes to a particular subject entity. Notification elements <NewSubject>, <ChangeSubject>, and
 224 <RetireSubject> define the basic transaction notifications that are available in a <ChangeNoti-
 225 fyRequest>.

```

226 <complexType name="ChangeSubjectType">
227     <sequence>
228         <choice>
229             <element ref="saml:BaseID"/>
230             <element ref="saml:NameID"/>
231             <element ref="saml:EncryptedID"/>
232         </choice>
233     </sequence>
234 </complexType>

```

235 2.7.1 Notification Element <NewSubject>

236 The <NewSubject> element has the complex type <NewSubjectType>, an extension of <Change-
 237 SubjectType> which requires that one or more identifier elements <saml:NameID>, <saml:Ba-
 238 seID>, or <saml:EncryptedID> elements be provided. In addition, the Issuer MAY also include a
 239 list of one or more <saml:Attribute> elements listing the attributes available for every identifier listed
 240 within the current <NewSubject> element.

241 The purpose of this element is to allow an Issuer to notify a Target server of principals that are “new” to
 242 the Issuer.

```

243 <element name="NewSubject" type="saml:NewSubjectType" minOccurs="0"
244         maxOccurs="unbounded" />
245 <complexType name="NewSubjectType">
246     <complexContent>
247         <extension base="saml:ChangeSubjectType">
248             <sequence>
249                 <element ref="saml:Attribute"
250                     minOccurs="0" maxOccurs="unbounded" />
251             </sequence>
252         </extension>
253     </complexContent>
254 </complexType>

```

255 2.7.2 Notification Element <ModifySubject>

256 The <ModifySubject> element has the complex type <ModifySubjectType>, an extension of
 257 <ChangeSubjectType> which requires that one or more SAML Identifier elements <saml:NameID>,
 258 <saml:BaseID>, or <saml:EncryptedID> elements be provided. In addition, the Issuer MAY in-
 259 clude a list of one or more <saml:Attribute> elements listing the modified attributes for each identifier
 260 listed within the current <ModifySubject> element.

261 The purpose of this element is to allow an Issuer to notify a Target server of changes to a subject's attrib-
262 utes.

```
263 <element name="ModifySubject" type="saml:ModifySubjectType"  
264     minOccurs="0" maxOccurs="unbounded" />  
265 <complexType name="ModifySubjectType">  
266     <complexContent>  
267         <extension base="saml:ChangeSubjectType">  
268             <sequence>  
269                 <element ref="saml:Attribute" minOccurs="0"  
270                     maxOccurs="unbounded" />  
271             </sequence>  
272         </extension>  
273     </complexContent>  
274 </complexType>
```

275 **2.7.3 Notification Element <RetireSubject>**

276 The <RetireSubject> element is based on the complex type <ChangeSubjectType> and allows for
277 one or more SAML Identifier elements to be specified.

278 The purpose of this element is to allow the issuer to notify the target server that the record is to be retired
279 or de-provisioned. The exact function (e.g. deletion, disablement, suspension) of this action is typically
280 defined in a Issuer/Target service level agreement.

```
281 <element name="RetireSubject" type="saml:ChangeSubjectType"  
282     minOccurs="0" maxOccurs="unbounded" />
```

283 **2.8 Element <ChangeNotifyResponse>**

284 The recipient of the <ChangeNotifyRequest> message MUST respond with a <ChangeNotifyRe-
285 sponse> message, which is of type <saml:ChangeNotifyResponseType>.

286 The <ChangeNotifyResponse> element allows for one or more OPTIONAL notification elements to al-
287 low acknowledgment to multiple change notifications to the Notifier by the Target. It includes the following
288 attributes:

289 endpoint [optional]

290 The URI of a service endpoint for the Notify Target associated with the protocol. When omitted,
291 the endpoint is assumed to be the current endpoint of the notify responder.

292 issuerInitiated [default=true]

293 A flag confirming whether the issuer is to initiate the action operation. The value of this attribute
294 overrides the value provided in the <ChangeNotifyRequest>.

295 redirect_uri [optional]

296 An optional URI that can be used to redirect the browser to a new site following the completion of
297 the action protocol specified in the <ChangeNotifyRequest>. For example, this option MAY be
298 used in the front-channel to redirect the browser back to the Notifier after completion of a an
299 operation at a Target service provider.

300 actionAfter [optional]

301 Specifies the time at which the initiator MAY begin the specified change action protocol step.
302 Default is immediately.

303 actionDeclined [default=false]

304 Allows the Notify Target to indicate that the request has been successfully accepted but that no
305 further action is required. This attribute is typically used in connection with <RetireSubject>
306 notification elements.

307 The following schema fragment defines the <ChangeNotifyResponse> protocol message:

```
308 <element name="ChangeNotifyResponse" type="saml:ChangeNotifyResponseType" />  
309 <complexType name="ChangeNotifyResponseType">
```

```

310     <complexContent>
311         <extension base="samlp:StatusResponseType">
312             <sequence>
313                 <choice>
314                     <element name="NewSubject" type="saml:NewSubjectType" minOccurs="0"
315                         maxOccurs="unbounded" />
316                     <element name="ModifySubject" type="saml:ModifySubjectType"
317                         minOccurs="0" maxOccurs="unbounded" />
318                     <element name="RetireSubject" type="saml:ChangeSubjectType"
319                         minOccurs="0" maxOccurs="unbounded" />
320                 </choice>
321             </sequence>
322             <attribute name="endpoint" type="anyURI" use="optional"/>
323             <attribute name="issuerInitiated" type="boolean"
324                 default="true"/>
325             <attribute name="redirect_uri" type="anyURI"
326                 use="optional"/>
327             <attribute name="actionAfter" type="dateTime"
328                 use="optional"/>
329             <attribute name="actionDeclined" type="boolean"
330                 default="false" use="optional"/>
331         </extension>
332     </complexContent>
333 </complexType>

```

2.9 Processing Rules

The Notify Issuer of the <ChangeNotifyRequest> message:

- MUST include at least one change notification element (<NewSubject>, <ModifySubject>, or <RetireSubject>);
- A notification element MAY include more than one SAML Identifier;
- A separate new notification element (e.g. <ModifySubject>) MUST be used for each differing set of attributes. Multiple subjects MAY be changed in ONE notification element provided the list of attributes remain the same;
- MUST indicate the protocol to be used to facilitate the changed by providing a protocol attribute value in the form of a URI;
- The Identifiers used within the change notification elements MUST be appropriate to the protocol URI defined in the protocol attribute;
- MAY include the attribute expires is present in the element <ChangeNotifyRequest>, the availability or validity of the changes contained will be deemed to have expired on the specified date/time. If the attribute is absent, the notification information is deemed not to expire;
- When using the <RetireSubject> change notifier element, the requestor MUST either sign the <ChangeNotifyRequest> message or use a binding-specific mechanism that ensures authenticity and integrity of the message.

The responding Notify Target of the <ChangeNotifyRequest> message:

- SHOULD respond with <Status> value of urn:oasis:names:tc:SAML:2.0:status:notify:protocol if the Notify Target is unable or does not wish to proceed with the protocol defined in the <ChangeNotifyRequest> message. After receiving such a status, the Notify Issuer MAY repeat the request with a new protocol;
- MAY include endpoint attribute which specifies the service endpoint for the Notify Target associated with the specified protocol;
- MAY include <NewSubject>, <ModifySubject>, <RetireSubject> sub-elements to indicate the processing action SHALL be restricted to only those NameID(s) specified in the notify sub-elements. If <NewSubject>, <ModifySubject>, <RetireSubject> sub-elements are not included, then the Notify Target is indicating that all changes will be process as per the original <ChangeNotifyRequest> message.

- 364
- 365
- 366
- 367
- MAY include `<saml:Attribute>` elements within the `<NewSubject>` or `<ModifySubject>` elements, to indicate the processing SHALL be restricted to the specified `<saml:Attribute>`s in a subsequent action. If `<saml:Attribute>` elements are not provided, the responder is indicating that the attributes specified in the `<ChangeNotifyRequest>` message SHALL be used;
- 368
- 369
- 370
- MAY include the attribute `actionAfter` to indicate to the Notify Issuer that action operations SHOULD begin on or after the date/time specified. If the attribute is absent, it is assumed that the responder intends action to begin immediately;
- 371
- 372
- 373
- MAY include the attribute `actionDeclined` to indicate to the Notify Issuer that no further action is required (e.g. as a result of receiving `<ReturnSubject>` notifications) and does not indicate an error condition;
- 374
- 375
- 376
- If the Notify Target does not recognize the `<ChangeNotifyRequest>`, the Notify Target MUST respond to the Notify Issuer with `<ChangeNotifyResponse>` with `<status>` of `urn:oasis:names:tc:SAML:2.0:status:Responder`.

377 **3 Bindings**

378 Mappings of the SAML Change Notify Protocol request-response message exchanges onto standard
379 messaging or communications protocols follow the core SAML Protocol Bindings specifications (saml-
380 bindings-2.0-os) [SAML2Bind].

381 4 Profile

382 The Change Notify Protocol has one universal profile that can be used in both front-channel and back-
383 channel modes and can be used in conjunction with other SAML Profiles such as the Web Browser SSO
384 Profile [SAML2Prof]. In front-channel mode, an “issuer site” (known as Issuer) MAY notify a “target site”
385 (Target) of a new or changed, or retired subject profile related to the currently authenticated subject. In
386 back-channel mode, a Notifier can notify a Target of several changes about subjects in “batch” mode. Fi-
387 nally, a mix mode is supported whereby a front-channel notification MAY be combined with a back-chan-
388 nel transfer of information (e.g. using SAML AttributeQuery). The Change Notify Protocol is used in con-
389 junction with HTTP Redirect, and HTTP Post.

390 4.1 Required Information

391 This section describes all of the required information of a profile as defined in section 2.1 of the Profile
392 the Profiles for the OASIS Security Assertion Markup Language (SAML) V2.0 [SAML2Prof].

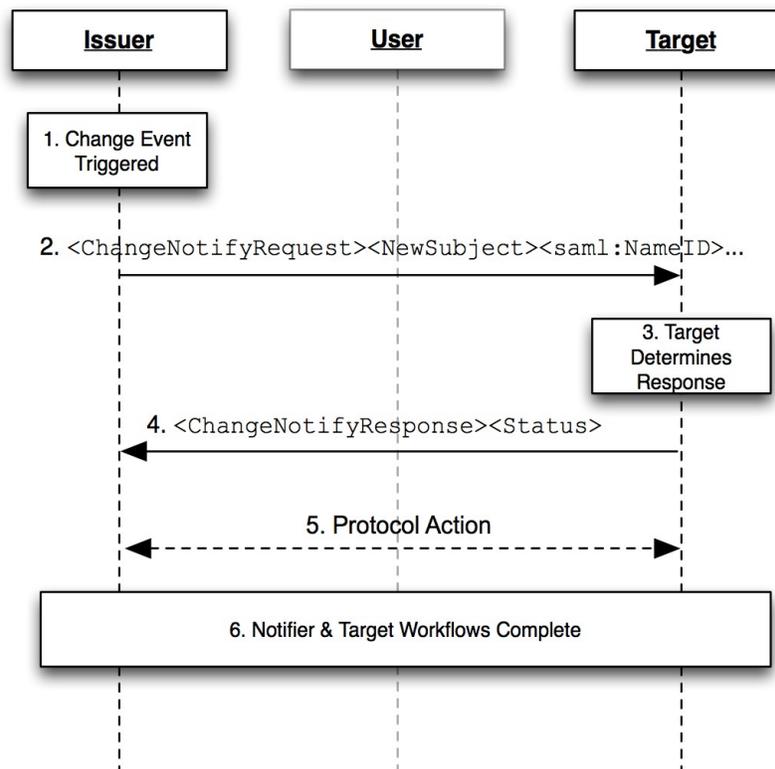
393 **Identification:** urn:oasis:names:tc:SAML:2.0:profiles:notify

394 **Contact Information:** security-services-comment@lists.oasis-open.org

395 **Description:** See below.

396 4.2 Profile Overview

397 In the Change Notify profile, a `<ChangeNotifyRequest>` is issued by a SAML Requester (known as
398 Notify Issuer) providing one or more changes impacting one or more subjects. The SAML Responder
399 (known as Notify Target) signals its agreement to exchange information in a subsequent step, known as
400 the action protocol step by responding with a `<ChangeNotifyResponse>` message. Following the pro-
401 tocol exchange, the requestor and responder begin an exchange of information using the protocol indic-
402 ated in the original `<ChangeNotifyRequest>`.



403

Figure 1: Base Change Notify Profile

404 The grayed-out user illustrates that the message exchange may pass through a user agent or may be a
405 direct exchange between notification entities (Issuer and Target), depending on the binding used to imple-
406 ment the profile.

407 The following steps are described by the profile. Within each step, there MAY be variation on the actual
408 message exchanges depending on the binding used for that step, and the subsequent protocol selected
409 for transfer of information between Notification parties.

410 Change Notify protocol flow is intended to allow an Issuer and Target to coordinate updates to entities of
411 common interest. Change Notify Protocol enables the Notifier to communicate changes that it believes to
412 be of interest without having to know the state of data within the Target. On receiving a change notifica-
413 tion, the Target is able to determine how to proceed and to place the change notification in a context that
414 makes sense within its service “domain”.

415 **1. Change Event Triggered**

416 A workflow event triggers the Notify Issuer node to determine that there is a change of interest to
417 a Notify Target server. An event can consist of one or more changes to one or more subjects.

418 **2. <ChangeNotifyRequest> issued by Notify Issuer**

419 The Notify node, takes the set of changes and forms a request by including one or more change
420 notify elements. As part of the request, the Notifier MUST indicate the protocol to be used in step
421 5, and which party is to initiate the step.

422 **3. Target Determines Response**

423 The Target server receives the change notification and determines how to process the incoming
424 change given its knowledge of the current state of potentially affected entities in its domain.

425 **4. Target Responds with <ChangeNotifyResponse>**

426 The Target issues a response containing either no notifications, or listing only those notification
427 elements and subject identifiers with which it wishes to proceed with. The Target also confirms
428 when processing time is to begin. The Target MAY also indicate that no further processing is
429 required by setting the attribute `actionDeclined`, or it MAY indicate a desire to change
430 protocols by responding with a <Status> of

431 `urn:oasis:names:tc:SAML:2.0:status:notify:protocol`

432 **5. Protocol Action**

433 Based on the protocol URI supplied in the <ChangeNotifyRequest> and the value of the
434 attribute `issuerInitiated`, the endpoints proceed to exchange information using an SAML 2
435 protocol, or by using another protocol. Note that the exact process for this exchange is out of
436 scope for this specification.

437 **6. Notifier & Target Workflow Completion**

438 Based on the selected protocol and the value of `redirect_uri` attribute, the endpoints
439 complete their processing and for front-channel cases, the user-agent is redirected appropriately.

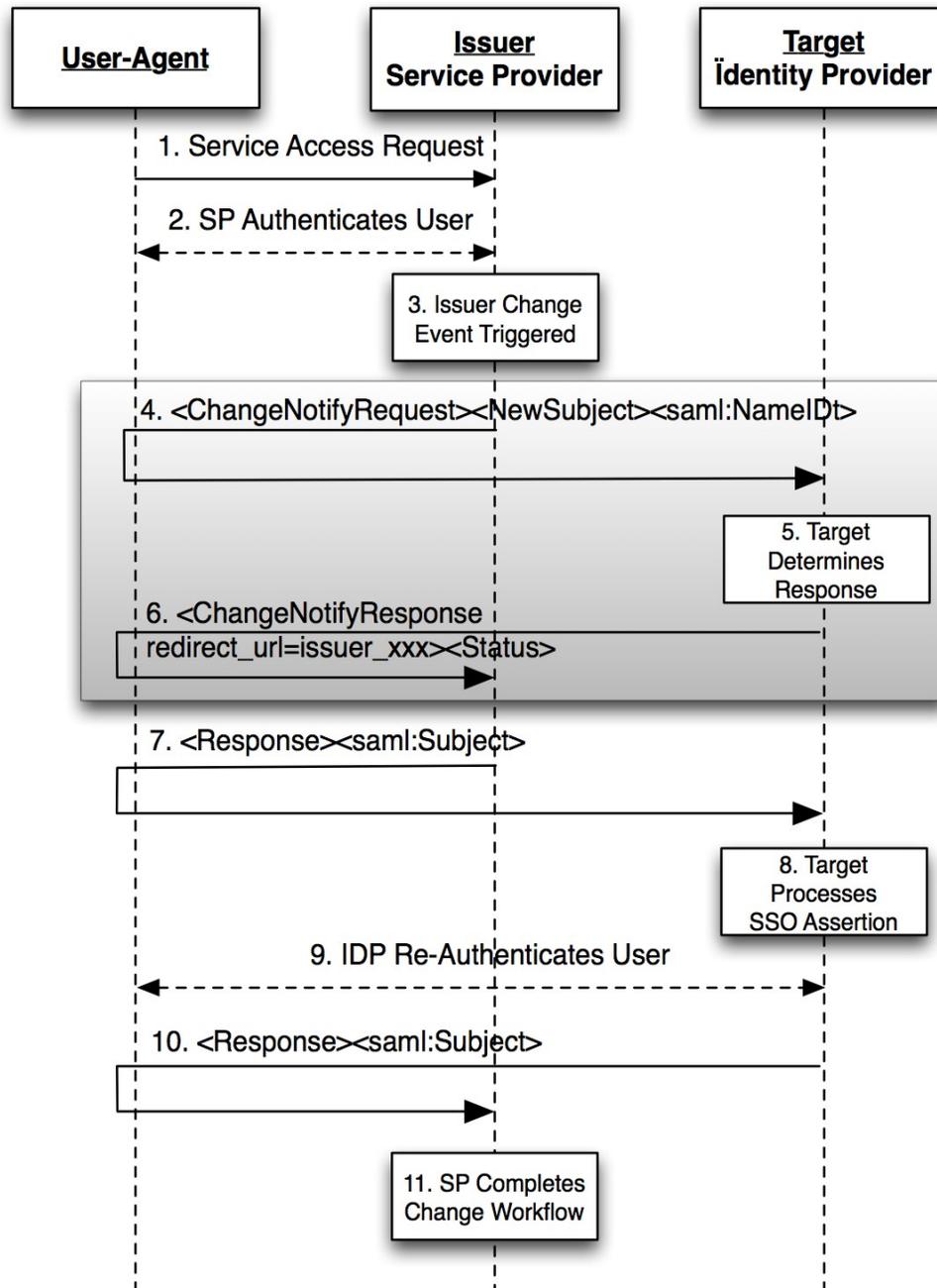
440 **4.3 Front-Channel Examples**

441 **4.3.1 SP Initiated Change Using Web Browser SSO**

442 This example demonstrates a web service provider transferring a signed on user context to an IDP for the
443 purpose of provisioning a user to the IDP. In this situation, it assumed, though not guaranteed, that the
444 SP is already familiar with the user, while the IDP likely does not have a pre-existing relationship with the
445 user. The effect is to allow the SP to provide a “warm-introduction” of the user to the IDP.

446 The following figure illustrates an example of transferring a subject from a Service Provider acting as a
447 Notify Issuer server to a Notify Target server (acting as an Identity Provider) using Web SSO to achieve
448 the transfer of attributes and to maintain authentication state between the parties. The service provider is-
449 sue a <ChangeNotifyRequest> notification request to the identify provider to add this user as a new
450 subject. Once the Change Notify protocol followed by the action protocol step are completed, the service

451 provider resumes the Web SSO authentication request, per the normal Web SSO Profile allowing the
 452 user to access a resource at the service provider using a SSO from the IDP.



453 *Figure 2: Change Notify Web SSO at Service Provider*

- 454 1) The user makes request for a secure resource at the service provider without security context;
 455 possibly triggering a provisioning workflow.
- 456 2) If not already performed, the SP authenticates the user, via local or other federated means.
- 457 3) Notify Issuer (service provider) interprets a locally generated change event and determines a Tar-
 458 get (identity provider) interested in potentially receiving the notification.
- 459 4) Notify Issuer (service provider) sends an <ChangeNotifyRequest> with <NewSubject> noti-
 460 fication element (or the notification MAY also be a <ModifySubject> or <RetireSubject>

461 element) including a `<saml:NameID>`. The Notifier, sets the attributes `issuerInitiated` to
462 `true`, and the `protocol` attribute to: `urn:oasis:names:tc:SAML:2.0:notify:pro-`
463 `ocol:SAML:FrontChannel`

464 Notify Target (IdP) processes the notification request and accepts the request.

465 5) In response, Notify Target send a `<ChangeNotifyResponse>`, to the Notify Issuer.

466 6) In response to the `protocol` and `issuerInitiated` attributes of the `<ChangeNotifyRe-`
467 `quest>`, the Notify Issuer initiates the protocol step by issuing an unsolicited `<samlp:re-`
468 `sponse>` to the Notify Target endpoint, thereby facilitating the new subject transfer and including
469 the user's SSO context.

470 Note: Step 4-6 are the procedures from Change Notify Protocol.

471 7) The Notify Target processes the inbound SSO SAML Assertion and provisions the new subject as
472 appropriate.

473 8) Optionally, the Notify Target MAY choose to re-authenticate the user within its own administrative
474 domain.

475 9) The Notify Target uses the value of `redirect_uri` passed in the initial `<ChangeNotifyRe-`
476 `quest>` to pass the user-agent back to the Notify Issuer, including a web SSO assertion from the
477 Identity Provider.

478 10) The Notify Issuer is now able to proceed with any final event workflow requirements (e.g. local
479 de-provisioning).

480 **4.3.1.1 Mixed Front and Back Channel Variation**

481 In a mixed channel variation, a front-channel notification is transmitted via the browser while SAML As-
482 sertion data is transferred in a back-channel. The intention here is to provide greater workflow flexibility
483 between providers.

484 In the previous example, the `protocol` URI in step 4 is set to `urn:oasis:names:tc:SAML:2.0:no-`
485 `tify:protocol:SAML:BackChannel`, while `issuerInitiated` is set to `false`. The effect would be
486 to cause step 7 to be replaced with a back-channel SAML Attribute Query initiated by the Notify Target in-
487 stead of an Unsolicited SAML Response from the Notify Issuer in step 7.

488 **4.3.2 IDP Initiated Change Using Web Browser SSO**

489 Figure 3 shows a user initially accessing an Identity Provider site action which triggers a change for a tar-
490 get Service Provider. This triggers the `<ChangeNotifyRequest>` to the Service Provider. Once the
491 Change Notify with the Action Protocol procedures are completed, the Identity Provider sends unsolicited
492 `<response>`, per the SAML Web SSO Profile [SAML2Prof]. Note that the grayed block area shows the
493 Change Notification protocol portion of the overall exchange sequence.

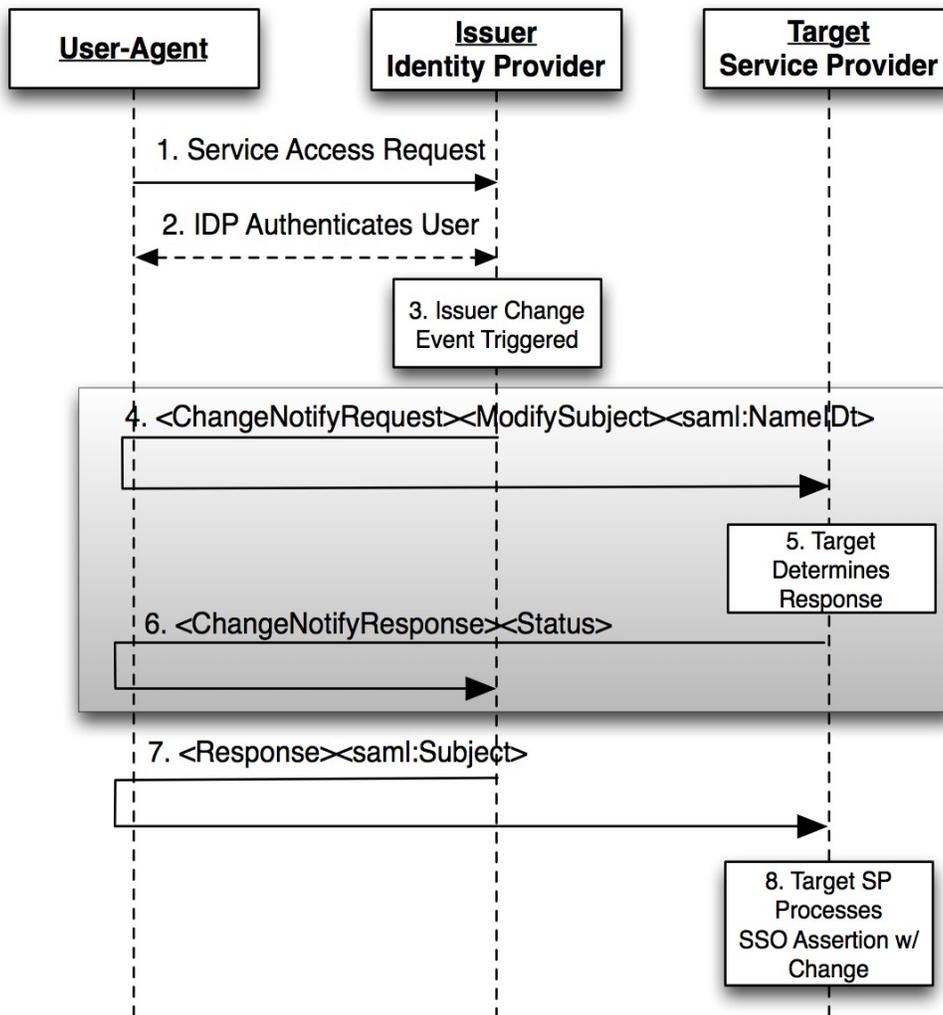


Figure 3: Web SSO Initiated Change at IdP

494

495 1) The user makes request for a secure resource at the Notify Issuer (Identity Provider) requiring au-
496 thentication.

497 2) The Notify Issuer (Identity Provider) authenticates the user.

498 3) Notify Issuer (Identity Provider) determines a change notification is required along with an “Unso-
499 licited” Web SSO Profile [SAML2Prof].

500 4) Notify Issuer (Identity Provider) sends an <ChangeNotifyRequest> with a <ModifySubject>
501 notification element (which MAY also be a <NewSubject> or <RetireSubject> element) and
502 SAML Name Identifier, the attribute protocol set to urn:oasis:names:tc:SAML:2.0:notify:protocol:SAML:FrontChannel with issuerInitiated set to true to the Notify Tar-
503 get (Service Provider), and a list of available SAML Attributes (except in the case of <Retire-
504 Subject> notification element).
505

506 5) Notify Target (Service Provider) process the request and accepts the notification request.

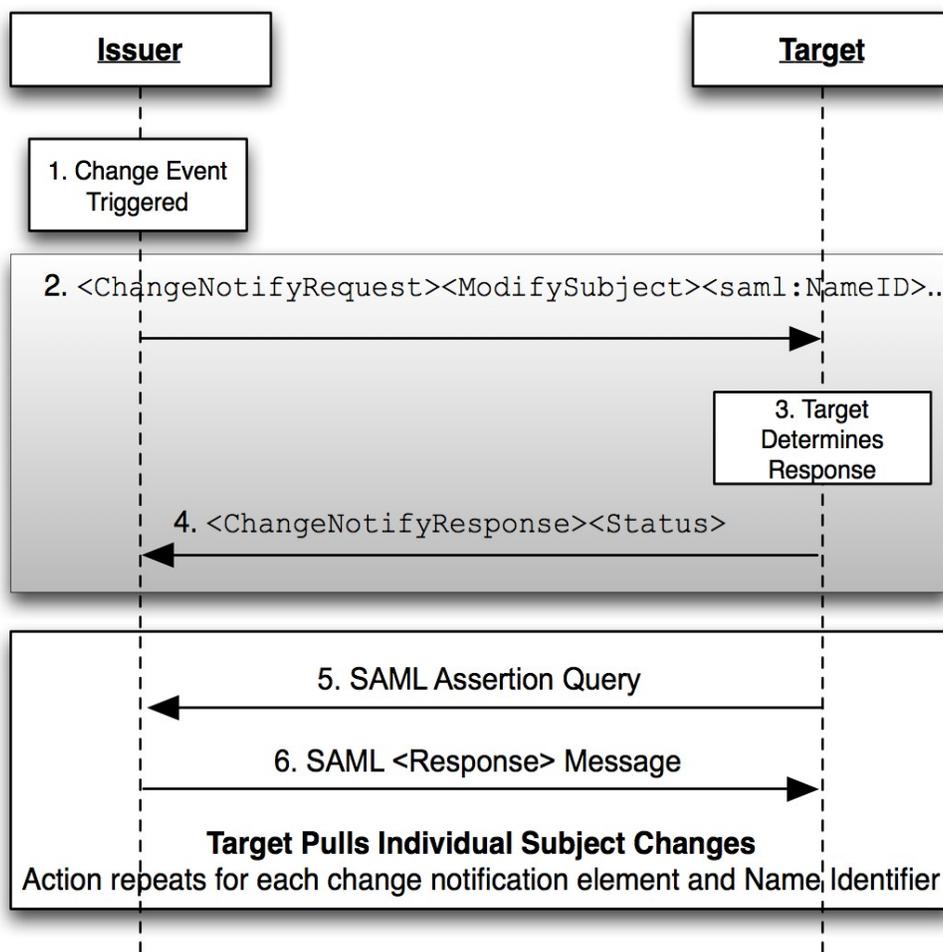
507 6) Notify Target sends an <ChangeNotifyResponse> to the Notify Issuer, with an accepted list of
508 SAML Attributes.

509 7) According to the protocol attribute defined in the original <ChangeNotifyRequest>, the Notify
510 Issuer completes the transaction by issuing an unsolicited SAML <Response> containing a
511 SAML <Subject> to the Notify Target endpoint, including the accepted list of SAML <Attrib-
512 ution> value assertions.

513 8) Based on the SAML <Response> message, the service provider processes the SSO assertion
514 containing the notified changes.

515 4.4 Back-Channel Change Notification to a SAML Subject

516 Figure 4 shows an update being propagated from a Notify Issuer to a Notify Target using a back-channel.
517 The grey-box shows the Change Notify Protocol while the second box shows how the payload for each
518 change MAY be exchanged using the SAML Assertion Query/Request profile [SAML2Prof].
519 For the purpose of this example, a Notify Issuer or Target MAY be any SAML endpoint such as a Service
520 Provider or Identity Provider.



521 *Figure 4: Back-Channel Change Using SAML Assertion Query*

- 522 1) The Notify Issuer (Identity Provider) determines a change has occurred that SHOULD be shared
523 with a particular target.
- 524 2) Notify Issuer sends an <ChangeNotifyRequest> with one or more notification elements
525 (<ModifySubject> is shown) along with one or more SAML Name Identifiers, the attribute
526 protocol set to urn:oasis:names:tc:SAML:2.0:notify:protocol:SAML:BackChan-
527 nel with issuerInitiated set to false to the Notify Target. For each notification elements, a
528 list of available SAML Attributes (except in the case of <RetireSubject> notification element).
- 529 3) Notify Target processes the request and accepts the notification request.
- 530 4) Notify Target sends an <ChangeNotifyResponse> to the Notify Issuer, with an accepted list of
531 SAML Attributes.

- 532 5) According to the protocol attribute defined in the original `<ChangeNotifyRequest>`, the Notify
533 Target completes the action phase of the notification by issuing SAML Assertion Queries accord-
534 ing to the SAML Assertion Query Profile [SAML2Prof]. A new query is issued for each `<NewSub-`
535 `ject>` or `<ModifySubject>` element and name identifier received in the change notify request.
- 536 6) As per the SAML Assertion Query/Response Profile, the Notify Issuer responds to each request
537 and returns a SAML `<Response>` completing the transfer of subject changes described in the
538 original `<ChangeNotifyRequest>`.

539 4.5 Profile Description

540 4.5.1 Change Event Triggers Notifications

541 An event occurs, either triggered directly by a user, workflow, or backend process, that causes a Notify Is-
542 suser to determine there is a change of interest to a particular Notify Target.

543 4.5.2 `<ChangeNotifyRequest>` issued to Notify Target

544 To initiate the profile, the Notify Issuer issues a `<ChangeNotifyRequest>` message to a target service
545 provider known as a Notify Target. Metadata (as in [SAML2Meta]) MAY be used to determine the location
546 of this endpoint and the bindings supported by the responding provider.

547 Synchronous Binding (Back-Channel)

548 The Notify Issuer MAY use a synchronous binding, such as the SOAP binding [SAML2Bind], to send
549 the request directly to the Notify Target provider. The requestor MUST authenticate itself to the other
550 provider, either by signing the `<ChangeNotifyRequest>` or using any other binding-supported
551 mechanism.

552 Asynchronous Binding (Front-Channel)

553 Alternatively, the Notify Issuer MAY (if the principal's user agent is present) use an asynchronous
554 binding, such as the HTTP Redirect, or POST [SAML2Bind] to send the request to the other provider
555 through the user agent.

556 It is RECOMMENDED that the HTTP exchanges in this step be made over either SSL 3.0 [SSL3] or
557 TLS 1.0 [RFC2246] to maintain confidentiality and message integrity. The `<ChangeNotifyRe-`
558 `quest>` message MUST be signed.

559 Each of these bindings provide a RelayState mechanism that the Notify Issuer MAY use to associate
560 the subsequent exchanges with the original request. The Notify Issuer SHOULD reveal as little in-
561 formation as possible in the RelayState value unless the use of profile does not require such privacy
562 measures.

563 The Notify Issuer server sends a `<ChangeNotifyRequest>`, and MUST include the attribute protocol
564 specifying the protocol to be used for the action step. The attribute `issuerInitiated` is defaulted to
565 true. If a different service will issue the action in 4.1.3.4, the Issuer SHALL include the endpoint of the
566 server issuing the SSO assertion.

567 In the case of `<NewSubject>`, or `<ModifySubject>`, the `<ChangeNotifyRequest>` MUST include
568 one of the notification type elements: `<NewSubject>`, or `<ModifySubject>`. Within the notification type
569 element is contained one identifier element `<saml:NameID>`, `<saml:BaseID>`, or `<saml:En-`
570 `crypteID>`. If the notification element is `<NewSubject>` or `<ModifySubject>` transaction, it MAY in-
571 clude one or more SAML Attribute names. No data is transferred.

572 In the case of `<RetireSubject>`, the `<ChangeNotifyRequest>` MUST include one of the notifica-
573 tion type elements: `<RetireSubject>`, MUST includes one identifier element `<saml:NameID>`,
574 `<saml:BaseID>`, or `<saml:EncryptedID>`, MUST one or more SAML Attribute names and MUST
575 NOT include attribute data.

576 4.5.2.1 Notify Target Determines Action

577 The Notify Target service provider, on receiving the `<ChangeNotifyRequest>` determines the internal
578 action it wishes to take regarding the request. The Target evaluates the notification and the `protocol` at-
579 tribute included in the request and prepares the server to handle any subsequent action protocol step.
580 This MAY include queuing and/or recording of transaction information such as Subject Identifiers trans-
581 ferred in the `<ChangeNotifyRequest>` message.

582 **4.5.2.2 Notify Target Responds With <ChangeNotifyResponse>**

583 The Notify Target, the recipient, MUST process the <ChangeNotifyRequest> as defined in section 2.9
584 Processing Rules. After processing the message or upon encountering an error, the Notify Target MUST
585 issue a <ChangeNotifyResponse> containing an appropriate status code to the requesting provider
586 (Notify Issuer) to complete the protocol exchange.

587 **Synchronous Bindings (Back-Channel)**

588 If the Notify Issuer used a synchronous binding, such as the SOAP binding [SAML2Bind], the re-
589 sponse is returned directly to complete the synchronous communication. The responder MUST au-
590 thenticate itself to the requesting provider, either by signing the <ChangeNotifyResponse> or us-
591 ing any other binding-supported mechanism.

592 **Asynchronous Bindings (Front-Channel)**

593 If the Notify Issuer used an asynchronous binding, such as the HTTP Redirect, or POST bindings
594 [SAML2Bind], then the <ChangeNotifyResponse> is returned through the user agent to the Notify
595 Issuer's endpoint. Metadata (as in [SAML2Meta]) MAY be used to determine the location of the end-
596 point and the bindings supported by the requesting provider (Notify Issuer). Any binding supported by
597 both entities MAY be used.

598 If the HTTP Redirect or POST binding is used, then the <ChangeNotifyResponse> message is de-
599 livered to the Notify Issuer (requesting provider) in this step.

600 It is RECOMMENDED that the HTTP exchanges in this step be made over either SSL 3.0 [SSL3] or
601 TLS 1.0 [RFC2246] to maintain confidentiality and message integrity. The <ChangeNotifyRe-
602 sponse> message MUST be signed.

603 The exact format of this HTTP response and the subsequent HTTP request to the assertion consumer
604 service is defined by the SAML binding used. Profile-specific rules on the contents of the <ChangeNoti-
605 fyResponse> are included in Section 2.8 and Section 2.9.

606 In the case of <NewSubject>, or <ModifySubject>, the <ChangeNotifyResponse> MAY includes
607 a different endpoint to receive the action protocol response by specifying it in the `endpoint` attribute.

608 If the Notify Target wishes to take no action due to error, the Target MUST issue a status response of
609 `urn:oasis:names:tc:SAML:2.0:status:Responder` to indicate an error condition. If the Notify
610 Target wishes to indicate a non-error status result but that no further action is necessary, the responder
611 SHOULD include the attribute `actionDeclined` with a value of `true`.

612 **4.5.2.3 Protocol Action**

613 After successful exchange of a <ChangeNotifyRequest> followed by a <ChangeNotifyResponse>, the
614 end points SHALL execute an exchange of information using the appropriate protocol and endpoints ne-
615 gotiated in the message exchange and per the processing rules of section 2.9.

616 The protocol used is defined by the attribute `protocol` and the entity initiating the exchange is deter-
617 mined by the attribute `issuerInitiated`. The protocol action step MAY be delayed until the date spe-
618 cified by the attribute `actionAfter`, or MAY be declined entirely if the responder sets the attribute `ac-`
619 `tionDeclined` to `true`.

620 The protocol used to transfer information SHOULD have security measures equivalent to or superior to
621 those specified in this binding to protect the confidentiality and message integrity of data transferred.

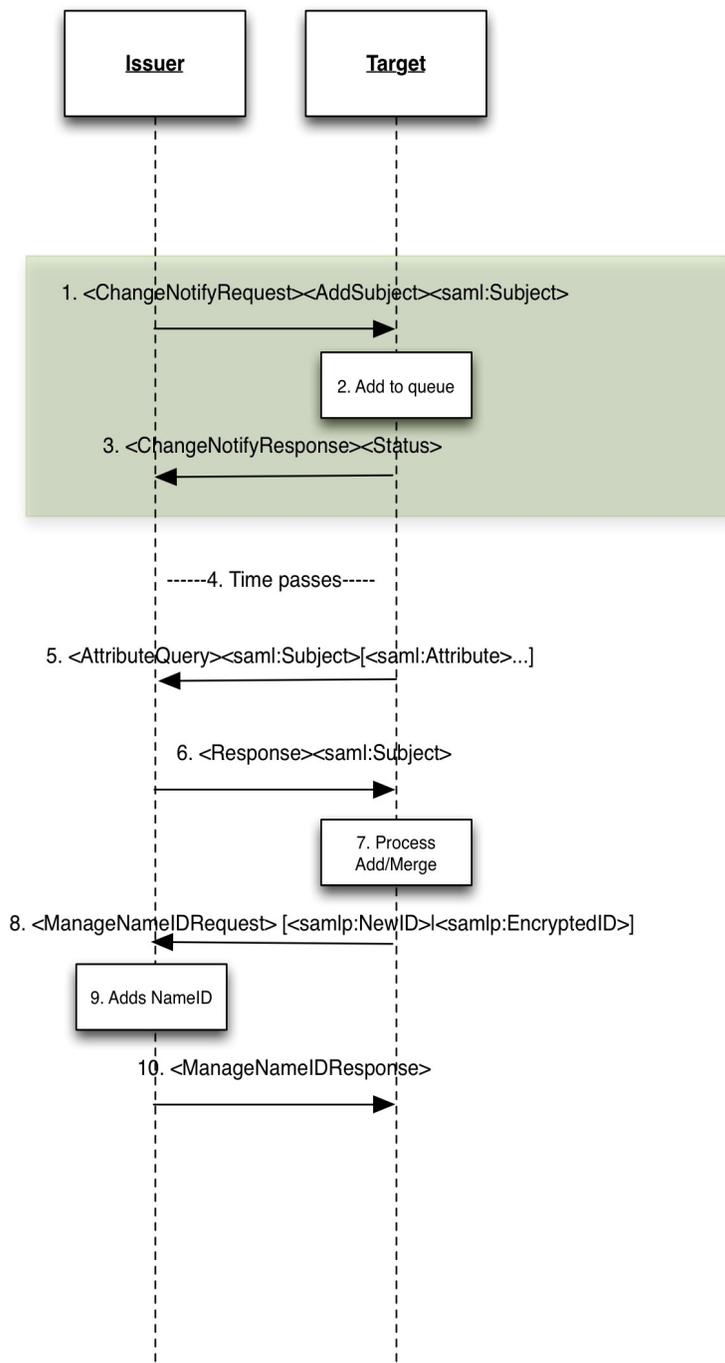
622 5 Conformance

- 623 Conformance Notify Issuers and Notify Targets SHOULD implement the Change Notify profile using the
624 HTTP Post, and HTTP redirect bindings.
- 625 Informational: Where appropriate, Notify Issuers and Notify Targets SHOULD have agreements in place
626 to define how action protocols will be implemented and used.
- 627 A service provider wishing to issue ChangeNotifyRequests, MUST support the protocols necessary to fa-
628 cilitate configured action protocol. An service provider using SAML as an action protocol MUST support
629 SAML Attribute Authority and SAML Authentication Authority functionality for the purpose of fulfilling
630 SAML action steps as described in the profile.
- 631 A Notify Issuer can claim to support Change Notify Protocol if it can issue `<ChangeNotifyRequest>`s,
632 respond to `<ChangeNotifyResponse>`s, and can support the use of at least ONE action protocol to fa-
633 cilitate transfer of change data to the Notify Target's designated protocol endpoint.
- 634 A Notify Target can claim to support Change Notify Protocol if it can respond to `<ChangeNotifyRe-`
635 `quest>`s, issue `<ChangeNotifyResponse>`s, and can support the use of at least ONE action pro-
636 tocol to support the transfer of change data from the Notify Issuer's designated protocol endpoint.
- 637 A Notify Issuer and Notify Target claiming to support Change Notify Protocol in the front-channel MUST
638 also be able to support the Web SSO Profile [SAML2Prof] bi-directionally.

639 Appendix A. Use Cases

640 An issuer notifies a target of new information that is available. The target MAY then request the data via
 641 either an AttributeQuery or an AuthnRequest in the case of the browser profile.

642 A.1. Offline/Backchannel Mode*:



643 1. The issuer notifies the target of some updated information regarding a particular subject. In this
 644 case an add subject indicates that the issuer believes this subject is new to the target (which may

645 or may not be true). The assertion only includes the issuers nameidentifier. The issuer can indic-
646 ate multiple requests in the same message. The issuer MAY indicate what attributes are available
647 in the message.

648 2. The target receives the request and either adds it to its queue processing (immediate or delayed).
649 The target MAY also choose to ignore the request, but MUST acknowledge the receipt of the re-
650 quest (step 3).

651 3. The target acknowledges the request. The target MAY indicate OK, or indicate declined. A re-
652 sponse of OK does not oblige the target to do anything further.

653 4. The target MAY optionally delay processing (the process is asynchronous)

654 5. The target issues an attributeQuery for each nameidentifier supplied by the issuer. If no attributes
655 are named, the attributes provided SHALL be the ones indicated in step 1, or all attributes as per
656 the normal AttributeQuery processing. OR, if arranged by prior agreement, the target MAY use a
657 different protocol to effect transfer (e.g SPML, OpenID, etc).

658 6. Issuer responds with the attributes requested.

659 7. The target MAY optionally update the issuer with its local name identifier depending on the rela-
660 tionship between issuer and target.

661 Note: for the purpose of this profile, issuer or target end-points can refer to either SP or IDP. E.g. An SP
662 notifying an IDP of a new user transfer, or an IDP notifying an SP of a new user (e.g. Employee in an en-
663 terprise IDP).

664 **A.2. Browser/Synchronous Profile**

665 In the synchronous mode, information transfer is accomplished via browser SSO. This MAY be useful in
666 cases where SSO transfer of context is desirable.

667 1. The issuer notifies the target of some updated information regarding a particular subject. In this
668 case an <NewSubject> indicates that the issuer believes this subject is new to the target (which
669 may or may not be true). The assertion only includes the issuer's name identifier. The issuer can
670 indicate multiple requests in the same message. The issuer MAY indicate what attributes are
671 available in the message.

672 2. The target receives the request and determines what it wants to do (e.g. process as add, modify,
673 or ignore). The target MAY also choose to ignore the request, but MUST acknowledge the receipt
674 of the request by issuing a <ChangeNotifyResponse> response.

675 **Appendix B. Acknowledgments**

676 The editor would like to acknowledge the contributions of the OASIS Security Services (SAML) Technical
677 Committee, whose voting members at the time of publication were:

- 678 • Rob Philpott, EMC Corporation
- 679 • Bob Morgan, Internet2
- 680 • Scott Cantor, Internet2
- 681 • Nathan Klingenstein, Internet2
- 682 • Chad La Joie, Internet2
- 683 • Thomas Hardjono, M.I.T.
- 684 • Frederick Hirsch, Nokia Corporation
- 685 • Thinh Nguyenphu, Nokia Siemens Networks GmbH & Co. KG
- 686 • Ari Kermaier, Oracle Corporation
- 687 • Hal Lockhart, Oracle Corporation
- 688 • Emily Xu, Oracle Corporation
- 689 • Anil Saldhana, Red Hat
- 690 • David Staggs, Veterans Health Administration

691 The editor would also like to acknowledge the contribution of an earlier draft from NSN entitled: "SAML
692 V2.0Attributes Management Protocol Version 1.0 Working Draft 06 November 2009", upon which this doc-
693 ument attempts to incorporate supporting requirements from.

694 **Appendix C. Revision History**

Document ID	Date	Committer	Comment
sstc-saml2-notify-protocol-01	07/19/10	Phil Hunt Thinh Nguyenphu	Initial draft
sstc-saml2-notify-protocol-02	09/17/10	Phil Hunt Thinh Nguyenphu	Editorial clean ups, saml:Subject changed to NameID etc
sstc-saml2-notify-protocol-03	10/01/10	Thinh Nguyenphu Phil Hunt	Updates to Profiles adding two overview flows
sstc-saml2-notify-protocol-04	10/21/10	Phil Hunt Thinh Nguyenphu	Removed ActionProtocol Element Completed profiles
sstc-saml2-notify-protocol-v1.0-wd05	5 May 2011	Thinh Nguyenphu	Editorial cleanup based on 30 days public review comments from Chapman Martin