SAML Metadata Extension for a Standalone Attribute Requester

Working Draft 01, 11 March 2005

Document identifier:

draft-saml-metadata-ext-01

Location:

http://www.oasis-open.org/committees/security

Editors:

Tom Scavo (trscavo@gmail.com), Individual

Scott Cantor (cantor.2@osu.edu), The Ohio State University

Contributors:

Abstract:

This specification defines an extension to the SAML 2.0 metadata specification ([SAML2Meta]). The extension defines a role descriptor that describes a standalone SAML 1.x or 2.0 attribute requester, that is, an attribute requester not bound to a SAML SSO profile. Readers should be familiar with [SAML2Meta] before reading this document.

This is a **working draft** and the text may change before completion.

Table of Contents

1 Introduction

This specification defines an extension to the SAML 2.0 metadata specification. The extension defines a role descriptor that describes a standalone SAML attribute requester, that is, an attribute requester not bound to a SAML SSO profile. The profile addresses both SAML 1.x and SAML 2.0.

Unless specifically noted, nothing in this document should be taken to conflict with the SAML 2.0 metadata specification ([SAML2Meta]). Readers are advised to familiarize themselves with that specification first.

1.1 Notation

This specification uses normative text to define an extension to the SAML 2.0 metadata specification.

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

> …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 `saml:` stands for the SAML 2.0 assertion namespace, `urn:oasis:names:tc:SAML:2.0:assertion`
- The prefix `md:` stands for the SAML 2.0 metadata namespace, `urn:oasis:names:tc:SAML:2.0:metadata`
- The prefix `mdext:` stands for the SAML 2.0 metadata extension namespace developed herein, `urn:oasis:names:tc:SAML:metadata:extension`
- The prefix `ds:` stands for the W3C XML Signature namespace, `http://www.w3.org/2000/09/xmldsig#`
- The prefix `xsd:` stands for the W3C XML Schema namespace, `http://www.w3.org/2001/XMLSchema` In schema listings, this is the default namespace and no prefix is shown.
- The prefix `xsi:` stands for the W3C XML Schema namespace for schema-related markup that appears in XML instances: `http://www.w3.org/2001/XMLSchema-instance`

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

1.2 Motivating Use Case

A primary SAML use case is browser SSO, but several non-browser use cases are emerging that incoporate a standalone attribute requester ([SAMLX509], [GridShib], [LionShare]). Such a role is not

supported by [SAML2Meta]. This specification defines a new role descriptor type designed to support a typical non-browser scenario.

A SAML metadata extension that supports this use case is described in section 2. Relevant references are listed in section 3.

2 Metadata Extension for SAML 2.0

This section defines a new role descriptor type that supports the non-browser use case described in section 1.

2.1 Namespaces

[SAML2Meta] defines the following namespace [SAML2Meta-xsd]:

```
urn:oasis:names:tc:SAML:2.0:metadata
```

By convention, the namespace prefix `md:` is used to refer to the above namespace.

This specification defines a new namespace:

```
urn:oasis:names:tc:SAML:metadata:extension
```

We use prefix `mdext:` to refer to this new namespace. In what follows, any unqualified type is assumed to belong to this new namespace.

2.2 Element <md:RoleDescriptor>

The `<md:RoleDescriptor>` element defined in [SAML2Meta] is an abstract extension point that contains descriptive information common across various entity roles. New roles can be defined by extending its abstract **md:RoleDescriptorType** complex type, which is the approach taken here.

2.3 Complex Type AttributeRequesterDescriptorType

Complex type **AttributeRequesterDescriptorType** extends complex type **md:RoleDescriptorType** with content specific to attribute requesters, that is, consumers of SAML attributes. The type **AttributeRequesterDescriptorType** contains the following additional attributes and elements:

`WantAssertionsSigned` [Optional]

> Optional attribute that indicates a requirement for assertions received by this service provider to be signed. If omitted, the value is assumed to be `false`. This requirement is in addition to any requirement for signing derived from the use of a particular profile/binding combination.

`<md:NameIDFormat>` [Zero or More]

> Zero or more elements of type **xsd:anyURI** that enumerate the name identifier formats supported by this service provider. See section 8.3 of [SAML2Core] for some possible values of this element.

`<md:AttributeConsumingService>` [Zero or More]

> Zero or more elements that describe an application or service provided by this service provider that requires or desires the use of SAML attributes. It is RECOMMENDED that deployers provide at least one such element to facilitate configuration of policy by atribute providers.

At most one `<md:AttributeConsumingService>` element can have the attribute `isDefault` set to `true`. When multiple elements are specified and none has the attribute `isDefault` set to `true`, then the first element whose `isDefault` attribute is not set to `false` is to be used as the default. If all elements have their `isDefault` attribute set to `false`, then the first element is considered the default.

Instances of **AttributeRequesterDescriptorType** are declared using the `<md:RoleDescriptor>` element with an `xsi:type` of **AttributeRequesterDescriptorType**. See the example in section 2.4.

See [SAML1xMeta] for specifics on the transformation and use of particular elements and attributes for use with SAML 1.x.

The following schema fragment defines the **AttributeRequesterDescriptorType** complex type:

```
<complexType name="AttributeRequesterDescriptorType">
  <complexContent>
    <extension base="md:RoleDescriptorType">
      <sequence>
        <element ref="md:NameIDFormat" minOccurs="0" maxOccurs="unbounded"/>
        <element ref="md:AttributeConsumingService" minOccurs="0"
maxOccurs="unbounded"/>
      </sequence>
      <attribute name="WantAssertionsSigned" type="boolean" use="optional"/>
    </extension>
  </complexContent>
</complexType>
```

2.4 Example

Here is a metadata example for a SAML attribute requester that supports both SAML 1.1 and SAML 2.0:

```
<md:EntityDescriptor
  xmlns:md="urn:oasis:names:tc:SAML:2.0:metadata"
  xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion"
  xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  entityID="https://gs.org/gridshib">
  <!-- insert ds:Signature element here -->
  <md:RoleDescriptor
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:mdext="urn:oasis:names:tc:SAML:2.0:metadata:extension"
    xsi:type="mdext:AttributeRequesterDescriptorType"
    protocolSupportEnumeration="urn:oasis:names:tc:SAML:1.1:protocol
urn:oasis:names:tc:SAML:2.0:protocol">
    <md:KeyDescriptor use="signing">
      <ds:KeyInfo>
        <ds:KeyName>Requester Key</ds:KeyName>
      </ds:KeyInfo>
    </md:KeyDescriptor>
    <md:NameIDFormat>
      urn:oasis:names:tc:SAML:1.1:nameid-format:X509SubjectName
    </md:NameIDFormat>
    <md:AttributeConsumingService isDefault="true" index="0">
      <md:ServiceName xml:lang="en">
        Shibbolized Grid Service
      </md:ServiceName>
      <md:RequestedAttribute
        NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri"
        Name="urn:oid:1.3.6.1.4.1.5923.1.1.1.7"
        FriendlyName="eduPersonEntitlement">
        <saml:AttributeValue xsi:type="xsd:anyURI">
          https://gs.org/gridshib/entitlements/123456789
        </saml:AttributeValue>
      </md:RequestedAttribute>
      <md:RequestedAttribute
        NameFormat="urn:mace:shibboleth:1.0:attributeNamespace:uri"
        Name="urn:mace:dir:attribute-def:eduPersonEntitlement">
        <saml:AttributeValue xsi:type="xsd:anyURI">
          https://gs.org/gridshib/entitlements/123456789
        </saml:AttributeValue>
      </md:RequestedAttribute>
    </md:AttributeConsumingService>
  </md:RoleDescriptor>
  <md:Organization>
    <md:OrganizationName xml:lang="en">
      GridShib Service Provider
    </md:OrganizationName>
    <md:OrganizationDisplayName xml:lang="en">
```

```
      GridShib Service Provider @ Some Location
    </md:OrganizationDisplayName>
    <md:OrganizationURL xml:lang="en">
      http://www.gs.org/
    </md:OrganizationURL>
  </md:Organization>
  <md:ContactPerson contactType="technical">
    <md:SurName>GridShib Support</md:SurName>
    <md:EmailAddress>gridshib-support@gs.org</md:EmailAddress>
  </md:ContactPerson>
</md:EntityDescriptor>
```

3 References

The following works are referenced directly or indirectly in the body of this specification.

3.1 Normative References

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

[RFC 2396] T. Berners-Lee et al. *Uniform Resource Identifiers (URI): Generic Syntax.* IETF RFC 2396, August 1998. http://www.ietf.org/rfc/rfc2396.txt.

[SAMLBind] E. Maler et al. *Bindings and Profiles for the OASIS Security Assertion Markup Language (SAML) V1.1*. OASIS, September 2003. Document ID oasis-sstc-saml-bindings-1.1. http://www.oasis-open.org/committees/security/.

[SAMLCore] E. Maler et al. *Assertions and Protocols for the OASIS Security Assertion Markup Language (SAML) V1.1*. OASIS, September 2003. Document ID oasis-sstc-saml-core-1.1. http://www.oasis-open.org/committees/security/.

[SAML2Core] S. Cantor et al., *Assertions and Protocols for the OASIS Security Assertion Markup Language (SAML) V2.0*. OASIS, January 2005. Document ID sstc-saml-core-2.0-cd-04. See http://www.oasis-open.org/committees/security/.

[SAML2Gloss] J. Hodges et al., *Glossary for the OASIS Security Assertion Markup Language (SAML) V2.0*. OASIS, January 2005. Document ID sstc-saml-glossary-2.0-cd-04. See http://www.oasis-open.org/committees/security/.

[SAML2Meta] S. Cantor et al., *Metadata for the OASIS Security Assertion Markup Language (SAML) V2.0*. OASIS, January 2005. Document ID sstc-saml-metadata-2.0-cd-04. See http://www.oasis-open.org/committees/security/.

[SAML2Meta-xsd] S. Cantor et al., SAML metadata schema V2.0 CD-04. OASIS, January 2005. Document ID sstc-saml-schema-metadata-2.0. See http://www.oasis-open.org/committees/security/.

[SAML2Prof] S. Cantor et al., *Profiles for the OASIS Security Assertion Markup Language (SAML) V2.0*. OASIS, January 2005. Document ID sstc-saml-profiles-2.0-cd-04. See http://www.oasis-open.org/committees/security/.

[SAML1xMeta] G. Whitehead and S. Cantor, *SAML 1.x Metadata Profile.* OASIS, March 2005. Document ID draft-saml1x-metadata-05. See http://www.oasis-open.org/committees/security/.

[SAMLX509] R. Randall, SAML X.509 *Authentication-based Attribute Sharing Profile.* OASIS, February 2005. Document ID sstc-saml-x509-authn-based-attribute-protocol-profile-2.0-draft-02. See http://www.oasis-open.org/committees/security/.

3.2 Non-Normative References

[GridShib] *GridShib: A Policy Controlled Attribute Framework.* See http://grid.ncsa.uiuc.edu/GridShib/.

[LionShare] *LionShare Peer-to-Peer File Sharing.* See http://lionshare.its.psu.edu/main/.