AttributeNaming

SAML Attribute Names

Every attribute must have its own unique representation in a SAML attribute assertion to ensure that there are no misinterpretations or communication failures. Thus SAML exchanges rely on consistent attribute naming to deliver information about users in a mutually understood way between the IdP and SP. This name must be expected and handled by relying parties. Values, vocabularies, and their meaning should be discussed as well, but are outside the scope of this document.

Much of the power of federated authentication is derived from the economies of scale accomplished by large numbers of providers speaking a lingua franca. Attributes are the language in which access control and release policies are written and are the piece of the infrastructure for which avoiding unnecessary proliferation of names is most important. Standards bodies have traditionally defined common attribute names and semantics (e.g., X.520, eduPerson, etc.) for LDAP and other information repositories. Some attributes have XML representations as well. Federations can also serve as centers for attribute convergence.

The names for attributes in back-end data stores and consuming applications is decoupled from the expression of attributes on the wire, and it's possible to name an attribute differently for every protocol. The mapping from data stores to SAML at the identity provider is performed using `attribute-resolver.xml`. At the service provider, these attributes are then made available to the web server and web applications using `attribute-map.xml`.

An existing eduPerson attribute

```
<saml:Attribute FriendlyName="telephoneNumber" Name="urn:oid:2.5.4.20"
    NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri">
    <saml:AttributeValue xmlns:xs="http://www.w3.org/2001/XMLSchema"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xsi:type="xs:string">555-5555</saml:AttributeValue>
</saml:Attribute>
```

SAML Naming Conventions

While software such as Shibboleth supports any `NameFormat` you'd like, it's strongly recommended that URIs be used for attribute naming because of the uniqueness and namespace control they provide.

The following steps should be followed when naming a new attribute:

1. Is this attribute standardized or defined by an organization that has already assigned it a unique identifier? If so, the existing name should be used if at all possible.
2. If the attribute is defined through an LDAP object class, there is probably already an OID assigned. When possible, leverage the existing urn:oid namespace (see below).
3. If no suitable name yet exists for this attribute, consider creating one by constructing a proper URL (see below). Another option is to use a delegated `urn:mace` namespace (see below).

OID Naming

The SAML V2.0 LDAP/X.500 Attribute Profile specifies that X.500/LDAP attributes be named by utilizing the `urn:oid` namespace. These names are simply constructed using the string `urn:oid` followed by the OID defined for the attribute. For example, a DN should be expressed as `urn:oid:1.3.6.1.4.1.1466.115.121.1.12`.

URL Naming

The recommended way of naming SAML attributes that don't have a corresponding OID is with a URL. The creation and meaning of URLs is generally well understood by many people, and the DNS namespace is already extremely structured. Be sure to define new URLs only in domains you control and do your part to prevent attribute proliferation.

To create a URL name for an attribute, design a URL to be used as the identifier. If this attribute will be shared by a community, consider a URL that is common to that community, such as `https://supervillain.edu/attributes/evilPersonUniqueID` for a campus-wide identifier.

URL attribute names may resolve into documentation, providing helpful information for deployers, but that's strictly optional. URI/URL naming is a widespread mechanism in XML and related standards and it is irrelevant to software whether a URL used as a name actually represents a real resource.

MACE URNs

The urn:mace namespace is a controlled namespace that is registered with the IETF and IANA for MACE working groups and organizations it works with. The namespace is intended to be delegated to individual organizations through registration with MACE. Once a subspace of `urn:mace` has been delegated to another organization (e.g., `urn:mace:switch.ch`) that organization is responsible for any naming and resolution within that subspace. However, it is **not** permissible to arbitrarily define new attributes within the `urn:mace` namespace, or in any subtree, unless you have been granted permission to do so by MACE.

Use this form to request a `urn:mace` namespace.