Data Minimization in Practice

It is well known that a service provider can request attributes by reference using the SAML AuthnRequest Protocol. Usually the reference is to an `<md:AttributeConsumingService>` element in metadata, but as discussed in a recent blog post on data minimization, metadata is not required. The identity provider may map the reference to an attribute bundle in whatever way makes sense.

Below we describe a **non**-use case for the `<md:AttributeConsumingService>` element in the InCommon Federation. In this case, the service provider doesn't request attributes at all. Instead, the identity provider releases a predefined bundle of attributes to **all** service providers in a class of service providers called an *Entity Category*.

There is a class of service providers called the Research and Scholarship Category of service providers (SPs). Associated with the Research and Scholarship (R&S) Category is an attribute bundle B. An identity provider (IdP) supports the R&S Category if it automatically releases attribute bundle B to every SP in the R&S Category.

Every SP in the R&S Category is required to have an `<md:AttributeConsumingService>` element in its metadata. Now suppose an SP encodes attribute bundle A in metadata such that A is a subset of B. In other words, the SP requests fewer attributes than what the IdP has agreed to release. In this case, an IdP can choose to release A and still support the R&S Category.

Although the software supports this behavior, few (if any) IdPs are configured this way. Instead, IdPs release B to all SPs regardless of the attributes called out in SP metadata. Why? Well, it's easier for the IdP to release B across the board but there's another more important reason that depends on the nature of attribute bundle, so let me list the attributes in the bundle along with some sample values:

> *eduPersonPrincipalName*:trscavo@internet2.edu
> *mail*:trscavo@internet2.edu
> *displayName*:Tom Scavo OR (*givenName*:Tom AND *sn*:Scavo)

Note that all of the attributes in this bundle are name-based attributes. If you're going to release one, you may as well release all of them since each attribute valu encodes essentially the same information. This is why IdPs choose to release the entire bundle across the board: there are no privacy benefits in releasing a strict subset, and so it's simply easier to release all attributes to all R&S SPs.

I'm sure one could come up with a hypothetical use case for which there are real privacy benefits in releasing subset A of bundle B but we haven't bumped into such a use case yet. In any event, note the following:

- AFAIK, no software implementation supports more than one `<md:AttributeConsumingService>` element in metadata so there isn't much point in calling out the index of such an element in the `<samlp:AuthnRequest>`.
- Use of the `AttributeConsumingServiceIndex` XML attribute as described in the blog post is interesting, but entity attributes give the same effect, and moreover, entity attributes are in widespread use today (at least in higher ed).
- I doubt any IdP in the InCommon Federation would be inclined to implement a liberal attribute release policy such as "release whatever attributes are called out in the `<md:AttributeConsumingService>` element in metadata" since this is a potentially serious privacy leak.

This leads to the following prediction: the `<md:AttributeConsumingService>` element in metadata and the `AttributeConsumingServiceIndex` XML attribute in the `<samlp:AuthnRequest>` will turn out to be historical curiosities in the SAML protocol. At this point, the best approach to attribute release appears to be the *Entity Category* (of which the R&S Category is an example).