

SubjectDerivedAttributeAttributeDefinition

The `SubjectDerivedAttribute`^{3.3} attribute definition exposes attributes values derived from the Java Subject(s) produced by the authentication flow (s) used to authenticate the subject of the profile request.

A configuration shortcut allows for the values from any `IdPAttribute` objects contained inside `IdPAttributePrincipal` objects to be pulled out, which is an effective way to tunnel attribute data from outside the IdP provided by the `External` authentication flow.

Schema Name and Location

This `xsi:type` is defined by the `urn:mace:shibboleth:2.0:resolver` namespace, the schema for which is located at <http://shibboleth.net/schema/idp/shibboleth-attribute-resolver.xsd>

Attributes

Any of the [common attributes](#) can be specified. Note that this attribute definition does **not** require a `sourceAttributeID` attribute since the information is **not** resolved from a dependent attribute. If one is supplied, it is ignored.

Additionally **exactly** one of the following must be provided (but not both):

Name	Type	Description
<code>principalAttributeName</code>	string	The name of an attribute found inside an <code>IdPAttributePrincipal</code> contained in one of the authenticated Subject(s)
<code>attributeValuesFunctionRef</code>	Bean reference	The name of a Spring Bean implementing <code>Function<Principal, List<IdPAttributeValue>></code> , this function will be invoked for each Principal found within the authenticated Subject(s)

Child Elements

Any of the [common child elements](#) can be specified. Note that this attribute definition does **not** require a `<Dependency>` child element since the information is **not** resolved from a dependent attribute. If any are supplied, then they are ignored.

Example

```
<AttributeDefinition xsi:type="SubjectDerivedAttribute" id="PD1" principalAttributeName="Whatever" >
```