

# Grouper and Shibboleth Integration

|                           |                                               |                                |                                          |                                         |                                              |
|---------------------------|-----------------------------------------------|--------------------------------|------------------------------------------|-----------------------------------------|----------------------------------------------|
| <a href="#">Wiki Home</a> | <a href="#">Grouper Release Announcements</a> | <a href="#">Grouper Guides</a> | <a href="#">Grouper Deployment Guide</a> | <a href="#">Community Contributions</a> | <a href="#">Internal Developer Resources</a> |
|---------------------------|-----------------------------------------------|--------------------------------|------------------------------------------|-----------------------------------------|----------------------------------------------|

## Overview



### Notice

If you are working with Grouper 2.1 or above, see the [newer documentation on Grouper and Shib Integration](#)

As of v1.5, the Grouper API distribution, grouper.jar, provides a [Data Connector Extension](#) and [Attribute Definition](#) Extensions to the Shibboleth Attribute Resolver.

The namespace and schema location are:

```
<AttributeResolver xmlns="urn:mace:shibboleth:2.0:resolver"
  xmlns:grouper="http://grouper.internet2.edu/shibboleth/2.0"
  xsi:schemaLocation="http://grouper.internet2.edu/shibboleth/2.0 classpath:/schema/shibboleth-2.0-grouper.xsd"
  ...
```

These were chosen as part of the design for the Grouper PSP. However, they also offer a new means of including Grouper information in Shibboleth-based SAML attribute assertions.

Sites interesting in integrating these new capabilities into their Shibboleth IdP are advised to conduct extensive testing prior to implementing in a production environment.

## Installation into your Shibboleth Identity Provider



### Warning

This is NOT the recommended way to integrate with your Shibboleth Identity Provider

To install the Grouper DataConnector into you need to copy all of the grouper jars into the /lib directory of your shibboleth installation. Then run install.sh. Next, you'll need to have your grouper configuration files including grouper.properties and subject.xml placed into /opt/shibboleth-idp/conf. You should then be able to edit your attribute-resolver.xml as above and it should be able to get the necessary attributes.

## Grouper Data Connectors

### Group Data Connector

The GroupDataConnector returns attributes which represent a Grouper Group.

#### GroupDataConnector - Attributes

By default, all attributes (default and custom) of a group are returned by the GroupDataConnector. The names of default attributes are defined in the [Group er Glossary](#) : *id*, *name*, *displayName*, *extension*, *displayExtension*, and *description*.

The following example will return an attribute named "description" whose value is the description of a group :

```
<resolver:DataConnector id="GroupDataConnector" xsi:type="grouper:GroupDataConnector" />

<resolver:AttributeDefinition id="description" xsi:type="ad:Simple">
  <resolver:Dependency ref="GroupDataConnector" />
</resolver:AttributeDefinition>
```

#### GroupDataConnector - Lists

By default, no lists are returned by the GroupDataConnector because they may be expensive to query. Lists which should be returned as attributes may be defined using the following naming convention :

```
<resolver:DataConnector id="GroupDataConnector" xsi:type="grouper:GroupDataConnector">
  <grouper:Attribute id="<members|group>[:<all|immediate|effective|composite>[:<list name>]]" />
</resolver:DataConnector>
```

## Default List

The following example will return an attribute named "member" whose values are the "name" of every Member of the default "members" list of a group :

```
<resolver:DataConnector id="GroupDataConnector" xsi:type="grouper:GroupDataConnector">
  <grouper:Attribute id="members" />
</resolver:DataConnector>

<resolver:AttributeDefinition id="member" xsi:type="grouper:Member" sourceAttributeID="members" >
  <resolver:Dependency ref="GroupDataConnector" />
  <grouper:Attribute id="name" source="jdbc" />
</resolver:AttributeDefinition>
```

## List Scope

The following example will return an attribute named "immediateMembers" whose values are the "name" of every immediate Member of the default "members" list of a group :

```
<resolver:DataConnector id="GroupDataConnector" xsi:type="grouper:GroupDataConnector">
  <grouper:Attribute id="members:immediate" />
</resolver:DataConnector>

<resolver:AttributeDefinition id="immediateMembers" xsi:type="grouper:Member" sourceAttributeID="members:immediate" >
  <resolver:Dependency ref="GroupDataConnector" />
  <grouper:Attribute id="name" source="jdbc" />
</resolver:AttributeDefinition>
```

## Custom List

The following example will return an attribute named "customMembers" whose values are the "name" of every Member of the "customList" list of a group :

```
<resolver:DataConnector id="GroupDataConnector" xsi:type="grouper:GroupDataConnector">
  <grouper:Attribute id="members:all:customList" />
</resolver:DataConnector>

<resolver:AttributeDefinition id="customMembers" xsi:type="grouper:Member" sourceAttributeID="members:all:customList" >
  <resolver:Dependency ref="GroupDataConnector" />
  <grouper:Attribute id="name" source="jdbc" />
</resolver:AttributeDefinition>
```

## Member Of List

The following example will return an attribute named "isMemberOf" whose values are the "name" of every Group of which the group is a member of :

```
<resolver:DataConnector id="GroupDataConnector" xsi:type="grouper:GroupDataConnector">
  <grouper:Attribute id="groups" />
</resolver:DataConnector>

<resolver:AttributeDefinition id="isMemberOf" xsi:type="grouper:Group" sourceAttributeID="groups" >
  <resolver:Dependency ref="GroupDataConnector" />
  <grouper:Attribute id="name" />
</resolver:AttributeDefinition>
```

## GroupDataConnector - Privileges

Attributes representing Subjects which have Access Privileges to a group may be defined by privilege name as defined in the [Grouper Glossary](#).

```
<resolver:DataConnector id="GroupDataConnector" xsi:type="grouper:GroupDataConnector">
  <grouper:Attribute id="admins" />
  <grouper:Attribute id="optins" />
  <grouper:Attribute id="optouts" />
  <grouper:Attribute id="readers" />
  <grouper:Attribute id="updaters" />
  <grouper:Attribute id="viewers" />
</resolver:DataConnector>
```

The following example will return an attribute named "admin" whose values are the "name" of every Subject which has the ADMIN privilege on a group :

```
<resolver:DataConnector id="GroupDataConnector" xsi:type="grouper:GroupDataConnector">
  <grouper:Attribute id="admins" />
</resolver:DataConnector>

<resolver:AttributeDefinition id="admin" xsi:type="grouper:Subject" sourceAttributeID="admins" >
  <resolver:Dependency ref="GroupDataConnector" />
  <grouper:Attribute id="name" source="jdbc" />
</resolver:AttributeDefinition>
```

## Member Data Connector

The MemberDataConnector returns attributes which represent a Grouper Member. The attributes, lists, and privileges to be returned must be defined.

```
<resolver:DataConnector id="MemberDataConnector" xsi:type="grouper:MemberDataConnector">
  <grouper:Attribute id="name" source="jdbc" />
  <grouper:Attribute id="description" source="jdbc" />
  <grouper:Attribute id="groups" />
  <grouper:Attribute id="admins" />
</resolver:DataConnector>
```

### Member Data Connector - Attributes

The following example will return an attribute named "name" whose value is the name of a Member :

```
<resolver:DataConnector id="MemberDataConnector" xsi:type="grouper:MemberDataConnector" >
  <grouper:Attribute id="name" source="jdbc" />
</resolver:DataConnector>

<resolver:AttributeDefinition id="name" xsi:type="ad:Simple">
  <resolver:Dependency ref="MemberDataConnector" />
</resolver:AttributeDefinition>
```

### Member Data Connector - Lists

The following example will return an attribute named "isMemberOf" whose values are the "name" of every Group to which the Member is a member of the default "members" list :

```
<resolver:DataConnector id="MemberDataConnector" xsi:type="grouper:MemberDataConnector">
  <grouper:Attribute id="groups" />
</resolver:DataConnector>

<resolver:AttributeDefinition id="isMemberOf" xsi:type="grouper:Group" sourceAttributeID="groups" >
  <resolver:Dependency ref="MemberDataConnector" />
  <grouper:Attribute id="name" />
</resolver:AttributeDefinition>
```

### Member Data Connector - Privileges

Attributes representing Groups to which a Member's subject has Access Privileges may be defined by privilege name as defined in the [Grouper Glossary](#).

```
<resolver:DataConnector id="MemberDataConnector" xsi:type="grouper:MemberDataConnector">
  <grouper:Attribute id="admins" />
  <grouper:Attribute id="optins" />
  <grouper:Attribute id="optouts" />
  <grouper:Attribute id="readers" />
  <grouper:Attribute id="updaters" />
  <grouper:Attribute id="viewers" />
</resolver:DataConnector>
```

The following example will return an attribute named "admin" whose values are the "name" of every Group to which the Member's subject has the ADMIN privilege :

```
<resolver:DataConnector id="MemberDataConnector" xsi:type="grouper:MemberDataConnector">
  <grouper:Attribute id="admins" />
</resolver:DataConnector>

<resolver:AttributeDefinition id="admin" xsi:type="grouper:Group" sourceAttributeID="admins" >
  <resolver:Dependency ref="MemberDataConnector" />
  <grouper:Attribute id="name" />
</resolver:AttributeDefinition>
```

## Stem Data Connector

The StemDataConnector returns stems from Groupers.

```
<resolver:DataConnector id="StemDataConnector" xsi:type="grouper:StemDataConnector" />
```

## Group Filters

The subset of Groups to be returned by the GroupDataConnector or memberships returned by the MemberDataConnector may be filtered.

```
<resolver:DataConnector id="GroupDataConnector" xsi:type="grouper:GroupDataConnector">
  <grouper:GroupFilter xsi:type="grouper:Minus">
    <grouper:GroupFilter xsi:type="grouper:StemName" name="um:manual" scope="SUB" />
    <grouper:GroupFilter xsi:type="grouper:ExactAttribute" name="GROUP.status" value="NO_PROVISIONING" />
  </grouper:GroupFilter>
</resolver:DataConnector>
```

## ExactAttributeGroupFilter

The ExactAttributeGroupFilter returns groups which possess an exact attribute value :

```
<resolver:DataConnector id="testFilterExactAttribute" xsi:type="grouper:GroupDataConnector">
  <grouper:GroupFilter xsi:type="grouper:ExactAttribute" name="name" value="stem:group_name" />
</resolver:DataConnector>
```

## StemNameGroupFilter

The StemNameGroupFilter returns groups which are children of the named stem with the given scope :

```
<resolver:DataConnector id="StemNameFilterONE" xsi:type="grouper:GroupDataConnector">
  <grouper:GroupFilter xsi:type="grouper:StemName" name="parentStem" scope="ONE" />
</resolver:DataConnector>

<resolver:DataConnector id="StemNameFilterSUB" xsi:type="grouper:GroupDataConnector">
  <grouper:GroupFilter xsi:type="grouper:StemName" name="parentStem" scope="SUB" />
</resolver:DataConnector>
```

## AndGroupFilter

The AndGroupFilter returns groups which match two group filters, e.g. an Intersection :

```
<resolver:DataConnector id="AndFilter" xsi:type="grouper:GroupDataConnector">
  <grouper:GroupFilter xsi:type="grouper:AND">
    <grouper:GroupFilter xsi:type="grouper:ExactAttribute" name="name" value="parentStem:group_name" />
    <grouper:GroupFilter xsi:type="grouper:StemName" name="parentStem" scope="ONE" />
  </grouper:GroupFilter>
</resolver:DataConnector>
```

## OrGroupFilter

The OrGroupFilter returns groups which match either of two group filters, e.g. a Union :

```
<resolver:DataConnector id="OrFilter" xsi:type="grouper:GroupDataConnector">
  <grouper:GroupFilter xsi:type="grouper:OR">
    <grouper:GroupFilter xsi:type="grouper:ExactAttribute" name="name" value="parentStem:group_name" />
    <grouper:GroupFilter xsi:type="grouper:StemName" name="parentStem:childStem" scope="ONE" />
  </grouper:GroupFilter>
</resolver:DataConnector>
```

## MinusGroupFilter

The MinusGroupFilter returns groups which match the result of the first group filter minus the result of the second group filter, e.g. the Complement :

```
<resolver:DataConnector id="MinusFilter" xsi:type="grouper:GroupDataConnector">
  <grouper:GroupFilter xsi:type="grouper:Minus">
    <grouper:GroupFilter xsi:type="grouper:StemName" name="parentStem" scope="ONE" />
    <grouper:GroupFilter xsi:type="grouper:ExactAttribute" name="name" value="parentStem:group_name" />
  </grouper:GroupFilter>
</resolver:DataConnector>
```

## Attribute Definition

### Group Attribute Definition

The Grouper GroupAttributeDefinition creates an attribute whose values are the attribute values of every Group.

For example, the following "isMemberOf" attribute will have values consisting of the "name" of every Group :

```
<resolver:AttributeDefinition id="isMemberOf" xsi:type="grouper:Group" sourceAttributeID="groups" >
  <resolver:Dependency ref="GroupDataConnector" />
  <grouper:Attribute id="name" />
</resolver:AttributeDefinition>
```

### Member Attribute Definition

The Grouper MemberAttributeDefinition creates an attribute whose values are the subject attribute values of every Member.

For example, the following "member" attribute will have values consisting of the "name" attribute of every Member whose subject is from the "jdbc" source :

```
<resolver:AttributeDefinition id="member" xsi:type="grouper:Member" sourceAttributeID="members" >
  <resolver:Dependency ref="GroupDataConnector" />
  <grouper:Attribute id="name" source="jdbc" />
</resolver:AttributeDefinition>
```

### Subject Attribute Definition

The Grouper SubjectAttributeDefinition creates an attribute whose values are attribute values of every Subject.

For example, the following "owner" attribute will have values consisting of the "name" attribute of every Subject from the "jdbc" source :

```
<resolver:AttributeDefinition id="owner" xsi:type="grouper:Subject" sourceAttributeID="members" >  
  <resolver:Dependency ref="GroupDataConnector" />  
  <grouper:Attribute id="name" source="jdbc" />  
</resolver:AttributeDefinition>
```

## See Also

[Exposing Groups Through Shibboleth](#)