Attribut-Management in Shibboleth

Wolfgang Pempe, DFN-Verein
pempe@dfn.de

DFN-AAI IdP-Workshop,
24./25. Juni 2015, HS Amberg-Weiden

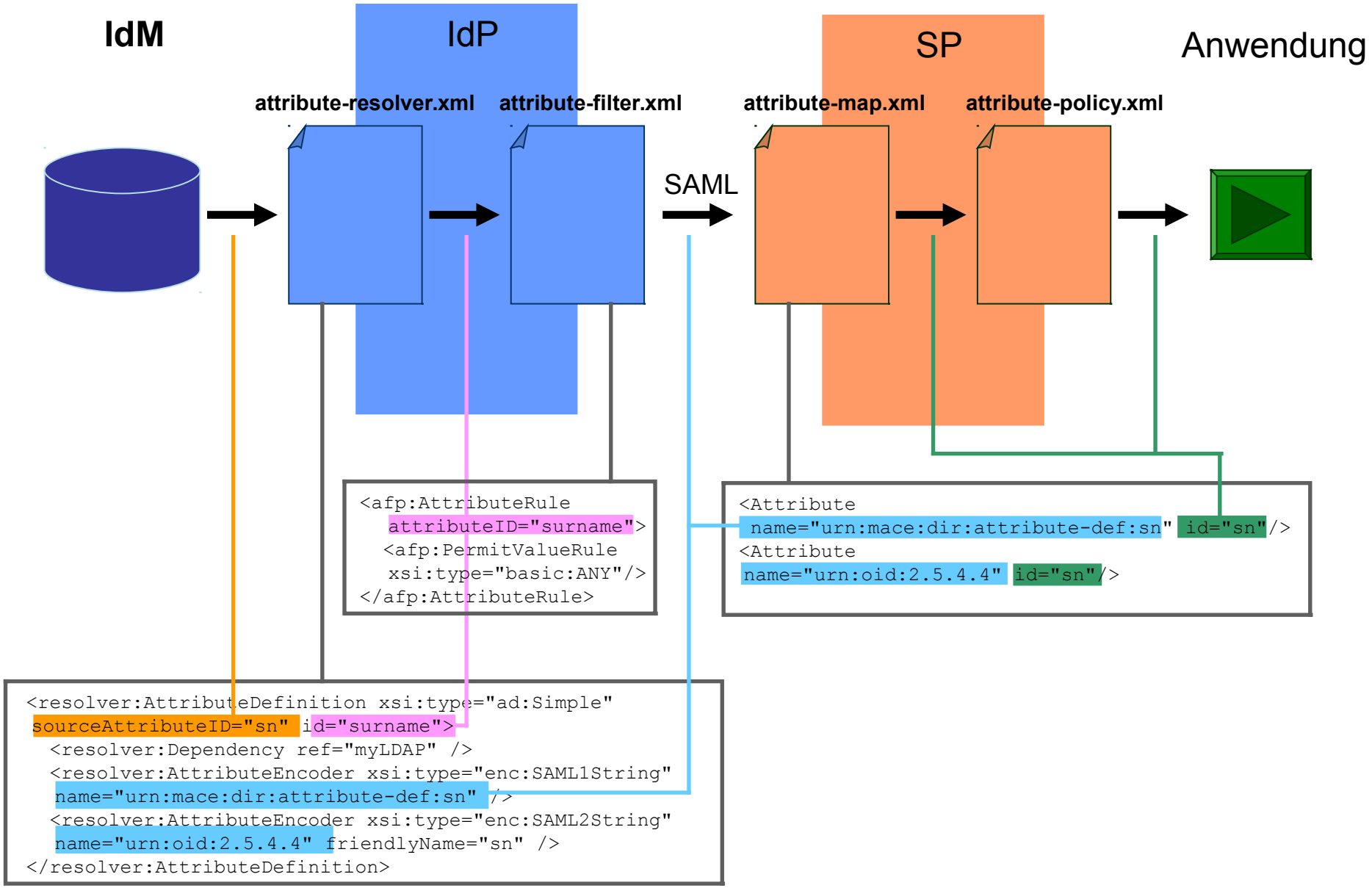
Inhalt

- Generelles zu Attributen
  - Attribut-Schemata: eduPerson, SCHAC
- Attribut-Handling im IdP
  - Resolver (attribut-resolver.xml)
  - Filter (attribut-filter.xml)
- SAML-Assertion
- Attribut-Handling im SP
  - Mapping (attribute-map.xml)
  - Filter (attribute-policy.xml)

Attribute: Generelles

- Attribute bilden die Grundlage für die Autorisierung (“was darf ich?”)
- Es gibt aber auch Attribute, die der eindeutigen Identifizierung des Nutzers dienen (z.B. ePPN)
- IdP stellt mithilfe von Attributen die notwendigen Informationen über den User zur Verfügung
- SP nimmt Attribute entgegen und stellt sie der Applikation zur Verfügung
- Applikation entscheidet anhand Ihrer Regeln über den Zugriff.

Überblick


IdM
IdP
SP
Anwendung
attribute-resolver.xml
attribute-filter.xml
attribute-map.xml
attribute-policy.xml
SAML
<afp:AttributeRule
attributeID="surname">
<afp:PermitValueRule
xsi:type="basic:ANY"/>
</afp:AttributeRule>
<Attribute
name="urn:mace:dir:attribute-def:sn" id="sn"/>
<Attribute
name="urn:oid:2.5.4.4" id="sn"/>
<resolver:AttributeDefinition xsi:type="ad:Simple"
sourceAttributeID="sn" id="surname">
<resolver:Dependency ref="myLDAP" />
<resolver:AttributeEncoder xsi:type="enc:SAML1String"
name="urn:mace:dir:attribute-def:sn" />
<resolver:AttributeEncoder xsi:type="enc:SAML2String"
name="urn:oid:2.5.4.4" friendlyName="sn" />
</resolver:AttributeDefinition>

Attribute: Schemata

- Schemata legen eine Menge von Attributen, die zulässigen Werte und deren Bedeutung fest.
- Im Föderationsumfeld hat sich etabliert:
  - eduPerson (Weltweit)
  - SCHAC (Erweiterungen für Europa)
  - dfnEduPerson (e-Learning, Deutschland)
  - aber auch einzelne Attribute aus inetOrgPerson
- Bindend für Kommunikation IdP ↔ SP, müssen aber nicht im IdM sein!

Attribute – Referenzen

- eduPerson (u.a.m.)
  http://macedir.org/specs/eduperson/
- dfnEduPerson
  https://www.aai.dfn.de/fileadmin/documents/attributes/20
  0811/dfneduperson-1.0.schema.txt
  OIDs:
  https://www.aai.dfn.de/fileadmin/documents/attributes/200
  811/Object_Identifier_DFN-AAI.pdf
- SCHAC (**Sch**ema for **Ac**ademia)
  https://wiki.refeds.org/display/STAN/SCHAC+Releases
- inetOrgPerson
  https://tools.ietf.org/html/rfc2798

Attribute: Beispiele

- eduPerson:
  - eduPersonScopedAffiliation (Status innerhalb der Heimateinrichtung)
  - eduPersonEntitlement (Berechtigung)
  - eduPersonPrincipalName (eindeutiger, nicht anonymer Username)
  - eduPersonTargetedID (eindeutiger, dienst-spezfischer anonymer bzw. pseudonymer Username)
- SCHAC:
  - schacPersonalUniqueCode (z.B. Mitarbeiternummer, Matrikelnummer)

Attribut-Handling im IdP

- Liest alle “rohen” Attribute des Users aus dem IdM (LDAP, SQL)
- “rohe” IdM-Attribute können gesplittet, zusammengefügt und umgeschrieben werden
- Neue Attribute können in Abhängigkeit von anderen Attributen / Werten generiert werden
- Die transformierten Attribute werden in SAML1- bzw. SAML2-Assertion (“Zusicherung”) verpackt.
- Assertion wird per HTTP-Post an den ACS des SPs geschickt (URL aus Föderations-Metadaten)

Dokumentation: https://wiki.shibboleth.net/confluence/display/SHIB2/IdPAddAttribute

IdP: IdM-Attribute auslesen

- attribute-resolver.xml:

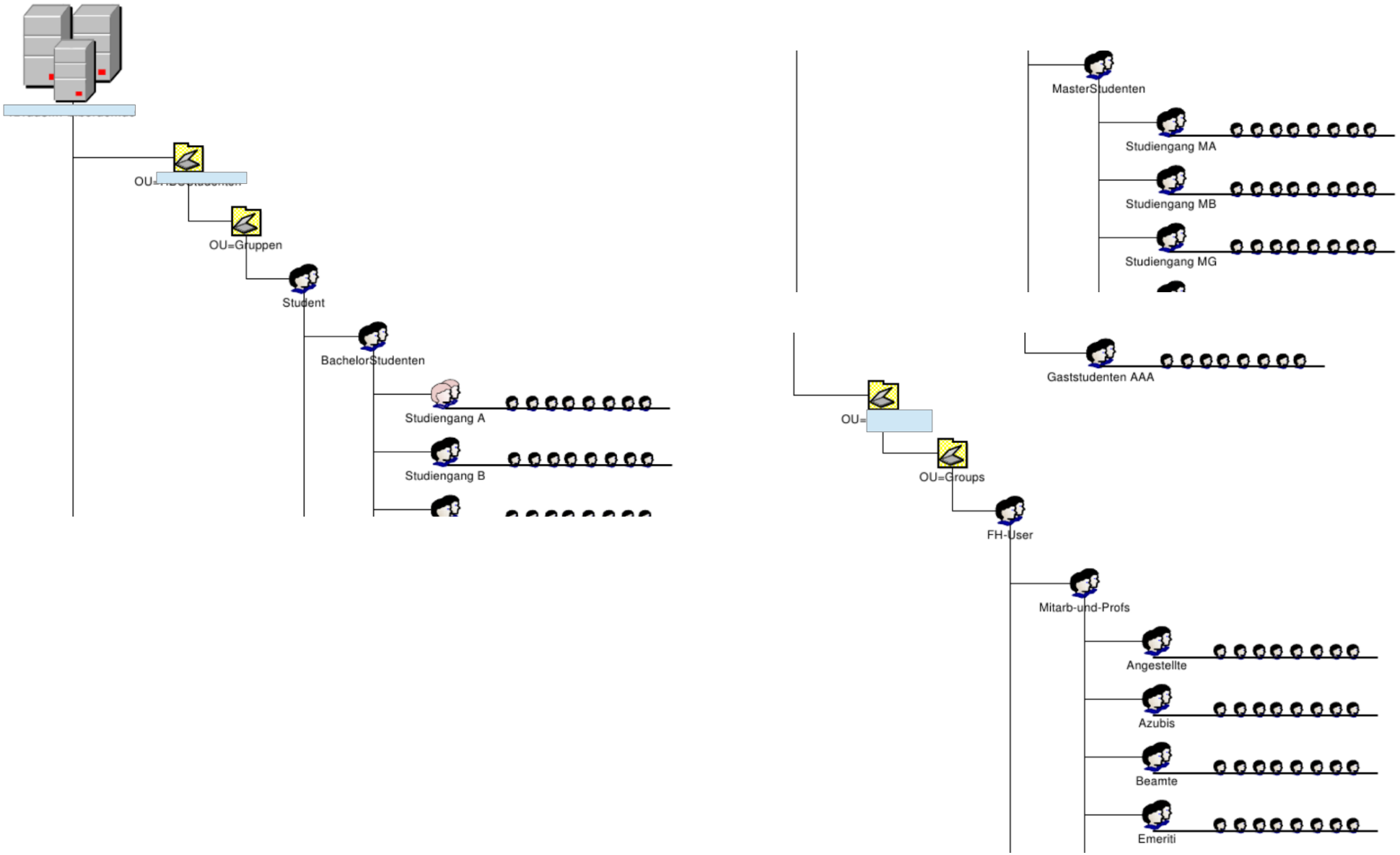
```
<resolver:DataConnector id="myLDAP" xsi:type="dc:LDAPDirectory"
     ldapURL="ldaps://ntserver.meineuni.de"
     baseDN="DC=users,DC=meineuni,DC=de"
     lowercaseAttributeNames="true"
     principal="CN=ldapextern,CN=systemusers,DC=meineuni,DC=de"
     principalCredential="strenggeheim007">

     <dc:FilterTemplate>
         <![CDATA[
             (cn=$requestContext.principalName)
         ]]>
     </dc:FilterTemplate>

</resolver:DataConnector>
```

Attribute für Web-SSO generieren

- “rohe” IdM-Attribute werden umgeschrieben auf eduPerson, SCHAC, dfnEduPerson, etc.
- Neue Attribute werden in Abhängigkeit von anderen Attributen bzw. -Werten generiert
- Mechanismus: “Attribute Definitions”
  - Simple
  - Mapped
  - Script
  - etc.
- Dies geschieht ebenfalls in attribute-resolver.xml

https://wiki.shibboleth.net/confluence/display/SHIB2/IdPAddAttribute#IdPAddAttribute-2.PrepareTheAttributes

IdP: Beispiel-IdM



IdP: Simple Attribute Definition

- attribute-resolver.xml:

```
<resolver:AttributeDefinition xsi:type="ad:Simple" id="email" sourceAttributeID="mail">
    <resolver:Dependency ref="myLDAP" />
    <resolver:AttributeEncoder xsi:type="enc:SAML1String" name="urn:mace:dir:attribute-def:mail" />
    <resolver:AttributeEncoder xsi:type="enc:SAML2String" name="urn:oid:0.9.2342.19200300.100.1.3" friendlyName="mail" />
</resolver:AttributeDefinition>

<resolver:AttributeDefinition xsi:type="ad:Simple" id="surname" sourceAttributeID="sn">
    <resolver:Dependency ref="myLDAP" />
    <resolver:AttributeEncoder xsi:type="enc:SAML1String" name="urn:mace:dir:attribute-def:sn" />
    <resolver:AttributeEncoder xsi:type="enc:SAML2String" name="urn:oid:2.5.4.4" friendlyName="sn" />
</resolver:AttributeDefinition>

<resolver:AttributeDefinition xsi:type="ad:Simple" id="givenName" sourceAttributeID="givenName">
    <resolver:Dependency ref="myLDAP" />
    <resolver:AttributeEncoder xsi:type="enc:SAML1String" name="urn:mace:dir:attribute-def:givenName" />
    <resolver:AttributeEncoder xsi:type="enc:SAML2String" name="urn:oid:2.5.4.42" friendlyName="givenName" />
</resolver:AttributeDefinition>

<resolver:AttributeDefinition xsi:type="ad:Scoped" id="eduPersonPrincipalName" scope="dfn.de" sourceAttributeID="uid">
    <resolver:Dependency ref="myLDAP" />
    <resolver:AttributeEncoder xsi:type="enc:SAML1ScopedString" name="urn:mace:dir:attribute-def:eduPersonPrincipalName" />
    <resolver:AttributeEncoder xsi:type="enc:SAML2ScopedString" name="urn:oid:1.3.6.1.4.1.5923.1.1.1.6" friendlyName="eduPersonPrincipalName" />
</resolver:AttributeDefinition>

<resolver:AttributeDefinition xsi:type="ad:Simple" id="employeeNumber" sourceAttributeID="uidNumber">
    <resolver:Dependency ref="myLDAP" />
    <resolver:AttributeEncoder xsi:type="enc:SAML1String" name="urn:mace:dir:attribute-def:employeeNumber" />
    <resolver:AttributeEncoder xsi:type="enc:SAML2String" name="urn:oid:2.16.840.1.113730.3.1.3" friendlyName="employeeNumber" />
</resolver:AttributeDefinition>
```

IdP: Mapped Attribute Definition

```
<resolver:AttributeDefinition xsi:type="ad:Mapped" id="eduPersonAffiliation" sourceAttributeID="memberof">
    <resolver:Dependency ref="myLDAP" />

    <resolver:AttributeEncoder xsi:type="enc:SAML1String" name="urn:mace:dir:attribute-def:eduPersonAffiliation" />
    <resolver:AttributeEncoder xsi:type="enc:SAML2String" name="urn:oid:1.3.6.1.4.1.5923.1.1.1.1" friendlyName="eduPersonAffiliation" />

    <!-- default to the generic value 'affiliate' -->
    <ad:DefaultValue>affiliate</ad:DefaultValue>

    <!-- die Gruppen aus dem AD der Hochschule. Siehe dazu DFN#: 20101227000051

       - nutzungsberechtigte Studentengruppen müssen in einer Gruppe sein
         die mit "CN=Studiengang" anfängt ("CN=Studiengang A", "CN=Studiengang B" etc.)
       - nutzungsberechtigte Mitarbeiter müssen in einer der folgenden Gruppen sein:
           CN=Angestellte, CN=Azubis, CN=Beamte, CN=Emeriti, CN=Gastprofessoren, CN=Professoren

       jede berechtige Gruppe bekommt hier eine eigene eduPersonAffiliation und zusätzlich "member" -->

    <ad:ValueMap>
        <ad:ReturnValue>student</ad:ReturnValue>
        <ad:SourceValue ignoreCase="true">cn=studiengang.+</ad:SourceValue>
    </ad:ValueMap>

    <ad:ValueMap>
        <ad:ReturnValue>staff</ad:ReturnValue>
        <ad:SourceValue ignoreCase="true">cn=Angestellte,.+</ad:SourceValue>
        <ad:SourceValue ignoreCase="true">cn=Azubis,.+</ad:SourceValue>
        <ad:SourceValue ignoreCase="true">cn=Beamte,.+</ad:SourceValue>
        <ad:SourceValue ignoreCase="true">cn=Emeriti,.+</ad:SourceValue>
        <ad:SourceValue ignoreCase="true">cn=Gastprofessoren,.+</ad:SourceValue>
        <ad:SourceValue ignoreCase="true">cn=Professoren,.+</ad:SourceValue>
    </ad:ValueMap>

    <ad:ValueMap>
        <ad:ReturnValue>member</ad:ReturnValue>
        <ad:SourceValue ignoreCase="true">cn=studiengang.+</ad:SourceValue>
        <ad:SourceValue ignoreCase="true">cn=Angestellte,.+</ad:SourceValue>
        <ad:SourceValue ignoreCase="true">cn=Azubis,.+</ad:SourceValue>
        <ad:SourceValue ignoreCase="true">cn=Beamte,.+</ad:SourceValue>
        <ad:SourceValue ignoreCase="true">cn=Emeriti,.+</ad:SourceValue>
        <ad:SourceValue ignoreCase="true">cn=Gastprofessoren,.+</ad:SourceValue>
        <ad:SourceValue ignoreCase="true">cn=Professoren,.+</ad:SourceValue>
    </ad:ValueMap>

</resolver:AttributeDefinition>
```

IdP: Script Attribute Definition

- attribute-resolver.xml:

```
<resolver:AttributeDefinition xsi:type="ad:Script" id="schacPersonalUniqueCode">
    <resolver:Dependency ref="myLDAP" />
    <resolver:Dependency ref="employeeNumber" />

    <resolver:AttributeEncoder xsi:type="enc:SAML1String" name="urn:mace:terena.org:schac:schacPersonalUniqueCode" />
    <resolver:AttributeEncoder xsi:type="enc:SAML2String" name="urn:oid:1.3.6.1.4.1.25178.1.2.14" friendlyName="schacPersonalUniqueCode" />

    <ad:Script><![CDATA[
        importPackage(Packages.edu.internet2.middleware.shibboleth.common.attribute.provider);

        // Create attribute to be returned from definition
        if (schacPersonalUniqueCode == null) {
              schacPersonalUniqueCode = new BasicAttribute("schacPersonalUniqueCode");
        }

        if (employeeNumber.getValues() != null) {
              schacPersonalUniqueCode.getValues().add("urn:mace:terena.org:schac:personalUniqueCode:de:dfn.de:uid:"
              +employeeNumber.getValues().get(0));
        }
        ]]>
    </ad:Script>

</resolver:AttributeDefinition>
```

IdP: Attribute Encoding

- Nach Umwandlung der Attribute werden diese in eine Form gebracht die der empfangende SP versteht:

```
<resolver:AttributeEncoder xsi:type="enc:SAML1String" name="urn:mace:terena.org:schac:schacPersonalUniqueCode" />
<resolver:AttributeEncoder xsi:type="enc:SAML2String" name="urn:oid:1.3.6.1.4.1.25178.1.2.14" friendlyName="schacPersonalUniqueCode" />
```

- Wird auch in attribute-resolver.xml konfiguriert.
- Die “üblichen” Attribute sind dort schon eingetragen → im Normalfall muss man sich nicht darum kümmern.

IdP: Attribut-Filter

- Legt fest welche Attribute an einen SP oder eine Gruppe von SPs versendet werden (Datenschutz!)
- Sehr flexibel, Regeln anhand bestimmter Kriterien
  - Föderation
  - SP (Entity ID)
  - User
  - Attribut-Wert (oder in Abhängigkeit von anderen)
  - Entity Attribute
  - Boolesche Kombinationen daraus
  - Scriptbasiert, ...

IdP: Attribut Filter

- attribute-filter.xml, Beispiel für Verlags-SPs:

```
<!-- Anonyme Angaben können an alle SP freigegeben werden,
    damit sind sehr viele SPs in der Föderation schon zufrieden -->

<afp:AttributeFilterPolicy id="releaseToAnyone">
    <afp:PolicyRequirementRule xsi:type="basic:ANY" />

    <!-- eduPersonEntitlement nur den relevanten Wert für die Verlage -->
    <afp:AttributeRule attributeID="eduPersonEntitlement">
      <afp:PermitValueRule xsi:type="basic:AttributeValueString"
         value="urn:mace:dir:entitlement:common-lib-terms"/>
    </afp:AttributeRule>

    <!-- eduPersonScopedAffiliation nur den anonymen Wert "member" -->
    <afp:AttributeRule attributeID="eduPersonScopedAffiliation">
      <afp:PermitValueRule xsi:type="basic:AttributeValueString"
         value="member" ignoreCase="true" />
    </afp:AttributeRule>

</afp:AttributeFilterPolicy>
```

IdP: Attribut Filter

- attribute-filter.xml, Beispiel für Moodle-SP:

```
<afp:AttributeFilterPolicy id="DFNMoodle">
    <afp:PolicyRequirementRule xsi:type="basic:AttributeRequesterString"
        value="https://moodle-dev.aai.dfn.de/shibboleth" />

    <afp:AttributeRule attributeID="givenName">
      <afp:PermitValueRule xsi:type="basic:ANY" />
    </afp:AttributeRule>

    <afp:AttributeRule attributeID="surname">
      <afp:PermitValueRule xsi:type="basic:ANY" />
    </afp:AttributeRule>

    <afp:AttributeRule attributeID="email">
      <afp:PermitValueRule xsi:type="basic:ANY" />
    </afp:AttributeRule>

    <afp:AttributeRule attributeID="eduPersonPrincipalName">
      <afp:PermitValueRule xsi:type="basic:ANY" />
    </afp:AttributeRule>

    <afp:AttributeRule attributeID="schacPersonalUniqueCode">
      <afp:PermitValueRule xsi:type="basic:ANY" />
    </afp:AttributeRule>
</afp:AttributeFilterPolicy>
```

SAML2-Assertion, Auszug

```
      </saml2:AuthnStatement>
      <saml2:AttributeStatement>
         <saml2:Attribute FriendlyName="eduPersonPrincipalName" Name="urn:oid:1.3.6.1.4.1.5923.1.1.1.6"
            NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri">
            <saml2:AttributeValue xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:type="xs:string">
               rgb@dfn.de
            </saml2:AttributeValue>
         </saml2:Attribute>
         <saml2:Attribute FriendlyName="sn" Name="urn:oid:2.5.4.4"
            NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri">
            <saml2:AttributeValue xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:type="xs:string">
               Borenius
            </saml2:AttributeValue>
         </saml2:Attribute>
         <saml2:Attribute FriendlyName="eduPersonScopedAffiliation" Name="urn:oid:1.3.6.1.4.1.5923.1.1.1.9"
            NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri">
            <saml2:AttributeValue xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:type="xs:string">
               member@dfn.de
            </saml2:AttributeValue>
         </saml2:Attribute>
         <saml2:Attribute FriendlyName="givenName" Name="urn:oid:2.5.4.42"
            NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri">
            <saml2:AttributeValue xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:type="xs:string">
               Raoul Gunnar
            </saml2:AttributeValue>
         </saml2:Attribute>
         <saml2:Attribute FriendlyName="schacPersonalUniqueCode" Name="urn:oid:1.3.6.1.4.1.25178.1.2.14"
            NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri">
            <saml2:AttributeValue xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:type="xs:string">
               urn:mace:terena.org:schac:personalUniqueCode:de:dfn.de:uid:60112
            </saml2:AttributeValue>
         </saml2:Attribute>
         <saml2:Attribute FriendlyName="mail" Name="urn:oid:0.9.2342.19200300.100.1.3"
            NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri">
            <saml2:AttributeValue xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:type="xs:string">
               borenius@dfn.de
            </saml2:AttributeValue>
         </saml2:Attribute>
         <saml2:Attribute FriendlyName="eduPersonEntitlement" Name="urn:oid:1.3.6.1.4.1.5923.1.1.1.7"
            NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri">
            <saml2:AttributeValue xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:type="xs:string">
               urn:mace:dir:entitlement:common-lib-terms
            </saml2:AttributeValue>
         </saml2:Attribute>
      </saml2:AttributeStatement>
   </saml2:Assertion>
</saml2p:Response>
```

SP: Assertion Consumer Service

- Empfängt die SAML-Assertion vom IdP
- Extrahiert die Attribute
- Bildet Attribute auf interne Variablen ab (attribute-map.xml)
- Filtert Variablen (attribute-policy.xml)
- Exportiert Variablen per CGI-Interface

SP: Attribute umschreiben

- attribute-map.xml

```
<Attribute name="urn:mace:dir:attribute-def:eduPersonPrincipalName" id="eppn">
    <AttributeDecoder xsi:type="ScopedAttributeDecoder"/>
</Attribute>
<Attribute name="urn:oid:1.3.6.1.4.1.5923.1.1.1.6" id="eppn">
    <AttributeDecoder xsi:type="ScopedAttributeDecoder"/>
</Attribute>

<Attribute name="urn:mace:dir:attribute-def:sn" id="sn"/>
<Attribute name="urn:oid:2.5.4.4" id="sn"/>

<Attribute name="urn:mace:dir:attribute-def:givenName" id="givenName"/>
<Attribute name="urn:oid:2.5.4.42" id="givenName"/>

<Attribute name="urn:mace:dir:attribute-def:mail" id="mail"/>
<Attribute name="urn:oid:0.9.2342.19200300.100.1.3" id="mail"/>

<Attribute name="urn:mace:terena.org:schac:schacPersonalUniqueCode" id="personalUniqueCode"/>
<Attribute name="urn:oid:1.3.6.1.4.1.25178.1.2.14" id="personalUniqueCode"/>
```

- Gängige Attribute sind vordefiniert!

SP: Attribute Filter

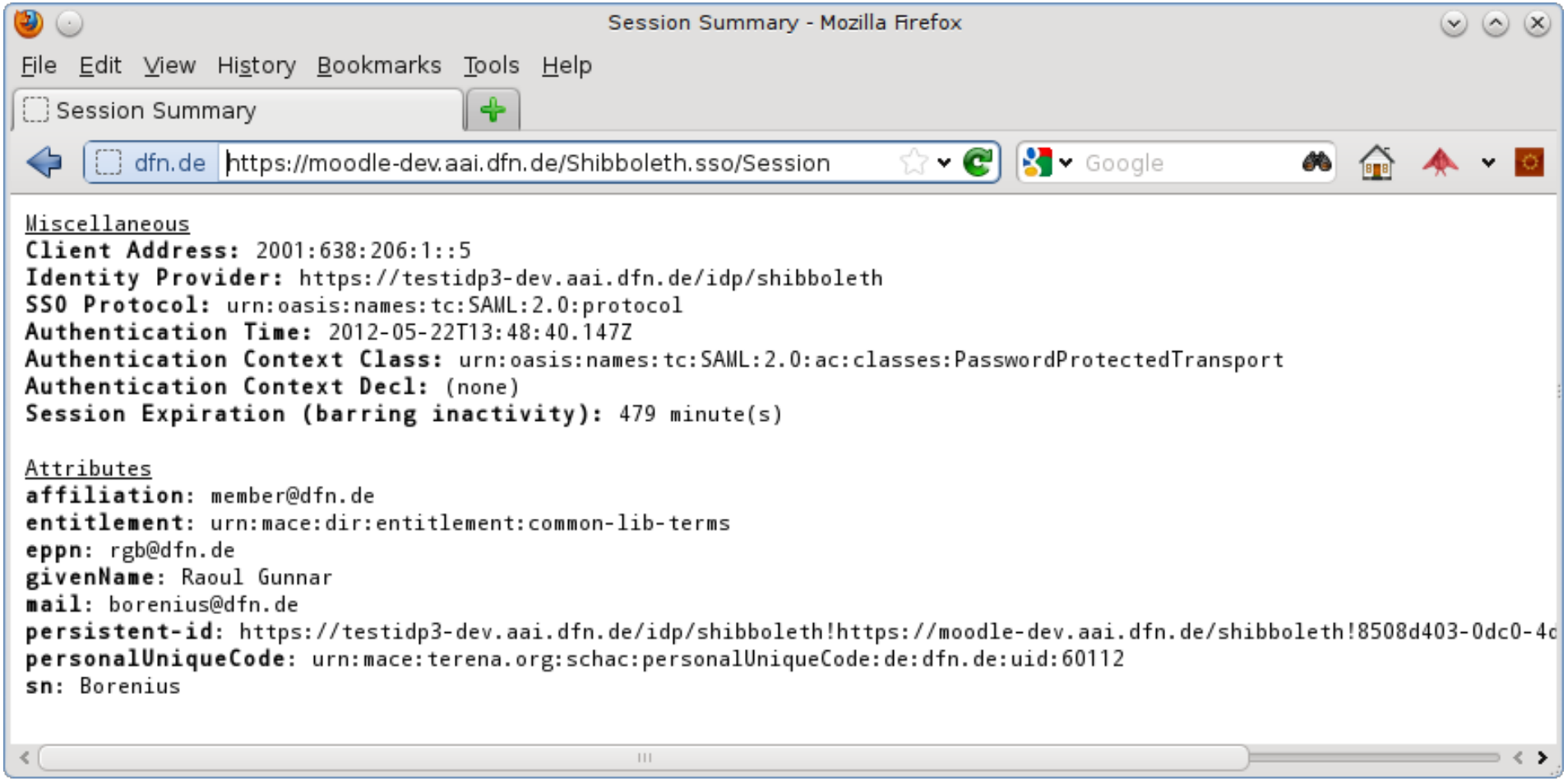
- Filtert Variablen und deren Werte
  - attribute-policy.xml, erzwingt bestimmte Werte bei eduPersonAffiliation:

```
<afp:PermitValueRule id="eduPersonAffiliationValues" xsi:type="OR">
        <Rule xsi:type="AttributeValueString" value="faculty"/>
        <Rule xsi:type="AttributeValueString" value="student"/>
        <Rule xsi:type="AttributeValueString" value="staff"/>
        <Rule xsi:type="AttributeValueString" value="alum"/>
        <Rule xsi:type="AttributeValueString" value="member"/>
        <Rule xsi:type="AttributeValueString" value="affiliate"/>
        <Rule xsi:type="AttributeValueString" value="employee"/>
        <Rule xsi:type="AttributeValueString" value="library-walk-in"/>
</afp:PermitValueRule>
```

- Default-Policy ist meist ausreichend

SP: Session-Informationen

- https://www.example.com/Shibboleth.sso/Session



SP: REMOTE_USER Variable

- spezielle CGI-Variable in der die Identität des Users enthalten ist.
- Eines oder mehrere Attribute aus attribute-map.xml können dafür verwendet werden!
- wird gesetzt in shibboleth2.xml:

```
<ApplicationOverride id="dfnmoodle"
        entityID="https://moodle-dev.aai.dfn.de/shibboleth"
        REMOTE_USER="eppn mail"
```

- Damit lassen sich auch Applikationen durch Shibboleth schützen, die keine direkte Shibboleth-Unterstützung mitbringen.

Überblick (nochmal)