

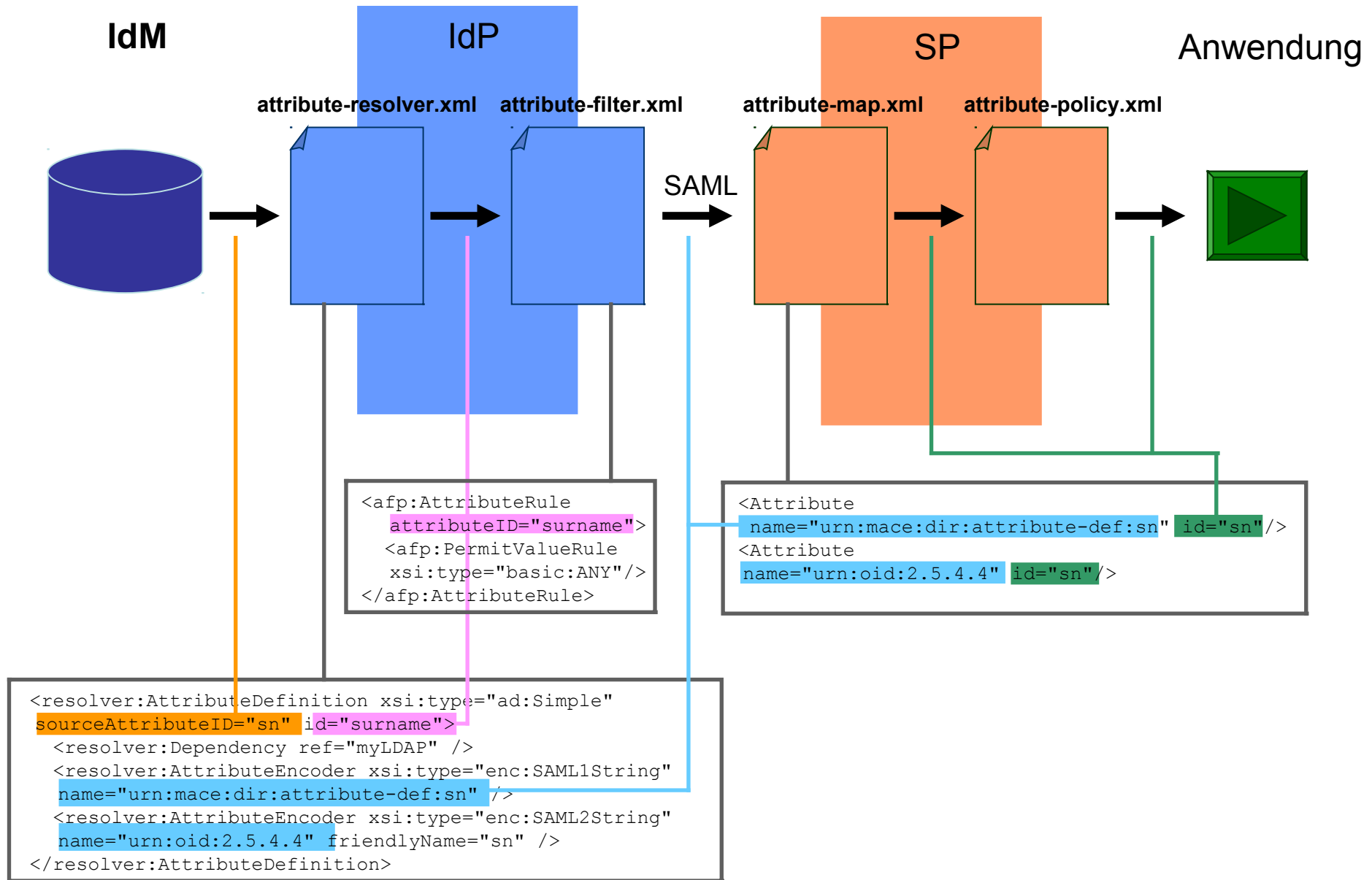
# Attribute

## Attribut-Schemata, -Generierung, -Übertragung und Verarbeitung am SP

Wolfgang Pempe, DFN-Verein  
pempe@dfn.de

DFN-AAI Workshop  
5./6. September 2017, FH Westküste

- Attribute bilden die Grundlage für die Autorisierung (“was darf ich?”)
- Daneben gibt es Attribute, die der eindeutigen Identifizierung des Nutzers und der Personalisierung des betreffenden Dienstes dienen (z.B. givenName, sn, eduPersonPrincipalName, mail)
- IdP stellt mithilfe von Attributen die notwendigen Informationen über den User zur Verfügung
- SP nimmt Attribute entgegen und stellt sie der jeweiligen Anwendung zur Verfügung
- SP und/oder Anwendung entscheiden anhand ihrer Regeln über den Zugriff.



- Schemata legen eine Menge von Attributen, die zulässigen Werte und deren Bedeutung fest.
- Im Föderationsumfeld hat sich etabliert:
  - eduPerson (international)
  - SCHAC (**S**chema for **A**cademia, international)
  - dfnEduPerson (e-Learning, Deutschland)
  - Sowie weitere Attribute z.B. aus inetOrgPerson
- Doku: <https://wiki.aai.dfn.de/de:attributes>
- Entscheidend für Kommunikation IdP ↔ SP, müssen aber nicht notwendigerweise im IdM vorhanden sein!

- eduPerson:
  - eduPersonScopedAffiliation (Status innerhalb der Heimateinrichtung + Scope),  
z.B. `student@uni-musterstadt.de`
  - eduPersonEntitlement (Berechtigung),  
z.B. `urn:mace:dir:entitlement:common-lib-terms`
  - eduPersonPrincipalName (eindeutiger, nicht anonymer Username), z.B. `hugo123@uni-musterstadt.de`
  - eduPersonUniqueid (eindeutiger, anonymer bzw. pseudonymer Username) z.B.  
`28c5353b8bb34984a8bd4169ba94c606@uni-musterstadt.de`
- SCHAC:
  - schacPersonalUniqueCode (insbes. Matrikelnummer), z.B.  
`urn:schac:personalUniqueCode:de:lmu.de:Matrikelnummer:1234567`

1. Liest alle “rohen” Attribute des Users aus dem IdM bzw. Nutzerverzeichnis (LDAP, AD, SQL)
2. Aus dem IdM gelesene Attribute können gesplittet, zusammengefügt und umgeschrieben werden
3. Neue Attribute können in Abhängigkeit von anderen Attributen / Werten generiert werden
4. Attribute Filter Policies entscheiden über die Weitergabe der Attribute / Attributwerte
5. Die transformierten Attribute werden in eine SAML-Assertion (“Zusicherung”) verpackt.
6. Assertion wird per HTTP-Post an den ACS des anfragenden SP geschickt (URL aus Föderations-Metadaten)

- Attribute Resolver: Data Connector
- Konfiguration: `./conf/attribute-resolver.xml`  
Vorbelegung der Werte in `./conf/ldap.properties`

```
<DataConnector id="myLDAP" xsi:type="LDAPDirectory"
  ldapURL="%{idp.attribute.resolver.LDAP.ldapURL}"
  baseDN="%{idp.attribute.resolver.LDAP.baseDN}"
  principal="%{idp.attribute.resolver.LDAP.bindDN}"
  principalCredential="%{idp.attribute.resolver.LDAP.bindDNCredential}"
  useStartTLS="%{idp.attribute.resolver.LDAP.useStartTLS:true}"
  connectTimeout="%{idp.attribute.resolver.LDAP.connectTimeout}"
  trustFile="%{idp.attribute.resolver.LDAP.trustCertificates}"
  responseTimeout="%{idp.attribute.resolver.LDAP.responseTimeout}">
  <FilterTemplate>
    <![CDATA[
      %{idp.attribute.resolver.LDAP.searchFilter}
    ]]>
  </FilterTemplate>
  <ConnectionPool
    minPoolSize="%{idp.pool.LDAP.minSize:3}"
    maxPoolSize="%{idp.pool.LDAP.maxSize:10}"
    blockWaitTime="%{idp.pool.LDAP.blockWaitTime:PT3S}"
    validatePeriodically="%{idp.pool.LDAP.validatePeriodically:true}"
    validateTimerPeriod="%{idp.pool.LDAP.validatePeriod:PT5M}"
    expirationTime="%{idp.pool.LDAP.idleTime:PT10M}"
    failFastInitialize="%{idp.pool.LDAP.failFastInitialize:false}" />
</DataConnector>
```

- [Beispiel]
- Siehe auch unter <https://wiki.aai.dfn.de/de:shibidp3config-idm>



- Aus dem IdM ausgelesene Attribute werden umgeschrieben auf eduPerson, SCHAC, dfnEduPerson, etc.
- Neue Attribute werden in Abhängigkeit von anderen Attributen bzw. -Werten generiert
- Mechanismus: “Attribute Definitions”
  - Simple
  - Mapped
  - ScriptedAttribute
  - u.a.m. → <https://wiki.shibboleth.net/confluence/x/JgIUAQ>
- Dies geschieht ebenfalls in attribute-resolver.xml

```
<!--
```

```
The uid is the closest thing to a "standard" LDAP attribute representing a local username, but you should generally *never* expose uid to federated services, as it is rarely globally unique.
```

```
-->  
<AttributeDefinition id="uid" xsi:type="Simple" sourceAttributeID="sAMAccountName">  
  <Dependency ref="myLDAP" />  
  <AttributeEncoder xsi:type="SAML1String" name="urn:mace:dir:attribute-def:uid" encodeType="false" />  
  <AttributeEncoder xsi:type="SAML2String" name="urn:oid:0.9.2342.19200300.100.1.1" friendlyName="uid" encodeType="false" />  
</AttributeDefinition>
```

```
<!--
```

```
In the rest of the world, the email address is the standard identifier, despite the problems with that practice. Consider making the EPPN value the same as your official email addresses whenever possible.
```

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

<https://wiki.shibboleth.net/confluence/display/IDP30/SimpleAttributeDefinition>

# IdP: Mapped Attribute Definition

```
<AttributeDefinition xsi:type="Mapped" id="eduPersonAffiliation" sourceAttributeID="memberof">
  <Dependency ref="myLDAP" />
  <AttributeEncoder xsi:type="SAML1String" name="urn:mace:dir:attribute-def:eduPersonAffiliation" />
  <AttributeEncoder xsi:type="SAML2String" name="urn:oid:1.3.6.1.4.1.5923.1.1.1.1"
    friendlyName="eduPersonAffiliation" />
  <DefaultValue>affiliate</DefaultValue>
  <ValueMap>
    <ReturnValue>staff</ReturnValue>
    <SourceValue>Mitarbeiter</SourceValue>
  </ValueMap>
  <ValueMap>
    <ReturnValue>employee</ReturnValue>
    <SourceValue>Mitarbeiter</SourceValue>
    <SourceValue>Professor</SourceValue>
    <SourceValue>Lehrbeauftragte</SourceValue>
  </ValueMap>
  <ValueMap>
    <ReturnValue>faculty</ReturnValue>
    <SourceValue>Professor</SourceValue>
    <SourceValue>Lehrbeauftragte</SourceValue>
  </ValueMap>
  <ValueMap>
    <ReturnValue>student</ReturnValue>
    <SourceValue>Studierende</SourceValue>
  </ValueMap>
  <ValueMap>
    <ReturnValue>member</ReturnValue>
    <SourceValue>Mitarbeiter</SourceValue>
    <SourceValue>Studierende</SourceValue>
    <SourceValue>Professor</SourceValue>
    <SourceValue>Lehrbeauftragte</SourceValue>
  </ValueMap>
</AttributeDefinition>
```

s.u. <https://wiki.shibboleth.net/confluence/display/IDP30/MappedAttributeDefinition>

```
<AttributeDefinition xsi:type="ScriptedAttribute" id="eduPersonEntitlement">
  <Dependency ref="myLDAP" />
  <Dependency ref="eduPersonAffiliation" />
  <DisplayName xml:lang="en">Entitlement</DisplayName>
  <DisplayName xml:lang="de">Berechtigung</DisplayName>
  <DisplayDescription xml:lang="en">URI that indicates a set of rights to specific resources</DisplayDescription>
  <DisplayDescription xml:lang="de">Zeichenkette, welche Rechte für spezifischen Ressourcen beschreibt</DisplayDescription>
  <AttributeEncoder xsi:type="SAML1String" name="urn:mace:dir:attribute-def:eduPersonEntitlement" />
  <AttributeEncoder xsi:type="SAML2String" name="urn:oid:1.3.6.1.4.1.5923.1.1.1.7"
    friendlyName="eduPersonEntitlement" />
  <Script>
    <![CDATA[
      if (eduPersonAffiliation.getValues().contains("member")) {
        eduPersonEntitlement.getValues().add("urn:mace:dir:entitlement:common-lib-terms");
        eduPersonEntitlement.getValues().add("http://bwidm.de/entitlement/bwLSDf-SyncShare");
        eduPersonEntitlement.getValues().add("http://bwidm.de/entitlement/bwLSDf-FileService");
        eduPersonEntitlement.getValues().add("http://bwidm.de/entitlement/bwUniCluster");
      }
      if (eduPersonAffiliation.getValues().contains("staff")) {
        eduPersonEntitlement.getValues().add("http://bwidm.de/entitlement/bwLehrpool");
      }
    ]]>
  </Script>
</AttributeDefinition>
```

Shibboleth Wiki:

<https://wiki.shibboleth.net/confluence/display/IDP30/ScriptedAttributeDefinition>

Zu den Unterschieden der Scripting Engines unter Java 7 und 8 siehe auch

<https://wiki.shibboleth.net/confluence/display/SHIB2/IdPJava1.8>

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

```
<AttributeEncoder xsi:type="SAML1String" name="urn:mace:dir:attribute-def:mail"
    encodeType="false" />
<AttributeEncoder xsi:type="SAML2String" name="urn:oid:0.9.2342.19200300.100.1.3"
    friendlyName="mail" encodeType="false" />
```

- Wird auch in attribute-resolver.xml konfiguriert.
- Die gängigsten Attribute sind dort schon eingetragen (ggf. ent-kommentieren), d.h. im Normalfall muss man sich nicht darum kümmern.

- Shibboleth Wiki:

<https://wiki.shibboleth.net/confluence/display/IDP30/AttributeEncoderPluginConfiguration>

- Legt fest, welche Attribute an einen SP oder eine Gruppe von SP versendet werden (Datenschutz!)
- Attribute werden anhand ihrer ID aus dem Attribute Resolver referenziert: `AttributeDefinition/@id`
- Sehr flexibel, Regeln anhand bestimmter Kriterien, z.B.:
  - Föderation
  - SP (Entity ID)
  - User
  - Attribut-Wert (oder in Abhängigkeit von anderen)
  - Entity Attribute
  - Boolesche Kombinationen daraus
- Shibboleth Wiki
  - <https://wiki.shibboleth.net/confluence/display/IDP30/AttributeFilterConfiguration>
  - <https://wiki.shibboleth.net/confluence/display/IDP30/AttributeFilterPolicyConfiguration>



Konfiguration: ./conf/attribute-filter.xml

```
<!-- Attribute an die DFN-Test-IdPs freigeben -->
<AttributeFilterPolicy id="dfn_test_sps">
  <PolicyRequirementRule xsi:type="OR">
    <Rule xsi:type="Requester" value="https://testsp.aai.dfn.de/shibboleth" />
    <Rule xsi:type="Requester" value="https://testsp2.aai.dfn.de/shibboleth" />
    <Rule xsi:type="Requester" value="https://testsp3.aai.dfn.de/shibboleth" />
  </PolicyRequirementRule>
  <AttributeRule attributeID="uid" permitAny="true"/>
  <AttributeRule attributeID="eduPersonPrincipalName" permitAny="true"/>
  <AttributeRule attributeID="mail" permitAny="true"/>
  <AttributeRule attributeID="surname" permitAny="true"/>
  <AttributeRule attributeID="givenName" permitAny="true"/>
  <AttributeRule attributeID="o" permitAny="true"/>
  <AttributeRule attributeID="eduPersonScopedAffiliation" permitAny="true"/>
  <AttributeRule attributeID="eduPersonAffiliation" permitAny="true"/>
  <AttributeRule attributeID="eduPersonEntitlement" permitAny="true"/>
  <AttributeRule attributeID="displayName" permitAny="true"/>
  <AttributeRule attributeID="schacHomeOrganization" permitAny="true"/>
  <AttributeRule attributeID="schacHomeOrganizationType" permitAny="true"/>
</AttributeFilterPolicy>
```

Konfiguration: `./conf/attribute-filter.xml`

```
<AttributeFilterPolicy id="LibraryTermsToAnyone">
  <PolicyRequirementRule xsi:type="ANY" />

  <AttributeRule attributeID="eduPersonEntitlement">
    <PermitValueRule xsi:type="Value" value="urn:mace:dir:entitlement:common-lib-terms"/>
  </AttributeRule>

  <AttributeRule attributeID="eduPersonScopedAffiliation">
    <PermitValueRule xsi:type="OR">
      <Rule xsi:type="Value" value="member" />
      <Rule xsi:type="Value" value="library-walk-in" />
    </PermitValueRule>
  </AttributeRule>
</AttributeFilterPolicy>
```

Weitere Beispiele im DFN-AAI Wiki unter  
<https://wiki.aai.dfn.de/de:shibidp3attributes>

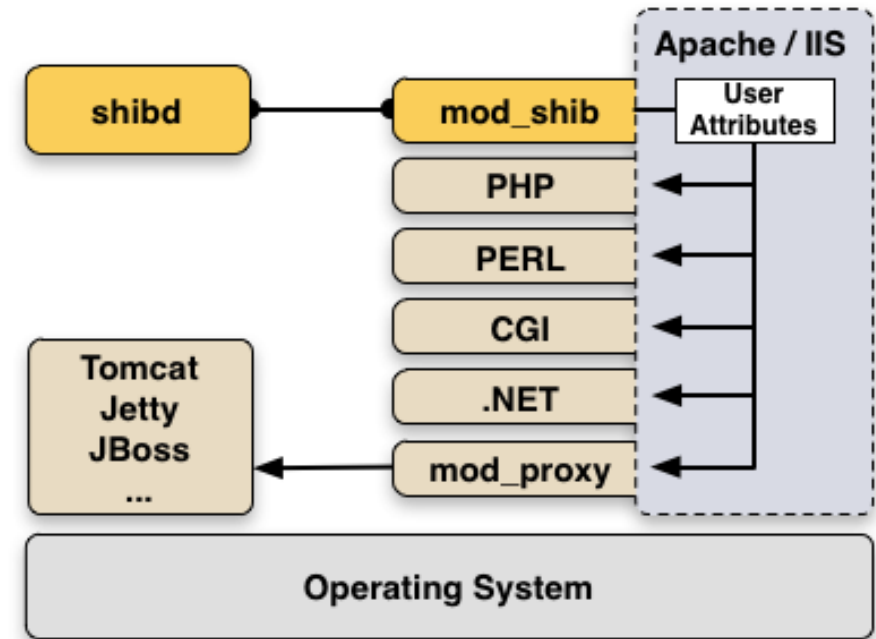


Attribute werden im Attribute Statement übertragen:

```
<saml2:AttributeStatement>
  <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>urn:mace:dir:entitlement:common-lib-terms</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>member@testscope.aai.dfn.de</saml2:AttributeValue>
  </saml2:Attribute>
</saml2:AttributeStatement>
```

Der empfangende SP identifiziert die Attribute anhand des URI-Wertes (meistens urn:oid) in **Name**

- Empfängt die SAML-Assertion vom IdP
- Extrahiert die Attribute
- Identifiziert die Attribute anhand eindeutiger Kennung (URI in `saml2:Attribute/@Name`)
- Bildet Attribute auf interne Variablen ab (`attribute-map.xml`)
- Filtert Variablen (`attribute-policy.xml`)
- Exportiert Variablen per CGI-Interface



Quelle: <https://www.switch.ch/aai/guides/sp/>

- /etc/shibboleth/attribute-map.xml
- Bildet Attribute auf interne Variablen ab
- Variablenname wird in **id** definiert

```
<Attribute name="urn:oid:1.3.6.1.4.1.5923.1.1.1.9" id="affiliation">
  <AttributeDecoder xsi:type="ScopedAttributeDecoder" caseSensitive="false"/>
</Attribute>
<Attribute name="urn:mace:dir:attribute-def:eduPersonScopedAffiliation" id="affiliation">
  <AttributeDecoder xsi:type="ScopedAttributeDecoder" caseSensitive="false"/>
</Attribute>

<Attribute name="urn:oid:1.3.6.1.4.1.5923.1.1.1.1" id="unscoped-affiliation">
  <AttributeDecoder xsi:type="StringAttributeDecoder" caseSensitive="false"/>
</Attribute>
<Attribute name="urn:mace:dir:attribute-def:eduPersonAffiliation" id="unscoped-affiliation">
  <AttributeDecoder xsi:type="StringAttributeDecoder" caseSensitive="false"/>
</Attribute>

<Attribute name="urn:oid:1.3.6.1.4.1.5923.1.1.1.7" id="entitlement"/>
<Attribute name="urn:mace:dir:attribute-def:eduPersonEntitlement" id="entitlement"/>
```

- /etc/shibboleth/attribute-policy.xml
- Filtert Variablen und deren Werte, i.d.R. genügen die Default-Einstellungen

```
<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>

<afp:PermitValueRule id="ScopingRules" xsi:type="AND">
  <Rule xsi:type="NOT">
    <Rule xsi:type="AttributeValueRegex" regex="@"/>
  </Rule>
  <Rule xsi:type="saml:AttributeScopeMatchesShibMDScope"/>
</afp:PermitValueRule>

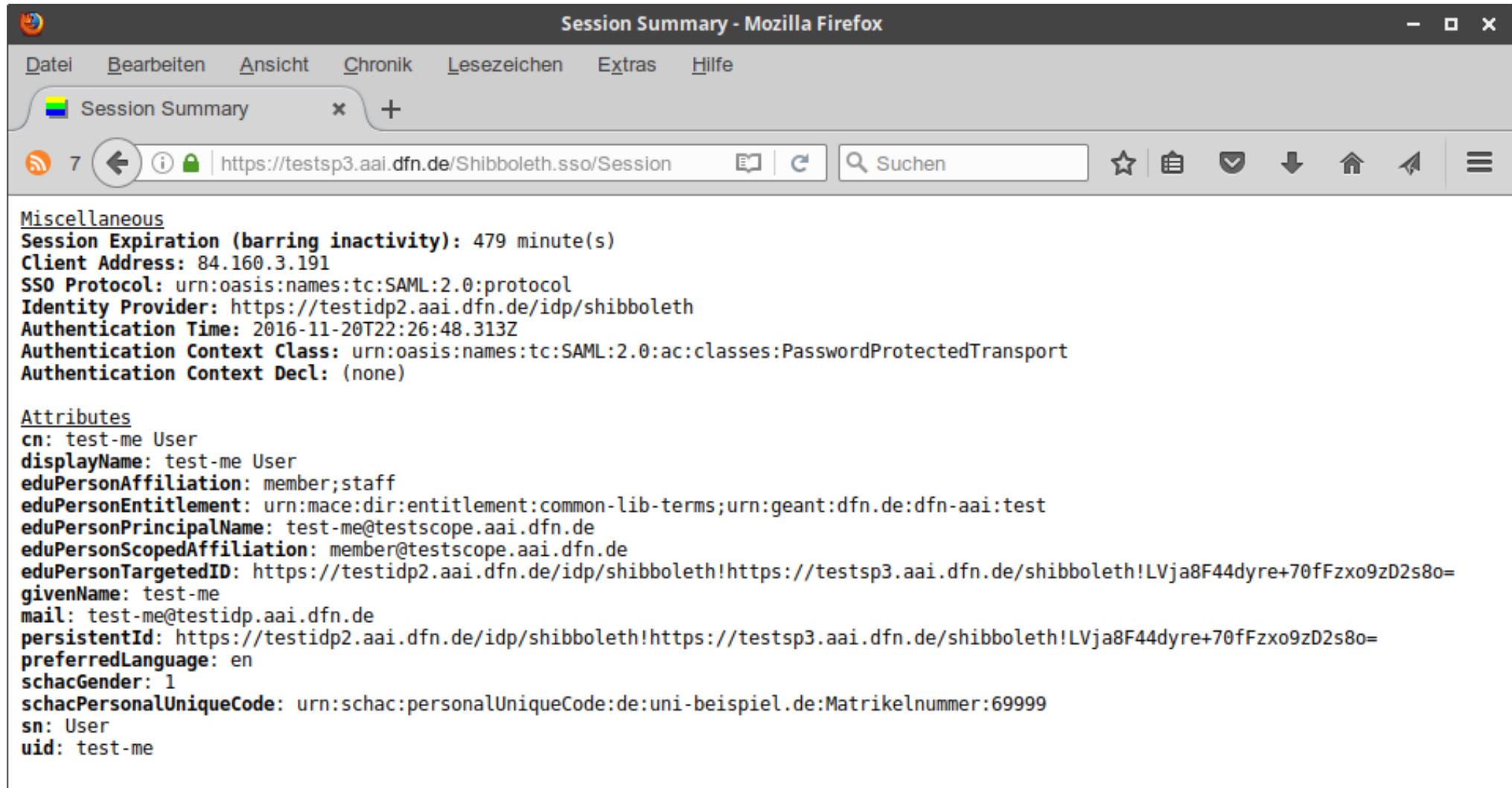
<afp:AttributeFilterPolicy>
  <!-- This policy is in effect in all cases. -->
  <afp:PolicyRequirementRule xsi:type="ANY"/>
  <!-- Filter out undefined affiliations and ensure only one primary. -->
  <afp:AttributeRule attributeID="affiliation">
    <afp:PermitValueRule xsi:type="AND">
      <RuleReference ref="eduPersonAffiliationValues"/>
      <RuleReference ref="ScopingRules"/>
    </afp:PermitValueRule>
  </afp:AttributeRule>
  <afp:AttributeRule attributeID="unscoped-affiliation">
    <afp:PermitValueRuleReference ref="eduPersonAffiliationValues"/>
  </afp:AttributeRule>
  <!-- Catch-all that passes everything else through unmolested. -->
  <afp:AttributeRule attributeID="*" permitAny="true"/>
</afp:AttributeFilterPolicy>
```

- Spezielle CGI-Variable, in der die Identität des Users transportiert wird.
- Eine oder mehrere der in attribute-map.xml definierten Variablen können verwendet werden
- wird gesetzt in shibboleth2.xml:

```
<ApplicationDefaults entityID="https://loa-check.aai.dfn.de/shibboleth"  
    REMOTE_USER="eppn persistent-id mail eduPersonUniqueId">
```

- Damit lassen sich auch Applikationen durch Shibboleth schützen, die keine direkte Shibboleth-(Attribut-)Unterstützung mitbringen  
→ Alternative zu Basic Auth
- Siehe auch unter <https://wiki.shibboleth.net/confluence/display/SHIB2/NativeSPApplication>

## Session Handler URL Extension bei Shibboleth SP: /Shibboleth.sso/Session



Session Summary - Mozilla Firefox

Datei Bearbeiten Ansicht Chronik Lesezeichen Extras Hilfe

Session Summary x +

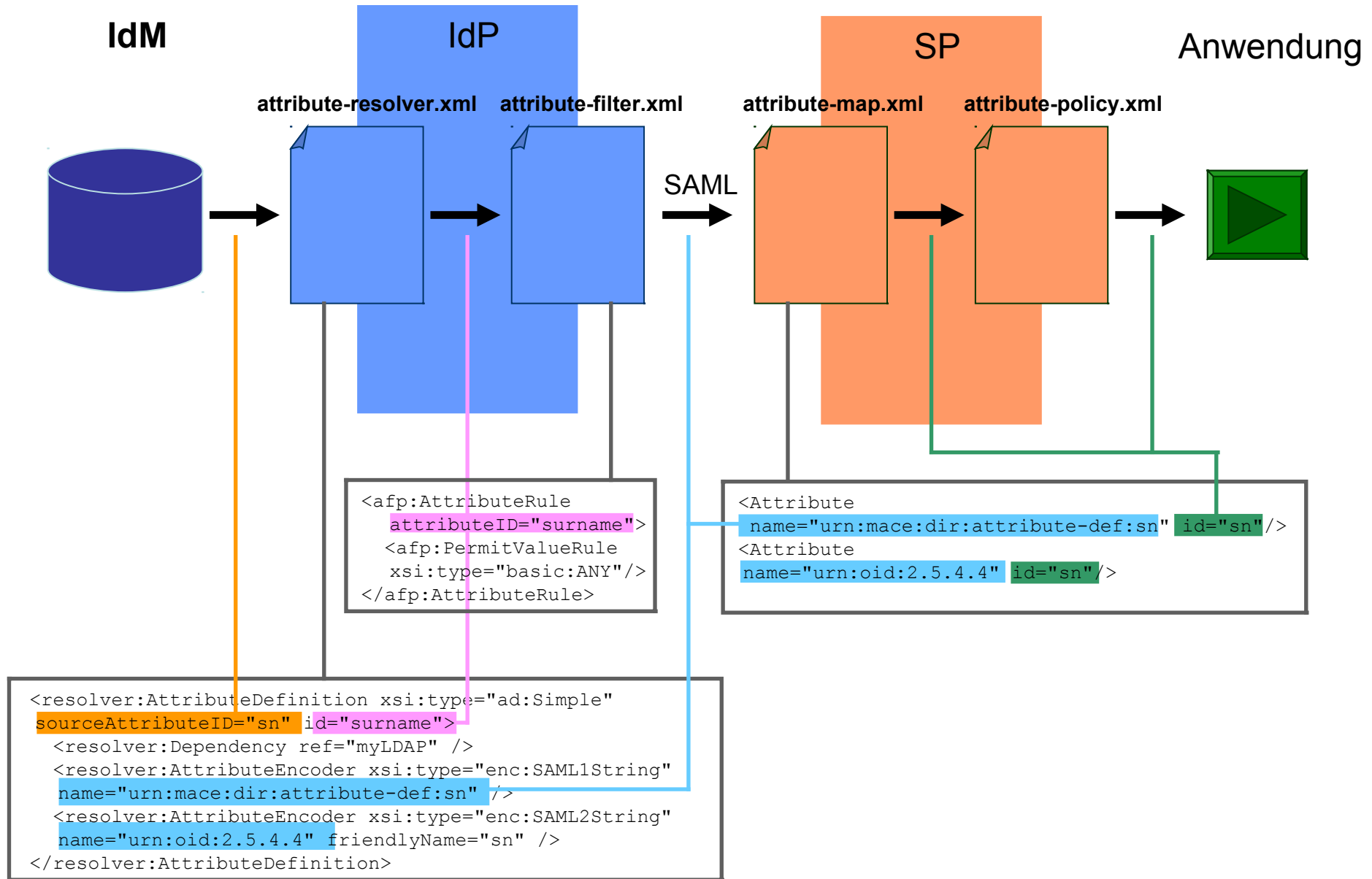
7 | <https://testsp3.aai.dfn.de/Shibboleth.sso/Session>

Miscellaneous  
**Session Expiration (barring inactivity):** 479 minute(s)  
**Client Address:** 84.160.3.191  
**SSO Protocol:** urn:oasis:names:tc:SAML:2.0:protocol  
**Identity Provider:** <https://testidp2.aai.dfn.de/idp/shibboleth>  
**Authentication Time:** 2016-11-20T22:26:48.313Z  
**Authentication Context Class:** urn:oasis:names:tc:SAML:2.0:ac:classes>PasswordProtectedTransport  
**Authentication Context Decl:** (none)

Attributes  
**cn:** test-me User  
**displayName:** test-me User  
**eduPersonAffiliation:** member;staff  
**eduPersonEntitlement:** urn:mace:dir:entitlement:common-lib-terms;urn:geant:dfn.de:dfn-aai:test  
**eduPersonPrincipalName:** test-me@testscope.aai.dfn.de  
**eduPersonScopedAffiliation:** member@testscope.aai.dfn.de  
**eduPersonTargetedID:** <https://testidp2.aai.dfn.de/idp/shibboleth!https://testsp3.aai.dfn.de/shibboleth!LVja8F44dyre+70fFzxo9zD2s8o=>  
**givenName:** test-me  
**mail:** test-me@testidp.aai.dfn.de  
**persistentId:** <https://testidp2.aai.dfn.de/idp/shibboleth!https://testsp3.aai.dfn.de/shibboleth!LVja8F44dyre+70fFzxo9zD2s8o=>  
**preferredLanguage:** en  
**schacGender:** 1  
**schacPersonalUniqueCode:** urn:schac:personalUniqueCode:de:uni-beispiel.de:Matrikelnummer:69999  
**sn:** User  
**uid:** test-me



# Überblick (nochmal)



- DFN-AAI Wiki:  
<https://wiki.aai.dfn.de/de:attributes>  
<https://wiki.aai.dfn.de/de:shibidp3attributes>
- eduPerson (u.a.m.)  
<http://macedir.org/specs/eduperson/>
- dfnEduPerson  
kommentierte Liste:  
[https://wiki.aai.dfn.de/de:elearning\\_attributes](https://wiki.aai.dfn.de/de:elearning_attributes)
- SCHAC (**S**chema for **A**cademia)  
<https://wiki.refeds.org/display/STAN/SCHAC+Releases>
- inetOrgPerson  
<https://tools.ietf.org/html/rfc2798>
- Shibboleth Wiki:  
<https://wiki.shibboleth.net/confluence/display/IDP30/AttributeResolverConfiguration>  
<https://wiki.shibboleth.net/confluence/display/IDP30/AttributeFilterConfiguration>  
<https://wiki.shibboleth.net/confluence/display/SHIB2/NativeSPAddAttribute>



# Vielen Dank für Ihre Aufmerksamkeit!

## Ideen? Fragen? Anmerkungen?

### Kontakt

Portal: <https://www.aai.dfn.de>

E-Mail: [aai@dfn.de](mailto:aai@dfn.de)

Tel.: +49 30 884299-9124