Einführung in Shibboleth 2

6. Shibboleth Workshop

08.05.2008, Frankfurt am Main

Franck Borel - UB Freiburg

Übersicht

- Was ist Shibboleth?
- Warum Shibboleth?
- Wie funktioniert Shibboleth 2?
- Authentifizierung
- Attribute
- Discovery-Service
- Metadaten

Was ist Shibboleth?

- **Shibboleth** ist ein einrichtungsübergreifender **SSO-Dienst** für den Zugriff auf geschützte **Web-Ressourcen**
- Wird von Internet2 entwickelt
  → http://shibboleth.internet2.edu
- Basiert auf SAML:
  Security **A**ssertion **M**arkup **L**anguage
- Open-Source-Lizenz

Warum Shibboleth?

- **Nutzer**
  - Zugriff auf Dienste von überall her
  - Alle Dienste sollen nach einmaliger Authentifizierung und Autorisierung zur Verfügung stehen (**Single-Sign-On**).
- **Einrichtungen** (etwa Hochschulen)
  - bestehende Benutzerverwaltung nutzen
  - Einfache Anbindung an die bestehende Benutzerverwaltung
- **Anbieter**
  - Schützen der lizenzpflichtigen Inhalte
  - Keine eigene Benutzerverwaltung
  - Nutzung kontrollieren (wer darf was?)

Wie funktioniert Shibboleth?

Benutzerin

Erster Zugriff

(1)

Anbieter

Shibboleth-Sitzung vorhanden?

(2) nein

(3)

Discovery-Service

(4)

Authentifizierungsanfrage

(5)

(6)

Login

Heimateinrichtung

AuthN & Attribute

(7)

(8)

Ja

Nein

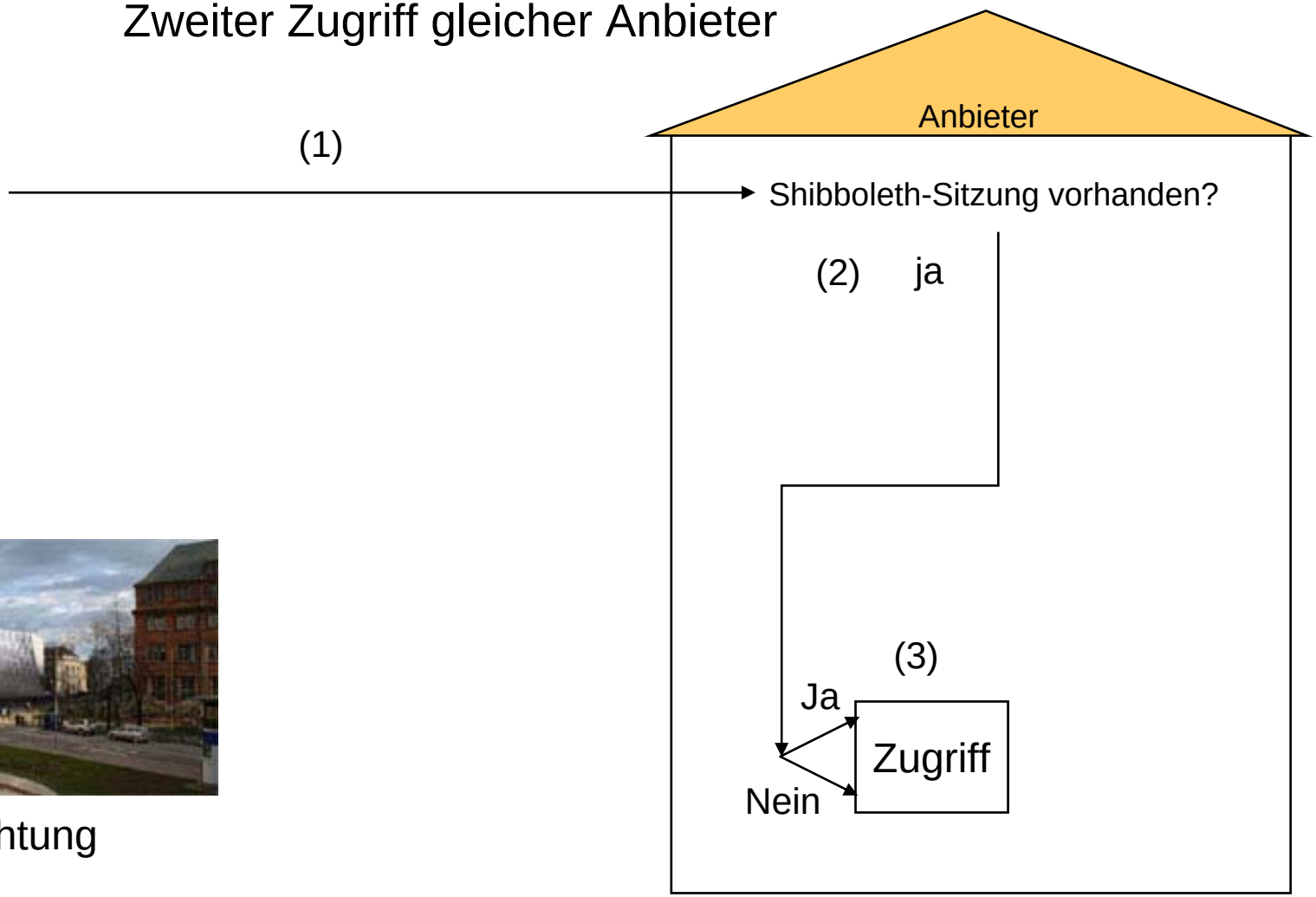
Zugriff

Wie funktioniert Shibboleth?

Benutzerin

Heimateinrichtung

Zweiter Zugriff gleicher Anbieter



Wie funktioniert Shibboleth?

Benutzerin

Erster Zugriff anderer Anbieter

Anbieter

(1)

Shibboleth-Sitzung vorhanden?

(2) nein

Discovery-Service

(3)

Authentifizierungsanfrage

(4)

Heimateinrichtung

(6)

Ja

Zugriff

Nein

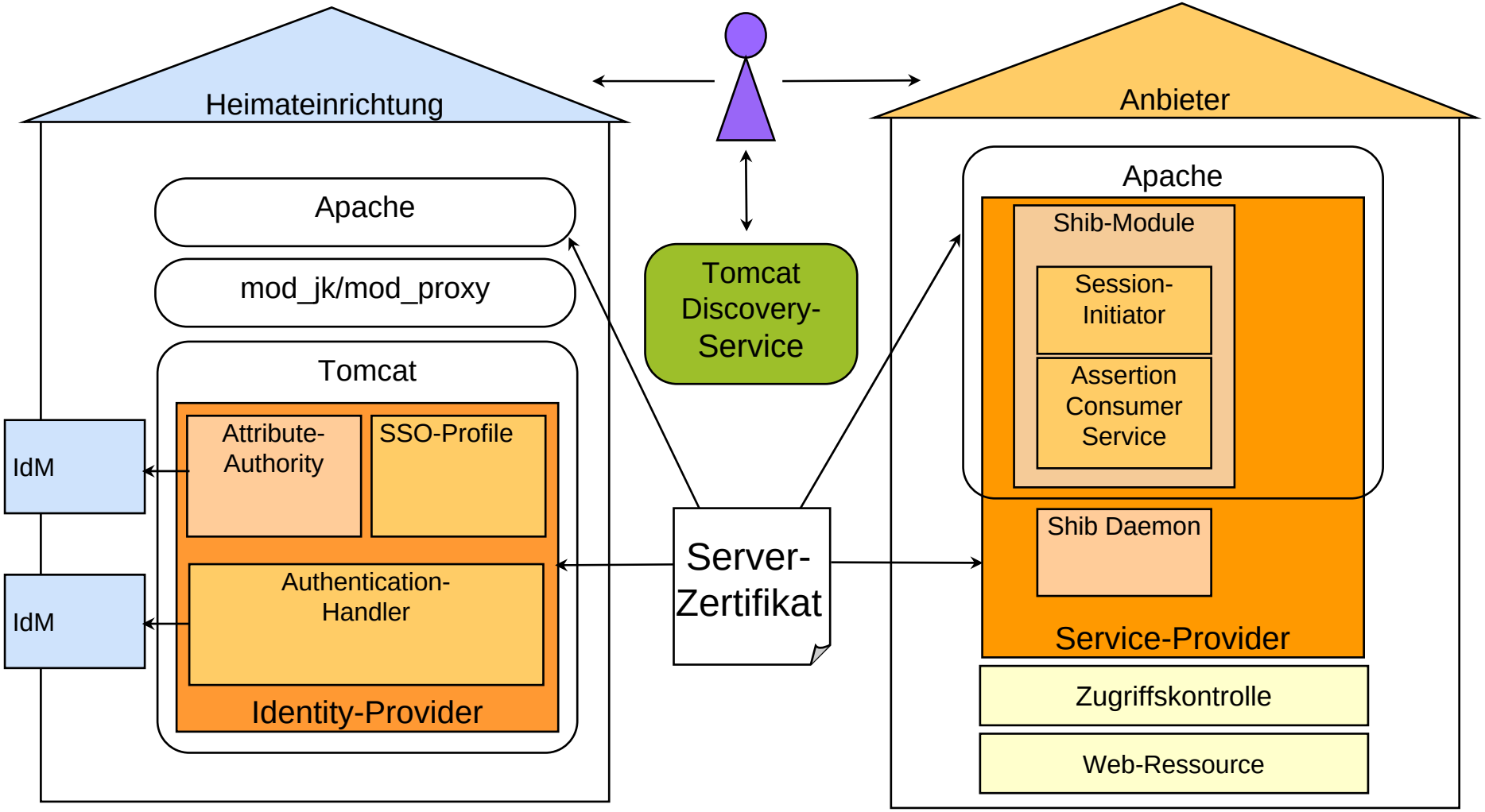
AuthN & Attribute

(5)

Bestandteile/Software

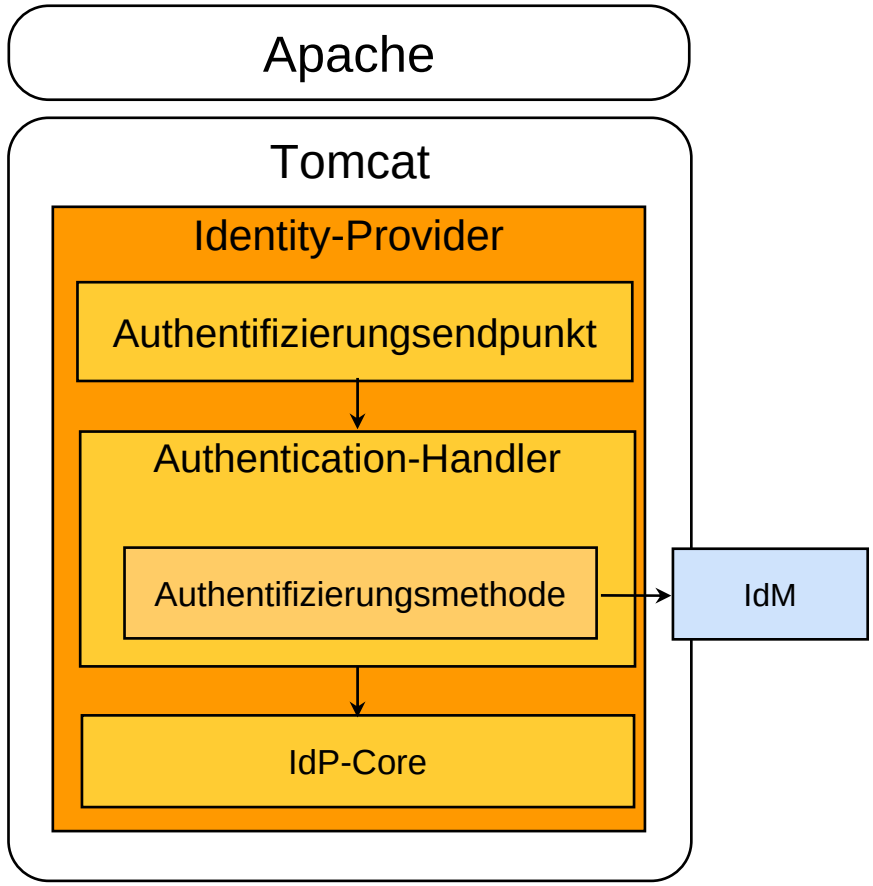
- Unterstützte Betriebssysteme:
  - Linux
  - Mac OS-X
  - Solaris
  - Windows
- Typische Software:
  - Identity Provider:
    - Tomcat 5.5.x/6.0.x
    - Apache 2.2 mit mod_ssl und mod_proxy (Verbindung zum Tomcat)
    - JDK 1.5/1.6
  - Service Provider:
    - Apache + mod_ssl (für HTTPS)

Bestandteile



Authentifizierung IdP

- Bei Shibboleth 2 übernimmt der IdP die Kontrolle über die Authentifizierung.
- Authentifizierung erfolgt über Authentication-Handler, die an verschiedenen Authentifizierungsendpunkten horchen:
  - *UsernamePassword*
  - *IPAddress*
  - *RemoteUser* (ähnlich wie beim IdP 1.3)
- Eigene Authentifizierungsverfahren können implementiert werden, indem man z.B. eine eigene Authentifizierungsmethode programmiert und diese am RemoteUser Authentication-Handler integriert



Authentifizierung Service-Provider

- Service-Provider kann
  - vorgeben, welche Authentifizierungsmethode verwendet werden darf
  - verlangen, dass der Benutzer sich erneut authentifiziert (Identity-Provider muss dies unterstützen)
  - verlangen, dass am Identity-Provider keine Interaktion mit dem Benutzer erfolgt (passive Authentifizierung)

Attribute

- **Attribute** bilden die Grundlage für die **Autorisierung und Zugriffskontrolle** in Shibboleth:
  - Identity-Provider stellen mit Attributen die notwendigen Informationen über ihre Benutzer zur Verfügung.
  - Service-Provider werten die Attribute anhand ihrer Regeln aus und gestatten oder verweigern je nach Ergebnis den Zugriff.
- Hierfür sind **Absprachen zwischen Identity- und Service-Providern** notwendig, die durch Verwendung eines einheitlichen Schemas vereinfacht werden

eduPerson-Schema

- Schemata legen eine Menge von Attributen, die zulässigen Werte und deren Bedeutung fest.
- **InCommon** empfiehlt das auf eduPerson basierende Schema für den Austausch von Attributen: http://www.incommonfederation.org/docs/policies/federatedattributes.pdf
- **Föderationen** und **internationale Anbieter** orientieren sich üblicherweise an dieses Schema:
  - DFN-AAI
  - SWITCH (swissEduPerson)
  - HAKA (funetEduPerson)

eduPerson-Schema

- **Service-Provider** kommen in den meisten Fällen mit folgenden **Attributen** aus:
  - *eduPersonAffiliation*
  - ***eduPersonEntitlement***
  - *eduPersonPrincipalName*
  - *eduPersonTargetedID*

eduPersonScopedAffiliation

- Ermöglicht die Zuordnung der Nutzer einer Einrichtung zu einigen grundlegenden Rollen.
- Zulässige Werte sind: member, faculty, staff, employee, student, alum, affiliate, library-walk-in.
- Beispiel: member@uni-freiburg.de
- Probleme:
  - Die Bedeutung der Werte ist auf internationaler Ebene nicht einheitlich festgelegt (z.B.: Was ist ein *student*?)

eduPersonEntitlement

- Ermöglicht den Austausch beliebiger Benutzerinformationen.
- Zulässige Werte: URIs, d.h. URNs oder URLs, wobei meistens URNs verwendet werden.
- Die Bedeutung der Entitlement-Werte muss zwischen Identity- und Service-Providern abgesprochen werden
- Absprachen auf Föderationsebene oder sogar föderationsübergreifend sind wünschenswert!

eduPersonEntitlement

- Wichtigster Entitlementwert im Bibliotheksbereich: urn:mace:dir:entitlement:common-lib-terms
- Bedeutung: „Nutzer ist berechtigt, die von seiner Einrichtung im Rahmen einer Standardlizenz lizenzierten Inhalte zu nutzen“ (bei Hochschulen: Mitglieder/Angehörige der Hochschule oder *Walk-in Patrons*).
- Die meisten (kommerziellen) Anbieter unterstützen oder erwarten sogar, dass dieser Entitlementwert in Standardfällen verwendet wird.

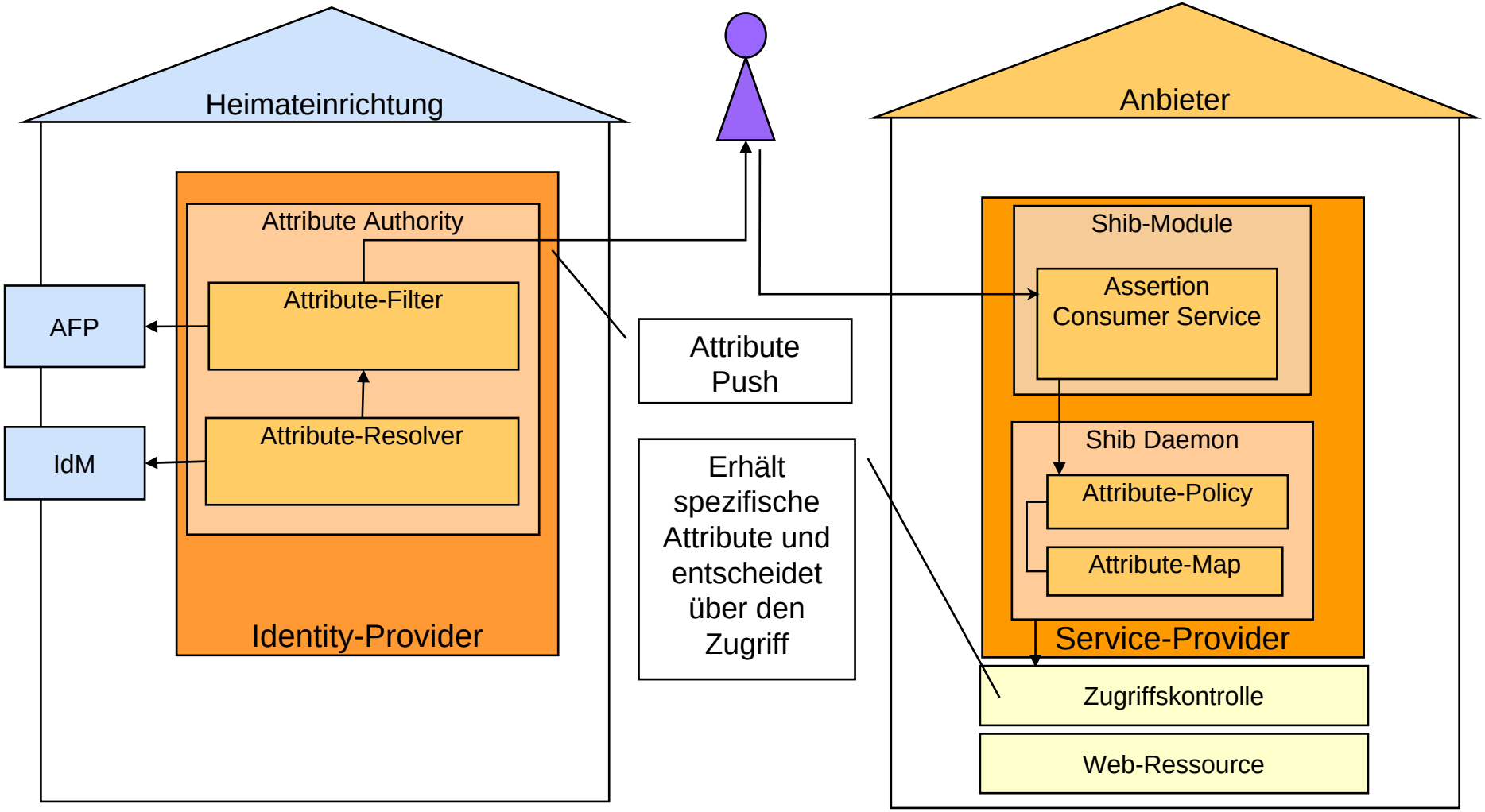
eduPersonPrincipalName

- Eindeutige, persistente Identität des Nutzers inklusive Domain („NetID“).
- Beispiel: borel@uni-freiburg.de
- Sollte aus Datenschutzgründen nur verwendet werden, wenn die Nutzung eines Dienstes nicht anonym oder pseudonym erfolgen kann!
- Beispiel: Schreibender Zugriff auf eine Anwendung, z.B. ein Wiki oder Forum, für den der Nutzer sich zu erkennen geben muss.

eduPersonTargetedID

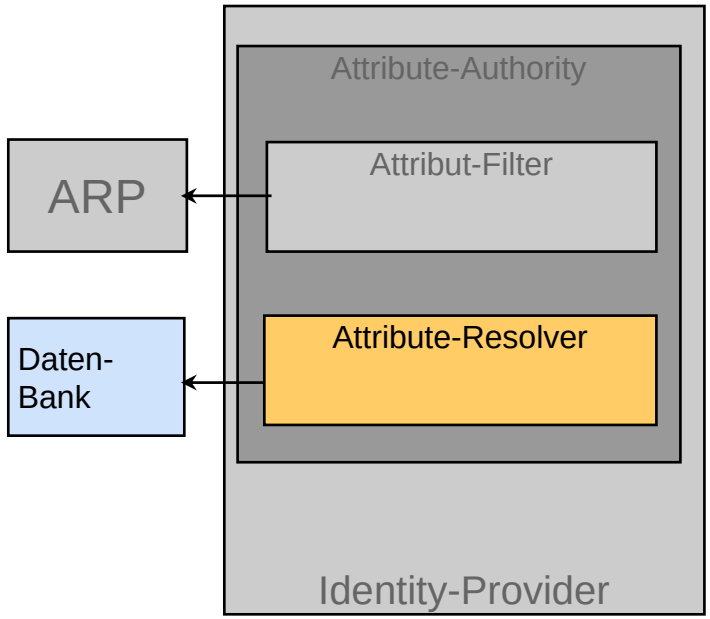
- Eindeutiges, persistentes Pseudonym des Nutzers für einen Service-Provider.
- Ermöglicht die Wiedererkennung des Nutzers (z.B. für personalisierte Anwendungen), ohne seine Identität kennen zu müssen.

Attribute und Zugriffskontrolle



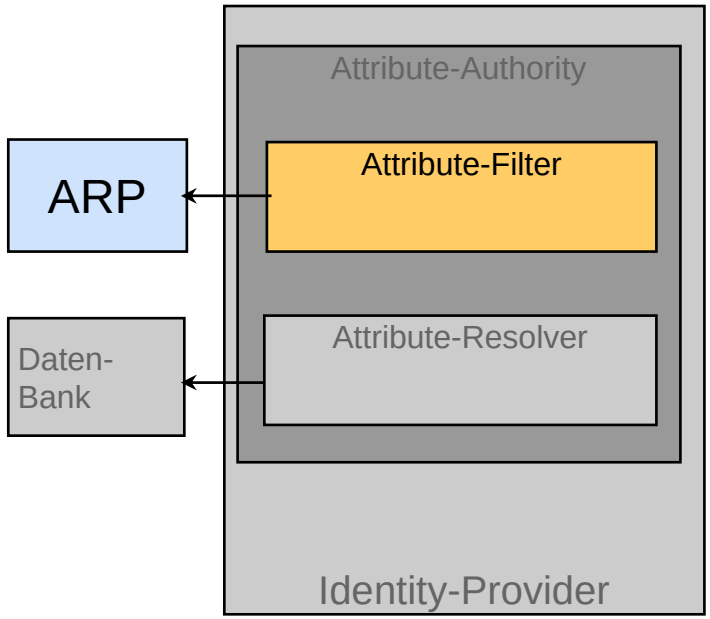
Attribute-Resolver

- *Attribute-Resolver*
  - erzeugt Attributliste
  - Konnektoren:
    - statischer Konnektor
    - Datenbank-Konnektor
    - LDAP-Konnektor
    - *Stored ID Data Connector*: eindeutige, persistente und opake Kennung für den Benutzer
  - Bei Bedarf können eigene Konnektoren eingebunden werden
  - Übernimmt das Encoding (Base64, XMLObject)



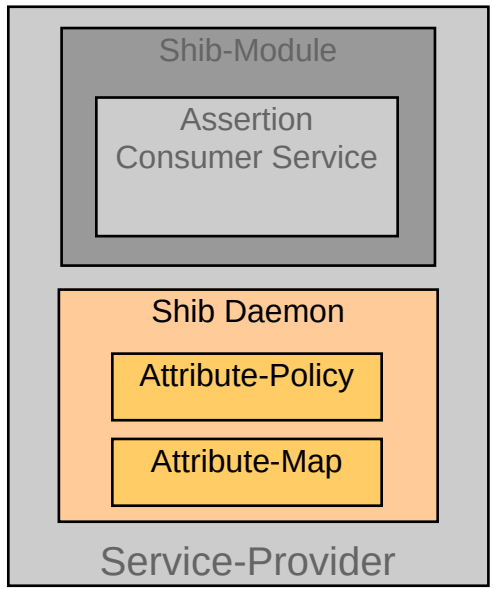
Attribute-Filter

- Attribut-Filter
  - Liste der benötigten Attribute
  - filtert Attribute und Attributwerte
  - filtert NameIDs abhängig von der Relying-Party
  - Möglichkeit eigene Filter zu definieren
- Attribute Release Policies (ARP)
  - Benutzergruppen
  - Gruppen von Service-Provider



Attribute filtern im SP

- Die ***Attribute-Policy*** des Service-Providers legt fest welche **Attribute und Attributwerte** von welchen Identity-Providern **akzeptiert** werden
- Die ***Attribute-Map*** legt fest wie diese für die **Zugriffskontrolle** an den Ressource-Manager bzw. an die Anwendung **weitergegeben** werden



Zugriffskontrolle

- Die vom IdP gelieferten und über die Attribute-Policy/Attribute-Mapping **aufbereiteten Attribute** bilden die **Basis für die Zugriffskontrolle**
- **Shibboleth** bietet **Ressource-Manager:**
  - für Apache (Apache-based access Control über mod_shib)
  - XML-Access Control (Regeln werden auf *Host* und *Path angewandt*)
- Alternative ist ein **eigener Ressource-Manager** in der Anwendung: Attribute werden über HTTP-Header an die Anwendung übergeben.
  - Beispiel ReDI: eduPersonEntitlements werden auf ReDI eigene Benutzergruppen abgebildet.

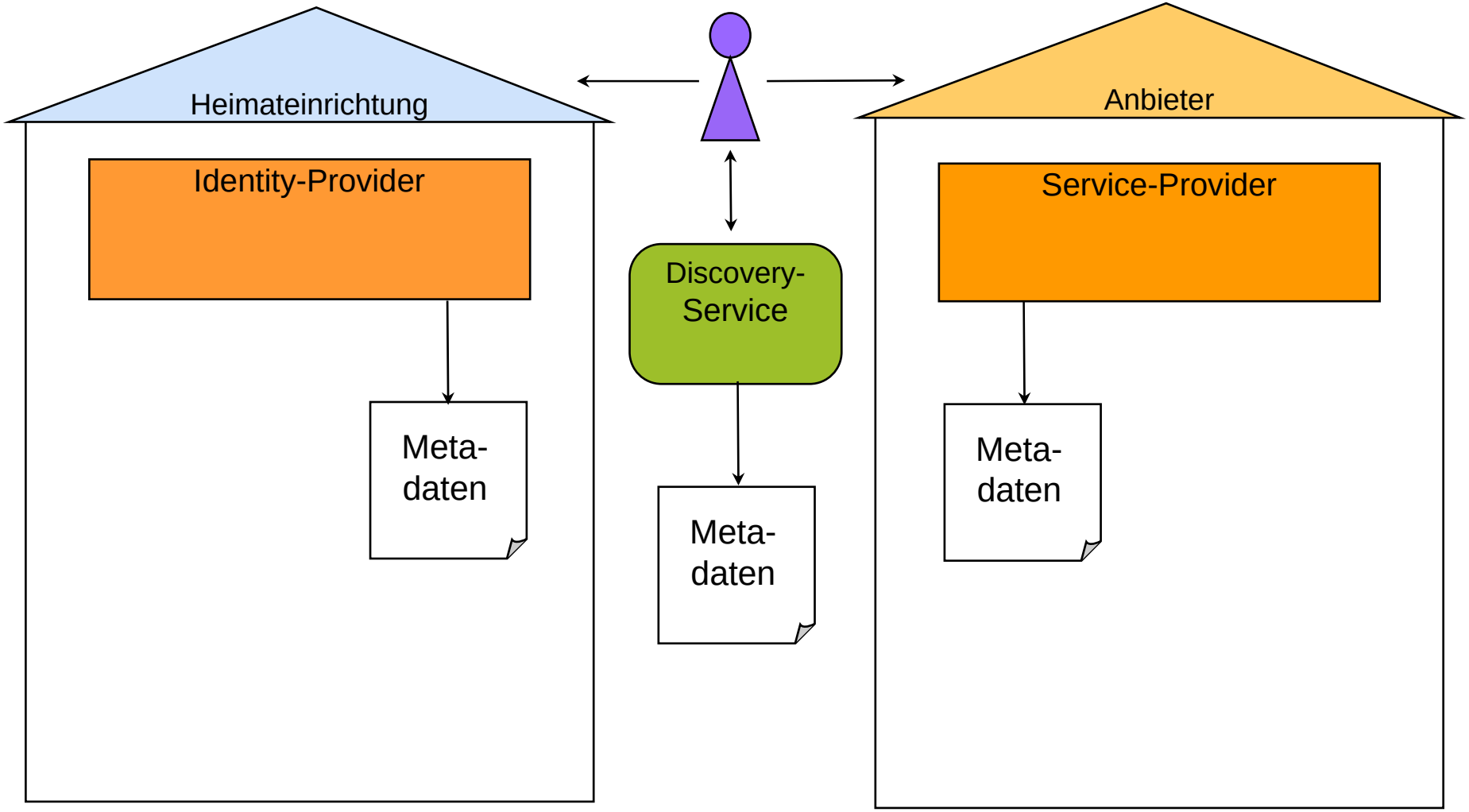
Discovery-Service

- Discovery-Service ist ein Java Servlet und wird offiziell unterstützt
- Discovery-Service unterstützt
  - SAML2 Discovery-Service-Protokoll
  - Shibboleth 1.3 WAYF-Protokoll
  - mehrere Förderationen
  - Plugins zur Filterung der IdP-Listen

Metadaten

- Die Metadaten bilden die Föderation auf technischer Ebene ab
- Sie beinhalten eine vollständige Liste aller teilnehmenden Provider (IdP, SP)
- Zertifikate und *entityID* garantieren, dass die Provider immer wissen, wer gerade mit Ihm spricht
- Metadaten werden signiert, um Ihre Authentizität und Integrität zu gewährleisten
- Metadaten müssen aktuell sein, sonst klappt die Interoperabilität nicht
- Beispiel: http://aar.vascoda.de/test/DEMO2-metadata.xml

Metadaten


Heimateinrichtung
Identity-Provider
Meta-
daten
Discovery-
Service
Meta-
daten
Anbieter
Service-Provider
Meta-
daten

Danke für Ihre Aufmerksamkeit!

Fragen?

AAR ist ein Projekt der UB Freiburg

Gefördert vom BMBF (PT-NMB+F)

http://aar.vascoda.de

borel@ub.uni-freiburg.de