deutsches forschungsnetz

Shibboleth IdP 4.x konfigurativ vorbereiten auf Version 5.x

DFN-AAI Workshop, 14. Juni 2023

Silke Meyer (smeyer@dfn.de)

Überblick

- Rückblick auf die wichtigsten Änderungen in den 4.x-Releases
- Ausblick auf den IdP 5.x
- Empfehlung zur Vorgehensweise

Aktueller Blick auf die DFN-AAI

Shib IdP-Version	Anzahl der IdPs	keine bekannten Sicherheitsprobleme	offiziell supportet
4.0.x	35		
4.1.x	50		
4.2.1	133	√	
4.3.0	32		
4.3.1	93	√	√
unbekannt	21		
andere Software	28		

IdP 4.x Security Advisories

Rückblick auf die letzten Releases

Shibboleth IdP 4.0.x

- Einführung der Passwortdatei `credentials/secrets.properties`
- Umstellung des Verschlüsselungsalgorithmus auf AES-GCM bei Neuinstallationen (Doku)
- Einführung der Attribute Registry
  - Umarbeiten der `conf/attribute-resolver.xml`
  - Anschalten in `conf/services.xml`
  - Anleitung

Umarbeiten der Attribut-Definitionen

- 3er-Syntax:

```
<AttributeDefinition xsi:type="Simple" id="mail">
    <InputDataConnector ref="myLDAP" attributeNames="email"/>
    <DisplayName xml:lang="en">E-mail</DisplayName>
    <DisplayName xml:lang="de">E-Mail</DisplayName>
    <DisplayDescription xml:lang="en">E-Mail address</DisplayDescription>
    <DisplayDescription xml:lang="de">E-Mail Adresse</DisplayDescription>
    <AttributeEncoder xsi:type="SAML2String"
        name="urn:oid:0.9.2342.19200300.100.1.3" friendlyName="mail"
        encodeType="false" />
</AttributeDefinition>
```

Umarbeiten der Attribut-Definitionen

- 4er-Syntax:
- conf/attribute-resolver.xml

```
<AttributeDefinition xsi:type="Simple" id="mail">
    <InputDataConnector ref="myLDAP" attributeNames="email"/>
</AttributeDefinition>
```

- conf/attributes/${schema}.xml (Standardattribute werden mitgeliefert)

```
<bean parent="shibboleth.TranscodingProperties">
    <property name="properties">
        <props merge="true">
            <prop key="id">mail</prop>
            <prop key="transcoder">SAML2StringTranscoder</prop>
            <prop key="saml2.name">urn:oid:0.9.2342.19200300.100.1.3</prop>
            <prop key="displayName.en">E-mail</prop>
            <prop key="displayName.de">E-Mail</prop>
            <prop key="description.en">E-Mail address</prop>
            <prop key="description.de">E-Mail-Adresse</prop>
        </props>
    </property>
</bean>
```

Shibboleth IdP 4.1.x

- Release mit vielen neuen Funktionalitäten
  - Einführung von Plugins und Modulen → schlankerer Dateibaum
- „Hello World"-Plugin: Test der Attribut-Auflösung am IdP ohne SP
  - Zugriff z.B. analog zum Zugriff auf die Admin-Interfaces konfigurieren:

    `conf/admin/admin.properties:`

    ```
    idp.hello.accessPolicy = AccessByIPAddress
    ```
- Hier war unsere Empfehlung die Neuinstallation zur Bereinigung des Dateibaumes.
- für Plugins relevante Änderung in `conf/idp.properties`:

```
idp.searchForProperties=true
```

Shibboleth IdP 4.2.x

- neues Aussehen der Standard-Webinterfaces
- vereinfachte Handhabung von Plugins, z.B.:
  - Installation `./bin/plugin.sh -I net.shibboleth.idp.plugin.xxx`
  - Upgrade `./bin/plugin.sh -u net.shibboleth.idp.plugin.xxx`

Shibboleth IdP 4.3.x

- Deprecation Warnings zu HTTPServletObjects

```
WARN [DEPRECATED:135] - Java class method 'EvaluableScript(parameters...)': This will be
removed in the next major version of this software; replacement is by using the setters
WARN [DEPRECATED:128] - Java class method 'setHttpServletResponse',
(net.shibboleth.idp.plugin.oidc.op.encoding.impl.NimbusResponseEncoder): This will be removed
in the next major version of this software; replacement is setHttpServletResponseSupplier
```

- aus den 4.3.0 Release Notes
- aus der IdP 5.x-Dokumentation

Shibboleth IdP 5.x – Was ist schon bekannt?

Systemanforderungen für Shib IdP 5.x

- Java 17
  - Amazon Corretto 17 für Linux oder Windows
  - Red Hats OpenJDK 17 (RHEL / CentOS)
  - Debian/Ubuntu OpenJDK 17 (wie immer „teilweise unterstützt“) → Repos
- Servlet Container: Servlet API 5.0+
  - Tomcat 10+ oder Jetty 11+
  - aus normalen Paketquellen kommt Tomcat 10.1 ab
    - Debian 12 (geplantes Release: 10. Juni 2023)
    - Ubuntu 24.04 LTS (bzw. ab 23.04)

eduPersonTargetedID

- Deprecation Warning seit 4.0.x

```
WARN [DEPRECATED:125] - xsi:type 'SAML2NameID', (file
  [/opt/shibboleth-idp/conf/attribute-resolver.xml]): This will be removed in
  the next major version of this software; replacement is (none)
```

- ab 5.x voraussichtlich kein offizielles Features mehr, das irgendwo erwähnt wird, es geht aber auch nichts kaputt
- Shibboleth-Entwickler Cantor: Bitte testhalber auf persistentID umstellen, bei den meisten (Shibboleth) SPs wird das wohl funktionieren.

Nashorn Plugin

- Ab JDK 15 (also auch in JDK 17) fehlt die mitgelieferte Javascript Scripting Engine.
- Scripted Attributes in `conf/attribute-resolver.xml`
- IdP-Plugin Nashorn muss installiert werden:

```
/opt/shibboleth-idp/bin/plugin.sh -I net.shibboleth.idp.plugin.nashorn
```

- keine weitere Konfiguration

Bild: Erich Ferdinand, Dortmunder Nashorn - Lady Rhino, CC BY 2.0

Datenbank-Zugriff über JDBC Plugin

- Ersetzen des JPA Storage Service durch JDBC Storage Service
- direkter Zugriff auf DB (statt über Hibernate ORM)
- Spalten ‚context' und ‚id' müssen case-sensitive behandelt werden (Collation)
- IdP-Plugin JDBCStorageService muss installiert werden:

```
/opt/shibboleth-idp/bin/plugin.sh -I net.shibboleth.plugin.storage.jdbc
```

- wie vorher: Java-DB-Treiber muss installiert sein (z.B. `libmariadb-java`)
- Konfiguration in `conf/global.xml`
- `idp.properties`: `JDBCStorageService` statt `shibboleth.JPAStorageService`

Collation prüfen

- _ci = case-insensitive versus _cs oder utf8 = case-sensitive
- MariaDB [idp]> show create table StorageRecords\G;

```
*************************** 1. row ***************************
Table: StorageRecords
Create Table: CREATE TABLE `StorageRecords` (
  `context` varchar(255) NOT NULL,
  `id` varchar(255) NOT NULL,
  `expires` bigint(20) DEFAULT NULL,
  `value` longtext NOT NULL,
  `version` bigint(20) NOT NULL,
  PRIMARY KEY (`context`,`id`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_bin          (ab MariaDB 10.6 utf8mb3_bin)
1 row in set (0.000 sec)
```

Collation ändern

- MariaDB [idp]> ALTER TABLE StorageRecords CONVERT TO CHARACTER SET utf8 COLLATE utf8_bin;
- Details zur Konvertierung: MariaDB-Dokumentation

global.xml

global.xml mit JPAStorageService (alt)

```
        <bean id="shibboleth.MySQLDataSource"
              class="%{mysql.class}"
              p:driverClassName="org.mariadb.jdbc.Driver"
              p:url="%{mysql.url}"
              p:username="%{mysql.username}"
              p:password="%{mysql.password}"
              p:maxWait="15000"
              p:testOnBorrow="true"
              p:maxActive="100"
              p:maxIdle="100"
              p:validationQuery="select 1"
              p:validationQueryTimeout="5" />

        <bean id="shibboleth.JPAStorageService"
              class="org.opensaml.storage.impl.JPAStorageService"
              p:cleanupInterval="%{idp.storage.cleanupInterval:PT10M}"
              c:factory-ref="shibboleth.JPAStorageService.EntityManagerFactory" />

        <bean id="shibboleth.JPAStorageService.EntityManagerFactory"
              class="org.springframework.orm.jpa.LocalContainerEntityManagerFactoryBean">
            <property name="packagesToScan" value="org.opensaml.storage.impl"/>
            <property name="dataSource" ref="shibboleth.MySQLDataSource"/>
            <property name="jpaVendorAdapter"
ref="shibboleth.JPAStorageService.JPAVendorAdapter"/>
            <property name="jpaDialect">
                <bean class="org.springframework.orm.jpa.vendor.HibernateJpaDialect" />
            </property>
        </bean>

        <bean id="shibboleth.JPAStorageService.JPAVendorAdapter"
              class="org.springframework.orm.jpa.vendor.HibernateJpaVendorAdapter"
              p:generateDdl="true"
              p:database="MYSQL"
              p:databasePlatform="org.hibernate.dialect.MySQL5Dialect" />
```

global.xml mit JDBCStorageService (neu)

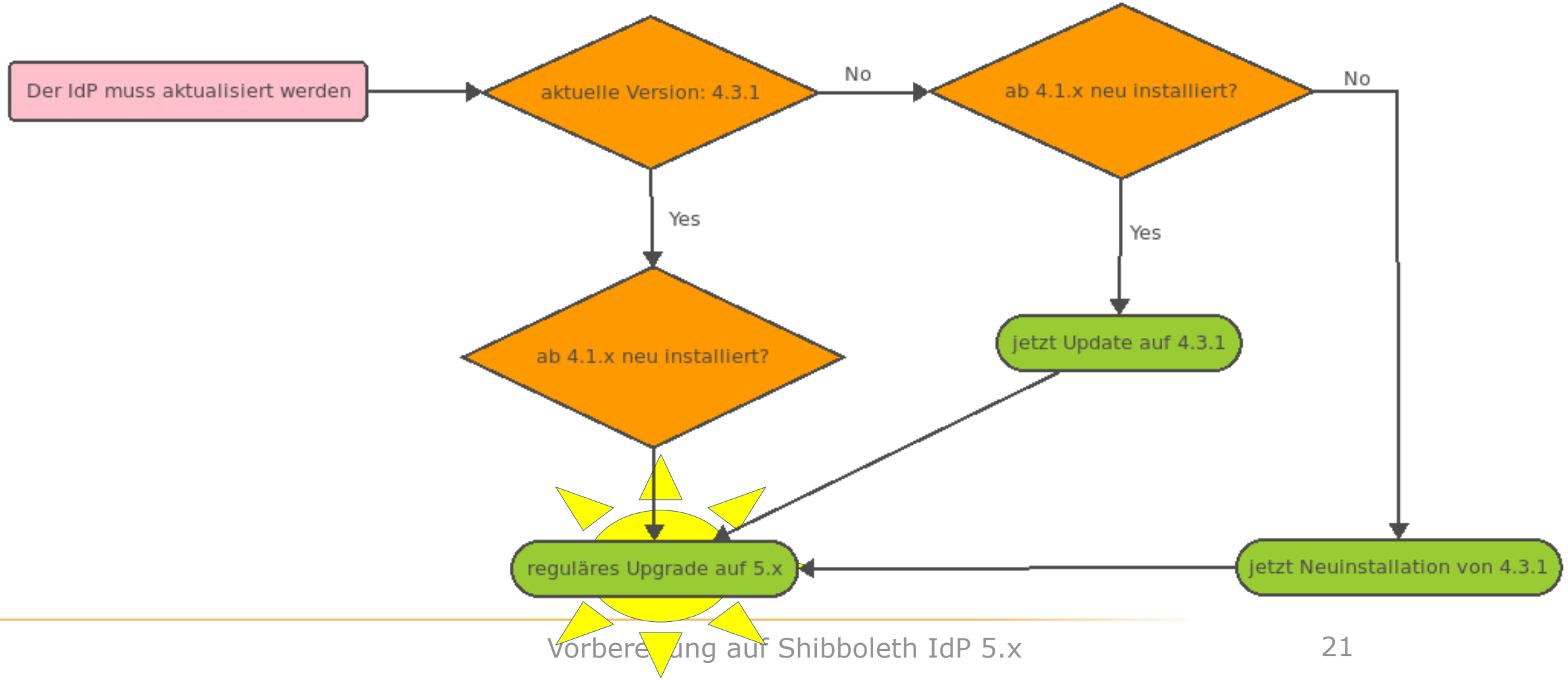
```
<bean id="shibboleth.MySQLDataSource"
                class="%{mysql.class}"
              p:driverClassName="org.mariadb.jdbc.Driver"
              p:url="%{mysql.url}"
              p:username="%{mysql.username}"
              p:password="%{mysql.password}"
              p:maxWait="15000"
              p:testOnBorrow="true"
              p:maxActive="100"
              p:maxIdle="100"
              p:validationQuery="select 1"
              p:validationQueryTimeout="5" />

        <bean id="JDBCStorageService"
              parent="shibboleth.JDBCStorageService"
              p:cleanupInterval="%
{idp.storage.cleanupInterval:PT10M}"
              p:dataSource-ref="shibboleth.MySQLDataSource" />

        <!-- (Rest entfällt) -->
```

Voraussichtliches Vorgehen beim Upgrade

- zuerst die Plugins aktualisieren
- dann wie gewohnt: Backup machen und „Drüberinstallieren“
- wie schon bei früheren Major Upgrades: **nicht** das alte `conf`-Verzeichnis in eine separate Neuinstallation kopieren
- unfertiger Entwurf für die v5 Release Notes, darin noch u.a.:
  - Änderung am Metadatenfilter „EntityRoleWhiteList“
  - Änderungen bei Property-Parametern, die bei der nicht-interaktiven Installation mitgegeben werden

Empfehlungen (Stand Juni 2023)

- alle: wegen Security Advisories so bald wie möglich auf 4.3.1 aktualisieren



Vielen Dank! Gibt‘s Fragen?

- Kontakt
  - DFN-AAI Team

    E-Mail: hotline@aai.dfn.de
    Tel.: +49-30-884299-9124
    Fax: +49-30-884299-370

    Anschrift:
    DFN-Verein, Geschäftsstelle
    Alexanderplatz 1
    10178 Berlin