Trucs et astuces

- Ajouter un administrateur
- Changer de groupe Administrators
- Configuration Shib
  - login
  - logout
- Accès direct Shib
- Saisie d'un groupe Shib complexe

Ajouter un administrateur

Une fois authentifier shib je suis connu dans nuxeo comme **bourges@univ-rennes1.fr**

Pour faire en sorte que je sois reconnu comme administrateur de la plateforme j'ai ajouté ceci dans dans le fichier **shibboleth-config.xml** :

```
<extension target="org.nuxeo.ecm.platform.usermanager.UserService" point="userManager">
 <userManager class="org.nuxeo.ecm.platform.usermanager.UserManagerWithComputedGroups">
  <defaultAdministratorId>bourges@univ-rennes1.fr</defaultAdministratorId>
  <defaultGroup>members</defaultGroup>
 </userManager>
</extension>
```

⚠ l'attribut **class="org.nuxeo.ecm.platform.usermanager.UserManagerWithComputedGroups"** est important afin de forcer nuxeo à utiliser la bonne implémentation de UserManager. **UserManagerWithComputedGroups** est celui qui est utilisé pour analyser les groupes Shibboleth.

Changer de groupe Administrators

Il est possible de brancher un groupe LDAP spécifique en tant que groupe pouvant administrer nuxeo. Exemple :

```
<extension target="org.nuxeo.ecm.platform.usermanager.UserService" point="userManager">
 <userManager>
  <administratorsGroup>groupes:applis:ORI-OAI:admins</administratorsGroup>
  <defaultGroup>members</defaultGroup>
 </userManager>
</extension>
?
```

⚠ Ici la balise **administratorsGroup** pointe vers un groupe LDAP généré par Grouper.

Configuration Shib

login

Dans dans le fichier **shibboleth-config.xml** :

```
<loginURL>http://sp-test3.univ-rennes1.fr/Shibboleth.sso/wayf</loginURL>
```

/wayf correspond à la définition SessionInitiator de type WAYF défini dans la la configuration du SP. Exemple :

```
<SessionInitiator type="Chaining" Location="/wayf" id="WAYF" relayState="cookie">
 <SessionInitiator type="WAYF" acsIndex="5" URL="https://services-federation.renater.fr/test/wayf"/>
</SessionInitiator>
```

logout

Dans dans le fichier **shibboleth-config.xml** :

```
<logoutURL>http://sp-test3.univ-rennes1.fr/Shibboleth.sso/Logout</logoutURL>
```

/Logout correspond à la définition LogoutInitiator de type Local défini dans la la configuration du SP. Exemple :

```
<LogoutInitiator type="Chaining" Location="/Logout">
 <LogoutInitiator type="Local"/>
</LogoutInitiator>
```

Accès direct Shib

Qui dit Shib dit WAYF. Mais quand on est dans un ENT d'un établissement et que :

- l'on sait de quel établissement vient l'utilisateur
- que l'utilisateur est déjà connecté

Alors il est particulièrement intéressant d'accéder à nuxeo sans passer à nouveau par le WAYF. Heureusement il est possible de court-circuiter le WAYF cf. https://federation.renater.fr/faq/wayf#court-circuiter_le_wayf

> ⚠ Plus exactement on passe par le WAYF mais il n'y a pas d'interaction avec l'utilisateur.

Voici un exemple d'URL permettant d'accéder à un workspace donné sans passé par le WAYF (et le SSO protégeant l'IdP si l'utilisateur est déjà connecté) :

```
https://ident-shib-test.univ-rennes1.fr/profile/Shibboleth/SSO?shire=http://sp-test3.univ-rennes1.fr/Shibboleth.
sso/SAML/POST&target=http://sp-test3.univ-rennes1.fr/nuxeo/nxdoc/default/8f336444-60b5-48fa-9c03-a4c3335f8575
/view_documents%3FtabId%3D%26conversationId%3D0NXMAIN&providerId=http://sp-test3.univ-rennes1.fr
```

Saisie d'un groupe Shib complexe

Il est possible de saisir la définition du groupe Shib complexe comme celle-ci :

```
(currentUser.user.affiliation=='student' and
currentUser.user.username.matches('^g')) or
(currentUser.user.username.contains('jaune')) and (not
(currentUser.user.email.matches('.*@etu.u-bordeaux1.fr') or
currentUser.user.company==('CNRS')))
```

On voit ici que l'on utilise **user.affiliation** alors que **affiliation** n'est pas un attribut standard de l'objet **user** de nuxeo. Il nous faut donc surcharger la définition de l'objet user de nuxeo. Pour cela nous créons un fichier **userShib.sxd** comme celui-ci :

```
<?xml version="1.0"?>
<xs:schema targetNamespace="http://www.nuxeo.org/ecm/schemas/user"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns:nxs="http://www.nuxeo.org/ecm/schemas/user">

  <xs:include schemaLocation="base.xsd" />

  <xs:element name="username" type="xs:string" />
  <xs:element name="password" type="xs:string" />
  <xs:element name="firstName" type="xs:string" />
  <xs:element name="lastName" type="xs:string" />
  <xs:element name="company" type="xs:string" />
  <xs:element name="email" type="xs:string" />
  <xs:element name="affiliation" type="xs:string" />

  <!-- inverse reference -->
  <xs:element name="groups" type="nxs:stringList" />

</xs:schema>
```

Le fichier est a placer dans **nxserver/config**. On y voit la définition de **<xs:element name="affiliation" type="xs:string" />**. Il faut ensuite référencer ce nouveau shéma :

```
  <extension target="org.nuxeo.ecm.core.schema.TypeService" point="schema">
    <schema name="user" src="userShib.xsd" override="true" />
  </extension
```

Attention au **override="true"** qui est indispensable si l'on veut surcharger la définition de **user** utilisé par défaut par nuxeo.