7) configuration shib + LDAP fonctionnelle

Configuration

ldap-config.xml

Il est dans nxser/config et contient :

```
<component name="sample.ldap.config">

  <require>org.nuxeo.ecm.directory.ldap.LDAPDirectoryFactory</require>
  <require>org.nuxeo.ecm.directory.sql.storage</require>

  <extension target="org.nuxeo.ecm.directory.multi.MultiDirectoryFactory"
    point="directories">
    <directory name="userDirectory">
      <schema>user</schema>
      <idField>username</idField>
      <readOnly>false</readOnly>
      <passwordField>password</passwordField>
      <source name="ldapUserDirectory">
        <subDirectory name="ldapUserDirectory"/>
      </source>
      <source name="sqlUserDirectory" creation="true">
        <subDirectory name="sqlUserDirectory"/>
      </source>
    </directory>
  </extension>

  <extension target="org.nuxeo.ecm.directory.multi.MultiDirectoryFactory"
    point="directories">
    <directory name="groupDirectory">
      <schema>group</schema>
      <idField>groupname</idField>
      <readOnly>false</readOnly>
      <source name="ldapGroupDirectory">
        <subDirectory name="ldapGroupDirectory"/>
      </source>
      <source name="sqlGroupDirectory" creation="true">
        <subDirectory name="sqlGroupDirectory"/>
      </source>
    </directory>
  </extension>

  <extension target="org.nuxeo.ecm.directory.ldap.LDAPDirectoryFactory"
    point="servers">
    <server name="default">
      <ldapUrl>ldap://ldap.univ.fr:389</ldapUrl>
    </server>
  </extension>

  <extension target="org.nuxeo.ecm.directory.ldap.LDAPDirectoryFactory"
    point="directories">
    <directory name="ldapUserDirectory">
      <server>default</server>
      <schema>user</schema>
      <idField>username</idField>
      <passwordField>password</passwordField>
```

```
      <searchBaseDn>ou=people,dc=univ-rennes1,dc=fr</searchBaseDn>
      <searchClass>person</searchClass>
      <searchScope>onelevel</searchScope>
      <readOnly>true</readOnly>
      <cacheTimeout>3600</cacheTimeout>
      <cacheMaxSize>100000</cacheMaxSize>
      <creationBaseDn>ou=people,dc=univ-rennes1,dc=fr</creationBaseDn>
      <creationClass>top</creationClass>
      <creationClass>person</creationClass>
      <creationClass>organizationalPerson</creationClass>
      <creationClass>inetOrgPerson</creationClass>
      <rdnAttribute>uid</rdnAttribute>
      <fieldMapping name="username">uid</fieldMapping>
      <fieldMapping name="firstName">givenName</fieldMapping>
      <fieldMapping name="lastName">sn</fieldMapping>
      <fieldMapping name="company">supannetablissement</fieldMapping>
      <fieldMapping name="email">mail</fieldMapping>
      <references>
        <inverseReference field="groups" directory="ldapGroupDirectory"
          dualReferenceField="members" />
      </references>
    </directory>
  </extension>

  <extension target="org.nuxeo.ecm.directory.ldap.LDAPDirectoryFactory"
    point="directories">
    <directory name="ldapGroupDirectory">
      <server>default</server>
      <schema>group</schema>
      <idField>groupname</idField>
      <searchBaseDn>ou=groups,dc=univ-rennes1,dc=fr</searchBaseDn>
      <searchFilter>(|(objectClass=groupOfNames)(objectClass=groupOfURLs))</searchFilter>
      <searchScope>subtree</searchScope>
      <readOnly>true</readOnly>
      <cacheTimeout>3600</cacheTimeout>
      <cacheMaxSize>10000</cacheMaxSize>
      <querySizeLimit>10000</querySizeLimit>
      <creationBaseDn>ou=groups,dc=univ-rennes1,dc=fr</creationBaseDn>
      <creationClass>top</creationClass>
      <creationClass>groupOfUniqueNames</creationClass>
      <rdnAttribute>cn</rdnAttribute>
      <fieldMapping name="groupname">cn</fieldMapping>
      <references>
        <ldapReference field="members" directory="ldapUserDirectory"
          forceDnConsistencyCheck="false"
          staticAttributeId="member"
          dynamicAttributeId="memberURL" />
        <ldapReference field="subGroups" directory="ldapGroupDirectory"
          forceDnConsistencyCheck="false"
          staticAttributeId="uniqueMember"
          dynamicAttributeId="memberURL" />
        <inverseReference field="parentGroups"
          directory="groupDirectory" dualReferenceField="subGroups" />
      </references>
    </directory>
  </extension>

</component>
```

Explications

Ce fichier définit deux choses :

1. Configuration d'un gestionnaire d'annuaires multiples (LDAP et SQL)

   ⚠ Il va notamment nous permettre d'enregistré manuellement des users et groupes à la main. Ex : un groupe non LDAP contenant des groupes ldap et/ou shib ou un user avec son email comme id afin de l'autoriser sur une ressource avant même que ce dernier se soit connecté une première fois. De plus, les infos sur le user comme le nom et le prénom seront automatiquement actualisé à la connexion.

2. Configuration de l'annuaire LDAP (user et groupes)

A noter :

- Les sources SQL et LDAP sont toutes préfixés par **sql** ou **ldap**. Exemple : **groupDirectory** devient **ldapGroupDirectory**. Ceci permet de ne définir qu'une fois **groupDirectory** au niveau du gestionnaire d'annuaires multiples
- **<passwordField>password</passwordField>** est indispensable si on veut pouvoir utiliser l'annuaire pour une identification pas user/password

- **creation="true"** me permet de dire que la base de données nuxeo peut être mise à jour. Indispensable quand un utilisateur distant se présente et, obligatoirement, n'est pas présent dans le LDAP de l'établissement.

- La balise **querySizeLimit** doit avoir une valeur suffisamment grande (200 étant la valeur par défaut) et être supérieurs au nombre total de groupes contenus dans le LDAP. Si la valeur est trop faible les groupes LDAP ne seront jamais vus dans l'IHM Nuxeo de positionnement des droits.
- **<fieldMapping name="username">eduPersonPrincipalName</fieldMapping>** doit être en phase avec la configuration de l'authentification Shib

shibboleth-groups-config.xml

Il est dans nxser/config et contient :

```
<component name="sample.shibboleth.config">

  <require>org.nuxeo.ecm.platform.ui.web.auth.WebEngineConfig</require>
  <require>org.nuxeo.ecm.platform.usermanager.UserManagerImpl</require>
  <require>org.nuxeo.opensocial.OAuthFilter</require>

  <extension
      target="org.nuxeo.ecm.platform.ui.web.auth.service.PluggableAuthenticationService"
      point="chain">
    <authenticationChain>
      <plugins>
        <plugin>BASIC_AUTH</plugin>
        <plugin>SHIB_AUTH</plugin>
        <plugin>ANONYMOUS_AUTH</plugin>
      </plugins>
    </authenticationChain>
  </extension>

 <extension
    target="org.nuxeo.ecm.platform.shibboleth.service.ShibbolethAuthenticationService"
    point="config">
    <config>
      <uidHeaders>
        <!--uidHeader idpUrl="url1">uid1</uidHeader>
        <uidHeader idpUrl="url2">uid2</uidHeader-->
        <uidHeader idpUrl="https://ident-shib-test.univ-rennes1.fr/shibboleth">uid</uidHeader>
        <default>mail</default>
      </uidHeaders>
      <loginURL>http://sp-test3.univ-rennes1.fr/Shibboleth.sso/wayf</loginURL>
      <logoutURL>http://sp-test3.univ-rennes1.fr/Shibboleth.sso/Logout</logoutURL>

      <fieldMapping header="mail">email</fieldMapping>
      <fieldMapping header="sn">lastName</fieldMapping>
      <fieldMapping header="givenName">firstName</fieldMapping>
      <fieldMapping header="departmentNumber">company</fieldMapping>
    </config>
  </extension>


  <!-- Add an Anonymous user -->
  <extension target="org.nuxeo.ecm.platform.usermanager.UserService"
    point="userManager">
    <userManager class="org.nuxeo.ecm.platform.computedgroups.UserManagerWithComputedGroups">
      <users>
        <anonymousUser id="Guest">
          <property name="firstName">Guest</property>
          <property name="lastName">User</property>
          <property name="email">foo@bidon.fr</property>
        </anonymousUser>
      </users>
    </userManager>
  </extension>

  <extension target="org.nuxeo.ecm.platform.computedgroups.UserService" point="userManager">
    <userManager class="org.nuxeo.ecm.platform.usermanager.UserManagerWithComputedGroups">
      <defaultAdministratorId>bourges</defaultAdministratorId>
      <administratorsGroup>groupes:applis:ORI-OAI:Administrateurs</administratorsGroup>
      <defaultGroup>members</defaultGroup>
    </userManager>
  </extension>



</component>
```

Explications

- **<uidHeader idpUrl="https://ident-shib-test.univ-rennes1.fr/shibboleth">uid</uidHeader>** permet de dire que pour l'IdP de Rennes je vais utiliser uid comme clé
- **<default>mail</default>** me permet de dire que pour tout autre IdP j'utilise l'email comme clé
- Je n'ai pas de **<fieldMapping header="mail">username</fieldMapping>** car nuxeo ne sait pas avoir 2 header avec la même clé !
- **<property name="email">foo@bidon.fr</property>** pour **guest** m'évite des erreurs dans les logs lors de l'accès à la page d'accueil en mode anonyme
- Pour que ça marche il faut un patch dans **nuxeo-platform-login-shibboleth** :

```
--- a/nuxeo-platform-login-shibboleth/src/main/java/org/nuxeo/ecm/platform/shibboleth/auth
/ShibbolethAuthenticationPlugin.java        Tue Nov 09 00:10:12 2010 +0100
+++ b/nuxeo-platform-login-shibboleth/src/main/java/org/nuxeo/ecm/platform/shibboleth/auth
/ShibbolethAuthenticationPlugin.java        Fri Dec 03 17:45:33 2010 +0100
@@ -116,6 +116,9 @@
             Map<String, Object> fieldMap = getService().getUserMetadata(httpRequest);
             DocumentModel entry = userDir.getEntry(userId);
             if (entry == null) {
+                    // patch RB :
+                    fieldMap.put("username", userId);
+                    // fin patch RB
                 userDir.createEntry(fieldMap);
             } else {
                 entry.getDataModel(userManager.getUserSchemaName()).setMap
```

- Ce patch permet 2 choses :
  - Utilisation d'une clé différente de default pour certain IdP
  - Contourner, sur le username, de la limitation de nuxeo qui ne sait pas avoir 2 header avec la même clé

default-sql-directories-bundle.xml

Il est dans templates/common/config (pour surcharger la conf par défaut. NB : Peut-être est-il possible de le mettre ailleurs mais je n'ai pas réussi à le mettre dans custom par exemple) et contient :

```
<?xml version="1.0"?>

<component name="org.nuxeo.ecm.directory.sql.storage">

  <implementation class="org.nuxeo.ecm.directory.sql.SQLDirectoryDescriptor" />

  <require>org.nuxeo.ecm.directory.sql.SQLDirectoryFactory</require>

  <extension target="org.nuxeo.ecm.directory.sql.SQLDirectoryFactory"
    point="directories">

    <directory name="sqlUserDirectory">

      <schema>user</schema>

      <dataSource>jdbc/nxsqldirectory</dataSource>

      <table>users</table>
      <idField>username</idField>
      <passwordField>password</passwordField>
      <passwordHashAlgorithm>SSHA</passwordHashAlgorithm>
      <autoincrementIdField>false</autoincrementIdField>
      <dataFile>users.csv</dataFile>
      <createTablePolicy>on_missing_columns</createTablePolicy>
      <querySizeLimit>15</querySizeLimit>

      <references>
        <inverseReference field="groups" directory="sqlGroupDirectory"
          dualReferenceField="members" />
      </references>

    </directory>

    <directory name="sqlGroupDirectory">

      <schema>group</schema>
      <dataSource>jdbc/nxsqldirectory</dataSource>
      <table>groups</table>
      <idField>groupname</idField>
      <dataFile>groups.csv</dataFile>
      <createTablePolicy>on_missing_columns</createTablePolicy>
      <autoincrementIdField>false</autoincrementIdField>

      <!-- Add 10 min cache to avoid refetching the groups during login -->
      <cacheTimeout>360</cacheTimeout>
      <cacheMaxSize>1000</cacheMaxSize>

      <references>
        <tableReference field="members" directory="sqlUserDirectory"
          table="user2group" sourceColumn="groupId" targetColumn="userId" schema="user2group"
          dataFile="user2group.csv" />
        <!-- Warning ! From Nuxeo 5.3.1, a wrong setting has been fixed. See
        http://jira.nuxeo.org/browse/NXP-4401 . Nuxeo upgrades would need a fix in the
        database (inverting parentGroupId and childGroupId in the group2group)  -->
        <tableReference field="subGroups" directory="sqlGroupDirectory"
          table="group2group" sourceColumn="parentGroupId"
          targetColumn="childGroupId" schema="group2group" />
        <inverseReference field="parentGroups" directory="sqlGroupDirectory"
          dualReferenceField="subGroups" />
      </references>

    </directory>

  </extension>
</component>
```

Explications

Le fichier est retravaillé pour préfixés par **sql** les différentes sources.