

Mines de Saint-Étienne

Contexte

L'Ecole des Mines de Saint-Etienne propose des cartes à ses étudiants qui sont :

- en Mifare Desfire EV2 livrées "pré-encodées" avec l'application IZLY (lescrous)
- affectés aux utilisateurs au travers de l'application de contrôle d'accès utilisé dans l'établissement, le contrôle d'accès écrit dans le même temps une application MiFare Desfire qui lui est dédiée.

Même si les cartes données aux étudiants sont bien encodées avec l'application Izly, elles ne sont pas utilisables pour les services Izly (crous) car non connues d'Izly (crous).

Objectif

Les cartes étant créées, imprimées et encodées par ailleurs, l'objectif est ici d'utiliser esup-sgc uniquement pour **pousser/synchroniser les cartes Mifare Desfire pré-encodées Izly dans le serveur IZLY via l'API lescrous.fr**

Les cartes sont imprimées, encodées et attribuées en dehors de l'environnement d'esup-sgc depuis le système de contrôle d'accès de l'établissement qui fait donc véritablement office de SGC ici.

Du point de vue d'esup-sgc, les cartes de l'établissement sont donc des cartes qui n'ont pas été initiées par lui-même : elles sont 'externes' à esup-sgc.

esup-sgc doit être configuré pour récupérer et envoyer à Izly toutes les informations correspondant à l'individu et 'sa carte izly'.

Seule cette brique "esup-sgc" est à installer ici ; nous pouvons dans ce cas d'usage nous passer d'esup-nfc-tag et des clients.

Carte Izly / SmartCard

Pour les informations spécifiques à la carte izly, l'importation du CSV Izly (récupéré avec l'ensemble des cartes à la livraison) est à réaliser en 1 import :

<https://www.esup-portail.org/wiki/pages/viewpage.action?pageId=614989864#Importationde'cartes'dansESUP-SGC/Migrationdesdonn%C3%A9es-Importdonn%C3%A9esCROUS>

Utilisateur / RightHolder

Les mises à jour des informations des utilisateurs et du CSN de la carte peuvent être réalisées en s'appuyant sur un ldap ou/et tables sql.

Les champs à renseigner parmi ceux listés sur la page de documentation [Configurations ESUP-SGC](#) sont ceux-ci :

eppn	
email	
supannCodeINE	affichage / construction identifiant ESCR
firstname	Affichage / moteur de recherche
name	Affichage / moteur de recherche
schacDateOfBirth	Date de naissance - obligatoire dans les contrôles d'accès
schacExpiryDate	Date de fin de droits – les cartes de l'individu sont marquées comme caduques cette date passée.
referenceStatut	Population crous (psg, etd, prs, hbg, fct, fpa, stg) - permet de calculer le tarif et société crous depuis le fichier ESIST.xml
indice	Indice du personnel - permet de calculer le tarif et société crous depuis le fichier ESIST.xml
supannEtablissement	Code RNE Établissement - permet de calculer le tarif et société crous depuis le fichier ESIST.xml
supannRefId4ExternalCard	supannRefId donnant des numéros de cartes 'externes', cad non issus du SGC (et donc issus d'un autre SGC) Dans ce champ multivalué, on attend une valeur comme {ISO15693}04010203ABCDEF80
eduPersonAffiliation	récupéré depuis ldap, nous permet notamment ici de calculer referenceStatut

Configurations

- Dans applicationContext-crous.xml, en plus de mettre 'true' à la propriété enable de apiCrousService, on met true à use4ExternalCard dans crousService :

```
<bean id="crousService" class="org.esupportail.sgc.services.crous.CrousService">
  <property name="use4ExternalCard" value="true"/>
</bean>
```

- Dans applicationContext-services.xml on a un userInfoService ainsi :

```
<bean id="ldapUserInfoService" class="org.esupportail.sgc.services.userinfos.LdapUserInfoService" p:order="2">
  <property name="eppnFilter" value=".*@emse\.fr"/>
  <property name="ldapTemplate" ref="ldapTemplate"/>
  <property name="sgcParam2ldapAttr">
    <map>
      <entry key="email" value="mail"/>
      <entry key="supannCodeINE" value="supannCodeINE"/>
      <entry key="firstname" value="givenname"/>
      <entry key="name" value="sn"/>
      <entry key="schacDateOfBirth" value="schacDateOfBirth"/>
      <entry key="supannRefId4ExternalCard" value="supannRefId"/>
      <entry key="jpegPhoto4ExternalCard" value="jpegPhoto"/>
      <entry key="schacExpiryDate" value="schacExpiryDate"/>
      <entry key="eduPersonAffiliation" value="eduPersonAffiliation"/>
    </map>
  </property>
</bean>
```

jpegPhoto4ExternalCard est optionnel, cela permet d'afficher la photo dans le sgc

On complète ces informations récupérées depuis ldap par des règles ainsi :

```
<bean id="spelUserInfoService" class="org.esupportail.sgc.services.userinfos.SpelUserInfoService" p:order="3">
  <property name="sgcParam2spelExp">
    <map>
      <entry key="referenceStatut" value="#userInfosInComputing['eduPersonAffiliation'] !=
Null and #userInfosInComputing['eduPersonAffiliation'].contains('student') ? 'etd' : 'psg' "/>
      <entry key="indice" value="'0' "/>
      <entry key="supannEtablissement" value="'{UAI}0420094Z' "/>
    </map>
  </property>
</bean>
```

Enfin afin que l'utilisateur et sa (ou ses) carte(s) soient considérés comme 'caduques' dès lors qu'il ne figure plus dans le ldap, on paramètre le caducIfEmpty en mettant un champ utilisateur correspondant à attribut présent pour toute personne dans le ldap de l'emse (le mail et donc email ici ; d'autant que celui-ci est obligatoire côté lzly) :

```
<bean id="userInfoService" class="org.esupportail.sgc.services.userinfos.UserInfoService">
  <property name="caducIfEmpty" value="email"/>
</bean>
```

- dans applicationContext-crous.xml on doit renseigner un fichier xml ESIST - rappel : ce fichier sert à calculer les codes tarif et société crous en fonction de l'établissement de l'individu (supannEtablissement), du type d'individu (referenceStatut) de l'indice du personnel (indice)
 - à l'EMSE la mise en oeuvre de cette synchronisation étant destinée dans un premier temps aux étudiants, les codes société / tarif résultant de ce calcul doivent être de 10 / 1
 - aussi le fichier ESIST.xml peut se ramener à cela simplement :

File Edit Options Buffers Tools XML Text
Help

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<esist>
<dateG>2018010209494986</dateG>
<RneEmployeur>
<row>
<rne>0420094Z</rne>
<employeur>1</employeur>
</row>
</RneEmployeur>
<Statut>
<row>
<statut>etd</statut>
<libelle>etudiant</libelle>
</row>
</Statut>
<EmployeurStatutIndiceSocieteTarif>
<row>
<codeemployeur>1</codeemployeur>
<referencestatut>etd</referencestatut>
<indicemin>0</indicemin>
<indicemax>9999</indicemax>
<codesociete>10</codesociete>
<codetarif>1</codetarif>
</row>
</EmployeurStatutIndiceSocieteTarif>
</esist>
```

- Par rapport au cas d'usage très contraint / limité ici d'esup-sgc, notons
 - que seul esup-sgc est nécessaire
 - que applicationContext-services.xml peut être nettoyé, on supprime notamment
 - les userInfoService autres que shib, ldap et le 'spel' donné ci-avant
 - cardIdsService,
 - esupNfcTagService,
 - ldapValidateService et beans ldap associés,
 - ipService
 - on peut aussi supprimer entièrement les fichiers (ou plutôt les beans contenus dans ces fichiers pour éviter les conflits lors des màj par git) de
 - applicationContext-acces-control.xml
 - applicationContext-paybox.xml
 - dans applicationContext-security.xml seul le rôle ROLE_ADMIN est à renseigner
 - dans applicationContext-crous.xml on peut supprimer les beans associés à la carte étudiante européenne : europeanStudentCardService et escUIdFactoryService

Fonctionnement

Pour qu'un utilisateur ait son compte et sa carte synchronisés avec izly par esup-sgc, il faut ensuite :

- qu'il se connecte à esup-sgc et active le service
- ou que cette importation/activation soit réalisée via un appel à l'API REST :

```
curl -X POST 'https://esup-sgc.emse.fr/wsrest/api/externalCardEnable?eppn=toto@emse.fr&crous=true'
```

Une fois importée, on retrouve la 'carte' dans la vue manager d'esup-sgc

La synchronisation est ensuite faite régulièrement, cf [la configuration du fichier applicationTasksContext.xml](#)

Si le CSN change côté LDAP, une nouvelle carte est créée et activée en conséquence, l'ancienne étant alors désactivée.