Site de démonstration en ligne

En plus de la Machine Virtuelle, nous avons mis en place une instance d'ESUP-SGC (et applicatifs associés) de démonstration accessible à l'ensemble de la communauté de l'ESR au travers de la fédération d'identités ESR portée par RENATER.

Dans cette instance, tous les utilisateurs connectés ont un rôle d'utilisateur **et de gestionnaire (lien Vue Manager)** ; nous nous sommes gardés le rôle d'administration pour nous uniquement, ce rôle pouvant permettre, sur une mauvaise manipulation, de mettre hors d'usage le service assez rapidement 😉

Chacun peut donc demander des cartes, **mais aussi** les accepter, les imprimer, les encoder ...

Les clients esup-sgc / esup-nfc-tag sont proposés.

L'application Android est donc fonctionnelle, tout comme sa version Desktop en Java. De même l'application java d'encodage est disponilbe ainsi que sa version 'robot' via une zxp3.

Pour y accéder : https://esup-sgc-demo.univ-rouen.fr

La partie esup-nfc est également disponible (pour voir les tags en temps réel notamment) : https://esup-nfc-tag-demo.univ-rouen.fr

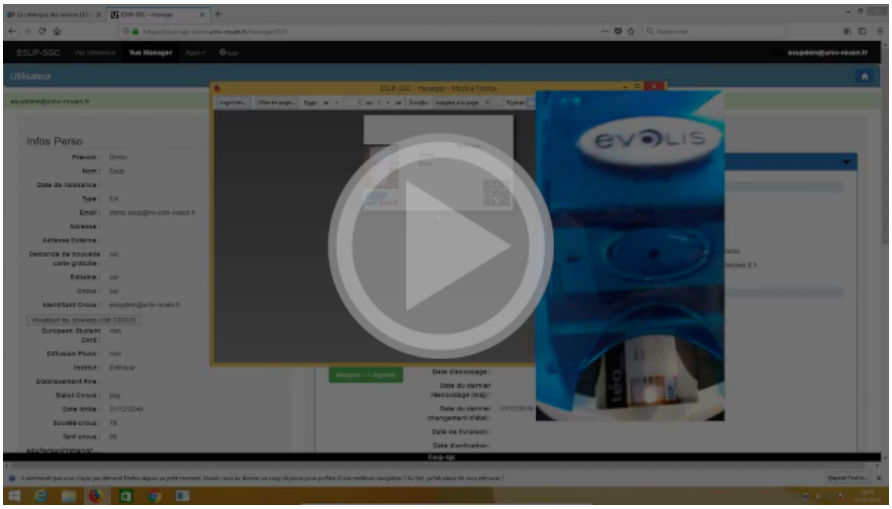
Les cartes et demandes de cartes sont purgées toutes les nuits.

- Vidéos de démonstration
- Usage
- Configurations
    - Configurations esup-sgc
    - Configurations esup-nfc-tag-server
- Bookworm et Jetty

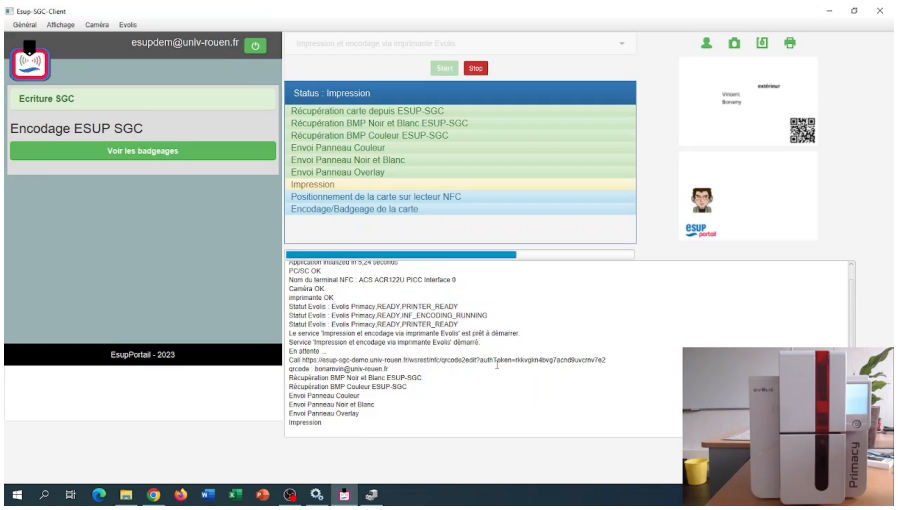
Vidéos de démonstration

Pour utiliser au mieux cette application de démonstration disponible en ligne, vous pouvez visualiser 3 vidéos de présentation :

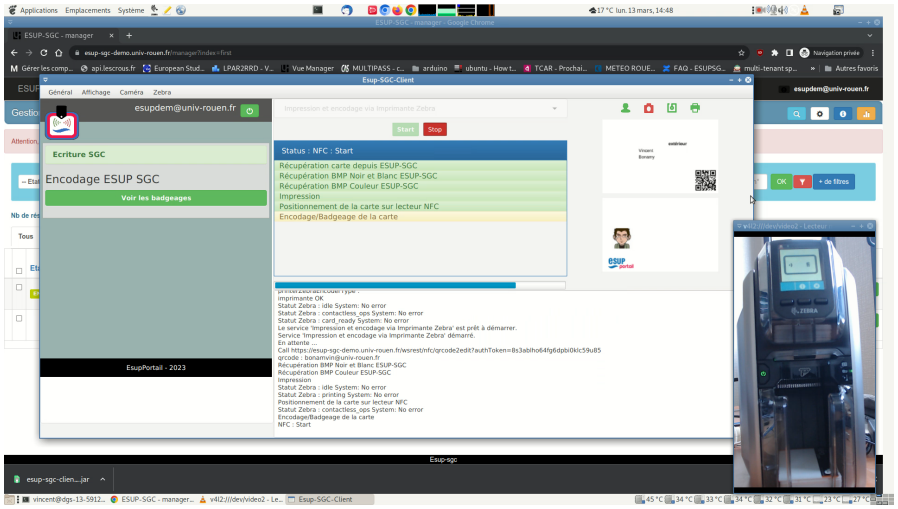
- La première vidéo présente l'édition en 2 temps :



- La deuxième vidéo présente l'édition en 1 temps avec Evolis Primacy :



- Et la troisième vidéo présente l'édition en 1 temps ave Zebra ZC300 :



Usage

Cette application de démonstration peut vous donner une idée de comment esup-sgc fonctionne. Elle peut vous permettre de tester un matériel d'encodage et d'impression par exemple sur des cartes Mifare Desfire (version 1 ou 2).

Configurations

Les configurations qui ont été faites pour monter ce site de démonstration ne présente que peu d'adhérence avec un Système d'Information.

- L'encodage de la carte consiste simplement ici à lier le CSN à la 'carte' de l'utilisateur dans esup-sgc.
- Les informations utilisateurs ne sont récupérées que par shibboleth, et on récupèr ainsi que très peu d'informations, les fiches sont donc peu renseignées.
- Il n'y a donc pas de synchronisation d'informations régulières depuis une source de données ldap ou sql
- Les cartes sont toutes éditables, tout le monde peut demander une carte gratuitement, les cartes ne deviennent jamais caduques (la date de fin étant à défaut à 2049).
- L'activation d'une carte n'engendre pas à une synchronisation/modification d'un ldap, un export dans un contrôle d'accès, la synchronisation dans le crous ou escr n'est pas non plus active.
- L'envoi de mails n'est pas actif
- La définition des groupes / rôles est très sommaire. Ne s'appuyant pas sur ldap, on utilise des 'règles spel' sur les champs utilisateurs.
- Etc.

Configurations esup-sgc

Par rapport aux configurations données par défaut, on a supprimé les userInfoServices d'exemple et les connexions au ldap.

Il a également fallu modifier la façon dont sont calculés les groupes et rôles pour faire sans ldap.

- Dans applicationContext-services.xml :

```
<bean id="groupService" class="org.esupportail.sgc.services.ldap.SpelGroupService">
  <property name="groups4eppnSpel">
    <map>
      <entry key="group_admin" value="#user.eppn==('bonamvin@univ-rouen.fr') or #user.eppn==('tranje1@univ-
rouen.fr') or #user.eppn==('lemaida3@univ-rouen.fr')"/>
      <entry key="group_manager" value="true"/>
      <entry key="group_livreur" value="true"/>
      <entry key="group_updater" value="true"/>
      <entry key="group_consult" value="true"/>
      <entry key="group_user" value="true"/>
    </map>
  </property>
</bean>
```

- Dans applicationContext-security.xml :

```
<util:map id="sgcMappingGroupesRoles">
  <beans:entry key="group_admin" value="ROLE_ADMIN" />
  <beans:entry key="group_manager" value="ROLE_SUPER_MANAGER" />
  <beans:entry key="group_livreur" value="ROLE_LIVREUR" />
  <beans:entry key="group_updater" value="ROLE_UPDATER" />
  <beans:entry key="group_consult" value="ROLE_CONSULT" />
  <beans:entry key="group_user" value="ROLE_USER" />
</util:map>
```

Configurations esup-nfc-tag-server

Même principe que pour esup-sgc.

- Dans applicationContext-security.xml :

```
<beans:bean id="groupService" class="org.esupportail.nfctag.security.SpelGroupService">
  <beans:property name="groups4eppnSpel">
    <beans:map>
      <beans:entry key="group_admin" value="#eppn=='bonamvin@univ-rouen.fr' or #eppn=='tranje1@univ-rouen.fr'
or #eppn=='lemaida3@univ-rouen.fr'"/>
      <beans:entry key="group_supervisor" value="true"/>
    </beans:map>
  </beans:property>
</beans:bean>

<util:map id="nfcMappingGroupesRoles">
  <beans:entry key="group_admin" value="ROLE_ADMIN" />
  <beans:entry key="group_supervisor" value="ROLE_SUPERVISOR" />
</util:map>
```

- Dans applicationContext-desfire.xml :

```
    <bean id="simpleTagEsupSgc" class="org.esupportail.nfctag.beans.DesfireTag">
    </bean>

    <bean id="desfireAuthConfigDaltonWriteEsupSgc" class="org.esupportail.nfctag.service.api.impl.
DesfireWriteConfig">
      <property name="desfireTag" ref="simpleTagEsupSgc" />
      <property name="description" value="Ecriture ESUP SGC"/>
    </bean>
```

Cette configuration d'encodage permet de ne rien effectuer sur la carte.

Ainsi la master-key n'est pas même nécessaire puisque aucune authentification ne sera effectuée.

La carte est donc laissée en l'état, seul le csn est lu et envoyé à esup-sgc pour enrollement.

Bookworm et Jetty

Ce site de démonstration est porté par une VM qui fonctionne depuis février 2024 sous debian bookworm.

On a choisi au passage ici de déployer esup-sgc et esup-nfc-tag non pas via un Tomcat mais via le Jetty installé via apt.

À toute fin utile, voici un extrait des fichiers de configurations nous permettant cette mise en place - le webapp d'esup-sgc (obtenu via mvn package) est ici déployé en tant que /opt/jetty-esup-sgc-demo/webapps/root

(pour esup-nfc-tag, les configurations jetty sont similaires)

- start.ini instance jetty - /opt/jetty-esup-sgc-demo/start.ini

```
--module=deploy,http,jsp,jstl,http-forwarded
jetty.http.port=8080
```

- systemd - /etc/systemd/system/jetty-esup-sgc-demo.service

```
[Unit]
Description=Jetty 9 Web Application Server
Documentation=https://www.eclipse.org/jetty/documentation/current/
After=network.target

[Service]

# Configuration
Environment="JETTY_HOME=/usr/share/jetty9/"
Environment="JETTY_STATE=/tmp/jetty-esup-sgc-demo.state"
Environment="JETTY_BASE=/opt/jetty-esup-sgc-demo"
Environment="JAVA_OPTS=-Djava.awt.headless=true"
EnvironmentFile=-/etc/default/jetty9

# Lifecycle
Type=simple
ExecStart=/usr/share/jetty9/bin/jetty.sh run
SuccessExitStatus=143
Restart=on-abort

# Logging (usage de logrotate pour la rotation)
StandardOutput=file:/var/log/jetty-esup-sgc-demo/esup-sgc.log
StandardError=file:/var/log/jetty-esup-sgc-demo/esup-sgc.log

# Security
User=jetty
Group=jetty
PrivateTmp=yes
AmbientCapabilities=CAP_NET_BIND_SERVICE
NoNewPrivileges=true
WorkingDirectory=/usr/share/jetty9/
ProtectSystem=strict
ReadWritePaths=/var/lib/jetty9/

[Install]
WantedBy=multi-user.target
```

- configuration apache proxypass - /etc/apache2/sites-enabled/esup-sgc-demo.univ-rouen.fr.conf

```
...
   <Location />
     AuthType shibboleth
     ShibRequestSetting requireSession 1
     require shib-session
     ShibUseHeaders On
   </Location>

    ProxyPreserveHost On
    RequestHeader set X-Forwarded-Proto "https"
    RequestHeader set X-Forwarded-Port 443
    ProxyPass / http://localhost:8080/ ttl=10 timeout=3600 loadfactor=100 retry=1
    ProxyPassReverse / http://localhost:8080
...
```

- configuration shibboleth (avec sp 3) - /etc/shibboleth/shibboleth2.xml

```
<SPConfig xmlns="urn:mace:shibboleth:3.0:native:sp:config"
    xmlns:conf="urn:mace:shibboleth:3.0:native:sp:config"
    xmlns:md="urn:oasis:names:tc:SAML:2.0:metadata"
    clockSkew="180">

    <RequestMapper type="Native">
        <RequestMap applicationId="default">
            <Host name="esup-sgc-demo.univ-rouen.fr" applicationId="default" authType="shibboleth"
requireSession="false"/>
            <Host name="esup-nfc-tag-demo.univ-rouen.fr" applicationId="esup-nfc-tag-demo" authType="
shibboleth" requireSession="false"/>
        </RequestMap>
    </RequestMapper>

    <ApplicationDefaults entityID="https://esup-sgc-demo.univ-rouen.fr"
        REMOTE_USER="eppn subject-id pairwise-id persistent-id"
        cipherSuites="DEFAULT:!EXP:!LOW:!aNULL:!eNULL:!DES:!IDEA:!SEED:!RC4:!3DES:!kRSA:!SSLv2:!SSLv3:!
TLSv1:!TLSv1.1">

        <Sessions lifetime="28800" timeout="3600" checkAddress="false"
            handlerURL="/Shibboleth.sso" handlerSSL="true" cookieProps="https" relayState="ss:mem"
            redirectLimit="exact"
            idpHistory="false" idpHistoryDays="7">

          <SessionInitiator type="Chaining" Location="/Login" isDefault="true" id="Login"
                            relayState="cookie">
            <SessionInitiator type="SAML2" acsIndex="1" acsByIndex="false" template="bindingTemplate.
html"/>
        <SessionInitiator type="Shib1"/>
        <SessionInitiator type="SAMLDS" URL="https://discovery.renater.fr/edugain/WAYF"/>
          </SessionInitiator>

            <md:AssertionConsumerService Location="/SAML2/POST" index="1"
                     Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST"/>

            <!-- Status reporting service. -->
            <Handler type="Status" Location="/Status"/>

            <!-- Session diagnostic service. -->
            <Handler type="Session" Location="/Session" showAttributeValues="true"/>

        </Sessions>

        <MetadataProvider type="XML" validate="false"
                url="https://metadata.federation.renater.fr/renater/main/main-idps-renater-metadata.xml"
              backingFilePath="/etc/shibboleth/metadatas/main-idps-renater-metadata.xml"
maxRefreshDelay="7200">
        </MetadataProvider>

        <AttributeExtractor type="XML" validate="true" reloadChanges="false" path="attribute-map.xml"/>

        <AttributeExtractor type="Metadata" errorURL="errorURL" DisplayName="displayName"/>

        <AttributeFilter type="XML" validate="true" path="attribute-policy.xml"/>

        <CredentialResolver type="File"
            key="esup-sgc-demo.univ-rouen.fr.key" certificate="esup-sgc-demo.univ-rouen.fr.crt"/>

    <ApplicationOverride id="esup-nfc-tag-demo" entityID="https://esup-nfc-tag-demo.univ-rouen.fr"
homeURL="https://esup-nfc-tag-demo.univ-rouen.fr"/>

    </ApplicationDefaults>

    <SecurityPolicyProvider type="XML" validate="true" path="security-policy.xml"/>

    <ProtocolProvider type="XML" validate="true" reloadChanges="false" path="protocols.xml"/>

</SPConfig>
```