Configuration de la sécurité

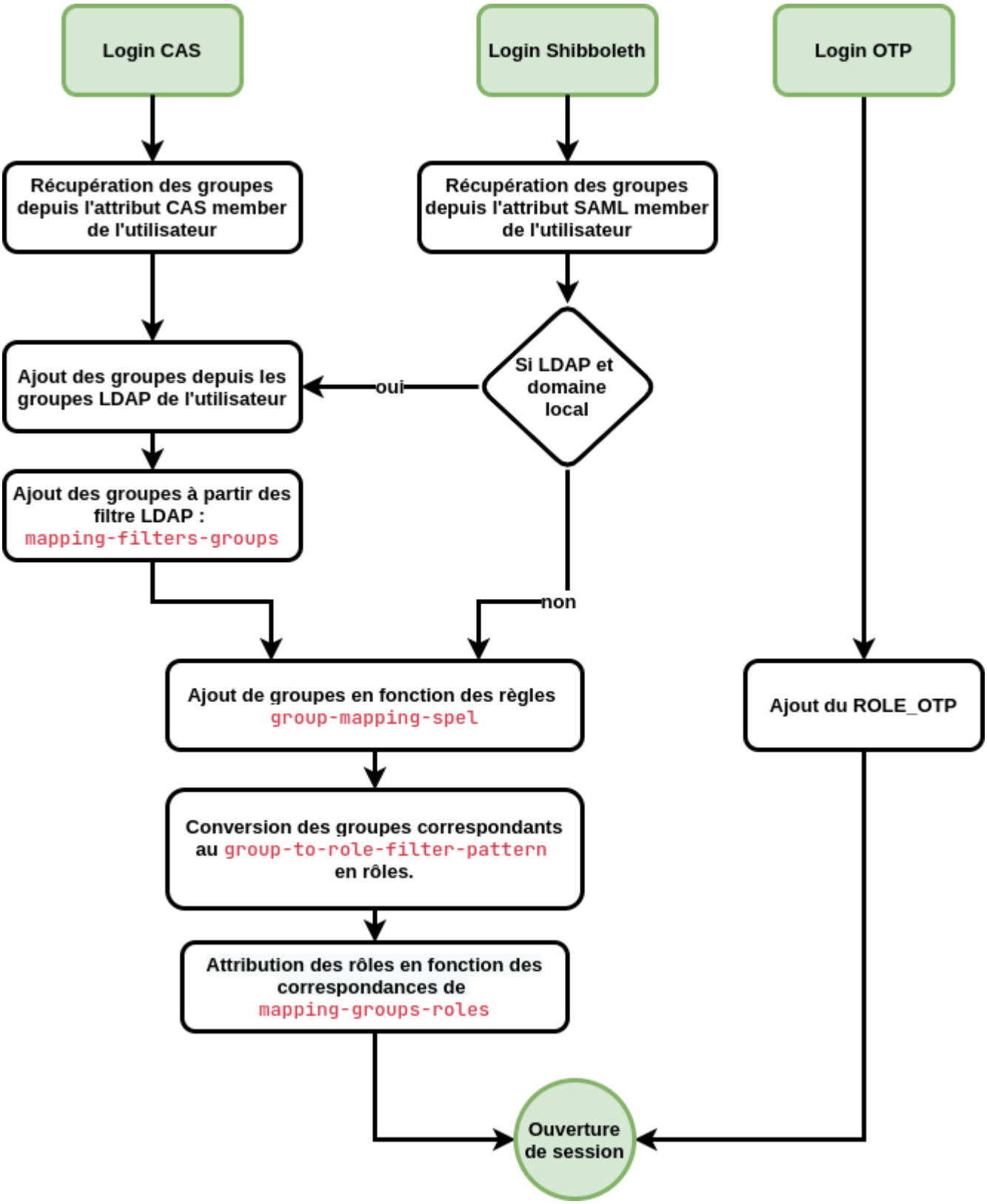
esup-signature propose plusieurs modes d'authentification qui peuvent être utilisés simultanément :

- Authentification CAS
- Authentification Shibboleth
- Authentification OTP (one time password)
- *Authentification OAuth (à l'étude)*

L'authentification d'un utilisateur donne lieu à l'attribution de rôles via un mécanisme de correspondance groupes rôles que l'on va détailler ici.

La configuration se fait dans le fichier de configuration src/main/resources/application.yml voir : Configuration#src/main/resources/application.yml

Mécanisme d'attribution des rôles



Voici le détail des étapes :

- Lorsqu'un utilisateur se connecte on récupère la liste des groupes dont il fait parti
- Si une connexion LDAP est configurée il est possible d'attribuer des groupes à l'utilisateur à l'aide de filtres ldap grâce à la propriété **mapping-filters-groups** : Configuration#ldap.1

- Dans tous les cas il est aussi possible d'attribuer des groupes via des règles **group-mapping-spel** utilisant la syntaxe SePL de Spring. Cela est limité à l'utilisation de l'attribut **#eppn** pour permettre d'attribuer un groupe à une personne en particulier. Il est possible aussi d'attribuer un groupe à tous les utilisateurs en utilisant "true"
- Ensuite vient l'attribution des rôles :
    - Tout d'abord à l'aide du **group-to-role-filter-pattern**. Il va détecter les groupes de l'utilisateur qui correspondent au pattern défini et attribuer le rôle correspondant. Ex : le groupe toto.tata pour le pattern toto.(\w*) donnera le rôle ROLE_TATA
    - Enfin l'attribution des autres rôles se fait à l'aide la liste de correspondance spécifiée dans **mapping-groups-roles** où l'on déclare nom_du_groupe = ROLE_NOM_DU_ROLE.

esup-signature possède quatre rôles particuliers :

- ROLE_USER : indispensable pour accéder à l'application
- ROLE_ADMIN : qui ouvre les droits à la partie administration
- ROLE_OTP : obtenu lorsqu'un utilisateur externe se connecte via OTP. Il obtient des droits pour effectuer des signatures
- ROLE_SEAL : permet de signer avec le certificat cachet d'etablissement lorsqu'il est disponible

Cas d'usages

Voici quelques exemples de configuration à mettre en place dans src/main/ressource/application.yml pour illustrer différents cas de figure.

J'ai un annuaire ldap et je souhaite attribuer le rôle ROLE_USER en fonction d'un filtre ldap

Dans la partie **ldap**, un exemple de filtre pour sélectionner les personnels **staff**. Toutes les personnes correspondant au filtre seront dans le groupe "mes-utilisateurs"

```
ldap:
    search-base: ou=people
        group-search-base: ou=groups
    user-id-search-filter: (uid={0})
    group-search-filter: member={0}
    member-search-filter: (&(uid={0})({1}))
    mapping-filters-groups:
        mes-utilisateurs : "(eduPersonAffiliation:=staff)"                        # ici le filtre ldap va
remplir un groupe virtuel propre à esup-signature 'mes-utilisateurs'
```

Attention vos requetes LDAP doivent impérativement être mises entre **parentheses**.

Pour affecter un rôle à ce groupe il suffit d'ajouter l'affectation dans la partie **security.web**

```
security:
        ...
        ...
        web:
                ...
            mapping-groups-
roles:
                mes-utilisateurs: ROLE_USER                                                    # on affecte
le groupe 'mes-utilisateurs' au role ROLE_USER
            ws-access-authorize-ips: 127.0.0.1
```

J'ai déjà des groupes dans mon annuaire je souhaite les utiliser pour attribuer des rôles spécifiques aux circuits et/ou pour donner le ROLE_USER

Prenons un cas ou l'utilisateur arrive dans esup-signature avec dans **member** les groupes ldap suivants:

- esup-signature.mes-utilisateurs (groupe auquel on souhaite donner le ROLE_USER)
- esup-signature.mesroles.circuit_toto (groupe que l'on souhaite utiliser pour donner des droits sur le circuit "toto")

Cela se configure dans la partie security.web :

```
security:
        ...
        ...
        web:
        group-to-role-filter-pattern: esup-signature.mesroles.(\w*)                          # on configure le
pattern permettant de retrouver automatiquement les groupes amenés à devenir

des rôles applicatifs
            mapping-groups-
roles:
                esup-signature.mes-utilisateurs: ROLE_USER                                          # on attribut
directement le ROLE_USER aux personnes du groupe 'esup-signature.mes-utilisateurs'
            ws-access-authorize-ips: 127.0.0.1
```

Je ne récupère pas de groupe dans l'attribut member suite à la connexion, je souhaite donc utiliser une requête ldap

Il est possible de retrouver les groupes d'un utilisateur via une requête ldap. Par exemple, si on veux affecter les membres d'un groupe présent dans ldap à un groupe "virtuel" d'esup-signature on peut faire comme suit :

```
ldap:
    search-base: ou=people
        group-search-base: ou=groups
    user-id-search-filter: (uid={0})
    group-search-filter: member={0}
    member-search-filter: (&(uid={0})({1}))
    mapping-filters-groups:
        mes-admins : "(memberOf:=cn=esup-signature_admin,ou=groups,dc=univ-ville,dc=fr)"
# ici le filtre ldap va remplir un groupe virtuel propre à esup-signature 'mes-admins'
```

Attention vos requetes LDAP doivent impérativement être mises entre **parentheses**.

On pourra utiliser "admin" par la suite dans **mapping-groups-roles**

J'ai configuré l'authentification Shibboleth et je ne suis pas connecté à un annuaire

Le cas général serait de donner le ROLE_USER à tout le monde est de donner le ROLE_ADMIN à certains. Dans ce cas il faut utiliser group-mapping-spel pour attribuer des rôles directement avec la syntaxe SePL.

```
security:
        ...
        ...
        web:
                ...
            mapping-groups-
roles:
                mes-utilisateurs:
ROLE_USER                                                                                   # on
attribut directement le ROLE_USER aux personnes du groupe 'mes-utilsateurs'
                mes-admins:
ROLE_ADMIN
# on attribut directement le ROLE_ADMIN aux personnes du groupe 'mes-admins'
            ws-access-authorize-ips: 127.0.0.1
                group-mapping-spel:
                        mes-utilisateurs:
"true"                                                                                          #
met tout le monde dans le groupe 'mes-utilisateurs"
                        mes-admins: "#eppn == 'user1@univ-ville.fr' or #eppn == 'user2@univ-ville.fr"     # met
user1 et user2 dans le groupe 'mes-admins'
```

À l'usage, cette configuration est assez contraignante car les utilisateurs qui ne se sont jamais connectés ne peuvent pas être retrouvés dans les recherches des destinataires des documents.

Pour envoyer une demande à une nouvelle personne, il faudra inscrire son adresse email complète suite à quoi un profil temporaire (de type shibboleth) sera créé. Lors de la connexion du destinataire pour signer le document, le profil sera complété des noms, prénoms est identifiants.

Configuration spécifique pour l'authentification avec Shibboleth

Installation du mod_sib voir : https://services.renater.fr/federation/documentation/guides-installation/sp3/chap04

Voici la configuration à ajouter coté Apache :

```
<Location />
  ShibUseHeaders Off
  ShibRequireSession Off
</Location>

<Location /user/nexu-sign>
  ShibUseHeaders Off
  ShibRequireSession Off
</Location>

<Location /Shibboleth.sso>
  SetHandler shib
</Location>

<Location /login/shibentry>
  AuthType shibboleth
  ShibRequireSession On
  Require shibboleth
  ShibUseHeaders On
</Location>
```