

Scribe - Evolution #12521

MAJ automatique du mot de passe ldap local

08/05/2015 06:33 AM - Pascal MIETLICKI

Status:	Nouveau	Start date:	08/05/2015
Priority:	Normal	Due date:	
Assigned To:		% Done:	0%
Category:		Estimated time:	0.00 hour
Target version:		Spent time:	0.00 hour
Distribution:	EOLE 2.4		

Description

Bonjour,

Nous avons besoin d'un ENT académique accessible à l'ensemble du personnel académique (professeur et administratif). Le fonctionnement via l'ajout du ldap principal académique est correct. Mais l'on a souhaité rajouter les personnes dans des groupes locaux ou faire évoluer leurs fiches dans le ldap eole local qui est plus facilement modifiable et moins "rigide" que celui académique car l'on peut difficilement le faire évoluer.

J'ai donc modifié dataproxy.py (dans /usr/share/pyshared/eoless/) pour effectuer une synchronisation des mots de passe du ldap académique vers le local.

La restriction principale est qu'il faut avoir créé le compte dans le ldap local via l'EAD mais on pourrait aussi effectuer une création automatique si le compte n'existe pas dans le local.

Nous avons fait ce choix pour éviter de polluer, inutilement, le ldap local sachant que nous effectuons une synchronisation du ldap local via AAF.

Sous EOLE 2.4, le fichier dataproxy.py se trouve ailleurs (dans les librairies python).

Les modifications :

39a40,41

```
from scribe.eoleldap import Ldap
from scribe.linker import _user_factory
```

119a122

```
#log.msg("Eole proxy %s" % eole_proxy)
```

224,232c227,248

```
< # vérification du résultat de l'authentification sur le serveur précédent
< if success False:
< # echec de l'authentification sur ce serveur
< if len(servers) > 0:
< # test sur le serveur suivant
< return self.authenticate(user_id, passwd, search_branch, servers)
< else:
< # on a essayé sur tous les serveurs
< return False, {}
---
```

```
if success True and "localhost" not in servers:
#log.msg("MAJ pwd vers localhost")
#recupere de ead2 backend/actions/scribe/userpwd.py
try:
conn = Ldap()
```

```
conn.connect()
ldapuser = _user_factory(user_id, conn.connexion)
ldapuser.c_mod_password(user_id, passwd)
conn.close()
except:
pass

    1. echec de l'authentification sur ce serveur
    if len(servers) > 0:
    2. test sur le serveur suivant
    return self.authenticate(user_id, passwd, search_branch, servers)

if success == False :

    1. on a essayé sur tous les serveurs
    return False, {}
```

Je partage si ça peut servir à d'autres.

Merci pour cet excellent projet qui gagnerait à être connu.

Bonne journée et à bientôt,
Pascal

Files

dataproxy.py	13 KB	08/05/2015	Pascal MIETLICKI
dataproxy_patch.txt	2.01 KB	08/05/2015	Pascal MIETLICKI