Passerelle - Development #66533

créer un « connecteur LDAP »

23 juin 2022 11:21 - Thomas Noël

Statut:	Fermé	**Début:**	23 juin 2022
Priorité:	Normal	**Echéance:**	
Assigné à:	Benjamin Dauvergne	**% réalisé:**	0%
Catégorie:		**Temps estimé:**	0:00 heure
Version cible:			
Patch proposed:	Oui	**Planning:**	Non

Description

Un connecteur permettant de faire des recherches et requêtes dans un annuaire LDAP.

Objectif :

- être utilisable comme source de donnée (avec q et id pour l'autocomplétion)
- proposer une API compatible avec la recherche Combo (cf SearchAPI)

Configuration : URL, binddn, passdn, basedn, certificat client

Système de requête : une requête =

- un filtre avec un %s ou des {{ request.GET.q }} (à voir en fonction des possibilités anti-injection)
- un template pour le "text" de la réponse

Révisions associées

Révision e905cdb5 - 13 septembre 2022 17:34 - Benjamin Dauvergne

add ldap connector (#66533)

Révision b32c2f58 - 13 septembre 2022 17:34 - Benjamin Dauvergne

base: prevent leak of opened fieldfile in export_json (#66533)

Historique

#2 - 23 juin 2022 11:25 - Thomas Noël

- Description mis à jour

#3 - 23 juin 2022 11:30 - Thomas Noël

L'occasion d'utiliser plutôt https://github.com/cannatag/ldap3 (doc : https://ldap3.readthedocs.io/, paquet : https://packages.debian.org/bullseye/python3-ldap3 = version 2.8.1)

#4 - 27 juin 2022 10:57 - Benjamin Dauvergne

- Assigné à mis à Benjamin Dauvergne

#7 - 03 août 2022 01:05 - Benjamin Dauvergne

- Fichier 0002-add-ldap-connector-66533.patch ajouté

- Fichier 0001-add-a-binary-file-field-66533.patch ajouté

- Patch proposed changé de Non à Oui

#8 - 03 août 2022 13:23 - Benjamin Dauvergne

- Fichier 0002-add-ldap-connector-66533.patch ajouté

- Fichier 0001-add-a-binary-file-field-66533.patch ajouté

- Statut changé de Nouveau à Solution proposée

Pas de système de requête pour l'instant, un seul endpoint "search" qui fait tout, l'authentification TLS est implémentée, je me suis dit qu'on s'en passerait au début puis j'ai vu que pour la deuxième fois on s'en sert, après le mincult, à Nantes :/

Les paramètres les plus importants pour l'appel search sont :

- ldap_base_dn , pas d'explication
- search_attribute, l'attribut dans lequel sera cherché la chaîne "q" via le filtre (search_attribute=*<q>*), j'ai hésité d'utiliser l'opérateur de recherche alternative (@search_attribute~=<q>) le comportement n'en est pas standardisé, sur OpenLDAP ça fait des histoires avec l'algo soundex et ça dépend beaucoup de la présence d'un index sur l'attribut visé et sur AD je ne sais pas ce que ça fait (d'un rapide test en local ça fait une recherche approximative pas trop mal, je matche "Benjamin Dauvergne" avec "bnjm dvrg" mais pas avec "dauvergne", une option future peut-être).
- id_attribute, l'attribut qui sert à générer l'id, uid par exemple ou sAMAccountName ou entryuuid, un truc idéalement stable, unique et intelligible,
- text_template est géré mais il faut penser à mettre les attributs utilisés dans "ldap_attributes" (c'est case insensitive dans le template et la liste, donc pas besoin de se souvenir de la casse de l'attribut sur l'annuaire visé),
- q/id mais ça c'est classique pour une source de donnée,
- ldap_filter, pour ajouter d'autres filtres, comme (objectClass=inetorgperson),
- scope, par défaut c'est subtree mais on peut mettre "onelevel" pour n'avoir que les fils directs du base_dn.

Ce ne serait pas difficile d'ajouter un système de requête par dessus, mais ça ne me parait pas utile à ce stade.

J'ai finalement retiré le ldap_base_dn au niveau du modèle du connecteur, ça ne servait à rien.

#9 - 03 août 2022 13:32 - Frédéric Péters

Un mot sur l'introduction de BinaryFileField, quel soucis avec le champ fichier existant ?

#10 - 03 août 2022 13:45 - Benjamin Dauvergne

Frédéric Péters a écrit :

> Un mot sur l'introduction de BinaryFileField, quel soucis avec le champ fichier existant ?

C'est fatiguant ces fichiers, ça traîne.

#11 - 03 août 2022 13:45 - Benjamin Dauvergne

- Assigné à changé de Benjamin Dauvergne à Thomas Noël

#12 - 03 août 2022 15:01 - Benjamin Dauvergne

- Fichier 0002-add-ldap-connector-66533.patch ajouté

- Fichier 0001-add-a-binary-file-field-66533.patch ajouté

Des tests sur l'import/export.

#13 - 03 août 2022 16:39 - Frédéric Péters

> C'est fatiguant ces fichiers, ça traîne.

Je n'ai pas compris.

#14 - 04 août 2022 12:15 - Thomas Noël

Frédéric Péters a écrit :

> > C'est fatiguant ces fichiers, ça traîne.
>
> Je n'ai pas compris.

Non plus...

#15 - 04 août 2022 14:37 - Benjamin Dauvergne

Je n'aime pas les FileField pour des chose qui ne méritent pas un fichier sur le disque mais je veux bien utiliser un TextField si vous voulez.

#16 - 05 août 2022 10:02 - Benjamin Dauvergne

- Fichier 0001-add-ldap-connector-66533.patch ajouté

#17 - 05 août 2022 14:10 - Benjamin Dauvergne

Le seul truc moche à mon goût c'est de devoir rematérialiser les certificats parce qu'en 2022 aucun truc qui fait du SSL en python (libldap ou le module ssl de python utilisé par python-ldap3) ne gère un certificat et une clé posés en mémoire (enfin si avec paramiko ça marche); c'est un peu triste.

#18 - 05 août 2022 14:29 - Thomas Noël

Je rate sans doute un truc mais je ne comprends pas bien pourquoi tu t'embêtes avec tout ça, tu peux pas juste poser un fichier client_cert bête-et-méchant, comme le "keystore" de HTTPResource ? Et utiliser « conn.set_option(ldap.OPT_X_TLS_CERTFILE, self.client_cert.path) » et « conn.set_option(ldap.OPT_X_TLS_KEYFILE, self.client_cert.path) ». Et pour la validation, on s'en fiche : à celui qui configure le connecteur de ne pas mettre n'importe quoi...

Et au passage, pour prévenir de l'enfer, sur la même idée de présence d'un self.cacert, ajouter un OPT_X_TLS_CACERTFILE ? (quoique bon, ça, on peut s'en charger en l'ajoutant globalement sur la machine).

#19 - 05 août 2022 14:40 - Benjamin Dauvergne

Thomas Noël a écrit :

> Et au passage, pour prévenir de l'enfer, sur la même idée de présence d'un self.cacert, ajouter un OPT_X_TLS_CACERTFILE ? (quoique bon, ça, on peut s'en charger en l'ajoutant globalement sur la machine).

La sécurité de taper sur le bon serveur ne me paraissait pas super important à ce stade pour l'instant je désactive juste la validation des certificats.

#20 - 06 septembre 2022 15:12 - Thomas Noël

- Assigné à changé de Thomas Noël à Benjamin Dauvergne

Mini bogue :

- err_clss → err_class dans la fonction search quand le serveur est pas là

Mini détails :

- ajouter l'entête AGPL en haut de passerelle/apps/ldap/forms.py

Pour le reste ça me semble ok.

Juste pour être sûr : au niveau de la recherche c'est sûr qu'on ne va pas trouver "Benjamin Dauvergne" quand on va taper "Dauvergne Benjamin", mais c'est ainsi que LDAP fonctionne. N'est-ce pas ?

#21 - 06 septembre 2022 15:32 - Benjamin Dauvergne

Thomas Noël a écrit :

> Juste pour être sûr : au niveau de la recherche c'est sûr qu'on ne va pas trouver "Benjamin Dauvergne" quand on va taper "Dauvergne Benjamin", mais c'est ainsi que LDAP fonctionne. N'est-ce pas ?

On a juste de la recherche sur sous-chaîne pure, donc on ne va même pas trouver "Benjamin Dauvergne" si on tape "Benjamin Dauvergne" (avec deux espaces), par contre ça devra être insensible à la casse si on tape sur CN, CN étant généralement déclaré comme insensible à la casse pour la recherche. Comme dit plus haut je n'ai pas tenté de proposer l'opérateur recherche approximative de LDAP.

#22 - 07 septembre 2022 09:30 - Benjamin Dauvergne

- Fichier 0001-add-ldap-connector-66533.patch ajouté

- Fichier 0002-base-prevent-leak-of-opened-fieldfile-in-export_json.patch ajouté

Voilà, quelques changements :

- typo signalée corrigée
- retrait des BinaryField pour des FileField
- support de TLS_CACERTFILE
- ajout d'un paramètre optionnel search_op permettant d'utiliesr au choix :
  - substring : mode par défaut "cn=*truc muche*"
  - prefix : si on sait que CN contient d'abord le nom "cn=truc*"
  - approx : si on sait que l'opérateur de recherche approchée fonctionne bien "cn~=truc muche"

On pourra éventuellement ajouter aussi la gestion de la pagination ou du trie coté LDAP ou coté passerelle plus tard/si nécessaire. Actuellement rien n'est trié.

PS: aussi factorisé une méthode upload_to générique utilisée plusieurs fois et entête AGPL ajouté à forms.py dans la branche.

#23 - 09 septembre 2022 17:56 - Thomas Noël

Perso, j'aurais préféré des cases à cocher dans la config pour ces deux choix :

```
+        conn.set_option(ldap.OPT_X_TLS_REQUIRE_SAN, ldap.OPT_X_TLS_NEVER)
+        conn.set_option(ldap.OPT_X_TLS_REQUIRE_CERT, ldap.OPT_X_TLS_NEVER)
```

mais je peux (vraiment) apprendre à vivre sans ! :)

Sur le search_op tu fais :

```
+        if search_op not in SEARCH_OPS:
+            search_op = SEARCH_OP_SUBSTRING
```

je pense que tu peux le retirer car juste en dessous tu fais un « raise APIError('unknown search_op %r' % search_op) » quand c'est hors clou. Tu pourras alors supprimer SEARCH_OPS qui n'aura plus d'usage.

Et sinon, mini typo :

```
+                'description': _('Identifier for exacte retrieval, using the id_attribute'),
                                                  ^
                                                  |__ exact
```

Et c'est vraiment tout.

#24 - 13 septembre 2022 17:06 - Benjamin Dauvergne

Branche à jour avec les modifications demandées (je rebaserai après validation).

#25 - 13 septembre 2022 17:34 - Thomas Noël

- Statut changé de Solution proposée à Solution validée

#26 - 13 septembre 2022 17:35 - Benjamin Dauvergne

- Statut changé de Solution validée à Résolu (à déployer)

```
commit b32c2f58dc7226a87310415fa62f05bb14b72da7
Author: Benjamin Dauvergne <bdauvergne@entrouvert.com>
Date:   Wed Sep 7 08:10:29 2022 +0200

    base: prevent leak of opened fieldfile in export_json (#66533)

commit e905cdb51622522c5c3bfcff2635ea79e9562d70
Author: Benjamin Dauvergne <bdauvergne@entrouvert.com>
Date:   Mon Aug 1 19:15:30 2022 +0200

    add ldap connector (#66533)
```

#27 - 16 septembre 2022 15:14 - Transition automatique

- Statut changé de Résolu (à déployer) à Solution déployée

#28 - 20 novembre 2022 04:42 - Transition automatique

Automatic expiration

Fichiers

0002-add-ldap-connector-66533.patch	31,1 ko	02 août 2022	Benjamin Dauvergne
0001-add-a-binary-file-field-66533.patch	4,76 ko	02 août 2022	Benjamin Dauvergne
0002-add-ldap-connector-66533.patch	38,4 ko	03 août 2022	Benjamin Dauvergne
0001-add-a-binary-file-field-66533.patch	4,76 ko	03 août 2022	Benjamin Dauvergne
0002-add-ldap-connector-66533.patch	43,9 ko	03 août 2022	Benjamin Dauvergne
0001-add-a-binary-file-field-66533.patch	5,96 ko	03 août 2022	Benjamin Dauvergne
0001-add-ldap-connector-66533.patch	50,6 ko	05 août 2022	Benjamin Dauvergne
0001-add-ldap-connector-66533.patch	54,9 ko	07 septembre 2022	Benjamin Dauvergne
0002-base-prevent-leak-of-opened-fieldfile-in-export_json.patch	1,33 ko	07 septembre 2022	Benjamin Dauvergne