Authentic 2 - Bug #67600

lookup_by_email dans LDAPBackend est sensible à la casse du mail, ce qui provoque des doublons

21 juillet 2022 12:08 - Thomas Noël

Statut:	Solution déployée	**Début:**	21 juillet 2022
Priorité:	Normal	**Echéance:**	
Assigné à:	Benjamin Dauvergne	**% réalisé:**	0%
Catégorie:		**Temps estimé:**	0:00 heure
Version cible:			
Patch proposed:	Oui	**Planning:**	Non

Description

Si un compte toto@example.net existe déjà alors que l'annuaire propose un TOTO@example.net.

Il faut utiliser le "get_by_email" ajouté lors de #64626 (qui va de plus gérer les collisions unicode).

Demandes liées:

Lié à Authentic 2 - Bug #67590: User.MultipleObjectsReturned	**Fermé**	**21 juillet 2022**

Révisions associées

Révision 39ee9d89 - 22 février 2024 12:57 - Benjamin Dauvergne

ldap: use get_by_email for ldap email lookup (#67600)

Révision 5726eadf - 22 février 2024 12:57 - Benjamin Dauvergne

tests: use a deterministic order on users (#67600)

Historique

#2 - 21 juillet 2022 12:18 - Frédéric Péters

- Lié à Bug #67590: User.MultipleObjectsReturned ajouté

#3 - 21 juillet 2022 15:00 - Thomas Noël

- Fichier 0001-use-get_by_email-for-ldap-email-lookup-67600.patch ajouté

- Patch proposed changé de Non à Oui

Ci-joint une tentative qui ne fonctionne pas, le tox :

```
$ tox -e py3-bullseye -- -vv tests/test_ldap.py -k test_get_users
...
    def test_get_users(slapd, settings, db, monkeypatch, caplog):
        ...
        slapd.add_ldif(ldif)
        conn = slapd.get_connection_admin()
        save.reset_mock()
        bulk_create.reset_mock()
        users = list(ldap_backend.LDAPBackend.get_users())
>       assert User.objects.count() == 6
E       assert 7 == 6
```

Donc : je n'arrive pas à faire ou correctif, ou le test, ou les deux...

#4 - 21 juillet 2022 15:20 - Paul Marillonnet

Je vois, dans le backend LDAP, la méthode _lookup_user_queryset :

```
return LDAPUser.objects.prefetch_related('groups').exclude(userexternalid__source=block['realm'])
```

i.e. on cherche des doublons sur des usagers qui ne proviennent pas du LDAP.

Je pense que l'esprit du code de ce backend est que si des doublons existent dans l'annuaire, c'est de sa faute et qu'il ne faut pas chercher à corriger/réconcilier ça dans authentic. Je pense que le code est bon, mais que le test devrait créer un usager lambda, hors LDAP avec un mail bidon,

et par ailleurs dans le LDAP un usager avec ce même mail BiDoN, et constater qu'aucun doublon n'est créé à l'import.

#5 - 21 juillet 2022 16:13 - Paul Marillonnet

- Fichier 0002-use-get_by_email-for-ldap-email-lookup-67600.patch ajouté

- Fichier 0001-ldap-perform-extensive-user-lookup-regardless-of-the.patch ajouté

- Statut changé de Nouveau à Solution proposée

Et donc, pour poursuivre cette idée, on pourrait imaginer que le lookup du backend LDAP va chercher parmi tous les usagers pas seulement les LDAPUser. Il faut juste un patch (0001) avant le tien (0002, légèrement modifié tel que dans ça précédente note — "test devrait créer un usager lambda, hors LDAP avec un mail bidon, et par ailleurs dans le LDAP un usager avec ce même mail BiDoN, et constater qu'aucun doublon n'est créé à l'import.").

#6 - 22 juillet 2022 11:04 - Thomas Noël

Dans 0002 le assert User.objects.count() == 7 : c'est 6 qu'on veut (pas de nouveau compte, on doit rester à 6).

Mais je suis super ultra frileux sur la modif LDAPUser/User, ça me semble très impactant.

Au final j'ai l'impression que les lookups sont bien prévus pour ne travailler que sur les comptes LDAP, que c'est fait à dessein. Donc s'il y a des doublons parce que d'autres comptes non-LDAP existaient, c'est "normal", ils n'auraient juste pas dû exister. Que la source soit une mauvaise manip ou défaut de configuration, ça doit être réglé autrement. Auquel cas je me serais trompé sur mon analyse et je rejetterai ce ticket.

Bref : je préfère que Benjamin vienne dire.

#7 - 22 juillet 2022 11:17 - Paul Marillonnet

Thomas Noël a écrit :

> Dans 0002 le assert User.objects.count() == 7 : c'est 6 qu'on veut (pas de nouveau compte, on doit rester à 6).

Cette fois-ci on crée bien un utilisateur hors LDAP dans le test (avec une adresse qui n'a rien à voir avec les "etienne.michu@…" créés jusque là), ça en fait un en plus de ceux de la synchro. On ajoute ce même usager avec une casse différente dans le ldif et on vérifie qu'on est bien à 7 (et non pas à 8 si doublon il y avait eu).

> Mais je suis super ultra frileux sur la modif LDAPUser/User, ça me semble très impactant.

Oui, changement de paradigme, on cherche à réconcilier les usagers issus de l'annuaire sur une base de comptes déjà existante dans Publik.

> Au final j'ai l'impression que les lookups sont bien prévus pour ne travailler que sur les comptes LDAP, que c'est fait à dessein. Donc s'il y a des doublons parce que d'autres comptes non-LDAP existaient, c'est "normal", ils n'auraient juste pas dû exister. Que la source soit une mauvaise manip ou défaut de configuration, ça doit être réglé autrement. Auquel cas je me serais trompé sur mon analyse et je rejetterai ce ticket.

Oui c'était l'esprit du code jusque là : un compartimentation des comptes LDAP sans chercher à résoudre les doublons avec des comptes hors-LDAP qui pré-existeraient à la synchro.

Est-ce que tu vois une autre solution à #67318 qui semble-t-il motivait la création de ce ticket ci ?

#8 - 22 juillet 2022 11:45 - Thomas Noël

Paul Marillonnet a écrit :

> Thomas Noël a écrit :
>
> > Dans 0002 le assert User.objects.count() 7 : c'est 6 qu'on veut (pas de nouveau compte, on doit rester à 6).
>
> Cette fois-ci on crée bien un utilisateur hors LDAP dans le test (avec une adresse qui n'a rien à voir avec les "etienne.michu@…" créés jusque là), ça en fait un en plus de ceux de la synchro. On ajoute ce même usager avec une casse différente dans le ldif et on vérifie qu'on est bien à 7 (et non pas à 8 si doublon il y avait eu).

Je vais retourner apprendre à lire :)

Je verrais bien un assert User.objects.count() 7 avant le users = list(ldap_backend.LDAPBackend.get_users()), qu'on voit bien que ça n'a pas bougé. Et aussi un test que len(users) est bien aussi à 7, un utilisateur de plus.

> > Mais je suis super ultra frileux sur la modif LDAPUser/User, ça me semble très impactant.
>
> Oui, changement de paradigme, on cherche à réconcilier les usagers issus de l'annuaire sur une base de comptes déjà existante dans Publik.

Voilà. Comme on a de la synchro LDAP à plein d'endroit, c'est bien ce changement de paradigme qu'il faut qu'on valide bien ensemble.

Aussi, si je comprends le code 0001, en cas de découverte d'un User qui a le même mail, alors on ne va pas mettre à jour cet utilisateur existant avec nom/prénom/etc venant du LDAP. Ce n'est pas vraiment ce qui est attendu/imaginé je pense : on préférerait que le compte User trouvé "devienne" un LDAPUser et qu'il soit dès lors synchronisé.

> Est-ce que tu vois une autre solution à #67318 qui semble-t-il motivait la création de ce ticket ci ?

Pour l'instant je propose un nettoyage du compte qui a été créé manuellement et qui "plante" la compréhension et le bon fonctionnement.

#9 - 22 juillet 2022 12:03 - Paul Marillonnet

- Statut changé de Solution proposée à En cours

Thomas Noël a écrit :

> Paul Marillonnet a écrit :
>
> > Thomas Noël a écrit :
> >
> > > Dans 0002 le assert User.objects.count() 7 : c'est 6 qu'on veut (pas de nouveau compte, on doit rester à 6).
> >
> > Cette fois-ci on crée bien un utilisateur hors LDAP dans le test (avec une adresse qui n'a rien à voir avec les "etienne.michu@…" créés jusque là), ça en fait un en plus de ceux de la synchro. On ajoute ce même usager avec une casse différente dans le ldif et on vérifie qu'on est bien à 7 (et non pas à 8 si doublon il y avait eu).
>
> Je vais retourner apprendre à lire :)
>
> Je verrais bien un assert User.objects.count() 7 avant le users = list(ldap_backend.LDAPBackend.get_users()), qu'on voit bien que ça n'a pas bougé. Et aussi un test que len(users) est bien aussi à 7, un utilisateur de plus.

Ok.

> > Mais je suis super ultra frileux sur la modif LDAPUser/User, ça me semble très impactant.
>
> Oui, changement de paradigme, on cherche à réconcilier les usagers issus de l'annuaire sur une base de comptes déjà existante dans Publik.
>
> Voilà. Comme on a de la synchro LDAP à plein d'endroit, c'est bien ce changement de paradigme qu'il faut qu'on valide bien ensemble.
>
> Aussi, si je comprends le code 0001, en cas de découverte d'un User qui a le même mail, alors on ne va pas mettre à jour cet utilisateur existant avec nom/prénom/etc venant du LDAP. Ce n'est pas vraiment ce qui est attendu/imaginé je pense : on préférerait que le compte User trouvé "devienne" un LDAPUser et qu'il soit dès lors synchronisé.

Ça pourrait être une nouvelle option du backend LDAP :)
Blague à part, je m'étais imaginé la précédence inverse, les comptes pré-existants ne devant pas être touchés par la synchro. Mais ça me va aussi avec cette priorité du LDAP sur le reste. Je vais revoir ma copie.

> Pour l'instant je propose un nettoyage du compte qui a été créé manuellement et qui "plante" la compréhension et le bon fonctionnement.

Ok.

#10 - 22 juillet 2022 18:10 - Benjamin Dauvergne

Le but ici c'est la réconciliation entre des comptes LDAP nouveaux et des comptes autocréés OIDCs existant (pour éviter la problème de poule et d'oeuf pour les villes raccordées en OIDC et LDAP), la recherche doit se faire dans l'OU visée par le LDAP sans considération pour l'état du raccordement au LDAP ou pas et bien sûr insensiblement à la casse (après on peut avoir des soucis de casse différente entre OIDC et LDAP, ben c'est nul, ça se mettra à jour à chaque fois, ce n'est pas l'objet du ticket).

#11 - 22 juillet 2022 18:37 - Thomas Noël

Benjamin Dauvergne a écrit :

> Le but ici c'est la réconciliation entre des comptes LDAP nouveaux et des comptes autocréés OIDCs existant (pour éviter la problème de poule et d'oeuf pour les villes raccordées en OIDC et LDAP), la recherche doit se faire dans l'OU visée par le LDAP sans considération pour l'état du raccordement au LDAP ou pas et bien sûr insensiblement à la casse (après on peut avoir des soucis de casse différente entre OIDC et LDAP,

> ben c'est nul, ça se mettra à jour à chaque fois, ce n'est pas l'objet du ticket).

On dirait qu'on est dans le même feeling donc.

Le problème que j'ai eu chez le client c'est la réconciliation avec des comptes créés manuellement avant la mise en place du LDAP : on peut considérer que c'est pareil qu'un compte créé précédemment via OIDC et donc, selon moi, on est dans un cas de figure équivalent.

Ce que je voudrais juste qu'on vérifie c'est que les comptes déjà créés (par OIDC ou autre) deviennent bien, lorsqu'il sont retrouvés selon le mail à l'occasion du sync-ldap, des comptes identifiables comme synchronisés en LDAP. Ca me semblerait logique et donc facile à comprendre.

(On s'écarte un poil un peu de l'objet du ticket qui parle de "sensible à la casse du mail", mais l'objectif reste bien de ne jamais provoquer la création de doublon)

#12 - 22 juillet 2022 19:31 - Benjamin Dauvergne

Thomas Noël a écrit :

> Ce que je voudrais juste qu'on vérifie c'est que les comptes déjà créés (par OIDC ou autre) deviennent bien, lorsqu'il sont retrouvés selon le mail à l'occasion du sync-ldap, des comptes identifiables comme synchronisés en LDAP. Ca me semblerait logique et donc facile à comprendre.

Oui c'est exactement ce qui se passe déjà, modulo le fait que ça n'ignore pas la casse et ça devrait (ma faute).

#13 - 22 juillet 2022 19:50 - Thomas Noël

Benjamin Dauvergne a écrit :

> Thomas Noël a écrit :
>
> > Ce que je voudrais juste qu'on vérifie c'est que les comptes déjà créés (par OIDC ou autre) deviennent bien, lorsqu'il sont retrouvés selon le mail à l'occasion du sync-ldap, des comptes identifiables comme synchronisés en LDAP. Ca me semblerait logique et donc facile à comprendre.
>
> Oui c'est exactement ce qui se passe déjà, modulo le fait que ça n'ignore pas la casse et ça devrait (ma faute).

Sauf erreur, même avec un mail totalement identique, il y a création d'un nouvel utilisateur. Ca semble lié (comme dit Paul) au fait qu'on recherche le lookup dans les LDAPUser et pas dans les User...?

#14 - 23 juillet 2022 15:20 - Benjamin Dauvergne

Thomas Noël a écrit :

> Benjamin Dauvergne a écrit :
>
> > Thomas Noël a écrit :
> >
> > > Ce que je voudrais juste qu'on vérifie c'est que les comptes déjà créés (par OIDC ou autre) deviennent bien, lorsqu'il sont retrouvés selon le mail à l'occasion du sync-ldap, des comptes identifiables comme synchronisés en LDAP. Ca me semblerait logique et donc facile à comprendre.
> >
> > Oui c'est exactement ce qui se passe déjà, modulo le fait que ça n'ignore pas la casse et ça devrait (ma faute).
>
> Sauf erreur, même avec un mail totalement identique, il y a création d'un nouvel utilisateur. Ca semble lié (comme dit Paul) au fait qu'on recherche le lookup dans les LDAPUser et pas dans les User...?

Je ne met pas en doute ce qui se passe, mais normalement LDAPUser est un modèle "proxy" donc il se comporte exactement comme User, sur mon instance locale publik.dev :

```
In [7]: User.objects.count()
django.db.backends DEBUG (0.001) SET search_path = authentic_dev_publik_love,public; args=None
django.db.backends DEBUG (0.001) SET application_name = authentic_dev_publik_love; args=None
django.db.backends DEBUG (0.004) SELECT COUNT(*) AS "__count" FROM "custom_user_user"; args=()
Out[7]: 10008

In [8]: LDAPUser.objects.count()
django.db.backends DEBUG (0.000) SET search_path = authentic_dev_publik_love,public; args=None
django.db.backends DEBUG (0.000) SET application_name = authentic_dev_publik_love; args=None
django.db.backends DEBUG (0.003) SELECT COUNT(*) AS "__count" FROM "custom_user_user"; args=()
```

```
Out[8]: 10008
```

C'est pareil, il y a certainement autre chose. J'ai regardé le ticket Nantes lié mais je n'y vois pas la configuration complète, donc je ne sais pas si ils ont joué avec les lookups, normalement la valeur actuelle fait tout bien comme il faut ('guid', 'external_id', 'username', 'email') mais si c'est modifié ça peut poser problème. Le mieux ce serait d'avoir un test qui reproduit le/les problèmes a minima (sans passer par OIDC, juste en créant d'autres utilisateurs avec le même mail, 1, plusieurs, dans différentes OUs).

#15 - 03 mars 2023 17:55 - Benjamin Dauvergne

Thomas Noël a écrit :

> Ci-joint une tentative qui ne fonctionne pas, le tox :
> [...]
>
> Donc : je n'arrive pas à faire ou correctif, ou le test, ou les deux...

Ton patch était bon et le comportement que vous avec compris aussi: ça ne cherche que dans les comptes pas encore reliés au même LDAP, effectivement si il y a un doublon du coté de l'annuaire LDAP on ne le résout pas, le premier compte qui tentera la liaison (par synchro ou au login) gagnera la liaison avec le compte existant.

Ton test ne marche pas car tu ne rajoutes pas de nouveau compte pour créer de liaison avec, il faut mettre une email quelconque avec des majuscules dans ton LDIF est créé un compte local avec la même email capitalisée autrement. Pas tenté de réutiliser l'EMAIL déjà présent, qui est déjà relié à un autre compte.

#16 - 03 mars 2023 17:55 - Benjamin Dauvergne

- Assigné à mis à Thomas Noël

#17 - 22 février 2024 11:27 - Benjamin Dauvergne

- Assigné à changé de Thomas Noël à Benjamin Dauvergne

#18 - 22 février 2024 11:38 - Robot Gitea

Benjamin Dauvergne (bdauvergne) a ouvert une pull request sur Gitea concernant cette demande :

- URL : https://git.entrouvert.org/entrouvert/authentic/pulls/258
- Titre : WIP: ignorer la casse lors de la recherche d'un nouvel utilisateur LDAP (#67600)
- Modifications : https://git.entrouvert.org/entrouvert/authentic/pulls/258/files

#19 - 22 février 2024 13:07 - Robot Gitea

- Statut changé de En cours à Solution proposée

#20 - 22 février 2024 13:09 - Benjamin Dauvergne

Je suis reparti du patch de Thomas ne voyant pas bien l'utilité du 0001 de Paul (et le changement LDAPUser -> User cassait pas mal de trucs). J'en ai profité pour faire du ménage dans test_get_users, pour bien séparer les cas testés, le cas générique et le cas casse de l'adresse de courriel utilisent maintenant la configuration par défaut de "lookups".

Au passage ça a changé l'ordre de remonté des utilisateurs et donc je l'ai rendu déterministe lorsqu'on cherche l'uuid du premier utilisateur dans test_sync_ldap_users.

#21 - 22 février 2024 14:26 - Robot Gitea

- Statut changé de Solution proposée à Solution validée

Serghei Mihai (smihai) a approuvé une pull request sur Gitea concernant cette demande :

- URL : https://git.entrouvert.org/entrouvert/authentic/pulls/258

#22 - 23 février 2024 20:18 - Robot Gitea

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

Benjamin Dauvergne (bdauvergne) a mergé une pull request sur Gitea concernant cette demande :

- URL : https://git.entrouvert.org/entrouvert/authentic/pulls/258
- Titre : ignorer la casse de l'adresse de courriel lors de la recherche d'un nouvel utilisateur LDAP (#67600)
- Modifications : https://git.entrouvert.org/entrouvert/authentic/pulls/258/files

#23 - 23 février 2024 21:23 - Transition automatique

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

Fichiers

0001-use-get_by_email-for-ldap-email-lookup-67600.patch	2,39 ko	21 juillet 2022	Thomas Noël
0002-use-get_by_email-for-ldap-email-lookup-67600.patch	2,21 ko	21 juillet 2022	Paul Marillonnet
0001-ldap-perform-extensive-user-lookup-regardless-of-the.patch	6,13 ko	21 juillet 2022	Paul Marillonnet