

Authentic 2 - Bug #27697

gestion des doublons LDAP selon casse

31 octobre 2018 10:17 - Frédéric Péters

Statut:	Fermé	Début:	31 octobre 2018
Priorité:	Haut	Echéance:	
Assigné à:	Sergei Mihai	% réalisé:	0%
Catégorie:		Temps estimé:	0:00 heure
Version cible:		Planning:	
Patch proposed:	Oui		

Description

Avec [#27147](#) on ne prête plus attention à la casse dans la recherche d'un utilisateur LDAP.

Comme on y prêtait attention avant, on peut se trouver maintenant avec des doublons et des crashes

```
File "/usr/lib/python2.7/dist-packages/authentic2/backends/ldap_backend.py", line 849, in lookup_existing_user
    return self.lookup_by_external_id(block, attributes)
File "/usr/lib/python2.7/dist-packages/authentic2/backends/ldap_backend.py", line 840, in lookup_by_external_id
    userexternalid__external_id__iexact=external_id, userexternalid__source=block['realm'])
File "/usr/lib/python2.7/dist-packages/django/db/models/query.py", line 338, in get
    (self.model._meta.object_name, num)
authentic2.backends.ldap_backend.MultipleObjectsReturned: get() returned more than one LDAPUser -- it returned 2!
```

Parfois ça concerne quelques utilisateurs mais parfois ça en concerne des centaines, quand il y a eu "nettoyage" de l'annuaire LDAP à un moment, pour retirer la casse partout, genre.

Révisions associées

Révision 4db66981 - 07 novembre 2018 13:57 - Sergei Mihai

ldap: don't crash on duplicated users (#27697)

Keep roles on the more recently used user, then delete the other ones'.

Historique

#1 - 31 octobre 2018 10:19 - Frédéric Péters

Je ne sais pas quelle approche on peut prendre ici, rendre le comportement optionnel ? faire un `.filter(...).first()` plutôt que le `.get(...)` ?

#2 - 31 octobre 2018 10:19 - Frédéric Péters

Tester d'abord en prêtant attention à la casse, et seulement si ça ne donne rien tenter sans ?

#3 - 31 octobre 2018 10:24 - Frédéric Péters

- Fichier `0001-ldap-lookup-users-with-exact-case-before-going-case-.patch` ajouté
- Fichier `0001-ldap-don-t-crash-on-duplicated-users-27697.patch` ajouté
- Statut changé de Nouveau à Solution proposée
- Patch proposed changé de Non à Oui

Brouillons de patch de ces dernières idées.

#4 - 31 octobre 2018 11:18 - Thomas Noël

Frédéric Péters a écrit :

Tester d'abord en prêtant attention à la casse, et seulement si ça ne donne rien tenter sans ?

J'aime bien cette idée là (donc `0001-ldap-lookup-users-with-exact-case-before-going-case-.patch` a ma préférence)

#5 - 31 octobre 2018 11:57 - Thomas Noël

Ceci dit l'erreur qu'on a c'est MultipleObjectsReturned : il faut éviter d'utiliser get, plutôt prendre l'élément le plus récent, selon la date de dernière connexion puis la date de création.

#6 - 03 novembre 2018 07:43 - Frédéric Péters

- Statut changé de Solution proposée à En cours
- Priorité changé de Normal à Haut
- Patch proposed changé de Oui à Non

Je mets une priorité à ce ticket parce que je ne pense pas que la situation actuelle puisse passer en production, qu'on doit convenir d'une approche. (qui peut très bien être ne rien faire ici et préemptivement retirer les doublons de la prod).

#7 - 05 novembre 2018 09:56 - Christophe Siraut

+1 pour faire une comparaison en 2 étapes (avec et sans casse).

#8 - 05 novembre 2018 11:48 - Benjamin Dauvergne

- Assigné à mis à Benjamin Dauvergne

#9 - 05 novembre 2018 11:48 - Benjamin Dauvergne

- Fichier 0001-Idap-don-t-crash-on-duplicated-users-27697.patch ajouté
- Statut changé de En cours à Solution proposée
- Patch proposed changé de Non à Oui

Autre proposition:

- on garde uniquement le plus récemment utilisé,
- on collecte le rôles de tous les users trouvés et on les affecte au premier,
- on supprime les autres.

#10 - 05 novembre 2018 11:51 - Frédéric Péters

```
u.delete()
```

→ other.delete().

#11 - 05 novembre 2018 11:53 - Serghei Mihai

- Assigné à Benjamin Dauvergne supprimé
- Patch proposed changé de Oui à Non

```
if len(user) > 1:
```

```
if len(users) > 1:
```

#12 - 05 novembre 2018 11:54 - Serghei Mihai

- Assigné à mis à Benjamin Dauvergne
- Patch proposed changé de Non à Oui

#13 - 05 novembre 2018 13:35 - Thomas Noël

Fonctionnellement je pense que ça passe (même si la magie me fait toujours un peu peur, mais c'est moi).

Passer le log.debug("found %d users ...") en log.info

Question sur le order_by(-last_login) : voir comment ça réagit quand il n'y a pas encore de last_login (parce que blank=True), typiquement pour les comptes qui viennent d'être synchronisés et sont donc ceux qu'il faut utiliser.

#14 - 05 novembre 2018 13:39 - Benjamin Dauvergne

<https://www.postgresql.org/docs/8.3/static/queries-order.html> :

The NULLS FIRST and NULLS LAST options can be used to determine whether nulls appear before or after non-null values in the sort ordering. By default, null values sort as if larger than any non-null value; that is, NULLS FIRST is the default for DESC order, and NULLS LAST otherwise.

Pas de nulls last dans Django, et je ne crois pas qu'on puisse trier par `__isnull` avant, enfin j'ai rien vu qui le confirme sur le net, donc oui faudra trier à la paluche après.

#15 - 05 novembre 2018 13:43 - Benjamin Dauvergne

- Fichier `0001-ldap-don-t-crash-on-duplicated-users-27697.patch` ajouté

Bêtises corrigées, tri en prenant compte des NULLs.

#16 - 05 novembre 2018 14:13 - Serghei Mihai

Il reste:

```
log.debug('found %d users, collectings roles into the first one and deleting the other one
s.',
len(users))
```

il faut un `len(users)`.

Et:

```
File "/home/serghei/dev/authentic/src/authentic2/backends/ldap_backend.py", line 844
key=lambda u: u is not None, u.last_login)
SyntaxError: non-keyword arg after keyword arg
```

#17 - 05 novembre 2018 15:16 - Serghei Mihai

- Fichier `0001-ldap-don-t-crash-on-duplicated-users-27697.patch` ajouté

Corrections.

#18 - 05 novembre 2018 15:22 - Thomas Noël

Je pense qu'il faut mettre les None en premier (utilisateur jamais connecté = compte tout nouveau, par ex. vient juste d'être synchronisé, et c'est celui ci qui est le bon pour l'avenir). Donc virer le `reserved=True` du `sorted`. IMveryveryHO seulement.

#19 - 05 novembre 2018 23:38 - Serghei Mihai

Thomas Noël a écrit :

Je pense qu'il faut mettre les None en premier (utilisateur jamais connecté = compte tout nouveau, par ex. vient juste d'être synchronisé, et c'est celui ci qui est le bon pour l'avenir). Donc virer le `reserved=True` du `sorted`. IMveryveryHO seulement.

En local avec un a2 qui tape dans une base Postgres je crée un nouvel utilisateur sans me connecter avec.

Ça donne:

```
In [29]: for u in User.objects.all().order_by('-last_login'):
...:     print u.username, u.last_login
...:
NeverLogguedUser None
neverloggueduser 2018-11-05 22:30:13.853561+00:00
```

J'ai bien l'impression qu'on aura celui qui ne s'est jamais connecté sans faire de tri manuel.

#20 - 06 novembre 2018 09:02 - Thomas Noël

Serghei Mihai a écrit :

J'ai bien l'impression qu'on aura celui qui ne s'est jamais connecté sans faire de tri manuel.

Comme a dit Benjamin en note 14, on n'en sait rien, ça dépend du sgbd, la requête SQL générée par Django ne précise rien (pour pgsq pas de "NULLS FIRST" ou "NULLS LAST" par exemple).

#21 - 06 novembre 2018 09:43 - Serghei Mihai

- Fichier 0001-ldap-don-t-crash-on-duplicated-users-27697.patch ajouté

Ok, tri avec le last_login à None toujours en premier. Et le log passé en info.

#22 - 06 novembre 2018 09:54 - Thomas Noël

Serghei Mihai a écrit :

Ok, tri avec le last_login à None toujours en premier. Et le log passé en info.

Ni l'un ni l'autre, sans doute un mauvais patch.

#23 - 06 novembre 2018 09:57 - Serghei Mihai

- Fichier 0001-ldap-don-t-crash-on-duplicated-users-27697.patch ajouté

Voici le bon.

#24 - 06 novembre 2018 10:15 - Thomas Noël

Arf, my bad, pour le coup le « users = sorted(users, key=lambda u: (u.last_login is not None, u.last_login)) » met les dates dans le mauvais sens. Il faut bien un reversed et donc remplacer is not par is.

#25 - 06 novembre 2018 10:16 - Thomas Noël

(et à force je me dis qu'il faudrait s'embêter à faire un test je pense :/)

#26 - 06 novembre 2018 11:06 - Benjamin Dauvergne

Thomas Noël a écrit :

Arf, my bad, pour le coup le « users = sorted(users, key=lambda u: (u.last_login is not None, u.last_login)) » met les dates dans le mauvais sens. Il faut bien un reversed et donc remplacer is not par is.

Hmm, trié True vient après False (1 > 0), donc ça :

- True, 2018-01-01
- False, None
- True, 2018-01-02

se trie en "reversed" (le tri par défaut c'est croissant, donc là c'est décroissant):

- True, 2018-01-02
- True, 2018-01-01
- False, None

N'est-ce pas ce que l'on veut ?

#27 - 06 novembre 2018 11:07 - Benjamin Dauvergne

```
In [1]: l = [  
...: (True, 2),  
...: (False, None),  
...: (True, 1)]  
  
In [2]: sorted(l, reverse=True)  
Out[2]: [(True, 2), (True, 1), (False, None)]
```

#28 - 06 novembre 2018 11:10 - Serghei Mihai

Benjamin Dauvergne a écrit :

Thomas Noël a écrit :

Arf, my bad, pour le coup le « users = sorted(users, key=lambda u: (u.last_login is not None, u.last_login)) » met les dates dans le mauvais sens. Il faut bien un reversed et donc remplacer is not par is.

Hmm, trié True vient après False (1 > 0), donc ça :

- True, 2018-01-01
- False, None
- True, 2018-01-02

Sauf si on change de u.last_login is not None en u.last_login is None.

#29 - 06 novembre 2018 14:12 - Benjamin Dauvergne

Serghei Mihai a écrit :

Benjamin Dauvergne a écrit :

Thomas Noël a écrit :

Arf, my bad, pour le coup le « users = sorted(users, key=lambda u: (u.last_login is not None, u.last_login)) » met les dates dans le mauvais sens. Il faut bien un reversed et donc remplacer is not par is.

Hmm, trié True vient après False (1 > 0), donc ça :

- True, 2018-01-01
- False, None
- True, 2018-01-02

Sauf si on change de u.last_login is not None en u.last_login is None.

Les valeurs que je donne sont bien avec u.last_login is not None il me semble.

#30 - 06 novembre 2018 14:18 - Serghei Mihai

Si on modifie la fonction lambda en u.last_login is None, avec le tri inverse les comptes qui ne se sont jamais connectés remontent en premier.

#31 - 06 novembre 2018 14:20 - Benjamin Dauvergne

Serghei Mihai a écrit :

Si on modifie la fonction lambda en u.last_login is None, avec le tri inverse les comptes qui ne se sont jamais connectés remontent en premier.

Mais comme il y a is not None dans mon code je ne vois pas le besoin de se poser la question.

#32 - 06 novembre 2018 14:32 - Frédéric Péters

Au cas où je suis toujours mentionné comme auteur alors que je n'en suis responsable que des premières lignes au tout début.

#33 - 06 novembre 2018 15:17 - Serghei Mihai

- Fichier 0001-Idap-don-t-crash-on-duplicated-users-27697.patch ajouté

Patch avec test.

#34 - 06 novembre 2018 18:34 - Benjamin Dauvergne

C'est is not None et pas is None ensuite le test ne semble pas tester le code en question vu qu'il n'y a pas de last_login de défini sur ces utilisateurs.

#35 - 06 novembre 2018 19:52 - Benjamin Dauvergne

Benjamin Dauvergne a écrit :

C'est is not None et pas is None ensuite le test ne semble pas tester le code en question vu qu'il n'y a pas de last_login de défini sur ces utilisateurs.

Erreur de ma part, il y a bien un test sur le last_login mais justement c'est le user avec last_login = None qui est conservé et on préfère conserver celui qui a éventuellement déjà été utilisé.

#36 - 06 novembre 2018 21:00 - Benjamin Dauvergne

- Assigné à Benjamin Dauvergne supprimé

Thomas Noël a écrit :

Je pense qu'il faut mettre les None en premier (utilisateur jamais connecté = compte tout nouveau, par ex. vient juste d'être synchronisé, et c'est celui ci qui est le bon pour l'avenir). Donc virer le reserved=True du sorted. IMveryveryHO seulement.

Pas d'accord le compte le plus ancien et le plus utile, la synchronisation sera refaite sur celui là, le code de synchronisation utilisant le même algorithme de recherche.

#39 - 06 novembre 2018 22:07 - Serghei Mihai

- Assigné à mis à Serghei Mihai

On a des cas inverses d'Alfortville: à Villeurbanne le compte le plus ancien est le bon (avec les bons rôles). Mais c'est une recette, on s'en fout. Allons sur la conservation du compte le plus récent.

Je prends et je mets à jour la branche.

#40 - 06 novembre 2018 22:10 - Benjamin Dauvergne

Serghei Mihai a écrit :

On a des cas inverses d'Alfortville: à Villeurbanne le compte le plus ancien est le bon (avec les bons rôles). Mais c'est une recette, on s'en fout. Allons sur la conservation du compte le plus récent.

Il n'y a pas de problèmes sur les rôles puisqu'on les garde tous.

#41 - 07 novembre 2018 10:06 - Serghei Mihai

Sur jenkins les tests échouent au niveau de RBAC:

```
> assert float(t) / 1000 < 0.008
E assert (8.29203200340271 / 1000) < 0.008
E + where 8.29203200340271 = float(8.29203200340271)
```

```
tests_rbac/test_rbac.py:107: AssertionError
```

mais tout est vert chez moi en local.

#42 - 07 novembre 2018 12:29 - Benjamin Dauvergne

Il devait manquer le commit [#27661](#) dans ta branche non ?

#43 - 07 novembre 2018 12:34 - Serghei Mihai

C'était ça. J'ai refait la branche est poussé.
Jenkins bosse.

#44 - 07 novembre 2018 12:49 - Serghei Mihai

C'est bon niveau jenkins.

#45 - 07 novembre 2018 13:48 - Benjamin Dauvergne

- Statut changé de Solution proposée à Solution validée

Ack.

#46 - 07 novembre 2018 14:00 - Serghei Mihai

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

```
commit 4db66981f007b16a2269fb1012a8b426e95ecaf4 (origin/master, origin/HEAD)
Author: Serghei Mihai <smihai@entrouvert.com>
Date: Tue Nov 6 15:15:26 2018 +0100
```

```
ldap: don't crash on duplicated users (#27697)
```

```
Keep roles on the more recently used user, then delete the other ones'.
```

#47 - 13 décembre 2018 22:44 - Benjamin Dauvergne

- Statut changé de Résolu (à déployer) à Fermé

Fichiers

0001-ldap-lookup-users-with-exact-case-before-going-case-.patch	1,58 ko	31 octobre 2018	Frédéric Péters
0001-ldap-don-t-crash-on-duplicated-users-27697.patch	1,54 ko	31 octobre 2018	Frédéric Péters
0001-ldap-don-t-crash-on-duplicated-users-27697.patch	2 ko	05 novembre 2018	Benjamin Dauvergne
0001-ldap-don-t-crash-on-duplicated-users-27697.patch	2,22 ko	05 novembre 2018	Benjamin Dauvergne
0001-ldap-don-t-crash-on-duplicated-users-27697.patch	2,15 ko	05 novembre 2018	Serghei Mihai
0001-ldap-don-t-crash-on-duplicated-users-27697.patch	2,15 ko	06 novembre 2018	Serghei Mihai
0001-ldap-don-t-crash-on-duplicated-users-27697.patch	2,16 ko	06 novembre 2018	Serghei Mihai
0001-ldap-don-t-crash-on-duplicated-users-27697.patch	3,65 ko	06 novembre 2018	Serghei Mihai