

Authentic 2 - Development #65411

auth_fc: locker le sub pendant les création de compte/liaison

19 mai 2022 07:56 - Benjamin Dauvergne

Statut:	En cours	Début:	19 mai 2022
Priorité:	Normal	Echéance:	
Assigné à:	Paul Marillonnet	% réalisé:	0%
Catégorie:		Temps estimé:	0:00 heure
Version cible:		Planning:	Non
Patch proposed:	Oui		
Description			
On le fait déjà pour l'email, mais pour éviter toute action concurrente on devrait aussi locker le sub, ça simplifiera le code normalement.			

Historique

#1 - 13 septembre 2022 16:00 - Paul Marillonnet

- Statut changé de Nouveau à En cours
- Assigné à mis à Paul Marillonnet

#2 - 13 septembre 2022 16:33 - Paul Marillonnet

- Fichier 0001-auth_fc-get-a-lock-on-the-sub-during-account-creatio.patch ajouté
- Statut changé de En cours à Solution proposée
- Patch proposed changé de Non à Oui

À discuter mais je ne pense pas qu'on puisse pour autant retirer les try: ... except sur les IntegrityError sur la liaison et la création, j'ai l'impression qu'ils proviennent de l'hypothèse de non-unicité des subs servis par FC auquel cas le verrou introduit ici ne peut pas s'y substituer.

#3 - 13 septembre 2022 16:37 - Paul Marillonnet

(La branche est basée sur le 0001 de [#65412](#).)

#4 - 13 septembre 2022 18:57 - Benjamin Dauvergne

- Statut changé de Solution proposée à En cours

En fait il faut remonter le atomic() (on ne peut verrouiller que dans une transaction, les tests sont trompeurs car ils tournent déjà dans une transaction, ici la méthode link() est cassée je pense plus haut dans handle_authorization_response et y gérer le lock ici, il sera unique.

Ensuite dans link, on peut maintenant faire un check explicite de l'existence du sub ou pas (et donc faudrait faire un lock aussi dans le code de suppression d'une liaison, mais là c'est du figolage), plutôt qu'utiliser IntegrityError comme indication de ce fait.

Dès qu'on s'assure d'avoir le lock dans tous les chemins, tous ces chemins peuvent être simplifiés en supposant qu'aucune exécution concurrente n'est possible, donc on peut faire tranquillement des SELECT sans se demander si les choses ne vont pas bouger sous nos pieds. C'est en ça que ça simplifie le code.

#5 - 14 septembre 2022 09:11 - Paul Marillonnet

- Fichier 0001-auth_fc-get-a-lock-on-the-sub-during-account-creatio.patch ajouté
- Statut changé de En cours à Solution proposée

Ok, pour la création de compte FC le try: ... except IntegrityError était là aussi pour attraper les violations de contrainte d'unicité sur le doublet (user, order) concernant l'objet FcAccount que l'on crée. Je ne me souviens plus pourquoi on tolère plusieurs liaisons FC pour un même utilisateurs, mais puisque c'est le cas il faut aussi un lock sur l'email de l'utilisateur concerné pour rétablir l'ordre des comptes déjà présents et créer la nouvelle liaison avec un order:=0.

#6 - 14 septembre 2022 09:12 - Paul Marillonnet

- Statut changé de Solution proposée à En cours

(Et il manque aussi la simplification sur le link(), je regarde.)

#7 - 14 septembre 2022 10:34 - Paul Marillonnet

- Fichier 0001-auth_fc-get-a-lock-on-the-sub-during-account-creatio.patch ajouté

- Statut changé de En cours à Solution proposée

Paul Marillonnet a écrit :

(Et il manque aussi la simplification sur le link(), je regarde.)

Ici, dans cette nouvelle version du patche.

Edit: et une correction pylint sur un import mort, et le retrait d'un commentaire obsolète "As we intercept IntegrityError [...]", présents dans la branche.

#8 - 20 octobre 2022 09:34 - Benjamin Dauvergne

- Statut changé de Solution proposée à En cours

Paul Marillonnet a écrit :

Ok, pour la création de compte FC le try: ... except IntegrityError était là aussi pour attraper les violations de contrainte d'unicité sur le doublet (user, order) concernant l'objet FcAccount que l'on crée. Je ne me souviens plus pourquoi on tolère plusieurs liaisons FC pour un même utilisateurs, mais puisque c'est le cas il faut aussi un lock sur l'email de l'usager concerné pour rétablir l'ordre des comptes déjà présents et créer la nouvelle liaison avec un order:=0.

On ne tolère pas plusieurs liaison, le order a été introduit pour conserver des liaisons existantes en double et pouvoir migrer vers la situation actuelle d'une seule liaison par sub et par utilisateur sans avoir à en supprimer au moment de cette migration. Donc plusieurs soucis :

```
+         if not models.FcAccount.objects.filter(sub=self.sub).exclude(user=request.user).count():
+             self.fc_account, created = models.FcAccount.objects.get_or_create(
+                 sub=self.sub,
+                 user=request.user,
+                 order=0,
+                 defaults={
+                     'token': json.dumps(self.token),
+                     'user_info': json.dumps(self.user_info),
+                 },
+             )
```

Ici la condition devrait être not FcAccount.objects.filter(Q(sub=sub) | Q(user=user)), pour justement éviter les doublons d'user ou de sub.

Ensuite on ne doit pas créer de liaison avec order>0 :

```
+         # reshuffle existing accounts order
+         Lock.lock_email(email=user.email)
+         for account in models.FcAccount.objects.filter(user=user).order_by('-order'):
+             account.order += 1
+             account.save()
```

Aussi tu introduis un filter_by_email() qui n'a pas de rapport avec ce ticket je pense.

Fichiers

0001-auth_fc-get-a-lock-on-the-sub-during-account-creatio.patch	1,12 ko	13 septembre 2022	Paul Marillonnet
0001-auth_fc-get-a-lock-on-the-sub-during-account-creatio.patch	6,19 ko	14 septembre 2022	Paul Marillonnet
0001-auth_fc-get-a-lock-on-the-sub-during-account-creatio.patch	8,1 ko	14 septembre 2022	Paul Marillonnet