

Hobo - Development #9396

Gérer les notifications de provisionnement hors requêtes, en deux temps

18 décembre 2015 15:52 - Benjamin Dauvergne

Statut:	Fermé	Début:	18 décembre 2015
Priorité:	Haut	Echéance:	
Assigné à:	Benjamin Dauvergne	% réalisé:	100%
Catégorie:		Temps estimé:	0:00 heure
Version cible:		Planning:	
Patch proposed:	Oui		
Description			
Dans des <code>pre_{save,delete,clear}</code> on stocke au niveau de la requête et/ou dans un contexte global (thread local) les objets qui ont changé, voir on peut vérifier ce qui a changé (diff entre le modèle en base et le modèle sauvé, pour éviter de provisionner les utilisateurs à chaque changement de <code>.last_login</code>).			
En fin de requêtes (via un middleware) ou en sortie de contexte (utilisation d'un <code>with differed_provisionning()</code>): pour les commandes comme <code>sync-ldap-users</code>) on balance la sauce si possible avec le moins de messages possibles.			
Demandes liées:			
Lié à Hobo - Bug #9293: ValueError: Problem installing fixture '/home/entrouv...		Fermé	10 décembre 2015
Lié à Publik - Bug #12957: 15 secondes pour la création d'un user à Tournai		Fermé	26 août 2016
Dupliqué par Hobo - Development #9643: Lors des modifications d'héritage tous...		Fermé	13 janvier 2016

Révisions associées

Révision ed08accd - 05 octobre 2016 14:03 - Benjamin Dauvergne

send provisioning messages after request treatment in a thread (fixes #9396)

All objects to provision are collected into the Provisionning singleton object in thread local dictionaries. When request processing is finished the ProvisionningMiddleware launch a thread which will send provisioning messages.

Historique

#1 - 18 décembre 2015 15:52 - Benjamin Dauvergne

- Lié à Bug #9293: ValueError: Problem installing fixture '/home/entrouvert/roles-to-prod.json': Cannot use None as a query value ajouté

#2 - 26 août 2016 10:03 - Frédéric Péters

- Lié à Bug #12957: 15 secondes pour la création d'un user à Tournai ajouté

#3 - 08 septembre 2016 11:18 - Frédéric Péters

- Priorité changé de Normal à Haut

C'est vraiment moche à Tournai (et Liège) (qui ont plus de champs que d'habitude dans les profils, pour des informations du registre national etc.).

#4 - 08 septembre 2016 11:28 - Benjamin Dauvergne

Ok je met ça sur le haut de la pile.

#5 - 13 septembre 2016 17:17 - Benjamin Dauvergne

- Fichier 0001-send-provisionning-messages-after-request-treatment-.patch ajouté

- Patch proposed changé de Non à Oui

#6 - 13 septembre 2016 17:19 - Benjamin Dauvergne

Ce n'est pas entièrement fini, il me reste à voir comment adapter la command `sync-ldap-users` pour qu'elle fonctionne avec ça, je pense ajouter des hooks et surcharger la commande dans l'application `hobo.agent.authentic2`.

#7 - 13 septembre 2016 17:21 - Benjamin Dauvergne

À noter que ça change le fonctionnement du provisionning des rôles, avant on envoyait systématiquement tous les rôles à chaque création d'un rôle

(full=True dans les messages). Maintenant on les envoie un par un, c'est un peu plus léger. Mais j'envisage d'ajouter une commande "provision-roles" au cas où.

#8 - 19 septembre 2016 13:44 - Frédéric Péters

Le fichier provisioning.py est absent du patch.

#9 - 20 septembre 2016 11:32 - Benjamin Dauvergne

- Fichier 0001-send-provisionning-messages-after-request-treatment-.patch ajouté

My bad.

#10 - 21 septembre 2016 13:54 - Thomas Noël

Benjamin Dauvergne a écrit :

Mais j'envisage d'ajouter une commande "provision-roles" au cas où.

Oui, ça sera bien utile, pour ne pas dire nécessaire, lors de l'initialisation d'une plateforme

#11 - 23 septembre 2016 09:33 - Frédéric Péters

D'abord un échec parce que le process_response ne renvoyait pas de réponse (corrigé en ajoutant un "return response" dedans) (je suis en django 1.8).

Ensuite, sur la modif d'un attribut de mon compte :

```
[2016-09-23 Fri 09:32:01] - - - ERROR hobo.agent.authentic2.provisionning.do_provision: error in provisioning
thread
Traceback (most recent call last):
  File "/home/fred/src/eo/hobo/hobo/agent/authentic2/provisionning.py", line 201, in do_provision
    self.notify_users(ous, saved.get(self.User, []))
  File "/home/fred/src/eo/hobo/hobo/agent/authentic2/provisionning.py", line 102, in notify_users
    for service, audience in self.get_audience(user):
  File "/home/fred/src/eo/hobo/hobo/agent/authentic2/provisionning.py", line 225, in get_audience
    qs = LibertyProvider.objects.filter(ou=ou)
  File "/home/fred/src/eo/venv/local/lib/python2.7/site-packages/django/db/models/manager.py", line 127, in ma
nager_method
    return getattr(self.get_queryset(), name)(*args, **kwargs)
  File "/home/fred/src/eo/venv/local/lib/python2.7/site-packages/django/db/models/query.py", line 679, in filt
er
    return self._filter_or_exclude(False, *args, **kwargs)
  File "/home/fred/src/eo/venv/local/lib/python2.7/site-packages/django/db/models/query.py", line 697, in _fil
ter_or_exclude
    clone.query.add_q(Q(*args, **kwargs))
  File "/home/fred/src/eo/venv/local/lib/python2.7/site-packages/django/db/models/sql/query.py", line 1310, in
add_q
    clause, require_inner = self._add_q(where_part, self.used_aliases)
  File "/home/fred/src/eo/venv/local/lib/python2.7/site-packages/django/db/models/sql/query.py", line 1338, in
_add_q
    allow_joins=allow_joins, split_subq=split_subq,
  File "/home/fred/src/eo/venv/local/lib/python2.7/site-packages/django/db/models/sql/query.py", line 1179, in
build_filter
    self.check_related_objects(field, value, opts)
  File "/home/fred/src/eo/venv/local/lib/python2.7/site-packages/django/db/models/sql/query.py", line 1074, in
check_related_objects
    self.check_query_object_type(value, opts)
  File "/home/fred/src/eo/venv/local/lib/python2.7/site-packages/django/db/models/sql/query.py", line 1058, in
check_query_object_type
    (value, opts.object_name)
ValueError: Cannot query "fred (fred)": Must be "OrganizationalUnit" instance.
```

#12 - 23 septembre 2016 10:56 - Benjamin Dauvergne

J'ai un patch mais j'essaie de voir pourquoi les tests passent quand même d'abord.

#13 - 23 septembre 2016 11:46 - Benjamin Dauvergne

Et donc ça passe parce que pour Django 1.7 tout ce qui a un champ .pk ou .id est un modèle, il ne regarde pas le type dans les filtres, et les tests Django 1.8 ne marchait pas parce qu'hobo a encore une contrainte < 1.8. Est-ce que quelqu'un voit un inconvénient à ce que je relâche cette contrainte pour autoriser hobo et Django 1.8 ?

#14 - 23 septembre 2016 16:24 - Frédéric Péters

Je tourne en hobo et django 1.8 depuis longtemps. (et montpellier aussi sur de la prod). Pas de problème pour moi donc.

#15 - 24 septembre 2016 10:39 - Frédéric Péters

(correction, j'étais resté sur l'idée que jessie était en 1.8 mais non, donc à part en local on n'a pas de 1.8 qui tourne vraiment toutes les briques)

#16 - 26 septembre 2016 11:15 - Benjamin Dauvergne

- Fichier 0001-send-provisionnement-messages-after-request-treatment-.patch ajouté

Rebasé sur les derniers commits, tourne en 1.7 et 1.8.

#17 - 27 septembre 2016 09:12 - Frédéric Péters

Le message est encore envoyé une fois par "audience"; le provisionning des utilisateurs pourrait avoir lieu comme celui des rôles, où audience reprend la liste des services ?

#18 - 27 septembre 2016 10:30 - Benjamin Dauvergne

J'ai un souci pour l'attribut is_superuser qui est spécifique au service et j'ai choisi la facilité.

Ce que je peux faire c'est vérifier si l'utilisateur a un is_superuser, si ce n'est pas le cas je le provisionne en une fois vers toutes les audiences, sinon j'utilise la méthode actuelle. Ça ira vite pour les usagers et plus lentement pour les agents, ok ?

#19 - 27 septembre 2016 10:35 - Frédéric Péters

Yep, surtout que les agents is_superuser, ça ne doit même pas être la majorité.

#20 - 04 octobre 2016 15:44 - Benjamin Dauvergne

- Fichier 0001-send-provisionnement-messages-after-request-treatment-.patch ajouté

Pas encore super chaud pour pousser y a pas de test des fonctionnalités nouvelles, i.e. le fait de pousser tous les utilisateurs d'un coup quand qu'ils n'ont pas d'attribut de rôle (je n'ai pas cherché à faire plus compliquer pour l'instant, si il y a dans le tas un utilisateur de ce type, technique lente pour tous, sinon technique rapide broadcast). Mais tu peux le tester de ton côté voir si ça diminue les messages.

#21 - 04 octobre 2016 18:12 - Frédéric Péters

Le commentaire dans le patch,

1. Find roles giving a superuser attribute
2. Split users between those with these roles and those without

Est trompeur; comme tu l'écris dans le ticket, ça fait "qu'ils n'ont pas d'attribut de rôle", pas le côté "superuser". Et du coup, dès qu'un usager a un rôle, et c'est fréquent (genre Alfortville les usagers ont un rôle "Parent"), on bascule sur le mode lent.

Pourquoi ne pas faire le filtre sur attribute__name="is_superuser" / __value="true", pour correspondre au code qui sera effectif plus loin :

```
for attribute in role.attributes.all():
    if attribute.name == 'is_superuser' and attribute.value == 'true':
        role_is_superuser = True
```

?

#22 - 05 octobre 2016 12:25 - Benjamin Dauvergne

- Fichier 0001-send-provisionnement-messages-after-request-treatment-.patch ajouté

Test complet et validation que pour l'attribut de role emails c'est bien le cas général qui est utilisé.

#23 - 05 octobre 2016 14:03 - Benjamin Dauvergne

- Fichier 0001-send-provisionnement-messages-after-request-treatment-.patch ajouté

Le bon patch.

#24 - 05 octobre 2016 17:05 - Frédéric Péters

Voilà, je l'ai un petit peu utilisé en local, pour moi c'est ok à pousser.

#25 - 05 octobre 2016 18:15 - Benjamin Dauvergne

- Statut changé de Nouveau à Résolu (à déployer)

- % réalisé changé de 0 à 100

Appliqué par commit [ed08accd602ef7af5fb4e7d9c874240e318013a5](#).

#26 - 23 décembre 2018 16:06 - Frédéric Péters

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

#27 - 29 janvier 2020 18:50 - Benjamin Dauvergne

- Dupliqué par Development #9643: Lors des modifications d'héritage tous les utilisateurs membres du rôle fils doivent être reprovisionnés ajouté

Fichiers

0001-send-provisionnement-messages-after-request-treatment-.patch	25,2 ko	13 septembre 2016	Benjamin Dauvergne
0001-send-provisionnement-messages-after-request-treatment-.patch	37,3 ko	20 septembre 2016	Benjamin Dauvergne
0001-send-provisionnement-messages-after-request-treatment-.patch	38,1 ko	26 septembre 2016	Benjamin Dauvergne
0001-send-provisionnement-messages-after-request-treatment-.patch	41,5 ko	04 octobre 2016	Benjamin Dauvergne
0001-send-provisionnement-messages-after-request-treatment-.patch	41,5 ko	05 octobre 2016	Benjamin Dauvergne
0001-send-provisionnement-messages-after-request-treatment-.patch	48,2 ko	05 octobre 2016	Benjamin Dauvergne