

Hobo - Development #72478

sortir le provisionning de l'opération de migration (?)

15 décembre 2022 12:02 - Paul Marillonnet

Statut:	Fermé	Début:	15 décembre 2022
Priorité:	Normal	Echéance:	
Assigné à:	Benjamin Dauvergne	% réalisé:	0%
Catégorie:		Temps estimé:	0:00 heure
Version cible:		Planning:	Non
Patch proposed:	Oui		

Description

Actuellement certaines migrations semblent ralenties parce que s'interpose du provisionning lors de l'exécution de la migration. Est-ce qu'on ne pourrait pas tout simplement faire des migrations sans provisionning, que celui-ci ait lieu dans un autre temps, qui ne soit pas celui du apt upgrade lançant les migrations etc.

Révisions associées

Révision a917ec7f - 04 janvier 2023 10:54 - Benjamin Dauvergne

agent: limit provisionning to some commands (#72478)

- authentic2.custom_user.management.commands.fix-attributes
- authentic2.management.commands.check-and-repair
- authentic2.management.commands.clean-unused-accounts
- authentic2.management.commands.cleanupauthentic
- authentic2.management.commands.deactivate-orphaned-ldap-users
- authentic2.management.commands.import_site
- authentic2.management.commands.sync-ldap-users
- authentic2_auth_oidc.management.commands.oidc-sync-provider
- hobo.multitenant.management.commands.runscript

Historique

#2 - 16 décembre 2022 09:16 - Benjamin Dauvergne

- *Projet changé de Authentic 2 à Hobo*

Le code de tout ça est dans hobo, c'est à gérer là.

#3 - 16 décembre 2022 09:17 - Benjamin Dauvergne

- *Assigné à mis à Benjamin Dauvergne*

#4 - 16 décembre 2022 09:29 - Benjamin Dauvergne

- *Fichier 0001-agent-limit-provisionning-to-some-commands-72478.patch ajouté*
- *Statut changé de Nouveau à Solution proposée*
- *Patch proposed changé de Non à Oui*

#5 - 16 décembre 2022 10:59 - Benjamin Dauvergne

- *Fichier 0001-agent-limit-provisionning-to-some-commands-72478.patch ajouté*

#6 - 20 décembre 2022 11:58 - Valentin Deniaud

Un patch qui aurait juste désactivé le provisionning sur migrate_schemas aurait paru plus sûr (ok le code est mieux après le patch car l'action est explicite, par contre le patch en lui-même est 0 explicite et s'en remet à l'imagination du lecteur pour comprendre quelles commandes sont impactées).

À part ça j'aimerais bien un cours de rattrapage sur le provisionning, il ne se fait plus pendant la migration et du coup il se fait quand ?

#7 - 20 décembre 2022 12:09 - Benjamin Dauvergne

Valentin Deniaud a écrit :

Un patch qui aurait juste désactivé le provisionning sur migrate_schemas aurait paru plus sûr (ok le code est mieux après le patch car l'action est explicite, par contre le patch en lui-même est 0 explicite et s'en remet à l'imagination du lecteur pour comprendre quelles commandes sont

impactées).

Au contraire tu as la liste explicite des commandes impactées, je ne comprends pas cette remarque.

À part ça j'aimerais bien un cours de rattrapage sur le provisionning, il ne se fait plus pendant la migration et du coup il se fait quand ?

Jamais. Si jamais il y avait besoin de provisionning dans une migration, il faut le prévoir explicitement, idem dans un shell. Pour runscript le provisionning est maintenu.

#8 - 20 décembre 2022 12:17 - Valentin Deniaud

Benjamin Dauvergne a écrit :

Au contraire tu as la liste explicite des commandes impactées, je ne comprends pas cette remarque.

J'ai la liste des commandes qui ne sont *pas* impactées *par le patch*, c'est à dire celles dont le fonctionnement va rester le même. Je n'ai pas la liste des commandes dont le comportement va être modifié.

#9 - 20 décembre 2022 12:21 - Benjamin Dauvergne

Valentin Deniaud a écrit :

Benjamin Dauvergne a écrit :

Au contraire tu as la liste explicite des commandes impactées, je ne comprends pas cette remarque.

J'ai la liste des commandes qui ne sont *pas* impactées *par le patch*, c'est à dire celles dont le fonctionnement va rester le même. Je n'ai pas la liste des commandes dont le comportement va être modifié.

Ok c'est bizarre mais tu considères qu'avoir le provisionning est le fonctionnement normal des commandes de Django mais ce n'est pas le cas. Là tu as la liste explicite des commandes pour lesquelles du provisionning sera fait, comme le problème c'est d'avoir du provisionning et non pas de ne pas en avoir ça me paraît ok. Le provisionning est en premier lieu fait pour avoir lieu au cours d'une requête pour des modifications faites en front ou back office. L'introduction du provisionning au niveau des commandes a été fait principalement pour la commande de synchro LDAP et les scripts, c'est maintenu.

#10 - 20 décembre 2022 17:24 - Valentin Deniaud

Benjamin Dauvergne a écrit :

Ok c'est bizarre mais tu considères qu'avoir le provisionning est le fonctionnement normal des commandes de Django

... Je dis juste que le patch ne se limite pas à corriger le problème du ticket, je ne dis même pas que c'est mal je dis que ça m'aurait « paru plus sûr ».

Peut-être que ça ne m'aurait pas donné cette impression s'il y avait eu des tests.

Bref revenons-en au fond,

il ne se fait plus pendant la migration et du coup il se fait quand ?

Jamais.

J'ai écrit une migration comme celle du ticket lié, zappé qu'il fallait prévoir le provisionning à l'avance, les rôles sont créés. Quelle sera la marche à suivre pour remettre les choses d'équerre ?

#11 - 20 décembre 2022 17:41 - Benjamin Dauvergne

Valentin Deniaud a écrit :

J'ai écrit une migration comme celle du ticket lié, zappé qu'il fallait prévoir le provisionning à l'avance, les rôles sont créés. Quelle sera la marche à suivre pour remettre les choses d'équerre ?

Le provisionning qui a lieu dans les migrations authentic est inutile, ces rôles d'administration ne servent à rien dans les services et ne changent rien dans les workflows. Le seul provisionning utile concerne les rôles utilisés dans des workflows ou des formulaires pour l'envoi des mails de notification et la mise à jour de l'adresse mail elle-même (c'est le seul cas où si un provisionning n'a pas lieu on aura un souci), le reste ne sert à rien ou n'a pas d'impact signifiant (genre changer nom/prénom d'un utilisateur pour l'affichage, c'est pas signifiant et ça se corrigera tout seul au prochain login).

Pour l'instant le code de provisionning est trop simple pour faire ces distinctions et se déclenche à la moindre modification du graphe des rôles ou d'un utilisateur. Dans le même ordre d'idée le provisionning vers les briques autre que combo et w.c.s. est inutile. Et sur combo le provisionning de l'appartenance aux rôles ne sert à rien non plus.

On a surtout encore un gros manque de provisionning en pull depuis w.c.s. et combo (une bête synchro) qui rendrait ces discussions encore moins utiles.

#12 - 03 janvier 2023 16:06 - Valentin Deniaud

- Statut changé de Solution proposée à Solution validée

Benjamin Dauvergne a écrit :

ça se corrigera tout seul au prochain login

Voilà ça ça me rassure, go

#13 - 04 janvier 2023 10:55 - Benjamin Dauvergne

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

```
commit a917ec7fec766089d265b6cf455ca0dcc8b835e0
Author: Benjamin Dauvergne <bdauvergne@entrouvert.com>
Date: Fri Dec 16 09:27:52 2022 +0100
```

```
agent: limit provisionning to some commands (#72478)
```

```
- authentic2.custom_user.management.commands.fix-attributes
- authentic2.management.commands.check-and-repair
- authentic2.management.commands.clean-unused-accounts
- authentic2.management.commands.cleanupauthentic
- authentic2.management.commands.deactivate-orphaned-ldap-users
- authentic2.management.commands.import_site
- authentic2.management.commands.sync-ldap-users
- authentic2_auth_oidc.management.commands.oidc-sync-provider
- hobo.multitenant.management.commands.runscript
```

#14 - 04 janvier 2023 20:14 - Transition automatique

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

#15 - 12 mars 2023 04:42 - Transition automatique

Automatic expiration

Fichiers

0001-agent-limit-provisionning-to-some-commands-72478.patch	2,93 ko	16 décembre 2022	Benjamin Dauvergne
0001-agent-limit-provisionning-to-some-commands-72478.patch	3,02 ko	16 décembre 2022	Benjamin Dauvergne