

Hobo - Bug #31227

import-wcs-roles : value violates unique constraint

08 mars 2019 16:15 - Thomas Noël

Statut:	Rejeté	Début:	08 mars 2019
Priorité:	Normal	Echéance:	
Assigné à:	Thomas Noël	% réalisé:	0%
Catégorie:		Temps estimé:	0:00 heure
Version cible:		Planning:	
Patch proposé:	Oui		

Description

Dans l'idée de remonter les rôles d'un w.c.s. existant vers Authentic (pour une migration auqu→publik), je me propose d'utiliser import-wcs-roles

Sur le tenant de l'authentic cible, je pose un HOBO_ROLE_EXPORT=True

Mais j'ai cette erreur lorsque je lance le manage.py import-wcs-roles :

```
$ authentic-multitenant-manage import-wcs-roles --traceback
```

Traceback (most recent call last):

```
File "/home/thomas/dev/publik/venv/bin/authentic2-ctl", line 7, in <module>
    exec(compile(f.read(), __file__, 'exec'))
File "/home/thomas/dev/publik/src/authentic/authentic2-ctl", line 21, in <module>
    execute_from_command_line(sys.argv[:1] + argv)
File "/home/thomas/dev/publik/venv/local/lib/python2.7/site-packages/django/core/management/__init__.py", line 364, in execute_from_command_line
    utility.execute()
File "/home/thomas/dev/publik/venv/local/lib/python2.7/site-packages/django/core/management/__init__.py", line 356, in execute
    self.fetch_command(subcommand).run_from_argv(self.argv)
File "/home/thomas/dev/publik/venv/local/lib/python2.7/site-packages/django/core/management/base.py", line 283, in run_from_argv
    self.execute(*args, **cmd_options)
File "/home/thomas/dev/publik/src/hobo/hobo/agent/authentic2/apps.py", line 45, in new_execute
    return old_execute(self, *args, **kwargs)
File "/home/thomas/dev/publik/venv/local/lib/python2.7/site-packages/django/core/management/base.py", line 330, in execute
    output = self.handle(*args, **options)
File "/home/thomas/dev/publik/src/hobo/hobo/agent/authentic2/management/commands/import-wcs-roles.py", line 172, in handle
    self.handle_tenant(tenant, **options)
File "/home/thomas/dev/publik/src/hobo/hobo/agent/authentic2/management/commands/import-wcs-roles.py", line 202, in handle_tenant
    importer.import_roles()
File "/home/thomas/dev/publik/src/hobo/hobo/agent/authentic2/management/commands/import-wcs-roles.py", line 61, in import_roles
    self.create_role(role_tpl)
File "/home/thomas/dev/publik/src/hobo/hobo/agent/authentic2/management/commands/import-wcs-roles.py", line 81, in create_role
    defaults=defaults)
File "/home/thomas/dev/publik/venv/local/lib/python2.7/site-packages/django/db/models/manager.py", line 85, in manager_method
    return getattr(self.get_queryset(), name)(*args, **kwargs)
File "/home/thomas/dev/publik/venv/local/lib/python2.7/site-packages/django/db/models/query.py", line 466, in get_or_create
    return self._create_object_from_params(lookup, params)
File "/home/thomas/dev/publik/venv/local/lib/python2.7/site-packages/django/db/models/query.py", line 509, in _create_object_from_params
    six.reraise(*exc_info)
File "/home/thomas/dev/publik/venv/local/lib/python2.7/site-packages/django/db/models/query.py", line 500, in _create_object_from_params
```

```

obj = self.create(**params)
File "/home/thomas/dev/publik/venv/local/lib/python2.7/site-packages/django/db/models/query.py",
line 394, in create
    obj.save(force_insert=True, using=self.db)
File "/home/thomas/dev/publik/src/authentic/src/authentic2/a2_rbac/models.py", line 209, in save
    return super(Role, self).save(*args, **kwargs)
File "/home/thomas/dev/publik/src/authentic/src/django_rbac/models.py", line 60, in save
    return super(AbstractBase, self).save(*args, **kwargs)
File "/home/thomas/dev/publik/venv/local/lib/python2.7/site-packages/django/db/models/base.py",
line 808, in save
    force_update=force_update, update_fields=update_fields)
File "/home/thomas/dev/publik/venv/local/lib/python2.7/site-packages/django/db/models/base.py",
line 838, in save_base
    updated = self._save_table(raw, cls, force_insert, force_update, using, update_fields)
File "/home/thomas/dev/publik/venv/local/lib/python2.7/site-packages/django/db/models/base.py",
line 924, in _save_table
    result = self._do_insert(cls._base_manager, using, fields, update_pk, raw)
File "/home/thomas/dev/publik/venv/local/lib/python2.7/site-packages/django/db/models/base.py",
line 963, in _do_insert
    using=using, raw=raw)
File "/home/thomas/dev/publik/venv/local/lib/python2.7/site-packages/django/db/models/manager.py
", line 85, in manager_method
    return getattr(self.get_queryset(), name)(*args, **kwargs)
File "/home/thomas/dev/publik/venv/local/lib/python2.7/site-packages/django/db/models/query.py",
line 1079, in _insert
    return query.get_compiler(using=using).execute_sql(return_id)
File "/home/thomas/dev/publik/venv/local/lib/python2.7/site-packages/django/db/models/sql/compil
er.py", line 1112, in execute_sql
    cursor.execute(sql, params)
File "/home/thomas/dev/publik/venv/local/lib/python2.7/site-packages/django/db/backends/utils.py
", line 79, in execute
    return super(CursorDebugWrapper, self).execute(sql, params)
File "/home/thomas/dev/publik/venv/local/lib/python2.7/site-packages/django/db/backends/utils.py
", line 64, in execute
    return self.cursor.execute(sql, params)
File "/home/thomas/dev/publik/venv/local/lib/python2.7/site-packages/django/db/utils.py", line 9
4, in __exit__
    six.reraise(dj_exc_type, dj_exc_value, traceback)
File "/home/thomas/dev/publik/venv/local/lib/python2.7/site-packages/django/db/backends/utils.py
", line 64, in execute
    return self.cursor.execute(sql, params)
django.db.utils.IntegrityError: duplicate key value violates unique constraint "a2_rbac_role_uniqu
e_idx_2"
DETAIL:  Key (ou_id, slug)=(1, debug-support-entrouvert) already exists.

```

Ça crashe donc sur hobo/agent/authentic2/management/commands/import-wcs-roles.py à ce niveau :

```

70     def create_role(self, role_tpl):
71         defaults = {
72             'name': role_tpl.name,
73             # w.c.s. will always provide a slug but for other services we do
74             # not know
75             'slug': role_tpl.slug or slugify(role_tpl.name),
76         }
77         # search role by external id, create if not found
78         role, created = Role.objects.get_or_create(
79             ou=self.service.ou,
80             external_id=role_tpl.external_id,
81             defaults=defaults) <----- ici

```

Une idée de la raison ?

(note: oui, le rôle existe bien déjà, mais normalement, le get_or_create devrait juste faire un get ?...)

Historique

#1 - 08 mars 2019 16:22 - Frédéric Péters

(note: oui, le rôle existe bien déjà, mais normalement, le `get_or_create` devrait juste faire un `get` ?...)

Oui mais sur ces critères :

```
79         ou=self.service.ou,  
80         external_id=role_tpl.external_id,
```

qui ne doivent pas matcher, alors il fait le `create`.

Vu l'unicité demandée, le problème n'est pas sur l'OU mais sur le `external_id`.

#2 - 08 mars 2019 16:38 - Benjamin Dauvergne

Je dirai que ça pourrait mieux marcher :

```
70     def create_role(self, role_tpl):  
71         # search role by external id, create if not found  
72         role, created = Role.objects.get_or_create(  
73             ou=self.service.ou,  
74             name=role_tpl.name, defaults={'slug': role_tpl.slug})
```

#3 - 11 mars 2019 09:56 - Thomas Noël

- Fichier `0001-import-wcs-roles-use-slug-as-an-key-identifier-31227.patch` ajouté
- Statut changé de *Nouveau* à *Solution proposée*
- Patch proposé changé de *Non* à *Oui*

J'ai tâtonné et relu du code, pour finalement comprendre que `Role.external_id` était apparu pour cette commande ([#7299](#)) et ne me semble plus du tout utilisé ailleurs, en fait... peut-être un nettoyage à faire ; ou pas.

L'unicité réelle étant sur `ou` et `slug`, qui sont invariants sur `w.c.s.`, je préfère m'appuyer sur ces deux là pour faire le `get_or_create`. D'où le patch ci-joint, testé "à la main" sur mon `Publik` local.

A confirmer puis à noter quelque part : aujourd'hui `import-wcs-roles` n'est plus utilisé en tant que système de synchro régulière des rôles. Il reste juste un outil pour faire un import "one shot" des rôles de `wcs` vers `Authentic` lors d'une migration d'un `wcs` seul vers un `Publik` (`wcs` appuyé sur `Authentic`).

#4 - 11 mars 2019 10:28 - Benjamin Dauvergne

Thomas Noël a écrit :

J'ai tâtonné et relu du code, pour finalement comprendre que `Role.external_id` était apparu pour cette commande ([#7299](#)) et ne me semble plus du tout utilisé ailleurs, en fait... peut-être un nettoyage à faire ; ou pas.

Oui à l'époque `w.c.s.` était la source des rôles et pour garder le lien on se basait sur le `role.slug` ou le nom "slugifié" venant de `w.c.s.`.

L'unicité réelle étant sur `ou` et `slug`, qui sont invariants sur `w.c.s.`, je préfère m'appuyer sur ces deux là pour faire le `get_or_create`. D'où le patch ci-joint, testé "à la main" sur mon `Publik` local.

Tu as perdu de vue ton objectif qui était de récupérer les rôles par leur nom au minimum sur un one-shot, tu t'assignes comme objectif de maintenir l'unicité comme si tu allais lancer cette commande régulièrement en cas de changement de nom, ce qui ne vas pas être le cas, je pense que tu aurais juste du faire ta recherche par `ou` et nom comme je le proposais, il faut juste espérer ici que tous les slugs correspondent, sinon tu aurais des doublons sur les noms.

A confirmer puis à noter quelque part : aujourd'hui `import-wcs-roles` n'est plus utilisé en tant que système de synchro régulière des rôles. Il reste juste un outil pour faire un import "one shot" des rôles de `wcs` vers `Authentic` lors d'une migration d'un `wcs` seul vers un `Publik` (`wcs` appuyé sur `Authentic`).

C'est pour ça que se baser uniquement sur le nom me paraît plus utile (après si on a une collision sur nom différent / slug identique, on la règle à la main, faut juste la rapporter proprement, je pense) à moins qu'on soit certain que la slugification des deux cotés donne toujours la même chose.

#5 - 11 mars 2019 10:28 - Benjamin Dauvergne

- Assigné à *mis* à *Thomas Noël*

#6 - 11 mars 2019 12:09 - Thomas Noël

- Statut changé de Solution proposée à Rejeté

Benjamin Dauvergne a écrit :

L'unicité réelle étant sur ou et slug, qui sont invariants sur w.c.s., je préfère m'appuyer sur ces deux là pour faire le get_or_create. D'où le patch ci-joint, testé "à la main" sur mon Publik local.

Tu as perdu de vue ton objectif qui était de récupérer les rôles par leur nom au minimum sur un one-shot, tu t'assignes comme objectif de maintenir l'unicité comme si tu allais lancer cette commande régulièrement en cas de changement de nom, ce qui ne vas pas être le cas, je pense que tu aurais juste du faire ta recherche par ou et nom comme je le proposais, il faut juste espérer ici que tous les slugs correspondent, sinon tu aurais des doublons sur les noms.

A confirmer puis à noter quelque part : aujourd'hui import-wcs-roles n'est plus utilisé en tant que système de synchro régulière des rôles. Il reste juste un outil pour faire un import "one shot" des rôles de wcs vers Authentic lors d'une migration d'un wcs seul vers un Publik (wcs appuyé sur Authentic).

C'est pour ça que se baser uniquement sur le nom me paraît plus utile (après si on a une collision sur nom différent / slug identique, on la règle à la main, faut juste la rapporter proprement, je pense) à moins qu'on soit certain que la slugification des deux cotés donne toujours la même chose.

J'ai définitivement rien compris, je retire ce patch ; de toute façon je vais utiliser cet import-wcs-roles sur un Authentic vierge donc il n'y aura pas de conflit.

Fichiers

0001-import-wcs-roles-use-slug-as-an-key-identifiant-31227.patch	1,67 ko	11 mars 2019	Thomas Noël
--	---------	--------------	-------------